MEMe: An Accurate Maximum Entropy Method for Efficient Approximations in Large-Scale Machine Learning

Diego Granziol, Binxin Ru, Stefan Zohren, Xiaowen Dong, Michael Osborbe, Stephen Roberts
Oxford University
...
Address line

Abstract

We develop a novel robust Maximum Entropy algorithm, capable of dealing with hundreds of moments, allowing for computationally efficient approximations which are shown to be significantly better than existing approaches. The usefulness of the approach is showcased across a set of applications. In particular we highlight its effectiveness in: Determinantal Point Processes; spectral decompositions of large sparse graphs; as well as information-theoretic Bayesian Optimisation.

Introduction

Algorithmic scalability is a keystone in the realm of modern machine learning. Making high quality inference, on large, feature rich datasets under a constrained computational budget is arguably the primary goal of the learning community. We develop a novel, robust Maximum Entropy algorithm using Newton conjugate gradient and Hessian information with a Legendre/Chebyshev basis, as opposed to power moments. Our algorithm is stable for a large number of moments, surpassing the $m \approx 8$ limit of previous MaxEnt algorithms Granziol and Roberts (2017b); Bandyopadhyay et al. (2005); Mead and Papanicolaou (1984). We show that the ability to handle more moment information, which can be calculated cheaply either analytically or with the use of stochastic trace estimation, leads to significantly enhanced performance. We showcase the utility of the algorithm by applying it to improve the scalability of Determinantal Point Processes, Learning the number of clusters in large graphs and Bayesian Optimisation. We further derive bounds on the error of the estimated underlying densities and establish a link between Maximum Entropy methods and constrained variational inference.

The Method of Maximum Entropy

The method of maximum entropy, hereafter referred to as *MaxEnt* (Pressé et al., 2013) is a procedure for generating the most conservative estimate[1] of a probability distribution possible with the given information, the most non-committal with regard to missing information (Jaynes, 1957a). Intuitively, on a bounded domain, the most conservative distribution, the distribution of maximum entropy, is the one that assigns equal probability to all the accessible states. Hence, the method of maximum entropy can be thought of choosing the flattest, or most equiprobable distribution, satisfying the given constraints. To determine the spectral density $p(\lambda)$ using MaxEnt, we maximise the entropic functional

$$S = -\int p(\lambda)\log p(\lambda)d\lambda - \sum_i \alpha_i \left[\int p(\lambda)\lambda^i d\lambda - \mu_i\right] \tag{1}$$

with respect to $p(\lambda)$, where $\mathbb{E}_p[\lambda^i] = \mu_i$ are the power moment constraints on the spectral density. The first term in equation (1) is referred to as the Boltzmann-Shannon-Gibbs (BSG) entropy, which has been applied in multiple fields, ranging from condensed matter physics (Giffin, Cafaro, and Ali, 2016), finance (Neri and Schneider, 2012), under the axioms of consistency, uniqueness, invariance under coordinate transformations, sub-set and system independence, it can be proved that for constraints in the form of expected values, drawing self-consistent inferences requires maximising the entropy (Shore and Johnson, 1980; Pressé et al., 2013).

Variational Inference as a special case of MaxEnt

Variational Methods MacKay (2003); Fox and Roberts (2012) in machine learning pose the problem of intractable density estimation from the application of Bayes' rule as a functional optimization problem,

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)} \approx q(z), \tag{2}$$

and finding the appropriate $q(z)$. Typically, whilst the functional form of $p(x|z)$ is known, calculating $p(x) = \int p(x|z)p(z)dz$ is intractable. Using Jensen's inequality we can show that,

$$\log p(x) \geq \mathbb{E}_q[\log p(x,z)] - \mathbb{E}_q[\log q(z)].^2 \tag{3}$$

It can be shown that the reverse KL divergence between the posterior and the variational distribution, $\mathbb{D}_{kl}(q|p)$, can be

[1] With respect to the uniform distribution.

[2] Where the R.H.S is known as the evidence lower bound (ELBO).

written as,

$$\log p(x) = \mathbb{E}_q[\log p(x,z)] - \mathbb{E}_q[\log q(z)] + \mathbb{D}_{kl}(q|p). \quad (4)$$

Hence maximising the evidence lower bound is equivalent to minimising the reverse KL divergence between p and q.

$$\frac{\partial}{\partial Q_i(x_i)}\Bigg\{ - \mathbb{D}_{kl}[Q_i(x_i|Q^*(x_i))] - \lambda_i\Bigg(\int Q_i dx_i - 1\Bigg)\Bigg\}, \quad (5)$$

leading to a Gibbs' distribution and an iterative update equation.

Log Determinant as a Variational Inference problem

We consider minimizing the reverse KL divergence between our surrogate posterior $q(\lambda)$ and our prior $p_0(\lambda)$ on the eigenspectrum,

$$\mathcal{D}_{kl}(q|p_0) = -H(q) - \int_0^1 q(\lambda)\log p_0(\lambda)d\lambda, \quad (6)$$

such that the normalization and moment constraints are satisfied. Here $H(q)$ denotes the differential entropy of the density q.

By the theory of Lagrangian duality, the convexity of the KL divergence and the affine nature of the moment constraints, we maximise the dual Boyd and Vandenberghe (2009),

$$-H(q) - \int q(\lambda)\log p_0(\lambda)d\lambda - \sum_{i=0}^{m}\alpha_i\Bigg(\int_0^1 q(\lambda)\lambda^i d\lambda - \mu_i\Bigg), \quad (7)$$

or alternatively we minimise

$$H(q) + \int q(\lambda)\log p_0(\lambda)d\lambda - \sum_{i=0}^{m}\alpha_i\Bigg(\int_0^1 q(\lambda)\lambda^i d\lambda - \mu_i\Bigg). \quad (8)$$

Link to Information Physics

In the field of information physics the minimization of Equation (8) is known as the method of relative entropy Caticha (2012). It can be derived as the unique functional satisfying the axioms of,

1. **Locality:** local information has local effects.
2. **Co-ordinate invariance:** the co-ordinate set carries no information.
3. **Sub-System Independence:** for two independent subsystem it should not matter if we treat the inference separately or jointly.
4. **Objectivity:** Under no new information, our inference should not change. Hence under no constraints, our posterior should coincide with our prior.

