Epigenetics of complex traits and diseases

Charles E. Breeze

UCL Cancer Institute
University College London

This thesis is submitted for the degree of Doctor of Philosophy
DECLARATION

I, Charles Edmund Breeze, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Charles E. Breeze

29 March 2017
The following publications have resulted from the work presented in this thesis and conducted through collaboration on other projects. The corresponding abstracts and title pages can be found in the Appendices.


ABSTRACT

Thousands of genetic and epigenetic variants have been identified for many common diseases including cancer through genome-wide association studies (GWAS) and epigenome-wide association studies (EWAS). To advance the complex interpretation of both GWAS and EWAS results, I developed new software tools (FORGE2 and eFORGE) for the analysis and interpretation of GWAS and EWAS data, respectively. Both tools determine the cell type-specific regulatory component of a set of target regions (either GWAS-identified genetic variants or EWAS-identified differentially methylated positions). This is achieved by detecting enrichment of overlap with histone mark peaks or DNase I hypersensitive sites across hundreds of tissues, primary cell types, and cell lines from the ENCODE, Roadmap Epigenomics, and BLUEPRINT projects. Application of both tools to publicly available datasets identified novel disease-relevant cell types for many common diseases, a stem cell-like signature in cancer EWAS, and also demonstrated the ability to detect cell-composition effects for EWAS performed on heterogeneous tissues. To complement these bioinformatics efforts and validate selected variants predicted by FORGE2, eFORGE and additional analyses, I performed conformation capture using 4C-seq to fine-map the 3D context of the genomic regions involved, uncovering novel interactions for autoimmunity-associated variants and IKZF3.
ACKNOWLEDGEMENTS

The majority of the results presented in chapter 3, relating to the eFORGE software tool, have been published in Cell Reports\textsuperscript{146}. The corresponding title page, journal cover and abstract can be found in the Appendices. I developed eFORGE during a secondment at the European Bioinformatics Institute (EBI), in collaboration with Dr Ian Dunham, Dr Valentina Iotchkova and Prof. Ewan Birney, and also in collaboration with QMUL researcher Dr Robert Lowe and UCL researchers Dr James Barrett, Dr Javier Herrero, Dr Dirk Paul, Prof. Andrew Teschendorff and Prof. Stephan Beck. Notably, the web interface for eFORGE was developed by Dr Javier Herrero at the Bill Lyons Informatics Centre at the UCL Cancer Institute.

I developed the FORGE2 software tool with advice from Dr Ian Dunham (EBI) and incorporating some eFORGE code improvements introduced by Dr Javier Herrero (UCL Cancer Institute).

I performed the 4C-seq experiments on disease-associated loci following advice from UCL researchers including Dr Dimitra Georgopolou, Dr José Ariza, Dr Suzana Hadjur, Dr Anna Köferle, Dr Andy Feber, Dr Miljana Tanic and Prof. Stephan Beck. In addition, Dr Pawan Dhami and Alex McLatchie aided with high-throughput sequencing. Dr Chris Barrington provided guidelines for 4C-seq computational analysis. Prof. Javier Novo (University of Navarra) aided in 4C-seq target selection.
CONTENTS

1 Introduction

1.1 Epigenetics

1.1.1 DNA methylation

1.2 Tools for analysing DNA methylation

1.2.1 Experimental tools

1.2.2 Bioinformatics tools

1.3 Open chromatin and DNase I hypersensitive sites

1.4 Histone marks

1.5 3D genomics

1.6 Genomics of disease

1.7 Epigenomics of disease

1.8 Thesis aims

1.8.1 Design and implementation of FORGE2

1.8.2 Design and implementation of eFORGE

1.8.3 4C-seq of neurological and autoimmune disorder-associated regions

2 Materials and Methods

2.1 4C-seq of neurological and autoimmune disorder-associated regions

2.1.1 Samples

2.1.2 4C-seq

2.2 eFORGE methods

2.2.1 Introduction

2.2.2 Sample processing

2.2.3 Preparation of sample overlaps

2.2.4 Analysis strategy

2.2.5 Annotation categories

2.2.6 Proximity filtering

2.2.7 eFORGE code structure

2.2.8 eFORGE input

2.2.9 eFORGE outputs

2.2.10 Multiple testing correction and false positive rates

2.2.11 Database extension and generation

2.2.12 tDMP and cDMP analysis

2.2.13 Source code

2.3 FORGE2 methods

2.3.1 Introduction

2.3.2 Sample processing

2.3.3 Preparation of sample overlaps

2.3.4 Analysis strategy

2.3.5 Annotation categories

2.3.6 LD filtering

2.3.7 FORGE2 input

2.3.8 FORGE2 outputs

2.3.9 Multiple testing correction and false positive rates

2.3.10 GWAS analysis

2.3.11 Source code
3 eFORGE Results .................................................................................................................. 103
   3.1 tDMP analysis ............................................................................................................. 104
   3.2 EWAS findings ............................................................................................................ 105
4 FORGE2 results .................................................................................................................. 118
   4.1 FORGE2 analysis of the GWAS catalogue ................................................................. 119
   4.2 Analysis of psoriasis GWAS data ............................................................................... 128
   4.3 Conclusions on GWAS analyses ............................................................................... 131
5 4C-seq results ..................................................................................................................... 132
   5.1 Introduction .................................................................................................................. 133
   5.2 4C-seq of regions homologous to neural disease loci in mNSCs ............................... 133
   5.3 4C-seq of immune disease loci in sorted blood cells ............................................... 147
         5.3.1 4C-seq analysis of the LRRC8B locus ................................................................. 148
         5.3.2 4C-seq analysis of the IKZF3 locus .................................................................... 152
         5.3.3 4C-seq analysis of the NCF4 locus .................................................................... 156
         5.3.4 4C-seq analysis of the CSF2 locus .................................................................... 158
         5.3.5 4C-seq analysis of the CCR6 locus .................................................................... 161
         5.3.6 4C-seq analysis of the TRAF1/C5 locus ............................................................. 164
6 Discussion .......................................................................................................................... 169
   6.1 eFORGE ....................................................................................................................... 170
         6.1.1 eFORGE aims and context .............................................................................. 170
         6.1.2 eFORGE findings and applications ..................................................................... 174
         6.1.3 Considerations and limitations of eFORGE ....................................................... 180
         6.1.4 Outlook on eFORGE ......................................................................................... 181
   6.2 FORGE2 ....................................................................................................................... 183
         6.2.1 FORGE2 aims and context .............................................................................. 183
         6.2.2 FORGE2 findings ............................................................................................. 184
         6.2.3 Considerations on FORGE and FORGE2 ......................................................... 187
         6.2.4 Outlook on FORGE2 ......................................................................................... 187
   6.3 4C-seq analysis ............................................................................................................. 188
         6.3.1 4C-seq analysis aims and context ..................................................................... 188
         6.3.2 4C-seq analysis findings ..................................................................................... 192
         6.3.3 Outlook on 4C-seq analysis .............................................................................. 194
7 References .......................................................................................................................... 198
LIST OF TABLES

Table 1.1: 450k packages ................................................................................................................. 24
Table 2.1: BLUEPRINT samples ....................................................................................................... 65
Table 2.2: Mouse target loci ............................................................................................................... 71
Table 2.3: Human target loci ............................................................................................................. 73
Table 2.4: Human-mouse homologous regions ..................................................................................... 75
Table 2.5: eFORGE sample processing ............................................................................................... 79
Table 2.6: eFORGE background probe matching ............................................................................. 83
Table 2.7: eFORGE background bin subcategories .......................................................................... 84
Table 2.8: eFORGE ENCODE false positives .................................................................................. 90
Table 2.9: eFORGE Epigenomics Roadmap false positives ............................................................... 91
Table 3.1: list of EWAS in the eFORGE catalogue ......................................................................... 117
Table 4.1: Common findings between FORGE2 and FORGE ......................................................... 125
Table 4.2: FORGE2 findings not detected in FORGE analyses ....................................................... 125
Table 4.3: FORGE2 findings that show a different tissue from FORGE ............................................. 125
Table 4.4: Additional tissue FORGE2 findings ................................................................................ 126
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>450k probe types</td>
<td>20</td>
</tr>
<tr>
<td>1.2</td>
<td>450k analysis pipeline</td>
<td>23</td>
</tr>
<tr>
<td>1.3</td>
<td>Multidimensional scaling</td>
<td>27</td>
</tr>
<tr>
<td>1.4</td>
<td>DNA methylation density plot</td>
<td>28</td>
</tr>
<tr>
<td>1.5</td>
<td>450k sample clustering</td>
<td>29</td>
</tr>
<tr>
<td>1.6</td>
<td>Singular value decomposition heatmap</td>
<td>32</td>
</tr>
<tr>
<td>1.7</td>
<td>FEM analysis</td>
<td>37</td>
</tr>
<tr>
<td>1.8</td>
<td>Cell composition effects</td>
<td>39</td>
</tr>
<tr>
<td>1.9</td>
<td>DNase-seq protocol</td>
<td>41</td>
</tr>
<tr>
<td>1.10</td>
<td>FAIRE-seq protocol</td>
<td>42</td>
</tr>
<tr>
<td>1.11</td>
<td>ATAC-seq protocol</td>
<td>43</td>
</tr>
<tr>
<td>1.12</td>
<td>Developmental landscape and open chromatin</td>
<td>44</td>
</tr>
<tr>
<td>1.13</td>
<td>Epigenetic Dashboard</td>
<td>46</td>
</tr>
<tr>
<td>1.14</td>
<td>Overview of 3C-related technologies</td>
<td>53</td>
</tr>
<tr>
<td>2.1</td>
<td>Process for selecting neurological disorder-associated variants</td>
<td>68</td>
</tr>
<tr>
<td>2.2</td>
<td>Process for selecting immunity-associated loci</td>
<td>69</td>
</tr>
<tr>
<td>2.3</td>
<td>FORGE2 H3K4me1 analysis of human target loci</td>
<td>74</td>
</tr>
<tr>
<td>2.4</td>
<td>eFORGE analysis strategy</td>
<td>81</td>
</tr>
<tr>
<td>2.5</td>
<td>eFORGE directory schematic</td>
<td>86</td>
</tr>
<tr>
<td>2.6</td>
<td>eFORGE code schematic</td>
<td>87</td>
</tr>
<tr>
<td>2.7</td>
<td>FORGE2 analysis schematic</td>
<td>96</td>
</tr>
<tr>
<td>3.1</td>
<td>eFORGE tDMP analysis</td>
<td>104</td>
</tr>
<tr>
<td>3.2</td>
<td>eFORGE analysis of autoimmune disease EWAS</td>
<td>106</td>
</tr>
<tr>
<td>3.3</td>
<td>eFORGE analysis of an EWAS performed on a surrogate tissue</td>
<td>108</td>
</tr>
<tr>
<td>3.4</td>
<td>eFORGE analysis of 5 cancer EWAS</td>
<td>111</td>
</tr>
<tr>
<td>3.5</td>
<td>Aggregated eFORGE results for analysis across 20 EWAS</td>
<td>113</td>
</tr>
<tr>
<td>3.6</td>
<td>Genomic distribution and tissue origin from studies in the eFORGE catalogue</td>
<td>115</td>
</tr>
<tr>
<td>4.1</td>
<td>Types of FORGE2 findings when compared to FORGE</td>
<td>120</td>
</tr>
<tr>
<td>4.2</td>
<td>FORGE and FORGE2 PR interval GWAS analysis</td>
<td>126</td>
</tr>
<tr>
<td>4.3</td>
<td>FORGE and FORGE2 Alzheimer’s disease GWAS analysis</td>
<td>127</td>
</tr>
<tr>
<td>4.4</td>
<td>FORGE2 psoriasis GWAS analysis</td>
<td>129</td>
</tr>
<tr>
<td>5.1</td>
<td>Schematic of 4C-seq mNSC computational and experimental analysis</td>
<td>135</td>
</tr>
<tr>
<td>5.2</td>
<td>4C-seq results for the mouse region homologous to the human locus that contains Alzheimer’s disease EWAS DMP cg02672452</td>
<td>139</td>
</tr>
<tr>
<td>5.3</td>
<td>4C-seq results for the mouse region homologous to the human locus that contains schizophrenia-associated GWAS SNP rs548181</td>
<td>140</td>
</tr>
<tr>
<td>5.4</td>
<td>4C-seq results for the Gcdh locus in mouse NSCs</td>
<td>143</td>
</tr>
<tr>
<td>5.5</td>
<td>4C-seq results for the Synj2 locus in mouse NSCs</td>
<td>144</td>
</tr>
<tr>
<td>5.6</td>
<td>4C-seq results for the Ppm1m locus in mouse NSCs</td>
<td>145</td>
</tr>
<tr>
<td>5.7</td>
<td>4C-seq positive control</td>
<td>146</td>
</tr>
<tr>
<td>5.8</td>
<td>Schematic of computational and experimental analysis for human immune GWAS regions</td>
<td>148</td>
</tr>
<tr>
<td>5.9</td>
<td>4C-seq results for the LRRC8B locus</td>
<td>151</td>
</tr>
<tr>
<td>5.10</td>
<td>4C-seq results for the IKZF3 locus</td>
<td>155</td>
</tr>
<tr>
<td>5.11</td>
<td>4C-seq results for the NCF4 locus</td>
<td>157</td>
</tr>
</tbody>
</table>
Figure 5.12: 4C-seq results for the CSF2 locus.................................................................160
Figure 5.13: 4C-seq results for the CCR6 locus.................................................................163
Figure 5.14: 4C-seq results for the TRAF1/C5 locus............................................................166
LIST OF ABBREVIATIONS AND ACRONYMS

ATAC-seq: Assay for Transposase Accessible Chromatin with high-throughput sequencing
BED: Browser extensible data
BH: Benjamini–Hochberg
BMI: Body mass index
BMIQ: Beta Mixture Quantile dilation
BSA: Bovine serum albumin
BY: Benjamini–Yekutieli
CHD: Congenital Heart Defect
CHiCP: Capture HiC Plotter
CLL: Chronic Lymphocytic Leukaemia
CNV: Copy number variation
COMET: Block of comethylation
CRISPR: Clustered regularly interspaced short palindromic repeats
ChIA-PET: Chromatin Interaction Analysis by Paired-End Tag Sequencing
ChIP-seq: Chromatin immunoprecipitation with high-throughput sequencing
CpG: Cytosine-guanine dinucleotide
DHS: DNase I hypersensitive site
DMC: Differentially methylated COMET
DMP: Differentially methylated position
DMR: Differentially methylated region
DNAm: DNA methylation
DNasel: Deoxyribonuclease I
DVP: Differentially variable position
EWAS: Epigenome-wide association study
FACS: Fluorescence-activated cell sorting
FAIRE: Formaldehyde-Assisted Isolation of Regulatory Elements
FDR: False discovery rate
FISH: Fluorescence in situ hybridisation
FORGE2: Functional element overlap analysis of Regions from GWAS Experiments 2
FPR: False positive rate
GEE: Generalised estimating equation
GO: Gene ontology
GSEA: Gene set enrichment analysis
GWAS: Genome-wide association study
IBD: Inflammatory bowel disease
IGR: Intergenic region
IHEC: International human epigenome consortium
IPA: Ingenuity pathway analysis
KEGG: Kyoto encyclopedia of genes and genomes
LD: Linkage disequilibrium
LMA: Ligation-mediated amplification
LMM: Linear mixed model
LPS: Lipopolysaccharide
MAF: Minor allele frequency
MALT: Mucosa-associated lymphoid tissue
MDS: Multidimensional scaling
MEGDEL: 3-methylglutaconic aciduria, deafness, encephalopathy, and Leigh-like disease

MS: Multiple sclerosis

MVP: Methylation variable position

MeDIP: Methylated DNA immunoprecipitation

PCA: Principal component analysis

PCHi-C: Promoter Capture-HiC

PCR: Polymerase chain reaction

RA: Rheumatoid arthritis

RNase: Ribonuclease

RRBS: Reduced representation bisulphite sequencing

SLE: Systemic lupus erythematosus

SNP: Single nucleotide polymorphism

SSc: Systemic sclerosis

SVD: Singular value decomposition

SWAN: Subset-quantile within array normalisation

T1D: Type 1 diabetes

TE: Tris and ethylenediaminetetraacetic acid

TFBS: Transcription factor binding site

TSH: Thyroid-stimulating hormone

TSS: Transcription start site

UC: Ulcerative Colitis

UTR: Untranslated region

WGBS: Whole-genome bisulphite sequencing

cDMP: Cell type-specific differentially methylated position
ccRCC: Clear cell renal cell carcinoma

eFORGE: Experimentally-derived Functional element Overlap analysis of ReGions from EWAS

eQTL: Expression quantitative trait locus
eQTM: Expression quantitative trait methylation

iPSC: Induced pluripotent stem cell

mNSC: Mouse neural stem cell

meQTL: Methylation quantitative trait locus

qPCR: Quantitative polymerase chain reaction

tDMP: Tissue-specific differentially methylated position

tDMR: Tissue-specific differentially methylated region
LIST OF APPENDICES

SUPPLEMENTARY TABLES

Table S1: sequences of PCR primers targeting mouse regions homologous to human GWAS/EWAS loci ..............................................................231
Table S2: BLUEPRINT 4C-seq primers ..................................................235
Table S3: tDMP sample IDs .....................................................................250

SUPPLEMENTARY FIGURES

Figure S1: Gcdh locus homology ...............................................................228
Figure S2: Ppm1m locus homology .........................................................230

PUBLICATIONS 251

eFORGE ANALYSIS CODE 261
1 Introduction
1.1 Epigenetics

Epigenetics studies the “stably heritable phenotype resulting from changes in a chromosome without alterations in the DNA sequence”\(^1\). Histone marks and DNA methylation (DNAm) rank among the main epigenetic factors. In addition, there is an increasing understanding of the importance of three-dimensional genomic context in regulating gene expression. Particularly, the role of enhancers, sequences of DNA that loop to gene promoters to stimulate transcription, is essential to achieve physiological transcription rates of target genes\(^2\). Here I present work that focuses on all of these aspects and their role in disease.

1.1.1 DNA methylation

5-methylcytosine is formed when a methyl (CH\(_3\)) group is added to the carbon-5 position of cytosine. This is currently the most commonly studied form of nucleotide methylation. Methylated genomic cytosines are predominantly found forming part of cytosine-guanine dinucleotides (CpG). Non-CpG DNAm, though less frequent, is also becoming increasingly relevant in research, thanks to its recent detection in adult tissues\(^3\). Humans are diploid organisms, that is, they have two full sets of chromosomes (23 chromosomes from the father and 23 chromosomes from the mother). The sequence of each set of chromosomes is a haploid genotype (or haplotype). Most of the approximately 28 million CpGs in the haploid human genome are present in a methylated state. Functionally, DNAm is generally considered a silencing mark, although the specific role of DNAm can vary with context. DNAm at transcription start sites can prevent
initiation of transcription, while DNAm at gene bodies may stimulate transcription elongation\(^4\).

### 1.2 Tools for analysing DNA methylation

#### 1.2.1 Experimental tools

Among the different technologies used to measure DNA methylation are sequence-based technologies and array-based technologies. Both of these technologies apply bisulphite to convert unmethylated cytosines to uracil, thus distinguishing methylated cytosines from unmethylated cytosines. Subsequent PCR reactions will amplify uracil as thymine, while 5-methylcytosines will be amplified as cytosines.

Sequence-based technologies include whole-genome bisulphite sequencing, reduced presentation bisulphite sequencing and pyrosequencing-related technologies. Whole-genome bisulphite sequencing is the most costly of the sequencing-based technologies, as it covers the entire genome and requires a read depth of at least 30x and, ideally, duplicates\(^5\). Because of the prohibitive cost of whole-genome bisulphite sequencing, reduced representation bisulphite sequencing was developed\(^6,7\). This technology enriches sequencing libraries for certain base content of the genome. Depending on the restriction enzymes used, regions with a high CpG, such as CpG islands, can be targeted.
Array-based technologies include those developed by Illumina and Affymetrix. Illumina arrays are the most common DNAm arrays used in the field of epigenetics research\(^8\). These arrays target a subset of genomic CpGs and present a cost-effective and highly reproducible way of assaying DNA methylation. Initially, Illumina developed the Golden Gate assay, focusing on 1505 cytosines of interest in disease and development\(^9\). Given the success of the GoldenGate assay, Illumina scaled up the number of regions in its successive arrays, which include the 27k array, the 450k array and the 850k (or EPIC array). The name of each of these arrays indicates the number of cytosines analysed. It is important to note that Illumina array development was not reduced to simply increasing the number of regions but also includes important modifications in the chemical assays employed to detect DNAm. Among these modifications one of the most important is the presence of type I and type II probes on Illumina arrays, which has been the focus of much statistical study in the development of analysis pipelines (including normalisation methods such as BMIQ or SWAN, in pipelines such as ChAMP\(^{10}\), minfi\(^{11}\), RnBeads\(^ {12}\), methylumi or wateRmelon\(^{13}\)).

The more detailed description of type I and II probes in Illumina array chemistry is shown in figure 1.1.
Figure 1.1: **450k probe types.** Type I (A) and type II probes (B). In type I design, each genomic locus is assayed by two bead types, one for unmethylated (U) and one for methylated (M), on the same colour channel. In type II design, each genomic locus is assayed by one bead type, with different colour channels for methylated and unmethylated cytosines. For both probe types DNAm is detected on bisulphite converted DNA by allele-specific single-base extension. A fluorescent label is incorporated in this extension reaction, allowing the identification of DNAm. Original figure from Bibikova et al., 2011\textsuperscript{14}. 
Reproducibility between Illumina arrays has been demonstrated to be very high ($R^2 > 0.95$)\cite{14,15}. This high reproducibility, coupled with easy implementation, a low price and the availability of multiple analysis pipelines has contributed to establishing Illumina arrays as the leaders in the field, especially as a main technology in epigenome-wide association studies (EWAS).

EWAS represent a novel approach in the study of human disease, studying changes in DNAm that may result from disease progression (as opposed to genetic predisposition), and/or environmental factors such as smoking\cite{16}. EWAS study the relationship of DNA methylation with disease in large cohorts in a way analogous to GWAS. However, unlike GWAS, EWAS are subject to scrutiny on potential confounding by cell composition effects\cite{8,17,18}. Therefore study design is vitally important in the field of EWAS, as EWAS benefit greatly from measures that reduce confounding, such as cell sorting methods.

An extensive comparison of DNAm technologies falls outside the scope of this thesis, but I recommend consulting the DNAm technology benchmarking paper produced by the BLUEPRINT consortium\cite{19}.
1.2.2 Bioinformatics tools

**DNAm array analysis tools.**

In epigenetics, one of the main areas of active development of bioinformatics pipelines focuses on Illumina DNAm array technologies, such as the Illumina 450k array.

After inaugural efforts from Illumina (Genome Studio software), the field saw a rapid development of a range of pipelines (e.g. ChAMP\textsuperscript{10}, minfi\textsuperscript{11}, RnBeads\textsuperscript{12}, methylumi\textsuperscript{20} and wateRmelon\textsuperscript{13}). Different pipelines offer varying utilities, and depending on the focus in analysis one pipeline or another may be preferred. An extensive list of 450k analysis packages is shown in table 1.1. In addition, a comparative list of analysis steps included in the main pipelines is shown in figure 1.2.
Figure 1.2: 450k analysis pipeline. List of main pipelines for 450k analysis, alongside the analysis steps performed. Colours indicate which pipeline includes each specific analysis step. Original figure from Morris and Beck, 2015."
**Introduction**

**Table 1.1: 450k packages.** A list of freely available packages for 450k analysis, including details on the application of each package. Original table from Morris and Beck, 2015\textsuperscript{21}.

Most of the pipelines for 450k analysis share a series of common elements. The main processes include\textsuperscript{10}:

1. Data loading
2. QC
3. Normalisation
4. SVD/PCA
5. Correction of batch effects
6. Identifying differentially methylated positions (DMPs) and regions (DMRs)
7. Downstream functional analysis (pathways, GO, FEM)

Additional elements in pipelines include\textsuperscript{10}:

- Detecting copy number variants (CNVs, e.g. ChAMP)
- Correcting cell composition effects (e.g. minfi, recent versions of ChAMP).

**Data loading**

There are a variety of data formats in 450k analysis. One of the most common formats, the IDAT format, comprises data from the raw images of the arrays. This “raw data” format produces very large files and is hard to transfer and manage,
especially for large-scale studies. Therefore another format, the “beta matrix”, is also used as part of 450k data analysis, especially as part of meta-analyses or for replication of studies. This matrix includes a series of beta-values (DNAm values between 0 and 1) for each of the probes on the 450k array. Some pipelines, including ChAMP, support initial analysis from a beta matrix as well as from IDAT files.

In addition to DNAm probes, there are 65 SNP probes on the 450k array (control probes). The analysis pipeline must filter these control probes and also any DNAm probes with failed measurements. The 450k array includes 485,512 DNAm probes, and the EPIC array includes 867,531 DNAm probes. Of these probes, a percentage may fail. Probe filtering usually includes:

- probes that have failed to hybridise (and thus have a detection p-value < 0.01). Samples with a high number of failed probes may be removed from analysis.
- probes represented by fewer than 3 beads on the array.
- SNP-related probes. If SNPs are shown to have a very large effect on DNAm results the relevant probes should be excluded from analysis.
- Cross-hybridising probes. A number of probes that target multiple genomic loci have been identified, including autosomal probes that cross-hybridise to the sex chromosomes.
- non-CpG probes. Non-CpG DNAm presents important features that distinguish it from the more common CpG DNAm, such as an enrichment
in stem cells and pluripotent cells\textsuperscript{23}. Non-CpG probes are typically excluded from analysis, though not always\textsuperscript{24}.

- probes located on chromosomes X and Y. These probes require separate analysis in cohorts that contain both men and women, as they could otherwise be driving spurious DNA methylation signals. These probes are often excluded from data processing.

**Quality Control (QC):** it is vital to check the quality of the data before normalisation. The strategies typically used to check data quality include Multidimensional Scaling (MDS), density plots and clustering methods\textsuperscript{10}:

-MDS (Multidimensional Scaling): this method aims to represent the degree of similarity between individual samples analysed. Each datapoint in MDS plots represents a sample from the study, and datapoints are grouped by similarity to other datapoints. MDS plots are a way to display information contained in a distance matrix (figure 1.3).
Figure 1.3: Multidimensional scaling. This MDS plot separates the green samples from the brown samples, reflecting the distance between sample groups C and T. Original figure from ChAMP package vignette\textsuperscript{10}.

-Density plots: DNAm follows a standard bimodal distribution. Alterations of this bimodal curve can be detected by observing the beta distribution line across samples (figure 1.4). Additional peaks between the two expected peaks from the bimodal distribution may signal a lack of sample quality.
**Figure 1.4: DNA methylation density plot.** Density plot with the classic bimodal distribution of DNAm beta values. The peak close to zero corresponds to unmethylated sites across the genome. The peak close to one corresponds to methylated sites, and the rest of the bimodal curve corresponds to sites that present intermediate methylation values. Original figure from ChAMP package vignette\textsuperscript{10}.

- Clustering methods: a typical hierarchical clustering plot (or dendrogram) shows the ordered similarity relationships between samples based on the data (figure 1.5). If a particular group of samples do not cluster well or if individual samples cluster in an unexpected group this may point to either low sample quality or labelling errors.
**Figure 1.5: 450k sample clustering.** Clustering dendrogram showing adequate clustering separating sample groups C (red) and T (blue). Original figure from ChAMP package vignette\textsuperscript{10}.

**Normalisation:**

I have previously explained the difference between type I and type II probes (figure 1.1). Due to this difference in design, type I and II probes have different beta distributions. This difference in distribution, if uncorrected, could potentially lead to spurious results in downstream analysis. To minimise the effect of probe type in a given study (which is a confounder when trying to find DNAm differences between cases and controls), it is necessary to apply a
normalisation method (several are available - e.g. SWAN\textsuperscript{25}, BMIQ\textsuperscript{26}, PBC\textsuperscript{27}, and Funnorm\textsuperscript{28}).

The characteristic feature of each of the main methods is the following:

SWAN: “subset-quantile within array normalisation”, this method makes the key assumption that the amount of CpGs within the 50 bp probe sequence reflect the underlying biology of the region in question\textsuperscript{25}. This information is then used to adjust type I and type II probe distribution, basing the adjustment on a subset of randomly selected type I and II probes that have one, two and three underlying CpGs.

BMIQ: this method first applies a three-state beta-mixture model to assign probes to DNAm states\textsuperscript{26}. Quantiles are then formed from the probabilities that result from the beta-mixture model. Finally, this method adjusts the data by performing a DNA methylation-dependent dilation transformation.

PBC: this method rescales type I and type II probe values simply by shifting the summits of the two peaks in the bimodal distribution of DNAm values (that is, the methylated and unmethylated peak).

Funnorm: this method considers any difference in the 848 control probes on the 450k array as unwanted technical variation, and adjusts the data accordingly. It is important to note that sometimes a proportion of these 848 control probes do not pass array QC. For these cases the effectiveness of this method is limited.
**SVD:**

SVD (singular value decomposition\(^{29}\)) analysis is a matrix factorisation process performed to identify the main components of variation. These components can be either technical or biological.

SVD analysis results can be plotted as a heatmap (**figure 1.6**) showing the top principal components correlated with the covariates information provided. SVD can highlight important confounders or batch effects in the data that may require adjustment in downstream analysis (as described in the next section).
Figure 1.6: **Singular value decomposition heatmap.** This heatmap identifies the top 20 principal components underlying variation from an EWAS on ovarian...
cancer before quantile normalisation (a), after quantile normalisation (b) and after adjustment for bisulphite conversion (BSC), DNA input and batch effects (c). The grey line separates phenotypic factors (age, case-control status, stage of cancer, cancer grade and histological subtype) from technical factors (batch number, DNA input, bisulphite conversion efficiency controls 1 and 2). Variation due to these technical factors represents a source of confounding, and is removed in (c). Original figure from Teschendorff et al., 2009.

Adjustment for batch effects:

A batch effect is a technical source of variation introduced during sample handling. For example, if one day the temperature in the lab is higher, or if the person handling the arrays is different, then the array measurements will differ slightly. Randomised sample design (in which cases and controls are assigned randomly to different array batches) helps to reduce the confounding caused by batch effects.

A computational tool named ComBat\textsuperscript{30} has been implemented in packages such as ChAMP to correct for batch effects in 450k array analysis. ComBat uses an empirical Bayes framework (with options for parametric and non-parametric priors) to input previously known information to estimate sources of technical
variation. ComBat has proven to be an efficient method for reducing unwanted sources of variation, and has been reported to reduce in certain cases up to 98% of batch effects.

**Differentially methylated probes/regions/comets:**

Differential DNAm can be expressed at different levels, from the single cytosine to large regions spanning megabases. For a brief description of the different categories employed in differential DNAm studies:

- **Differentially methylated position (DMP):** differential DNA methylation detected at the level of a single cytosine. As noted above, the 450k array contains 485,512 different probes, and each probe targets a specific genomic cytosine. Probes that detect significant DNA methylation changes between cases and controls are classed as Differentially Methylated Positions (DMPs).

- **Differentially methylated region (DMR):** a group of proximal cytosines with similar methylation change can be grouped into a differentially methylated region (DMR). Methods for identifying DMRs include Probe Lasso\textsuperscript{32}, bumphunter\textsuperscript{33}, comb-p\textsuperscript{34} and DMRcate\textsuperscript{35}, among others.

  In bumphunter, DMRs are detected as peaks in a representation of smoothed DNAm values. Probe Lasso gathers proximal significant
positions using a flexible window approach, in an effort to combat CpG density bias. DMRcate tackles the same problem using a Gaussian kernel to fit repeated DNAm measurements spatially across the genome. Finally, comb-p employs spatially assigned p-values to detect regions of enrichment.

- **COMETs**: The study of the oscillations of genomic DNAm values has led to the definition of COMETs\(^\text{36}\), or blocks of comethylation, which are segmentations of the genome into areas of continual smoothed DNAm level estimates. One of the uses of COMETs is that they allow for the recovery of methylome feature information for non-saturating levels of sequencing depth in whole-genome bisulphite sequencing. A package, named “cometvintage”, has been developed to identify differentially methylated comets (DMCs). In regards to COMET structure, it is important to consider that a significant correlation between COMETs and haplotype blocks has been reported\(^\text{36}\), signalling a potential epigenetic equivalent of the haplotype map. At a local level, it had previously been known that DNAm levels correlated within 1kb\(^\text{37,38}\).
Functional/pathway analysis

After uncovering differentially methylated positions, regions or COMETs, there are further steps that have to be taken to identify the biological relevance of the differential DNAm observed. One of the classical ways to do this is to perform a Gene Ontology (GO) analysis\textsuperscript{39,40}, which uses a hierarchical set of terms encompassing most of the known biological processes, cellular components and molecular functions. Additional pathway analysis may be done, for example using the pathway catalogue developed by the Kyoto Encyclopaedia of Genes and Genomes (KEGG)\textsuperscript{41}.

Another way to extract biological meaning from observed differential DNAm is to perform a Gene Set Enrichment Analysis (GSEA)\textsuperscript{42}. This method presents a markedly different approach to the “statistical enrichment of overlap analysis” performed by tools such as DAVID\textsuperscript{43,44}, Ingenuity (IPA; Qiagen; http://www.qiagen.com/ingenuity) or AmiGO\textsuperscript{45}. In the aforementioned tools, the study list of genes (e.g. those genes with promoters that overlap differentially methylated regions) is compared to a randomly sampled set of gene lists from the appropriate background pool (e.g. the array used for the study). The top study list is thus assigned an enrichment score through a statistical test such as a binomial test, a Fisher’s exact test, a Chi-squared test or a hypergeometric test. However, the GSEA approach does not only use the top study list of genes, it uses the whole list of genes from the study (e.g. those that are differentially methylated and those that are not). This inclusion of additional information makes for a different approach, in which random permutations of the total list of
Introduction

genes are performed to calculate a normalised enrichment value for the distribution of genes of a particular group. In this approach one group of genes may have a tendency to be higher up on the list, thus showing association with the phenotype under study. The appropriate statistical tests used for this list permutation analysis are the Wilcoxon test and the Kolmogorov-Smirnov test.

Complementing the previous methods, the FEM packageintegrates protein interaction networks with DNAm data, identifying a subnetwork of protein interactions formed by genes highlighted by the differential DNAm results. An example of FEM output is given in figure 1.7. FEM can be used to integrate DNAm and gene expression data, or alternatively upon DNAm data alone.

![FEM analysis](image)

**Figure 1.7: FEM analysis.** Results show one protein interaction network identified using FEM analysis of differential DNAm results. Original figure from ChAMP package vignette.
Finally, the iEVORA package\textsuperscript{47} complements standard EWAS analysis of Differentially Methylated Positions by analysing Differentially Variable Positions. In the analysis of Differentially Methylated Positions linear mixed models are employed to detect genomic cytosines that present different methylation values in cases and controls. In iEVORA a different approach is implemented with the aim to find differentially \textit{variable} regions, in which the \textit{variation} of DNAm is different between cases and controls (e.g. there is a difference in the standard deviation of methylation values between cases and controls for a given genomic cytosine). In order to detect this iEVORA performs Bartlett’s test, and, due to the high sensitivity of this initial test to single outliers, iEVORA also performs a t-test to rank significant Differentially Variable Positions (DVPs).

\textbf{Additional elements in an EWAS pipeline:}

\textbf{Cell composition effects:}

As mentioned before, one of the main problems in EWAS is confounding by cell composition effects\textsuperscript{48,49,50,51,52} [reviewed in \textsuperscript{8,17,18,53,54} and described in \textbf{figure 1.8}]. To this end, in 2012 Andres Houseman developed a correction method applicable to whole blood, which applies DNAm reference data from FACS sorted blood cells\textsuperscript{48}.\vspace{0.5cm}
Because of the need to apply cell composition correction to non whole-blood-based EWAS, a variety of reference-free correction methods have been developed\textsuperscript{49,51,52}. These methods assume that confounding by cell type will be one of the main sources of variance, and some of these methods try to correct the effects of the main principal components detected when analysing the data. In addition, some researchers claim that overcorrection is being performed by some of these methods\textsuperscript{50}.

**Figure 1.8: Cell composition effects.** Proportions of cell types may be different between cases and controls, as is shown for cell types A (blue) and B (red). This difference in proportions may drive spurious DNAm signals and is thus a strong confounder to consider in EWAS analysis. Original figure from Liang and Cookson, 2014\textsuperscript{53}.

Correction for cell composition effects has become so commonplace in the field of epigenetics that some of the main bioinformatics pipelines (e.g. ChAMP) have incorporated a variety of these methods into their framework\textsuperscript{10}. The resulting competition from different groups will no doubt bring forward advances in the correction of cell composition effects for a variety of tissues.
1.3 Open chromatin and DNase I hypersensitive sites

There are currently three main methods to identify open chromatin sites, namely DNase-seq\textsuperscript{55}, ATAC-seq\textsuperscript{56} and FAIRE-seq\textsuperscript{57,58}. One way to measure sites of open chromatin in the genome is by the study of DNase I hypersensitivity. DNase I, as the name indicates, is an enzyme that degrades DNA. In the DNase-seq protocol, open chromatin sites are degraded, leaving blunt DNA ends at either side (see \textbf{figure 1.9}). These blunt ends are ligated with a biotinylated linker, digested with MmeI and bound to Dynal beads. A second linker is then ligated and a PCR is performed prior to sequencing. After sequencing, bioinformatics approaches are applied to identify the location in the genome of the segments that were initially cleaved and degraded with DNase I.
Figure 1.9: DNase-seq protocol. Peaks of DNase-seq are shown in the lower panel. Each one of these peaks corresponds to an open chromatin site (in this case also named a DNase I hypersensitive site or DHS). Original figure from Song and Crawford, 2010.55

Another way to detect sites of open chromatin is to perform FAIRE-seq (formaldehyde-assisted isolation of regulatory elements). As the name indicates, this method employs formaldehyde to link proteins and DNA together. DNA is
then fragmented through sonication and extracted using a phenol-chloroform mixture. The phenol-chloroform mixture will retain the protein-linked DNA, and the DNA not linked to proteins (i.e. “open chromatin”) will remain in the aqueous phase (figure 1.10). Sequencing ensues, followed by bioinformatics analysis to pinpoint the location of the open chromatin sites.

**Figure 1.10: FAIRE-seq protocol.** Formaldehyde crosslinking of proteins and DNA, followed by sonication, ensures that open chromatin sites will be separated from the rest of the genome through a phenol-chloroform extraction. Original figure from Furey, 2012.59

In ATAC-seq, however, a different approach is used, based on the properties of transposons. Transposons, initially proposed by Barbara McClintock in the 1940s, are “DNA sequences that are able to move from one location to another in the genome”60. Because of this property of moving from one genomic location to
another, transposons are involved in genomic instability, and perform a well-studied role in cancer\textsuperscript{61}. However, in ATAC-seq the enzyme that catalyses the movement of transposons to other parts of the genome (transposase) is used to cut exposed DNA and ligate specific sequences named adapters to this region\textsuperscript{56} (\textbf{figure 1.11}). Fragments of DNA are then isolated, PCR-amplified, sequenced and mapped to detect open chromatin sites. Currently, ATAC-seq mainly employs a mutated hyperactive version of transposase Tn5.

\textbf{Figure 1.11: ATAC-seq protocol.} A schematic of ATAC-seq is shown [A], showing the integration of Tn5 transposome in open chromatin sites, which is then detected by sequencing. In addition, ATAC-seq requires fewer cells as starting material and a shorter preparation time than competing methods DNase-seq and FAIRE-seq [B]. Original figure from Buenrostro et al., 2013\textsuperscript{56}.

As open chromatin regions are representative of enhancers and other genomic elements with a role in regulating gene expression\textsuperscript{62}, it is natural to link the study of open chromatin to the study of genomic regulation. One specific case where this has been explored in depth is the case of development\textsuperscript{63}. For differentiated NK and Th2 immune cells, it has been shown that approximately
one third of DHSs are embryonic, and the other two thirds originate either from early differentiation phases or from late differentiation phases. These findings have been put in relation to the epigenetic landscape initially proposed by Conrad Waddington (figure 1.12). Other epigenetic features affected by this differentiation process include histone marks, DNAm and 3D genomics. These different epigenetic features are related, as is shown by the success in computational approaches that use DHS and histone mark information to predict the 3D genomic landscape, linking up several of the main epigenetic features which are in turn linked to the regulation of gene expression and cellular phenotype.

**Figure 1.12: Developmental landscape and open chromatin.** Waddington’s epigenetic landscape and DHS origin for several immune cell types. The left panel shows the variation in size for the accessible chromatin landscape across several differentiation stages (stem cell, haematopoietic progenitors and differentiated immune cells). The right panel shows the gradual loss of stem cell DHSs (blue)
1.4 Histone marks

In the chromatin context, DNA is wrapped around structures named nucleosomes. These nucleosomes are octamers, and each part of the octamer is a histone protein (H2A, H2B, H3 and H4). These four histone proteins are present twice in each nucleosome, forming the octamer. In addition, histone H1 is associated with the DNA that links consecutive nucleosomes. Histone H1 is known as the “linker” histone.

In addition to providing structural support for DNA, histones also provide a substrate for signalling chemicals to bind. Ongoing research continuously identifies many new types of histone modifications, and currently well over 100 histone modifications are known. Many of these modifications are listed in Huang et al., 2014. Much effort has been made to unravel a “histone code” that reflects the activity and assembly of the genome. It is known that certain histone marks co-localise with specific types of cis-regulatory elements. Mechanistically, histone modifications may also act to establish regulatory elements both through cis (electrostatic effects that weaken the histone-DNA interaction) and trans effects (driven by effector protein binding and protein-protein recognition). A “dashboard” listing the regulatory states associated with the main histone marks is shown in figure 1.13.
Introduction

**Figure 1.13: Epigenetic Dashboard.** This dashboard contains the main histone marks and links to activation/repression for specific genomic contexts. Different marks separate active, inactive and poised promoters [A], including H3K4me3 (active and poised promoters) and DNAm (inactive promoters). Gene bodies can present repression-associated marks (e.g. H3K9me3) or marks linked to active transcription (e.g. H3K36me3) [B]. Enhancers typically present H3K4me1 and H3K27ac when active and can also present DNAm (inactive enhancers) [C]. Repressed regions can be separated into those that present transient repression (characterised by H3K27me3) and those that form constitutive heterochromatin (H3K9me3) [D]. Original figure from Zhou et al., 201173.

A brief description of each of the 6 main histone modifications74 (according to IHEC guidelines, [http://www.ihec-epigenomes.org/](http://www.ihec-epigenomes.org/)) is listed below:

- H3K4me3: the characteristic mark of the promoters of actively transcribed genes71. H3K4me3 is deposited by SET1A/SET1B75, and readers of H3K4me3 include the V(D)J recombinase subunit RAG262 and PHD finger proteins76. One of
the PHD finger proteins is the RNAPII TAF3 subunit (part of TFIID). H3K4me3 is thus connected to the initiation of transcription through TFIID. H3K4me3 can also be found alongside H3K27me3 (a repressive mark) in the so-called bivalent domains, typical of the promoters of important developmental genes. It has been posited that these domains keep these genes inactive, but in a “poised” state for active transcription. Detectable levels of H3K4me3 can also be found in active enhancers. It is important to note that both for H3K4me3 and for subsequent histone modifications, I indicate tendencies for marks to co-localise with particular elements. These tendencies are not exclusive categorisations (e.g. H3K4me3 is typical of active promoters, but can also be found elsewhere). In addition, certain marks can co-localise with genomic elements other than the canonical/established elements. H3K4me3 in particular forms broad non-canonical genomic domains during early development. These findings suggest that potential new biological associations could be found for this classic promoter-associated histone mark, including an association with maternal-to-zygotic transition.

-H3K4me1: the most tissue specific mark among those assayed by the Epigenomics Roadmap Consortium. H3K4me1 is predominantly enriched at enhancer regions. In a similar way to promoters, a region may be marked as an enhancer, but it may not yet be active (it is a “poised” enhancer). H3K4me1 is a mark that covers both poised and active enhancers. H3K4me1 often precedes nucleosomal depletion, pre-marking potential enhancers prior to activation. H3K3me1 is deposited by MLL3/MLL4. Mechanistically, H3K4me1 binds the
protein acetyltransferase TIP60\textsuperscript{83}, which leads to subsequent steps of enhancer activation.

-H3K27ac: this mark, along with other histone acetylations, is a mark that co-localises with active regulatory elements (i.e. active promoters and enhancers), and has been shown to separate active enhancers from those in a poised state\textsuperscript{84}. H3K27ac can be deposited both by p300 and CREB binding protein (CBP)\textsuperscript{85}. In the search for histone modifications associated with active regulatory regions, H3K27ac can be complemented by H3K9ac and H3K18ac\textsuperscript{86}. Mechanistically, one way for the activation of regulatory elements by H3K27ac is through electrostatic (or cis) effects: the negative charge of acetyl groups repels the negative charge of the DNA phosphate backbone, provoking a disassembly of compact chromatin. A second mechanism is formed by the so-called trans effects: protein with domains such as bromodomains (e.g. H3K27ac associated factors p300 and CBP\textsuperscript{87}) recognise H3K27ac and recruit other factors that regulate chromatin remodelling (e.g. SWI/SNF\textsuperscript{72}).

