Composite Goodness-of-fit Tests with Kernels

Oscar Key
University College London
oscar.key.20@ucl.ac.uk

Tamara Fernandez
Adolfo Ibañez University
tamara.fernandez@uai.cl

Arthur Gretton
University College London
arthur.gretton@gmail.com

François-Xavier Briol
University College London
f.briol@ucl.ac.uk

Abstract

Model misspecification can create significant challenges for the implementation of probabilistic models, and this has led to development of a range of inference methods which directly account for the issue. However, these method tend to lose efficiency and should only be used when the model is really misspecified. Unfortunately, there is a lack of generally applicable methods to test whether this is the case or not. One set of tools which can help are goodness-of-fit tests, which can test whether a dataset has been generated from a *fixed* distribution. Kernel-based tests have been developed to for this problem, and these are popular due to their flexibility, strong theoretical guarantees and ease of implementation in a wide range of scenarios. In this paper, we extend this line of work to the more challenging *composite goodness-of-fit* problem, where we are instead interested in whether the data comes from any distribution in some parametric family.

1 Introduction

One approach to answering the question of whether a model is misspecified is goodness-of-fit testing (Lehmann et al. 2005, Chapter 14). Given a *fixed* distribution $\mathbb{P}$ and observations $x_1, \ldots, x_n$ generated from an unknown distribution $\mathbb{Q}$, goodness-of-fit tests compare the null hypothesis $H_0 : \mathbb{P} = \mathbb{Q}$ against the alternative hypothesis $H_1 : \mathbb{P} \neq \mathbb{Q}$. Tests using the kernel Stein discrepancy (KSD) as test statistic are popular for this task because they can be applied to a wide-range of data types and can accommodate models with unnormalised densities (Liu et al. 2016; Chwialkowski et al. 2016).

Although these tests are very useful in practice, we will often be interested in answering the more complex question of whether our data was generated by *any element of some parametric family* of distributions $\{\mathbb{P}_\theta\}_{\theta\in\Theta}$ with parameter space Θ. Specifically, the null hypothesis (corresponding to a well-specified model) is $H_0^C : \exists\theta^* \in \Theta$ such that $\mathbb{P}_{\theta^*} = \mathbb{Q}$, and the alternative (corresponding to a misspecified model) is $H_1^C : \mathbb{Q} \notin \{\mathbb{P}_\theta\}_{\theta\in\Theta}$. This type of test is known as a *composite goodness-of-fit test*, and can be much more challenging to construct since θ^* is usually unknown.

Our paper fills an important gap in the literature by proposing the first set of *kernel-based composite hypothesis tests* applicable to a wide range of parametric models. These are in contrast to previously introduced composite tests which are limited to very specific parametric families (Kellner et al. 2019; Fernandez et al. 2020). To devise these new tests, we make use of recently developed minimum distance estimators based on the KSD (Barp et al. 2019). A key challenge is that the dataset is used twice, both to estimate the parameter and the test statistic, and this is done without splitting it into estimation and test sets. To achieve the correct level, the test must take account of this dependence, both via a more in-depth theoretical analysis and by using a suitable method for approximating

NeurIPS 2021 Workshop Your Model Is Wrong: Robustness and Misspecification in Probabilistic Modeling.

the threshold. In this initial version of the work, we only consider the second aspect by using the parametric bootstrap (Stute et al. 1993) to achieve the correct level without data-splitting.

2 Background: Kernel Stein Discrepancies for Testing and Estimation

We will now briefly review the use of KSD for testing and estimation. Denote by $\mathcal{X}$ the data space and $\mathcal{P}(\mathcal{X})$ the set of all Borel distributions on $\mathcal{X}$. For simplicity, we will focus on $\mathcal{X} = \mathbb{R}^d$, but note that KSDs have been developed for other data types (Yang et al. 2018; Yang et al. 2019; Kanagawa et al. 2020; Fernández et al. 2019; Fernandez et al. 2020; Xu et al. 2020; Xu et al. 2021). The KSD is a function $\mathcal{P}(\mathcal{X}) \times \mathcal{P}(\mathcal{X}) \to \mathbb{R}^+$ which measures the similarity between two distributions $\mathbb{P}, \mathbb{Q} \in \mathcal{P}(\mathcal{X})$. Although it is not a probability metric, it is closely connected to the class of integral pseudo-probability metrics (IPMs) (Muller 1997), which measure similarity as follows:

$$D_{\mathcal{F}}(\mathbb{P}, \mathbb{Q}) = \sup_{f \in \mathcal{F}} |\mathbb{E}_{X \sim \mathbb{P}}[f(X)] - \mathbb{E}_{Y \sim \mathbb{Q}}[f(Y)]| \tag{1}$$