These lead to the generalised entropic functional,

$$-\int q(x)\log\frac{q(x)}{m(x)}dx - \sum_i \alpha_i\Bigg(\int_{x\in\mathcal{D}} f_i(x)dx - \mu_i\Bigg). \quad (9)$$

Here the justification for restricting ourselves to a functional is derived from considering the set of all distributions $q_i(\lambda)$ compatible with the constraints and devising a transitive ranking scheme. It can be shown, further, that Newton's laws, non-relativistic quantum mechanics and Bayes' rule can all be derived under this formalism.

In the case of a flat prior over the spectral domain, we reduce to the method of maximum entropy with moment constraints Jaynes (1982, 1957b). Conditions for the existence of a solution to this problem have been proved for the case of the Hausdorff moment conditions Mead and Papanicolaou (1984), of which our problem is a special case.

Algorithm

The generalised dual objective function which we minimise is,

$$\mathcal{S}(q,q_0) = \int_0^1 q_0(\lambda)\exp(-[1+\sum_i \alpha_i\lambda^i])d\lambda + \sum_i \alpha_i\mu_i, \quad (10)$$

which can be shown to have gradient

$$\frac{\partial\mathcal{S}(q,q_0)}{\partial\alpha_j} = \mu_j - \int_0^1 q_0(\lambda)\lambda^j\exp(-[1+\sum_i \alpha_i\lambda^i])d\lambda, \quad (11)$$

and Hessian

$$\frac{\partial^2\mathcal{S}(q,q_0)}{\partial\alpha_j\partial\alpha_k} = \int_0^1 q_0(\lambda)\lambda^{j+k}\exp(-[1+\sum_i \alpha_i\lambda^i])d\lambda. \quad (12)$$

Log Determinant

A common hindrance, appearing in Gaussian graphical models, Gaussian Processes Rue and Held (2005); Rasmussen (2006), sampling, variational inference MacKay (2003), metric/kernel learning Davis et al. (2007); Van Aelst and Rousseeuw (2009), Markov random fields Wainwright and Jordan (2006), Determinantal Point Processes (DPP's) and Bayesian Neural networks MacKay (1992), is the calculation of the log determinant of a large positive definite matrix. For a large positive definite matrix $K \in \mathcal{R}^{n\times n}$, the canonical solution involves the Cholesky decomposition, $K = LL^T$. The log determinant is then trivial to calculate as $\log\mathrm{Det}(K) = 2\sum_{i=1}^{n}\log L_{ii}$. This computation invokes a computational complexity $\mathcal{O}(n^3)$ and storage complexity $\mathcal{O}(n^2)$ and is thus unfit for purpose for $n > 10^4$, i.e. even a small sample set in the age of big data.

Related Work

Recent work in machine learning combined stochastic trace estimation with Taylor approximations for Gaussian process parameter learning Zhang and Leithead (2007); Boutsidis et al. (2017), Further developments included improved performance using Chebyshev polynomials Han, Malioutov, and Shin (2015) and Lanczos techniques with structured kernel interpolation (SKI) in order to accelerate MVMs to $\mathcal{O}(n + i\log i)$, where i is the number of inducing points Dong et al. (2017).

This approach relies on an extension of Kronecker and Toeplitz methods, which are limited to low dimensional (typically $D \leq 5$) data, which cannot be assumed in general. Secondly, whilst Lanczos methods have a convergence rate of double that of the Chebyshev approaches, the derived bounds require $\mathcal{O}(\sqrt{\kappa})$ Lanczos steps Ubaru, Chen, and Saad (2016), where κ is the matrix condition number. In many practical cases of interest $\kappa > 10^{10}$ and thus the large number of m matrix vector multiplications becomes prohibitive. We restrict ourselves to the high-dimensional, high-condition number, big data limit.

Log Determinants as a Density Estimation Problem

Any symmetric positive definite (PD) matrix K, is diagonalizable by a unitary transformation U, i.e $K = U^t D U$, where D is the matrix with the eigenvalues of K along the diagonal. Hence we can write the log determinant as:

$$\log \text{Det} K = \log \prod_i \lambda_i = \sum_{i=1}^{n} \log \lambda_i = n \mathbb{E}_\mu(\log \lambda). \quad (13)$$

Here we have used the cyclicity of the determinant and $\mathbb{E}_\mu$ denotes the expectation under the spectral measure. The latter can be written as:

$$\int_{\lambda_{min}}^{\lambda_{max}} d\mu(\lambda) \log \lambda = \int_{\lambda_{min}}^{\lambda_{max}} \sum_{i=1}^{n} \frac{1}{n} \delta(\lambda - \lambda_i) \log \lambda d\lambda. \quad (14)$$

Given that the matrix is PD, we know that $\lambda_{min} > 0$ and we can divide the matrix by an upper bound, $\lambda_u \geq \lambda_{max}$, via the Gershgorin circle theorem Gershgorin (1931) such that,

$$\begin{aligned} &\log \text{Det} \frac{K}{\lambda_u} = n \mathbb{E}_\mu(\log \lambda') = n \mathbb{E}_\mu(\log \lambda) - n \lambda_u \\ &\therefore \log \text{Det} K = n \mathbb{E}_\mu(\log \lambda') + n \lambda_u. \end{aligned} \quad (15)$$

Here $\lambda_u = \arg\max_i (\sum_{j=1}^{n} |K_{ij}|)$, i.e the max sum of the rows of the absolute of the matrix K. Hence we can comfortably work with the transformed measure,

$$\int_{\lambda_{min}/\lambda_u}^{\lambda_{max}/\lambda_u} p(\lambda') \log \lambda' d\lambda' = \int_0^1 p(\lambda') \log \lambda' d\lambda', \quad (16)$$

as the spectral density $p(\lambda)$ is 0 outside of its bounds, which are bounded by $[0, 1]$ respectively.