-H3K27me3: at the same position, but with a different chemical modification, there is a mark with completely different regulatory associations. Deposited by the polycomb repressive complex 2 (PRC2), this mark is associated with the repression of gene expression\textsuperscript{71}. Mechanistically, H3K27me3 recruits another polycomb complex (PRC1) that blocks RNA polymerase and causes repression of transcription\textsuperscript{88}. H3K27me3 has been linked to the repression of alternative pathways to maintain lineage fidelity during development\textsuperscript{89}. It is important to note that, during differentiation, repression is as important as activation, and in
the same way as H3K27ac allows for the development of active regions during cellular differentiation, H3K27me3 represses alternate lineage formation, thus avoiding the activation of unnecessary processes\(^9\).

-H3K36me3: This mark is associated with actively transcribed regions\(^9\), and is deposited by HYPB and NSD1\(^9\). H3K36me3 has been linked to the repression of transcription initiation in gene bodies, dosage compensation and DNA repair\(^9\). This mark is recognised by ZMYND11, which in turn is linked to the control of transcription elongation\(^9\). In addition, some evidence suggests a role for H3K36me3 in defining exons. Exons are enriched for nucleosomes with H3K36me3\(^9\). H3K36me3 is also proposed to affect alternative splicing, with mechanistic evidence for the downstream involvement of MRG15 and PTB in the repression of the silenced exon\(^9\).

-H3K9me3: This mark is typical of constitutive heterochromatin\(^7\), co-localising with silenced regions that contain repetitive elements, transposons and other repressed elements. One of the readers of H3K9me3 is heterochromatin protein 1 (HP1), a chromodomain protein that extends the formation of inaccessible chromatin\(^7\). The adequate repression of constitutive heterochromatin is vital, as defects in this mechanism can result in genomic abnormalities, including chromosomal rearrangements\(^9\).

Knockout mice experiments further highlight the relevance of the aforementioned histone modifications. Many of the related factors are required for embryonic development, including p300\(^9\) and CBP\(^9\) (H3K27ac), HP1\(^9\).
(H3K9me3), SET1\textsuperscript{98} (H3K4me3), core components of PRC2, including EZH2 and SUZ12\textsuperscript{99} (H3K27me3), HYPB\textsuperscript{100} (H3K36me3) and TIP60 (H3K4me1)\textsuperscript{101}. 


Introduction

1.5 3D genomics

DNA in the eukaryotic nucleus presents a remarkable capacity for folding. Each diploid human cell contains around 2 metres of DNA if the chromosomes are placed end to end, yet the nucleus is only about 6 micrometres in diameter\textsuperscript{102}. The necessary packaging is achieved by specialised proteins that generate a series of coils and loops, providing increasingly higher levels of organisation.

One of the most studied mechanisms for chromatin folding is mediated through the DNA-binding factor CTCF and the cohesin complex, with CTCF binding to specific sequences in the genome and cohesin providing the structural framework for chromatin loops\textsuperscript{62}.

At a macroscopic scale, DNA folding is also remarkably fine-tuned, as is shown by the careful assembly of chromosome structure during mitosis\textsuperscript{102,103}. In addition, at a local level, DNA folding exerts a remarkable influence on the functional mechanisms that control gene expression, as is exemplified by the conformational structures of enhancers, insulators and repressors\textsuperscript{62,104}. This relationship with gene expression is dynamic, as is shown by the dramatic changes in DNA folding during cellular differentiation\textsuperscript{65}. The disruption of regions that control 3D genomic structure through genome editing has highlighted the importance of chromosome conformation in essential biological processes\textsuperscript{65}. 
Initial techniques for the study of chromosome organisation include the use of microscopy to detect specific regions of the nucleus\textsuperscript{105,106}. In the early 1980s, the development of fluorescence in situ hybridisation (FISH) allowed for the study of local DNA folding in finer detail \textsuperscript{107,108}. This technique uses fluorescent probes that only bind to genomic regions which show a high degree of sequence complementarity. Technical advances in the study of DNA folding in recent decades include the development of chromosome conformation capture (3C) techniques\textsuperscript{109,110}. The idea underlying these methods is that DNA contact frequencies can be determined through the quantification of ligation junctions in chromatin that has undergone a process of fixation, digestion and religation\textsuperscript{111}. The analysis of these DNA contact frequencies offers insights into chromosome topology\textsuperscript{111}. These techniques use a combination of molecular biology methods such as DNA cross-linking, restriction enzyme digestion and DNA ligation, coupled with different product quantitation techniques (array detection, sequencing) to analyse genomic organisation in a cell’s natural state. Originally proposed in the year 2002, 3C has paved the way for a family of related techniques with different applications, including 4C, 5C, Hi-C, capture Hi-C, capture-C\textsuperscript{104}, and ChIA-PET\textsuperscript{111} (figure 1.14).
Selection of a specific 3C-based technique can depend on the level of throughput required:

- 3C (chromosome conformation capture\textsuperscript{109}): this technique is used to analyse interactions of one genomic region with another region (one vs one interactions). The basic 3C technique has five experimental steps (cross-linking, restriction digest, intramolecular ligation, reverse cross-links and quantitation). Quantitation is usually performed with qPCR or agarose gel detection.

Figure 1.14: Overview of 3C-related technologies. Capture-C and capture Hi-C are not shown as they are more recent than the review that contains this figure\textsuperscript{104,112}. Original figure from de Wit and de Laat, 2012\textsuperscript{111}.
Capture-C: this technique combines 3C library preparation with sonication fragmenting, oligonucleotide capture technology and high-throughput sequencing, allowing for many regions to be assayed in one experiment. The sonication step in Capture-C presents an important advantage compared to 4C and 5C as it allows Capture-C to accurately quantify unique ligation junctions within the 3C library (sonication randomly generates unique fragments and overamplified PCR duplicates can be computationally removed).

4C (circularised chromosome conformation capture): this technique has a significant advantage over 3C in that only the sequence of one of the sites of interest needs to be known. This fragment is known as the “bait”. 4C studies the interactions of the “bait” region against all other chromosome regions (one vs all). 4C follows the same steps as 3C, with the exception that additional processing is needed before quantification of the fragments of interest. 4C includes the 3C steps of cross-linking, restriction digest, intramolecular ligation, reverse cross-links and then the 4C-specific steps of a second restriction digest, self-circularisation, inverse PCR and quantitation. Quantitation can be performed by sequencing (4C-seq) or by array detection (4C-chip). The inverse PCR step is the step in which primers based on the “bait” sequence amplify circularised molecules in which the unknown captured fragment is also present.

5C (carbon-copy chromosome conformation capture): This technique allows for the parallel analysis of interactions between many selected loci (many vs many). 5C follows the same first four steps as 3C, but then relies on performing multiplex ligation-mediated amplification (LMA) after cross-links
reversal to generate a library with fragments representative of many genomic interactions. As in 4C, quantitation can be performed by sequencing or by array detection.

-Hi-C\textsuperscript{115}: This technique allows for the parallel analysis of interactions between all genomic loci (all vs all)\textsuperscript{115}. Hi-C follows the same first four steps as 3C, however, in Hi-C, a biotin-labelled nucleotide is incorporated at the ligation junction. This enables selective purification of chimeric DNA ligation junctions as fragments can be pulled down with streptavidin. Quantitation of chromatin interactions is then performed through sequencing. A variant of this technique exists, named Capture Hi-C, that uses specific RNA baits to enrich target loci in a Hi-C library\textsuperscript{112}.

-ChIP-loop (Chromatin Immunoprecipitation-loop\textsuperscript{116}): This method differs from the previous techniques in that a specific protein mediates the interaction between two chromosomal regions. Following the first steps of 3C, after the cross-linking and digestion stages chromatin immunoprecipitation (ChIP) is performed to pull down the protein bound to the site of interest\textsuperscript{116}. After this step normal 3C procedures are conducted (quantitation can be performed by qPCR)\textsuperscript{116}. This technique can assay one vs one interactions.

-ChIA-PET (Chromatin Interaction Analysis by Paired-End Tag Sequencing\textsuperscript{117}): this technique combines ChIP (chromatin immunoprecipitation)-based enrichment, chromatin proximity ligation, paired-end tags and high-throughput sequencing to determine de chromatin interactions on a genome-wide scale\textsuperscript{117}. 

After digestion of crosslinked chromatin, the steps of addition of adapters and ligation (of the crosslinked chromatin with these) are performed on antibody-coated beads. This is then followed by reverse crosslinking, restriction enzyme digestion, addition of PCR adapters and PCR. Quantification is performed by sequencing. ChIA-PET is used to assay all vs all interactions.

1.6 Genomics of disease.

Modern medical research into the genetics of disease began with mendelian disorders, such as the study of alkaptonuria by Archibald Garrod\textsuperscript{118}. Insights into the genetics of non-mendelian disorders (such as asthma\textsuperscript{119}, as well as many other autoimmune, neurodegenerative and metabolic diseases) were provided by studies on identical twins, through which it became apparent that these disorders are affected by non-genetic as well as genetic factors, and are thus termed complex diseases. After the completion of the human genome project\textsuperscript{120}, SNP arrays were used in large cohorts to investigate the association of common genetic variation and disease in an approach termed genome-wide association studies (GWAS)\textsuperscript{121}. Despite initial encouraging findings for macular degeneration\textsuperscript{121}, these efforts did not find common variants with a large effect size for common disorders (with a few exceptions, including ApoE variants and Alzheimer's disease). GWAS, however, did find thousands of replicated SNPs with small effect sizes. While a substantial portion of the genetic contribution to many diseases remains unexplained, efforts in the GWAS field continue to detect new associations, often using large sample sizes\textsuperscript{122}.
A typical GWAS involves the comparison of genetic variants between a group of cases and controls for a specific disease. Sample sizes currently reach the hundreds of thousands for certain studies, to increase power in order to detect the small effect sizes observed for many GWAS SNPs. Genotyping of variants is performed using SNP arrays, such as those produced by Illumina or Affymetrix, which can identify millions of variants. Subsequent steps include data normalisation and further bioinformatics processing to exclude any technical variation generated during array or sample processing. Once variants are known for the sample under study a series of statistical approaches, including generalised estimating equations (GEEs) or linear mixed models (LMMs), are applied to identify those variants that are present in significantly higher (or lower) numbers in cases compared to controls. Those variants that pass the genome-wide significance threshold (5x10^{-8} for Bonferroni correction) are reported as associated with the disease under study. These are thus included in the GWAS catalogue, which of late also includes GWAS summary statistics. For the analysis of GWAS results it is important to consider linkage disequilibrium (LD). It is known that many genomic variants segregate independently from each other through evolutionary history. In these cases the presence of one variant at a given genomic position will not condition the presence of another independent variant at another position. However, when proximal variants form part of the same haplotype the phenomenon of LD will occur. LD is defined as the non-random association of alleles at different loci, that is, the presence of one allele will condition the presence of another. For GWAS results many variants in a haplotype may be significantly associated with a given disease, and it is hard to prove which of these SNPs is the most influential variant for a GWAS-reported
association. It must also be highlighted that GWAS SNP arrays may not cover all the SNPs in a given region, and thus specialised arrays (such as the Immunochip\textsuperscript{125} and the Metabochip\textsuperscript{126}) as well as fine-mapping efforts\textsuperscript{127} are being applied to follow up GWAS findings. In addition, other experimental approaches and bioinformatics efforts may be applied to analyse GWAS data\textsuperscript{128,129}, including approaches that use epigenetic mapping information\textsuperscript{130}.

Given the use of a stringent Bonferroni genome-wide significance threshold of 5$\times$10$^{-8}$, many SNPs that do not reach this threshold may have an influence on the disease under study. Some approaches involve the use of epigenetic data to prioritise candidate regions, which are then validated experimentally\textsuperscript{131}. These efforts are aiding the detection of causal SNPs below this threshold.

In addition, many diseases and traits are a result of grouping cases belonging to different subtypes. To improve current understanding of the genetics of disease the use of intermediate phenotypes, or endophenotypes, has recently been championed as a way to gain further knowledge on many disease-specific associations\textsuperscript{132}. Endophenotypes are defined as intermediate physiological or psychological traits, for example metabolite abundance, a specific neuronal function, or gene expression patterns\textsuperscript{133}.

While significant advances are being made in the understanding of disease-associated protein-coding regions\textsuperscript{134}, variation in non-protein-coding regions presents important challenges for interpretation, often requiring multiple techniques for the elucidation of mechanistic impact\textsuperscript{130}. These approaches can
Introduction

include genome editing, 3D genomics and epigenetic mapping of histone marks and DHSs, as in the recent case of the mechanism underlying the obesity-associated variant rs1421085 located in an intron of the FTO gene. For rs1421085, several layers of evidence were used to uncover disease mechanism. Multidimensional epigenetic data analysis highlighted adipose tissue as a potential target tissue for GWAS SNP action. Specifically, rs1421085 was found to be located in an enhancer region in adipose tissue. In addition, a distal interaction between this region and distal gene IRX3 was identified through 4C-seq in human and mouse. Irx3 knockdown in mice led to a reduction in body weight and increased energy dissipation, which were not due to a change in physical activity or appetite. CRISPR/Cas9 genome editing of the rs1421085 risk allele in primary human adipocytes restored IRX3 expression (and expression of proximal gene IRX5), activated adipocyte browning expression programs and restored thermogenesis. These several layers of evidence point to rs1421085 variants underlying a distal enhancer for IRX3/IRX5, which in turn are involved in regulation of adipocyte browning and thermogenesis. This example highlights how multiple layers of evidence will be necessary for uncovering the action of many GWAS SNPs. Specifically, the use of 3D genomics data coupled with multidimensional epigenetic information (e.g. chromatin state, open chromatin and histone mark data) constitutes a powerful tool to aid the identification of non-protein-coding GWAS SNP mechanism. The need for more research in this area is further highlighted by the fact the majority of GWAS disease-associated SNPs locate to non-protein-coding regions.
Bioinformatics provides one way to gain information on the potential function of disease-associated SNPs located in non-protein-coding regions, by integrating SNP information with multidimensional epigenetic mapping data across cell types and tissues. Initial efforts in this direction include Haploreg\textsuperscript{137}, RegulomeDB\textsuperscript{138} and FORGE\textsuperscript{139}. The concept underlying these approaches is the comparison of overlap counts between the input list and a selected background. Alternative approaches include permutation-based analysis to evaluate significance of overlap, as implemented in GOshifter\textsuperscript{140}. A more complete list of such methods can be found in Tak and Farnham, 2015\textsuperscript{141}.

1.7 Epigenomics of disease

The study of non-genetic factors in disease presents important confounders. A particular example of this is the study of DNAm variation. Detected DNAm changes between cases and controls can be confounded by cell composition effects and genetic sequence between individuals, as well as by other factors such as age and sex. It is important to consider that DNAm changes can be a consequence of disease mechanisms. EWAS are therefore subject to reverse causation\textsuperscript{18}. Evidence of causality in EWAS is thus very different to evidence of causality in GWAS (SNPs are not subject to this phenomenon of change in response to environmental influences). Inference methods (such as two step epigenetic mendelian randomisation, that uses an instrumental variable approach with genetic instruments to asses directionality of effect) are required in EWAS to test for evidence of causality\textsuperscript{142}. In short, the correct design of EWAS is challenging. Despite this, the introduction of epigenome-wide association
studies (EWAS) in 2011\textsuperscript{17} has been followed by significant progress, one of the main advances being the development of software for cell composition correction \textsuperscript{48,49,50,51}. Additional approaches to reduce confounding in study design include the use of monozygotic twins to correct for genetic variation, sorted cell types to avoid confounding by cell composition and the annotation of subject information such as sex and age in an effort to correct for these confounders\textsuperscript{143}. A large percentage of published EWAS (the number of which exceeds 250 studies\textsuperscript{9}) have correlated DNAm levels with case-control status in cross-sectional cohorts. Examples of externally replicated findings include differentially methylated positions (DMPs) for smoking behaviour\textsuperscript{144} and Alzheimer’s disease\textsuperscript{145}.

1.8 Thesis aims

Multiple genetic and epigenetic variants have been identified for a range of diseases through GWAS and EWAS. However, many of the identified variants are located in non-protein-coding regions and thus interpretation is challenging. In this thesis I have implemented both bioinformatics and experimental approaches to aid the challenging interpretation of these non-protein-coding variants. Two computational tools (FORGE2 and eFORGE) have been developed for the interpretation of GWAS and EWAS data, respectively. In addition, 4C-seq has been applied to study the three-dimensional context of non-protein-coding variants in mouse neural stem cells (mNSCs) and a range of primary human immune cells.
1.8.1 Design and implementation of FORGE2.

FORGE has provided an automated tool for the analysis of tissue-specific DNase I hotspot enrichment of GWAS SNPs in order to improve understanding of biological context and underlying disease mechanisms\textsuperscript{139}. However, a tool that extends the FORGE approach to the wide catalogue of histone mark data has not yet been implemented. My objective was to design and implement FORGE2, an automated tool that analyses the cell type-specific histone mark enrichment of GWAS SNPs. In addition, I aimed to analyse the whole GWAS catalogue in search of novel tissue-disease associations detected by this new tool (chapter 3).

1.8.2 Design and implementation of eFORGE

One way to shed light on the interpretation of non-protein-coding variants is to use epigenetic data, highlighting candidate tissues and cell types for further study. The FORGE tool has automated this analysis approach for GWAS variants. However, to date no such software has been designed for EWAS. My objective was to design and implement eFORGE (experimentally-derived Functional element Overlap analysis of ReGions from EWAS\textsuperscript{146}), a tool for the analysis of cell type-specific DNase I hotspot (and histone mark) enrichment of EWAS DMPs. Due to the radical differences between GWAS and EWAS, new approaches for background adjustment were necessary. In addition, eFORGE application across a range of different EWAS was required to assess both the potential insights and limitations of the tool (chapter 4).
1.8.3 4C-seq of neurological and autoimmune disorder-associated regions.

Some disease-associated variants act through conformation-dependent distal regulatory interactions, as exemplified in the case of FTO variant rs1421085\textsuperscript{130}. Large-scale 3D genomics efforts have been performed to map such interactions across a range of cell types and tissues\textsuperscript{112,147}. However, these efforts typically involve genome-wide approaches such as Hi-C or Capture Hi-C. Such methods lack the power of targeted approaches such as 4C-seq, which map regulatory interactions for specific genomic regions. My objective was to perform 4C-seq across a range of cell types to map the regulatory interactions of selected genetic and epigenetic non-protein-coding variants. Cell types included mNSCs and sorted primary immune cell types (CD14+, CD4+, CD8+, NK and B cells). The high-definition maps generated from this effort are intended to aid interpretation of selected neurological and autoimmune disease-associated variants respectively. Candidate variants were selected by applying FORGE2, eFORGE and additional analyses involving multidimensional epigenetic data. I then performed conformation capture on these candidate variants using 4C-seq to fine-map the 3D context of the genomic regions involved (chapter 5).
2 Materials and Methods
Materials and Methods

2.1 4C-seq of neurological and autoimmune disorder-associated regions.

2.1.1 Samples

Mouse Neural Stem Cells (mNSCs) were obtained from Suzana Hadjur’s lab at the UCL Cancer Institute. Primary human CD14+, CD4+, CD8+, NK and B cells were obtained from the Cambridge Blood Donor Center. The ethical approval for these BLUEPRINT consortium samples was issued under REC reference 12/EE/0040 by the NRES Committee East of England-Hertfordshire. Sample-specific information is described in table 2.1.

4C-seq methods shown below are based on protocols which are further detailed in van de Werken et al. (2012)\textsuperscript{148,149}.

<table>
<thead>
<tr>
<th>Donor Barcode</th>
<th>Gender</th>
<th>Cell Type</th>
<th>Cell count</th>
</tr>
</thead>
<tbody>
<tr>
<td>S01086</td>
<td>M</td>
<td>Monocytes</td>
<td>$3.2 \times 10^7$</td>
</tr>
<tr>
<td>S0104E</td>
<td>F</td>
<td>CD4 cells</td>
<td>$9.8 \times 10^7$</td>
</tr>
<tr>
<td>S0104E</td>
<td>F</td>
<td>CD8 cells</td>
<td>$5.5 \times 10^7$</td>
</tr>
<tr>
<td>S010FT</td>
<td>F</td>
<td>B cells</td>
<td>$3 \times 10^7$</td>
</tr>
<tr>
<td>S010HP</td>
<td>F</td>
<td>NK cells</td>
<td>$5.2 \times 10^7$</td>
</tr>
</tbody>
</table>

Table 2.1: BLUEPRINT samples. BLUEPRINT consortium sample IDs and gender, cell type and cell count information data for primary patient samples for which 4C-seq was performed.
2.1.2 4C-seq

2.1.2.1 Intranuclear chromatin fixation
Starting with 10 million cells, mNSCs were fixed in media with 1% formaldehyde for 10 minutes in rotation. After glycine quenching, pellets were flash-frozen for storage. BLUEPRINT samples were fixed according to the protocol listed in Javierre et al.150, as samples from both studies were part of the same preparation pipeline at Mattia Frontini’s group in Cambridge. Subsequent steps in 4C library preparation were the same for mNSCs and BLUEPRINT samples unless indicated otherwise.

2.1.2.2 Cell lysis and DpnII digestion
Cells were thawed and lysed for 20 minutes on ice, and were subsequently digested overnight using 750 U DpnII (New England Biolabs, in buffer with 50 mM Bis-Tris-HCl, 100 mM NaCl, 10 mM MgCl₂, 1 mM DTT, pH 6.0). The next day the DpnII buffer and enzyme were replaced and a second overnight digestion performed. On the third day the digestion profile was checked by electrophoresis on an agarose gel.

2.1.2.3 Intranuclear ligation and chromatin decrosslinking
Nuclei were spun down and resuspended in buffer with 1600 U of T4 DNA ligase. Ligation was performed overnight at 16°C without rotating. The next day the liglation profile was checked by electrophoresis on an agarose gel. Proteinase K was added, and samples were incubated overnight at 65°C to reverse crosslinks. The next day samples were treated with RNase A at 37°C for two hours to remove RNA. Phenol/chloroform extraction ensued and DNA was resuspended in TE buffer (ThermoFisher Scientific, 10 mM Tris-HCl, 0.1 mM EDTA, pH 8.0).
2.1.2.4 *Csp6I digestion*

The concentration of purified ligated DNA was measured according to the manufacturer's protocol (Qubit hs DNA assay, Invitrogen), and DNA digestion was performed overnight in a total volume of 300 microlitres using 120 U Csp6I (ThermoFisher scientific, in buffer with 10 mM Tris-HCl, 10 mM MgCl₂, 0.1 mg/mL BSA, pH 7.5). The digestion profile was checked on the next day by electrophoresis on an agarose gel. After confirmation of the correct digestion profile, enzyme was heat-deactivated for 20 minutes at 65°C. DNA was obtained through a second phenol/chloroform extraction and resuspended in nuclease-free water (Ambion).

2.1.2.5 *Intramolecular ligation*

A high-volume ligation reaction was performed using 1600 U of T4 DNA ligase, incubating samples overnight at 16°C without rotation. On the next day a third phenol/chloroform extraction was performed, and samples were resuspended in TE buffer.

2.1.2.6 *Target selection*

GWAS and EWAS regions were prioritised for 4C-seq through multidimensional epigenetic data analysis incorporating several bioinformatics tools. For this analysis data was included for DNase I hotspots, expression Quantitative Trait Loci (eQTL), FANTOM5 enhancer annotations, promoter Capture Hi-C contacts, genomic context and disease association (see figures 2.1 and 2.2). Data underlying the selection of each region is shown in table 2.2 for mNSC loci and table 2.3 and figure 2.3 for immune cell loci.
Figure 2.1: Process for selecting neurological disorder-associated variants.

GWAS and EWAS variants were selected depending on several factors including DNase I hypersensitivity in human brain, previous contacts in PCHi-C, eQTL/eQTM status in brain (eQTM data from Gutierrez-Arcelus et al., 2013\textsuperscript{151}) and hg19-mm10 sequence homology.
Figure 2.2: Process for selecting immunity-associated loci. GWAS variants were selected depending on several factors including FANTOM5 enhancer status in blood, FANTOM5 cell-type enhancer specificity, previous contacts in PCHi-C and eQTL status in blood.
<table>
<thead>
<tr>
<th>ID</th>
<th>Previous PCHi-C data</th>
<th>eQTL/eQTM and target gene</th>
<th>Genomic context in human</th>
<th>Genomic context in mouse</th>
<th>Disease association</th>
</tr>
</thead>
<tbody>
<tr>
<td>rs7618915</td>
<td>No</td>
<td>ITIH4 (brain).</td>
<td>1kb away from the</td>
<td>1kb away from the</td>
<td>Bipolar disorder-associated SNP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>promoter of PPM1M</td>
<td>promoter of Ppm1m</td>
<td></td>
</tr>
<tr>
<td>rs8012</td>
<td>No</td>
<td>GCDH (brain)</td>
<td>Last exon of GCDH</td>
<td>Last exon of Gcdh</td>
<td>Blood metabolite level-associated SNP</td>
</tr>
<tr>
<td>rs1009014</td>
<td>Yes (blood)</td>
<td>SYNJ2 (cerebellum)</td>
<td>In an intron of SYNJ2</td>
<td>In an intron of Synj2</td>
<td>None, meQTL for SYNJ2 (Marzi et al., 2016).</td>
</tr>
<tr>
<td>rs548181</td>
<td>Yes (blood)</td>
<td>CCDC15 CHEK1, DDX25,</td>
<td>In an intron of STT3A</td>
<td>In an intron of</td>
<td>Several (including bipolar disorder and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FEZ1 HEPACAM, HEPN1,</td>
<td>ANTISENSE RNA 1</td>
<td>AK018988 (potential</td>
<td>schizophrenia).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PUS3 ROBO3, SRPR</td>
<td>(STT3A-AS1).</td>
<td>homologue to STT3A-</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS1)</td>
<td></td>
</tr>
<tr>
<td>cg05066959</td>
<td>Yes (blood)</td>
<td>Not known if eQTM</td>
<td>In an intron of ANK1 and</td>
<td>In an intron of Ank1, but</td>
<td>Alzheimer’s disease-associated DMP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NKK6-3</td>
<td>not Nkx6-3.</td>
<td></td>
</tr>
<tr>
<td>cg02672452</td>
<td>No.</td>
<td>Not known if eQTM</td>
<td>In an intron of non-</td>
<td>Between Otop1 and</td>
<td>Alzheimer’s disease-associated DMP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>coding RNA LOC654342.</td>
<td>Drd5</td>
<td></td>
</tr>
</tbody>
</table>
Table 2.2: Mouse target loci. Multidimensional epigenetic analysis results for prioritised neurological disease associated-GWAS/EWAS regions including genomic context, eQTL status and previous promoter capture Hi-C contacts (using the CHiCP tool\textsuperscript{152}, https://www.chicp.org/). Disease association category is also shown. One non-GWA SNV was included due to evidence suggesting interaction between genetics and epigenetics, specifically rs1009014 was found to determine allele-specific DNAm in cerebellum in a recent study by Marzi et al.\textsuperscript{153}. eQTL data were obtained from Braineac and GTEx databases\textsuperscript{154,155}. All these loci were found to overlap DNase I hypersensitive sites from either Epigenomics Roadmap or ENCODE consortia using the FORGE and eFORGE tools\textsuperscript{139,146}.
<table>
<thead>
<tr>
<th>Enhancer position (hg19)</th>
<th>Cell specificity (FANTOM5 enhancer data)</th>
<th>CHiCP in CD34</th>
<th>eQTL (Walsh et al.)</th>
<th>GTEx Whole Blood</th>
<th>Gene</th>
<th>GTEx Spleen</th>
<th>Gene</th>
<th>Genomic context</th>
<th>Immune disease/trait association by GWAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr1:90022562-90022945</td>
<td>Monocyte</td>
<td>LRRCD8</td>
<td>LRRCD8B</td>
<td>rs4658279</td>
<td>LRRCBB</td>
<td>rs4658279</td>
<td>LRRCBB</td>
<td>Intron of LRRCD8B</td>
<td>Neutropenia/leucopenia after chemotherapy</td>
</tr>
<tr>
<td>chr17:37912171-37912498</td>
<td>B cell, CD8+ cell, NK cell</td>
<td>None</td>
<td>GSDMB, ORMDL3, PGAP3</td>
<td>rs77013147</td>
<td>GSDMA</td>
<td>rs12946510</td>
<td>GSDMB, ORMDL3</td>
<td>Between GRB7 and IKZF3</td>
<td>Ulcerative colitis, Crohn's disease</td>
</tr>
<tr>
<td>chr22:37258335-37258758</td>
<td>Monocyte</td>
<td>NCF4 and PVALB</td>
<td>rs4821544</td>
<td>PVALB</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>Intron of NCF4</td>
<td>Atopic dermatitis</td>
</tr>
<tr>
<td>chr5:131430055-131430830</td>
<td>Monocyte and NK cell</td>
<td>IL3 and P4HA2-AS1</td>
<td>rs657075</td>
<td>AC0342</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>Between CSF2 and P4HA2</td>
<td>Rheumatoid arthritis</td>
</tr>
<tr>
<td>chr6:167534229-167534357</td>
<td>B cell</td>
<td>CCR6 and RNASET2</td>
<td>rs3093023</td>
<td>RNASET2</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>Intron of CCR6</td>
<td>Rheumatoid arthritis</td>
</tr>
<tr>
<td>chr9:123652531-123653088</td>
<td>None</td>
<td>TRAF1, PSMD5-AS1</td>
<td>rs881375</td>
<td>PSMD5-AS1</td>
<td>rs881375</td>
<td>PSMD5-AS1</td>
<td>Between PHF19, TRAF1</td>
<td>Rheumatoid arthritis</td>
<td></td>
</tr>
</tbody>
</table>
**Table 2.3: Human target loci.** Multidimensional epigenetic analysis results for prioritised human immune cell regions including cell type-specific enhancer status from FANTOM5, previous promoter capture Hi-C contacts (using the CHiCP tool\textsuperscript{152}, \url{https://www.chicp.org/}), eQTL status from a study by Walsh et al. (2016)\textsuperscript{156} and the GTEx consortium (in whole blood and spleen)\textsuperscript{155}, genomic context and immune disease/trait GWAS association. All these loci were found to overlap DNase I hypersensitive sites from either Epigenomics Roadmap or ENCODE consortia using the FORGE tool\textsuperscript{139}. All of these loci were also found to overlap H3K4me1 broadPeaks from the Epigenomics Roadmap consortium using the FORGE2 tool (**figure 2.3**), which I developed as part of this doctoral project.
Figure 2.3: FORGE2 H3K4me1 analysis of human target loci. Analysis was performed on GTEx Whole Blood eQTL SNPs prioritised for 4C-seq analysis in BLUEPRINT consortium primary samples. H3K4me1 is a mark enriched for enhancer regions, and even from this small set of SNPs a clear immune cell enrichment signature can be noted. SNPs rs4658279, rs77013147, rs4821544, rs657075, rs3093023 and rs881375 were used for analysis. All these SNPs were found to overlap H3K4me1 broadPeaks in B cells, T cells and haematopoietic stem cells.
2.1.2.7 Library amplification

DNA concentration was measured according to the manufacturer's protocol (Qubit hs DNA assay, Invitrogen). I ordered oligonucleotides containing pre-designed PCR primers from van de Werken et al. (2012)\textsuperscript{149} (http://compgenomics.weizmann.ac.il/tanay/?page_id=367) for target regions in mNSCs (Table 2.4, Appendices Table S1), and sorted blood cells (Appendices Table S2). PCR reactions were set using Expand polymerase (Roche) and 100 ng of 4C template. PCR conditions were 94°C 3 minutes, (94°C 10 seconds, 55°C 1 minute, 68°C 3 minutes, 30 cycles) and 68°C 10 minutes. PCR product was purified according to the manufacturer's instructions (Roche High Pure PCR product purification kit) to remove primer dimers. 4C libraries were then concentrated down to 15 microlitres using Ampure XP beads according to the manufacturer's instructions. PCR amplification was checked on the Bioanalyser using the HS DNA kit (Agilent).

<table>
<thead>
<tr>
<th>chr (hg19)</th>
<th>start (hg19)</th>
<th>end (hg19)</th>
<th>chr (mm10)</th>
<th>start (mm10)</th>
<th>end (mm10)</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr3</td>
<td>52279344</td>
<td>52279844</td>
<td>chr9</td>
<td>106199189</td>
<td>106199646</td>
<td>rs7618915</td>
</tr>
<tr>
<td>chr19</td>
<td>13010270</td>
<td>13010770</td>
<td>chr8</td>
<td>84886598</td>
<td>84887064</td>
<td>rs8012</td>
</tr>
<tr>
<td>chr6</td>
<td>158486893</td>
<td>158487393</td>
<td>chr17</td>
<td>6012819</td>
<td>6013423</td>
<td>rs1009014</td>
</tr>
<tr>
<td>chr11</td>
<td>125461459</td>
<td>125461959</td>
<td>chr9</td>
<td>36768370</td>
<td>36768857</td>
<td>rs548181</td>
</tr>
<tr>
<td>chr8</td>
<td>41519308</td>
<td>41519309</td>
<td>chr8</td>
<td>23141313</td>
<td>23141314</td>
<td>cg05066959</td>
</tr>
<tr>
<td>chr2</td>
<td>91818189</td>
<td>91818190</td>
<td>chr5</td>
<td>38308967</td>
<td>38308968</td>
<td>cg02672452</td>
</tr>
</tbody>
</table>

Table 2.4: Human-mouse homologous regions. List of prioritised human GWAS/EWAS regions and corresponding homologous mouse loci identified for 4C-seq analysis. GWAS RSIDs and EWAS 450k array cg probe numbers are also
Materials and Methods

included. Human regions were lifted over to the mouse genome using UCSC LiftOver (minimum ratio of remapping bases set to 0.95) \(^{157}\).

\[2.1.2.8 \quad 4C \text{ library sequencing}\]

Libraries were combined into a single vial and the mix was quantified using the Qubit method (Qubit hs DNA assay, Invitrogen). The theoretical molarity of the mix was calculated using the formula:

\[
[\text{Library mix [c]} \text{ microgram/microlitre : (650x600)}] \times 10^9 = \text{nM library mix}
\]

The molarity was adjusted to 2nM, and the DNA was then denatured for 5 minutes by adding 10 microlitres of 0.2 N NaOH, and taken for sequencing at a final molarity of 10pM.

Sequencing for mouse regions homologous to human GWAS/EWAS regions was performed on a MiSeq using a v2 (300 cycle) single-end sequencing kit. The sequencing process was subsequently validated using a v3 (150 cycle) single-end MiSeq kit. Finally, for the multiplexed regions across the BLUEPRINT primary samples (including CD14+, CD4+, CD8+, B cell and NK cells) a HiSeq run was performed using the following kits:

- One unit of HiSeq Rapid SBS Kit v2 (50 cycles): 50 cycle SBS kit for HiSeq 2500.
- Two units of HiSeq Rapid PE Cluster Kit v2: Paired-end cluster generation kit for HiSeq 2500.
2.1.2.9 4C bioinformatics analysis

I applied the 4Cseqpipe software described in van de Werken et al. (2012)\textsuperscript{149} for analysis of 4C-seq data. 4Cseqpipe was downloaded from http://compgenomics.weizmann.ac.il/tanay/. This pipeline maps 4C-seq libraries to the genome (hg19/mm10), performs data normalisation and subsequently generates near-cis domainograms to visualise 3D-genomic contacts. For analyses I used 4Cseqpipe version 0.7 on the UCL computer science department cluster with trimming settings set to 50 bp.

2.2 eFORGE methods

2.2.1 Introduction

eFORGE (experimentally-derived Functional element Overlap analysis of ReGions from EWAS) is a computational tool that analyses whether there is cell type-specific enrichment of overlap for DMPs from a given EWAS compared to matched background sites from the EWAS array.

In short, eFORGE analyses the cell type-specific regulatory component of a set of EWAS DMPs. Enrichment is computed on a per cell type basis, since hotspots and histone mark broadPeaks vary between different cell types, and hence can expose cell type-specific signals of enrichment for the given test DMP set. This can reveal the regulatory sites underlying the EWAS signal, and, in addition, detect cell composition effects.
The functional elements considered in eFORGE are DNase I hotspots from the ENCODE, Epigenomics Roadmap and BLUEPRINT consortia, and histone mark broadPeaks from the Epigenomics Roadmap consortium. Hotspots (areas of enriched DNase I cleavage) were generated using the HOTSPOT program, and histone mark broadPeaks were generated by the MACS2 method. BroadPeaks correspond to broader domains of enrichment for a given histone mark, as opposed to narrowPeaks which are less broad, contiguous regions of enrichment. In a similar way, hotspots correspond to wider domains of DNase I hypersensitivity when compared to DNase I peaks, which are 150 basepair regions with a very high DNase I density relative to background. Hotspots have been shown to reveal more tissue-specific enrichment signal than peaks\textsuperscript{139}.

2.2.2 Sample processing

DNase I hotspots were obtained from three consortia: ENCODE, Roadmap Epigenomics and BLUEPRINT. ENCODE data (Thurman et al., 2012) were taken from the ENCODE European Bioinformatics Institute (EBI) ftp site (all URLs for sample processing listed in table 2.5). BLUEPRINT data were downloaded from the Genomatix interface. For Roadmap Epigenomics 2012 data DNase I sequencing tag alignments were downloaded from the Genboree webpage. Only files within the data use embargo agreement were used, these files belong to the section of Gene Expression Omnibus (GEO) accession number GSE18927. The alignments were processed using the Hotspot method\textsuperscript{158,159}, applying the default parameters. For consolidated Roadmap Epigenomics 2015 data, DNase I
Materials and Methods

hotspots and BroadPeak Histone mark files were downloaded from the Roadmap Epigenomics 2015 webpage. Cell and tissue names were assigned to each dataset using consortium data or custom perl scripts from the decodings that can be obtained from the ENCODE Data Coordination Centre tables or sample group SAMEG31306 from the BioSamples database.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENCODE (EBI ftp site)</td>
<td>ftp://ftp.ebi.ac.uk/pub/databases/ensembl/encode/integration_data_</td>
</tr>
<tr>
<td></td>
<td>jan2011/byDataType/openchrom/jan2011/combined_hotspots/</td>
</tr>
<tr>
<td>BLUEPRINT (Genomatix)</td>
<td><a href="https://blueprint.genomatix.de/grid/experiments/browse">https://blueprint.genomatix.de/grid/experiments/browse</a></td>
</tr>
<tr>
<td>Roadmap 2012 (Genboree)</td>
<td><a href="http://www.genboree.org/EdaccData/Current-Release/experiment-sample/">http://www.genboree.org/EdaccData/Current-Release/experiment-sample/</a></td>
</tr>
<tr>
<td>Hotspot method</td>
<td><a href="http://www.uwencode.org/proj/hotspot/">http://www.uwencode.org/proj/hotspot/</a></td>
</tr>
<tr>
<td>Roadmap Epigenomics 2015</td>
<td><a href="http://egg2.wustl.edu/roadmap/data/byFileType/peaks/consolidated/">http://egg2.wustl.edu/roadmap/data/byFileType/peaks/consolidated/</a></td>
</tr>
<tr>
<td></td>
<td>broadPeak/</td>
</tr>
<tr>
<td>ENCODE Data Coordination Centre</td>
<td><a href="https://genome.ucsc.edu/encode/cellTypes.html">https://genome.ucsc.edu/encode/cellTypes.html</a></td>
</tr>
<tr>
<td>BioSamples</td>
<td><a href="http://www.ebi.ac.uk/biosamples/">http://www.ebi.ac.uk/biosamples/</a></td>
</tr>
</tbody>
</table>

**Table 2.5: eFORGE sample processing.** URLs for eFORGE sample annotation and data processing.
2.2.3 Preparation of sample overlaps

450k array probes were overlapped with DNase I hotspots and histone mark broadPeaks using the bedtools tool\textsuperscript{160}. Overlaps were stored in an SQLite database (http://www.sqlite.org) that was organised by datasets. Data was coded in bitstrings: for each probe I generated a binary string. In this string 1 represented the presence of an overlap with a hotspot in each dataset (either Roadmap, BLUEPRINT or ENCODE). For each string 0 represented the absence of such an overlap. For 27k analysis the 94\% of overlapping probes between both arrays were used. For location information I used the HumanMethylation450 v1.2 manifest file (https://support.illumina.com/array/array_kits/infinium_humanmethylation450_beadchip_kit/downloads.html).

2.2.4 Analysis strategy

eFORGE was designed in order to analyse cell type-specific enrichment signal for a list of 450k array probes. This can be performed by the following data processing strategy, as shown below in figure 2.4:
Figure 2.4: eFORGE analysis strategy. A list of top probes from an EWAS is input into eFORGE. The software then creates 1000 lists of random probes from an array of choice (27k or 450k). Background probes are selected for similar annotations to the input probes regarding gene relationship and CpG island relationship. The overlap of the 1000 random lists with DNase I hotspot data across cell types is used to create a background distribution. The overlap of the input list is then tested for enrichment against this background through a binominal test. The –log10 (p-value) of the EWAS list compared to the background lists is represented to visualise enrichment. Figure obtained from Breeze et al. (2016)\textsuperscript{146}.

The eFORGE analysis strategy evaluates the input probe list by comparison with 1000 randomly selected background probe lists. eFORGE selects similar background probes to the input list through annotation-based matching (e.g. if one probe from the input list is located in a promoter, then its matching probe in
each of the 1000 background lists is also in a promoter). DNase I hotspot enrichment for a particular tissue is computed by comparing the DNase I hotspot overlap of the input list to the DNase I hotspot overlap of the 1000 background lists. This process is also applied in the analysis of histone mark broadPeaks.

### 2.2.5 Annotation categories

Two main annotation levels were employed for classing probes: “gene annotation” (including categories 1stExon, 3’UTR, 5’UTR, Body, IGR, TSS1500, TSS200) and CpG island annotation (with categories Island, Shore_Shelf, NA or “open sea”). Probes were then classed into 21 bin categories, which result from all possible combinations of these two levels of annotation. The “Shelf” and “Shore” categories (from CpG island annotation) were merged in order to ensure that each bin category contained more than 1000 probes. This is important given that small bin sizes could introduce bias as background selection would be dependent on the overlaps of a few probe lists. The aforementioned 450k array Illumina annotation HumanMethylation450 v1.2 manifest file ([https://support.illumina.com/array/array_kits/infinium_humanmethylation450_beadchip_kit/downloads.html](https://support.illumina.com/array/array_kits/infinium_humanmethylation450_beadchip_kit/downloads.html)) was used to create these data categories (tables 2.6 and 2.7).
<table>
<thead>
<tr>
<th>Gene annotation</th>
<th>CpG island annotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intergenic region (IGR)</td>
<td>Island IGR</td>
</tr>
<tr>
<td></td>
<td>Shore_Shelf IGR</td>
</tr>
<tr>
<td></td>
<td>NA IGR</td>
</tr>
<tr>
<td>Within the 200 bp prior to a transcription start site (TSS200)</td>
<td>Island TSS200</td>
</tr>
<tr>
<td></td>
<td>Shore_Shelf TSS200</td>
</tr>
<tr>
<td></td>
<td>NA TSS200</td>
</tr>
<tr>
<td>1st Exon</td>
<td>Island 1st Exon</td>
</tr>
<tr>
<td></td>
<td>Shore_Shelf 1st Exon</td>
</tr>
<tr>
<td></td>
<td>NA 1st Exon</td>
</tr>
<tr>
<td>Within the 1500 bp prior to a transcription start site (TSS1500)</td>
<td>Island TSS1500</td>
</tr>
<tr>
<td></td>
<td>Shore_Shelf TSS1500</td>
</tr>
<tr>
<td></td>
<td>NA TSS1500</td>
</tr>
<tr>
<td>3’ untranslated region (3’ UTR)</td>
<td>Island 3’ UTR</td>
</tr>
<tr>
<td></td>
<td>Shore_Shelf 3’ UTR</td>
</tr>
<tr>
<td></td>
<td>NA 3’ UTR</td>
</tr>
<tr>
<td>5’ untranslated region (5’ UTR)</td>
<td>Island 5’ UTR</td>
</tr>
<tr>
<td></td>
<td>Shore_Shelf 5’ UTR</td>
</tr>
<tr>
<td></td>
<td>NA 5’ UTR</td>
</tr>
<tr>
<td>Gene Body</td>
<td>Island gene body</td>
</tr>
<tr>
<td></td>
<td>Shore_Shelf gene body</td>
</tr>
<tr>
<td></td>
<td>NA gene body</td>
</tr>
</tbody>
</table>

**Table 2.6: eFORGE background probe matching.** Background probe matching in eFORGE is performed using two levels of data categories. The first level is based on gene-centric annotation and the second level corresponds to CpG island annotation.
Table 2.7: eFORGE background bin subcategories. eFORGE background bin subcategories are listed corresponding to the two levels of annotation. Numbers of probes within each category are indicated.

2.2.6 Proximity filtering

One of the other issues in FORGE is the problem of analysing SNPs that are in close proximity, which could mean testing variants with strong Linkage Disequilibrium (LD) against the same DNase I hotspot. This is a problem as it
could result in higher tissue-specific enrichment signal than is real (due to the LD several variants may be associated with a given trait or disease when only one SNP underlies disease mechanism). Therefore FORGE includes an LD filter in its code, and all SNPs in strong LD ($r^2 > 0.8$) are filtered to leave one SNP for each haplotype. For eFORGE there is a similar problem with DNAm correlation in proximal regions. This correlation means that I am facing a similar problem to the LD problem that FORGE faces. Therefore a proximity filter of 1kb was included in eFORGE code, which, after selecting one EWAS probe, cannot select another probe within 1kb of that probe (500 bp upstream and downstream). The choice of selecting 1kb as a limit for filtering was based on previous studies showing strong correlation of DNA methylation levels between CpGs fewer than 1kb apart\textsuperscript{38,161}.

