Learning from Features of Sets and Probabilities

Zoltán Szabó (Gatsby Unit, UCL)

Department of Computing
Imperial College London

March 9, 2016

Introduction

- Inference: uncertain inputs/probabilities.
- 2 motivating examples:
  1. games:
     - regression on distributions.
  2. sustainability:
     - regression on sampled distributions $=$ labelled bags.

Example-1: game

- Online gaming service created by Microsoft:

Example-1: game

- Online gaming service created by Microsoft:
- TrueSkill:
  - skill based ranking system for Xbox Live → game outcome.
  - Application: competitive matchmaking.
  - About 48M users.

Example-1: game

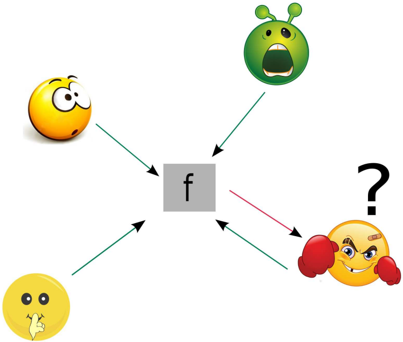
- Online gaming service created by Microsoft:

- TrueSkill:
  - skill based ranking system for Xbox Live $\rightarrow$ game outcome.
  - Application: competitive matchmaking.
  - About 48M users.
- Related fields: social recommender systems, search advertising.

Example-1: continued

Skill prediction:

- input: probabilities = beliefs of the players' skills,
- output: parameter = new belief.

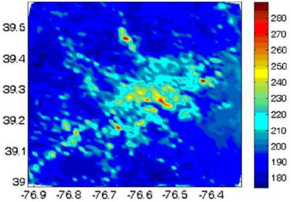


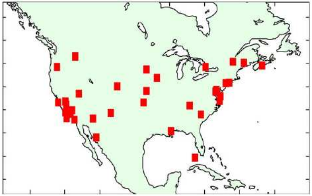
One-page summary: games

- Infer.NET:
  - small class of parametric models (e.g, normal).
- <u>Contribution</u>:
  - distribution regression phrasing:
    - flexibility: KJIT,
    - speed ⇐ random Fourier features.
  - exponentially tighter guarantee,
  - NIPS-2015 (spotlight - 3.65%).

- **Goal**: aerosol prediction = air pollution $\rightarrow$ climate.

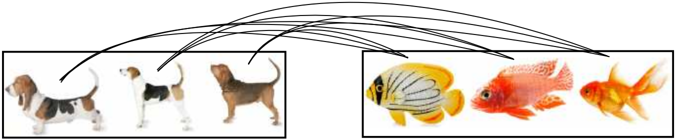
- Prediction using labelled bags:
  - bag := multi-spectral satellite measurements over an area,
  - label := local aerosol value.





Multi-instance learning:

- [Haussler, 1999, Gärtner et al., 2002] (set kernel):



- sensible methods in regression: few,
  1. restrictive technical conditions,
  2. super-high resolution satellite image: would be needed.

One-page summary: sustainability

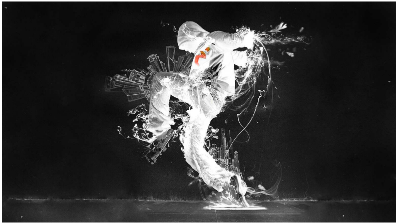
Contributions:

1. Practical: state-of-the-art accuracy (aerosol).
2. Theoretical:
   - General bags: graphs, time series, texts, ...
   - Consistency of set kernel in regression (17-year-old open problem).
   - How many samples/bag?

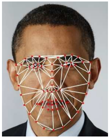
One-page summary: sustainability

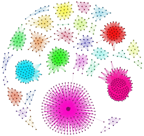
Contributions:

1. Practical: state-of-the-art accuracy (aerosol).
2. Theoretical:
   - General bags: graphs, time series, texts, . . .
   - Consistency of set kernel in regression (17-year-old open problem).
   - How many samples/bag?
   - AISTATS-2015 (oral – 6.11%) → JMLR in revision.