Stochastic Trace Estimation

Using the expectation of quadratic forms, for any multivariate random variable v with mean m and variance Σ, we can write

$$\mathbb{E}(zz^t) = mm^t + \Sigma \xrightarrow[m=0]{\Sigma = I} I, \quad (17)$$

where in the last equality we have assumed that the variable possesses zero mean and unit variance. By the linearity of trace and expectation for any $m \geq 0$ we can write

$$\sum_{i=1}^{n} \lambda^m = n \mathbb{E}_\mu(\lambda^m) = \text{Tr}(IK^m) = \mathbb{E}(zK^m z^t). \quad (18)$$

In practice we approximate the expectation over all random vectors with a simple Monte Carlo average. i.e for d random vectors ,

$$\mathbb{E}(zK^m z^t) \approx \frac{1}{d} \left(\sum_{j=1}^{j=d} z_j K^m z_j^t \right), \quad (19)$$

where we take the product of the matrix K with the vector z_j, m times, so as to avoid costly $\mathcal{O}(n^3)$ matrix matrix multiplication. This allows us to calculate the non central moment expectations in $\mathcal{O}(dmn^2)$ for dense matrices, or $\mathcal{O}(dm \times nnz)$ for sparse matrices, where $d \times m << n$. The random unit vector z_j can be drawn from any distribution, such as a Gaussian.

Algorithm 1 Computing Log Determinant using Constrained Variational Inference

1: **Input:** PD Symmetric Matrix A, Order of stochastic trace estimation k, Tolerance ϵ
2: **Output:** Log Determinant Approximation $\log |K|$
3: $B = K/\lambda_u$
4: μ (moments)$\leftarrow$ StochasticTraceEstimation(B, k)
5: α (coefficients) $\leftarrow$ VBALD(μ, ϵ)
6: $q(\lambda) \leftarrow q(\lambda|\alpha)$
7: $\log |A| \leftarrow n \int \log(\lambda) q(\lambda) d\lambda + n \log(\lambda_u)$

Experiments

For simplicity we have kept all the formula's in terms of power moments, however we find vastly improved performance and conditioning when we switch to another polynomial basis. Many alternative and orthogonal Polynomial bases exist (so that the errors between moment estimations are uncorrelated), we implement both Chebyshev and Legendre moments in our Lagrangian and find similar performance for both. The use of Chebyshev moments in Machine Learning and Computer Vision has been reported to be of practical siginifcance previously Yap, Raveendran, and Ong (2001). We use Python's SciPy minimize standard newton-conjugate gradient algorithm to solve the objective, given the gradient and hessian to within a gradient tolerance $gtol$. To make the Hessian better conditioned so as to achieve convergence we add jitter $1e-8$ along the diagonal. The pseudo code is given in Algorithm 2. The Log Determinant is then calculated using Algorithm 1.

Comparison to Previous MaxEnt algorithm

In Granziol and Roberts (2017a) it was proven that the addition of an extra constraint cannot increase the entropy of the MaxEnt solution. For the problem of log determinants, this signifies that the entropy of the spectral approximation should decrease with the addition of every moment constraint. We implement the Maximum Entropy algorithm Bandyopadhyay et al. (2005) in the same manner as applied in Entropic trace estimation Fitzsimons et al. (2017). We show results for the Themomech UFL SuiteSparse dataset, with $n = 80,900$, for which the true log determinant can be

Table 1: Absolute relative error for VBALD, Chebyshev & Lanczos methodson varying length-scale l, with varying condition number κ on squared exponential kernel matrices $K \in \mathbb{R}^{1000\times1000}$.

κ	l	VBALD	CHEBYSHEV	LANCZOS
3×10^1	0.05	**0.0014**	0.0037	0.0024
1.1×10^3	0.15	0.0522	**0.0104**	0.0024
1.0×10^5	0.25	0.0387	0.0795	**0.0072**
2.4×10^6	0.35	0.0263	0.2302	**0.0196**
8.3×10^7	0.45	**0.0284**	0.3439	0.0502
4.2×10^8	0.55	**0.0256**	0.4089	0.0646
4.3×10^9	0.65	**0.00048**	0.5049	0.0838
1.4×10^{10}	0.75	**0.0086**	0.5049	0.1050
4.2×10^{10}	0.85	**0.0177**	0.5358	0.1199

calculated. We see that for the MaxEnt algorithm Bandyopadhyay et al. (2005) implementation used inFitzsimons et al. (2017) that for $m > 8$ moments, the error (Figure 2) and the entropy (Figure1) begin to increase. For our algorithm, beyond seeing that the performance is vastly increased (Figure 4), the error continually decreases with increasing moment information and the Entropy (Figure 3) decreases smoothly.

Synthetic Kernel Data

We simulate the kernel matrices from a Gaussian/Determinental Point Process Rasmussen and Williams (2006), by generating a typical squared exponential kernel matrix $K \in \mathbb{R}^{n\times n}$ using the Python GPy package with 6 dimensional, Gaussian inputs. We then add noise of variance 10^{-8} along the diagonals. We employ a variety of realistic uniform length-scales. We use $m = 30$ Moments, $d = 50$ Hutchinson probe vectors and compare our novel MaxEnt algorithm against the Taylor approximation, Chebyshev Han, Malioutov, and Shin (2015) and Lanczos Ubaru, Chen, and Saad (2017) in Table . We see that for low condition numbers (Figure 6) The benefit of framing the log determinant as an optimization problem is marginal, whereas for large condition numbers (Figures 7) the benefits are substantial, with orders of magnitude better results than competing methods. We also provide the number of Chebyshev and Lanczos steps required to achieve similar performance.

Machine Learning Application 1: DPPs

Determinantal point processes (DPPs) Macchi (1975) are probabilistic models capturing global negative correlations. They describe Fermions [3] in Quantum Physics, Eigenvalues of random matrices and non-intersecting random walks.

In machine learning their natural selection of diversity has found applications in the field of summarization Gong et al. (2014), human pose detection Kulesza (2012), clustering Kang (2013), Low rank kernel matrix approximations Li,