### 2.2.7 eFORGE code structure

The eFORGE tool consists of a series of Perl scripts that use SQLite databases and make R calls for plotting. For the development of the tool I had to first learn Perl and study the SQLite database call nomenclature.

**Figure 2.5** displays a schematic of the files included in eFORGE, and **figure 2.6** shows a schematic of the code calls performed by the tool.
Figure 2.5: eFORGE directory schematic. This schematic shows the directory structure required for an eFORGE installation. eforge.pl is the main script, which also lists the eFORGE documentation (contained in a Perl Pod). eforge.ini contains a directory address which is used by the main script (some eFORGE files can be placed in another directory if eforge.ini is modified). The eForge folder contains the Perl modules eForge.pm, ePlot.pm and eStats.pm, which contain the functions accessed by the main script. eforge.db contains the SQLite database and mvp_bins contains the Perl hash required for background selection of similar probes (according to the two levels of annotation) and proximity filtering.
**Figure 2.6: eFORGE code schematic.** This diagram shows the code calls made by eFORGE. A Perl script performs the main processing of the data (supported by several Perl modules), and external calls are made in SQLite (green) for database requests and R (red) for plotting.

**2.2.8 eFORGE input**

DMPs can be input into eFORGE in two formats: Illumina 450k/27k probe IDs, or BED format (0 based BED format, with chromosome location input as chrN, corresponding to the human genome assembly GRCh37 genomic coordinates). eFORGE can retrieve probe IDs if genome location is input in this format. For analysis I suggest a maximum probe list size of 1000 and a minimum of 20 probes. Any probes that are not present on the 450k array (or on the 450k array-27k array shared set) are excluded from the analysis.
2.2.9 eFORGE outputs

eFORGE generates tables and graphic descriptions of the overlap enrichment for the test DMPs for each DNase I hotspot sample. A tab-separated values (TSV) file is output, including the following columns: z-score, p-value, cell, tissue, hotspot sample filename, overlapping DMPs, sample GEO accession number, and BY adjusted q-value. The Datatables (https://datatables.net/) plug-in for the jQuery Javascript library is used to generate an interactive table containing these data. Datatables is accessed through the rCharts package (http://ramnath.github.io/rCharts/).

For eFORGE chart generation, each of the graphical outputs presents the -log 10 (binomial p-value) by cell sample. PDF graphics are generated using base R graphics (http://www.r-project.org). The interactive JavaScript graphic is generated using the rCharts package (http://ramnath.github.io/rCharts/), which interfaces with the dimple d3 libraries (http://dimplejs.org). Cells are grouped alphabetically within each tissue in the PDF and the interactive graphic (tissues also follow alphabetical order). For the interactive graphic replicate samples are stacked at the same X coordinate. In addition, blue (q-value > 0.05), pink (q-value < 0.05), and red (q-value < 0.01) are consistently used to colour BY-corrected q-value results in each of the graphics.
2.2.10 Multiple testing correction and false positive rates

One of the main differences in multiple testing correction between eFORGE and FORGE is that FORGE uses a Bonferroni correction, therefore dividing the p-values by the number of different tissues (and not cell types) present within the dataset. This does not take into account DNase I hotspot correlation both within a tissue and between tissues, and does not permit statements about the enrichment of a specific cell type to be made, as correction is tissue-based and not cell type-based.

For eFORGE I consider that it is important to be able to make statements about the enrichment of a specific cell type, and therefore I require a cell type-level correction that does not assume independence between cell types (as there is a correlation in DNase I hotspot/histone mark profiles that occurs both within a tissue and between tissues). Approaches to control for multiple testing in this context are found in the False Discovery Rate (FDR) approach. The FDR approach does not focus on keeping an alpha level of significance of 0.05 in each test but rather on controlling the level of false positives below a certain threshold. Within the FDR approach the appropriate multiple testing correction in this case is a Benjamini-Yekutieli (BY) correction, which controls the false discovery rate under positive dependence assumptions\(^{162}\). BY was compared with the FDR-based Benjamini-Hochberg correction (which is more appropriate for independent tests). For this comparison 5000 probe sets were tested containing between 5 and 100 DMPs on eFORGE. This analysis was performed using both BH and BY for multiple testing correction. Given that each of the 5000
DMP sets contained 299 samples (Roadmap 2012 data), the total number of sample-tests is 1,495,000. For all these tests I observed 1 false positive for BH, and 0 false positives for BY, for a significance level of q-value<0.01. False positives at a q-value< 0.05 were estimated to be around 0.36% for 5000 tests applied across ENCODE and Roadmap (BY correction). However, for analyses with eFORGE I consider a q-value<0.05 as only an intermediate level of significance. In addition, these false positive rates do not show a clear tendency with the size of the probe sets (tables 2.8 and 2.9). In regard to the variation in numbers for false positives (at a level q<0.05) in both tables, further extensive testing (2,990,000 sample tests, Epigenomics Roadmap data) revealed more homogeneous results (1598 ± 54 sample-test false positives at q<0.05 for set sizes 10-100). This suggests that the aforementioned variation is due to the number of tests performed. Furthermore, these results suggest that most false positives at q-value<0.05 come from a low number of random probe lists that present borderline enrichment for many tissues/cell lines. Taken together, these results suggest that it is highly improbable to get cell type-specific enrichment with random probe lists in eFORGE.

<table>
<thead>
<tr>
<th>SetSize</th>
<th>N</th>
<th>p&lt;0.05</th>
<th>p&lt;0.01</th>
<th>q&lt;0.05</th>
<th>q&lt;0.01</th>
<th>F</th>
<th>p&lt;0.05</th>
<th>p&lt;0.01</th>
<th>q&lt;0.05</th>
<th>q&lt;0.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>625000</td>
<td>2418</td>
<td>190</td>
<td>0</td>
<td>0</td>
<td>5000</td>
<td>936</td>
<td>104</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>625000</td>
<td>23686</td>
<td>3602</td>
<td>23</td>
<td>0</td>
<td>5000</td>
<td>2829</td>
<td>943</td>
<td>11</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>625000</td>
<td>5721</td>
<td>597</td>
<td>0</td>
<td>0</td>
<td>5000</td>
<td>1338</td>
<td>283</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>625000</td>
<td>9313</td>
<td>1226</td>
<td>32</td>
<td>0</td>
<td>5000</td>
<td>1887</td>
<td>426</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>30</td>
<td>625000</td>
<td>13487</td>
<td>1591</td>
<td>4</td>
<td>0</td>
<td>5000</td>
<td>2274</td>
<td>542</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>40</td>
<td>625000</td>
<td>10877</td>
<td>1356</td>
<td>6</td>
<td>0</td>
<td>5000</td>
<td>1999</td>
<td>475</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>50</td>
<td>625000</td>
<td>13714</td>
<td>1554</td>
<td>16</td>
<td>0</td>
<td>5000</td>
<td>2222</td>
<td>570</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>625000</td>
<td>12973</td>
<td>1593</td>
<td>1</td>
<td>0</td>
<td>5000</td>
<td>2229</td>
<td>554</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 2.8: eFORGE ENCODE false positives. False positive results after testing 5000 random probe sets on ENCODE data. N indicates number of sample tests. F indicates number of tool tests.
Materials and Methods

<table>
<thead>
<tr>
<th>SetSize</th>
<th>N</th>
<th>p&lt;0.05</th>
<th>p&lt;0.01</th>
<th>q&lt;0.05</th>
<th>q&lt;0.01</th>
<th>F</th>
<th>p&lt;0.05</th>
<th>p&lt;0.01</th>
<th>q&lt;0.05</th>
<th>q&lt;0.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1495000</td>
<td>4161</td>
<td>313</td>
<td>0</td>
<td>0</td>
<td>5000</td>
<td>1013</td>
<td>142</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>1495000</td>
<td>54069</td>
<td>7930</td>
<td>264</td>
<td>0</td>
<td>5000</td>
<td>2998</td>
<td>1065</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>1495000</td>
<td>10519</td>
<td>904</td>
<td>0</td>
<td>0</td>
<td>5000</td>
<td>1544</td>
<td>316</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>1495000</td>
<td>16263</td>
<td>1570</td>
<td>1</td>
<td>0</td>
<td>5000</td>
<td>1894</td>
<td>431</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>30</td>
<td>1495000</td>
<td>29526</td>
<td>3279</td>
<td>0</td>
<td>0</td>
<td>5000</td>
<td>2214</td>
<td>602</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>40</td>
<td>1495000</td>
<td>22855</td>
<td>2557</td>
<td>129</td>
<td>0</td>
<td>5000</td>
<td>1997</td>
<td>496</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>50</td>
<td>1495000</td>
<td>30460</td>
<td>3430</td>
<td>1</td>
<td>0</td>
<td>5000</td>
<td>2341</td>
<td>672</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>100</td>
<td>1495000</td>
<td>23173</td>
<td>2480</td>
<td>1</td>
<td>0</td>
<td>5000</td>
<td>2051</td>
<td>510</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 2.9: eFORGE Epigenomics Roadmap false positives. False positive results after testing 5000 random probe sets on Epigenomics Roadmap data. N indicates number of sample-tests. F indicates number of tool tests.

2.2.11 Database extension and generation

Given the need for continuity and tool extensibility in bioinformatics, a procedure was designed for generating the eFORGE database from consortium data. All the code to do this is openly available on GitHub (at https://github.com/charlesbreeze/eFORGE/blob/master/database/README.txt with examples given). To extend the eFORGE database, bitstrings from a new consortium or project can be added as new tables to the SQLite database, which, in contrast with the FORGE database, is modular. Indications are also provided for the production of bitstring tables from raw data from a new consortium.

2.2.12 tDMP and cDMP analysis

Analysed tissue-specific Differentially Methylated Positions (tDMPs) were provided by Queen Mary University of London (QMUL) researcher Dr Robert
Lowe, and were obtained through the following procedure. Normalised data were downloaded for tissues with at least 50 samples in Marmal-aid\textsuperscript{163}. Tissue-specific DNAm differences were initially called using the dmpFinder function from the minfi package\textsuperscript{164}. In order to do this two categories were defined: data for the tissue of interest were contained in group 1, and data for all other tissues were contained in group 2. In order to visually inspect the data the profiles of the top 2 probes were observed for each of the samples. If a sample was found to be closer to the mean of group 2 than group 1 the sample was removed. From the remaining data 50 samples for each tissue were randomly selected. Differences were then called using dmpFinder as described above. Analysed sample IDs are listed in table S3 (see Appendices).

For cell type-specific Differentially Methylated Position (cDMP) obtention I used data from Jaffe and Irizarry, 2014\textsuperscript{165}. In order to define a cDMP I used two criteria: first the DNAm beta value for a probe in the cell type of interest should be lower than that of other cell types, second, the difference between this DNA methylation value and the next closest DNA methylation value should be greater than 0.4.

\textbf{2.2.13 Source code}

eFORGE source code can be downloaded from GitHub at https://github.com/charlesbreeze/eFORGE. eFORGE installation and analysis have been performed on Mac OS X 10.8.4, 10.10.1 and Red Hat Linux. For eFORGE installation download of the eforge.db database and the background selection
hash table is required, and this can be done from [http://eforge.cs.ucl.ac.uk/?download](http://eforge.cs.ucl.ac.uk/?download). eFORGE is also available as a web tool ([http://eforge.cs.ucl.ac.uk/](http://eforge.cs.ucl.ac.uk/)). The web page includes documentation on eFORGE code and the different data analysis options available.

### 2.3 FORGE2 methods

#### 2.3.1 Introduction

The FORGE2 ([Functional element Overlap analysis of Regions from GWAS Experiments 2](http://eforge.cs.ucl.ac.uk/)) tool performs a Functional Overlap analysis on histone mark broadPeaks to identify cell type-specific signal for a given set of GWAS SNPs.

The main aim of FORGE2 is to analyse the cell type-specific regulatory component of a set of GWAS SNPs. FORGE2 takes a set of SNPs, such as those variants from a GWAS that are above the genome-wide significance threshold, and detects whether there is enrichment of overlap in regulatory elements compared to matched background SNPs from a similar array technology. This analysis is repeated for each cell type, since histone mark broadPeaks vary between different cell types. This cell type-specificity of epigenetic marks can reveal associations between cell types and the test SNP set through FORGE2 enrichment patterns. This enrichment can thus uncover the regulatory sites underlying GWAS signal. It may also help to confirm GWAS findings where a cell type-specific mechanism is projected or known for a specific phenotype. In
addition, novel cell type involvements that were previously unknown can also be revealed.

Epigenetic data considered for the initial implementation of FORGE2 include histone mark broadPeaks from the Epigenomics Roadmap project generated by the MACS2 method. BroadPeaks correspond to broader domains of enrichment for a given histone mark, as opposed to narrowPeaks which are less broad, contiguous regions of enrichment. Previous functional overlap analysis tools such as FORGE\textsuperscript{139} have focused on other epigenetic datasets, such as DNase I hotspots. To further explore the associations obtained through FORGE hotspot analyses it is necessary to separate the different classes of genomic elements driving the observed GWAS-hotspot enrichment (e.g. promoters, enhancers). One way to do this is by analysing histone mark data. I have developed FORGE2 to explore this wide and rich dataset. Thus, in addition to computational and algorithmic advances when compared to FORGE, FORGE2 extends functional overlap analyses to histone mark data.

2.3.2 Sample processing

I obtained histone mark broadPeak files for H3K4me1, H3K4me3, H3K27me3, H3K9me3 and H3K36me3 from http://egg2.wustl.edu/roadmap/data/byFileType/peaks/consolidated/broadPeak/. The Epigenomics Roadmap consortium had previously generated all files using MACS2. Cell and Tissue assignments were obtained using labels from the Epigenomics Roadmap consortium.
2.3.3 Preparation of sample overlaps

Histone mark broadPeak regions in BED format were overlapped with SNP positions using bedtools \(^{160}\), resulting in a substantial improvement in computing efficiency compared to the approach implemented by FORGE, which computed overlaps using tabix in a distributed approach on the EBI computing farm. I stored the overlaps for each SNP and histone mark in an indexed SQLite database named forge2.db. I generated a binary string for each SNP, thus representing the presence (1) or absence (0) of an overlap with a histone mark broadPeak file. SNP-sample overlaps are precomputed and stored to increase analysis speed and minimise computational requirements.

2.3.4 Analysis strategy

For each set of test SNPs, FORGE2 performs a separate overlap analysis against the regulatory elements of each cell sample (39 samples for Epigenomics Roadmap), and counts the number of overlaps. FORGE2 then picks background SNP sets of the same number of SNPs as the test set, matched in bins for minor allele frequency (MAF), distance to transcription start site (TSS) and GC content. The background sets are overlapped with the regulatory elements of each cell sample and overlap numbers computed. These background overlap numbers are then used to generate a background distribution of the expected overlap counts for the test SNP set. 1000 matched sets are used by default. The enrichment value for the test SNP set is then expressed as the \(-\log_{10}\) (binomial p-value). Enrichments outside the nominal 95th and 99th percentile of the binomial
distribution (after Benjamini-Yekutieli multiple testing correction) are considered significant. **Figure 2.7** contains a schematic of FORGE2 analysis.

**Figure 2.7: FORGE2 analysis schematic.** For the input list of SNPs FORGE2 retrieves all overlap counts in bitstring format from the SQLite database. For each SNP in the list the corresponding bitstrings are then unpacked and overlaps with histone mark broadPeaks for each cell type are computed. FORGE2 then generates a background distribution by identifying SNPs matching the test SNPs by minor allele frequency (MAF), distance to transcription start site (TSS) and GC content. 1000 background SNP sets are generated, and the overlap analysis is repeated for each background set. Finally FORGE2 computes the -log10
Materials and Methods

(binomial p-value) of the overlap count of the input list compared to the background distribution.

2.3.5 Annotation categories

SNP annotation is important as background SNPs to the test SNP set are selected by picking SNP sets equal in SNP number to the test SNP set, with SNPs matching for GC content (GC), minor allele frequency (MAF) and distance to the nearest transcription start site (TSS). Decile bins are used to class SNPs in a specific category. Annotation for all these categories was obtained from the FORGE tool\textsuperscript{[139]}. Selection for these annotation categories is required when choosing background sites to avoid spurious enrichments due to a genomic bias for certain regions known to be enriched for certain histone marks (e.g. promoters). By matching the background to the input set for 3 levels of annotation FORGE\textsuperscript{2} limits enrichments caused by a bias in the input list for certain classes of genomic elements.

2.3.6 LD filtering

Proximal SNPs in strong linkage disequilibrium (LD) will tend to segregate together, and through this phenomenon of unequal segregation they may be associated with the same trait, echoing the association signal of a single causal SNP that underlies disease mechanism. Inclusion of a group of proximal SNPs without filtering for LD will over-estimate the enrichment signal as the same
regulatory element will be counted as a separate overlap several times. To avoid such an event FORGE2 includes an LD filter, which by default selects only one SNP for each LD cluster ($r^2 >= 0.8$). FORGE2 reports SNPs that are removed to the user. Filter options include no filter, $r^2 >= 0.1$ for a highly stringent filter, and $r^2 >= 0.8$ (default and used in all analyses unless reported otherwise). For filtering the first SNP that appears in the filter set is retained and other SNPs from the same haplotype are filtered from analysis.

### 2.3.7 FORGE2 input

The user can supply a comma-separated list of SNP RSIDs or can run the RSIDs supplied from the default dataset of PR interval-associated SNPs.

Alternatively you can use a file. File data should be input in one of the following formats:

- a list of SNPs by RSID, one per line
- a list of genomic positions in BED format, corresponding to SNP positions

SNPs have to be present either on the Illumina_HumanOmni2.5 array (Omni array SNPs) or on the general set of arrays used in GWAS (GWAS typing arrays). GWAS typing arrays include the following arrays: Illumina_Human1M-duoV3, Illumina_Human660W-quad, HumanHap300v2, HumanHap550v3.0, Illumina_Cardio_Metabo, Affy_GeneChip_100K_Array, Affy_GeneChip_500K_Array, Affy_SNP6, HumanCNV370-Quadv3.
Materials and Methods

In addition to being present on one of the aforementioned arrays, SNPs also have to be present in the 1000 genomes phase I integrated call dataset (ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/phase1/analysis_results/integrated_call_sets/) to work in the analysis. FORGE2 excludes any SNPs that are not in this set from analysis. A minimum number of 5 test SNPs and a maximum number of 1000 test SNPs may be analysed in a given run.

2.3.8 FORGE2 outputs

Outputs include:

1. A PDF chart of FORGE2 -log10 (binomial p-values) generated using base R graphics. This chart is designed for use in slides and publications.

2. An interactive table showing the overlaps of individual SNPs against regulatory elements from each of the cell types.

3. An interactive chart to visually explore the data.

4. R code for regenerating these files (PDF, and interactive chart and table).

5. A raw text format file containing the original input data.

6. A tab-separated format (tsv) file with FORGE2 analysis results. Columns include z-score, p-value, cell, tissue, file (filename of the histone mark file analysed), SNP (a list of SNPs that overlap broadPeaks in the histone mark file analysed), accession (URL for the Epigenomics Roadmap data) and q value.
The -log10 (binomial p-values) are presented in each of the graphics by cell sample. Samples are listed alphabetically both by tissue and by cell type name. Point locations are plotted by -log10 (binomial p-value), and points are coloured according to BY-corrected q-values. The colouring is consistent for each of the graphics, blue (q > 0.05), pink (q < 0.05), and red (q < 0.01).

Results are presented by sample in either graphical (i.e. static PDF or interactive Dimple chart) or tabular forms (such as an interactive DataTables table or tab-separated file). A typical FORGE2 result for a given GWAS SNP set is the detection of enrichment (coloured in red or pink) in a cell type known or projected to be involved in the mechanism underlying the GWAS phenotype, for example blood cell types for rheumatoid arthritis SNPs (H3K4me1 analysis).

Alternatively no enrichment may be present, and all points will be blue. Causes for this could be technical reasons (for example, low overlap numbers), the absence of the relevant tissue, or the lack of a regulatory component underlying the GWAS association.

A list of GWAS catalogue SNPs associated with PR interval is available as a default dataset in FORGE2.
2.3.9 Multiple testing correction and false positive rates

To measure false positive rates, 1000 sets of SNPs at each of a series of SNP counts of 5, 10, 15, 20, 30, 40, 50, 100, 200, and 300 SNPs were randomly chosen from the 1000 genome phase 1 integrated SNP set. FORGE2 analysis was run for each set across the H3K4me1 Roadmap Epigenomics data, and the number of tests with q values below a threshold of 0.05 were recorded. These represent the false positives from 10 groups of 1000 trials at each of 39 samples i.e. 390,000 tests, and were used to calculate false positive rates at each significance threshold. FORGE2 incorporates an FDR-based multiple testing correction method (Benjamini-Yekutieli\textsuperscript{162}). The choice of Benjamini-Yekutieli was established in the eFORGE tool\textsuperscript{146}, which demonstrated this correction approach to be the most appropriate method when dealing with epigenetic data from multiple samples that exhibit dependence due to cellular lineage relationships. The other main FDR-based correction method, Benjamini-Hochberg\textsuperscript{166}, assumes that different samples are independent.

FORGE false positive levels are reported in the FORGE paper as 0.5-0.75% after 424,000 overlap tests (Bonferroni correction). FORGE2 presents a false positive rate (FPR) of 0% after 380,000 overlap tests. However, it is important to point out that methods such as BY that control the False Discovery Rate (FDR) are not directly comparable to methods such as Bonferroni that control the Familywise error rate. As these methods are doing two different things, I can only conclude that for FORGE2 (disregarding comparisons to FORGE) I have not been able to detect false positives after 380,000 overlap tests.
2.3.10 GWAS analysis

I applied FORGE2 to disease-associated SNPs from the GWAS catalogue\textsuperscript{123}. The entire GWAS catalogue was downloaded as of 12-6-2015 from https://www.ebi.ac.uk/gwas/ and a parallel approach to analyse every disease-associated SNP list with FORGE2 was implemented.

2.3.11 Source code

FORGE2 source code can be downloaded from GitHub at https://github.com/charlesbreeze/FORGE2. FORGE2 installation and analysis have been performed on Mac OSX 10.10.1.
3 eFORGE Results
3.1 tDMP analysis

I first applied eFORGE to analyse probe sets of known tissue and cell type-specificity, focusing on probe sets obtained from studies on tissue-specific DNAm. These probe sets would serve as a positive control for confirmation of eFORGE function. I thus analysed 3 sets of tissue-specific DMPs (tDMPs) and 3 sets of cell type-specific DMPs (cDMPs). Figure 3.1 shows analysis results, confirming the ability of eFORGE to detect tissue and cell type-specific patterns in DNAm data. Tissues analysed include whole blood, kidney and lung. For cell types I analysed data for CD14+ cells, NK cells and T cells.

Figure 3.1: eFORGE tDMP analysis. eFORGE analysis results for tDMPs from whole blood, kidney and lung and cDMPs from CD14+ cells, NK cells and T cells.
cells\textsuperscript{165}. For each of these sets of tissue and cell type-associated positions eFORGE can detect the correct category using DNase I hotspot data from the Epigenomics Roadmap project. For each row (tissue) and column (450k probe set) eFORGE generates a random background. eFORGE then computes the enrichment in open chromatin sites of the 450k probe set against the random background. Original figure from Breeze et al., 2016\textsuperscript{146}.

3.2 EWAS findings

For eFORGE analysis I initially focused on autoimmune diseases. Given that autoimmune diseases are known to affect white blood cells, the important question in this area is the identification of the precise leucocyte cell types (e.g. CD4+, CD8+) that are most enriched for regions associated with a specific condition. Knowledge of the precise cell types involved would aid therapeutic target development, potentially helping to uncover key processes in immune system function. I analysed top probes from 3 EWAS on autoimmune diseases: rheumatoid arthritis (RA), systemic lupus erythematosus (SLE) and Sjögren syndrome. Each of these studies showed a different blood-specific enrichment in eFORGE analysis (figure 3.2). Specifically, the RA EWAS highlighted CD14+ cells (q value=5.53e-04). CD14+ cells have been shown to present accelerated maturation in RA\textsuperscript{168}. This finding is in contrast to the SLE EWAS, which pointed to T cells (q value=2.56e-05). T cells (especially CD4+ T cells) are crucial to the development of SLE\textsuperscript{169}. eFORGE also pointed to T cells for the Sjögren's
Results

Syndrome EWAS (q value = 1.31e-49). For Sjögren’s syndrome T cells have been put forward as specific therapeutic targets\textsuperscript{170}.

Figure 3.2: eFORGE analysis of autoimmune disease EWAS. This figure shows a tissue-specific signal for top probes from 3 autoimmune disease EWAS. In the top panel it can be observed that for a given SLE EWAS set of 86 probes I obtain a blood, intestine and thymus-specific signal beyond the cutoff threshold (red points). In the middle panel I observe a more general pattern of enrichment, with a strong blood signal for a set of 100 RA EWAS probes. In the lower panel I observe a blood and thymus-specific enrichment for a set of 753 probes for an EWAS on Sjögren syndrome. I obtained all top probes from the supplementary files of the studies \textsuperscript{171,172,35}. Original figure from Breeze et al., 2016\textsuperscript{146}.
I then applied eFORGE analysis to autoimmune EWAS performed on non-blood tissue, focusing on an EWAS on Multiple Sclerosis (MS)\textsuperscript{174}. This EWAS measured DNAm differences in non-pathological brain tissue between MS cases and controls, using the aforementioned 450k array. The study attributed the DNAm differences mainly to neuronal processes and not to inflammation (hence the specification of “non-pathological brain tissue”, measured by the aspect of the tissue under the microscope). However, the Gene Ontology analysis performed by the study shows high enrichment for some immune processes. I decided to investigate the matter further, especially given that there is a resident immune cell type, termed microglia, that constitutes up to 15% of all cells in the central nervous system of mammals\textsuperscript{175}.

Preliminary analyses with eFORGE showed a strong immune signal for the top 1000 hypomethylated probes across DNase I hotspot data from several different consortia. In addition, eFORGE analysis for the same probe set across all histone marks narrowed the signal down to an immune enhancer-specific signal (H3K4me1) underlying the DNase I hotspot enrichment. I then studied the intersection of the top 1235 hypomethylated probes with active enhancers (n=1158) identified previously in microglial cells (Lavin et al. 2014). Significant co-localisation of these probes with the microglial-specific active enhancers was confirmed by a Fisher’s exact test (p-value= 2.70e-07, OR= 5.88, 95% CI= 3.19-9.96). This suggests that microglial activation may be a driving signal behind the observed DNAm differences.

One other aspect of EWAS research is the use of surrogate tissues. In this context,
researchers measure changes in DNAm from easily accessible tissues (e.g. whole blood or buccal cells) instead of the target tissue that is most relevant to disease mechanism. In the literature it has been suggested that changes in DNAm in surrogate tissues can “mimic” or reflect epigenomic perturbations happening in the target tissue\textsuperscript{176}. An alternate perspective would state that the observed DNAm changes are not mimicking DNAm changes in the target tissue but instead are specific to the surrogate tissue. To resolve a particular case of surrogate tissue use in EWAS, eFORGE analysis was performed on the top 110 probes from an EWAS on ovarian cancer\textsuperscript{29}. This EWAS was performed on whole blood using a pre-treatment discovery cohort, with the aim of identifying a DNAm signature separating ovarian cancer cases from healthy controls. Enrichment was found for CD14+ cells (q-value=1.37e-12, see figure 3.3). However, enrichment was not detected for ovary (q-value=1) or solid cancer tissues (q-value=1). This analysis pointed to a surrogate-tissue based EWAS in which the DNAm differences were independent of the target tissue, despite predicting ovarian cancer status. For a mechanistic understanding of this biomarker, I postulate that this immune process reflects an immune reaction to the ovarian cancer, rather than mimicry of DNAm changes occurring during ovarian cancer transformation.

\textbf{Figure 3.3: eFORGE analysis of an EWAS performed on a surrogate tissue.}

For analysis I took 110 top probes from an ovarian cancer prediction EWAS
Results measured in whole blood. Results indicate a CD14+ cell-specific signal for Epigenomics Roadmap DNase I hotspot data.

As I have previously mentioned, among the potential confounders in EWAS, cell composition effects form one of the main sources of variance. The detection and correction of this large confounder has been one of the major breakthroughs that have allowed the field to perform large-scale EWAS for a variety of diseases. This is epitomised when using the main surrogate tissue for EWAS, namely whole blood. DNAm differences measured in this heterogeneous tissue, when uncorrected, many times result from a difference in cell numbers. For example, in a given EWAS there may be a higher proportion of CD4+ cells in cases than in controls, and therefore CD4+ cDMPs drive the DNAm differences observed in the study, despite these not being DNAm differences associated directly with disease mechanism, but rather with cell type ratios. Thus a difference in cell numbers is defined as a confounder in EWAS.

Because cell composition effects are driven by cell type-specific Differentially Methylated Positions, which in turn are co-localised with cell type-specific DNase I hotspots and detected by eFORGE, eFORGE is also a tool for detecting cell composition effects. One particular example of this is the detection of a skew in the granulocyte/lymphocyte ratio in the aforementioned EWAS on ovarian cancer. The DNAm signature detected (which is predictive of ovarian cancer status) was driven by a difference in cell counts between cases and controls. Specifically, the difference in CD14+ cells was the main driver of the DNAm
Results
differences found in the study. CD14+ cell-specific Differentially Methylated Positions were the main source of variance in the study, and make up most of the study top hits. If analysed by eFORGE, this whole blood-based study strikingly points to CD14+ cells only. This is unexpected as typically different immune cell types will show epigenetic differences as part of an immune response or an environmental perturbation. This is further highlighted if we take into account that in a typical humoral response T cell activation and B cell activation are both required for antibody production\textsuperscript{177}. In addition, cytokine signalling constitutes a standard component of immune response, in which a crosstalk between different immune cell types occurs\textsuperscript{177}. Therefore it is unexpected to only find one cell type linked to the vast majority of DNAm changes in a study, and a much simpler explanation can be found by considering this phenomenon a result of cell composition effects.

One of the clearest areas for eFORGE application is in the study of cancer. Using eFORGE I discovered a clear stem cell signal across 5 cancer EWAS (\textbf{figure 3.4}), indicative of a preferential co-localisation of DNAm differences with stem cell-specific DNase I hotspots. No significant overlaps were observed for any other cell type. The following cancer EWAS were analysed: breast cancer (Fang et al., 2011)\textsuperscript{178}, colorectal cancer (Kibriya et al., 2011)\textsuperscript{179}, sporadic colorectal cancer (Laczmanska et al., 2013)\textsuperscript{180}, clear cell renal cell carcinoma (Arai et al., 2012)\textsuperscript{181}, and adrenocortical carcinoma (Barreau et al., 2013)\textsuperscript{182}. eFORGE analysis was performed for the top 330, 450, 240, 801 and 362 EWAS hits, respectively.
Figure 3.4: eFORGE analysis of 5 cancer EWAS. A stem cell-specific eFORGE DNase I hotspot signal can be observed. Cancer-associated probe sets analysed (columns) include 330 top probes from an EWAS on breast cancer metastatic behaviour\textsuperscript{178} (left), 450 probes from an EWAS on colorectal carcinoma (second from left)\textsuperscript{179} and 240 probes from an EWAS on sporadic colorectal cancer (centre)\textsuperscript{180}. In addition, I analysed 362 probes from an EWAS on adrenocortical carcinoma\textsuperscript{182} (second from right), and 801 probes from an EWAS on clear cell renal cell carcinoma\textsuperscript{181} (right). Stem cell row names are highlighted in red (corresponding to Epigenomics Roadmap DNase I hotspot stem cell samples). Original figure from Breeze et al, 2016\textsuperscript{146}.
However, until now I have only focused on specific diseases and traits. It is also important to perform an unbiased analysis applying eFORGE across a variety of previously unselected traits, in order to ascertain if tissue-specific eFORGE enrichment is a general phenomenon or rather a partial phenomenon that only occurs for a few pre-selected traits. In order to perform an unbiased selection of EWAS, I selected studies with \( n > 100 \) from Supplementary Table 1 of the latest relevant EWAS review\(^8\). Analyses of the 20 studies that presented eFORGE signal confirmed that tissue-specific enrichment is a general phenomenon (figures 3.5 and 3.6). 14 studies showed a pattern of tissue-specific enrichment, including blood-specific enrichment for 6 blood-based EWAS and stem cell-specific enrichments for five ageing and cancer EWAS (figure 3.5).
Figure 3.5: Aggregated eFORGE results for analysis across 20 EWAS. This analysis confirms the general nature of tissue-specific eFORGE enrichment. In the red box a clear stem cell signature for cancer and ageing EWAS can be observed. In the blue box an immune blood cell signature for EWAS performed on whole blood can be seen. The orange and pink boxes highlight lung- and kidney-specific enrichments observed for particular studies. The yellow box highlights the remainder of studies (6) which did not show enrichment for a specific tissue but rather a mixed enrichment across several tissues. Further research is needed to deconvolute these mixed signals. It is to be noted that this graph confirms that the majority of EWAS with eFORGE enrichment (70%) showed an enrichment that was tissue-specific, thus confirming the general
nature of EWAS tissue-specific enrichment for an unbiased selection of studies. Original figure from Breeze et al., 2016\textsuperscript{146}.

Unlike the field of GWAS, which has a well developed and extended catalogue (\url{https://www.ebi.ac.uk/gwas/}), the field of EWAS does not have a centralised repository with a list of studies, analyses, and related data for use by the community\textsuperscript{123}. I assembled the selected studies, along with eFORGE analyses, to form part of the first EWAS catalogue, the eFORGE catalogue (\textbf{figure 3.6 and table 3.1}). Inclusion of eFORGE analyses provides key data on the predicted cell type- or tissue-specific action of top EWAS hits for each study, thus aiding study interpretation, which has previously been posited as challenging\textsuperscript{18}. This preliminary list is the starting point for a general EWAS catalogue, projected in years to come to be an invaluable resource for the epigenetics community.
Figure 3.6: Genomic distribution and tissue origin from studies in the eFORGE catalogue. [A]: The karyotype plot contains top 10 study hits from each of the eFORGE catalogue EWAS. Bioinformatically the image was generated with ensembl KaryoView ([http://www.ensembl.org/Homo sapiens/Location/Genome](http://www.ensembl.org/Homo sapiens/Location/Genome)). No signal can be observed in X and Y chromosomes (probes on these locations are filtered in EWAS pipelines). Other than this, the location of top EWAS probes seems to show no particular genomic bias. [B]: This bar chart shows the tissues analysed by the EWAS in the catalogue. Blood, the most easily accessible tissue, is also unsurprisingly the most analysed category (10 studies). Original figure from Breeze et al., 2016.146
<table>
<thead>
<tr>
<th>Study number</th>
<th>Number of samples</th>
<th>PMID</th>
<th>Disease/Trait</th>
<th>Platform</th>
<th>Analysed tissue</th>
<th>eFORGE tissue</th>
<th>eFORGE link ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2442</td>
<td>23034122</td>
<td>Ageing in blood and brain <strong>;</strong>*</td>
<td>27 and 450k</td>
<td>Whole blood, leucocytes, brain</td>
<td>Stem cell signal</td>
<td><a href="http://eforge.cs.ucl.ac.uk/eFORGEv1.2/files/0xC8FF623812ED11E6B8D459BF8274B10/">http://eforge.cs.ucl.ac.uk/eFORGEv1.2/files/0xC8FF623812ED11E6B8D459BF8274B10/</a></td>
</tr>
<tr>
<td>2</td>
<td>135</td>
<td>23093492</td>
<td>Adrenocortical carcinoma**</td>
<td>27k</td>
<td>Adrenocortical carcinoma tumours</td>
<td>Stem cell signal</td>
<td>0x4D980C8425B311E6B2D33DCF8274B10/</td>
</tr>
<tr>
<td>3</td>
<td>910</td>
<td>23578854</td>
<td>Breast cancer</td>
<td>27k</td>
<td>Blood</td>
<td>Mixed enrichment</td>
<td>0x439DAF16242011E6B2D23DDCF8274B10/</td>
</tr>
<tr>
<td>4</td>
<td>138</td>
<td>22610075</td>
<td>Clear cell renal cell carcinomas (RCCs)**</td>
<td>27k</td>
<td>29 renal cortex, 109 tumors, 107 normal</td>
<td>Stem cell and other mixed enrichment</td>
<td>0x5550140A115211E6AF74B300F9274B10/</td>
</tr>
<tr>
<td>5</td>
<td>367</td>
<td>21297937</td>
<td>Congenital Heart Defects (CHDs)</td>
<td>27k</td>
<td>Blood</td>
<td>Mixed enrichment</td>
<td>0xF3288FF2240A11E6A6E27FB6F8274B10/</td>
</tr>
<tr>
<td>6</td>
<td>192</td>
<td>20687937</td>
<td>Diabetic nephropathy**</td>
<td>27k</td>
<td>Whole blood</td>
<td>Mixed enrichment</td>
<td>0x149CA7F4120E11E6F480F5F8274B10/</td>
</tr>
<tr>
<td>7</td>
<td>1230</td>
<td>22714737</td>
<td>Non-blood cancers, leucocyte profiles</td>
<td>27k</td>
<td>Purified peripheral blood cell types</td>
<td>CD14+ cell signal</td>
<td>0x62FCB8C240D11E6A13229BAFF8274B10/</td>
</tr>
<tr>
<td>8</td>
<td>160</td>
<td>21453505</td>
<td>Parental age**</td>
<td>27k</td>
<td>Cord blood</td>
<td>Blood signal</td>
<td>0x451A5388160D11E6945594AF8274B10/</td>
</tr>
<tr>
<td>9</td>
<td>201</td>
<td>21308970</td>
<td>Racial differences at birth**</td>
<td>27k</td>
<td>Cord blood</td>
<td>Blood signal</td>
<td>0xBC47687C161411E6AF88B9AFF8274B10/</td>
</tr>
<tr>
<td>10</td>
<td>255</td>
<td>23579546</td>
<td>Breast cancer</td>
<td>27k, 450k</td>
<td>Breast</td>
<td>Blood and thymus signal</td>
<td>0xC4946164242011E6E468D4F8274B10/</td>
</tr>
<tr>
<td>11</td>
<td>317</td>
<td>23526956</td>
<td>Clear cell renal cell carcinoma (ccRCC)</td>
<td>27k, 450k</td>
<td>Kidney (tumour and non-tissue)</td>
<td>Kidney signal</td>
<td>0xE35BF962441F11E693220FD2F8274B10/</td>
</tr>
<tr>
<td>12</td>
<td>656</td>
<td>23177740</td>
<td>Ageing***</td>
<td>450k</td>
<td>Whole blood</td>
<td>Stem cell signal</td>
<td>0x2B74E1A8137B11E6AF31C3D2F8274B10/</td>
</tr>
<tr>
<td>13</td>
<td>192</td>
<td>23332324</td>
<td>Child maltreatment***</td>
<td>450k</td>
<td>Salivary DNA</td>
<td>Mixed enrichment with blood skew</td>
<td>0x6B81FA2E12E11E6A150A9F7F8274B10/</td>
</tr>
<tr>
<td>14</td>
<td>691</td>
<td>23334450</td>
<td>Rheumatoid arthritis</td>
<td>450k</td>
<td>Blood</td>
<td>Blood enrichment, with other tissues</td>
<td>0xA38CF9B8241E11E6AFF55DF8274B10/</td>
</tr>
<tr>
<td>15</td>
<td>184</td>
<td>23175441</td>
<td>Smoking, breast/colon cancer</td>
<td>450k</td>
<td>Blood</td>
<td>Stomach enrichment</td>
<td>0x0E0FA7E240E11E6A42C34BBF8274B10/</td>
</tr>
<tr>
<td>16</td>
<td>1793</td>
<td>23691101</td>
<td>Tobacco smoking***</td>
<td>450k</td>
<td>Whole blood</td>
<td>Blood and thymus</td>
<td>0x3C0CE5161211E6B61CBFA8274B10/</td>
</tr>
<tr>
<td>17</td>
<td>138</td>
<td>22232023</td>
<td>Tobacco smoking***</td>
<td>450k</td>
<td>119 lymphoblasts and 19 lung macrophages</td>
<td>Breast epithelial signature</td>
<td>0xB1C7A0412D911E6B76CFF8274B10/</td>
</tr>
<tr>
<td>18</td>
<td>153</td>
<td>23064414</td>
<td>Two subtypes of CLL***</td>
<td>450k</td>
<td>Tumor samples and B cells from the controls</td>
<td>Mixed enrichment</td>
<td>0xDA9362C0E2311E6A4BC0DDDF8274B10/</td>
</tr>
<tr>
<td>19</td>
<td>261</td>
<td>20019873</td>
<td>Ovarian cancer**</td>
<td>27k</td>
<td>Peripheral blood</td>
<td>CD14+ blood signal</td>
<td>0x33752B89CD311E4BEA3AC3AA596114 (v1.1)</td>
</tr>
<tr>
<td>20</td>
<td>114</td>
<td>23666970</td>
<td>Thyroid cancer subtypes**</td>
<td>27k</td>
<td>114 snap-frozen thyroid samples</td>
<td>Cross-tissue mixed enrichment</td>
<td>0xA8797EC0E011E69480E2F8274B10/</td>
</tr>
</tbody>
</table>
Table 3.1: list of EWAS in the eFORGE catalogue. Columns 1-6 contain study information, column 7 lists enriched cell types and tissues, and column 8 contains the eFORGE link ID (an example of a full URL is given in line 1, all IDs correspond to eFORGE v1.2 unless indicated).
4 FORGE2 results
4.1 FORGE2 analysis of the GWAS catalogue

Functional interpretation of disease-associated GWAS variants is a challenging task given that most variants are located in non-protein-coding regions\textsuperscript{136}. The lack of data for non-protein-coding regions is one of the main obstacles impeding the correct interpretation of these variants. This problem has been addressed by consortia such as Epigenomics Roadmap, ENCODE and BLUEPRINT which have generated abundant genome-wide epigenomic data covering both protein-coding and non-protein-coding regions \textsuperscript{81,183,184}. Epigenomics Roadmap and ENCODE data have been applied by the FORGE tool to analyse the enrichment of GWAS SNP sets in DNase I hotspots across a variety of tissues, yielding tissue-specific associations that take us one step closer to variant interpretation\textsuperscript{139}. I have developed FORGE2 to analyse the cell type-specific enrichment of GWAS SNP sets across 5 histone marks (H3K4me1, H3K4me3, H3K27me3, H3K9me3 and H3K36me3).

FORGE2 analysis was applied to the complete GWAS catalogue as of 12-6-2015 (see Methods). I observed many cell type- and tissue-specific patterns, a number of which were new, and many of which were also detected by FORGE for the same traits (table \textbf{4.1}, \textbf{figure 4.1}). For example, in my analysis of SNPs associated with PR interval FORGE and FORGE2 both detect heart tissue as enriched (using DNase I hotspot and H3K4me1 data respectively, \textbf{figure 4.2}). This suggests that in certain cases regulatory elements represented by FORGE DNase I hotspots and FORGE2 histone marks overlap. In addition, in the case of mean platelet volume I observe two different associations in FORGE2: an enhancer
signature (H3K4me1) and a promoter signature (H3K4me3). Both associations point to CD34+ haematopoietic progenitor cells, a precursor of platelet-producing megakaryocytes and thus a plausible candidate for trait mechanism. This same cell type is enriched in FORGE DNase I hotspot analysis. The FORGE2 dual histone mark enrichment provides evidence for the utility of separating distinct classes of regulatory elements in functional overlap analysis. Importantly, I also report a series of novel findings not detected by FORGE. Specifically, four types of novel findings were observed.

**Figure 4.1: Types of FORGE2 findings when compared to FORGE.** Most FORGE2 findings detect the same tissue-specific signal as FORGE for the same GWAS SNP sets (n=38). However, FORGE2 also detected new signals for SNP sets for which FORGE showed no signal (n=9). In addition, FORGE2 showed a lack of agreement with FORGE for a few studies, with each software tool pointing to different tissue-specific signals (n=3). FORGE2 also demonstrated the ability to extend signal detection to additional tissues not detected by FORGE (n=2) and
also showed evidence of signal improvement (defined as FORGE2 signal <0.01 for GWAS SNP sets with a FORGE p-value < 0.05, n=1).

In one case I found that a low enrichment signal (significance level <0.05) already present in FORGE was found to present a higher enrichment in FORGE2 (significance level <0.01), this was the case for an H3K4me1 NK-specific signal in my analysis of “Systemic Lupus Erythematosus (SLE) and Systemic Sclerosis (SSc)” GWAS data. The overlapping SNP sets in FORGE2 analyses were found to include the majority (66%) of SNPs overlapping in FORGE analysis, which suggests that some NK cell-specific H3K4me1 sites in FORGE2 analysis are reflecting NK cell-specific DNase I hotspots in FORGE. These regions may potentially be NK cell-specific enhancers, as H3K4me1 is an enhancer-enriched mark. Mechanistically, SLE involves an immune response against multiple tissues while SSc involves the targeting of connective tissue. NK cells have an established role in the pathology of both SLE and SSc185.