Let $\mathcal{H}_k$ be the reproducing kernel Hilbert space (RKHS) associated to the symmetric positive definite kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ (Berlinet et al. 2004) and let $\mathcal{H}_k^d = \mathcal{H}_k \times \ldots \times \mathcal{H}_k$ denote the d-dimensional tensor product of $\mathcal{H}_k$. The *Langevin kernel Stein discrepancy* (Oates et al. 2016; Chwialkowski et al. 2016; Liu et al. 2016; Gorham et al. 2015) is obtained by considering an IPM with

$$\mathcal{F}_{\mathrm{KSD}} = \{\mathcal{S}_{\mathbb{P}}[f] \mid \|f\|_{\mathcal{H}_k^d} \leq 1\} \text{ where } \mathcal{S}_{\mathbb{P}}[f](x) = f(x) \cdot \nabla \log p(x) + \nabla \cdot f(x).$$

The operator $\mathcal{S}_{\mathbb{P}}$ is called the Langevin Stein operator and p is the Lebesgue density of $\mathbb{P}$. Using $\mathcal{F}_{\mathrm{KSD}}$, Equation (1) simplifies to:

$$\mathrm{KSD}(\mathbb{P}, \mathbb{Q}) := \sup_{\|f\|_{\mathcal{H}_k^d} \leq 1} |\mathbb{E}_{X \sim \mathbb{Q}}[\mathcal{S}_{\mathbb{P}}[f](X)]| = \sqrt{\mathbb{E}_{X,Y \sim \mathbb{Q}}[h(X,Y)]}$$

$$\begin{aligned} \text{where } h(x,y) = k(x,y) \nabla \log p(x) \cdot \nabla \log p(y) + \nabla_x \cdot \nabla_y k(x,y) \\ + \nabla \log p(x) \cdot \nabla_y k(x,y) + \nabla \log p(y) \cdot \nabla_x k(x,y). \end{aligned}$$

Let $\mathbb{Q}_n = \frac{1}{n} \sum_{i=1}^n \delta(x_i)$ where $\delta(x)$ denotes a Dirac measure at $x \in \mathcal{X}$. The KSD^2 can be straightforwardly computed in this case using a single V-statistic $\mathrm{KSD}^2(\mathbb{P}, \mathbb{Q}_n) = \frac{1}{n^2} \sum_{i,j=1}^n h(x_i, x_j)$ at a cost of $O(n^2 d)$. Under mild regularity conditions, the KSD is a statistical divergence meaning that $\mathrm{KSD}(\mathbb{P}, \mathbb{Q}) = 0$ if and only if $\mathbb{P} = \mathbb{Q}$; see Chwialkowski et al. (2016, Theorem 2.1) & Barp et al. (2019, Proposition 1). The KSD is convenient in the context of unnormalised models since it depends on $\mathbb{P}$ only via $\nabla \log p$, which can be evaluated without knowledge of the normalisation constant of p. We now briefly recall how it can be used for goodness-of-fit testing and estimation.

Goodness-of-fit testing with KSD In goodness-of-fit testing, we would like to test $H_0 : \mathbb{P} = \mathbb{Q}$ against $H_1 : \mathbb{P} \neq \mathbb{Q}$. A natural approach is to compute $\mathrm{KSD}^2(\mathbb{P}, \mathbb{Q})$ and check whether this quantity is zero (i.e. H_0 holds) or not (i.e. H_1 holds) (Liu et al. 2016; Chwialkowski et al. 2016). Of course, since we only have access to $x_1, \ldots, x_n$ instead of $\mathbb{Q}$, this idealised procedure is replaced by the evaluation of $\mathrm{KSD}^2(\mathbb{P}, \mathbb{Q}_n)$. The question then becomes whether or not this quantity is further away from zero than expected under H_0 given the sampling error associated with a dataset of size n.