[3] as a consequence of the spin-statistics theorem

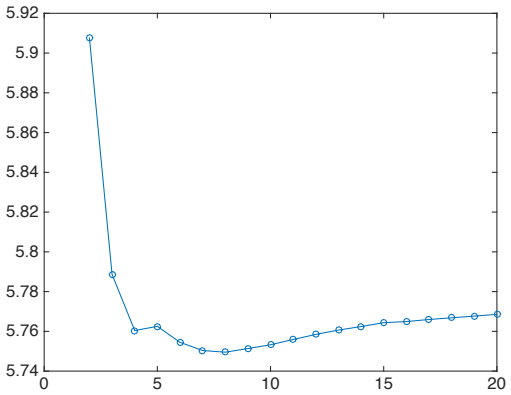

Figure 1: Previous Maximum Entropy Algorithm

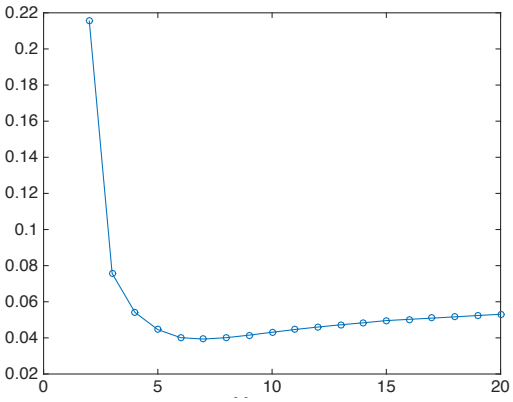

Figure 2: Previous Maximum Entropy Algorithm

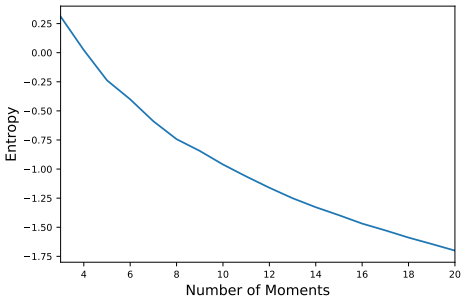

Figure 3: New Maximum Entropy Algorithm

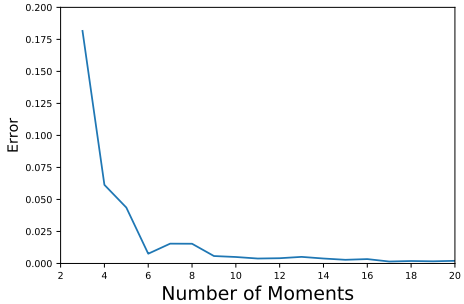

Figure 4: Previous Maximum Entropy Algorithm

Figure 5: New Maximum Entropy Algorithm

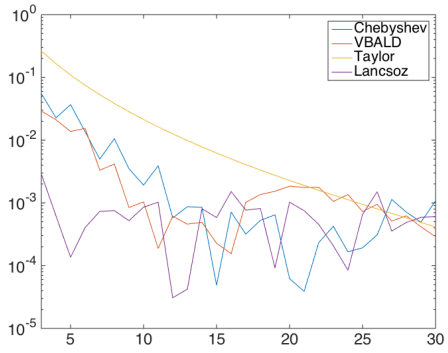

Figure 6: Length Scale = 0.1, Condition number = 16

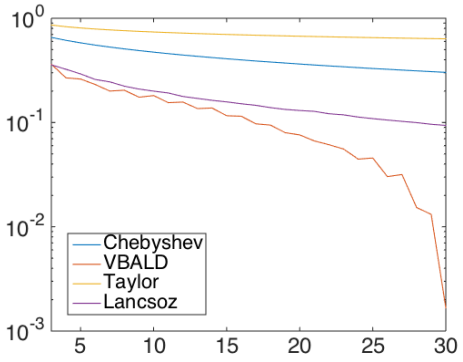

Figure 7: Length Scale = 0.33, Condition number = 2×10^7 Equivalent Chebyshev steps $n = 1200$, Lanczos steps $n \approx 100$.

Figure 8: Comparison of VBALD against Taylor, Lanczos and Chebyshev algorithms for calculating the log determinant of synthetic squared exponential kernel matrices of different condition number. The absolute relative error is on the y-axis and number of moments used on the x-axis.

Jegelka, and Sra (2016) and Manifold learning Wachinger and Golland (2015).

Formally, it defines a distribution on 2^y, where $y = [n]$ is the finite ground set. For a random variable $X \subseteq Y$ drawn from a given DPP we have

$$P(X = x) \propto \det(K_X) = \frac{\det(K_x)}{\det(K + I)}, \tag{20}$$

where $K \in R^{d \times d}$ is a positive definite matrix referred to as the L-ensemble kernel. Greedy algorithms that find the most diverse set Y of y that achieves the highest probability, i.e $\text{argmax}_{X \subseteq y} \det(K_Y)$ require the calculation of the marginal gain,

$$\log \det K_{X \cup \{i\}} - \log \det L_X. \tag{21}$$

with $\mathcal{O}(n^3)$ computational complexity. Previous work has looked at limiting the burden of the computational complexity by employed Chebyshev approximations to the Log Determinant Han et al. (2017). However their work is limited Kernel Matrices with minimum eigenvalues of 10^{-34}, which does not cover the class of realistic kernel matrix spectra, notably the popular squared exponential kernel. We develop a method in section **??**method and an algorithm which is capable of handling very high condition numbered matrices.

Machine Learning Application 2: Learning Cluster Number

For most clustering algorithms, including spectral clustering, estimating the number of clusters is a challenging problem (Von Luxburg, 2007), with likelihood, ad-hoc, information theoretic, stability and spectral approaches advocated. For many large scale spectral approaches, the number of clusters is taken as a given input Liu et al. (2013),Cucuringu et al. (2016).

In the latter, one analyses the spectral gap in the eigenvalue spectrum, which we refer to as *eigengap* for short. Applying this approach in the era of big-data (where social networks such as Facebook are approaching $n = 2 \times 10^9$ users) means that standard approaches, such as the canonical Cholesky decomposition, with computational complexity $\mathcal{O}(n^3)$ and storage $\mathcal{O}(n^2)$ are completely prohibitive.

CLUSTERING USING THE EIGENSPECTRA

It is well known Von Luxburg (2007), that the multiplicity of the 0 eigenvalue is equal to the number of disconnected components in the graph. Hence were we to define a community as a completely disconnected component, we could simply count the number of 0 eigenvalues. However given that real world networks are rarely completely disconnected, this procedure would be of little practical utility.

We hence adopt a looser definition of the word cluster and consider groups of nodes containing far greater intra-group connections than inter-group connections.

If the graph is connected, but consists of k subgraphs which are "weakly" linked to each other, the unnormalized

[4] or alternatively low condition numbers

Laplacian has one zero eigenvalue and all the other eigenvalues positive. This is easily seen by looking at

$$\mathbf{u}^T L\mathbf{u} = \sum_{i,j=1}^{n} w_{ij}(u_i - u_j)^2 \tag{22}$$

which is positive, as $w_{ij} > 0$ and given a connected graph G has a single 0 eigenvalue, all other eigenvalues are thus positive. For small changes in the Laplacian, we expect from matrix perturbation theory (Bhatia, 2013) that the next $k-1$ smallest eigenvalues will be close to 0.