In other cases a completely new signal is detected, with no signal present in FORGE for the same SNPs (for example in my analysis of late-onset Alzheimer’s disease GWAS data, table 4.2). The specific case of late-onset Alzheimer’s disease data pointed to an immune response, as several immune cell types, including haematopoietic stem cells and monocytes, showed enrichment (figure 4.3). This is in agreement with previous findings on Alzheimer’s disease81. FORGE2 analysis also identified an involvement of T cells in two immune-related conditions (vein graft stenosis in coronary artery bypass grafting and Graves’
Results

disease). In the former T cells constitute a candidate immune cell type associated with inflammation, a key factor in the pathophysiological process of stenosis after coronary artery bypass grafting. In the latter (Graves’ disease), a defect in immune system function causes the production of abnormal TSH-mimicking antibodies, that overstimulate the thyroid leading to a pathological condition. FORGE2 analysis points to T cells as one of the main cell types involved in the regulatory process leading to antibody production in Graves’ disease.

In stark contrast to previous approaches that have applied histone mark data to analyse GWAS SNPs, I report FORGE2 enrichment for a previously GWAS-uninformative mark, namely H3K27me3 (a repressive mark linked to the polycomb repressive complex 2 - PRC2). Among other cases (tables 4.1, 4.2 and 4.3), H3K27me3 enrichment was observed for B cells when analysing a GWAS on C-reactive protein (CRP) levels. Considering the aforementioned FORGE2 FPR analysis (see Methods), it is statistically unlikely that this enrichment is a false positive. FORGE2 results suggest an immune mediation of CRP level-associated GWAS SNP action, potentially through disruption of PRC2-repressed regions (which would represent a novel mechanism of action for GWAS-linked variants).

In certain cases a completely different tissue is indicated by FORGE2 when compared to FORGE results (for example in my analysis of food allergy GWAS data, table 4.3). In the case of food allergy FORGE shows enrichment for spinal cord, whereas FORGE2 shows H3K4me3 stomach-specific enrichment. I propose that stomach represents a more plausible mechanistic tissue for this condition, which may be potentially mediated through mucosa-associated lymphoid tissue.
Results

(MALT). For F-cell distribution, that is, the abnormal presence of foetal haemoglobin in adults, previous FORGE data pointed to an intestinal signal. However, the main candidate cell type proposed by FORGE2 is CD34+ haematopoietic progenitor cells (H3K4me1 signal). This finding links GWAS SNPs to an erythrocyte precursor, providing a series of haematopoietic enhancers for further study (locations corresponding to variants rs10172646, rs10189857, rs1427407, rs6545816, rs6706648, rs6738440, rs7565301, rs7606173, rs9399137 and rs9399137).
<table>
<thead>
<tr>
<th>Disease</th>
<th>FORGE2 signal</th>
<th>FORGE signal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rheumatoid arthritis</td>
<td>H3K4me1 blood enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>UC</td>
<td>H3K4me1 blood enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>Crohn’s Disease</td>
<td>H3K4me1 blood enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>IBD</td>
<td>H3K4me1 blood enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>T1D</td>
<td>H3K4me1 blood enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>Self-reported allergy</td>
<td>H3K4me1 blood enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>Red blood cell traits</td>
<td>H3K4me1 CD34+ cell enrichment</td>
<td>DNase I hotspot CD34+ cell enrichment</td>
</tr>
<tr>
<td>QT interval</td>
<td>H3K4me1 heart enrichment</td>
<td>DNase I hotspot heart enrichment</td>
</tr>
<tr>
<td>MS</td>
<td>H3K4me1 blood enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>PR interval</td>
<td>H3K4me1 heart enrichment</td>
<td>DNase I hotspot heart enrichment</td>
</tr>
<tr>
<td>Restless legs syndrome</td>
<td>H3K4me1 heart enrichment</td>
<td>DNase I hotspot heart enrichment</td>
</tr>
<tr>
<td>Breast cancer</td>
<td>H3K4me1 general enrichment</td>
<td>DNase I hotspot general enrichment</td>
</tr>
<tr>
<td>Waist−hip ratio adjusted for BMI</td>
<td>H3K4me1 general enrichment</td>
<td>DNase I hotspot general enrichment</td>
</tr>
<tr>
<td>Height</td>
<td>H3K4me1 general enrichment</td>
<td>DNase I hotspot general enrichment</td>
</tr>
<tr>
<td>Electrocardiographic traits</td>
<td>H3K4me1 heart enrichment</td>
<td>DNase I hotspot heart enrichment</td>
</tr>
<tr>
<td>Platelet count</td>
<td>H3K4me1 CD34+ cell enrichment</td>
<td>DNase I hotspot CD34+ cell enrichment</td>
</tr>
<tr>
<td>Mean corpuscular haemoglobin</td>
<td>H3K4me1 CD34+ cell enrichment</td>
<td>DNase I hotspot CD34+ cell enrichment</td>
</tr>
<tr>
<td>Mean platelet volume</td>
<td>H3K4me1 CD34+ cell enrichment</td>
<td>DNase I hotspot CD34+ cell enrichment</td>
</tr>
<tr>
<td>Chronic lymphocytic leukaemia</td>
<td>H3K4me1 blood lymphoid enrichment</td>
<td>DNase I hotspot blood and thymus enrichment</td>
</tr>
<tr>
<td>Coeliac disease</td>
<td>H3K4me1 blood and thymus enrichment</td>
<td>DNase I hotspot blood and thymus enrichment</td>
</tr>
<tr>
<td>Mean platelet volume</td>
<td>H3K4me3 CD34+ cell enrichment</td>
<td>DNase I hotspot CD34+ cell enrichment</td>
</tr>
<tr>
<td>IBD</td>
<td>H3K4me3 CD34+ cell and blood</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>Platelet count</td>
<td>H3K4me3 CD34+ cell enrichment</td>
<td>DNase I hotspot CD34+ cell enrichment</td>
</tr>
<tr>
<td>QT interval</td>
<td>H3K4me3 heart enrichment</td>
<td>DNase I hotspot heart enrichment</td>
</tr>
<tr>
<td>Chronic lymphocytic leukaemia</td>
<td>H3K4me3 blood enrichment</td>
<td>DNase I hotspot blood and thymus enrichment</td>
</tr>
<tr>
<td>Crohn’s disease</td>
<td>H3K4me3 CD34+ cell enrichment</td>
<td>DNase I hotspot blood and thymus enrichment</td>
</tr>
<tr>
<td>MS</td>
<td>H3K4me3 blood enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>Height</td>
<td>H3K4me3 skin enrichment</td>
<td>DNase I hotspot general enrichment</td>
</tr>
<tr>
<td>Coeliac disease</td>
<td>H3K4me3 T cell enrichment</td>
<td>DNase I hotspot blood and thymus enrichment</td>
</tr>
<tr>
<td>Breast cancer</td>
<td>H3K4me3 immune and skin enrichment</td>
<td>DNase I hotspot general enrichment</td>
</tr>
<tr>
<td>UC</td>
<td>H3K27me3 several organ enrichment</td>
<td>DNase I hotspot several organ enrichment</td>
</tr>
<tr>
<td>Height</td>
<td>H3K36 very high enrichment in all tissues except stem cells</td>
<td>DNase I hotspot general enrichment</td>
</tr>
<tr>
<td>MS</td>
<td>H3K36 blood enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>Rheumatoid arthritis</td>
<td>H3K36 thymus enrichment</td>
<td>DNase I hotspot blood and thymus enrichment</td>
</tr>
<tr>
<td>IBD</td>
<td>H3K36 T cell and B cell enrichment</td>
<td>DNase I hotspot blood enrichment</td>
</tr>
<tr>
<td>Mean platelet volume</td>
<td>H3K36 thymus enrichment</td>
<td>DNase I hotspot blood and thymus enrichment</td>
</tr>
<tr>
<td>Primary biliary cirrhosis</td>
<td>H3K36 NK enrichment</td>
<td>DNase I hotspot NK enrichment</td>
</tr>
</tbody>
</table>
Table 4.1: Common findings between FORGE2 and FORGE. Findings in which FORGE2 identified a tissue listed among FORGE tissues are also included. The most common tissue category for the traits analysed is blood, with 20 out of 36 categories (55.55%) showing a blood-related enrichment. BMI: Body mass index. IBD: Inflammatory bowel disease. MS: Multiple sclerosis. UC: Ulcerative colitis. T1D: Type 1 diabetes.

<table>
<thead>
<tr>
<th>Disease</th>
<th>FORGE signal</th>
<th>FORGE2 signal</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Vein graft stenosis in coronary artery bypass grafting</strong></td>
<td>No signal</td>
<td>T Cell signal (H3K4me1)</td>
</tr>
<tr>
<td><strong>Graves’ disease</strong></td>
<td>No signal</td>
<td>T Cell signal (H3K4me1)</td>
</tr>
<tr>
<td><strong>Attention deficit hyperactivity disorder combined symptoms</strong></td>
<td>No signal</td>
<td>Stem cell (H3K4me1)</td>
</tr>
<tr>
<td><strong>Late-onset Alzheimer’s disease</strong></td>
<td>No signal</td>
<td>Strong blood signal (H3K4me1)</td>
</tr>
<tr>
<td><strong>C-reactive protein levels</strong></td>
<td>No signal</td>
<td>B Cell (H3K27me3)</td>
</tr>
<tr>
<td><strong>Political ideology</strong></td>
<td>No signal</td>
<td>Brain (H3K27me3)</td>
</tr>
<tr>
<td><strong>Blood metabolite levels</strong></td>
<td>No signal</td>
<td>Many cell types (H3K36me3)</td>
</tr>
<tr>
<td><strong>Total ventricular volume</strong></td>
<td>No signal</td>
<td>H3K27me3 low gastric enrichment</td>
</tr>
<tr>
<td><strong>Oesophageal cancer squamous cell carcinoma</strong></td>
<td>No signal</td>
<td>H3K36 general enrichment</td>
</tr>
</tbody>
</table>

Table 4.2: FORGE2 findings not detected in FORGE analyses.

<table>
<thead>
<tr>
<th>Disease</th>
<th>FORGE signal</th>
<th>FORGE2 signal</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>F cell distribution</strong></td>
<td>Intestinal signal</td>
<td>Brain and CD34+ cells (H3K4me1)</td>
</tr>
<tr>
<td><strong>Food allergy</strong></td>
<td>Spinal cord</td>
<td>Stomach signal (H3K4me3)</td>
</tr>
<tr>
<td><strong>Thyroid cancer</strong></td>
<td>Stem cells</td>
<td>CD34+ cells (H3K27me3)</td>
</tr>
</tbody>
</table>

Table 4.3: FORGE2 findings that show a different tissue from FORGE.
Table 4.4: Additional tissue FORGE2 findings. These FORGE2 findings show enrichment for additional tissues when compared to FORGE (data for these tissues is also present in FORGE).

<table>
<thead>
<tr>
<th>Disease</th>
<th>FORGE signal</th>
<th>FORGE2 signal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary biliary cirrhosis</td>
<td>CD56+ cells</td>
<td>Signal across blood (H3K4me1)</td>
</tr>
<tr>
<td>Psoriasis</td>
<td>Blood</td>
<td>Skin and blood (H3K36me3)</td>
</tr>
</tbody>
</table>

Figure 4.2: FORGE and FORGE2 PR interval GWAS analysis. FORGE results for PR interval SNPs show enrichment for DNase I hotspots in heart cells. FORGE2 H3K4me1 results for the same SNPs also highlight heart tissue.
Results

Figure 4.3: FORGE and FORGE2 Alzheimer’s disease GWAS analysis. FORGE results for Alzheimer’s disease SNPs show no significant enrichments, FORGE2 results show enrichment for immune cell types.

In some cases additional tissues not enriched in FORGE are detected in FORGE2 analysis (for example in my analysis of psoriasis GWAS data, table 4.4). Analysis of psoriasis GWAS data was found to extend the range of tissues associated with this autoimmune condition that targets the skin. Surprisingly, I observed not only enrichment for the proposed effector tissue (immune blood cells), but also for the target tissue (skin). This enrichment was driven by the transcription-associated mark H3K36me3. For psoriasis GWAS data no enrichment was observed for other histone marks. This H3K36me3-specific enrichment suggests that the variants linked with psoriasis detected through GWAS do not preferentially affect enhancers or promoters (unlike many of the analyses that have been presented here), but rather sites that direct transcription-related processes (e.g. transcription elongation, activation and alternative splicing). This is yet another mechanism of action for GWAS-related regions, and I propose that,
as more and more data become available, different mechanisms of action of GWAS SNPs on gene regulation will appear.

4.2 Analysis of psoriasis GWAS data

Further analysis of the psoriasis signal revealed that some of the regions underlying enrichment in blood (with B cells as the highest category) are different from the regions underlying enrichment in skin. Six out of 34 B cell-overlapping SNPs are specific to B cells (12%) and 12 out of 40 skin-overlapping SNPs are specific to skin (26.1%). Twenty-eight SNPs were found to overlap H3K36me3 broadPeaks in both skin and B cells (60.9%). Applying FORGE2 analysis I can detect separate H3K36me3 signal for these independent SNP sets (figure 4.4). This shows that some of the SNPs do not only overlap distinct elements between B cells and skin but also that the independent SNP sets tested show specificity to B cells and skin cells when compared to a range of other tissues.
Figure 4.4: FORGE2 psoriasis GWAS analysis. FORGE2 analysis with “B cell only” SNPs shows a clear skew towards B cells and blood (top panel). Analysis with the “skin only” SNPs shows skin as distinct top category, with other tissues at a much lower level (note the logarithmic scale). Analysis with all 138 psoriasis-associated GWAS SNPs shows signal for both blood and skin.

I also report data on the distinct molecular pathways underlying these findings based on the genes where this H3K36me3 signal is found (using Ensembl data through bioMaRt in R)\textsuperscript{186}. Many of the genes linked to B cells showed links to immune-specific mechanisms. For example IFNLR1 (interferon lambda receptor 1) and IFIH1 (interferon induced with helicase C domain 1) show key links to viral response, of which B cells are a key component\textsuperscript{187}. Another immune-related gene, ELMO1 (engulfment and cell motility 1), is related to phagocytosis and motility-linked immune response. Historically B cells were considered distinct from cells with a phagocytic activity (which include, among others, monocytes.
Results

and macrophages). However, recent findings have shown that certain B cell subsets have a phagocytic activity across a range of vertebrates\textsuperscript{188}, including mammals\textsuperscript{189}. Confirming the immune-associated nature of the list of genes obtained for B cell regions, analysis with amiGO\textsuperscript{45} revealed immune response pathways among the highest categories. Many of the genes linked to skin were related to dermal and connective tissue processes, with connective tissue as the top category in amiGO\textsuperscript{45} analysis. These genes include COG6 (component of oligomeric Golgi complex 6) and TNIP1 (TNFAIP3 interacting protein 1). COG6 is expressed in skin tissue, specifically sweat glands, and is associated with skin-related mendelian conditions\textsuperscript{190}. In addition, research on TNIP1 has evidenced links between this gene and psoriasis\textsuperscript{191}. Proliferation of human keratinocytes was shown to increase when TNIP1 was downregulated, and this was further linked to more severe psoriasis-like conditions in a mouse model of dermatitis\textsuperscript{191}. In addition, genes also linked to skin include IL13 (interleukin 13), a cytokine shown to induce B cell proliferation\textsuperscript{192}, TRAF3IP2 (TRAF3 interacting protein 2), a gene involved in immune regulation and nuclear factor kappa B signalling (also a characteristic of TNIP1). Nuclear factor kappa B signalling has been described as an essential process in psoriasis\textsuperscript{193}. Four other genes were linked to skin including PTRF (polymerase I and transcript release factor), ETS1 (ETS proto-oncogene 1), ZNF816-ZNF321P (annotated as ZNF816 ZNF321P readthrough) and ZNF816 (zinc finger protein 816).

This analysis of psoriasis GWAS data shows that there is a separation in H3K36me3 enrichment signal between two distinct SNP sets that act differently in different
tissues. Through FORGE2 I can separate the original GWAS SNP list into these 2 subsets and show that they contain SNPs located in genes linked to distinct biological processes that act both in skin cells and B cells, potentially underlying the mechanisms driving psoriasis pathology.

4.3 Conclusions on GWAS analyses

I have uncovered many cell type- and tissue-specific enrichments for disease-associated GWAS variants. Some of these enrichments replicate findings made by FORGE, and many are novel, such as the uncovering of a previously unreported H3K27me3 enrichment across several traits, and the discovery of skin regions associated with psoriasis GWAS SNPs. I have explored the case of psoriasis further using computational approaches, showing potential gene targets associated with this H3K36me3 enrichment signal, and separating skin and blood components for further study. It is important to consider that one of the most important applications of FORGE2 is to prioritise variants for experimental analysis. As an example of this application I have used FORGE2 to prioritise variants for 4C-seq analysis, complementing FORGE and demonstrating the utility of studying variants in a tissue- and histone mark-specific context.
5 4C-seq results
5.1 Introduction

Intergenic and intronic disease-associated GWAS and EWAS variants present a challenge for functional interpretation\textsuperscript{17,136}. Improved data and novel computational approaches are required to rise up to this challenge. In previous sections I have shown the application of eFORGE and FORGE2 to uncover the cell type-specific action of disease-associated variants. I have also highlighted that these bioinformatics approaches prioritise targets in a cell type-specific context for experimental study. In addition to designing and developing these computational approaches, I have implemented 4C-seq experiments to analyse regions prioritised by eFORGE and FORGE2, providing valuable data on the cell type-specific interactions of disease-associated variants.

5.2 4C-seq of regions homologous to neural disease loci in mNSCs

The laboratory mouse is one of the main model organisms in biology\textsuperscript{194}. Substantial resources are available when studying the mouse genome, which has provided many insights in the study of human diseases\textsuperscript{194,195}. In addition, an important reason to use mNSCs as a test system for initial 4C-seq experiments was that mNSCs were an established cell type for 4C-seq in the protocols of my collaborators (Hadjur group)\textsuperscript{196,197}. Using mNSCs I can focus on regulatory interactions in a developmental context where many variants are posited to act\textsuperscript{198}. 
Results

In order to study neural disease-associated loci I prioritised 6 neural disease-associated regions from published GWAS and EWAS for 4C-seq analysis (see Methods for prioritisation strategy). As mentioned previously, to prioritise these SNPs I took into account multiple levels of information, including DNase I hotspot data, eQTL status, PCHi-C data and sequence homology (figure 5.1). While some caveats can be made regarding the evolutionary distance between human and mouse, including the fact that only 35.6% of DHSs are conserved between the two species\textsuperscript{199}, evidence suggests that for homologous sites a much higher proportion of DHSs (59.8%) is conserved\textsuperscript{199}. Examples of sites with conserved regulation between human and mouse include GWAS sites such as the obesity-associated FTO locus\textsuperscript{130,135} and the QRS duration-associated SCN10A locus\textsuperscript{200}. 

134
Figure 5.1: Schematic of 4C-seq mNSC computational and experimental analysis. Human and mouse sections are clearly separated. Computational analysis included GWAS/EWAS data which were intersected with data for DNase I hotspots, PCHi-C contacts and eQTL/eQTM. Evidence for region homology was taken from UCSC LiftOver (remap ratio 0.95) and loci were subject to visual examination on the UCSC genome browser. 4C-seq experiments were then performed in mNSCs for each of the selected regions. Human and mouse icons were taken from Odom et al., 2007201.

In my analyses of mNSC interactomes I found evidence of distal interaction in two cases. Specifically, homologous regions to cg02672452 and rs548181 showed interaction with Wdr1 and Fez1, with relevance for the aetiology of Alzheimer’s disease and schizophrenia.
cg02672452 is an EWAS DMP found to be associated with Alzheimer's disease by two separate studies\textsuperscript{202,203}. These EWAS studied different areas of the brain in search of DMPs associated with Alzheimer's disease. To my knowledge, the region containing cg02672452 was not previously linked to any gene by Hi-C or PCHi-C. In addition, this region is not known to be an eQTM (expression Quantitative Trait Methylation, a genomic CpG associated through DNA methylation to the expression of a particular gene)\textsuperscript{151}. cg02672452 is located in an intron of Homo sapiens lymphocyte-specific protein 1 pseudogene (LOC654342). The homologous region in mouse was located between Otop1 and Drd5 genes. cg02672452 was found to overlap DNase I hotspots across many human tissues (including brain samples from the Roadmap Epigenomics Consortium).

For the homologous region to cg02672452 in mNSCs I observe downstream interaction (\textit{figure 5.2}), associating cg02672452 with Wdr1. WDR1 in humans codes for a protein that induces disassembly of actin filaments in conjunction with the ADF/cofilin family of proteins\textsuperscript{204}. ADF/cofilin-actin rods have been proposed as a therapeutic target for Alzheimer’s disease\textsuperscript{205}. The proposed mechanism is thus: the EWAS DMP cg02672452 is located in an enhancer region (marked by a DNase I hotspot) that loops with Wdr1, potentially regulating expression of Wdr1, which would have downstream effects in the mechanism of Alzheimer’s disease through the regulation of actin rod formation.
rs548181 is a GWAS SNP found to be associated with schizophrenia by several studies, including a family-based replication and refinement study\textsuperscript{206}. To my knowledge, the functional role of this SNP in schizophrenia is unknown, and all previous studies, including the aforementioned refinement study, could not elucidate a functional role for this SNP. rs548181 is located in an intron of Homo sapiens STT3A antisense RNA 1 (STT3A-AS1), a long non-coding RNA. This GWAS SNP overlaps DNase I hotspots in human brain samples from the Roadmap Epigenomics Consortium, and, in addition, is an eQTL for many human genes, including CCDC15, CHEK1, DDX25, EI24, ESAM, FEZ1, HEPACAM, HEPN1, PUS3, ROBO3, SRPR, STT3A and TBRG1 (source: Braineac database\textsuperscript{154}). A homologous region in mouse was identified using UCSC LiftOver. This homologous region is located in an intron of Mouse gene AK018988 (uc009otz.1, RIKEN identified cDNA), potentially a non-classified homologue of Homo sapiens STT3A antisense RNA 1 (STT3A-AS1).

For the region homologous to the human locus containing rs548181 in mNSCs I observe downstream interaction (\textit{figure 5.3}), associating rs548181 with Fez1, a gene strongly linked to schizophrenia by GWAS. I thus propose the following mechanism: rs548181 is found in an enhancer (marked by a human brain DNase I hotspot) that loops with the promoter and the gene body of Fez1. This variant, which is an eQTL for FEZ1 in human, is posited to directly regulate expression levels of FEZ1, thus potentially affecting the aetiology of schizophrenia. FEZ1 is already a schizophrenia-associated gene\textsuperscript{207}. In this example I observe eQTL mapping, GWAS variants and 4C-seq converging on functional disease mechanism.
Results
Figure 5.2: 4C-seq results for the mouse region homologous to the human locus that contains Alzheimer's disease EWAS DMP cg02672452. The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The colour scale was taken from Van de Werken et al., 2012. The positions of local genes for mm10 (middle) and homologous genes in other species (lower section, taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue.
Figure 5.3: 4C-seq results for the mouse region homologous to the human locus that contains schizophrenia-associated GWAS SNP rs548181. The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to
grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes for mm10 (middle) and homologous genes in other species (lower section, taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue.

Of the loci analysed, I also found several cases in which only local interactions could be detected within 2Mb (see Methods for loci features and association data). For these variants evidence seems to point to a local mechanism of action of the GWAS/EWAS signal.

Specifically, for the Gcdh locus I can observe proximal interactions with Calr, Farsa, Syce2, MIR7069, Gcdh and Klf1 (figure 5.4). Of all these genes three have a role in neurological processes. Farsa codes for a protein related to axonal development of hippocampal neurons\(^{208}\), GCDH deficiency has been linked to late-onset neurological disease\(^{209}\) and the human KLF family of genes has been linked to several neurological conditions\(^{210}\). In my opinion these three candidates merit further study to elucidate the action of GWAS SNP rs8012.

For the Synj2 locus I can observe interactions with Snx9, Synj2 and Serac1 (figure 5.5). Of these three genes two are involved in neuronal processes: the specific role of Synj2 is vesicle uncoating in neurons, linking this region to this key neurological process\(^{211}\). Serac1 is linked to the neurological conditions Leigh
Results

syndrome and MEGDEL syndrome\textsuperscript{212}. In my opinion these two candidates merit further study to elucidate the meQTL action of SNP rs1009014.

For the Ppm1m locus interactions can be observed with a range of genes including Glyctk, Mirlet7g, Wdr82, Ppm1m, Twf2, Tlr9, Alas1 and Poc1a (figure 5.6). Of all these genes Glyctk plays an important metabolic role in neural tissue\textsuperscript{213} and Tlr9 activation has been linked to effects on spatial learning and memory\textsuperscript{214}. In my opinion these two candidates merit further research to elucidate the action of GWAS SNP rs7618915.

In order to validate the 4C-seq pipeline, which I established in the Beck lab for the first time, I chose to run a positive control for the 4C-seq experiment on mNSCs (figure 5.7). Region 6381 (a region proximal to the Deptor gene with an established 4C-seq profile in the experiments of the Hadjur team) was chosen and identical primers were used for my 4C-seq run. Furthermore, in addition to using positive controls from the same lab and with the same PCR probes for generating the 4C-seq libraries (figure 5.7), I also performed a duplicate sequencing run on the same 4C-seq library. Results obtained were identical to the previous sequencing run, showing no differences introduced by the sequencing process. Given the performance of positive controls and the replicate sequencing runs, I am confident that library generation was performed adequately, and that 4C-seq results are robust.
Figure 5.4: 4C-seq results for the Gcdh locus in mouse NSCs (homologous to human variant rs8012). The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue. For this region positional information for homologous regions in human is shown in figure S1 (see Appendices).
Figure 5.5: 4C-seq results for the Synj2 locus in mouse NSCs (homologous to human variant rs1009014). The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue.
Figure 5.6: 4C-seq results for the Ppm1m locus in mouse NSCs (homologous to human SNP rs7618915). The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue. For this region positional information for homologous regions in human is shown in figure S2 (see Appendices).
Figure 5.7: **4C-seq positive control.** Positive control for a prior 4C-seq experiment (upper panel) compared to the same 4C-seq control in my experiment (lower panel). Peaks are located at the same genomic positions. The PCR primers for generating 4C-seq libraries were identical and obtained from the same group (the Hadjur lab at UCL). The black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (left red arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the
enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value.

5.3 4C-seq of immune disease loci in sorted blood cells

eFORGE, FORGE2, and other approaches have shown that variants associated with several autoimmune conditions are preferentially located in leucocyte enhancer loci\textsuperscript{81,146}. However, the identification of target genes for such enhancers is challenging as different immune cell types may present radically different interactomes at a local level\textsuperscript{147}. In addition, even if only one cell type is affected, this cell type may have far-reaching importance in immune regulation through cytokines and other signalling molecules, affecting multiple cell types and pathways\textsuperscript{177}. Therefore it is important to obtain high-resolution interactome data for different immune cell types in order to gain a precise understanding of the functional consequences of disease-associated variants. I have thus performed 4C-seq across a range of sorted immune cell types for selected candidate loci, with the aim of uncovering cell type-specific interactions. These highly-specific immune interactions may offer key insights into the mechanism of action of disease-associated variants (figure 5.8).
Figure 5.8: Schematic of computational and experimental analysis for human immune GWAS regions. Computational analysis included the intersection of GWAS data with data for enhancers, PCHi-C contacts and eQTL. 4C-seq experiments were then performed in CD14+, CD4+, CD8+, B cells and NK cells for each of the selected regions. The human icon was taken from Odom et al., 2007.201

5.3.1 4C-seq analysis of the LRRC8B locus

For the LRRC8B locus I sought to characterise a potential enhancer located within an intron of LRRC8B (hg19 coordinates chr1:90022562-90022945). Variants within this region have been associated with adverse response to paclitaxel/carboplatin chemotherapy (neutropenia/leucopenia).
Results

For this region I can observe a higher number of interactions in CD4+ cells (figure 5.9), both within the body of LRRC8B (an enhancer region, marked by an orange chromatin state in GM12878 B lymphocyte-derived cells), the LRRC8B promoter (marked by a red chromatin state across the 9 ENCODE cell lines), regions proximal to the promoter (2 potential enhancers, however, the chromatin state track does not find an enhancer state for these locations in the 9 ENCODE cell lines) and also with LRRC8C. B cells also seem to show interactions with one of the potential enhancers (which also is present in very low signal for CD8+ and NK cells, but not CD14+ cells, this seems to point to the fact that this is a lymphoid-specific enhancer). The absence of interactions with other genes within 2Mb (e.g. LRRC8D) seems to point to LRRC8B and LRRC8C as the interacting genes for this GWAS-associated region. CD14+ cells seem to also present enhancers for LRRC8B (the first CD4+ cell enhancer, and not the lymphoid-specific enhancer). In addition, CD14+ cells seem to also present interactions with LRRC8C.

LRRC8 is a component of volume-regulated anion channels (VRAC). VRACs have been shown to mediate cisplatin/carboplatin uptake, and are also linked to cisplatin/carboplatin resistance\textsuperscript{215}. I propose that variants within this enhancer region have a role in VRAC formation in immune cells by affecting LRRC8B and LRRC8C, thus mediating adverse response to paclitaxel/carboplatin chemotherapy and leading to neutropenia/leucopenia. Of note, one candidate variant within the region (rs4658279) was shown to be an eQTL for LRRC8B in GTEx data\textsuperscript{155}. 
Results

B

CD4+

CD8+

CD14+

NK
Figure 5.9: **4C-seq results for the LRRC8B locus.** Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80\textsuperscript{th} and 20\textsuperscript{th} percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue. In addition, several epigenetic data tracks are included comprising H3K27ac data across 7 ENCODE cell lines (GM12878, H1-hESC, K562, HUVEC, HSMM, NHEK and NHLF), DNase I hypersensitivity clusters across 125 ENCODE cell types, ENCODE transcription factor ChIP-seq data (161 factors), and Hidden Markov Model (HMM) chromatin state segmentation across 9 ENCODE cell lines (GM12878, H1-hESC, K562, HepG2, HUVEC, HMEC, HSMM, NHEK and NHLF, from top to bottom). Different colours in the HMM track correspond to different chromatin states: Active Promoter (Bright Red), Transcriptional elongation/transition (Dark Green), Polycomb-repressed (Gray), Heterochromatin (Light Gray), Weak transcribed (Light Green), Weak Promoter (Light Red), Strong enhancer (Orange), Inactive/poised Promoter (Purple), Weak/poised enhancer (Yellow), Insulator (Blue).
5.3.2 4C-seq analysis of the IKZF3 locus

For the IKZF3 locus I sought to characterise a potential enhancer located between GRB7 and IKZF3 (hg19 coordinates chr17:37912171-37912498). Variants within this region have been associated with inflammatory bowel disease (IBD), specifically ulcerative colitis and Crohn’s disease.

For this region I can observe contacts with the promoter of a short isoform of IKZF3 in B cells (marked by a red chromatin state in GM12878 B lymphocyte-derived cells, figure 5.10). In CD4+ cells I can observe strong contacts with the promoters of several isoforms of IKZF3 (marked by red chromatin states in GM12878 cells), as well as with several parts of the IKZF3 gene body. I can also observe a cluster of interactions with a group of proximal genes, including ERBB2, MIEN1, GRB7 and MIR4728 (cluster 1). This suggests a concerted set of genomic contacts between these genes, which has previously been suggested in studies on the coordinated activation of these genes in breast cancer\textsuperscript{216}. In addition, I can observe interactions with the promoter of STARD3. In CD8+ cells I can only observe contacts with ERBB2 and IKZF3 (and not STARD3 or the rest of the cluster previously observed in CD4+ cells). In CD14+ I can observe strong contacts with the short isoform of IKZF3. Contacts are also present with ERBB2 and cluster 1. I can also observe contacts with STARD3 and two new regions, one proximal to NEUROD2 and the other proximal to PPP1R1B. In NK cells I can observe strong interactions with the promoter of the short isoform of IKZF3, and
weak interactions with the longer isoform of IKZF3. Strong interactions are present with cluster 1 and with ERBB2 (and MIR4728). No interactions with STARD3 can be observed.

Experimental results show evidence for an intricate regulation of this critical immune-related region, epitomising the complex mosaic of alternate enhancer contacts in different immune cell types. I propose that variants within this region have a role in IBD through the regulation of IKZF3 expression, leading to effects on downstream immune regulation. This is supported by previous findings that have linked differential expression of IKZF3 with IBD\textsuperscript{217}. To further this association I uncover the cell type-specific nature of enhancer-IKZF3 interactions for different IKZF3 isoforms. IKZF3 has also been associated with other autoimmune diseases such as MS\textsuperscript{127}. Previous studies on IKZF3 have proposed enhancer activity for a variant in my candidate region (rs12946510)\textsuperscript{127}. I have demonstrated that this proposed enhancer loops to IKZF3 and provided the first high-resolution 4C-seq map across several immune cell types for this locus. Importantly, these contacts cannot be observed at the resolution of BLUEPRINT promoter Capture Hi-C data\textsuperscript{150}. 

Results
Results

DNase Clusters

Transcription Factor ChIP-seq (161 factors) from ENCODE with Factorbook Motifs

Chromatin State Segmentation by HMM from ENCODE/Broad

UCSC Genes (RefSeq, GenBank, CCDS, Rfam, tRNAs & Comparative Genomics)

H3K27Ac Mark (Often Found Near Active Regulatory Elements) on 7 cell lines from ENCODE

DNaseI Hypersensitivity Clusters in 125 cell types from ENCODE (V3)

CD4+

CD8+

CD14+

NK
**Figure 5.10: 4C-seq results for the IKZF3 locus.** Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, [https://genome.ucsc.edu/](https://genome.ucsc.edu/)) are also indicated in blue. In addition, several epigenetic data tracks are included comprising H3K27ac data across 7 ENCODE cell lines (GM12878, H1-hESC, K562, HUVEC, HSMM, NHEK and NHLF), DNase I hypersensitivity clusters across 125 ENCODE cell types, ENCODE transcription factor ChIP-seq data (161 factors), and Hidden Markov Model (HMM) chromatin state segmentation across 9 ENCODE cell lines (GM12878, H1-hESC, K562, HepG2, HUVEC, HMEC, HSMM, NHEK and NHLF, from top to bottom). Different colours in the HMM track correspond to different chromatin states: Active Promoter (Bright Red), Transcriptional elongation/transition (Dark Green), Polycomb-repressed (Gray), Heterochromatin (Light Gray), Weak transcribed (Light Green), Weak Promoter (Light Red), Strong enhancer (Orange), Inactive/poised Promoter (Purple), Weak/poised enhancer (Yellow), Insulator (Blue).
5.3.3 4C-seq analysis of the NCF4 locus

For the NCF4 locus I sought to characterise a potential enhancer located in an intron of NCF4 (hg19 coordinates chr22:37258335-37258758). Variants within this region have been associated with atopic dermatitis.

For this region I can observe several interactions with NCF4 in B cells (figure 5.11). I can also observe interactions with NCF4 for CD4+ cells, and also with the promoter of one of the isoforms of CSF2RB. FANTOM5 found this region to be a monocyte-specific enhancer. I find interactions to be the strongest in monocytes, covering various points across NCF4 and CSF2RB. In addition, there are monocyte-specific contacts with an intergenic region (potential enhancer) between CSF2RB and TEX33. For NK cells I observe a radically different landscape, with no contacts detected within 2Mb.

I propose that variants within this region have a role in atopic dermatitis through the cell type-specific modulation of NCF4 and CSF2RB in CD14+ cells. NCF4 is a component of phagocyte NADPH-oxidase, which has a key role in myeloid-mediated host defence. CSF2RB is a high-affinity receptor for several cytokines including IL-3, IL-5 and CSF2 (which has a role in monocyte activation\(^\text{218}\)). Of note, the NCF4 variant rs4821544 has been associated with other autoimmune conditions such as ileal Crohn’s disease\(^\text{219}\). Importantly, I uncover contacts for this region outside NCF4 (specifically with CSF2RB and a proposed enhancer located between CSF2RB and TEX33), with a strong cell type-specific CD14+ signal.
Results

Figure 5.11: 4C-seq results for the NCF4 locus. Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD14+, and NK). For this region CD8+ 4C contacts were excluded from final set due to quality filters. For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80th and 20th percentiles of normalised read coverage at the same window size are also shown (upper and
lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue.

5.3.4 4C-seq analysis of the CSF2 locus

For the CSF2 locus I sought to characterise a potential enhancer located between CSF2 and P4HA2 (hg19 coordinates chr5:131430055-131430830). Variants within this region have been associated with rheumatoid arthritis.

For this region I can observe a highly conserved interaction downstream from the bait for CD4+ cells, CD8+ and CD14+ cells (but not NK or B cells, figure 5.12). Other downstream interactions are present for B cells (interaction 1), CD4+ cells (interaction 2, with P4HA2) and CD14+ cells (interaction 3), these interactions are highly cell type-specific. B cells also present an upstream interaction with CSF2. NK cells present a non-interacting landscape with no contacts outside the bait.

I propose that variants within this region have a role in RA through the cell type-specific modulation of P4HA2 in CD4+ cells and CSF2 in B cells. P4HA2 codes for a key enzyme in collagen synthesis. Collagen has been proposed as an autoantigen in Rheumatoid Arthritis (RA)\(^{220}\) and has been used in trials as a treatment for RA\(^{221}\). P4HA2 is expressed in whole blood and lymphoblastoid cells, among other tissues\(^{222}\).
Results

CSF2 (also known as granulocyte-macrophage colony-stimulating factor) has been proposed as a target in RA\textsuperscript{223}. The cytokine produced by the CSF2 gene can be detected in high quantities in joints with rheumatoid arthritis\textsuperscript{224} and drugs are being developed to block CSF2\textsuperscript{224}. Of note, this particular genomic region has also been associated with inflammatory bowel disease (IBD), juvenile idiopathic arthritis and asthma\textsuperscript{225}. P4HA2 has also been associated with other autoimmune conditions such as giant cell arteritis\textsuperscript{222}. 
**Figure 5.12: 4C-seq results for the CSF2 locus.** Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80th and 20th percentiles
of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue.

5.3.5 4C-seq analysis of the CCR6 locus

For the CCR6 locus I sought to characterise a potential enhancer located in an intron of CCR6 (hg19 coordinates chr6:167534229-167534357). Variants within this region have been associated with rheumatoid arthritis.

For this region I can observe clear differences in interactions between cell types (figure 5.13). B cells present several interactions within CCR6. In CD4+ cells I can observe interactions within CCR6 short isoforms (and GPR31 and the promoter of TCP10L2), and also 6 upstream interactions, some of which overlap the longer CCR6 isoform and the body of the FGFR1OP gene. The furthermost interaction overlaps the RNASET2 gene. In CD8+ cells I can observe strong interactions within CCR6, with the promoter of TCP10L2 and with FGFR1OP. A weaker interaction with GPR31 can be noted. In CD14+ cells I can see some interactions shared with CD4+ and CD8+ cells, with CCR6, GPR31, TCP10L2 and FGFR1OP targeted. Furthermore, the RNASET2 interaction observed in CD4+ cells is also
Results

observed here. In addition, a novel interaction with TCP10 only present in CD14+ cells can be observed. In NK cells I can observe interactions with CCR6, TCP10L2 (but not GPR31) and FGFR1OP.

SNP rs3093023 (within this region) is an eQTL for RNASET2 in GTEx data and other variants within this region have been associated with RNASET2 expression. I detect RNASET2 interactions for CD4+ and CD14+ cells. This suggests that this region is an enhancer that directly interacts with RNASET2 in CD4+ and CD14+ cells. RNASET2 is a ribonuclease that has previously been associated with AI conditions such as Graves’ disease. Another variant within the interacting area (rs3093024) was found to be a CCR6 eQTL. This SNP has also been associated with interleukin-17 quantities in the serum of subjects with RA. rs3093024 has also been associated with other autoimmune conditions such as Graves’ and Crohn’s diseases. CCR6 is a chemokine with a key role in inflammatory response. I propose that variants within this region have a role in rheumatoid arthritis through the modulation of the inflammatory cytokine CCR6 across the 5 immune cell types, and the cell type-specific modulation of RNASET2 in CD4+ and CD14+ cells.
Figure 5.13: 4C-seq results for the CCR6 locus. Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80\textsuperscript{th} and 20\textsuperscript{th} percentiles of normalised read coverage at the same window size are also shown (upper and
lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue.

5.3.6 4C-seq analysis of the TRAF1/C5 locus

For the TRAF1/C5 locus I sought to characterise a potential enhancer located between PHF19 and TRAF1 (hg19 coordinates chr9:123652531-123653088). Variants within this region have been associated with rheumatoid arthritis.

For this region I can observe several interactions with PHF19 and TRAF1 in B cells (figure 5.14). In CD4+ cells I can observe interactions with TRAF1, PHF19, C5 and PSMD5. In CD8+ cells I can observe similar interactions to B cells, within PHF19 and TRAF1. In CD14+ cells I can observe similar interactions to CD4+ cells, that is, with TRAF1, PHF19, C5 and PSMD5. In addition, a weak interaction (potentially an enhancer) can be observed upstream of FBXW2. In NK cells I can only observe interactions with TRAF1 and PHF19, in a similar way to B cells. To summarise, I can observe a few local interactions for B cells, CD8+ cells and NK cells, and radically different interaction maps for CD4+ and CD14+ cells, with more distal genes involved.
Results

Despite the fact that interactions are detected for PHF19 and PSMD5, the main biological evidence supports a role for TRAF1 and C5 in this area\textsuperscript{228}. TRAF1 is a negative regulator of inflammation with a role in rheumatic diseases\textsuperscript{229}. C5 is a crucial component of the complement system in innate immunity. C5 has been targeted successfully in treatments for autoimmune disease models such as experimental anti-phospholipid antibody syndrome\textsuperscript{230}. I propose that variants within this region have a role in rheumatoid arthritis through the modulation of TRAF1 across the 5 immune cell types, and the cell type-specific modulation of C5 in CD4+ and CD14+ cells. Of note, variants in this region are eQTL for C5, PSMD5-AS1 and TRAF1 in several immune cell types\textsuperscript{156}.
Figure 5.14: 4C-seq results for the TRAF1/C5 locus. Each panel corresponds to one cell type indicated below (i.e. B cell, CD4+, CD8+, CD14+, and NK). For each of these panels the black line corresponds to the median of normalised read coverage (computed for running 5 kb windows, top section). Peaks in this line indicate contact with the bait region (black arrow). The 80\textsuperscript{th} and 20\textsuperscript{th} percentiles
of normalised read coverage at the same window size are also shown (upper and lower limits of the grey area). Below this, the colour-coded multiscale diagram (ranging from red to grey) represents the enrichment of median values at different sliding window sizes (2kb at the top to 50kb at the bottom) relative to the maximum attainable 12kb median value. The positions of local genes (taken from the UCSC genome browser, https://genome.ucsc.edu/) are also indicated in blue.

In summary, I have applied 4C-seq to disease-associated regions across several immune cell types, uncovering a range of cell type-specific interactions, the most notable of which is a previously unreported IKZF3 interaction for a region containing IBD-associated SNP rs12946510. IKZF3 is a druggable target, and anti-IKZF3 drug lenalidomide (a variant of thalidomide) has already been developed for multiple myeloma\textsuperscript{231,232,233}. Evidence of repurposing of thalidomide variants for IBD is inconclusive, and despite adverse effects there is a scope for thalidomide application in certain cases of IBD\textsuperscript{234}. Further research and target validation efforts are necessary, yet the detection of IKZF3 as a target gene is an important step forward in view of uncovering druggable IBD-associated pathways.

In conclusion, the associations uncovered by my approach merit further experimental study. I have highlighted cell type-specific interactions across a range of cell types, generating the first 4C-seq maps for an important immune
cell type (NK cells), and revealing a range of novel interactions for non-protein-coding regions of the genome.
6 Discussion
6.1 eFORGE

6.1.1 eFORGE aims and context

One of the major objectives of performing an EWAS is to obtain novel biological information about a certain disease or trait. When dealing with large amounts of complex data the analysis of the results of the study (in order to obtain this novel biological information) is typically performed using bioinformatics methods. Currently, many methods used to analyse EWAS results are gene-centric, that is, they are based on gene annotations (or relevant proteins). Standard examples of this approach include Gene Ontology (GO) or Kyoto Encyclopaedia of Genes and Genomes (KEGG) pathway analysis. Many EWAS obtain gene lists from differentially methylated regions (e.g. differentially methylated promoters). These lists are then analysed with gene ontology (GO) algorithms to detect over-represented annotation groups. Such methods are the general bioinformatics analysis choice, not only for EWAS, but also for other fields including GWAS and gene expression studies.

Intergenic probes make up a quarter of the 450k array and a higher proportion of the 850k array. Many intergenic EWAS DMPs do not easily map to target genes, and are therefore discarded when performing GO analysis or similar gene-centric approaches. In addition, gene-centric analyses typically result in weaker enrichment scores than those obtained from gene expression studies. Alternative approaches
Discussion

are thus required to allow for a more appropriate method to analyse EWAS results, using information for all differentially methylated positions. A range of EWAS analysis tools have been developed that use genome-wide features^8 (and thus are distinct from gene-centric approaches). It is in this context that eFORGE has been designed, bringing forwards the FORGE genome-wide functional overlap analysis approach to EWAS data.