To determine whether H_0 should be rejected, we need to select an appropriate threshold $c_\alpha \in \mathbb{R}$, which will depend on the level of the test $\alpha \in [0, 1]$. More precisely, c_α should be set to the $(1-\alpha)$-quantile of the distribution of $\mathrm{KSD}^2(\mathbb{P}, \mathbb{Q}_n)$ under H_0. This distribution will usually be unknown a-priori, but can be approximated using a bootstrap method. A common example is the wild bootstrap (Shao 2010; Leucht et al. 2013), which was specialised for kernel tests by Chwialkowski et al. (2014).

Minimum distance estimation with KSD As shown by Barp et al. 2019, the KSD can also be used for parameter estimation through *minimum distance estimation* (Wolfowitz 1957). Given a parametric family $\{\mathbb{P}_\theta\}_{\theta \in \Theta}$ indexed by $\Theta \subseteq \mathbb{R}^p$. Given $x_1, \ldots, x_n \sim \mathbb{Q}$, a natural estimator is

$$\hat{\theta}_n := \arg\min_{\theta \in \Theta} \mathrm{KSD}^2(\mathbb{P}_\theta, \mathbb{Q}_n).$$

Under regularity conditions, the estimator approaches $\theta^* := \arg\min_{\theta \in \Theta} \mathrm{KSD}^2(\mathbb{P}_\theta, \mathbb{Q})$ as $n \to \infty$. The use of KSD was later extended by Grathwohl et al. (2020), Matsubara et al. (2021), and Gong et al. (2021). These estimators are also closely related to score-matching estimators (Hyvärinen 2006); see Barp et al. (2019, Theorem 2) for details.

Algorithm 1: Wild bootstrap test

Input: $\mathbb{P}_{\hat{\theta}}$, $\mathbb{Q}_n$, α, b
for $k \in \{1, \ldots, b\}$ **do**
  $w^{(k)} \sim \text{Rademacher}(n)$;
  $\Delta^{(k)} = \frac{1}{n} \sum_{i,j=1}^{n} w_i^{(k)} w_j^{(k)} h(x_i, x_j)$
$c_\alpha = \text{quantile}(\{\Delta^{(1)}, \ldots, \Delta^{(b)}\}, 1-\alpha)$;
$\Delta = n \, \text{KSD}^2(\mathbb{P}_{\hat{\theta}}, \mathbb{Q}_n)$;
if $\Delta \geq c_\alpha$, reject the null, **else** do not reject.

Algorithm 2: Parametric bootstrap test

Input: $\mathbb{P}_\theta$, $\mathbb{Q}_n$, $\hat{\theta}_n$, α, b
for $k \in \{1, \ldots, b\}$ **do**
  $\mathbb{Q}_n^{(k)} = \frac{1}{n} \sum_{i=1}^{n} \delta_{y_i^{(k)}}$, $\{y_i^{(k)}\}_{i=1}^{n} \sim \mathbb{P}_{\hat{\theta}_n}$;
  $\hat{\theta}_n^{(k)} = \arg\min_{\theta \in \Theta} \text{KSD}^2(\mathbb{P}_\theta, \mathbb{Q}_n^{(k)})$;
  $\Delta^{(k)} = n \, \text{KSD}^2(\mathbb{P}_{\hat{\theta}_n^{(k)}}, \mathbb{Q}_n^{(k)})$;
$c_\alpha = \text{quantile}(\{\Delta^{(1)}, \ldots, \Delta^{(b)}\}, 1-\alpha)$;
$\Delta = n \, \text{KSD}^2(\mathbb{P}_{\hat{\theta}_n}, \mathbb{Q}_n)$;
if $\Delta \geq c_\alpha$, reject the null, **else** do not reject.

3 Methodology

We now consider a novel composite goodness-of-fit test where both estimation and testing is based on the KSD^2 with some fixed kernel k. The new test contains the two following stages:

Stage 1 (Estimation): $\hat{\theta}_n = \arg\min_{\theta \in \Theta} \text{KSD}^2(\mathbb{P}_\theta, \mathbb{Q}_n)$.

Stage 2 (Testing): reject if $n \, \text{KSD}^2(\mathbb{P}_{\hat{\theta}_n}, \mathbb{Q}_n) \geq c_\alpha$, using one of Algorithm 1 or Algorithm 2.