Estimating the Cluster Number using Maximum Entropy

From the previous section, we see that the number of clusters is equal to the number of near zero eigenvalues. Assuming there is a clear spectral gap, i.e there exists a λ_* which upper bounds the largest perturbed eigenvalue and lower bounds the smallest non-zero eigenvalue pre perturbation, we can write the total number of clusters as

$$C = n\int_0^{\lambda_*} p(\lambda)d\lambda \tag{23}$$

with n being the number of nodes $L \in \mathbb{R}^{n\times n}$ and $p(\lambda)$ denoting the spectral density. A naive eigen-decomposition, which would give $p(\lambda)$ as a sum of delta functions, has an infeasible $\mathcal{O}(n^3)$ computational complexity. As previously mentioned in section **??**, the Lancsoz algorithm, which exploits matrix sparsity by working with matrix vector multiplications, has computational complexity $\mathcal{O}(n_{\text{nz}} \times m + nm^2) \times d$, where for very large sparse matrices, the second term becomes dominant. Given that empirically many social, biological and technical communities are sparse and that all we need for detecting cluster count is an estimation of the eigenvalues and not the eigenvectors, we look for an alternative computationally more effective method of estimating the spectral density. We use the method of maximum entropy, with computational complexity equivalent to Lancsoz without the second term.

EXPERIMENTS

We use $d = 100$ Gaussian random vectors for our stochastic trace estimation, for both MaxEnt and Lanczos (Ubaru, Chen, and Saad, 2017). When comparing MaxEnt with Lanczos we set the number of moments m equal to the number of Lanczos steps, as they are both matrix vector multiplications in the Krylov subspace. We implement a quadrature MaxEnt algorithm 2. We use a grid size of 10^{-4} over the interval $[0, 1]$ and add diagonal noise on the Hessian to improve conditioning and symmetrise it. We further use Chebyshev polynomial input instead of power moments for improved performance and conditioning. In order to normalise the moment input we use the normalised Laplacian with eigenvalues bounded by $[0, 2]$ and divide by 2. To make a fair comparison we take the output from Lancsoz (Ubaru, Chen, and Saad, 2017) and apply kernel smoothing (Lin, Saad, and Yang, 2016) before applying our cluster estimator. We explain the details of our kernel smoothing in section .

Algorithm 2 MaxEnt Algorithm

1: **Input:** Moments $\{\mu_i\}$, Tolerance ϵ, Hessian noise η
2: **Output:** Coefficients $\{\alpha_i\}$
3: Initialize $\alpha_i = 0$.
4: Minimize $\int_0^1 p_\alpha(\lambda)d\lambda + \sum_i \alpha_i\mu_i$
5: Gradient $\mu_j - \int_0^1 p_\alpha(\lambda)\lambda^j d\lambda$
6: Hessian $= \int_0^1 p_\alpha(\lambda)\lambda^{j+k}d\lambda$
7: Hessian $= (H + H')/2 + \eta$
8: Until $\forall j$ Gradient$_j < \epsilon$

Algorithm 3 Cluster Estimator Algorithm

1: **Input:** Lagrange Multipliers α_i, Matrix Dimension n, Tolerance ϵ
2: **Output:** Number of Clusters N_c
3: Initialize $p(\lambda) \to p(\lambda|\alpha_i) = \exp -[1 + \sum_i \alpha_i x^i]$.
4: Minimize $\lambda*$ s.t $\frac{dp(\lambda)}{d\lambda}|_{\lambda=\lambda*} \leq \epsilon$
5: Calculate $N_c = n\int_0^{\lambda*} p(\lambda)d\lambda$

Synthetic Data

In order to test the robustness of the approach to networks with clusters of different structures, we implement a mixture of Erdös-Rényi , Watts-Strogatz and Barabsi-Albert networks using the Python package *NetworkX* and conduct multiple experiments using networks that have from 9 to 240 clusters, with each cluster containing 30 nodes. We connect the nodes between clusters randomly, with a single inter-cluster connection.

Figure **??** shows the community detection errors, expressed in the logarithm to the base 10, for networks of 9, 30, 90, 240 clusters over number of matrix vector calculations (i.e. number of moments). We see that for both methods, the detection error generally decreases as more moments are used. For an equivalent number of matrix vector calculations, MaxEnt outperforms the Lanczos algorithm. As there is no accepted prescription by which we can determine when the spectral minimum has been best learned, the occasional dips in error produced by Lanczos (such as for 15 moments in Figure **??** are unlikely to be replicated in real world experiments.

Table 2 displays the fractional errors in community detection when we apply Lanczos and MaxEnt, both using 80 moments, to synthetic networks of different sizes and cluster numbers. In each case, lower detection error is highlighted in bold. It is evident that MaxEnt outperforms Lanczos as the number of clusters and the network size increase. We observe a general improvement in performance for larger graphs, visible in the differences between fractional errors for MaxEnt and not Lanczos. This is to be expected as the true spectral density

$$p(\lambda) = \frac{1}{n}\sum_i^n \delta(\lambda - \lambda_i) \tag{24}$$

becomes continuous in the $n \to \infty$ limit and hence we expect the density to be better approximated by a continuous

Table 2: Fractional error in community detection for synthetic networks using MaxEnt and Lanczos with 80 moments

# OF CLUSTERS (N)	LANCZOS	MAXENT
9 (270)	$\mathbf{3.20 \times 10^{-3}}$	9.70×10^{-3}
30 (900)	1.41×10^{-2}	$\mathbf{6.40 \times 10^{-3}}$
90 (2700)	1.81×10^{-2}	$\mathbf{5.80 \times 10^{-3}}$
240 (7200)	2.89×10^{-2}	$\mathbf{3.50 \times 10^{-3}}$

distribution for larger n (Fitzsimons et al., 2017).

To test the performance of our approach for networks that are too big to apply eigen-decomposition, we also generate two large networks by mixing Erdös-Rényi, Watts-Strogatz and Barabsi-Albert networks. The first large network has a size of 201,600 nodes and comprises 350 interconnected clusters whose size varies from 500 to 1000 nodes. The other large network has a size of 404,420 nodes and comprises interconnected 1355 clusters whose size varies from 200 to 800 nodes. The results in Figure **??** show that our MaxEnt approach outperforms Lanczos for both large synthetic networks.