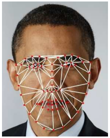
Objects in the bags

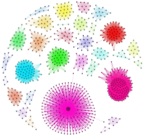




- Examples:
  - time-series modelling: user = set of time-series,
  - computer vision: image = collection of patch vectors,
  - NLP: corpus = bag of documents,
  - network analysis: group of people = bag of friendship graphs, . . .

Objects in the bags



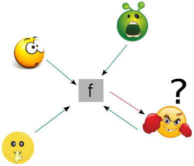


- Examples:
  - time-series modelling: user = set of time-series,
  - computer vision: image = collection of patch vectors,
  - NLP: corpus = bag of documents,
  - network analysis: group of people = bag of friendship graphs, ...
- Wider context (statistics): point estimation tasks.

Contents

1. Regression on distributions:
   - scaling up = Random Fourier features.
2. Regression on labelled bags.
3. Further applications.

Ridge regression on distributions

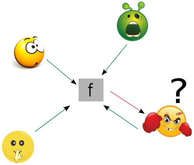


- Given: $\{(\underbrace{P_i}_{\text{non-standard}}, y_i)\}_{i=1}^{\ell}$, new P; $\hat{y} =?$

  Example:
  - ℓ: number of matches used for training.
  - P_i: distribution on skills.
- Learning from features of distributions:

$$w^* = \arg\min_w \frac{1}{\ell} \sum_{i=1}^{\ell} \left[\left\langle w, \underbrace{\psi(P_i)}_{\text{feature of } P_i} \right\rangle - y_i\right]^2 + \lambda \left\|w\right\|^2,$$

Ridge regression on distributions



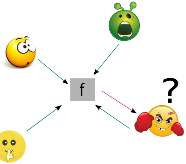
- Given: $\{(P_i, y_i)\}_{i=1}^{\ell}$, new P; $\hat{y} = ?$
- Learning from features of distributions:

$$w^* = \arg\min_w \frac{1}{\ell} \sum_{i=1}^{\ell} \left[\left\langle w, \psi(P_i)\right\rangle - y_i\right]^2 + \lambda \left\|w\right\|^2,$$

$$\hat{y}(P) = \langle w^*, \psi(P)\rangle = \mathbf{g}^T(\mathbf{K} + \lambda\ell\mathbf{I})^{-1}\mathbf{y}.$$

- Prediction: relies on $\mathbf{g} = [K(P_i, P)], \mathbf{K} = [\underbrace{K(P_i, P_j)}_{:=\left\langle\psi(P_i),\psi(P_j)\right\rangle}], \mathbf{y} = [y_i]$.

Ridge regression on distributions



- Given: $\{(P_i, y_i)\}_{i=1}^{\ell}$, new P; $\hat{y} = ?$
- Learning from features of distributions:

$$w^* = \arg\min_w \frac{1}{\ell} \sum_{i=1}^{\ell} \left[\left\langle w, \psi(P_i)\right\rangle - y_i\right]^2 + \lambda \left\| w \right\|^2,$$

$$\hat{y}(P) = \langle w^*, \psi(P) \rangle = \mathbf{g}^T(\mathbf{K} + \lambda\ell\mathbf{I})^{-1}\mathbf{y}.$$

- Prediction: relies on $\mathbf{g} = [K(P_i, P)], \mathbf{K} = [\underbrace{K(P_i, P_j)}_{:=\left\langle \psi(P_i), \psi(P_j)\right\rangle}], \mathbf{y} = [y_i]$.