Note that we could possibly replace stage 1 with another estimator such as maximum likelihood estimation, but this would require knowledge of any normalisation constant of the likelihood which may not be possible in many cases. To implement Stage 2, we require a bootstrap algorithm to estimate the threshold c_α. A natural first approach is to use the wild bootstrap, since it is the current gold-standard for kernel-based goodness-of-fit tests. Algorithm 1 gives the implemention of a composite test using the wild bootstrap, where α is the desired test level and b is the number of bootstrap samples. Note that in our implementation the choice of sampling $w^{(k)}$ from a Rademacher distribution assumes that $x_1, \ldots, x_n$ are independent.

If we make the unrealistic assumption that $\hat{\theta}_n = \theta^*$, then we are back in the setting considered by Chwialkowski et al. (2016) and Liu et al. (2016) and the wild bootstrap works as follows. Under H_0^C, as $n \to \infty$, $\Delta^{(k)}$ and Δ converge to the same distribution, thus c_α will converge to the $(1-\alpha)$-quantile of Δ (Chwialkowski et al. 2014, Theorem 1, 2), and the test will reject H_0^C with probability α, as desired. Under H_1^C, Δ will diverge while $\Delta^{(k)}$ will converge to some distribution, hence the probability that $\Delta > c_\alpha$) goes to 1 as $n \to \infty$.

However, it is clear that we cannot assume that $\hat{\theta}_n = \theta^*$ in the finite data case. In fact, using the wild bootstrap in this fashion for composite tests may result in an incorrect type I error rate under H_0^C, and lost power under H_1^C. This is because the estimation stage of the composite test introduces a second source of error which this approach does not take account of when computing c_α. The two sources of error that the test encounters are as follows. Recall that in an idealised setting we would use $\text{KSD}^2(\mathbb{P}_{\theta^*}, \mathbb{Q})$ as the test statistic. The first source of error is introduced because we must estimate this statistic with $\text{KSD}^2(\mathbb{P}_{\theta^*}, \mathbb{Q}_n)$, as we only have access to a sample from $\mathbb{Q}$. This source of error also occurs in non-composite tests, and is accounted for correctly by the wild bootstrap. The second source of error is specific to composite tests, and occurs because we must further estimate $\text{KSD}^2(\mathbb{P}_{\theta^*}, \mathbb{Q}_n)$ with $\text{KSD}^2(\mathbb{P}_{\hat{\theta}_n}, \mathbb{Q}_n)$, as we do not have access to θ^*. Algorithm 1 fixes the parameter estimate $\hat{\theta}_n$ and then applies the bootstrap, thus failing to take account of the error in in $\hat{\theta}_n$ and potentially computing an incorrect threshold. We can also view Algorithm 1 as a non-composite goodness-of-fit test against the wrong null hypothesis. By estimating the parameter and then applying the test, the test is not evaluating $H_0 : \mathbb{P}_{\theta^*} = \mathbb{Q}$, but instead $H_0' : \mathbb{P}_{\hat{\theta}_n} = \mathbb{Q}$.

Figure 1 demonstrates the impact of ignoring this error. It shows the power of our test with $H_0^C : \mathbb{P}_\theta = \mathcal{N}(\mu, 1)$ for $\mu \in \mathbb{R}$ and $\mathbb{Q} = \mathcal{N}(1.3, \sigma^2)$, as σ^2 varies, see Appendix B for details. We can see that the wild bootstrap test has lower power for most values of σ^2, including when $\sigma^2 = 1$, thus H_0^C holds, and it corresponds to the type I error. It also demonstrates that the error depends on the size of n. The estimator is consistent: as n becomes large we expect the estimate of the parameter

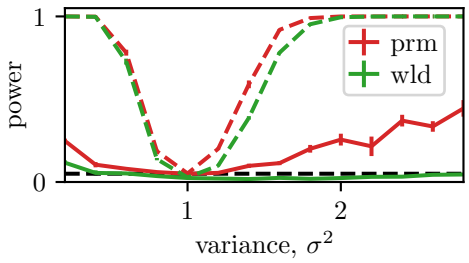

Figure 1: Power of of the test when using the wild and parametric bootstraps for $n = 10$ (solid) and $n = 200$ (dashed). The dashed horizontal line shows the test level, $\alpha = 0.05$. The error bars show one standard error over 4 random seeds.