Real Data

Small Real World Data When the number of nodes $n \approx 10^3$, it is possible to compute the eigen-decomposition exactly and hence to benchmark the performance of our algorithm in the real world.

The first real-world dataset we use is the Email network, which is generated using email communication data among $1,005$ members of a large European research institution and is an undirected graph of $n = 1,005$ nodes. We calculate the ground-truth by computing the eigenvalues explicitly and finding the spectral gap near 0. As shown in Figure 11, we count 20 very small eigenvalues before a large jump in magnitude and set this as the ground truth for the number of clusters in the network. This corresponds to a drop in spectral eigendensity as displayed in the lower subplot of Figure 11.

We note that this differs from the value of 42 given by the number of departments at the research institute. A likely reason for this ground truth inflation is that certain departments, Astrophysics, Theoretical Physics and Mathematics for example, may collaborate to such an extent that their division in name may not be reflected in terms of node connection structure.

We display the process of spectral learning for both MaxEnt and Lanczos, by plotting the spectral density of both methods against the true eigenvalue spectral density in Figure **??**. In order to make a valid comparison, we smooth the implied density using a Gaussian kernel, with $\sigma = 10^{-3}$. We note that both MaxEnt and Lanczos approximate the ground truth better with a greater number of moments/steps m and that Lanczos learns the extrema before the bulk of the distribution.

We plot the log error against the number of moments for

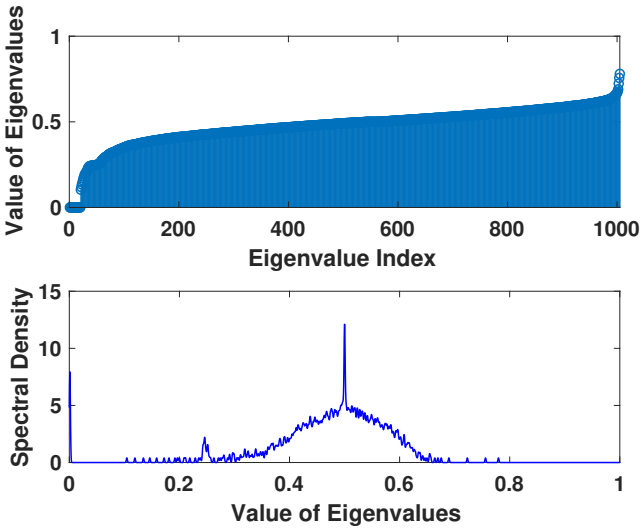

Figure 9: Stem Graph and Eigen-Spectrum

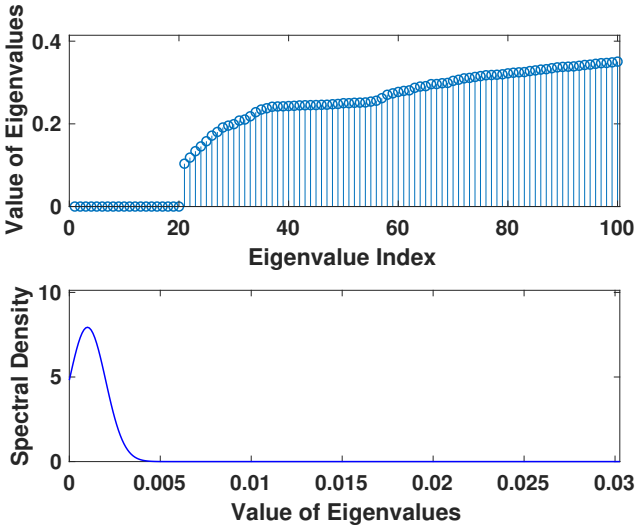

Figure 10: Zoomed-in Graphs

Figure 11: Stem graph of the eigen-spectrum of the Email Dataset. The subplot (a) shows all the eigenvalues and the whole eigenvalue spectrum. The subplot (b) is a zoomed-in version of (a), which displays the smallest 100 eigenvalues with a clear spectral gap at 20 and the corresponding spectral density near the origin. The area under the spectral density up to 0.005 multiplied by the number of nodes n predicts the number of clusters.

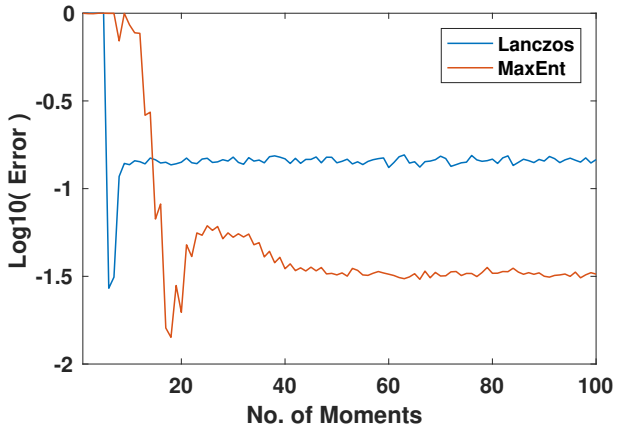

Figure 12: Email Dataset

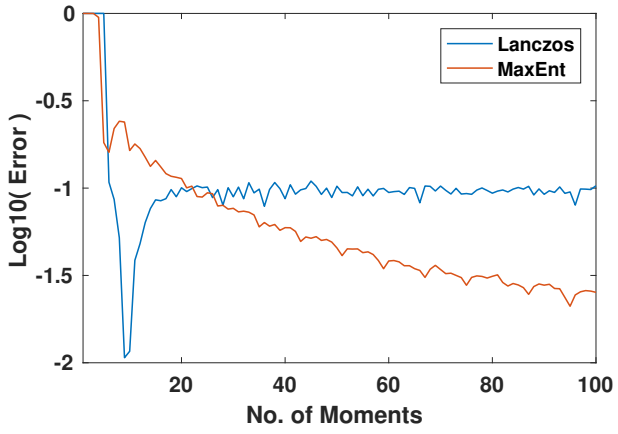

Figure 13: NetScience Dataset

Figure 14: Log error of community detection using MaxEnt and Lanczos algorithms on for differing number of moments m.

both MaxEnt and Lanczos in Figure 12, with MaxEnt showing superior performance.