Challenges

1. Inner product of distributions: $K(P_i, P_j) = ?$
2. Computation: $\mathcal{O}(\ell^3)$ – expensive.

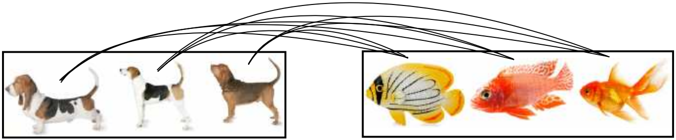
Similarity on bags and distributions

We define inner product on distributions $[K(P_i, P_j)]$:

1. Set kernel: $A = \{a_i\}_{i=1}^N$, $B = \{b_j\}_{j=1}^N$.

$$K(A,B) = \frac{1}{N^2}\sum_{i,j=1}^N k(a_i, b_j) = \left\langle \underbrace{\frac{1}{N}\sum_{i=1}^N \varphi(a_i)}_{\text{feature of bag } A}, \frac{1}{N}\sum_{j=1}^N \varphi(b_j) \right\rangle.$$

Remember:



Similarity on bags and distributions

We define inner product on distributions [$K(P_i, P_j)$]:

1. Set kernel: $A = \{a_i\}_{i=1}^N$, $B = \{b_j\}_{j=1}^N$.

$$K(A,B) = \frac{1}{N^2}\sum_{i,j=1}^N k(a_i, b_j) = \left\langle \underbrace{\frac{1}{N}\sum_{i=1}^N \varphi(a_i)}_{\text{feature of bag } A}, \frac{1}{N}\sum_{j=1}^N \varphi(b_j) \right\rangle.$$

2. Taking 'limit': $a \sim P, b \sim Q$

$$K(P,Q) = \mathbb{E}_{a,b} k(a,b) = \left\langle \underbrace{\mathbb{E}_a \varphi(a)}_{\text{feature of distribution } P =: \psi(P)}, \mathbb{E}_b \varphi(b) \right\rangle.$$

Example (Gaussian kernel): $k(\mathbf{a},\mathbf{b}) = e^{-\|\mathbf{a}-\mathbf{b}\|_2^2/(2\sigma^2)}$.

Random Fourier features reduce computational time

- Prediction on a new P:

$$\hat{y}(P) = \mathbf{g}^T(\mathbf{K} + \lambda\ell I)^{-1}\mathbf{y}, \quad K(P,Q) = \mathbb{E}_{\mathbf{a},\mathbf{b}} k(\mathbf{a},\mathbf{b}), \quad \mathbf{a} \sim P, \mathbf{b} \sim Q.$$

Scaling challenge! Computational time $= \mathcal{O}(\ell^3)$. ℓ can be huge!
Random Fourier features help: $\mathcal{O}(\ell m^2), m \ll \ell$.

Random Fourier features reduce computational time

- Prediction on a new P:

$$\hat{y}(P) = \mathbf{g}^T(\mathbf{K} + \lambda\ell I)^{-1}\mathbf{y}, \quad K(P,Q) = \mathbb{E}_{\mathbf{a},\mathbf{b}} k(\mathbf{a},\mathbf{b}), \quad \mathbf{a} \sim P, \mathbf{b} \sim Q.$$

Scaling challenge! Computational time $= \mathcal{O}(\ell^3)$. ℓ can be huge!
Random Fourier features help: $\mathcal{O}(\ell m^2), m \ll \ell$.

- For *any* k continuous and shift-invariant kernel

$$k(\mathbf{a},\mathbf{b}) = \mathbb{E}_{\boldsymbol{\omega}\sim\Lambda} \cos\left(\boldsymbol{\omega}^T(\mathbf{a}-\mathbf{b})\right), \ \Lambda : \text{given for many } k\text{-s!}$$

Random Fourier features reduce computational time

- Prediction on a new P:

$$\hat{y}(P) = \mathbf{g}^T(\mathbf{K} + \lambda\ell I)^{-1}\mathbf{y}, \quad K(P,Q) = \mathbb{E}_{\mathbf{a},\mathbf{b}} k(\mathbf{a},\mathbf{b}), \quad \mathbf{a} \sim P, \mathbf{b} \sim Q.$$

> Scaling challenge! Computational time $= \mathcal{O}(\ell^3)$. ℓ can be huge!
> Random Fourier features help: $\mathcal{O}(\ell m^2), m \ll \ell$.