When analysing genome-wide data for a given cell type one can focus on different epigenetic marks: histone marks (e.g. the activating H3K4me3, or H3K27me3, which is found in facultatively repressed genes^71), DNase I Hypersensitive Sites (which signal for classical cis-regulatory elements^159) and DNA methylation (which is associated with the regulation of gene expression^238). In the FORGE project (the analysis method designed for GWAS), the focus was on DNase I Hypersensitive Sites (or DHSs), as DHSs mark all major classes of cis-regulatory elements^199. The FORGE tool thus performs a functional overlap analysis (using a processed form of DNase I cleavage data known as DNase I hotspots^158) with top GWAS SNPs. The idea behind this functional overlap analysis is that top GWAS SNPs will tend to overlap with more DNase I hotspots in tissues where the disease is relevant, as a portion of the non-protein-coding GWAS SNPs will be acting through classical cis-regulatory elements, regulating gene expression in target tissues. For gene expression to be affected in target tissues, these cis-regulatory elements must be active in those tissues. In addition, it is expected that there will be a higher number of top GWAS SNPs overlapping DNase I hotspots in those tissues when compared with tissues with no disease relevance. The proposed underlying phenomenon is that top GWAS SNPs are
Discussion

enriched for tissue-specific cis-regulatory elements in relevant tissues for the trait or
disease under study.

The analysis approach performed by FORGE avoids the aforementioned gene-centric
bias and points to a link between particular diseases and tissue-specific DNase I
hotspot enrichment. Using epigenomic data to obtain disease-associated tissue-
specific enrichment signals is one of the aims of ENCODE, Epigenomics Roadmap and
BLUEPRINT data generation. The objective of the eFORGE project was to develop a
bioinformatics tool that analyses top EWAS hits and returns tissue-specific
enrichment results for Epigenomics Roadmap, ENCODE and BLUEPRINT tissues. For
my eFORGE project hypothesis, I proposed that the tissue-specific signal that is
observed for GWAS in FORGE would also occur for EWAS as a systematic and
consistent phenomenon. I expected tDMR and EWAS regions to present tissue-
specific enrichment for DNase I Hotspots in the same way as GWAS regions do, thus
highlighting relevant tissues and cell types for EWAS. The underlying idea behind this
is that a proportion of top EWAS DMPs reflects alterations in cis-regulatory elements
in relevant tissues (some EWAS arrays, such as the 850k array, have been
purposefully designed to capture this phenomenon²³⁷). If this is the case, in a similar
way to GWAS I expect a higher number of overlaps between EWAS DMPs and DNase
I hotspots in relevant tissues when compared with tissues with no disease relevance.
In a similar way, the rationale behind expecting tDMRs to be enriched for tissue-
specific DNase I hotspots is based on the notion that a proportion of tDMRs are
formed due to tissue-specific cis-regulatory processes, and are therefore expected to
overlap tissue-specific DNase I hotspots (a proxy for active regions in a given tissue).
Early in the project it became clear that one of the main differences in eFORGE (in relation to FORGE) would be the background used in the statistical test. The background in FORGE consists of SNPs from two different types of arrays (GWAS typing arrays and the Illumina_HumanOmni2.5 array). The GWAS typing arrays include a variety of arrays listed in the Methods section, including, among others, the Affymetrix GeneChip 500K array and the Illumina Cardio-MetaboChip array. In the case of EWAS there are a variety of different technologies that can be applied. When choosing the background for eFORGE I had to choose among the following approaches:

- Illumina GoldenGate assay
- Illumina 27k array
- Illumina 450k array
- RRBS (Reduced Representation Bisulphite Sequencing)
- MeDIP-chip
- MeDIP-seq

Among these choices I selected the Illumina 450k array, due to several reasons: it is a widely used method, which allows for a high level of reproducibility between studies. In addition, DMPs are limited to array probes, which simplifies background generation. Using an array for eFORGE background design also makes the approach conceptually more similar to FORGE analysis. In addition, as 94% of 27k array probes
are present on the 450k array, a 27k array background can be generated as a subset of a 450k array background, allowing for the analysis of a higher number of EWAS.

Going forward, the eFORGE approach could be extended to analyse EWAS performed using other technologies. One of the most obvious candidates in terms of current numbers would be MeDIP-seq. MeDIP-seq is not array-based but sequencing-based, in a similar way to ChIP-seq\textsuperscript{239}. This fact means that MeDIP-seq allows for genome-wide discovery of differentially methylated regions (DMRs). However, MeDIP-seq presents some disadvantages, including a bias for regions with a high CG content\textsuperscript{240}. Considering the limited nature of Illumina DNAm arrays, and the CG-skewed data obtained with MeDIP-seq, another alternative would be Whole-Genome Bisulphite Sequencing (WGBS). With the current decrease in the costs of sequencing it is not unlikely that WGBS will be the method of choice for EWAS in the future.

6.1.2 eFORGE findings and applications

Given the implementation of FORGE and FORGE2, I sought to address the following question: does the field of epigenome-wide association studies (EWAS) present this same phenomenon of tissue-specific enrichment for histone marks and DNase I hotspots? The answer obtained after analysing many EWAS is yes, the field of EWAS
presents the same phenomenon of tissue-specific enrichment for DNase I hotspots and histone marks (see Results).

It is important to highlight that the correct interpretation of this tissue-specific enrichment can vary, as EWAS are subject to many confounders, including cell composition effects as mentioned previously\textsuperscript{18}. Given that these cell composition effects are driven by tissue-specific DNA methylation (DNAm), and that tissue-specific DNA methylation and tissue-specific DNase I hotspots tend to co-localise, eFORGE has also demonstrated the capability of detecting cell composition effects. This adds a novel approach for the detection of cell composition effects, an area of vital importance to the field of EWAS.

eFORGE has been applied across a range of diseases and traits\textsuperscript{146,241,242,243}. Specifically for three autoimmune diseases (rheumatoid arthritis, systemic lupus erythematosus and Sjögren’s syndrome) eFORGE has identified target immune cell types. eFORGE has also detected an immune component in a multiple sclerosis EWAS performed on non-pathological brain tissue. Despite a non-pathological macroscopic tissue aspect, it may well be the case that subtle changes in immune activation (reflected in DNA methylation differences) were an important component detected by the study. “Priming” or “pre-activation” changes in immune cells could be present in a tissue with a non-pathological macroscopic aspect. These subtle changes could however be detected by the EWAS array and be reflected in the change in DNA methylation status for a specific genomic site. In this analysis eFORGE supports the notion of immune activation underlying EWAS signal.
eFORGE has also been applied to detect cell composition effects, as is demonstrated by the study of granulocyte/lymphocyte ratio changes in the immune response to ovarian cancer. From a theoretical perspective, I consider an unexpectedly cell type-specific signal for a whole blood study as a sign of cell composition effects at play. However, it is important to also consider disease aetiology and factors such as probe filtration. I have also analysed cell composition corrected whole blood studies, such as the Liu et al. EWAS on rheumatoid arthritis\textsuperscript{172}. For this analysis I have found less striking cell type-specific enrichment patterns, which are more consistent with the expected aetiology of this complex disease. My analyses show that eFORGE results for EWAS driven by cell composition effects differ from those for studies showing true cell type-specific effects, and I report these analyses as examples for future applications of eFORGE.

Cancer EWAS are characterised by radical changes in DNAm. Some of these changes may be involved in the silencing of tumour suppressor genes and the activation of oncogenes, for example by altered CpG island DNAm patterns. From a genome-wide perspective many of these changes form part of cancer epigenetic reprogramming that transforms cells to a state in which self-renewal and other cancer hallmarks are phenotypically present. Previous observations report cases in which the cancer epigenomic signature leads cells to a more “stem cell-like” state\textsuperscript{244}. In cancer, to achieve stem cell-like phenotypes such as pluripotency or dedifferentiation a cell may retrace its position in the developmental gradient of Waddington’s landscape (see Introduction) and reprogram itself to a “higher”, more primaeval state. As I have
noted in the chapter on DNase I hypersensitivity, for differentiated cells around 1/3 of DHSs are from a primaeval state, 1/3 are from a state of intermediate differentiation and 1/3 are from a fully differentiated state. In cancer epigenetic reprogramming a cell may reactivate many of the “primaeval state” DHSs, in a similar way to epigenetic reprogramming in iPSCs. This reprogramming will have a reflection on both DHSs and DNAm levels. Prior studies on DHSs and cancer have shown a messy trend, in which cells do not neatly regress to a primaeval state but “highjack” DHSs and cellular machinery to acquire primaeval state-associated traits among other phenotypes\textsuperscript{63}. In contrast to this, I report a clear stem cell signal across 5 cancer EWAS (see Results).

Due to the design of cancer EWAS, DNAm differences found are a reflection of common features between the many different cancer samples studied. Each individual cancer sample may show a mixture of functional and non-functional epigenetic alterations. However, when studying events across cancer samples, there may be a selection for the purely functional changes that ensure cell survival amidst the “noise” of arbitrary genomic events. Therefore, it is perhaps not surprising that I observe a clear stem cell signal across cancer EWAS, because this epigenetic reprogramming to an earlier stage is a key element to ensure cell survival (and is associated with the key feature of self-renewal, as also observed in iPSCs). I thus observe a more “purified” signal in my meta-analysis of these studies, which can be typical for an analysis of common trends originating as a consensus from many samples.
Discussion

In eFORGE analyses of cancer EWAS I can therefore observe a trend that supports notions of cancer epigenetic reprogramming and the reactivation of stem cell DHSs, in agreement with previous research. Further research is needed to uncover how this is achieved mechanistically. For now, I can state that this finding, surprising in its clarity, merits a closer look in regards to the underlying mechanisms.

Certain analyses have observed high probe signal across all tissues when analysing certain probe sets with eFORGE. This is likely to indicate that those sites show high constitutive activity (indicated by DNase I hotspot overlap) compared to background probes that are similar for MAF, distance to TSS and GC content.

eFORGE has shown many applications. The first application includes analysis of EWAS data to identify disease-relevant cell types from a heterogeneous tissue (e.g. whole blood). In addition, eFORGE can detect cell composition effects in EWAS data. In this context, eFORGE provides an approach that is complementary to DNAm-based tools. eFORGE detection of cell composition effects is based on the assumption that tissue or cell type-specific differentially methylated positions (tDMPs or cDMPs) are driving the confounding signal. Because eFORGE can detect any strong influence of tDMPs in an EWAS probe list, eFORGE is capable of detecting cell composition effects, as has been shown in the aforementioned study on ovarian cancer. It is important to add that eFORGE is provided as a web tool, facilitating analysis in a way not currently offered by competing methods.
eFORGE signal can also be used as a measure of quality control in tissue- and cell type-specific DNA methylation (DNAm) studies (as mentioned above, tDMP and cDMP sets can be traced independently back to their original tissue or cell type using eFORGE).

eFORGE also suggests that DNase I hotspot data can be used inversely to predict tissue-specific differentially methylated positions, supporting other approaches that focus on tDMP prediction using other layers of epigenetic data \(^{86}\). eFORGE has also revealed insights into mechanistic understanding of disease- and trait-associated regions (e.g. stem cell targets identified for cancer EWAS), and serves a key role in prioritising regions overlapping cell type-specific DNase I hotspots/histone mark peaks to pave the way for further experimental analyses (e.g. 4C-seq, or modifications of the epigenome through CRISPR/Cas9-based methods\(^{245}\)). The design of eFORGE is geared towards downstream experimental follow-up of EWAS DMPs. For this, eFORGE provides tables detailing cell type-specific DNase I hotspot/histone mark peak overlap results for each DMP analysed. Finally, eFORGE can prioritise histone marks relevant to disease-associated probe sets for further computational analyses (e.g. TFBS analysis following a histone mark-specific signal for a particular EWAS).

Technical improvements introduced by eFORGE (when compared to FORGE) include faster scaling of the binomial test by the use of logarithms instead of decimal digits, the introduction of histone mark datasets, faster scaling of bitstring-generating code, the redesign of the SQLite database to facilitate the addition of new datasets as tables and the open design of the database, which can now be generated from
scratch simply by the execution of a single script. These changes, and many other minor code additions, have been placed on an open repository as a resource for the bioinformatics community. In addition, a study has been performed on software reproducibility taking eFORGE as an example\textsuperscript{246}. Other advances made possible by eFORGE design include the development of the first EWAS catalogue, the eFORGE catalogue (as, unlike GWAS, EWAS are not yet listed on a centralised registry), and contributions to the design of FORGE2. The development and methods of eFORGE prepared the stage for the development of FORGE2, which in turn has provided further insights into the genomics of diseases. In an example of reproducibility and code reutilisation, eFORGE code improvements (such as the faster scaling of the binomial test) have been passed on to the FORGE2 tool.

6.1.3 Considerations and limitations of eFORGE

As DNA hypermethylation and hypomethylation are both present in EWAS, two different top probe sets may be analysed by eFORGE. I recommend that DNA hypo- and hypermethylated probes be analysed separately when possible in order to improve functional interpretation of EWAS results. In addition, in order to allow for a complementary analysis to enrichment I developed depletion analysis for eFORGE.

It is important to add that eFORGE can only provide analysis for cell types for which there are data available. These data may be processed open chromatin data (e.g. from DNase-seq or ATAC-seq), or processed histone mark data in broadPeak or
similar formats. Guidelines and code have been provided to add data in these formats to a local eFORGE database. eFORGE currently contains 454 samples in its online database. However, many cell types are still not present in the database and to analyse these I have had to turn to complementary datasets, such as in the aforementioned analysis of microglial enhancers on sites from the MS EWAS by Huynh et al.\textsuperscript{174}.

6.1.4 Outlook on eFORGE

Despite the fact that DNase I hotspot-based tools have been implemented for the field of GWAS\textsuperscript{139,137,138,247}, eFORGE represents a dramatic conceptual shift, focusing on a different field, the field of EWAS, which is noted for radical differences (such as cell composition effects)\textsuperscript{18}. Before eFORGE it had not been demonstrated in a systematic way that EWAS DMPs showed enrichment for tissue-specific DNase I hotspots. However, I have shown that tissue-specific DNase I hotspot enrichment is a systematic phenomenon across the EWAS field. After demonstrating this for an unbiased selection of EWAS, I have examined particular examples in more depth in order to unveil functional mechanistic insights, leading to differing biological interpretation for a published study on multiple sclerosis and the characterisation of a cell type-specific signal for ovarian cancer, in addition to demonstrating the detection of cell composition effects, and the observation of a stem cell signature across a range of cancer EWAS. Furthermore, analyses of blood-based EWAS for
autoimmune diseases have highlighted immune-specific regulatory elements for further study.

With eFORGE I have developed a tool that does not only highlight the EWAS probes that are likely to be functional in a cell type- or tissue-specific context, but also shows particular trends for sets of probes associated with different traits and diseases. The development of this tool makes the automated cell type-specific enrichment analysis of any array-based EWAS possible. This constitutes a useful addition to the current limited toolbox for the interpretation and analysis of EWAS data. From now onwards, any scientist performing an EWAS can obtain cell type-specific enrichment results in minutes, from an online tool that has been rigorously tested for performance. As mentioned previously, false positives are highly improbable, eFORGE results are highly reproducible and the execution time has been dramatically reduced from previous code. The lack of requirement of a local installation (due to the availability of a web tool) is an even stronger support for the already high reproducibility standard of this particular software. In addition, by providing eFORGE results as part of a first-in-field catalogue, the eFORGE catalogue, I have made study-specific graphs and tables publicly available for further functional exploration.

As a long-term product of this research I expect an increase in the understanding of the epigenetics of complex traits and diseases, and more specifically in the understanding and interpretation of EWAS results. eFORGE analyses press the case for the generation of additional epigenetic data for all main cell types. These datasets,
coupled with improved computational analysis of EWAS regions, will drive understanding of the mechanisms underlying the observed epigenetic changes, potentially leading to an improvement in the diagnosis and treatment of complex diseases.

6.2 FORGE2

6.2.1 FORGE2 aims and context

GWAS represent one of the biggest advances in the last decade for the genetic study of disease, highlighting scores of SNPs associated with a variety of disorders, including type 2 diabetes and schizophrenia\(^{123,248}\). However, the interpretation of these associations remains challenging, especially in view of uncovering the functional genomics underlying disease mechanism\(^{249,250}\). Recent breakthroughs in genomic methods (including CRISPR/Cas9\(^{251}\) and 3C-related technologies\(^{111}\)) have added momentum to this field of research, and aid the process of linking genetics and disease\(^{130}\). However, the community has not yet completely explored the possibilities offered by large-scale epigenomics data gathered through the ENCODE\(^{183}\), BLUEPRINT\(^{184}\) and Epigenomics Roadmap initiatives\(^{81}\). Mapping GWAS-associated potential regulatory elements to the relevant tissues through functional overlap analysis has been approached using DNase I hotspot data\(^{139,198}\), but not recent genome-wide histone mark data obtained through ChIP-seq\(^{81}\). To further the insights obtained through DNase I hotspot data it is necessary to separate the
Discussion

different classes of genomic elements (e.g. promoters, enhancers) driving the observed GWAS-DNase I hotspot enrichment. This problem can be approached using histone mark data. I have developed FORGE2 to explore this rich dataset.

6.2.2 FORGE2 findings

Applying FORGE2 to disease-associated SNPs as described in the GWAS catalogue yielded potential insights into disease aetiology. FORGE2 analysis includes five histone marks: H3K4me1, H3K4me3, H3K27me3, H3K36me3 and H3K9me3. I found the first four marks to be informative in the study of GWAS results. FORGE2 histone mark analysis proved an important complement to FORGE DNase I hotspot analysis for a variety of diseases, revealing previously undetected associations of particular cell types with specific diseases. In addition, many histone mark analysis results confirmed previous FORGE findings. This is true especially for certain autoimmune diseases, for which blood association was detected in H3K4me1, H3K4me3 and H3K36me3. Heart phenotypes such as QT interval and PR interval, which in FORGE show enrichment for heart tissue, also show enrichment for heart tissue in H3K4me1 and H3K4me3 analysis. These repeated findings add support to the FORGE2 histone mark analysis approach.

In addition to confirming previous findings, FORGE2 sheds light on novel disease-cell type associations. Novel associations of cell types with specific diseases include, for H3K4me1, the association of T cells with Graves’ disease and with vein graft stenosis in coronary artery bypass grafting, the association of haematopoietic stem cells with
mean platelet volume and F cell distribution, and the association of stem cells with attention deficit hyperactivity disorder. In addition, I observe a CD14+/haematopoietic stem cell signature for Alzheimer’s disease-associated SNPs. I also report improvements on cell type-specific enrichment patterns previously observed in FORGE. Using data for H3K4me1 (the most informative mark I have tested) I can find a stronger cell type-specific signal for shared systemic lupus erythematosus/systemic sclerosis SNPs. For this GWAS SNP set I observe an enrichment that is specific to natural killer cells. These findings support the relevance of studying the enhancer-associated mark H3K4me1 in the analysis of GWAS findings.

For H3K4me3, a promoter-associated mark\textsuperscript{183}, I find a novel enrichment in stomach tissue for SNPs associated with food allergy. In addition I also confirm findings for mean platelet volume also discovered in the H3K4me1 analysis. In total I can confirm 9 FORGE findings through H3K4me3 analysis. H3K4me3 enrichments allow for a more straightforward interpretation than DNase I hotspot enrichments, as many individual overlaps can be linked to promoter elements and a network can be built with the corresponding genes.

In my analysis of H3K27me3 regions, I find some remarkable associations including, among others, a B cell signal for SNPs associated with C reactive protein levels and a haematopoietic stem cell enrichment for thyroid cancer-associated SNPs. These findings may have an association with Polycomb-repressed regions, and a study of these may yield further insight into the reported associations. For H3K36me3, in
addition to confirming several FORGE findings, I detect a high general enrichment for SNPs associated with blood metabolite levels, and a skin- and blood-specific enrichment for SNPs associated with psoriasis. These findings highlight the relevance of transcription-associated regions and could also provide a link between alternative splicing and GWAS SNP mechanism.

For many GWAS datasets I observe an enrichment signature in relevant tissues for H3K4me1 broadPeaks, in agreement with previous findings that propose that many GWAS SNPs act through enhancer disruption\textsuperscript{252}. However, for some diseases I detect an enrichment for other histone mark broadPeaks. It is possible that different complex diseases act through different classes of regulatory elements, potentially meriting a distinction between diseases that have a genetic component mainly mediated through enhancers, or “enhancer diseases”, and diseases that are mainly mediated through other genomic elements (such as promoters, transcription-related regions, and Polycomb-repressed regions). FORGE2 allows for the separate analysis of 5 histone mark datasets enriched for different classes of regulatory elements, focusing analysis of cell type-specific enrichment signal to this finer level. A potential next step would be to generate a FORGE-like tool using chromatin state annotations derived from ChromHMM or ChromImpute to explore these enrichments further\textsuperscript{86,253}.
6.2.3 Considerations on FORGE and FORGE2

Several factors must be taken into account when comparing FORGE and FORGE2. These tools use different datasets, the first focusing on DNase I hotspot data and the latter focusing on histone mark broadPeaks. Although in many cases DNase I hotspots and histone marks can represent the same elements (such as potential enhancers in the case of the NK cell-specific signal for SNPs associated with SLE and Systemic Sclerosis), DNase I hotspots can represent many different regulatory element classes, including promoters, enhancers, and insulators. A tissue-specific DNase I hotspot enrichment signal may be indicative of an overlap with any or all of these regulatory element classes, whereas the histone marks in FORGE2 each present enrichment for a subgroup of regulatory elements, thus presenting a more interpretable analysis and in some ways deconvoluting FORGE signal. FORGE and FORGE2 also use different frameworks for dealing with multiple testing correction. FORGE uses tissue-level Bonferroni correction and FORGE2 uses an FDR-based approach (Benjamini-Yekutieli). In addition, FORGE and FORGE2 contain data for different tissues although in the cases for which I observe completely new enrichments (see Methods) all the FORGE2 tissues described are also present in FORGE.

6.2.4 Outlook on FORGE2

FORGE2 is a powerful new tool for cell type-specific enrichment analysis of complex disease-associated variation, highlighting new results not detected by FORGE. In
addition, FORGE2 facilitates the prioritisation of candidate variants, aiding experimental approaches. I provide this tool for use by the community, in order to aid the functional characterisation of GWAS SNPs in a cell type-specific manner, a pressing issue in view of the current lack of functional understanding of GWAS SNPs. I also report a list of novel tissue-disease associations in order to aid the characterisation of relevant GWAS-associated mechanisms through functional genomics approaches. In conclusion, I anticipate that FORGE2 will be a valuable tool for current research on disease genomics, detecting novel cell type-specific enrichments for GWAS SNP lists in a histone mark-specific context.

6.3 4C-seq analysis

6.3.1 4C-seq analysis aims and context

A large proportion of variants associated with diseases by EWAS and GWAS are either intergenic or intronic, and thus present challenges for functional interpretation. It has been proposed that a proportion of these variants may affect phenotype by influencing regulatory elements that control gene expression. This is supported by the fact that many EWAS and GWAS variants are preferentially located in open chromatin regions from tissues that have a known involvement in disease. Open chromatin regions, and particularly DHSs, are representative of all the main classes of cis-regulatory elements, including enhancers. Histone
Discussion

Mark enrichment analyses shown here (FORGE2, eFORGE) and also by other groups indicate that many tissue-specific open chromatin enrichments are replicated by analyses using data for H3K4me1, a classic enhancer mark, suggesting that many GWAS and EWAS variants do not only act through cis-regulatory elements but specifically enhancers, thus potentially affecting gene expression through enhancer modulation. This is also supported by complementary analyses that use chromatin state enhancer annotations. It is known that enhancers loop in three-dimensional space to regulate their target genes. Thus the identification of enhancer-located variants seems to point a mechanism of action in which variants affect enhancer activity and thus the expression of target genes. It is, however, hard to establish target genes for many enhancers, with some enhancer-promoter contacts spanning megabases. It has also been shown that the strategy to link variants to the closest gene is a poor predictor of enhancer-promoter contacts. In this context the application of 3D genomic techniques has proved effective, as in the study of the obesity-associated FTO locus and the QRS duration-associated SCN10A locus. In both of these cases a distal enhancer was linked to a gene involved in a key disease-related process. Through these examples it has become clear that knowledge of the 3D-genomic context is of great help in the characterisation of certain disease-associated loci. Furthering this effort, I proposed to study the three-dimensional context of selected GWAS loci in mNSCs (homologous regions) and human immune cell types, in order to increase knowledge of the 3D structure of these regions and establish candidate target genes for GWAS and EWAS variant action.
When studying genomic interactions there are many techniques to choose from, including FISH and 3C techniques such as 4C, 5C, Hi-C, capture Hi-C, capture-C, and ChIA-PET\textsuperscript{104,111} (discussed in the Introduction). Each one of these techniques shows specific advantages optimised for a particular application, and thus no one technique fits all 3D genomic studies. Of all these techniques the choice depends, as always, on the specific scientific question posed. In my study of the interactomes of GWAS and EWAS regions I also have had to balance cost with efficiency to select the most adequate technique for the purpose. Specifically, selection of FISH would have been inadequate considering the resources available and the scalability offered by current 3C-based technologies. Within 3C-based technologies, simple 3C (which measures one vs one interactions) would have been inadequate due to the low throughput it presents. In my study I sought to fine-map the proximal interactome for each of my candidate regions. Therefore a one vs all approach such as 4C-seq or capture-C was the most suitable, as I could repeat the technique for each of my candidate regions and I could analyse these with a high resolution that large-scale methods such as 5C, Hi-C or ChIA-PET cannot provide. Given the 4C-seq in-house expertise at the UCL Cancer Institute (Hadjur group) I decided to favour 4C-seq as my method of choice. To summarise, 4C-seq and capture-C both provide the ability to map fine details of local 3D-genomic structure, and I decided to use 4C-seq based on in-house expertise (facilitating the sharing of specific protocols, reagents and experimental material with comparable data).
In order to prioritise variants for 4C-seq analysis I have favoured a multidimensional data analysis approach using eFORGE and FORGE2, thus integrating the computational and experimental sides of this project (see Methods). In addition I have also used several other computational tools including FORGE, the Braineac database, the GTEx database, UCSC LiftOver and the CHiCP tool. Multiple datasets were used including DNase I hotspots, expression Quantitative Trait Loci (eQTL), FANTOM5 enhancer annotations and promoter Capture Hi-C contacts. It is important to highlight that for all these datasets the cell type-specific context of the data was taken into account, as each one of these epigenomic features varies between cell types. In addition, information regarding sequence conservation was taken into account for selecting mouse regions homologous to human GWAS SNP loci. Genomic context and disease association data were taken into account for all loci analysed.

Complementing efforts to characterise the disparity in 3D genomic structure among different types of blood cells I prioritised the study of cell type-specific interactions among immune cell types (rather than studying one cell type under several conditions). This was done in order to gain an understanding of the cell type-specific nature of candidate interactions, complementing my bioinformatics approaches that seek to uncover the candidate cell types involved in disease mechanism.

Given the availability of sorted human immune cell types from the BLUEPRINT project, of which I was a participant, and my objective to study the cell type-specific
nature of selected loci, I favoured a study design involving 5 immune cell types (CD14+, CD4+, CD8+, B cells and NK cells). Future approaches, using these data as a reference dataset for the proximal interactomes of selected loci, may choose to further research on these regions by analysing a single cell type from this study design under different conditions (e.g. activation with LPS) or in a case-control design, obtaining the same sorted cell type from healthy and diseased subjects.

6.3.2 4C-seq analysis findings

I have found a range of cell type-specific interactions for selected disease-associated loci. The most notable of the cell type-specific interactions detected was for the IKZF3 locus, linking intergenic variant rs12946510 to different isoforms of IKZF3 in CD14+, CD4+, CD8+, NK and B cells. This shows relevance for the aetiology of IBD, linking a GWAS variant to a target pathway of lenalidomide, a candidate drug projected to have efficiency in the treatment of IBD subtypes\textsuperscript{234}. Importantly, interactions between rs12946510 and IKZF3 cannot be observed at the resolution of BLUEPRINT promoter Capture Hi-C data\textsuperscript{150}. The uncovering of this interaction supports the implementation of a targeted high-resolution 4C-seq approach to study GWAS loci, providing evidence for enhancer contacts not observable in current state-of-the-art data.

Additionally, several eQTL associations were studied. Notably, a homologous region to that containing rs548181 (indicated by the Braineac database to be an eQTL
for 13 proximal genes) was shown to interact only with Fez1 in mNSCs. eQTL associations can be mediated through cis or trans effects, and my research on the locus homologous to rs548181 in mNSCs suggests cis effects for Fez1 regulation. Crucially, FEZ1 has been identified as a schizophrenia-associated gene in humans for independent reasons\(^{207}\). Given that rs548181 is an intronic GWAS SNP it is important to characterise its gene targets and potential regulatory effects. Knowledge of its direct interaction with Fez1 is of utility in establishing a functional role for this schizophrenia-associated variant. This is important as previous family-based studies have been unable to put forward any links between rs548181 and downstream mechanisms\(^ {206}\). It is also important to highlight, however, that further research in human cells would be required to confirm these findings.

In addition to uncovering target pathways for disease-associated loci I have also discovered potential new enhancers and shown a mosaic of cell type-specific interactions at high resolution, in some cases on related genes in the same donors (as is the case of CSF2 and CSF2RB), providing a resource for the immunology community to explore potential mechanisms of GWAS SNP action. The discovery of potential new enhancers is an advantage resulting from the “one vs all” approach of 4C-seq, which has allowed me to evaluate potential “other” interactions not involving known promoters or enhancers to investigate potentially novel mechanisms underlying disease aetiology.
Last, but not least, I have generated what are, to my knowledge, the first 4C-seq maps for NK cells ever made, validating the use of 4C-seq in one of the most important immune cell types.

6.3.3 Outlook on 4C-seq analysis

In line with previous research that has established *Mus musculus* as one of the main model organisms in mammalian biology\textsuperscript{194}, I applied 4C-seq in mouse neural stem cells as a test system to study regions homologous to human GWAS loci. Resulting 4C-seq analyses of these mouse regions suggest potential targets for further analysis. Despite positive findings in mouse NSCs, there is still an open question as to whether these findings hold for equivalent human neural progenitors. In order for these findings to have further impact on our understanding of disease mechanism it would be necessary to perform one of two actions: either to demonstrate the same 4C-seq interaction in human cells comparing healthy and diseased subjects, or to perform CRISPR/Cas9 genome editing of the disease-associated variant in human cells, followed by elucidating the influence of CRISPR/Cas9-related genome editing on gene expression. A subsequent step would be to characterise the link between gene expression and phenotype. Similarly, CRISPR/Cas9 genome editing in sorted blood cell types will serve to clarify the phenotypic effect of immune disease-associated variants, in addition to aiding the identification of druggable targets. The immune cell type-specific interactomes I have generated will aid the prioritisation of candidate cell types for genome editing efforts on selected loci. In line with previous...
studies on variants in the FTO locus\textsuperscript{130,135} and the SCN10A locus\textsuperscript{200}, I have generated 3D genomic data for a range of disease-associated loci, an important milestone on the way to uncovering disease-relevant mechanisms for selected variants.

Several additional caveats must be highlighted. For instance, disease-relevant interactions may not be present for the cell type studied. Cell type selection is critical when studying enhancer-promoter contacts and it is possible that more advanced stages of neural differentiation will show interactions not present in mNSCs for the neural disease-associated regions I have analysed. It is also possible that other immune cell types not studied here may reveal immune disease-relevant interactions for selected loci. I have, however, taken steps to optimise target region selection in view of the cell types available. When selecting candidate loci (see Methods) I have used cell type-specific data for a range of epigenomic datasets. I have fine-tuned this analysis to the level of distinguishing related immune cell types (e.g. CD4+ and CD8+ T cell epigenomic datasets were separated for analysis). I have also used tissue-level data in certain cases. For example, I have prioritised GWAS SNPs that may exert a regulatory effect in NSCs by selecting variants that are eQTL across different brain regions.

Another caveat involves analysing the correct target region within the locus studied. Given the requirement for unique 20 bp sequences within each target, due to the need for effective PCR primers in 4C, some of the pre-designed PCR primers from van de Werken et al. (2012)\textsuperscript{149} (http://compgenomics.weizmann.ac.il/tanay/?page_id=367) were proximal (>5kb)
but not exactly overlapping target positions. Despite the fact that for such small genomic distances I do not expect a significant loss of evidence for genomic contacts, the fact that some primers do not exactly overlap target positions (due to the aforementioned constraint for 4C primer design) adds a consideration when interpreting the resulting data that must be taken into account.

It is also important to note that the absence of evidence for genomic interaction does not constitute evidence for the absence of regulatory links between GWAS SNPs and eQTL-affected genes. The way eQTL associations are identified does not necessarily imply three-dimensional genomic contact between participating regions, as there are other ways a GWAS SNP may affect the expression of a particular gene (including regulation involving downstream pathways).

Given the dynamic nature of the epigenome of immune cells, it is possible that certain enhancer contacts only take place in certain physiological conditions, such as LPS-induced activation. Here I have presented analyses from healthy donors in a normal physiological state. Further approaches may apply immune activation protocols to study cells in an activated state.

In conclusion, I do not only provide detailed 4C-seq results for a list of genomic loci but also a method for ranking candidate GWAS SNPs (and EWAS DMPs) for interactome analysis. I thus provide analytical and experimental resources to face one of the main challenges in the study of disease-associated intergenic regions: the identification of target genes.
Discussion
7 References


108. Williamson, I. *et al.* Spatial genome organization: contrasting views from
chromosome conformation capture and fluorescence in situ hybridization.

109. Dekker, J., Rippe, K., Dekker, M. & Kleckner, N. Capturing chromosome

identify chromosomal interactions: a comparison of approaches. *Nat.

111. Wit, E. de & Laat, W. de. A decade of 3C technologies: insights into nuclear

112. Mifsud, B. *et al.* Mapping long-range promoter contacts in human cells with

113. Zhao, Z. *et al.* Circular chromosome conformation capture (4C) uncovers
extensive networks of epigenetically regulated intra- and interchromosomal

114. Dostie, J. *et al.* Chromosome Conformation Capture Carbon Copy (5C): A
massively parallel solution for mapping interactions between genomic

115. Belton, J.-M. *et al.* Hi-C: a comprehensive technique to capture the

116. Horike, S., Cai, S., Miyano, M., Cheng, J.-F. & Kohwi-Shigematsu, T. Loss of
silent-chromatin looping and impaired imprinting of DLX5 in Rett

117. Fullwood, M. J. *et al.* An oestrogen-receptor-α-bound human chromatin


doi:10.1080/15592294.2014.1003744


236. Homuth, G. *et al.* Extensive alterations of the whole-blood transcriptome are associated with body mass index: results of an mRNA profiling study


References


6 Appendices
Figure S1: Gcdh locus homology. Human gene positions are shown for the Gcdh locus in mouse (supporting figure for mNSC 4C-seq figure 5.4). Gene positions for other species are also included.
Figure S2: Ppm1m locus homology. Human gene positions are shown for the Ppm1m locus in mouse (supporting figure for mNSC 4C-seq figure 5.6). Gene positions for other species are also included.

### mNSC 4C-seq primers

<table>
<thead>
<tr>
<th>Csp6l side primers</th>
<th>Sequence (Illumina adaptor-4C PCR primer).</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr5:3869961</td>
<td>CAAGCAGAAGACGGCATA CGAGATCGGTGCTCGGCATTCCTGCTGAACCG</td>
</tr>
<tr>
<td>6-38699311_Cs p6l</td>
<td>CTCTTCCGATCT-GTCATTCTTATTTGCAGGC</td>
</tr>
<tr>
<td>chr8:2425350</td>
<td>CAAGCAGAAGACGGCATA CGAGATCGGTGCTCGGCATTCCTGCTGAACCG</td>
</tr>
<tr>
<td>7-24253114_Cs p6l</td>
<td>CTCTTCCGATCT-TAACCCAGTTAGGTGCTGTC</td>
</tr>
<tr>
<td>chr8:8741133</td>
<td>CAAGCAGAAGACGGCATA CGAGATCGGTGCTCGGCATTCCTGCTGAACCG</td>
</tr>
<tr>
<td>4-87410950_Cs p6l</td>
<td>CTCTTCCGATCT-TTGTAAATTCTGTGCTTCTGC</td>
</tr>
<tr>
<td>chr9:3657673</td>
<td>CAAGCAGAAGACGGCATA CGAGATCGGTGCTCGGCATTCCTGCTGAACCG</td>
</tr>
<tr>
<td>3-36576228_Cs p6l</td>
<td>CTCTTCCGATCT-AAAGAGAAAGGTATTAGTGTAGG</td>
</tr>
<tr>
<td>chr9:1061012</td>
<td>CAAGCAGAAGACGGCATA CGAGATCGGTGCTCGGCATTCCTGCTGAACCG</td>
</tr>
<tr>
<td>43-106101777_Csp6l</td>
<td>CTCTTCCGATCT-TTGTAAATTCTGTGCTTCTGC</td>
</tr>
<tr>
<td>chr17:601423</td>
<td>CAAGCAGAAGACGGCATA CGAGATCGGTGCTCGGCATTCCTGCTGAACCG</td>
</tr>
<tr>
<td>5-6013909_Csp6l</td>
<td>CTCTTCCGATCT-CATCTCTGTAGGCCATAAC</td>
</tr>
</tbody>
</table>

### DpnII side primers

<table>
<thead>
<tr>
<th>Sequence (Illumina adaptor-4C PCR primer).</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr5:3869961 6-38699311_Dp nll</td>
</tr>
<tr>
<td>(chr17:601423 5-6013909_Dp nll</td>
</tr>
<tr>
<td>chr8:2425350 7-24253114_Dp nll</td>
</tr>
<tr>
<td>chr8:8741133</td>
</tr>
</tbody>
</table>
Table S1: sequences of PCR primers targeting mouse regions homologous to human GWAS/EWAS loci. Illumina adapters are also included in the sequence. Indexes are not necessary as the subsequent data analysis pipeline performs in silico deconvolution using the 20bp primer sequence. Pre-designed primers were selected from the 4C database by van de Werken et al. (2012)\textsuperscript{149}

<table>
<thead>
<tr>
<th>Csp6I side primer</th>
<th>Sequence (Illumina adaptor segment 1-index-Illumina adaptor segment 2-4C PCR primer).</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr1:90022562-90022945Csp_NK</td>
<td>CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCGATC-GAGTTGGAGATTGGAAG</td>
</tr>
<tr>
<td>chr17:37912171-37912498Csp_NK</td>
<td>CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCGATC- CTAGTTCCATCCACCA</td>
</tr>
<tr>
<td>chr22:37258335-37258758Csp_NK</td>
<td>CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCGATC- ATTTACTGCTGTTCCATC</td>
</tr>
<tr>
<td>chr5:131430055-131430830Csp_NKr</td>
<td>CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCGATC- TCTAGTATCTGACCTC</td>
</tr>
<tr>
<td>chr6:167534229-167534357Csp_N</td>
<td>CAAGCAGAAGACGGCATACGAGAT-AAGCTA- GTGACTGGAGTTCAGACGTGTGCTCTTCGATC-</td>
</tr>
<tr>
<td>Chromosome</td>
<td>Position Start</td>
</tr>
<tr>
<td>------------</td>
<td>---------------</td>
</tr>
<tr>
<td>chr9:123652531-123653088</td>
<td>CAAACGTTCCAAATCTTC</td>
</tr>
<tr>
<td>chr1:90022562-90022945</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GGAGTGGTAGAGAAGAAG</td>
</tr>
<tr>
<td>chr17:37912171-37912498</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr22:37258335-37258758</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GAGTGGTGAGGATTGAGAAG</td>
</tr>
<tr>
<td>chr5:131430055-131430830</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr6:167534229-167534573</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr9:123652531-123653088</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr1:90022562-90022945</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr17:37912171-37912498</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr22:37258335-37258758</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr5:131430055-131430830</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr6:167534229-167534573</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr9:123652531-123653088</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr1:90022562-90022945</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr17:37912171-37912498</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>chr22:37258335-37258758</td>
<td>CAAGCAGAAGACGGCATAAGAGAT-GTGACTGGAGTTCAAGCGGCTTCGATC-GCTAGTTCATCCACCAC</td>
</tr>
<tr>
<td>Chromosome</td>
<td>Position</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
</tr>
<tr>
<td>chr5:131430055-131430830</td>
<td>Csp_CD8</td>
</tr>
<tr>
<td>chr6:167534229-167534357</td>
<td>Csp_CD8</td>
</tr>
<tr>
<td>chr9:123652531-123653088</td>
<td>Csp_CD8</td>
</tr>
<tr>
<td>chr1:90022562-90022945</td>
<td>Csp_Bcel</td>
</tr>
<tr>
<td>chr17:37912171-37912498</td>
<td>Csp_Bcel</td>
</tr>
<tr>
<td>chr22:37258335-37258758</td>
<td>Csp_Bcel</td>
</tr>
<tr>
<td>chr5:131430055-131430830</td>
<td>Csp_Bcel</td>
</tr>
<tr>
<td>chr6:167534229-167534357</td>
<td>Csp_Bcel</td>
</tr>
<tr>
<td>chr9:123652531-123653088</td>
<td>Csp_Bcel</td>
</tr>
<tr>
<td>chr1:90022562-90022945</td>
<td>Csp_NK</td>
</tr>
<tr>
<td>chr17:37912171-37912498</td>
<td>Csp_NK</td>
</tr>
<tr>
<td>chr22:37258335-37258758</td>
<td>Csp_NK</td>
</tr>
<tr>
<td>chr5:131430055-131430830</td>
<td>Csp_NK</td>
</tr>
<tr>
<td>chr6:167534229-167534357</td>
<td>Csp_NK</td>
</tr>
<tr>
<td>chr9:123652531-123653088</td>
<td>Csp_NK</td>
</tr>
<tr>
<td>Chromosome</td>
<td>Position</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
</tr>
<tr>
<td>chr1</td>
<td>90022562-90022945</td>
</tr>
<tr>
<td>chr17</td>
<td>37912171-37912498</td>
</tr>
<tr>
<td>chr22</td>
<td>37258335-37258758</td>
</tr>
<tr>
<td>chr5</td>
<td>131430055-131430830</td>
</tr>
<tr>
<td>chr6</td>
<td>167534229-167534357</td>
</tr>
<tr>
<td>chr9</td>
<td>123652531-123653088</td>
</tr>
<tr>
<td>chr1</td>
<td>90022562-90022945</td>
</tr>
<tr>
<td>chr17</td>
<td>37912171-37912498</td>
</tr>
<tr>
<td>chr22</td>
<td>37258335-37258758</td>
</tr>
<tr>
<td>chr5</td>
<td>131430055-131430830</td>
</tr>
<tr>
<td>chr6</td>
<td>167534229-167534357</td>
</tr>
<tr>
<td>chr9</td>
<td>123652531-123653088</td>
</tr>
<tr>
<td>chr1</td>
<td>90022562-90022945</td>
</tr>
<tr>
<td>chr17</td>
<td>37912171-37912498</td>
</tr>
<tr>
<td>chr22</td>
<td>37258335-37258758</td>
</tr>
<tr>
<td>chr5</td>
<td>131430055-131430830</td>
</tr>
<tr>
<td>Chr</td>
<td>Position</td>
</tr>
<tr>
<td>--------</td>
<td>------------</td>
</tr>
<tr>
<td>D8r</td>
<td></td>
</tr>
<tr>
<td>chr6:167534229-167534357Csp_C D8r</td>
<td></td>
</tr>
<tr>
<td>chr9:123652531-123653088Csp_C D8r</td>
<td></td>
</tr>
<tr>
<td>chr1:90022562-90022945Csp_Bc elr</td>
<td></td>
</tr>
<tr>
<td>chr17:37912171-37912498Csp_Bc elr</td>
<td></td>
</tr>
<tr>
<td>chr22:37258335-37258758Csp_Bc elr</td>
<td></td>
</tr>
<tr>
<td>chr5:131430055-131430830Csp_Bc elr</td>
<td></td>
</tr>
<tr>
<td>chr6:167534229-167534357Csp_Bc elr</td>
<td></td>
</tr>
<tr>
<td>chr9:123652531-123653088Csp_Bc elr</td>
<td></td>
</tr>
</tbody>
</table>

**Table S2: BLUEPRINT 4C-seq primers.** Shown are sequences of PCR primers targeting human immune cell regions for 4C-seq analysis across CD14+, CD4+, CD8+, B cell and NK cell samples from the BLUEPRINT consortium. Illumina adapters are also included in the sequence. Due to the fact that more than one
sample had to be sequenced in a single HiSeq run, Illumina indexes had to be included in the primers for demultiplexing sample-specific data. The DpnII primer set was the same across all cell types. Indexes were included for each sample and replicate in the Csp6I primer.