We repeat the experiment on the Net Science collaboration network, which represents a co-authorship network of $1,589$ scientists ($n = 1,589$) working on network theory and experiment (Newman, 2006). The results in Figure 14 show that MaxEnt quickly outperforms the Lanczos algorithm after around 20 moments.

Appendix

First order Eigenvalue perturbation

We formalise this intuition by considering a small perturbation of the Laplacian $\tilde{L} = L + \delta L$, where $||\delta L|| \ll ||L||$. Considering the vectors $\vec{u_i}$ to be the normalised eigenvectors of the unperturbed Laplacian L we solve the equation

$$(L + \delta L)(\vec{u_i} + \delta\vec{u}_i) = (\lambda_i + \delta\lambda_i)(\vec{u_i} + \delta\vec{u}_i) \tag{25}$$

Cancelling and dropping second order terms we have

$$\delta L\vec{u_i} + L\delta\vec{u}_i = \lambda_i\delta\vec{u}_i + \delta\lambda_i\vec{u_i} \tag{26}$$

expressing the vector $\delta\vec{u}_i$ in the basis of the eigenvectors $\vec{u_i}$ of L, which can always be done as the L is normal and hence its eigenvectors span the space of $\mathbb{R}^{n\times n}$, hence $\delta\vec{u}_i = \sum_j^n \epsilon_j\vec{u_j}$. Hence we can write equation (26) as

$$\delta L\vec{u_i} + L\sum_j^n \epsilon_j\vec{u_j} = \lambda_i\sum_j^n \epsilon_j\vec{u_j} + \delta\lambda_i\vec{u_i}$$

$$\vec{u_i}^T\delta L\vec{u_i} + \sum_j^n \lambda_j\epsilon_j\vec{u_i}^T\vec{u_j} = \lambda_i\sum_j^n \epsilon_j\vec{u_i}^T\vec{u_j} + \delta\lambda_i\vec{u_i}^T\vec{u_i}$$

$$\vec{u_i}^T\delta L\vec{u_i} = \delta\lambda_i \tag{27}$$

Where we have used the eigenvalue equation $L\vec{u_i} = \lambda_i\vec{u_i}$ and that the eigenvectors are orthonormal $\vec{u_j}^t\vec{u_i} = \delta_{i,j}$. We can now bound this term using the Cauchy-Schwarz inequality

$$\delta\lambda_i = \vec{u_i}^T\delta L\vec{u}_i \leq ||\delta L||\,||\vec{u_i}^T\vec{u}_i|| = ||\delta L|| \tag{28}$$

If we consider the natural variant of the Laplacian, normalised by the number of vertices in the graph, i.e $L_{\text{natural}} = (D - A)/n$, then by adding R vertices between previously disconnected subgraphs, for each vertex, we alter a two diagonal components by $+1$ and two off diagonal components by -1.Thus, using the Frobenius norm, our bound B goes as

$$B = \sqrt{\sum_{i=1}^n\sum_{j=1}^n \left|\delta L_{i,j}\right|^2} = \frac{2R}{n} \tag{29}$$

We note that our derived bound using the Cauchy-Schwarz inequality is exactly the same as Weyl's perturbation theorem for Hermitian matrices, which uses the min-max principle (Bhatia, 2013).

Hence we expect the eigenvalue perturbations to die off as $\mathcal{O}(n^{-1})$ for a constant number of connections between clusters as we increase the number of nodes n in the network. Even if the number of such connections grows with n but is sparse such that the total number is $\mathcal{O}(ns)$ with small sparsity s, the perturbation would only be of order s. For small sparsity s we would expect the spectral gap between the perturbed eigenvalues which were at 0 pre perturbation and the non zero eigenvalues to remain non-negligible. In these cases, we expect our cluster detection algorithm, introduced in the next section to also work.

If we choose to work with the normalised Laplacian defined in (**??**), then for each new connection between previously disconnected components we get a term of the form

$$\sum_{j=1}\left|\left|\frac{1}{\sqrt{d_i d_j}} - \frac{1}{\sqrt{(d_i+1)d_j}}\right|\right|^2 + \frac{2}{(d_i+1)(d_{k+1})}$$
$$+\sum_{l=1}\left|\left|\frac{1}{\sqrt{d_k d_l}} - \frac{1}{\sqrt{(d_k+1)d_l}}\right|\right|^2 \tag{30}$$

where nodes k and i are being connected and nodes j and l are the nodes connected to k and i, respectively. By taking the degrees to be a fraction of the total number of nodes n and taking n to be large we observed a similar n^{-1} scaling. The idea of strong communities being nearly disconnected components, is not novel (McGraw and Menzinger, 2008) and has been used in community detection algorithms (Capocci et al., 2005). However we have not come across a simple exposition of the results from matrix perturbation theory, or the application of the Cauchy-Schwarz inequality to bound the increase in the 0 eigenvalues as a function of node number n or degrees d_i amongst the connected components.

Polynomial approximations to the Log Determinant

Recent work Han, Malioutov, and Shin (2015); Dong et al. (2017); Zhang and Leithead (2007) has considered incorporating knowledge of the non central moments [5] of a normalised eigenspectrum by replacing the logarithm with a finite polynomial expansion,

$$\mathbb{E}_\mu = \int_0^1 p(\lambda)\log(\lambda)d\lambda = \int_0^1 p(\lambda)\log(1-(1-\lambda))d\lambda. \tag{31}$$

Given that $\log(\lambda)$ is not analytic at $\lambda = 0$, it can be seen that, for any density with a large mass near the origin, a very large number of polynomial expansions, and thus moment estimates, will be required to achieve a good approximation, irrespective of the choice of basis.

Taylor approximations are probabilistically unsound

In the case of a Taylor expansion equation (31) can be written as,

$$-\int_0^1 p(\lambda)\sum_{i=1}^{\infty}\frac{(1-\lambda)^i}{i} \approx -\int_0^1 p(\lambda)\sum_{i=1}^{m}\frac{(1-\lambda)^i}{i}. \tag{32}$$

The error in this approximation can be written as the difference of the two sums,

$$-\sum_{i=m+1}^{\infty}\frac{\mathbb{E}_\mu(1-\lambda)^i}{i}, \tag{33}$$

where we have used the Taylor expansion of $\log(1-x)$ and $\mathbb{E}_\mu$ denotes the expectation under the spectral measure.