- For *any* k continuous and shift-invariant kernel

$$k(\mathbf{a},\mathbf{b}) = \mathbb{E}_{\boldsymbol{\omega}\sim\Lambda} \cos\left(\boldsymbol{\omega}^T(\mathbf{a}-\mathbf{b})\right), \ \Lambda : \text{given for many } k\text{-s!}$$

$$\hat{k}(\mathbf{a},\mathbf{b}) = \frac{1}{m}\sum_{j=1}^{m} \cos\left(\boldsymbol{\omega}_j^T(\mathbf{a}-\mathbf{b})\right) \leftarrow \text{[Rahimi and Recht, 2007].}$$

- Error propagates nicely from $\hat{k}$ to $\hat{K}$.

Our result: exponentially tighter bound

- **Goal**: approximation error of $\hat{k}$ on domain $\mathcal{S}$ with m random Fourier features.
- Crude existing bound [Rahimi and Recht, 2007]:

$$\max_{\mathbf{a},\mathbf{b}\in\mathcal{S}} |k(\mathbf{a},\mathbf{b}) - \hat{k}(\mathbf{a},\mathbf{b})| = \mathcal{O}\left(\underbrace{|\mathcal{S}|}_{\text{linear}} \sqrt{\frac{\log m}{m}}\right).$$

- Our finite-sample guarantee implies $\mathcal{O}\left(\frac{\sqrt{\log|\mathcal{S}|}}{\sqrt{m}}\right)$.

Our bound proves that regression with RFF is practical.

Aerosol prediction = regression on labelled bags

- Game example: exact P_i, approximate K.
- Now: approximate P_i, exact K.

Aerosol prediction result ($100 \times RMSE$)

We perform on par with the state-of-the-art, hand-engineered method.

- Zhuang Wang, Liang Lan, Slobodan Vucetic. IEEE Transactions on Geoscience and Remote Sensing, 2012: $7.5 - 8.5$ ($\pm 0.1 - 0.6$):
  - hand-crafted features.
- Our prediction accuracy: 7.81 (± 1.64).
  - no expert knowledge.
- Code in ITE: #2 on mloss,

`https://bitbucket.org/szzoli/ite/`

Regression on labelled bags: $\hat{P}_i \rightarrow P_i$ performance?

- Given:
  - labelled bags: $\hat{\mathbf{z}} = \left\{\left(\hat{P}_i, y_i\right)\right\}_{i=1}^{\ell}$, $\hat{P}_i$: bag from P_i, $N := |\hat{P}_i|$.
  - test bag: $\hat{P}$.

Regression on labelled bags: $\hat{P}_i \to P_i$ performance?

- Given:
  - labelled bags: $\hat{\mathbf{z}} = \left\{\left(\hat{P}_i, y_i\right)\right\}_{i=1}^{\ell}$, $\hat{P}_i$: bag from P_i, $N := |\hat{P}_i|$.
  - test bag: $\hat{P}$.
- Estimator:

$$w_{\hat{\mathbf{z}}}^{\lambda} = \arg\min_{w} \frac{1}{\ell} \sum_{i=1}^{\ell} \left[\left\langle w, \underbrace{\psi(\hat{P}_i)}_{\text{feature of } \hat{P}_i} \right\rangle - y_i\right]^2 + \lambda \left\|w\right\|^2.$$

Regression on labelled bags: $\hat{P}_i \to P_i$ performance?

- Given:
  - labelled bags: $\hat{\mathbf{z}} = \left\{\left(\hat{P}_i, y_i\right)\right\}_{i=1}^{\ell}$, $\hat{P}_i$: bag from P_i, $N := |\hat{P}_i|$.
  - test bag: $\hat{P}$.
- Estimator:

$$w_{\hat{\mathbf{z}}}^{\lambda} = \arg\min_{w} \frac{1}{\ell} \sum_{i=1}^{\ell} \left[\left\langle w, \psi(\hat{P}_i)\right\rangle - y_i\right]^2 + \lambda \|w\|^2 .$$

- Quality of estimator, baseline:

$$\mathcal{R}(w) = \mathbb{E}_{(\psi(Q),y)\sim\rho}[\langle w, \psi(Q)\rangle - y]^2,$$

$$w_\rho = \text{best regressor}.$$

How many samples/bag to get the accuracy of w_ρ? Possible?