<table>
<thead>
<tr>
<th>ID</th>
<th>Tissue</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSM822626</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990067</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989984</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990028</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1013189</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1079483</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1052018</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989848</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1023120</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1079482</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1052191</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989844</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1051536</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990360</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990352</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1023121</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1051549</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990061</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1051605</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989979</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM796665</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990054</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989927</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM796667</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990273</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM880157</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1051907</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM931103</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989998</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990064</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990355</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989842</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990321</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM861680</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1013198</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1051802</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1051813</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990471</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989837</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990318</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1051813</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989990</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990425</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1051691</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1052131</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990618</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM990101</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1009686</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989948</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM989853</td>
<td>Blood</td>
</tr>
<tr>
<td>GSM1075928</td>
<td>Blood</td>
</tr>
<tr>
<td>TCGA-BH-A0E1-11A-13D-A10Q-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0DG-11A-43D-A12R-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0DH-11A-31D-A10Q-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-E9-A1RD-11A-33D-A161-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-A7-A0DB-11A-33D-A093-05</td>
<td>Breast</td>
</tr>
<tr>
<td>GSM815144</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0HA-11A-31D-A12R-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0BS-11A-11D-A12R-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-A7-A13F-11A-42D-A12R-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-A7-A0D9-11A-53D-A10Q-05</td>
<td>Breast</td>
</tr>
<tr>
<td>GSM815145</td>
<td>Breast</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>TCGA-BH-A0B8-11A-41D-A03-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0BM-11A-12D-A093-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-E9-A1N5-11A-41D-A14H-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A1F0-11B-23D-A138-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A1EY-11B-21D-A13T-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0DV-11A-22D-A12R-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-E2-A1LI-11A-23D-A161-05</td>
<td>Breast</td>
</tr>
<tr>
<td>GSM815146</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A1F6-11B-94D-A13T-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0E0-11A-13D-A10Q-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0DI-11A-32D-A12R-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0BC-11A-22D-A093-05</td>
<td>Breast</td>
</tr>
<tr>
<td>GSM815149</td>
<td>Breast</td>
</tr>
<tr>
<td>GSM849975</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-E9-A1RC-11A-33D-A161-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0BT-11A-21D-A12R-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-AC-A23H-11A-12D-A161-05</td>
<td>Breast</td>
</tr>
<tr>
<td>GSM815142</td>
<td>Breast</td>
</tr>
<tr>
<td>GSM999359</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0AU-11A-11D-A12R-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA sample code</td>
<td>Tumor Type</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------</td>
</tr>
<tr>
<td>TCGA-E9-A1NE-11A-43D-A14N-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-BH-A0BZ-11A-61D-A12R-05</td>
<td>Breast</td>
</tr>
<tr>
<td>GSM815148</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-AC-A2FG-11A-22D-A17F-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-AC-A2FF-11A-13D-A17F-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-A7-A13G-11A-51D-A13T-05</td>
<td>Breast</td>
</tr>
<tr>
<td>TCGA-AA-3502-11A-01D-1407-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049405</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AA-3660-11A-01D-1721-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049400</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-A6-2682-11A-01D-1551-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049388</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049384</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AA-3509-11A-01D-1407-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049404</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049369</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049387</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AA-3713-11A-01D-1721-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AZ-6601-11A-01D-1772-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-A6-2675-11A-01D-1721-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AA-3488-11A-01D-1407-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049402</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-A6-2679-11A-01D-1551-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049381</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-G4-6295-11A-01D-1721-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-G4-6297-11A-01D-1721-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-A6-4107-11A-01D-</td>
<td>Colon</td>
</tr>
<tr>
<td>Sample ID</td>
<td>Tissue</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>GSM1049374</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AZ-6600-11A-01D-1772-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-A6-2671-11A-01D-1551-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-G4-6314-11A-01D-1721-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049398</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AA-3492-11A-01D-1407-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049377</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049399</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049376</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049379</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AZ-6599-11A-01D-1772-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049383</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-A6-2686-11A-01D-1551-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049403</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AA-3510-11A-01D-1407-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049393</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049394</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-A6-2680-11A-01D-1551-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049386</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049378</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049385</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AA-3495-11A-01D-1407-05</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049406</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049391</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049373</td>
<td>Colon</td>
</tr>
<tr>
<td>GSM1049370</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-G4-6320-11A-01D-1721-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-G4-6322-11A-01D-1721-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-AA-3712-11A-01D-1721-05</td>
<td>Colon</td>
</tr>
<tr>
<td>TCGA-B0-5092-11A-01D-1418-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-5173-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CJ-4902-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>Sample ID</td>
<td>Tissue Type</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>TCGA-B0-5710-11A-01D-1670-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4713-11A-01D-1275-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CJ-4918-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BQ-7053-11A-01D-1963-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CJ-4882-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4714-11A-01D-1275-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4823-11A-01D-1418-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CZ-5465-11A-01D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-5184-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4707-11A-01D-1275-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-4795-11A-01D-1418-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BQ-7049-11A-01D-1963-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-5098-11A-01D-1418-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-5186-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>GSM868049</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BQ-5886-11A-01D-1590-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CZ-5456-11A-01D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BQ-7059-11A-01D-1963-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4852-11A-01D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4848-11A-01D-1275-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-5121-11A-01D-1418-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>GSM999343</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4828-11A-01D-1275-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4688-11A-01D-1275-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-5195-11A-01D-</td>
<td>Kidney</td>
</tr>
<tr>
<td>Accession</td>
<td>Tissue</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
</tr>
<tr>
<td>GSM868051</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BQ-5892-11A-01D-1590-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BQ-7050-11A-01D-1963-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-5190-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4710-11A-02D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-5189-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CZ-5451-11A-01D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-5107-11A-01D-1418-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-5177-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-4993-11A-01D-1418-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CJ-4897-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CJ-4912-11A-01D-1424-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CZ-4866-11A-01D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-CZ-5460-11A-01D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4844-11A-01D-1275-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BP-4770-11A-01D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-BQ-5877-11A-01D-1590-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-5083-11A-01D-1418-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-A4-7288-11A-01D-2137-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-5400-11A-01D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-A3-3370-11A-01D-1418-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>TCGA-B0-4816-11A-02D-1500-05</td>
<td>Kidney</td>
</tr>
<tr>
<td>GSM999339</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A116-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A11D-11A-12D-</td>
<td>Liver</td>
</tr>
<tr>
<td>Sample ID</td>
<td>Tumor Type</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>TCGA-DD-A3A3-11A-11D-A22H-05</td>
<td>Liver</td>
</tr>
<tr>
<td>GSM868054</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-G3-A25W-11A-12D-A16X-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A113-11A-12D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>GSM868052</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A1EG-11A-11D-A20Z-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-BC-A10R-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-BD-A2L6-11A-21D-A20Z-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A11C-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A119-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A39W-11A-11D-A20Z-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-ES-A2HT-11A-11D-A17Z-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A115-11A-12D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A39Z-11A-21D-A20Z-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A1EF-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A11A-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-FV-A2QR-11A-11D</td>
<td>Liver</td>
</tr>
<tr>
<td>Sample ID</td>
<td>Tissue</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------</td>
</tr>
<tr>
<td>A20Z-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-BC-A10U-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A1EB-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-BC-A10X-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-FV-A23B-11A-11D-A16X-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-BC-A110-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A1EL-11A-11D-A153-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-ES-A2HS-11A-11D-A17Z-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A3A2-11A-11D-A20Z-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-G3-A25X-11A-11D-A16X-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A114-11A-12D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-BC-A10Z-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-BC-A10Q-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A118-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-BC-A10T-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A1EE-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A1EH-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A11B-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-DD-A39X-11A-11D-A20Z-05</td>
<td>Liver</td>
</tr>
<tr>
<td>GSM868055</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-BC-A112-11A-11D-A132-05</td>
<td>Liver</td>
</tr>
<tr>
<td>TCGA-44-6146-11A-01D-1756-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-18-5595-11A-01D-1633-05</td>
<td>Lung</td>
</tr>
<tr>
<td>Accession</td>
<td>Disease</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------</td>
</tr>
<tr>
<td>TCGA-22-5485-11A-01D-1633-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-50-6594-11A-01D-1756-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-39-5028-11A-01D-1440-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-44-6145-11A-01D-1756-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-44-6778-11A-01D-1856-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-22-5471-11A-01D-1633-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-39-5036-11A-01D-1440-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-50-5932-11A-01D-1756-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-43-6771-11A-01D-1818-05</td>
<td>Lung</td>
</tr>
<tr>
<td>GSM999345</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-22-5474-11A-01D-1633-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-39-5037-11A-01D-1440-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-49-4488-11A-01D-1756-05</td>
<td>Lung</td>
</tr>
<tr>
<td>GSM868074</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-39-5011-11A-01D-1440-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-22-5473-11A-11D-1633-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-22-5482-11A-01D-1633-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-33-4589-11A-01D-1440-05</td>
<td>Lung</td>
</tr>
<tr>
<td>GSM999358</td>
<td>Lung</td>
</tr>
<tr>
<td>GSM868072</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-39-5034-11A-01D-1440-05</td>
<td>Lung</td>
</tr>
<tr>
<td>GSM868008</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-33-4583-11A-01D-1440-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-50-5930-11A-01D-1756-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-33-4582-11A-01D-1440-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-50-6593-11A-01D-1756-05</td>
<td>Lung</td>
</tr>
<tr>
<td>TCGA-22-5478-11A-11D-1633-05</td>
<td>Lung</td>
</tr>
<tr>
<td></td>
<td>TCGA-73-4676-11A-01D-1756-05</td>
</tr>
<tr>
<td>-----</td>
<td>-------------------------------</td>
</tr>
<tr>
<td></td>
<td>TCGA-44-2662-11A-01D-1551-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-44-5643-11A-01D-1626-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-44-2659-11A-01D-1551-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-49-6745-11A-01D-1856-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-39-5035-11A-01D-1440-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-50-6592-11A-01D-1756-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-05-5420-11A-01D-1626-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-39-5039-11A-01D-1440-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-43-3394-11A-01D-1551-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-44-6148-11A-01D-1756-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-50-5935-11A-01D-1756-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-39-5019-11A-01D-1818-05</td>
</tr>
<tr>
<td></td>
<td>GSM999391</td>
</tr>
<tr>
<td></td>
<td>TCGA-22-5472-11A-11D-1633-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-50-5931-11A-01D-1756-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-44-2668-11A-01D-1551-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-38-4632-11A-01D-1756-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-50-5936-11A-01D-1626-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-22-5480-11A-01D-1633-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-44-2656-11A-01D-1551-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-HC-7745-11A-01D-2116-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-G9-6367-11A-01D-1787-05</td>
</tr>
<tr>
<td></td>
<td>TCGA-G9-6353-11A-02D-1963-05</td>
</tr>
<tr>
<td>TCGA-CH-5768-11A-01D-1578-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>TCGA-EJ-7785-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7328-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7784-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6384-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>GSM847574</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6373-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6348-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6332-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>GSM847571</td>
<td>Prostate</td>
</tr>
<tr>
<td>GSM847573</td>
<td>Prostate</td>
</tr>
<tr>
<td>GSM937263</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-HC-7211-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7327-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-CH-5763-11A-01D-1578-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7125-11A-01D-1963-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-CH-5762-11A-01D-1578-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-CH-5767-11B-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>GSM847569</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-CH-5766-11A-01D-1578-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6333-11A-01D-1963-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6496-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6494-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6356-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7782-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7786-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7794-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>Sample ID</td>
<td>Tumor Type</td>
</tr>
<tr>
<td>-----------</td>
<td>------------</td>
</tr>
<tr>
<td>TCGA-G9-6385-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-HC-7737-11A-02D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7781-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-HC-7819-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6363-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6351-11A-01D-1963-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>GSM937267</td>
<td>Prostate</td>
</tr>
<tr>
<td>GSM999369</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-CH-5771-11A-01D-1578-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7789-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-CH-5765-11A-01D-1578-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-G9-6362-11A-01D-1787-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7123-11A-01D-1963-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7317-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-HC-7820-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>GSM937269</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-HC-7742-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>GSM847572</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-EJ-7792-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-HC-7752-11A-01D-2116-05</td>
<td>Prostate</td>
</tr>
<tr>
<td>TCGA-ET-A25N-11A-01D-A16P-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EM-A1CW-11A-12D-A13Z-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-BJ-A28X-11A-11D-A22G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-ET-A3DP-11A-22D-A21B-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-BJ-A2N8-11A-11D-A18G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EM-A1YD-11A-11D-</td>
<td>Thyroid</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>A14Z-05</td>
<td>Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3TB-11A-11D-A22G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3ZO-11A-12D-A230-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3T6-11A-11D-A223-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-Bj-A2NA-11A-11D-A19K-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-Bj-A290-11A-11D-A17Y-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-E8-A2JQ-11A-11D-A19K-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3T2-11A-11D-A22G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-Bj-A3PU-11A-11D-A223-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EM-A1CT-11A-11D-A13Z-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3ZL-11A-11D-A230-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-ET-A3DW-11A-11D-A19K-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-H2-A3RI-11A-11D-A223-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3N3-11A-11D-A211-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3T0-11A-12D-A22G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-ET-A2MY-11B-11D-A18G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3ZP-11A-11D-A230-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3MW-11A-11D-A211-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EM-A1CS-11A-21D-A13Z-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3T8-11A-11D-A22G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-Bj-A2N7-11A-11D-A18G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-DO-A1JZ-11A-11D-A13Z-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-Bj-A28W-11A-11D-A16P-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3H1-11A-11D</td>
<td>Thyroid</td>
</tr>
</tbody>
</table>

Appendices
<table>
<thead>
<tr>
<th>Sample ID</th>
<th>Tissue</th>
</tr>
</thead>
<tbody>
<tr>
<td>A21B-05</td>
<td>Gland</td>
</tr>
<tr>
<td>TCGA-GE-A2C6-11A-11D-A16P-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3ZK-11A-11D-A230-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3N2-11A-11D-A211-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3T1-11A-11D-A22G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-BJ-A28T-11A-11D-A16P-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3H7-11A-11D-A21B-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-FY-A3TY-11A-12D-A231-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3MX-11A-11D-A21B-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EM-A1YE-11A-11D-A14Z-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3H2-11A-11D-A211-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EM-A1YC-11A-11D-A14Z-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3TA-11A-12D-A22G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-ET-A2N5-11B-11D-A18G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3ZH-11A-11D-A230-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3MY-11A-12D-A21B-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-BJ-A28R-11A-11D-A16P-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-ET-A25J-11A-01D-A16P-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-BJ-A2N9-11A-11D-A18G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3T3-11A-11D-A22G-05</td>
<td>Thyroid Gland</td>
</tr>
<tr>
<td>TCGA-EL-A3T7-11A-21D-A22G-05</td>
<td>Thyroid Gland</td>
</tr>
</tbody>
</table>

**Table S3: tDMP sample IDs.** tDMP samples were analysed to identify tissue-specific differentially methylated positions using dmpFinder. GEO/TCGA IDs are listed for all tDMP samples.
PUBLICATIONS

eFORGE: A TOOL FOR IDENTIFYING CELL TYPE-SPECIFIC SIGNAL IN EPIGENOMIC DATA, BREEZE ET AL., CELL REPORTS (2016)

Epigenome-wide association studies (EWAS) provide an alternative approach for studying human disease through consideration of non-genetic variants such as altered DNA methylation. To advance the complex interpretation of EWAS, we developed eFORGE (http://eforge.cs.ucl.ac.uk/), a new standalone and web-based tool for the analysis and interpretation of EWAS data. eFORGE determines the cell type-specific regulatory component of a set of EWAS-identified differentially methylated positions. This is achieved by detecting enrichment of overlap with DNase I hypersensitive sites across 454 samples (tissues, primary cell types, and cell lines) from the ENCODE, Roadmap Epigenomics, and BLUEPRINT projects. Application of eFORGE to 20 publicly available EWAS datasets identified disease-relevant cell types for several common diseases, a stem cell-like signature in cancer, and demonstrated the ability to detect cell-composition effects for EWAS performed on heterogeneous tissues. Our approach bridges the gap between large-scale epigenomics data and EWAS-derived target selection to yield insight into disease etiology.
Structural Basis for the Activation of IKK1/α
GENETIC AND ENVIRONMENTAL INFLUENCES INTERACT WITH AGE AND SEX IN SHAPING THE HUMAN METHYLOME, VAN DONGEN ET AL.,

NATURE COMMUNICATIONS (2016)

The methylome is subject to genetic and environmental effects. Their impact may depend on sex and age, resulting in sex- and age-related physiological variation and disease susceptibility. Here we estimate the total heritability of DNA methylation levels in whole blood and estimate the variance explained by common single nucleotide polymorphisms at 411,169 sites in 2,603 individuals from twin families, to establish a catalogue of between-individual variation in DNA methylation. Heritability estimates vary across the genome (mean = 19%) and interaction analyses reveal thousands of sites with sex-specific heritability as well as sites where the environmental variance increases with age. Integration with previously published data illustrates the impact of genome and environment across the lifespan at methylation sites associated with metabolic traits, smoking and ageing. These findings demonstrate that our catalogue holds valuable information on locations in the genome where methylation variation between people may reflect disease-relevant environmental exposures or genetic variation.
THE INTERNATIONAL HUMAN EPIGENOME CONSORTIUM: A BLUEPRINT FOR SCIENTIFIC COLLABORATION AND DISCOVERY, STUNNENBERG ET AL., CELL (2016)

The International Human Epigenome Consortium: A Blueprint for Scientific Collaboration and Discovery

Hendrik G. Stunnenberg,1,∗ The International Human Epigenome Consortium,4 and Martin Hirst2,3,∗
1Department of Molecular Biology, Faculties of Science and Medicine, Radboud University, Nijmegen, 6525AG, the Netherlands
2Department of Microbiology and Immunology, Michael Smith Laboratories, University of British Columbia, Vancouver, BC V6T 1Z4, Canada
3Canada’s Michael Smith Genome Sciences Centre, BC Cancer Agency, Vancouver, BC V5Z 4S6, Canada
4http://ihec-epigenomes.org/
∗Correspondence: h.stunnenberg@ncmris.ru.nl (H.G.S.), mhirst@bcgsc.ca (M.H.)
http://dx.doi.org/10.1016/j.cell.2016.11.007

The International Human Epigenome Consortium (IHEC) coordinates the generation of a catalog of high-resolution reference epigenomes of major primary human cell types. The studies now presented (see the Cell Press IHEC web portal at http://www.cell.com/consortium/IHEC) highlight the coordinated achievements of IHEC teams to gather and interpret comprehensive epigenomic datasets to gain insights in the epigenetic control of cell states relevant for human health and disease.
Tissue-independent and tissue-specific patterns of DNA methylation alteration in cancer

Yuting Chen, Charles E. Breeze, Shao Zhen, Stephan Beck and Andrew E. Teschendorff

Abstract

Background: There is growing evidence that DNA methylation alterations contribute to carcinogenesis. While cancer tissue exhibits widespread DNA methylation changes, the proportion of tissue-specific versus tissue-independent DNA methylation alterations in cancer is unclear. In addition, it is unknown which factors determine the patterns of aberrant DNA methylation in cancer.

Results: Using HumanMethylation450 BeadChips (450k), we here analyze genome-wide DNA methylation patterns of ten types of fetal tissue, in addition to matched normal-cancer data for corresponding tissue types, encompassing over 3000 samples. We demonstrate that the level of aberrant cancer DNA methylation in gene promoters and gene bodies is highly correlated between cancer types. We estimate that up to 60% of the DNA methylation variation in a cancer genome of a given tissue type is explained by the corresponding variation in a cancer genome of another type, implying that much of the cancer DNA methylation landscape is tissue independent. We further show that histone marks in normal cells are better predictors of aberrant cancer DNA methylation than the corresponding signals in human embryonic stem cells. We build predictors of cancer DNA methylation patterns and show that although inclusion of three histone marks (H3K4me3, H3K27me3 and H3K36me3) improves model accuracy, the bivalent marks are the most predictive. Finally, we show that chromatin accessibility of gene promoters in normal tissue dictates the promoter’s propensity to acquire aberrant DNA methylation in cancer so far as it determines its level of DNA methylation in normal tissue.

Conclusions: Our data show that a considerable fraction of the aberrant cancer DNA methylation landscape results from a mechanism that is largely tissue specific. Histone marks as specified in the normal cell of origin provide highly predictive models of aberrant cancer DNA methylation and outperform those derived from the same marks in hESCs.
Epigenetic reprogramming of fallopian tube fimbriae in BRCA mutation carriers defines early ovarian cancer evolution

Thomas E. Bartlett1,2,*, Kantaraja Chinder1,*, Jacqueline McDermott1, Charles E. Breeze1, William R. Cooke1, Allison Jones1, Daniel Reisel1, Smita T. Karegodar1, Rupali Arora4, Stephan Beck3, Usha Menon1, Louis Dubeau5 & Martin Widschwendter1

The exact timing and contribution of epigenetic reprogramming to carcinogenesis are unclear. Women harbouring BRCA1/2 mutations demonstrate a 30–40-fold increased risk of high-grade serous extra-uterine Müllerian cancers (HGSEMC), otherwise referred to as ‘ovarian carcinomas’, which frequently develop from fimbrial cells but not from the proximal portion of the fallopian tube. Here we compare the DNA methylome of the fimbrial and proximal ends of the fallopian tube in BRCA1/2 mutation carriers and non-carriers. We show that the number of CpGs displaying significant differences in methylation levels between fimbrial and proximal fallopian tube segments are threefold higher in BRCA mutation carriers than in controls, correlating with overexpression of activation-induced deaminase in their fimbrial epithelium. The differentially methylated CpGs accurately discriminate HGSEMCs from non-serous subtypes. Epigenetic reprogramming is an early pre-malignant event integral to BRCA1/2 mutation-driven carcinogenesis. Our findings may provide a basis for cancer-preventative strategies.
A comparison of reference-based algorithms for correcting cell-type heterogeneity in Epigenome-Wide Association Studies

Andrew E. Teschendorff1,2,3*, Charles E. Breeze4, Shijie C. Zheng1,5 and Stephan Beck4

Abstract

Background: Intra-sample cellular heterogeneity presents numerous challenges to the identification of biomarkers in large Epigenome-Wide Association Studies (EWAS). While a number of reference-based deconvolution algorithms have emerged, their potential remains underexplored and a comparative evaluation of these algorithms beyond tissues such as blood is still lacking.

Results: Here we present a novel framework for reference-based inference, which leverages cell-type specific DNAse Hypersensitive Site (DHs) information from the NIH Epigenomics Roadmap to construct an improved reference DNA methylation database. We show that this leads to a marginal but statistically significant improvement of cell-count estimates in whole blood as well as in mixtures involving epithelial cell-types. Using this framework we compare a widely used state-of-the-art reference-based algorithm (called constrained projection) to two non-constrained approaches including CIBERSORT and a method based on robust partial correlations. We conclude that the widely-used constrained projection technique may not always be optimal. Instead, we find that the method based on robust partial correlations is generally more robust across a range of different tissue types and for realistic noise levels. We call the combined algorithm which uses DHs data and robust partial correlations for inference, EpiDISH (Epigenetic Dissection of Intra-Sample Heterogeneity). Finally, we demonstrate the added value of EpiDISH in an EWAS of smoking.

Conclusions: Estimating cell-type fractions and subsequent inference in EWAS may benefit from the use of non-constrained reference-based cell-type deconvolution methods.
WHERE NEXT FOR THE REPRODUCIBILITY AGENDA IN COMPUTATIONAL BIOLOGY?, LEWIS ET AL., BMC SYSTEMS BIOLOGY (2016)

BMC Systems Biology

Where next for the reproducibility agenda in computational biology?

Joanna Lewis1,2*, Charles E. Breeze3*, Jane Charlesworth4, Oliver J. Maclaren5,6 and Jonathan Cooper7

*Equal contributors

Abstract

Background: The concept of reproducibility is a foundation of the scientific method. With the arrival of fast and powerful computers over the last few decades, there has been an explosion of results based on complex computational analyses and simulations. The reproducibility of these results has been addressed mainly in terms of exact replicability or numerical equivalence, ignoring the wider issue of the reproducibility of conclusions through equivalent, extended or alternative methods.

Results: We use case studies from our own research experience to illustrate how concepts of reproducibility might be applied in computational biology. Several fields have developed 'minimum information' checklists to support the full reporting of computational simulations, analyses and results, and standardised data formats and model description languages can facilitate the use of multiple systems to address the same research question. We note the importance of defining the key features of a result to be reproduced, and the expected agreement between original and subsequent results. Dynamic, updatable tools for publishing methods and results are becoming increasingly common, but sometimes come at the cost of clear communication. In general, the reproducibility of computational research is improving but would benefit from additional resources and incentives.

Conclusions: We conclude with a series of linked recommendations for improving reproducibility in computational biology through communication, policy, education and research practice. More reproducible research will lead to higher quality conclusions, deeper understanding and more valuable knowledge.
DNA METHYLOME ANALYSIS REVEALS DISTINCT EPIGENETIC PATTERNS OF ASCENDING AORTIC DISSECTION AND BICUSPID AORTIC VALVE, PAN ET AL., CARDIOVASCULAR RESEARCH (2017)

DNA methylome analysis reveals distinct epigenetic patterns of ascending aortic dissection and bicuspid aortic valve

Sun Pan\textsuperscript{1}, Hao Lai\textsuperscript{1}, Yiru Shen\textsuperscript{2}, Charles Breeze\textsuperscript{3}, Stephan Beck\textsuperscript{3}, Tao Hong\textsuperscript{1}, Chunsheng Wang\textsuperscript{1,*}, and Andrew E. Teschendorf\textsuperscript{2,4,5,*}

Aims
Epigenetics may mediate the effects of environmental risk factors on disease, including heart disease. Thus, measuring the DNA methylome offers the opportunity to identify novel disease biomarkers and novel insights into disease mechanisms. The DNA methylation landscape of ascending aortic dissection (AD) and bicuspid aortic valve (BAV) with aortic aneurysmal dilatation remain uncharacterized. The present study aimed to explore the genome-wide DNA methylation landscape underpinning these two diseases.

Methods and results
We used Illumina 450k DNA methylation beadarrays to analyse 21 ascending aorta samples, including 10 cases with AD, 5 with BAV and 6 healthy controls. We adjusted for intra-sample cellular heterogeneity, providing the first unbiased genome-wide exploration of the DNA methylation landscape underpinning these two diseases. We discovered that both diseases are characterized by loss of DNA methylation at non-CpG sites. We validate this non-CpG hypomethylation signature by pyrosequencing. In contrast to non-CpGs, AD and BAV exhibit distinct DNA methylation landscapes at CpG sites, with BAV characterized mainly by hypermethylation of EZH2 targets. In the case of AD, integrative DNA methylation gene expression analysis reveals that AD is characterized by a dedifferentiated smooth muscle cell phenotype. Our integrative analysis further reveals hypomethylation associated overexpression of RARA in AD, a pattern which is also seen in cells exposed to smoke toxins.

Conclusion
Our data supports a model in which increased cellular proliferation in AD and BAV underpins loss of methylation at non-CpG sites. Our data further supports a model, in which AD is associated with an inflammatory vascular remodeling process, possibly mediated by the epigenome and linked to environmental risk factors such as smoking.
eFORGE is the epigenetic equivalent of the FORGE tool. For details of the FORGE analysis approach see documentation in the FORGE web version at

http://www.1000genomes.org/forge-analysis

eFORGE is also available as a web tool at

http://eforge.cs.ucl.ac.uk/

1. The script itself is currently called eforge.pl written in Perl. It has the following Perl dependencies.

   use 5.012;
   use warnings;
   use DBI;
   use Sort::Naturally;
   use Cwd;
   use Storable;
   use Getopt::Long;

2. The sqlite3 db file that stores the bitstrings. This file is called eforge_1.1.db currently.

3. A stored hash containing the parameters for the background selection. Currently two files, one with the 27k and one with the 450k annotation data (mvp_bins).

   The database and the hashes are downloadable from:
   http://eforge.cs.ucl.ac.uk/?download

4. An eforge.ini file in the same directory as the script. Edit this to provide the directory in which the database and hash are stored.

5. An R 3.0 installation with the "devtools" and "rCharts" packages installed. See

   https://github.com/ramnathv/rCharts. You will need to install the latest version e.g.

   require(devtools)
   install_github('rCharts', 'ramnathv', ref = "dev")

The input data is one of several options.

a. A list of 450k probeids (DMPs)

b. BED format (for 450k probes)

The analysis requires a minimum of 5 DMPs (this is not a strict limit but
operationally is best).

To work DMPs currently have to be on either the 27k or the 450k array. The script gives warnings on DMPs not found.

It also warns for background sets that do not have the right number of probes chosen, but this is really for information only.

It takes a series of command line options as follows

- \( f \): the file to run on
- \( \text{data} \): whether to analyse ENCODE (encode) or Epigenome Roadmap (erc) data
- \( \text{label} \): a name for the files that are generated and for the plot titles
  where there is a title.
- \( \text{format} \): for the input data format. If this is location data e.g.
  bed format, the probeid is obtained from
  the sqlite3 database.

Some of these default as described in the perldoc. Minimally the command line is:

\[
\text{eforge.pl -f probeidfile -label Some_label}
\]

which will by default run on Epigenome Roadmap data

OUTPUT

\[
\text{======}
\]

there are several outputs generated

1. A pdf static chart, that would be good for download.
2. A d3 interactive chart.
3. A Datatables table.
4. R code files for generating the charts and the table.
5. There is also a tsv file of the results.

WEBSERVER

\[
\text{======}
\]

To install the web interface, please refer to the INSTALL document in the webserver folder.

\[
#!/usr/bin/env perl
\]

=head1 NAME

eforge.pl - Experimentally derived Functional element Overlap analysis of ReGions from EWAS.

=head1 SYNOPSIS
eForge::eForge, eForge::ePlot and eForge::eStats modules.

=head1 OPTIONS

=item B<--dataset TAG>

Set of functional data to look for enrichment. Either ENCODE data ('encode'), unconsolidated Roadmap Epigenome data ('erc'), consolidated Roadmap Epigenome data ('erc2'), or Blueprint data ('blueprint').

erc by default.

Use --dataset ? to get a list of available datasets on your local install.

=item B<--array TAG>

Array (FKA background) is set at default to 450k array ('450k'), the Illumina Infinium HumanMethylation450 BeadChip.

For the time being, it is sufficient for MVPs to be on the 450k array. Probes within 1kb of each other will undergo filtering.
Use --array to get a list of available backgrounds on your local install.

=item B<--label STRING>

Supply a label that you want to use for the plotting titles, and filenames.

=item B<--f FILENAME>

Supply the name of a file containing a list of MVPs. Format must be given by the -format flag. If not supplied the analysis is performed either on mvps provided as probeids (cg or ch probes) in a comma separated list through the mvps option or on a set of data from a default ewas study, namely a set of monocyte tDMPs from Jaffe AE and Irizarry RA, Genome Biol 2014. Note that at least 5 MVPs are required at a minimum by default.

=item B<--mvps probe_id,probe_id...>

Can provide the mvps as probeids in a comma separated list.

=item B<--min_mvps INT>

Specify the minimum number of MVPs to be allowed. Default is 5 now we are using binomial test.

=item B<--thresh FLOAT,FLOAT>

Alter the default binomial p value thresholds. Give a comma separate list of three e.g. 0.05,0.01 for the defaults.

=item B<--format STRING>

If f is specified, specify the file format as follow:

probeid = list of mvps as probeids each on a separate line. Optionally can add other fields after the probeid which are ignored, unless the pvalue filter is specified, in which case eForge assumes that the second field is the minus log10 pvalue.

bed = File given is a bed file of locations (chr\tbeg\tend). bed format should be 0 based and the chromosome should be given as chrN. However we will also accept chromosomes as just N (ensembl) and 1-based format where beg and end are the same*.

.tabix = File contains MVPs in tabix format.

=item B<--filter FLOAT>

Set a filter on the MVPs based on the -log10 pvalue. This works for files in the probeid' format.
Appendices

Give a value as the lower threshold and only MVPs with -log10 pvalues >= to the threshold will be analysed. Default is no filtering.

=item B<--save_stats>

Output annotation stats for the original and the random picks.

=item B<--reps INT>

The number of background matching sets to pick and analyse. Default 1000.

=item B<--proxy TAG>

Apply filter for MVPs in proximity (within 1 kb of another test MVP). With proximity filter specified, eForge will report MVPs removed due to proximity with another MVP in the list and will randomly pick one of the probes among the set of probes that are in proximity (within 1 kb of each other).

At the moment, this is a dummy flag as only one proximity filter is available for each array. It will become useful if the database and code support more than one. At the moment to turn off proximity filtering, simply specify -noproxy

=item B<--noproxy>

Turn off proximity filtering.

=item B<--depletion>

Analyse for depletion pattern instead of the default enrichment analysis. Use when dealing with datasets suspected not to overlap with DHS (or the relevant functional assay). Specifying depletion will be indicated on the label (the text "Depletion Analysis" will be added to the file label).

=item B<--noplot>

Just make the data file, don't plot.

=item B<--help|-h|-?>

Print a brief help message and exits.

=item B<--man|-m>

Print this perldoc and exit.

=back
Appendices

=head1 LICENCE AND COPYRIGHT

eforge.pl Functional analysis of EWAS MVPs

Copyright (C) [2014-2015] EMBL - European Bioinformatics Institute and University College London

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 dated June, 1991 or at your option any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is available in the source tree; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

=head1 CONTACT

Charles Breeze, C<< <c.breeze at ucl.ac.uk> >>

Javier Herrero, C<< <javier.herrero at ucl.ac.uk> >>

=head1 ACKNOWLEDGEMENTS

This software is based on the FORGE tool developed by Ian Dunham at the EMBL-EBI

Javier Herrero <javier.herrero@ucl.ac.uk>

cut

use strict;
use 5.010;
use warnings;
use DBI; #database link to sqlite database
use Sort::Naturally;
use Cwd;
use Getopt::Long; #check this module
use File::Basename;
use Config::IniFiles;
use Pod::Usage;
use Scalar::Util qw(looks_like_number);
use eForge::eStats;
use eForge::ePlot;
use eForge::eForge;
use Data::UUID;
use Statistics::Multtest qw(BY);

my $cwd = getcwd;
my $dbname = "eforge_1.2.db";

my $array; # Default value
my $array_label;
my $format = 'probeid'; # Input format
my $label = 'Unnamed'; # Label for plots
my $reps = 1000;
# set binomial p values, multiple test correction is used
my $thresh; # string for command line option
my $t_marginal = 0.05; # default marginal p-value threshold
my $t_strict = 0.01; # default strict p-value threshold

my $min_num_probes = 5; # the minimum number of probes allowed for test. Set to 5 as we have binomial p

my ($dataset, $filename, $save_probe_annotation_stats, $noplot, $help, $man, $proxy, $noproxy, $depletion, $filter, $out_dir, $probe_list, $web, $autoopen);

GetOptions {
    'dataset=s' => \$dataset,
    'save_stats|bkgrd' => \$save_probe_annotation_stats,
    'array|bkgd=s' => \$array,
    'label=s' => \$label,
    'f=s' => \$filename,
    'format=s' => \$format,
    'probes|mvps=s@' => \$probe_list,
    'min_num_probes|min_mvps=i' => \$min_num_probes,
    'noplot' => \$noplot,
    'reps=i' => \$reps,
    'thresh=s' => \$thresh,
    'proxy=s' => \$proxy,
    'noproxy' => \$noproxy,
    'depletion' => \$depletion,
    'filter=f' => \$filter,
    'out_dir=s' => \$out_dir,
    'web=s' => \$web,
    'autoopen' => \$autoopen,
    'help|h|?' => \$help,
    'man|m' => \$man,
};

pod2usage(1) if ($help);
pod2usage(-verbose => 2) if ($man);

if (!$out_dir) {
    my $ug = new Data::UUID;
    $out_dir = $ug->to_hexstring($ug->create());
}
mkdir $out_dir;
Define the thresholds to use.
if ($thresh) {
  ($t_marginal, $t_strict) = parse_pvalue_thresholds($thresh);
}

Connect to the DB

This reads the config file and sets up the $datadir variable
my $dirname = dirname(__FILE__);
my $cfg = Config::IniFiles->new(-file => "$dirname/eforge.ini");
my $datadir = $cfg->val('Files', 'datadir');

unless (-s "$datadir/$dbname") {
  die "Database $dbname not found or empty";
}
my $dsn = "dbi:SQLite:dbname=$datadir/$dbname";
my $dbh = DBI->connect($dsn, "", "") or die DBI::errstr;

Check the dataset against the info on the DB

my $all_datasets = get_all_datasets($dbh);
if (!defined($all_datasets)) {
  die "Empty database: no dataset loaded!\n";
} elsif (!defined($dataset)) {
  $dataset = $all_datasets->[0]->{tag};
  print "Using default dataset: [\$dataset ]\$all_datasets->[0]->{name}.\n"
} elsif ($dataset eq "") {
  print "Available datasets:\n - [\"join("\n - [\", map \$_->{tag}.\"]\".$->{name}} @$all_datasets).\n";
  exit();
} elsif ($_.eq $dataset) map {$_->{tag}} @$all_datasets {
  die "Dataset $dataset unknown\nAvailable datasets:\n - [\"join("\n - [\", map \$_->\{tag}.\"]\".$->\{name}} @$all_datasets).\n";
}
Appendices

### Check the array name (A.K.A. background) against DB

```perl
my $all_arrays = get_all_arrays($dbh);
if (!defined($all_arrays)) {
    die "Empty database: no background loaded!\n";
} elsif (!defined($array)) {
    $array = $all_arrays->[0]->{tag};
    print "Using default background: [array] ".所有_arrays->[0]->{name}).\n";
    $array_label = $all_arrays->[0]->{name};
} elsif ($array eq "") {
    print "Available arrays:\n - [".join("\n - ", map \$_->{tag}]." ",->{name})
@所有_arrays).\n";
    exit();
} elsif (!grep {\$_ eq $array} map \$_->{tag} @$所有_arrays) {
    die "Array $array unknown\nAvailable arrays:\n - [".join("\n - ", map \$_->{tag}]."
".->{name}) @所有_arrays).\n";
} else {
    foreach my $this_array (@所有_arrays) {
        if ($this_array->{tag} eq $array) {
            $array_label = $this_array->{name};
            last;
        }
    }
}
```

### Check the proxy_filter (A.K.A. filter) against DB

```perl
# Set proximity filter
if (defined $noproxy) {
    $proxy = undef;
} else {
    my $all_proxy_filters = get_all_proximity_filters($dbh);
    if ($all_proxy_filters->{$array}) {
        $proxy = $all_proxy_filters->{$array};
    }
}
```
Appendices

```perl
if (defined $depletion) {
    $label = "$label.depletion";
}

(my $lab = $label) =~ s/\s+/\g; # Avoid whitespaces on the label
$lab = "$lab.$array.$dataset";

# Read and process the input MVPs

my $mvps = get_input_probes($filename, $probe_list);
my $original_mvps = @($mvps);
my $num_of_input_mvps = scalar(@$mvps);

# Apply the proximity filter if requested
if (defined $proximity) {
    ($proximity_excluded, $mvps) = proximity_filter($dbh, $array, $mvps);
    while (my ($excluded_mvp, $mvp) = each %$proximity_excluded) {
        warn "$excluded_mvp excluded for $proximity proximity filter with $mvp"
    }
}

# $annotated_probes is an arrayref with probe_id, sum, bit, gene_group, cgi_group for each input probe
my $annotated_probes = get_probe_annotations_and_overlap_for_dataset($dbh, $dataset, $array, $mvps);
my $existing_probes = {map {$_->[0] => 1} @$annotated_probes};
$mvps = [keys %$existing_probes];

# Detect and remove the missing probes.
my $num_missing_probes = find_missing_probes($original_mvps, $existing_probes, $proximity_excluded);
```
# Print summary of filtering and checks:
my $msg = "For $label, $num_of_input_mvps MVPs provided, ". scalar @$mvps.
      " retained: $num_missing_probes were not found";
if (defined $proxy) {
    $msg .= ", and " . scalar(keys %$proximity_excluded) . " excluded using $proxy proximity filter";
}
warn $msg, "\n";

# Check we have enough MVPs left
my $num_of_valid_probes = scalar @$mvps;
if ($num_of_valid_probes < $min_num_probes) {
    die "Fewer than $min_num_probes MVPs. Analysis not run\n"
}
#
# get the cell list array and the hash that connects the cells and tissues
# $samples is a hash whose keys are the $cells (short name for the cell type/lines) and value is
# another hash with 'tissue', 'datatype', 'file' and 'acc' keys.
# IMPORTANT: $cells contains the list of cells in the order defined in the DB. This is critical
# to correctly assign each bit to the right sample.
my ($cells, $samples) = get_samples_from_dataset($dbh, $dataset);

# unpack the bitstrings and store the overlaps by cell.
# $overlaps is a complex hash like:
# $overlaps->{MVPS}->{probe_id}->{SUM} (total number of overlaps of this probe with features in this dataset)
# $overlaps->{MVPS}->{probe_id}->{PARAMS} (gene and CGI annotations for this probe)
# $overlaps->{CELLS}->{cell}->{COUNT} (number of input MVPs that overlap with the signal on this sample)
# $overlaps->{CELLS}->{cell}->{MVPS} (list of input MVPs that overlap with the signal on this sample)
my $overlaps = process_overlaps($annotated_probes, $cells, $dataset);

# generate stats on the background selection
if (defined $save_probe_annotation_stats) {
    save_probe_annotation_stats($overlaps, $out_dir, $lab, "test");
}

# only pick background mvps matching mvps that had bitstrings originally.
# reference to hash key 'MVPS' is due to use of eforge.pm module from eForge tool
# (in subroutines process_overlaps, etc)

# Identify the feature and cpg island relationship, and then make random picks
warn "[" . scalar(localtime()) . "] Loading the $array background...\n";
my $random_picks = get_random_matching_picks($overlaps, $array, $datadir, $reps);

###########check below lines:

# for bkgrd set need to get distribution of counts instead
# make a hash of data -> cell -> bkgrd-Set -> overlap counts
my %overlaps_per_cell; #this hash is going to store the overlaps for the random picks, per cell

# Get the bits for the background sets and process
my $total_num_probes_in_random_picks;

warn "[".scalar(localtime())."] Running the analysis with $num_of_valid_probes MVPs..\n";
my $count = 0;
foreach my $this_random_pick (@$random_picks) {
  warn "[".scalar(localtime())."] Repetition $count out of "$reps."\n" if ++$count%100 == 0);
  $annotated_probes = get_probe_annotations_and_overlap_for_dataset($dbh, $dataset, $array, $this_random_pick);
  $total_num_probes_in_random_picks += scalar @$annotated_probes;

  unless (scalar @$annotated_probes == $num_of_valid_probes) {
    warn "Random pick #$count only has ".scalar @$annotated_probes . " probes compared to $num_of_valid_probes in the input set.\n";
  }

  my $this_pick_overlaps = process_overlaps($annotated_probes, $cells, $dataset);

  # accumulate the overlap counts by cell
  foreach my $cell (keys %{$this_pick_overlaps->{CELLS}}) {
    push @{$overlaps_per_cell{$cell}}, $this_pick_overlaps->{CELLS}{$cell}->{COUNT};
  }

  if (defined $save_probe_annotation_stats) {
    save_probe_annotation_stats($this_pick_overlaps, $out_dir, $lab, $count);
  }
}

$dbh->disconnect();
warn "[".scalar(localtime())."] All repetitions done.\n";

warn "[".scalar(localtime())."] Calculating p-values..\n";
#Having got the test overlaps and the bkgd overlaps now calculate p values and output
#the table to be read into R for plotting.

if (!$web) {
  open(BACKGROUND, "| gzip -9 > $out_dir/background.tsv.gz") or die "Cannot open background.tsv";
}
my @results;
my @pvalues;

### ncmp is a function from Sort::Naturally
foreach my $cell (sort {ncmp($$samples{$a}{'tissue'},$$samples{$b}{'tissue'}) || ncmp($a,$b)} @$cells) {
    # above line sorts by the tissues alphabetically (from $samples hash values)
    # ultimately want a data frame of names(results)<-c("Zscore", "Cell", "Tissue", "File", "MVPs")
    if (!$web) {
        print BACKGROUND join("\t", @{$overlaps_per_cell{$cell}}), "\n";
    }
    my $teststat = ($overlaps->{'CELLS'}->{$cell}->{'COUNT'} > 0); #number of overlaps for the test MVPs
    my $tesstat = ($overlaps->{'CELLS'}->{$cell}->{'COUNT'} > 0); #number of overlaps
    my $total_num_overlaps_in_random_picks;
    foreach (@{$overlaps_per_cell{$cell}}) {
        $total_num_overlaps_in_random_picks += $_;
    }
    my $p = sprintf("%.6f", $total_num_overlaps_in_random_picks / $total_num_probes_in_random_picks);
    # binomial probability for $teststat or more hits out of $mvpcount mvps
    # sum the binomial for each k out of n above $teststat
    my $pbinom;
    if (defined $depletion) {
        foreach my $k (0 .. $teststat) {
            $pbinom += binomial($k, $num_of_valid_probes, $p);
        }
    } else {
        foreach my $k ($teststat .. $num_of_valid_probes) {
            $pbinom += binomial($k, $num_of_valid_probes, $p);
        }
    }
    if ($pbinom > 1) {
        $pbinom=1;
    }
    # Store the p-values in natural scale (i.e. before log transformation) for FDR correction
    push(@pvalues, $pbinom);
    $pbinom = sprintf("%.2e", $pbinom);
    # Z score calculation (note: this is here only for legacy reasons. Z-scores assume normal distribution)
    my $zscore = zscore($teststat, $overlaps_per_cell{$cell});
    my $mvp_string = "";
    $mvp_string = join("", @{$overlaps->{'CELLS'}->{$cell}->{'MVPS'}})
        if defined $overlaps->{'CELLS'}->{$cell}->{'MVPS'};
# This gives the list of overlapping MVPs for use in the tooltips. If there are a lot of
them this can be a little useless

my ($shortcell, undef) = split('\}', $cell); # undo the concatenation from earlier to deal
with identical cell names.