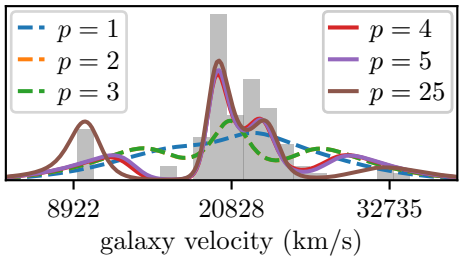

Figure 2: Goodness-of-fit of a kernel exponential family model with increasing p. The grey histogram shows the dataset, and the lines the modelled density. Dashed lines indicate that the test rejected H_0^C, and solid that it did not.

to converge on the true value, thus minimizing the impact of the error. However, for smaller n it is necessary to use an alternative approach which is able to take account of the additional error.

To take account of both types of error we apply the parametric bootstrap described in Algorithm 2. This bootstrap approximates the distribution of the test statistic under H_0^C by repeatedly resampling the observations and re-estimating the parameter, thus taking account of the error in the estimate of the parameter. In comparison to the wild bootstrap, this is substantially more computationally intensive because it requires repeatedly computing the kernel matrix on fresh data, whereas the wild bootstrap only computes the kernel matrix once. However, when n is large and this extra computation is likely to be an issue, the size of the estimation error should also be reduced as the estimator converges on the true value of the parameter. Thus, under this high data regime, the wild and parametric bootstraps may achieve comparable powers, and it may be reasonable to use the cheaper wild bootstrap. In the setting considered in Figure 1 we find that this is the case, with the parametric bootstrap performing substantially better than the wild bootstrap for $n = 10$, but comparably for $n = 200$.

4 Illustration: The Kernel Exponential Family Model

We apply our method to a density estimation task from Matsubara et al. (2021), to test whether a robust method was really necessary. The model is in the kernel exponential family $p_\theta(x) \propto q(x) \exp(f(x))$, where f is assumed to belong to some RKHS but approximated by a finite sum of basis functions, and $q(x) = \mathcal{N}(0, 3^2)$ is a reference density. Specifically, we follow Steinwart et al. (2006) and consider $f(x) = \sum_{i=1}^{p} \theta_i \phi_i(x)$, with $\phi_i(x) = x^i / \sqrt{i!} \exp\left(-x^2/2\right), \theta \in \mathbb{R}^p$. The dataset is comprised of the velocities of 82 galaxies (Postman et al. 1986; Roeder 1990). An open question is how large p needs to be for the model to have enough capacity to fit the data. Matsubara et al. (2021) set $p = 25$, and we test this assumption. See Appendix B for the full experiment configuration. Figure 2 shows the fit of the model for increasing values of p, and whether the test rejected the null hypothesis. We find that the test rejects H_0^C for $p = 1, 2, 3$, but does not reject for $p = 4, 5, 25$. This suggests that $p = 25$ is a suitable choice, though it would be reasonable to use a smaller value of p which would decrease the computational cost of inference.

5 Conclusion

In this initial work, we introduced a flexible composite goodness-of-fit test which requires only an unnormalized density. In the full version of this paper we will consider a general framework for composite goodness-of-fit tests, which will also include a tests based on the maximum mean discrepancy. This will allow us to tackle a much wider set of problems, including composite testing for generative models with no computable density function. We will also include theoretical results showing that the test statistic converges under H_0^C, the test is consistent under H_1^C, and that the wild and parametric bootstraps behave appropriately.

References

Barp, A., Briol, F.-X., Duncan, A., Girolami, M., and Mackey, L. (2019). "Minimum Stein Discrepancy Estimators". In: *Advances in Neural Information Processing Systems*. Vol. 32.

Berlinet, A. and Thomas-Agnan, C. (2004). *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. New York.

Chwialkowski, K., Sejdinovic, D., and Gretton, A. (2014). "A Wild Bootstrap for Degenerate Kernel Tests". In: *NIPS*, pp. 3608–3616.

Chwialkowski, K., Strathmann, H., and Gretton, A. (2016). "A Kernel Test of Goodness of Fit". In: *International Conference on Machine Learning*. PMLR, pp. 2606–2615.

Fernandez, T., Rivera, N., Xu, W., and Gretton, A. (2020). "Kernelized Stein Discrepancy Tests of Goodness-of-Fit for Time-to-Event Data". In: *International Conference on Machine Learning*. PMLR, pp. 3112–3122.

Fernández, T. and Gretton, A. (2019). "A Maximum-Mean-Discrepancy Goodness-of-Fit Test for Censored Data". In: *Artificial Intelligence and Statistics*.