Our result: how many samples/bag

- Known: best/achieved rate

$$\mathcal{R}(w_{\mathbf{z}}^{\lambda}) - \mathcal{R}(w_{\rho}) = \mathcal{O}\left(\ell^{-\frac{bc}{bc+1}}\right),$$

b – size of the input space, c – smoothness of w_{ρ}.

Our result: how many samples/bag

- Known: best/achieved rate

$$\mathcal{R}(w_{\mathbf{z}}^{\lambda}) - \mathcal{R}(w_{\rho}) = \mathcal{O}\left(\ell^{-\frac{bc}{bc+1}}\right),$$

  b – size of the input space, c – smoothness of w_{ρ}.
- Let $N = \tilde{\mathcal{O}}(\ell^{a})$. N: size of the bags. ℓ: number of bags.

Our result

- If $2 \leq a$, then $w_{\hat{\mathbf{z}}}^{\lambda}$ attains the best achievable rate.

Our result: how many samples/bag

- Known: best/achieved rate

$$\mathcal{R}(w_{\mathbf{z}}^{\lambda}) - \mathcal{R}(w_{\rho}) = \mathcal{O}\left(\ell^{-\frac{bc}{bc+1}}\right),$$

  b – size of the input space, c – smoothness of w_{ρ}.
- Let $N = \tilde{\mathcal{O}}(\ell^{a})$. N: size of the bags. ℓ: number of bags.

Our result

- If $2 \leq a$, then $w_{\hat{\mathbf{z}}}^{\lambda}$ attains the best achievable rate.
- In fact, $a = \frac{b(c+1)}{bc+1} < 2$ is enough.
- Consequence: regression with set kernel is consistent.

+Applications, with Gatsby students

- Bayesian manifold learning [NIPS-2015]:
  - App.: climate data $\rightarrow$ weather station location.

+Applications, with Gatsby students

- Bayesian manifold learning [NIPS-2015]:
  - App.: climate data $\rightarrow$ weather station location.
- Fast, adaptive sampling method based on RFF [NIPS-2015]:
  - App.: approximate Bayesian computation, hyperparameter inference.

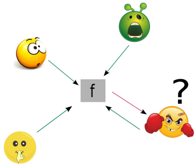
+Applications, with Gatsby students

- Bayesian manifold learning [NIPS-2015]:
  - App.: climate data → weather station location.
- Fast, adaptive sampling method based on RFF [NIPS-2015]:
  - App.: approximate Bayesian computation, hyperparameter inference.
- Interpretable 2-sample testing [ICML-2016 submission]:
  - App.:
    - random → smart features,
    - discriminative for doc. categories, emotions.
  - empirical process theory (VC subgraphs).

Summary

Regression on

- distributions:
    - random Fourier features.
    - exponentially tighter bounds.
- bags:
    - minimax optimality,
    - set kernel is consistent.



Several applications (with open source code).

Acknowledgments: This work was supported by the Gatsby Charitable Foundation.

Gärtner, T., Flach, P. A., Kowalczyk, A., and Smola, A. (2002).
Multi-instance kernels.
In *International Conference on Machine Learning (ICML)*, pages 179–186.

Haussler, D. (1999).
Convolution kernels on discrete structures.
Technical report, Department of Computer Science, University of California at Santa Cruz.
(`http://cbse.soe.ucsc.edu/sites/default/files/convolutions.pdf`).

Rahimi, A. and Recht, B. (2007).
Random features for large-scale kernel machines.
In *Neural Information Processing Systems (NIPS)*, pages 1177–1184.