push(@results, [$zscore, $pbinom, $shortcell, $$samples{$cell}{'tissue'},
$$samples{$cell}{'datatype'}, $$samples{$cell}{'file'}, $mvp_string,
$$samples{$cell}{'acc'}]);

if (!$web) {
    close(BACKGROUND);
}

##
====================================================================
========
##
Correct the p-values for multiple testing using the Benjamini-Yekutieli FDR control
method
##
====================================================================
====
my $qvalues = BY(@pvalues);
$qvalues = [map {sprintf("%.2e", $_)} @qvalues];

##
====================================================================
========
##
Write the results to a tab-separated file
##
====================================================================
========
my $results_filename = "$lab.chart.tsv.gz";
open(TSV, "| gzip -9 > $out_dir/$results_filename") or die "Cannot open
$out_dir/$results_filename: "$;
"Accession", "Qvalue"), "\n";
for (my $i = 0; $i < @results; $i++) {
    print TSV join("|t", @{$results[$i]}, $qvalues->[i]), "\n";
}

##
====================================================================
========
##
Generate plots
##
====================================================================
========
WARN "[.scalar(localtime()).] Generating plots...
"

unless (defined $noplot){
# Plotting and table routines
Chart($results_filename, $lab, $out_dir, $samples, $cells, $label, $t_marginal, $t_strict, $dataset); # basic pdf plot
dChart($results_filename, $lab, $out_dir, $dataset, $label, $t_marginal, $t_strict, $web); # Dimple chart
table($results_filename, $lab, $out_dir, $web); # Datatables chart
}

WARN "[.scalar(localtime()).] Done.
"

if ($autoopen) {
  system("open $out_dir/$lab.table.html");
  system("open $out_dir/$lab.dchart.html");
  system("open $out_dir/$lab.chart.pdf");
}

=head2 parse_pvalue_thresholds

Arg[1] : string $thresholds
Returns : arrayref of marginal and strict thresholds (floats)
Example : ($t_marginal, $t_strict) = parse_pvalue_thresholds("0.05,0.01");
Description : This function returns the both marginal and strict p-value thresholds as read from
the command line option. The input string should contain both numbers separated by
a comma.
Exceptions : Dies if $thresholds is empty, does not contain numbers or are not defined between
0 and 1 and/or the marginal threshold is not larger or equal to the strict one.
=cut
sub parse_pvalue_thresholds {
    my ($thresh) = @_;    
    my ($t_marginal, $t_strict);

    if ($thresh) {
        die "Cannot read p-value thresholds from an empty string\n";
    }

    ($t_marginal, $t_strict) = split(",", $thresh);
    unless (looks_like_number($t_marginal) && looks_like_number($t_strict)) {
    die "You must specify numerical p-value thresholds in a comma separated list\n";
    }

    unless ((1 >= $t_marginal) && ($t_marginal >= $t_strict) && ($t_strict >= 0)) {
        die "The p-value thresholds should be 1 >= T.marginal >= T.strict >= 0\n";
    }
    return ($t_marginal, $t_strict);
}

=head2 get_input_probes

Arg[1] : string $filename
Arg[2] : arrayref $probe_list
Returns : arrayref of probe IDs (string)
Example : $myps = get_input_probes("input.txt", undef);
Example : $myps = get_input_probes(undef, ["cg13430807",
"cg10480329,cg06297318,cg19301114"]) ;
Example : $myps = get_input_probes(undef, undef);
Description : This function returns the list of input probe IDs. This can come from either
$filename if defined or from $probe_list otherwise. Each element in $probe_list
is a
    string which contains one or more probe IDs separated by commas (see
Examples).
    Falls back to the default data set from Jaffe and Irizarry.
    The set of probe IDs is checked to remove redundant entries.
Exceptions : Dies if the file is not found or cannot be opened for whatever reason.
=cut

sub get_input_probes {
    my ($filename, $probe_list) = @_;    
    my $probes;

    if (defined $filename) {
        my $fh;
        if ($filename =~ /\bgz$/) {
            open($fh, "gunzip -c $filename |") or die "cannot open file $filename : $!";
        } elsif ($filename =~ /\bbz2$/) {
            open($fh, "bunzip2 -c $filename |") or die "cannot open file $filename : $!";
        } else {
            open($fh, "$filename") or die "cannot open file $filename : $!";
        }
        $probes = process_file($fh, $format, $dbh, $array, $filter);
    }
elif ($probe_list and @$probe_list) {
    @$probes = split(/,/, join(',', @$probe_list));
}

else{
    # Test MVPs from Liu Y et al. Nat Biotechnol 2013 Pulmonary_function.snps.bed
    (*put EWAS bedfile here)
    # If no options are given it will run on the default set of MVPs
    warn "No probe input given, so running on default set of probes, a set of monocyte
tDMPs from Jaffe AE and Irizarry RA, Genome Biol 2014."
    @$probes = qw(cg00839584 cg02497428 cg02780988 cg03055440 cg05445326
    cg10045881 cg11051139 cg11058932 cg12091331 cg12962778 cg16303562
    cg16501235 cg18589858 cg18712919 cg18854666 cg21792432 cg22081096
    cg25059899 cg26989103 cg27443224);
}

    # Remove redundancy in the input
    my %probes_hash;
    foreach my $probe ( @$probes ) {
        $probes_hash{$probe}++;  
    }

    while ( my ($probe, $num) = each %probes_hash ) {
        if ($num > 1) {
            say "$probe is present $num times in the input. Analysing only once.
    }
}

    @$probes = keys %probes_hash;

    return($probes);
}

=head2 find_missing_probes

Arg[1] : arrayref of strings $original_probe_ids
Arg[2] : hashref $existing_probe_ids (keys are probe_ids, values are ignored)
Arg[3] : hashref $excluded_probe_ids (keys are probe_ids, values are ignored)
Returns : int $num_missing_probes
Example : my $num_missing_probes = find_missing_probes(['cg001', 'cg002',
    'cg003', 'cg004'],
    {'cg001' => 1, 'cg003 => 1}, {'cg002' => 'cg001');
Description : Detects and prints the list of missing probes if any.
Exceptions :

=cut

sub find_missing_probes {
    my ($original_probes, $existing_probes_hash, $excluded_probes_hash) = @_; 
    my $num_missing_probes = 0;

    my $missing_probes = [];
    foreach my $probe_id (@$original_probes) {
unless ($existing_probes_hash->{\$probe_id} or $excluded_probes_hash->\$probe_id) {
  push @$missing_probes, $probe_id;
}
$\nummissingprobes = scalar @$missing_probes;

if ($\nummissingprobes > 0) {
  warn "The following $\nummissingprobes MVPs have not been analysed because they were not found on the selected array\n";
  warn join("\n", @$missing_probes). "\n";
}

return $\nummissingprobes;
}

[Files]
\datadir=/Users/charles/Desktop/EBI_nfs_encode_work_cbrane/\directory_for\_eforge
\datadir=/nfs/encode/work/cbrane/\directory_for\_eforge
\datadir=/\homes/cbrane/bin/perl/eforge\_directory

e\FORGE was developed by Charles Breeze while on secondment at the European Bioinformatics Institute as part of the EpiTrain Initial Training Network.

The code, webserver and database are currently maintained by Charles Breeze and Javier Herrero.

==> pasted/eforge/eforge.pm ==>
package e\FORGE;

=head1 NAME

eforge::eforge - Interface with the DB and various other common functions for eforge

=head1 VERSION

Version 0.01

=head1 LICENCE AND COPYRIGHT

Copyright (C) [2014-2015] EMBL - European Bioinformatics Institute and University College London

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 dated June, 1991 or at your option any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is available in the source tree; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
=head1 CONTACT

Charles Breeze, C<< c.breeze at ucl.ac.uk >>

Javier Herrero, C<< javier.herrero at ucl.ac.uk >>

=head1 ACKNOWLEDGEMENTS

This software is based on the FORGE tool developed by Ian Dunham at the EMBL-EBI

cut

use 5.010;
use strict;
use warnings FATAL => 'all';
use Storable;

my $MAX_SQL_VARIABLES = 999;
our $VERSION = '0.01';
our (@ISA, @EXPORT);
use Exporter;
@ISA = qw(Exporter);
@EXPORT = qw(get_all_datasets get_all_arrays get_all_proximity_filters process_file
get_random_matching_picks process_overlaps
generate_annotations and_overlap_for_dataset get_samples_from_dataset assign
save_probe_annotation_stats proximity_filter);

=head1 SYNOPSIS

Provide functional modules for eForge

=head1 EXPORT

generate_all_datasets
generate_all_arrays
get_all_proximity_filters
process_file
get_random_matching_picks
process_overlaps
generate_annotations_and_overlap_for_dataset
generate_samples_from_dataset assign
save_probe_annotation_stats
proximity_filter

=head1 SUBROUTINES/METHODS

=head2 save_probe_annotation_stats

Arg[1] : hashref $overlaps (see get_overlaps)
Arg[2] : string $outdir
Appendices

Arg[3] : string $label
Returns :
Example : save_probe_annotation_stats($overlaps, ".", "Unnamed", "test");
Example : save_probe_annotation_stats($overlaps, ".", "Unnamed", 23);
Description : Save stats about the probe annotations on a text file. The $set_id is either
"test" (which relates to the input probe_ids, the ones to be tested for
enrichment) or the random pick number.
The file will contain for each set the whole list of gene features and CpG islands
relationships for the probes in that set.
Exceptions : Dies if file cannot be opened

=cut

sub save_probe_annotation_stats {
  my ($overlaps,
       $out_dir,
       $lab,
       $flag) = @_;

  my $fh;
  my $file = "$out_dir/$lab.overlaps.stats.txt";
  open(STATS, ">>$file") or die "cannot open $file";
  my (@gene_features, @cpg_island_relationships);
  foreach my $probeid (keys %{$overlaps->{'MVPS'}}){
    my ($this_gene_feature, $this_cpg_island_relationship) =
      split("\t", $overlaps->{'MVPS'}->{'probeid'}->{'PARAMS'});
    push @gene_features, $this_gene_feature;
    push @cpg_island_relationships, $this_cpg_island_relationship;
  }
  say STATS join("\t", $flag, "gene_features", @gene_features);
  say STATS join("\t", $flag, "cpg_island_relationships", @cpg_island_relationships);
  close(STATS);
}

=head2 process_file

Arg[1] : FILEHANDLE $input_fh
Arg[2] : string $format
Arg[4] : string $array
Returns : arrayref of $probe_ids (string)
Example : my $probe_ids = process_file($fh, "probes", $dbh, '450k', undef);
Description : Reads the list of probe IDs or locations from the file. If the file contains
locations, these will be translated into probe_ids
Exceptions : 

=cut

sub process_file {
  my ($fh, $format, $dbh, $array, $filter) = @_;
  my $probe_ids = [];

  if ($format =~ /^probe/i) {
    while (<$fh>) {

next if /^#/;
chomp;
my $probe_id;
if (defined $filter) {
    my $pval;
    ($probe_id, $pval) = split /\s+/,
    next unless $pval >= $filter;
} else {
    ($probe_id, undef) = split /\s+/,
    # remove anything that is not supposed to be there :-)
}
push @$probe_ids, $probe_id;

} elsif ($format =~ /^bed/i) {
    if (defined $filter) {
        warn "You have specified p-value filtering, but this isn't implemented for files of format $format. No filtering will happen."
    }
    my $locations = [];
    while (<$fh>) {
        next if /^track/;
        chomp;
        my ($chr, $from, $to) = split "\t",
        next if (!defined($to));
        unless ($chr =~ /^chr/){
            $chr = "chr".$chr;
        }
        push(@$locations, [$chr, $from]);
    }
    $probe_ids = fetch_all_probe_ids($dbh, $array, $locations);
} elsif ($format =~ /^tabix/i) {
    if (defined $filter) {
        warn "You have specified p-value filtering, but this isn't implemented for files of format $format. No filtering will happen."
    }
    my $locations = [];
    while (<$fh>) {
        chomp;
        my ($chr, $from, $to) = $._ =~ /(\d+)-(\d+)/;
        push(@$locations, [$chr, $from]);
    }
    $probe_ids = fetch_all_probe_ids($dbh, $array, $locations);
} return $probe_ids;

=head2 get_random_matching_picks

Arg[1] : hashref $overlaps
Arg[2] : string $array_tag
Arg[3] : string $data_dir
Appendices

Returns : arrayref of arrays of $probe_ids (string)
Example : my $random_picks = get_random_matching_picks($overlaps, "450k", ".", 1000);
Description : Get several random picks of probes matching the criteria defined in the $overlaps hash. The random picks are selected from a pre-built hash stored in the $data_dir called mvp_450k_bins (or so).
Exceptions : Cannot find the bins file for the selected array.

sub get_random_matching_picks {
    my ($overlaps, $array, $datadir, $num_random_picks) = @_;
    my $picks = [];

    # load up the stored hashes that contain the bins of mvps by feature and cpg island relationship.
    my %bins;
    my $bins_file = $datadir . "/mvp_${array}_bins";
    if (-e $bins_file) {
        %bins = %{
            retrieve($bins_file)
        };
    } else {
        die "Cannot retrieve the file $bins_file";
    }

    foreach my $probe_id (keys %{$overlaps->{MVPS}}) {
        my ($feature, $cpg_island_relationship) = split("\t", join("\t", $overlaps->{MVPS}->{$probe_id}{PARAMS}));

        #range has to be the number of probes to choose from in that hash subclass
        my $range = scalar @{$bins{$feature}{$cpg_island_relationship}};

        for (my $n = 0; $n < $num_random_picks; $n++) {
            my $picked_probe_id;
            while (1) {
                my $pick = int(rand($range));
                $picked_probe_id = ${$bins{$feature}{$cpg_island_relationship}}[$pick]; #pick the $pick'th element in the array as the chosen mvp
                last unless $picked_probe_id eq $probe_id; # must not pick the test mvp itself.
            }
            push(@{$picks->[0]}, $picked_probe_id);
        }
    }

    return $picks;
}

=head2 process_overlaps

Arg[1] : arrayref of arrays $annotated_probes ($probe_id, $sum, $bit_string, $feature, $CGI_context)
Appendices

Arg[2] : arrayref of strings $cells (shortname for the cells in the dataset)
Arg[3] : string $dataset
Returns : hashref $stats
Example : my $result = process_overlaps($rows, $cells, 'erc');
Description : Returns a reference to a complex hash with stats from the rows. These are split
into 'MVPS' and 'CELLS'. The former contains 'SUM' and 'PARAMS' for each probe ID
while the latter contains 'COUNT' and 'MVPS' for each cell.
Exceptions : Dies if the number of cells does not match the length of the bit string.

=cut

sub process_overlaps {
  my ($rows, $cells, $data) = @_;  
  my $overlaps;
  my @overlapping_probes_per_cell;
  my @indexes = 0..(@$cells-1);
  foreach my $row (@{$rows}){
    my ($probeid, $sum, $bit_string, $feature, $cpg_island_relationship) = @$row;
    $overlaps->{MVPS}->{$probeid} = $sum;
    $overlaps->{MVPS}->{$probeid}={PARAMS} = join("t", $feature,
$cpg_island_relationship);
    die "For $data, found ", scalar(@$cells), " cells for ", length($bit_string), " bits\n" if (scalar(@$cells) ne length($bit_string));
    foreach my $index (@indexes) {
      ## $bit_string is a string made of 0s and 1s. If it is a 1 for this position, count and push
      if (substr($bit_string, $index, 1) {  
        push @{$overlapping_probes_per_cell[$index]}, $probeid;
      }
    }
    my $index = 0;
    foreach my $cell (@$cells){
      if ($overlapping_probes_per_cell[$index] and
@$overlapping_probes_per_cell[$index]) {
        $overlaps->{CELLS}->{$cell}->{'COUNT'} = scalar(@$overlapping_probes_per_cell[$index]);
        $overlaps->{CELLS}->{$cell}->{'MVPS'} = $overlapping_probes_per_cell[$index];
      }
      $index++;
    }
    return $overlaps;
  }
}

=head2 get_all_datasets

Arg[1] : DB-handle $dbh
Arg[2] : (optional) string $species_name
Returns : arrayref of hashes (tag/name)
Example : my $all_datasets = get_all_dataset($dbh);
Appendices

Description : Returns the list of all datasets in the DB. It returns an arrayref of hashes. Each hash has two keys: 'tag' (string ID of the dataset) and 'name' (full name). You can limit the dataset to the ones available for a given species.

Exceptions :

=cut

sub get_all_datasets {
    my ($dbh, $species_name) = @_;  
    my $datasets;

    my $sql = "SELECT dataset_tag, dataset_name FROM dataset ORDER BY dataset_id DESC";
    my @bind_params = ();
    if ($species_name) {
        $sql .= " WHERE species_name = ?";
        push(@bind_params, $species_name);
    }
    my ($tag, $name);
    my $sth = $dbh->prepare($sql);
    $sth->execute(@bind_params);
    $sth->bind_columns($tag, $name);
    while ($sth->fetch) {  
        push(@$datasets, {tag => $tag, name => $name});
    }

    return($datasets);
}

=head2 get_all_arrays

Arg[1]  : DB-handle $dbh
Arg[2]  : (optional) string $species_name
Returns : arrayref of hashes (tag/name)
Example  : my $all_arrays = get_all_arrays($dbh);
Description : Returns the list of all arrays in the DB. It returns an arrayref of hashes. Each hash has two keys: 'tag' (string ID of the array) and 'name' (full name). You can limit the arrays to the ones available for a given species.

Exceptions :

=cut

sub get_all_arrays {
    my ($dbh, $species_name) = @_;  
    my $arrays;

    my $sql = "SELECT array_tag, array_name FROM array order by array_id DESC";
    my @bind_params = ();
    if ($species_name) {
        $sql .= " WHERE species_name = ?";
        push(@bind_params, $species_name);
    }
    my ($tag, $name);
    my $sth = $dbh->prepare($sql);
    $sth->execute(@bind_params);
    $sth->bind_columns($tag, $name);
    while ($sth->fetch) {  
        push(@$arrays, {tag => $tag, name => $name});
    }

    return($arrays);
}
Appendices

my ($tag, $name);
my $sth = $dbh->prepare($sql);
$sth->execute();
$sth->bind_columns($tag, $name);
while ($sth->fetch) {
    push(@$arrays, {tag => $tag, name => $name});
}

return($arrays);

=head2 get_all_proximity_filters

Arg[1] : DB-handle $dbh
Arg[2] : (optional) string $array
Returns : hashref of proxy-filters (string)
Example : my $proximity_filters = get_all_proximity_filters($dbh, '450k');
Description : Returns the list of proxy filters available in the DB. It returns a hashref whose
    keys are the bkgd names (i.e. array tags) and values are the name of the proxy
    filter. At the moment the schema supports just one proxy filter per array.
    If you provide a $array, the result will be limited to that array. Note that you
    will still get a hashref with exactly the same structure. Only the content will
    be limited.
Exceptions :

=cut

sub get_all_proximity_filters {
    my ($dbh, $array_tag) = @_;  
    my $proximity_filters;
    my $sql = "SELECT array_tag, description FROM proxy_filter_info JOIN array USING (array_id)";
    my @bind_params = ();
    if ($array_tag) {
        $sql .= " WHERE array_tag = '?";
        push(@bind_params, $array_tag);
    }
    my ($this_array_tag, $description);
    my $sth = $dbh->prepare($sql);
    $sth->execute();
    $sth->bind_columns($this_array_tag, $description);
    while ($sth->fetch) {
        $proximity_filters->{$this_array_tag} = $description;
    }
    return($proximity_filters);
}

=head2 get_probe_annotations_and_overlap_for_dataset

285
Appendices

Arg[1] : DB-handle $dbh
Arg[3] : string $array
Returns : arrayref of arrays containing the probe_id, sum, bitstring, gene-group and cgi-group
Example : my $annotated_probes =
get_probe_annotations_and_overlap_for_dataset($dbh, 'erc', '450k', $probe_list);
Description : Fetches the gene and CGI related annotation for the set of probes as well as the
overlaps with the features defined in the selected dataset
Exceptions :

=cut

sub get_probe_annotations_and_overlap_for_dataset {  
  my ($dbh, $dataset_tag, $array_tag, $probe_ids) = @_;  
  my $results = [];
  for (my $loop = 0; $loop * $MAX_SQL_VARIABLES < @$probe_ids; $loop++) {  
    my $start = $loop * $MAX_SQL_VARIABLES;
    my $end = ($loop + 1) * $MAX_SQL_VARIABLES - 1;
    $end = @$probe_ids - 1 if ($end >= @$probe_ids);
    my $sql = "SELECT probe_id, sum, bit, gene_group, cgi_group
  FROM probe_annotation
  JOIN probe_bitstring USING (array_id, probe_id)
  JOIN dataset USING (dataset_id)
  JOIN array ON (array.array_id = probe_annotation.array_id)
  WHERE array_tag = ? and dataset_tag = ?
  AND probe_id IN (?"." ,?" x ($end - $start)).")";  
    my $sth = $dbh->prepare_cached($sql);
    $sth->execute($array_tag, $dataset_tag, @$probe_ids[$start..$end]);
    while (my $row = $sth->fetchrow_arrayref()) {
      push @$results, [@$row];
    }
    $sth->finish();
  }
  return $results;
}

=head2 fetch_all_probe_ids

Arg[1] : DB-handle $dbh
Arg[2] : string $array
Returns : arrayref of probe IDs (string)
Description : Given a list of chromosome names and location pairs, the method fetches
the name
of the probe ID for that location. To do this successfully, this method requires

to know the background (i.e. array) you want to translate the locations to.

Exceptions :

=cut

sub fetch_all_probe_ids {
    my ($dbh, $array, $locations) = @_;

    my $sth = $dbh->prepare("SELECT probe_id
    FROM probe_mapping
    WHERE chr_name = ? AND position = ?");
    my $probe_ids = [];

    foreach my $this_loc (@$locations) {
        my ($chr, $pos) = @$this_loc;
        $sth->execute($chr, $pos);
        my $result = $sth->fetchall_arrayref();
        my $probe_id;
        foreach my $row (@{$result}) {
            push(@$probe_ids, $$row[0]);
        }
    }
    $sth->finish();
    return $probe_ids;
}

=head2 proximity_filter

Filter MVPs from the MVP list if they are within 1 kb of each other. The rationale is that
the first MVP to be identified in a block is chosen, and others are removed.

=cut

sub proximity_filter {
    my ($dbh, $array_tag, $probe_ids, $filter) = @_;
    my %prox_excluded_probes; # a hash to store MVPs found in proximity (1 kb) with an
MVP in the list
    my %filtered_probes; # The list of MVPs filtered (i.e. after filtering, the ones to keep)
    my %missing_probes;

    # Get the full list of probes as a hash (also removes redundancy)
    my %probe_id_hash;
    foreach my $probe_id (@$probe_ids){
        $probe_id_hash{$probe_id} = 1;
    }
    $probe_ids = [keys %probe_id_hash];

    for (my $loop = 0; $loop * $MAX_SQL_VARIABLES < @probe_ids; $loop++) {
        my $start = $loop * $MAX_SQL_VARIABLES;
        my $end = ($loop + 1) * $MAX_SQL_VARIABLES - 1;
        $end = @probe_ids - 1 if ($end >= @probe_ids);
my $sql = "SELECT probe_id, proxy_probes FROM proxy_filter JOIN array USING (array_id) WHERE \\
  " array_tag = ? AND probe_id IN (?" . ($end - $start)) . ");"
my $sth = $dbh->prepare($sql); # get the blocks form the ld table
$sth->execute($array_tag, @probe_ids[$start..$end]);
my $result = $sth->fetchAll_arrayref();
$sth->finish();

foreach my $row (@{$result}){
  my ($probe_id, $probe_id_list) = @$row;
  # if the probe is in the proximity filtered set already ignore it
  next if exists $prox_excluded_probes{$probe_id};
  # if this is the first time it is seen, add it to the filtered mvps, and remove anything
  # in proximity with it
  $filtered_probes{$probe_id} = 1;
  next if $probe_id_list =~ /NONE/; # nothing is in proximity
  my @other_probe_ids = split (/\|/, $probe_id_list);
  foreach my $other_probe_id (@other_probe_ids) {
    if (exists $probe_id_hash{$other_probe_id}) {
      $prox_excluded_probes{$other_probe_id} = $probe_id; #Add to the excluded
    }
  }
}

#note that if an MVP doesn’t exist in the proximity file it will be rejected regardless, may need to add these back
return (%prox_excluded_probes, [keys %filtered_probes]);
}

=head2 get_samples_from_dataset

Read the correct cell list based on data (erc, erc2, blueprint or encode). Also gets the tissue names for the cells.

=cut

sub get_samples_from_dataset {
  my ($dbh, $dataset_tag) = @_; 
  my ($cells, $samples);

  my $sth = $dbh->prepare("SELECT shortcell, tissue, datatype, file, acc FROM dataset \\
JOIN sample USING (dataset_id) WHERE dataset_tag = ? ORDER BY sample_order");
  $sth->execute($dataset_tag);
  my ($shortcell, $tissue, $datatype, $file, $acc); 
  $sth->bind_columns($shortcell, $tissue, $datatype, $file, $acc);
  while ($sth->fetch) {
    $shortcell = "$shortcell|$file"; # Sometimes the same cell is used twice, with a
different file so need to record separately (e.g. WI-38).
    push @$cells, $shortcell;
    $$samples{$shortcell}{'tissue'} = $tissue; # this is the hash that is used to connect
cells and tissues and ultimately provide the sorting order
$$\text{samples}\{\text{shortcell}\}\{\text{datatype}\} = \text{datatype};
$$\text{samples}\{\text{shortcell}\}\{\text{file}\} = \text{file};
$$\text{samples}\{\text{shortcell}\}\{\text{acc}\} = \text{acc};
}

# use Data::Dumper;
# print Dumper %$tissues;

return ($cells, $samples); # return
}

1;

==&gt; pasted/eForge/ePlot.pm &lt;=
package eForge::ePlot;

=head1 NAME
eForge::ePlot - Plotting utilities for eForge

=head1 VERSION
Version 0.01

=head1 LICENCE AND COPYRIGHT

Copyright (C) [2014-2015] EMBL - European Bioinformatics Institute and University College London

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 dated June, 1991 or at your option any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is available in the source tree; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

=head1 CONTACT
Charles Breeze, C&lt;&lt; c.breeze at ucl.ac.uk &gt;&gt;

Javier Herrero, C&lt;&lt; javier.herrero at ucl.ac.uk &gt;&gt;

=head1 ACKNOWLEDGEMENTS
This software is based on the FORGE tool developed by Ian Dunham at the EMBL-EBI

=cut
use 5.010;
use strict;
use warnings FATAL => 'all';
use Sort::Naturally;

our $VERSION = '0.01';

our (@ISA, @EXPORT, @EXPORT_OK);
use Exporter;
@ISA = qw(Exporter);
@EXPORT = qw(Chart dChart table); # Symbols to export by default

=head1 SYNOPSIS

Provise plotting utilities for different plots to eForge

=head1 EXPORT

Chart
dChart

table

=head1 SUBROUTINES/METHODS

=head2 Chart

This is the original code using standard R plot to generate a static pdf.

=cut

sub Chart{
    print "Making static chart.\n";
    my ($filename, $lab, $resultsdir, $tissues, $cells, $label, $t_marginal, $t_strict, $data) = @_; 
    my $Rdir = $resultsdir;
    my $chart = "$lab.chart.pdf";
    my $rfile = "$Rdir/$lab.chart.R";
    # set some colors
    my ($sig, $msig, $ns, $abline, $tline) = qw(red palevioletred1 steelblue3 lightpink1 brown); # alternate $msig = pink2

    open my $rfh, "">", $rfile;
    # results $Class<-cut(results$Pvalue, breaks =c(min(results$Pvalue), $t_marginal,
    $t_strict, max(results$Pvalue)), labels=FALSE, include.lowest=TRUE) # 99 and 95% CIs 1, 2, 3
    $t_marginal = sprintf("%.2f", $t_marginal);
    $t_strict = sprintf("%.2f", $t_strict);
    print $rfh 'setwd("$Rdir")'
    results<-read.table("$filename", header=TRUE,sep='\t')

    # Class splits the data into non-significant, marginally significant and significant according to $t_marginal and $t_strict (in -log10 scale)
}
results$Class <- cut(results$Pvalue, breaks = c(0, $t_strict, $t_marginal, 1)/length(unique(results[, 'Tissue'])), labels=FALSE, include.lowest=TRUE)
#	Class splits the data into non-significant, marginally significant and significant according to q-value (B-Y FDR adjusted)
results$Class2 <- cut(results$Qvalue, breaks = c(0, $t_strict, $t_marginal, 1), labels=FALSE, include.lowest=TRUE)
#	Re-order the entries according to tissue first and then cell type/line

tissue.cell.order <- unique(results[, c('Tissue', 'Cell')])
tissue.cell.order <- tissue.cell.order[order(tissue.cell.order[,1], tissue.cell.order[,2]), ]
#	Collapse into a single string (to support same cell type in different tissues)
tissue.cell.order2 <- apply(tissue.cell.order, 1, paste, collapse = ' -- ')
results$TissueCell <- apply(results[, c('Tissue', 'Cell')], 1, paste, collapse = ' -- ')
results$TissueCell <- factor(results$TissueCell, levels=tissue.cell.order2)
# Plot an empty chart first
pdf('chart', width=22.4, height=8)

ymax = max(-log10(results$Pvalue), na.rm=TRUE)*1.1
ymin = -0.1

par(mar=c(15.5,4.3,1)+0.1)

plot(NA,ylab='', xlab='', main='MVPs in DNase1 sites (probably TF sites) in cell lines for $data $label',
ylim=c(ymin,ymax), las=2, pch=19, col = results$Class2, xaxt='n',
xlim=c(0,length(levels(results$TissueCell))), cex.main=2)
#	Add horizontal guide lines for the Y-axis
abline(h=par('yaxp')[1]:par('yaxp')[2],lty=1, lwd=0.1, col='#e0e0e0')
#	Add vertical lines and labels to separate the tissues
tissues <- c(0, cumsum(summary(tissue.cell.order[, 'Tissue'])))

abline(v=tissues[2:(length(tissues)-1)]+0.5, lty=6, col='$tline')
text((tissues[1:(length(tissues)-1)] + tissues[2:length(tissues)]) / 2 + 0.5, ymax, names(tissues[2:length(tissues)]), col='$tline', adj=1, srt=90, cex=1.4)
#	Add points (internal color first)

palette(c('$sig', '$msig', 'white'))
points(results$TissueCell, -log10(results$Pvalue), pch=19, col = results$Class2, xaxt='n')
#	Add contour to the points

palette(c('black', '$msig', '$ns'))
points(results$TissueCell, -log10(results$Pvalue), pch=1, col = results$Class2, xaxt='n')
#	Add X-axis (use cell name only and not TissueCell)
axis(1, seq(1,length(tissue.cell.order[,2])), labels=tissue.cell.order[,2], las=2, cex.axis=0.9)
mtext(1, text='Cell', line=14, cex=1.4)
mtext(2, text='-log10 binomial p-value', line=2, cex=1.4)
#	Add legend (internal color first)
palette(c('white', '$msig', '$sig'))
Appendices

legend('topleft', pch=19, legend=c('q < $t_{strict}$', 'q < $t_{marginal}$', 'non-sig'), col = 3:1, cex=0.8, inset=c(0.001, 0.005), box.col='white', title='FDR q-value', text.col='white', bg='white')

# Add contour to the points in the legend
palette(c('ns', 'msig', 'black'))
legend('topleft', pch=1, legend=c('q < $t_{strict}$', 'q < $t_{marginal}$', 'non-sig'), col = 3:1, cex=0.8, inset=c(0.001, 0.005), box.col='darkgrey', title='FDR q-value')

palette('default')
dev.off()

#run the R code
system("R", "--no-save", "--quiet", "--slave", "--file=$rfile");

=head2 dChart

Make dimple interactive chart.

=cut

sub dChart{
  my ($filename, $lab, $resultsdir, $data, $label, $t_marginal, $t_strict, $web) = @_;

  print "Making dChart:\n";
  my $chart = "$lab.dchart.html"
  my $Rdir = $resultsdir;
  my $rfile = "$Rdir/$lab.dChart.R"
  open my $rcfh, ">", $rfile;
  print $rcfh "setwd("$Rdir")"
  results<-read.table("$filename", header = TRUE, sep="\\t")

  # Class splits the data into non-significant, marginally significant and significant
  # according to $t_marginal and $t_strict (in -log10 scale)
  results<$Class <- cut(results$Pvalue, breaks =c(0, $t_strict, $t_marginal, 1)/length(unique(results[, 'Tissue'])), labels=FALSE, include.lowest=TRUE)

  # Class splits the data into non-significant, marginally significant and significant
  # according to q-value (B-Y FDR adjusted)
  results<$Class2 <- cut(results$Qvalue, breaks =c(0, $t_strict, $t_marginal, 1), labels=FALSE, include.lowest=TRUE)
  color.axis.palette = c();
  if (length(which(results<$Class2 == 1)) > 0 ) {
    color.axis.palette = c('red');
  }
  if (length(which(results<$Class2 == 2)) > 0 ) {
    color.axis.palette = c(color.axis.palette, '#FF82ab');
  }
  color.axis.palette = c(color.axis.palette, 'lightblue');
  if (length(color.axis.palette) < 2) {

  }
}
Appendices

color.axis.palette = c(color.axis.palette, 'lightblue'); # Add it twice to force the color if only non-significant values
}

results$log10pvalue <- -log10(results$pvalue)

# Re-order the entries according to tissue first and then cell type/service

tissue.cell.order <- unique(results[, c('Tissue', 'Cell')])
tissue.cell.order <- tissue.cell.order[order(tissue.cell.order[,1], tissue.cell.order[,2]),]

# Collapse into a single string (to support same cell type in different tissues)
tissue.cell.order2 <- apply(tissue.cell.order, 1, paste, collapse = ' -- ')
results$TissueCell <- apply(results[, c('Tissue', 'Cell')], 1, paste, collapse = ' -- ')
results$TissueCell <- factor(results$TissueCell, levels=tissue.cell.order2)

# Count number of cell types for each tissue (to be able to draw the vertical separation lines afterwards)
tissues <- c(0, cumsum(summary(tissue.cell.order[, 'Tissue'])))

require(rCharts)
dplot.height=900
dplot.width=2000
bounds.x=60
bounds.y=50
bounds.height=dplot.height - 300
bounds.width=dplot.width - bounds.x - 20

# Create a dimple plot, showing p-value vs cell, split data by tissue, cell, probe, etc to see individual points instead of aggregate avg
d1 <- dPlot(
y = 'log10pvalue',
x = c('TissueCell'),
groups = c('TissueCell', 'Accession', 'Pvalue', 'Qvalue', 'Datatype', 'Probe'),
data = results,
type = 'bubble',
width = dplot.width,
height = dplot.height,
bounds = list(x=bounds.x, y=bounds.y, height=bounds.height, width=bounds.width),
id = 'chart.$lab'
)

# Force the order on the X-axis
d1$xAxis( type = 'addCategoryAxis', grouporderRule = 'Cell', orderRule = tissue.cell.order[,2])
d1$xAxis( type = 'addCategoryAxis', grouporderRule = 'TissueCell', orderRule = as.factor(tissue.cell.order2))

d1$yAxis( type = 'addMeasureAxis' )

# Color points according to the q-value
d1$colorAxis(
type = 'addColorAxis',
colorSeries = 'Class2',
palette = color.axis.palette)
# Builds a JS string to add labels for tissues
labels.string = paste(paste0(" // Adds labels for tissues
myChart.svg.insert('text', 'g')
  .attr('x', 0)
  .attr('y', 0)
  .attr('font-size', 20)
  .attr('font-family', 'Arial')
  .style('fill', 'brown')
  .attr('transform', 'translate(', bounds.x - 5 + bounds.width * (tissues[1:length(tissues)-1] + tissues[2:length(tissues)]) / (2 * max(tissues))), ",60) rotate(90))'
  .attr('text-anchor', 'top')
  .text("", names(tissues[2:length(tissues)]), ")
\")
\"), collapse='')

# Builds a JS string to add vertical lines to separate tissues
lines.string = paste(paste0(" // Adds vertical lines between tissues
myChart.svg.append('line')
  .attr('x1', ", (bounds.x + bounds.width * tissues[2:length(tissues)-1])/max(tissues)), ")
  .attr('y1', 50)
  .attr('x2', ", (bounds.x + bounds.width * tissues[2:length(tissues)-1])/max(tissues)), ")
  .attr('y2', ", (50 + bounds.height), ")
  .style('stroke', 'brown')
  .style('stroke-dasharray', '10,3,3,3')
\")
\"), collapse='')

# Adds some JS to be run after building the plot to get the image we want
d1\$setTemplate(afterScript = paste0("<script>
myChart.draw()
// Substitutes TissueCell labels in X-axis by Cell labels
myChart.axes[1].shapes
  .selectAll('text')
    .text(function (d)
      var i;
      for (i = 0; i < data.length; i += 1) {
        if (data[i].TissueCell == d) {
          return data[i].Cell;
        }
      }
    )
  );

  // Adds title for X-axis
myChart.axes[1].titleShape
  .style('font-size', 20)

  // Adds title for Y-axis
myChart.axes[2].titleShape
Appendices

// Adds main title
myChart.svg.append('text')
  .attr('x', '
    ', (dplot.width / 2), '
  ')
  .attr('y', '
    ', (bounds.y / 2), '
  ')
  .attr('font-size', 24)
  .attr('font-family', 'Arial')
  .attr('font-weight', 'bold')
  .style('fill', 'black')
  .attr('text-anchor', 'middle')
  .text('MVPs in DNase1 sites (probably TF sites) in cell lines for $data$ label')

// Adds vertical line at the far right of the plot
myChart.svg.append('line')
  .attr('x1', '
    ', (bounds.x + bounds.width), '
  ')
  .attr('y1', '
    ', bounds.y, '
  ')
  .attr('x2', '
    ', (bounds.x + bounds.width), '
  ')
  .attr('y2', '
    ', (bounds.y + bounds.height), '
  ')
  .style('stroke', 'rgb(0,0,0)')
</script>
"
")

$d1\$save('chart', cdn = F)\n";

system("R", "--no-save", "--quiet", "--slave", "--file=$rfile");

if ($web) {
  $web =~ s/\/$//;
  open(FILE, "$resultsdir/$chart") or die;
  my @lines = <FILE>;
  close(FILE);
  open(FILE, ">", "$resultsdir/$chart") or die;
  foreach my $line (@lines) {
    $line =~ s/src='.*'\js/src='$web\libraries/dimple/\js/';
    print FILE $line;
  }
  close(FILE);
}

=head2 table

=cut

sub table{
  my ($filename, $lab, $resultsdir, $web) = @_;

  # Make Datatables table
print "Making Table.\n";
my $chart = "$lab.table.html";
my $Rdir = $resultsdir;
my $rfile = "$Rdir/$lab.table.R";
open my $rcfh, "">", $rfile;
print $rcfh "setwd('$Rdir')
results <- read.table('$filename', header = TRUE, sep='\t')
results <- subset(results, T, select = c('Cell', 'Tissue', 'Datatype', 'Accession', 'Pvalue', 'Qvalue', 'Probe'))
require(rCharts)
dt <- dTable(
results,
sScrollY = '600',
bPaginate = F,
sScrollX = '100%',
width = '680px'
)
dt\$save('$chart', cdn = F)\n";

system("R", "--no-save", "--quiet", "--slave", "--file=$rfile");

if ($web) {
$web =~ s/\//$//;
open(FILE, "$resultsdir/$chart") or die;
my @lines = <FILE>;
close(FILE);
open(FILE, "">", "$resultsdir/$chart") or die;
foreach my $line (@lines) {
$line =~ s/href='.*\//href='$web\libraries\datatables\css/;
$line =~ s/src='.*\//src='$web\libraries\datatables\js/;
print FILE $line;
}
close(FILE);
}
}

1;

1;
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 dated June, 1991 or at your option any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

A copy of the GNU General Public License is available in the source tree; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

=head1 CONTACT

Charles Breeze, C<< c.breeze at ucl.ac.uk >>

Javier Herrero, C<< javier.herrero at ucl.ac.uk >>

=head1 ACKNOWLEDGEMENTS

This software is based on the FORGE tool developed by Ian Dunham at the EMBL-EBI

=cut

use 5.010;
use strict;
use warnings FATAL => 'all';

our $VERSION = '0.01';

our (@ISA, @EXPORT);
use Exporter;
@ISA = qw(Exporter);
@EXPORT = qw(mean variance std zscore log10 binomial factorial fdr); # Symbols to export by default

=head1 SYNOPSIS

Provide various stats for eForge to do its stuff

=head1 EXPORT

mean
variance
std
log10
binomial
factorial
fdr

=head1 SUBROUTINES/METHODS

=head2 mean

Calculates the biased mean of an array

pass it a float array and it will return the mean

reused from Ben Brown

=cut

sub mean {
    my $sum = 0;
    foreach (@_){
        $sum += $_;
    }
    return $sum/($#_+1);
}

=head2 variance

Calculates the biased variance of an array

Pass it a float array and it will return the variance

Reused from Ben Brown

=cut

sub variance {
    my $ev = mean(@_);
    my $sum = 0;
    foreach (@_){ $sum += ($_- $ev)**2; }
    return $sum/($#_+1);
}

=head2 std

Calculates the standard deviation of an array: this is just the sqrt of the var

=cut

sub std { sqrt(variance(@_)) }

=head2 log10

log 10 since perl doesn't have

=cut
sub log10 {
    my $n = shift;
    return log($n)/log(10);
}

=head2 zscore

Calculates the z-score for a given result and an array of values

=cut

sub zscore {
    my ($teststat, $values) = @_;  
    my $zscore;
    
    my $mean = mean(@$values);
    my $sd = std(@$values);
    if ($sd == 0) {
        $zscore = "NA";
    } else {
        $zscore = sprintf("%.3f", ($teststat-$mean)/$sd);
    }

    return $zscore;
}

=head2 binomial

Exact solution of binomial probability for k picks out of n, for n or greater need to sum for each k up to n

=cut

sub binomial {
    my ($k, $n, $p) = @_;  
    my $prob = exp($k * log($p) + ($n-$k) * log(1 - $p) + log_factorial($n) - log_factorial($k) - log_factorial($n - $k));

    return $prob;
}

=head2 log_factorial

Calculate log(N!). Required for binomial.
Uses a cache to speed up the calculation.

=cut

my %log_factorial_cache;

sub log_factorial{

}
my ($n) = shift;
return 0 if($n <= 1); # log(1) = 0
return $_log_factorial_cache{$_} if (exists($_log_factorial_cache{$_}));
my $result = 0; # log(1) = 0;
for (my $i = $n; $i > 1; $i--) {
    $result += log($i);
}$_log_factorial_cache{$n} = $result;
return $_log_factorial_cache{$n};

=head2 fdr

Empirical false discovery rate = FP/TP+FP.

Need to modify this now have switched to binomial p values.
=cut

sub fdr{
    my ($tp, $mvps, $cells) = @_; if ($tp == 0){
        return "NA";
    } else{
        my $fpr = 0.0085 * exp(-0.04201 * $mvps) + 0.00187; # from simulations of random data (Forge, not eForge->) 0.0085*exp(-0.04201.SNPs) + 0.00187
        my $fdr = ($cells * $fpr) / $tp;
        return $fdr;
    }  
}

1;

This directory contains the scripts and files necessary to rebuild the eFORGE database from scratch.

In summary, the steps are:
- Create an empty DB
- Load the information about the arrays
- Load the dataset with DHS, Histone peaks, etc
- Move the DB to its final destination

1. CREATE AN EMPTY DATABASE

Please refer to the help of init_db.pl for information about the different options

rm eforge_1.2.db
perl init_db.pl --db_name eforge_1.2.db

2. LOAD THE ARRAYS

Please refer to the help of load_450k_array.pl for information about the different options
3. LOAD THE DATASETS

Please refer to the help of load_dataset.pl for information about the different options

perl load_dataset.pl --db_name eforge_1.2.db --tag erc --name 'Roadmap Epigenomics (2012 data) - DHS' --decode_file erc.decode --work_dir input/erc/
perl load_dataset.pl --db_name eforge_1.2.db --tag encode --name 'ENCODE - DHS' --decode_file encode.decode --work_dir input/encode/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-DHS --name 'Consolidated Roadmap Epigenomics - DHS' --decode_file erc2.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K27me3 --name 'Consolidated Roadmap Epigenomics - H3K27me3' --decode_file erc2-H3K27me3.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K36me3 --name 'Consolidated Roadmap Epigenomics - H3K36me3' --decode_file erc2-H3K36me3.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K4me3 --name 'Consolidated Roadmap Epigenomics - H3K4me3' --decode_file erc2-H3K4me3.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K9me3 --name 'Consolidated Roadmap Epigenomics - H3K9me3' --decode_file erc2-H3K9me3.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag erc2-H3K4me1 --name 'Consolidated Roadmap Epigenomics - H3K4me1' --decode_file erc2-H3K4me1.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag H3-all --name 'Consolidated Roadmap Epigenomics - All H3 marks' --decode_file H3-all.decode --work_dir input/erc2/
perl load_dataset.pl --db_name eforge_1.2.db --tag blueprint --name 'Blueprint - DHS' --decode_file blueprint.decode --work_dir input/blueprint/

4. MOVE THE DATABASE TO ITS FINAL LOCATION

mv eforge_1.2.db ..