Gong, W., Li, Y., and Hernández-Lobato, J. M. (2021). "Sliced Kernelized Stein Discrepancy". In: *International Conference on Learning Representations*.

Gorham, J. and Mackey, L. (2015). "Measuring Sample Quality with Steins Method". In: *Advances in Neural Information Processing Systems*. Vol. 28.

Grathwohl, W., Wang, K.-C., Jacobsen, J.-H., Duvenaud, D., and Zemel, R. (2020). "Learning the Stein Discrepancy for Training and Evaluating Energy-Based Models Without Sampling". In: *International Conference on Machine Learning*, pp. 9485–9499.

Hyvärinen, A. (2006). "Estimation of Non-Normalized Statistical Models by Score Matching". In: *Journal of Machine Learning Research* 6, pp. 695–708.

Kanagawa, H., Jitkrittum, W., Mackey, L., Fukumizu, K., and Gretton, A. (2020). *A Kernel Stein Test for Comparing Latent Variable Models*. arXiv: 1907.00586.

Kellner, J. and Celisse, A. (2019). "A One-Sample Test for Normality with Kernel Methods". In: *Bernoulli* 25.3, pp. 1816–1837.

Lehmann, E. L. and Romano, J. P. (2005). *Testing Statistical Hypotheses*. 3rd ed. Springer Texts in Statistics.

Leucht, A. and Neumann, M. H. (2013). "Dependent Wild Bootstrap for Degenerate U- and V-Statistics". In: *Journal of Multivariate Analysis* 117, pp. 257–280.

Liu, Q., Lee, J., and Jordan, M. (–June 22, 2016). "A Kernelized Stein Discrepancy for Goodness-of-Fit Tests". In: *Proceedings of the 33rd International Conference on Machine Learning*. Vol. 48. Proceedings of Machine Learning Research, pp. 276–284.

Matsubara, T., Knoblauch, J., Briol, F.-X., and Oates, C. J. (2021). *Robust Generalised Bayesian Inference for Intractable Likelihoods*. arXiv: 2104.07359.

Muller, A. (1997). "Integral Probability Metrics and Their Generating Classes of Functions". In: *Advances in Applied Probability* 29.2, pp. 429–443.

Oates, C. and Girolami, M. (–May 11, 2016). "Control Functionals for Quasi-Monte Carlo Integration". In: *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*. Vol. 51. Proceedings of Machine Learning Research, pp. 56–65.

Postman, M., Huchra, J. P., and Geller, M. J. (1986). "Probes of Large-Scale Structure in the Corona Borealis Region." In: *The Astronomical Journal* 92, pp. 1238–1247.

Roeder, K. (1990). "Density Estimation with Confidence Sets Exemplified by Superclusters and Voids in the Galaxies". In: *Journal of the American Statistical Association* 85.411, pp. 617–624.

Shao, X. (2010). "The Dependent Wild Bootstrap". In: *Journal of the American Statistical Association* 105.489, pp. 218–235.

Steinwart, I., Hush, D., and Scovel, C. (2006). "An Explicit Description of the Reproducing Kernel Hilbert Spaces of Gaussian RBF Kernels". In: *IEEE Transactions on Information Theory* 52.10, pp. 4635–4643.

Stute, W., Manteiga, W. G., and Quindimil, M. P. (1993). "Bootstrap Based Goodness-of-Fit-Tests". In: *Metrika* 40.1, pp. 243–256.

Wolfowitz, J. (1957). "The Minimum Distance Method". In: *The Annals of Mathematical Statistics* 28.1, pp. 75–88.

Xu, W. and Matsuda, T. (2020). "A Stein Goodness-of-Fit Test for Directional Distributions". In: *Artificial Intelligence and Statistics*.

Xu, W. and Reinert, G. (2021). "A Stein Goodness of Fit Test for Exponential Random Graph Models". In: *Artificial Intelligence and Statistics*.

Yang, J., Liu, Q., Rao, V., and Neville, J. (2018). "Goodness-of-Fit Testing for Discrete Distributions via Stein Discrepancy". In: *International Conference on Machine Learning*, pp. 5561–5570.
Yang, J., Rao, V., and Neville, J. (2019). "A Stein-Papangelou Goodness-of-Fit Test for Point Processes". In: *International Conference on Artificial Intelligence and Statistics*, pp. 226–235.