===> pasted/database/init_db.pl <==
#! /usr/bin/env perl
use strict;
use warnings;

use DBI;
use Getopt::Long;

my $db_dir = ".";
my $db_name = "eforge_1.2.db";

my $help;

my $desc = q{init_db.pl [--db_name $db_name] [--db_dir $db_dir]};
where:
--db_name is the name of the SQLite file [def: $db_name]
--db_dir is the location of the SQLite file [def: $db_dir]

);

GetOptions(
    "help" => \$help,
    "db_name=s" => \$db_name,
    "db_dir=s" => \$db_dir,
);

if ($help) {
    print $desc;
    exit(0);
}

my $dsn = "dbi:SQLite:dbname=$db_dir/$db_name";
my $dbh = DBI->connect($dsn, "", "") or die $DBI::errstr;

$dbh->do("CREATE TABLE IF NOT EXISTS assembly ( assembly_id INTEGER PRIMARY KEY AUTOINCREMENT, species_name, assembly_name, UNIQUE (species_name, assembly_name))");

$dbh->do("CREATE TABLE IF NOT EXISTS array ( array_id INTEGER PRIMARY KEY AUTOINCREMENT, array_tag UNIQUE, array_name UNIQUE, species_name)");
$dbh->do("CREATE TABLE IF NOT EXISTS probe_mapping_info ( probe_mapping_id INTEGER PRIMARY KEY AUTOINCREMENT, array_id INTEGER NOT NULL REFERENCES array(array_id), assembly_id INTEGER NOT NULL, url, UNIQUE (array_id, assembly_id))");
$dbh->do("CREATE TABLE IF NOT EXISTS probe_mapping ( probe_mapping_id INTEGER NOT NULL REFERENCES probe_mapping_info(probe_mapping_id), probe_id, chr_name, position INTEGER, UNIQUE (probe_mapping_id, probe_id))");
$dbh->do("CREATE INDEX probe_mapping_idx1 on probe_mapping (position, chr_name)");

$dbh->do("CREATE TABLE IF NOT EXISTS proxy_filter_info ( array_id INTEGER NOT NULL REFERENCES array(array_id), description, "");
UNIQUE(array_id));
$dbh->do("CREATE TABLE IF NOT EXISTS proxy_filter (array_id INTEGER NOT NULL REFERENCES array(array_id), probe_id, proxy_probes, UNIQUE (array_id, probe_id))");

$dbh->do("CREATE TABLE IF NOT EXISTS probe_annotation_info (array_id INTEGER NOT NULL REFERENCES array(array_id), gene_reference_name NOT NULL, cgi_reference_name NOT NULL, url, UNIQUE(array_id))");
$dbh->do("CREATE TABLE IF NOT EXISTS probe_annotation (array_id INTEGER NOT NULL REFERENCES probe_annotation_info(array_id), probe_id NOT NULL, gene_group, cgi_group, UNIQUE (array_id, probe_id))");

$dbh->do("CREATE TABLE IF NOT EXISTS dataset (dataset_id INTEGER PRIMARY KEY AUTOINCREMENT, dataset_tag UNIQUE, dataset_name UNIQUE, species_name)");
$dbh->do("CREATE TABLE IF NOT EXISTS sample (dataset_id INTEGER REFERENCES dataset(dataset_id), sample_order INTEGER, file, lab, datatype, cell, tissue, shortcell, individual, acc, url)");
$dbh->do("CREATE TABLE IF NOT EXISTS probe_bitstring (array_id INTEGER NOT NULL REFERENCES array(array_id), probe_id INTEGER NOT NULL, dataset_id INTEGER NOT NULL REFERENCES dataset(dataset_id), sum INTEGER, bit, UNIQUE(array_id,probe_id,dataset_id))");

==> pasted/database/load_27k_array.pl	<==
#! /usr/bin/env perl
use strict;
use warnings;
use Getopt::Long;
use File::Spec qw(splitpath);
use DBI;

my $db_dir = ".";
my $db_name = "eforge_1.2.db";
my $array_tag = "27k";
my $array_name = "Illumina Human 27k array (subset of 450k only)";
my $species = "Homo sapiens";
my $proxy_threshold = 1000;
my $illumina27k_bpm_file = 'ftp://webdata:webdata@ussd-ftp.illumina.com/downloads/ProductFiles/HumanMethylation27/humanmethylation27_270596_v1-2.bpm';
my $illumina450k_csv_file = 'ftp://webdata2:webdata2@ussd-ftp.illumina.com/downloads/ProductFiles/HumanMethylation450/HumanMethylation450_15017482_v1-2.csv';
my $work_dir = ".";
my $bedtools = "bedtools";
my $help;

my $desc = qq{load_27k_array.pl [options]

DESCRIPTION:

This script loads the subset of probe locations and annotations of the Illumina Human 27k methylation array that are shared with the 450k array into the eFORGE database.

Note that you *must* load the arrays before loading the datasets. If you want to include new arrays at a later date, you will have to reload all the datasets again (i.e. you will have to re-start from scratch).

Optional arguments:
--db_name <name>
  is the name of the SQLite file [def: $db_name]
--db_dir <path>
  is the location of the SQLite file [def: $db_dir]
--work_dir <path>
  is the location where temporary files will be downloaded/created [def: $work_dir]
--bedtools <name>
  is the name of the bedtools executable you want to use [def: $bedtools]
};

GetOptions(
  "help" => \\
$help,
  "db_name=s" => \\
$db_name,
  "db_dir=s" => \\
$db_dir,
  "work_dir=s" => \\
$work_dir,
  "bedtools=s" => \\
$bedtools,
);

my $dsn = "dbi:SQLite:dbname=$db_dir/$db_name";
my $dbh = DBI->connect($dsn, "", "") or die $DBI::errstr;
my ($vol, $path, $file) = File::Spec->splitpath($illumina450k_csv_file);

if (!-e "$work_dir/$file") {
    download_url($illumina450k_csv_file, $work_dir);
}

my $array450k_info = parse_450k_file("$work_dir/$file");

my ($vol27k, $path27k, $file27k) = File::Spec->splitpath($illumina27k_bpm_file);

if (!-e "$work_dir/$file27k") {
    download_url($illumina27k_bpm_file, $work_dir);
}

my $array27k_info = parse_27k_file("$work_dir/$file27k");

my $array_info = subset_array($array450k_info, $array27k_info);

load_array($dbh, $db_name, $species, $array_tag, $array_name, $array_info);

load_proxy_filter($dbh, $db_name, $array_tag, "GRCh37", $work_dir, $proxy_threshold);
exit();

sub download_url {
    my ($url, $work_dir) = @_;

    print "Downloading $url...
";
    system("wget -q -N -P $work_dir $url");
}

sub parse_450k_file {
    my ($illumina450k_csv_file) = @_;
    my $array;

    open(CSV, $illumina450k_csv_file) or die;
    my $annotation;
    my @gene_annotations = ("TSS200", "TSS1500", "1stExon", "Body", "3'UTR", "5'UTR", "IGR");
    my @cpg_annotations = {
        'N_Shelf' => "Shelf_Shore",
        'S_Shelf' => "Shelf_Shore",
        'N_Shore' => "Shelf_Shore",
        'S_Shore' => "Shelf_Shore",
        'Island' => "Island"};
    while (<CSV>) {
        chomp;
        my @data = split("", $_);
        my $probe_id = $data[0];
        next if ($probe_id !~ /^cg/ and $probe_id !~ /^chr\./);
        my $probe_chr37 = $data[11];
my $probe_loc37 = $data[12];
my $probe_chr36 = $data[14];
my $probe_loc36 = $data[15];
my $this_gene_annotation_arrayStr = $data[23];
my $this_cpg_annotation = $data[25]?$cpg_annotations->{$data[25]}:"NA";

my $this_gene_annotation;
if (!defined($this_gene_annotation_arrayStr) or $this_gene_annotation_arrayStr eq 
"") {
  $this_gene_annotation = "IGR";
} else {
  my @this_gene_annotations = split(";", $this_gene_annotation_arrayStr);
  foreach my $this_a (@this_gene_annotations) {
    if (grep {$_ eq $this_a} @gene_annotations) {
      $this_gene_annotation = $this_a;
      last;
    }
  }
}

$annotation->{this_gene_annotation."-".$this_cpg_annotation}++;

$array->{probe_id} = [$probe_chr36, $probe_loc36, $probe_chr37, $probe_loc37, $this_gene_annotation, $this_cpg_annotation];
}
close(CSV);
# foreach my $a (keys %$annotation) { #@gene_annotations) {
#  print $annotation->{this_a}, ",", $this_a, "\n";
#  }
#}
return $array;
}

sub parse_27k_file {
  my ($illumina27k_bpm_file) = @_;  
  my $array;

  open(CSV, $illumina27k_bpm_file) or die;
  my $block = "";
  while (<CSV>) {
  if (/^\[(\w+)\]/) {
    $block = $1;
  }
  next if ($block ne "Assay");

chomp;
  my @data = split("", $_);
  my $probe_id = $data[0];
  my $name = $data[1];
  my $probe_chr36 = $data[8];
  my $probe_loc36 = $data[9];
  my $probe_loc37 = $data[12];
  my $probe_chr36 = $data[14];
  my $probe_loc37 = $data[15];
  my $this_gene_annotation = $data[23];
  my $this_cpg_annotation = $data[25]?$cpg_annotations->{$data[25]}:"NA";

  my $this_gene_annotation;
  if (!defined($this_gene_annotation_arrayStr) or $this_gene_annotation_arrayStr eq 
"") {
    $this_gene_annotation = "IGR";
  } else {
    my @this_gene_annotations = split(";", $this_gene_annotation_arrayStr);
    foreach my $this_a (@this_gene_annotations) {
      if (grep {$_ eq $this_a} @gene_annotations) {
        $this_gene_annotation = $this_a;
        last;
      }
    }
  }

  $annotation->{this_gene_annotation."-".$this_cpg_annotation}++;

  $array->{probe_id} = [$probe_chr36, $probe_loc36, $probe_chr37, $probe_loc37, $this_gene_annotation, $this_cpg_annotation];
}
close(CSV);
# foreach my $a (keys %$annotation) { #@gene_annotations) {
#  print $annotation->{this_a}, ",", $this_a, "\n";
#  }
#}
return $array;
}
# print " Skipping $probe_id\n";
    next;
  }
  if ($probe_id ne $name) {
    print " ERROR: $probe_id ne $name\n";
  }
  $array-{$probe_id} = [$probe_chr36, $probe_loc36];
}
close(CSV);

return $array;
}

sub subset_array {
  my ($full_array, $sub_array) = @_;  
  my $array;
  foreach my $probe_id (keys %$full_array) {
    if ($sub_array-{$probe_id}) {
      $array-{$probe_id} = $full_array-{$probe_id};
      print "MISMATCH $probe_id\n" if (($full_array-{$probe_id}->[0] ne $sub_array-{$probe_id}->[0])
                             or
                             ($full_array-{$probe_id}->[1] ne $sub_array-{$probe_id}->[1]));
    }
  }
  return $array;
}

sub load_array_1 {
  my ($dbh, $code_name, $array_info) = @_;  
  my $table_name = $code_name;
  $table_name =~ s/W//g;
  $table_name = "array_$table_name";
  $dbh->do("CREATE TABLE IF NOT EXISTS $table_name (probe_id, location, gene_group, cgi_group)");

  my $sth = $dbh->prepare("INSERT INTO array_$code_name VALUES (?, ?, ?, ?)");
  foreach my $this_probe_id (sort keys $array_info) {
    my ($chr, $loc, $gene_group, $cgi_group) = @{$array_info-{$this_probe_id}};
    my $location = "$chr:$loc-$loc";
    $sth->execute($this_probe_id, $location, $gene_group, $cgi_group);
  }
  $sth->finish();

  return $table_name;
}

sub load_array {
  my ($dbh, $db_name, $species_name, $array_tag, $array_name, $array_info) = @_;
my $sth;

$sth = $dbh->prepare("INSERT OR IGNORE INTO array (array_tag, array_name,
species_name) VALUES (?, ?, ?)");
$sth->execute($array_tag, $array_name, $species_name);
my $array_id = $dbh->last_insert_id("", "", "", "");
$sth->finish();
if ($array_id == 0) {
    $array_id = $dbh->selectrow_array("SELECT array_id FROM array WHERE
    array_name = '$array_name' AND species_name = 'Homo sapiens'");
}

$sth = $dbh->prepare("INSERT OR IGNORE INTO assembly (species_name,
assembly_name) VALUES (?, ?)");
$sth->execute($species_name, "GRCh37");
$sth->execute($species_name, "NCBI36");
$sth->finish();
my $human36_assembly_id = $dbh->selectrow_array("SELECT assembly_id
    FROM assembly WHERE species_name = 'Homo sapiens' AND assembly_name = 'NCBI36'");
my $human37_assembly_id = $dbh->selectrow_array("SELECT assembly_id
    FROM assembly WHERE species_name = 'Homo sapiens' AND assembly_name = 'GRCh37'");

$sth = $dbh->prepare("INSERT OR IGNORE INTO probe_mapping_info (array_id,
assembly_id, url) VALUES (?, ?, ?)");
$sth->execute($array_id, $human36_assembly_id, "");
my $probe_mapping_human36_id = $dbh->last_insert_id("", "", "", "");
$sth->execute($array_id, $human37_assembly_id, "");
my $probe_mapping_human37_id = $dbh->last_insert_id("", "", "", "");
$sth->finish();

$sth = $dbh->prepare("INSERT OR IGNORE INTO probe_annotation_info (array_id,
gene_reference_name, cgi_reference_name, url) VALUES (?, ?, ?, ?)");
$sth->execute($array_id, "RefSeq", "UCSC CpG Islands", "");
# my $probe_annotation_id = $dbh->last_insert_id("", "", "", "");
$sth->finish();

open(CSV1, "probe_mapping.csv") or die;
open(CSV2, "probe_annotation.csv") or die;
# my $sth1 = $dbh->prepare("INSERT OR IGNORE INTO probe_mapping
(probe_mapping_id, probe_id, chr_name, position) VALUES (?, ?, ?, ?)");
# my $sth2 = $dbh->prepare("INSERT OR IGNORE INTO probe_annotation
(probe_annotation_id, probe_id, gene_group, cgi_group) VALUES (?, ?, ?, ?)");
foreach my $this_probe_id (sort keys $array_info) {
    my ($chr36, $loc36, $chr37, $loc37, $gene_group, $cgi_group) = @{$array_info
   ->[0]$this_probe_id};
    print CSV1 join(",", $probe_mapping_human36_id, $this_probe_id, "chr$chr36",
    $loc36), "\n";
    print CSV1 join(",", $probe_mapping_human37_id, $this_probe_id, "chr$chr37",
    $loc37), "\n";
    # print CSV2 join(",", $probe_annotation_id, $array_id, $this_probe_id, $gene_group,
    # $cgi_group), "\n";
    print CSV2 join(",", $array_id, $this_probe_id, $gene_group, $cgi_group), "\n";
    # $sth1->execute($probe_mapping_human36_id, $this_probe_id, "chr$chr36",
    $loc36) if ($probe_mapping_human36_id);
Appendices

```perl
# $sth1->execute($probe_mapping_human37_id, $this_probe_id, "chr$chr37",
# $loc37) if ($probe_mapping_human37_id);
# $sth2->execute($probe_annotation_id, $this_probe_id, $gene_group, $cgi_group) if
# ($probe_annotation_id);
}
close(CSV1);
close(CSV2);
system("echo '.mode csv
.import probe_mapping.csv probe_mapping
.import probe_annotation.csv probe_annotation' | sqlite3 $db_name");
# $sth1->finish();
# $sth2->finish();
}

sub load_proxy_filter {
  my ($dbh, $db_name, $array_tag, $assembly_name, $work_dir, $distance_threshold) = @_;

  my $array_id = $dbh->selectrow_arrayref("SELECT array_id FROM array WHERE array_tag = '$array_tag'")->[0];
  my $bed_file = dump_array_bed_file($dbh, $array_id, $assembly_name, $work_dir);
  my $this_output_bed_file = "$work_dir/array_${array_id}.proxy.bed";
  my $runstr = "$bedtools window -w $distance_threshold -a $bed_file -b $bed_file >
  $this_output_bed_file";
  system($runstr) == 0 or die "Error while running bedtools: $?";

  my $sth = $dbh->prepare("INSERT OR IGNORE INTO proxy_filter_info (array_id, description) VALUES (?, ?)");
  $sth->execute($array_id, "1kb");

  open(BED, $this_output_bed_file) or die;
  my $all_probes;
  my $mapping_probes;
  while (<BED>) {
    chomp;
    my ($chr1, $start1, $end1, $probe1, $chr2, $start2, $end2, $probe2) = split("\t", $_);
    $all_probes->{$probe1} = 1;
    if ($probe1 ne $probe2) {
      $mapping_probes->{$probe1}->{$probe2} = 1;
    }
  }
  close(BED);
  open(CSV, ">proxy_filter.csv") or die;
  foreach my $probe (sort keys %$all_probes) {
    if (defined($mapping_probes->{${$probe}})) {
      print CSV "$array_id,$probe," join("|", sort keys %{$mapping_probes->{$probe}}), "\n";
    } else {
      print CSV "$array_id,$probe,NONE\n";
    }
  }
  close(CSV);
  system("echo '.mode csv
.import probe_mapping.csv probe_mapping
.import probe_annotation.csv probe_annotation' | sqlite3 $db_name");"
sub dump_array_bed_file {
    my ($dbh, $this_array_id, $assembly_name, $work_dir) = @_;  
    my $sql = "SELECT probe_mapping_id
             FROM probe_mapping_info
             JOIN assembly USING (assembly_id)
             WHERE array_id = $this_array_id
             AND assembly_name = '$assembly_name'";

    my $probe_mapping_id = $dbh->selectrow($sql);
    if (!$probe_mapping_id) {
        die "Cannot find a mapping for array $this_array_id and assembly $assembly_name\n";
    }

    $sql = "SELECT probe_id, chr_name, position FROM probe_mapping WHERE
           probe_mapping_id = $probe_mapping_id";
    my $probes = $dbh->selectall_arrayref($sql);

    my $bed_file = "$work_dir/array_$this_array_id.bed";
    open(BED, ">$bed_file") or die;
    foreach my $this_probe (sort_sort_probes @$probes) {
        print BED join("\t", $this_probe->[1], $this_probe->[2], $this_probe->[2]+1,
                       $this_probe->[0]), "\n";
    }
    close(BED);

    return $bed_file;
}

sub_sort_probes {
    my $chr_a = $a->[1];
    my $chr_b = $b->[1];
    my $loc_a = $a->[2];
    my $loc_b = $b->[2];
    $chr_a =~ s/chr//;
    $chr_b =~ s/chr//;

    if ($chr_a eq $chr_b) {
        return $loc_a <=> $loc_b;
    } elsif ($chr_a =~ /^\d/ and $chr_b =~ /^\d/) {
        return $chr_a <=> $chr_b;
    } elsif ($chr_a =~ /^\d/) {
        return -1;
    } elsif ($chr_b =~ /^\d/) {
        return 1;
    } else {
        return $chr_a cmp $chr_b;
    }  
}

exit();
#!/usr/bin/env perl
use strict;
use warnings;
use Getopt::Long;
use File::Spec qw(splitpath);
use DBI;

my $db_dir = ".";
my $db_name = "eforge_1.2.db";
my $array_tag = "450k";
my $array_name = "Illumina Human 450k array";
my $species = "Homo sapiens";
my $proxy_threshold = 1000;
my $illumina450k_csv_file = 'ftp://webdata2:webdata2@ussd-ftp.illumina.com/downloads/ProductFiles/HumanMethylation450/HumanMethylation450_15017482_v1-2.csv';
my $work_dir = ".";
my $bedtools = "bedtools";

my $help;

my $desc = qq{load_450k_array.pl [options]
DESCRIPTION:
This script loads the probe locations and annotations of the Illumina Human 450k methylation array into the eFORGE database.

Note that you *must* load the arrays before loading the datasets. If you want to include new arrays at a later date, you will have to reload all the datasets again (i.e. you will have to re-start from scratch).

Optional arguments:
--db_name <name>
is the name of the SQLite file [def: $db_name]
--db_dir <path>
is the location of the SQLite file [def: $db_dir]
--work_dir <path>
is the location where temporary files will be downloaded/created [def: $work_dir]
--bedtools <name>
is the name of the bedtools executable you want to use [def: $bedtools]
};

GetOptions(
    "help" => \$help,
    "db_name=s" => \$db_name,
    "db_dir=s" => \$db_dir,
    "work_dir=s" => \$work_dir,
);
"bedtools=s" => \$bedtools,
);

my $dsn = "dbi:SQLite:dbname=$db_dir/$db_name";
my $dbh = DBI->connect($dsn,"","") or die $DBI::errstr;

my ($vol, $apth, $file) = File::Spec->splitpath($illumina450k_csv_file);

if (!-e "$work_dir/$file") {
    download_url($illumina450k_csv_file, $work_dir);
}

my $array_info = parse_450k_file("$work_dir/$file");
load_array($dbh, $db_name, $species, $array_tag, $array_name, $array_info);
load_proxy_filter($dbh, $db_name, $array_tag, "GRCh37", $work_dir, $proxy_threshold);
exit();

sub download_url {
    my ($url, $work_dir) = @_;

    print "Downloading $url...\n";
    system("wget -q -N -P $work_dir $url");
}

sub parse_450k_file {
    my ($illumina450k_csv_file) = @_;  
    my $array;

    open(CSV, $illumina450k_csv_file) or die;
    my $annotation;
    my @gene_annotations = ("TSS200", "TSS1500", "1stExon", "Body", "3'UTR", "5'UTR", "IGR");
    my $cpg_annotations = {
        'N_Shelf' => "Shelf_Shore",
        'S_Shelf' => "Shelf_Shore",
        'N_Shore' => "Shelf_Shore",
        'S_Shore' => "Shelf_Shore",
        'Island'  => "Island"};
    while (<CSV>) {
        chomp;
        my @data = split("," ,$_);
        my $probe_id = $data[0];
        next if ($probe_id !~ /^cg/ and $probe_id !~ /^ch\d/);
        my $probe_chr37 = $data[11];
        my $probe_loc37 = $data[12];
        my $probe_chr36 = $data[14];
        my $probe_loc36 = $data[15];
        my $this_gene_annotation_arrayStr = $data[23];
        my $this_cpg_annotation = $data[25]?$cpg_annotations->{$data[25]}:"NA";
    }
my $this_gene_annotation;
if (!defined($this_gene_annotation_arrayStr) or $this_gene_annotation_arrayStr eq "") {
    $this_gene_annotation = "IGR";
} else {
    my @this_gene_annotations = split(semicolon, $this_gene_annotation_arrayStr);
    foreach my $this_a (@this_gene_annotations) {
        if (grep {$_ eq $this_a} @gene_annotations) {
            $this_gene_annotation = $this_a;
            last;
        }
    }
}

$annotation->{$this_gene_annotation."-".$this_cpg_annotation}++;

$array->{[$probe_id] = [[$probe_chr36, $probe_loc36, $probe_chr37, $probe_loc37, $this_gene_annotation, $this_cpg_annotation];}
close(CSV);
# foreach my $this_a (keys %$annotation) { #@gene_annotations) {
#     print $annotation->{$this_a}, "\t", $this_a, "\n";
# }

return @array;
}

sub load_array_1 { my ($dbh, $code_name, $array_info) = @_; my $table_name = $code_name; $table_name =~ s/\W//g;
$table_name = "array_$table_name";
$dbh->do("CREATE TABLE IF NOT EXISTS $table_name (probe_id, location, gene_group, cgi_group)");

my $sth = $dbh->prepare("INSERT INTO array_$code_name VALUES (?, ?, ?, ?)");
foreach my $this_probe_id (sort keys %$array_info) {
    my ($chr, $loc, $gene_group, $cgi_group) = @{$array_info->{$this_probe_id}};
    $sth->execute($this_probe_id, "$chr:$loc - $loc", $gene_group, $cgi_group);
}

return $table_name;
}

sub load_array {
my ($dbh, $db_name, $species_name, $array_tag, $array_name, $array_info) = @_; my $sth;
$sth = $dbh->prepare("INSERT OR IGNORE INTO array (array_tag, array_name, species_name) VALUES (?, ?, ?)");
$sth->execute($array_tag, $array_name, $species_name);
my $array_id = $dbh->last_insert_id("", "", "", "");
$sth->finish();
if ($array_id == 0) {
    $array_id = $dbh->selectrow_array("SELECT array_id FROM array WHERE array_name = 'array_name' AND species_name = 'Homo sapiens'");
}

$sth = $dbh->prepare("INSERT OR IGNORE INTO assembly (species_name, assembly_name) VALUES (?, ?)");
$sth->execute($species_name, "GRCh37");
$sth->execute($species_name, "NCBI36");
$sth->finish();
my $human36_assembly_id = $dbh->selectrow_array("SELECT assembly_id FROM assembly WHERE species_name = 'Homo sapiens' AND assembly_name = 'NCBI36'");
my $human37_assembly_id = $dbh->selectrow_array("SELECT assembly_id FROM assembly WHERE species_name = 'Homo sapiens' AND assembly_name = 'GRCh37'");

$sth = $dbh->prepare("INSERT OR IGNORE INTO probe_mapping_info (array_id, assembly_id, url) VALUES (?, ?, ?)");
$sth->execute($array_id, $human36_assembly_id, "");
$sth->execute($array_id, $human37_assembly_id, "");
$sth->finish();

$sth = $dbh->prepare("INSERT OR IGNORE INTO probe_annotation_info (array_id, gene_reference_name,/cgi_reference_name, url) VALUES (?, ?, ?, ?)");
$sth->execute($array_id, "RefSeq", "UCSC CpG Islands", "");
my $probe_annotation_id = $dbh->last_insert_id("", "", "", "");
$sth->finish();

open(CSV1, ">probe_mapping.csv") or die;
open(CSV2, ">probe_annotation.csv") or die;
# my $sth1 = $dbh->prepare("INSERT OR IGNORE INTO probe_mapping (probe_mapping_id, probe_id, chr_name, position) VALUES (?, ?, ?, ?)");
# my $sth2 = $dbh->prepare("INSERT OR IGNORE INTO probe_annotation (probe_annotation_id, probe_id, gene_group, cgi_group) VALUES (?, ?, ?, ?)");
foreach my $this_probe_id (sort keys $array_info) {
    my ($chr36, $loc36, $chr37, $loc37, $gene_group, $cgi_group) = @{$array_info->{$this_probe_id}};
    print CSV1 join(",", $probe_mapping_human36_id, $this_probe_id, \"chr\$chr36\", $loc36), "\n";
    print CSV1 join(",", $probe_mapping_human37_id, $this_probe_id, \"chr\$chr37\", $loc37), "\n";
    print CSV2 join(",", $probe_annotation_id, $array_id, $this_probe_id, $gene_group, $cgi_group), "\n";
    print CSV2 join(",", $array_id, $this_probe_id, $gene_group, $cgi_group), "\n";
    $sth1->execute($probe_mapping_human36_id, $this_probe_id, \"chr\$chr36\", $loc36) if ($probe_mapping_human36_id);
    $sth1->execute($probe_mapping_human37_id, $this_probe_id, \"chr\$chr37\", $loc37) if ($probe_mapping_human37_id);
sub load_proxy_filter {
    my ($dbh, $db_name, $array_tag, $assembly_name, $work_dir, $distance_threshold) = @_;

    my $array_id = $dbh->selectrow_arrayref("SELECT array_id FROM array WHERE array_tag = '$array_tag'");
    my $bed_file = dump_array_bed_file($dbh, $array_id, $assembly_name, $work_dir);
    my $this_output_bed_file = "$work_dir/array_${array_id}.proxy.bed";
    my $runstr = "$bedtools window -w $distance_threshold -a $bed_file -b $bed_file > $this_output_bed_file";
    system($runstr) == 0 or die "Error while running bedtools: $?";

    my $sth = $dbh->prepare("INSERT OR IGNORE INTO proxy_filter_info (array_id, description) VALUES (?, ?)");
    $sth->execute($array_id, "1kb");

    open(BED, $this_output_bed_file) or die;
    my $all_probes;
    my $mapping_probes;
    while (<BED>) {
        chomp;
        my ($chr1, $start1, $end1, $probe1, $chr2, $start2, $end2, $probe2) = split("\t", $_);
        $all_probes->{$probe1} = 1;
        if ($probe1 ne $probe2) {
            $mapping_probes->{$probe1}->{$probe2} = 1;
        }
    }
    close(BED);

    open(CSV, "$this_output_bed_file") or die;
    foreach my $probe (sort keys %$all_probes) {
        if (defined($mapping_probes->{$probe})) {
            print CSV "$array_id,$probe", join("|", sort keys %{$mapping_probes->{$probe}}), \n; }
        else { 
            print CSV "$array_id,$probe,NONE
; }
    }
    close(CSV);
    system("echo '.mode csv.import proxy_filter.csv proxy_filter'	| sqlite3 $db_name");
sub dump_array_bed_file {
    my ($dbh, $this_array_id, $assembly_name, $work_dir) = @_; 
    my $sql = "SELECT probe_mapping_id
        FROM probe_mapping_info
        JOIN assembly USING (assembly_id)
        WHERE array_id = $this_array_id
        AND assembly_name = '$assembly_name';"
    my $probe_mapping_id = $dbh->selectrow_array($sql);
    if (!$probe_mapping_id) {
        die "Cannot find a mapping for array $this_array_id and assembly $assembly_name\n";
    }
    $sql = "SELECT probe_id, chr_name, position FROM probe_mapping WHERE
        probe_mapping_id = $probe_mapping_id";
    my $probes = $dbh->selectall_arrayref($sql);
    my $bed_file = "$work_dir/array_${this_array_id}.bed";
    open(BED, ">$bed_file") or die;
    foreach my $this_probe (sort_sort_probes @$probes) {
        print BED join("\t", $this_probe->[1], $this_probe->[2], $this_probe->[2]+1, $this_probe->[0]), "\n";
    }
    close(BED);
    return $bed_file;
}

sub_sort_probes {
    my $chr_a = $a->[1];
    my $chr_b = $b->[1];
    my $loc_a = $a->[2];
    my $loc_b = $b->[2];
    $chr_a =~ s/chr//;
    $chr_b =~ s/chr//;
    if ($chr_a eq $chr_b) {
        return $loc_a <=> $loc_b;
    } elsif ($chr_a =~ /^\d/ and $chr_b =~ /^\d/) {
        return $chr_a <=> $chr_b;
    } elsif ($chr_a =~ /^\d/) {
        return -1;
    } elsif ($chr_b =~ /^\d/) {
        return 1;
    } else {
        return $chr_a cmp $chr_b;
    }
}

exit();

===> pasted/database/load_dataset.pl <==
#!/usr/bin/env perl
use strict;
use warnings;

use Getopt::Long;
use DBI;

my $db_dir = ".";
my $db_name = "eforge_1.1.db";
my $dataset_name = "ENCODE";
my $dataset_tag = "encode";
my $decode_file = "encode.decode";
my $work_dir = "/tmp";
my $species_name = "Homo sapiens";
my $assembly_name = "GRCh37";
my $bedtools = "bedtools";

my $help;

my $desc = qq{load_dataset.pl [options]

DESCRIPTION:

This script reads a 'decode' file which contains references to a list of samples, each of them corresponding to a given dataset. Each of these files is a BED or BED-like file that is read by bedtools to find overlaps with each and every array loaded in the eFORGE database.

Note that you *must* load the arrays first. If you want to include new arrays at a later date, you will have to reload all the datasets again (i.e. you will have to re-start from scratch).

Required parameters:
--tag <tag> or --dataset_tag <tag>
  the ID for this database. This needs to be unique.
--name <name> or --dataset_name <name>
  the name for this database. This can be a longer description. It will be used in the web interface.
--decode_file <file>
  the file with the information about each sample in this dataset

Optional parameters:
--db_name <name>
  is the name of the SQLite file [def: $db_name]
--db_dir <path>
  is the location of the SQLite file [def: $db_dir]
--work_dir <path>
  is the location where temporary files will be downloaded/created [def: $work_dir]
--bedtools <name>
  is the name of the bedtools executable you want to use [def: $bedtools]
--species <species>
  is the name of the species [def: $species_name]
--assembly <assembly>
is the name of the assembly [def: $assembly_name]
);

GetOptions(
    "help" => \$help,
    "db_name=s" => \$db_name,
    "db_dir=s" => \$db_dir,
    "tag|dataset_tag=s" => \$dataset_tag,
    "name|dataset_name=s" => \$dataset_name,
    "decode_file=s" => \$decode_file,
    "work_dir=s" => \$work_dir,
    "species=s" => \$species_name,
    "assembly=s" => \$assembly_name,
);

if ($help) {
    print $desc;
    exit(0);
}

my $dsn = "dbi:SQLite:dbname=$db_dir/$db_name";
my $dbh = DBI::connect($dsn, "", "") or die $DBI::errstr;

system("mkdir -p $work_dir");

my $decode_table = get_decode_table($decode_file);

my $dataset_id = load_dataset($dbh, $species_name, $decode_table, $dataset_name, $dataset_tag);

download_bed_files($decode_table, $work_dir);

my $arrays = get_all_arrays_for_species($dbh, $species_name);

foreach my $this_array (@$arrays) {
    my ($this_array_id, $this_array_name) = @$this_array;
    my $sorted_array_bed_file = dump_array_bed_file($dbh, $this_array_id, $assembly_name, $work_dir);
    my $single_overlaps_bed_files = run_single_overlap_bedtools($bedtools, $work_dir, $sorted_array_bed_file, $decode_table, $this_array_id, $this_array_name);
    my $concatenated_overlaps_bed_file = paste_files($single_overlaps_bed_files, $work_dir, $this_array_id);
    my $final_overlaps_bed_file = add_sum_column_to_concatenated_bedfile($concatenated_overlaps_bed_file, $work_dir, $this_array_id);
    load_bitstrings($dbh, $db_name, $final_overlaps_bed_file, $this_array_id, $dataset_id);
}

my $input_bed_files = get_input_bed_files_from_decode_table($decode_file);
exit(0);

sub load_dataset {
    my ($dbh, $species_name, $decode_table, $dataset_name) = @_;
    my $dataset_id;

    my $sth;
    $sth = $dbh->prepare("INSERT OR IGNORE INTO dataset (dataset_tag, dataset_name, species_name) VALUES (?, ?, ?)");
    $sth->execute($dataset_tag, $dataset_name, $species_name);
    $dataset_id = $dbh->last_insert_id("", "", "");
    $sth->finish();
    if ($dataset_id == 0) {
        $dataset_id = $dbh->selectrow_array("SELECT dataset_id FROM dataset WHERE dataset_name = '$dataset_name' AND species_name = '$species_name'");
    }

    $sth = $dbh->prepare("INSERT OR IGNORE INTO sample (dataset_id, sample_order, file, lab, datatype, cell, tissue, shortcell, individual, acc, url) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)");
    my $sample_order = 1;
    foreach my $this_sample (@$decode_table) {
        $sth->execute($dataset_id, $sample_order, $this_sample->{file}, $this_sample->{lab}, $this_sample->{datatype}, $this_sample->{cell}, $this_sample->{tissue}, $this_sample->{shortcell}, $this_sample->{individual}, $this_sample->{acc}, $this_sample->{url});
        $sample_order++;
    }

    return($dataset_id);
}

sub load_bitstrings {
    my ($dbh, $db_name, $bed_file, $array_id, $dataset_id) = @_; 

    # my $sql = "INSERT INTO probe_bitstring (array_id, probe_id, dataset_id, sum, bit)
    # VALUES (?, ?, ?, ?)";
    # my $sth = $dbh->prepare($sql);
    open(BED, $bed_file) or die "Cannot open BED file $bed_file\n";
    open(CSV, ">probe_bitstring.csv") or die "Cannot open CVS temporary file probe_bitstring.csv\n";
    while(<BED>) {
        chomp;
        my ($chr, $start, $end, $probe_id, $sum, $bitstring) = split("\t", $.
        print CSV join("", $array_id, $probe_id, $dataset_id, $sum, $bitstring), "\n";
Appendices

```perl
# $sth->execute($array_id, $probe_id, $dataset_id, $sum, $bitstring);
}
close(BED);
close(CSV);
system("echo '.mode csv .import probe_bitstring.csv probe_bitstring' | sqlite3 $db_name");
#
$sth->finish();
}
=head2 add_sum_column_to_concatenated_bedfile

Arg[1] : string $concatenated_overlaps_bed_file (location of the input BED file with 0/1 flags on the 4th column)
Arg[2] : string $work_dir (where to put the temporary files)
Example  : my $final_overlaps_bed_file = add_sum_column_to_concatenated_bedfile($concatenated_overlaps_bed_file, $work_dir);
Description : Reads the input BED file ($concatenated_overlaps_bed_file) which contains a series of 0 and 1 flags in the 4th column. This function reads the number of ones in that column and include that value in the output BED file. The output BED file will contain that number in the 4th column and the series of flag in the 5th column.
Returns : string $final_overlaps_bed_file (the location of the resulting BED file)
 Exceptions : Dies if error when opening the files
=cut

sub add_sum_column_to_concatenated_bedfile {
    my ($concatenated_overlaps_bed_file, $work_dir, $array_id) = @_; 
    my $final_overlaps_bed_file = "$work_dir/final_overlaps.array_${array_id}.bed";

    open(BED_IN, $concatenated_overlaps_bed_file) or die "Cannot open BED file $concatenated_overlaps_bed_file\n";
    open(BED_OUT, ">$final_overlaps_bed_file") or die "Cannot open BED file $final_overlaps_bed_file\n";
    while(<BED_IN>) {
        chomp;
        my ($this_chr, $this_start, $this_end, $this_probe, $this_bitstring) = split("\t", $_);
        my $num = $this_bitstring =~ tr/1/1/;
        print BED_OUT join("\t", $this_chr, $this_start, $this_end, $this_probe, $num, $this_bitstring), "\n";
    }
    close(BED_IN);
    close(BED_OUT);

    return $final_overlaps_bed_file;
}
=head2 run_single_overlap_bedtools

Arg[1] : string $bedtools (either full path or just the binary if in the $PATH)
Arg[2] : string $work_dir (where to put the temporary files)
```
Appendices

Arg[3] : string $sorted_450k_bed_file (location of the BED file with sorted 450K features)
Example : my $single_overlaps_bed_files = run_single_overlap_bedtools($bedtools,
$work_dir, $sorted_450k_bed_file, $decode_table);
Description : Run bedtools on the sorted 450K features vs all the BED DNAse features, one at a time.
Returns : arrayref of string $total_overlaps_bed_files (the locations of the resulting BED files)
Exceptions : Dies if error when running bedtools
=cut

sub run_single_overlap_bedtools {
    my ($bedtools, $work_dir, $sorted_array_bed_file, $decode_table, $array_id,
    $array_name) = @_;  
    my $single_overlap_bed_files = [];
    foreach my $this_input_bed_file (map {$_->"{file}"} @$decode_table) {
        print "Overlap between $array_name and $this_input_bed_file...
        my $this_output_bed_file = $this_input_bed_file;
        $this_output_bed_file =~ s/.+///;
        $this_output_bed_file = "$work_dir/single_overlaps.array_${array_id}.$this_output_bed_file";
        my $runstr = "$bedtools intersect -c -a $sorted_array_bed_file -b
        $work_dir/$this_input_bed_file > $this_output_bed_file";
        system($runstr) == 0 or die "Error while running bedtools: $?";
        push(@$single_overlap_bed_files, $this_output_bed_file);
    }
    return $single_overlap_bed_files;
}

sub get_decode_table {
    my ($decode_file) = @_;  
    my $decode_table;
    open(DECODE, $decode_file) or die "Cannot open decode file <$decode_file>\n";
    my $whole_text = join("\n", <DECODE>);
    $whole_text =~ s/[\r\n]+//g;
    my @lines = split("\n", $whole_text);
    close(DECODE);
    my @header = split("\t", shift(@lines));
    foreach my $this_line (@lines) {
        my @data = split("\t", $this_line);
        my $decode_record = {};
        for (my $i=0; $i<@header; $i++) {
            $decode_record->{$header[$i]} = $data[$i];
        }
    }
    # Check that file and URL match
Appendices

die "File: ".file_record->"file"."\nURL: ".file_record->"url"."\nOn line: 
$file
if ($file_record->"url" !~ $file_record->"file");
$file = $file_record->"url";
$file =~ s/+///g;
push(@$decode_table, $file_record);
}
return $decode_table;
}
sub download_bed_files {
  my ($decode_table, $work_dir) = @_;;

  foreach my $this_decode_entry (@$decode_table) {
    my $url = $this_decode_entry->"url";
    my $file = $this_decode_entry->"file";
    if (!-e "$work_dir/$file") {
      download_url($url, $work_dir);
    }
  }
}

sub download_url {
  my ($url, $work_dir) = @_;;

  print "Downloading $url...\n";
  system("wget", "-q", "-N", "-P", $work_dir, $url) == 0
    or die "Error: wget -q -N $work_dir $url\n$!";
}

sub paste_files {
  my ($files, $work_dir, $array_id) = @_;;
  my $output_file = "$work_dir/concat_overlaps.array_${array_id}.bed";

  my $max_num_files = 200;
  my $c = 0;
  my $temp_output_file;
  my @original_files = @$files;
  while (@$files > 1) {
    $c++;
    $temp_output_file = "$work_dir/temp.$$.$c.paste_files.txt";
    my $input_files = [splice(@$files, 0, $max_num_files)];
    #    print STDERR "Merging ", join(" ", @input_files), " into $temp_output_file\n";
    merge_files($temp_output_file, $input_files);
    unshift(@$files, $temp_output_file);
  }
  rename($temp_output_file, $output_file);

  unlink glob "$work_dir/temp.$$.paste_files.txt";
  #
sub merge_files {
    my ($output_file, $input_files) = @_;  
    my @fhs;  
    my $c = 0;  
    foreach my $this_input_file (@$input_files) {
        open($fhs[$c++], $this_input_file) or die "Cannot open $this_input_file: $!
        \n";
    }
    open(OUT, "->$output_file") or die "Cannot open $output_file: $!
        \n";

    my $first_fh = shift(@fhs);
    while (1) {
        my $line = <$first_fh>;
        chomp($line);
        my ($chr, $start, $end, $probe_id, $value) = split("t", $line);
        print OUT join("t", $chr, $start, $end, $probe_id, $value);
        foreach my $this_fh (@fhs) {
            my $line = <$this_fh>;
            chomp($line);
            my ($this_chr, $this_start, $this_end, $this_probe_id, $this_value) = split("t", $line);
            if (($this_chr ne $chr) or ($this_start != $start) or ($this_end != $end) or ($this_probe_id ne $probe_id)) {
                die "Files do not contain the same $chr-$start-$end-$probe_id lines\n";
            }
            print OUT $this_value;
        }
        print OUT "\n";
        if (eof($first_fh)) {
            last;
        }
    }

    foreach my $this_fh (@fhs) {
        close($this_fh);
    }
    close(OUT);
}

sub get_all_arrays_for_species {

    return $output_file;
}
my ($dbh, $species) = @_;  
my $arrays;

my $sth = $dbh->prepare("SELECT array_id, array_name FROM array WHERE 
species_name = ?"); 
$sth->execute($species);  
$arrays = $sth->fetchall_arrayref(); 
$sth->finish();

return $arrays;

sub get_probe_mapping_id {
    my ($dbh, $this_array_id, $assembly_name) = @_;  
    my $sql = "SELECT probe_mapping_id
                FROM probe_mapping_info
                JOIN assembly USING (assembly_id)
                WHERE array_id = $this_array_id
                AND assembly_name = '$assembly_name';";  
    my $probe_mapping_id = $dbh->selectrow_array($sql);  
    return($probe_mapping_id);
}

sub dump_array_bed_file {
    my ($dbh, $this_array_id, $assembly_name, $work_dir) = @_;  
    my $sql = "SELECT probe_mapping_id
                FROM probe_mapping_info
                JOIN assembly USING (assembly_id)
                WHERE array_id = $this_array_id
                AND assembly_name = '$assembly_name';";  
    my $probe_mapping_id = $dbh->selectrow_array($sql);

    $sql = "SELECT probe_id, chr_name, position FROM probe_mapping WHERE 
    probe_mapping_id = $probe_mapping_id";
    my $probes = $dbh->selectall_arrayref($sql);

    my $bed_file = "$work_dir/array_${this_array_id}.bed";  
    open(BED, ">$bed_file") or die "Cannot open temporary BED file <$bed_file>\n";
    foreach my $this_probe (sort_sort_probes @$probes) {
        print BED join("\t", $this_probe->[1], $this_probe->[2], $this_probe->[2]+1,
                        $this_probe->[0]),"\n";
    }  
    close(BED);

    return $bed_file;
}

sub_sort_probes {
    my $chr_a = $a->[1];  
    my $chr_b = $b->[1];  
    my $loc_a = $a->[2];  
    my $loc_b = $b->[2];  
    $chr_a =~ s/chr//;  
    $chr_b =~ s/chr//;
if ($chr_a eq $chr_b) {
    return $loc_a <=> $loc_b;
} elsif ($chr_a =~ /^\d/ and $chr_b =~ /^\d/) {
    return $chr_a <=> $chr_b;
} elsif ($chr_a =~ /^\d/) {
    return -1;
} elsif ($chr_b =~ /^\d/) {
    return 1;
} else {
    return $chr_a cmp $chr_b;
}
}

exit();