A Closed-form expression for KSD estimator

For exponential family models, such as the Gaussian and kernel exponential families considered in this paper, there is a closed-form expression for the KSD estimator. Let the density of a model in the exponential family be

$$p_\theta(x) = \exp(\eta(\theta) \cdot t(x) - a(\theta) + b(x)) \,,$$

with $\eta : \Theta \to \mathbb{R}^k$ an invertible map, $t : \mathbb{R}^d \to \mathbb{R}^k$ any sufficient statistic for some $k \in \mathbb{N}_1$, $a : \Theta \to \mathbb{R}$ and $b : \mathbb{R}^d \to \mathbb{R}$. Then the KSD estimator of theta is given by

$$\hat{\theta}_n := \arg\min \mathrm{KSD}^2(\mathbb{P}_\theta \,\|\, \mathbb{Q}_n) = \eta^{-1}\left(-\frac{1}{2}\Lambda_n^{-1} v_n\right) \,,$$

where $\Lambda_n \in \mathbb{R}^{k\times k}$ and $\nu_n \in \mathbb{R}^k$ are defined as $\Lambda_n := \frac{1}{n^2}\sum_{i=1}^n \sum_{j=1}^n \Lambda(x_i, x_j)$ and $\nu_n := \frac{1}{n^2}\sum_{i=1}^n \sum_{j=1}^n \nu(x_i, x_j)$. These are based on functions $\Lambda : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^{k\times k}$ and $\nu : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^k$ which depend on the specific model, defined as

$$\Lambda(x, x') := k(x, x')\nabla t(x) \nabla t(x')^\top$$
$$\nu(x, x') := k(x, x')\nabla b(x)\nabla t(x')^\top + \nabla t(x)\nabla_{x'} k(x, x') + k(x, x')\nabla b(x')\nabla t(x)^\top + \nabla t(x')\nabla_x k(x', x) \,.$$

The detailed derivation of this result can be found in Barp et al. (2019, Appendix D3) or Matsubara et al. (2021).

B Experiment details

B.1 Figure 1

For each datapoint in the plot, the power is computed as follows:

1. For four random seeds:
   (a) For 2000 repeats:
      i. Sample n points from $\mathbb{Q} = \mathcal{N}(1.3, \sigma^2)$
      ii. Run the test
   (b) Compute the power, the fraction of repeats during which the null hypothesis is rejected
2. Compute the mean and standard error over the seeds

We find the standard error to be sufficiently small that it is not visible in the figure.

Estimator We estimate only the mean, which we denote by θ. The variance, σ^2, is known and specified by the null hypothesis. We use the closed-form KSD estimator as defined above, with

$$\eta(\theta) = \frac{\mu}{\sigma^2}, \qquad t(x) = x, \qquad b(x) = \log \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{x^2}{2\sigma^2} \,.$$

Kernel Gaussian kernel:

$$k(x_1, x_2) = \exp\left(-\frac{\|x_1 - x_2\|_2^2}{2l^2}\right) \,.$$

We choose $l = 0.7$.

Bootstrap and test The wild bootstrap (Algorithm 1) is configured with $b = 500$. The parametric bootstrap (Algorithm 2) is configured with $b = 300$. We set $\alpha = 0.05$ in both cases.

B.2 Figure 2

Data normalisation Following Matsubara et al. (2021), we normalize the dataset by

$$y'_i = \frac{y_i - \text{mean}(y_1, \ldots, y_n)}{\frac{1}{2}\,\text{std}(y_1, \ldots, y_n)} \qquad \forall i \in \{1, \ldots, n\}\,.$$

Estimator We use the closed-form KSD estimator as defined above, with

$$\eta(\theta) = \theta, \qquad t(x) = (\phi_1(x), \ldots, \phi_p(x))^\top, \qquad b(x) = \log q_0(x)\,.$$

Kernel IMQ kernel:

$$k(x_1, x_2) = \left(1 + \frac{\|x_1 - x_2\|_2^2}{l^2}\right)^{-\frac{1}{2}}\,.$$

We select l using the median heuristic,

$$l_{\text{med}} = \sqrt{\text{median}(d_{ij}/2)} \text{ with } d_{ij} = \|y_i - y_j\|_2^2 \text{ for all } i, j = 1, \ldots, n\,,$$

which in this case results in $l \approx 0.9$.

Bootstrap and test Parametric bootstrap (Algorithm 2), with $b = 400$ and $\alpha = 0.05$.