De-Finetti De Finetti (1974) showed that Kolmogorov's axioms of probability Kolmogorov (1950) could be derived by manipulating probabilities in such a manner so as to not make a sure loss on a gambling system based on them. Such a probabilistic framework, of which the Bayesian is a special case Walley (1991a), satisfies the conditions of,

1. **Non Negativity:** $p_i \geq 0\,\forall i$**,**
2. **Normalization:** $\sum_i p_i = 1$**,**
3. **Finite Additivity:** $P(\cup_{n=1}^N A_n) = \sum_{n=1}^N P(A_n)$**.** [6]

The intuitive appeal of De-Finetti's sure loss arguments, is that they are inherently performance based. A sure loss is a practical cost, which we wish to eliminate.

Keeping within such a very general formulation of probability and thus inference. We begin with complete ignorance about the spectral density $p(\lambda)$ (other than its domain $[0,1]$) and by some scheme after seeing the first m non-central moment estimates we propose a surrogate density $q(\lambda)$. The error in our approximation can be written as,

$$\int_0^1 [p(\lambda)-q(\lambda)]\log(\lambda)d\lambda$$
$$= \int_0^1 -[p(\lambda)-q(\lambda)]\sum_{i=1}^{\infty}\frac{(1-\lambda)^i}{i}d\lambda. \tag{34}$$

For this error to be equal to that of our Taylor expansion (33), our implicit surrogate density must have the first m non-central moments of $(1-\lambda)$ identical to the true spectral density $p(\lambda)$ and all others 0.

For any PD matrix K, for which $E_\mu(1-\lambda)^i > 0$, $\forall i \leq m$[7], for equation (34) to be equal to (33), we must have,

$$\int_0^1 q(\lambda)\sum_{m+1}^{\infty}\frac{(1-\lambda)^i}{i}d\lambda = 0. \tag{35}$$

Given that $0 \leq \lambda \leq 1$ and that we have removed the trivial case of the spectral density (and by implication its surrogate) being a delta function at $\lambda = 1$, the sum is manifestly positive and hence $q(\lambda) < 0$ for some λ, which is incompatible with the theory of probability De Finetti (1974); Kolmogorov (1950).

Prior Spectral Belief If we assume complete ignorance over the spectral domain, then the natural maximally entropic prior is the uniform distribution and hence $q(\lambda) = 1$.[8] An alternative prior over the $[0,1]$ domain is the Beta distribution, the maximum entropy distribution of that domain under a mean and log mean constraint,

$$\frac{\Gamma(\gamma+\beta)}{\Gamma(\gamma)\Gamma(\beta)}\lambda^{\gamma-1}(1-\lambda)^{\beta-1}. \tag{36}$$

The log mean constraint is particularly interesting as we know that it must exists for a valid log determinant to exist, as is seen for equation (14). We set the parameters of by maximum likelihood, hence,

$$\gamma = \frac{\mu_1(\mu_1-\mu_2)}{\mu_2-\mu_1^2}\,,\ \beta = \left(\frac{1}{\mu_1}-1\right)\frac{\mu_1(\mu_1-\mu_2)}{\mu_2-\mu_1^2}\,. \tag{37}$$

[5] Also using stochastic trace estimation.

[6] for a sequence of disjoint sets A_n.

[7] we except the trivial case of a Dirac distribution at $\lambda = 1$, which is of no practical interest

[8] Technically as the log determinant exists and is finite, we cannot have any mass at $\lambda = 0$, hence we must set the uniform between some $[\delta\epsilon, 1]$, where $\delta\epsilon > 0$.

Analytical surrogate form Our final equation for $q(\lambda)$ can be written as,

$$q(\lambda) = \frac{\Gamma(\gamma+\beta)}{\Gamma(\gamma)\Gamma(\beta)}\lambda^{\gamma-1}(1-\lambda)^{\beta-1} \times \exp(-[1+\sum_{i=0}^{m}\alpha_i\lambda^i]) \tag{38}$$

for the beta prior and

$$q(\lambda) = \exp(-[1+\sum_{i=0}^{m}\alpha_i\lambda^i]) \tag{39}$$

for the uniform. The exponential factor can be thought of altering the prior beta/uniform distribution so as to fit the observed moment information.

Lagrangian Duality

Consider a generic optimization problem of the form,

$$\begin{aligned} &\text{minimize } f_0(x) \\ &\text{subject to } f_i(x) \leq 0,\ i = 1...m \\ &\text{subject to } h_i(x) = 0,\ i = 1...p \end{aligned} \tag{40}$$

where $x \in \mathbb{R}^n$ and the domain $\mathcal{D} = \bigcap_{i=0}^{m} f_i \cap \bigcap_{i=1}^{p} h_i$. We define the Lagrangian dual function as the infimum of the Lagrangian over the domain of x,

$$\begin{aligned} g(\lambda,\nu) &= \inf_{x\in\mathcal{D}} L(x,\lambda,\nu) \\ &= \inf_{x\in\mathcal{D}}\left(f_0(x) + \sum_{i=1}^{m}\lambda_i f_i(x) + \sum_{i=1}^{p}\nu_i h_i(x)\right). \end{aligned} \tag{41}$$

As the dual is the pointwise infimum of a family of affine functions of (λ,ν), it is concave, irrespective of the convexity of f_0, f_i, h_i. Boyd and Vandenberghe (2009). It is easily verifiable due to the net negativity of the two summation terms in $g(\lambda,\nu)$ that the dual provides a lower bound on the optimal value p^* of the primal problem. This is known as weak duality. In the case of equality constraints this bound is tight.

For general inequality constraints the difference between the primal and dual optimal solution (duality gap) is not 0. However, for $f_0...f_m$ convex, affine equality constraints and certain regularity conditions, we have a duality gap of 0, this is known as strong duality. An example of such a constraint qualification is Slater's condition, which states that there is an $x \in$ **relint** $\mathcal{D}$ which satisfies the constraints.

Application to Probability Distributions

We consider a probability distribution $p : \mathcal{R}^n \to \mathcal{R}$ which satisfies the general axioms of non-negativity, associativity and normalizability. This defines a very general space of probability theories, of which the Bayesian formalism is a special case Walley (1991b). Thus $p(x) \geq 0$ for all $x \in C$ and $\int p(x)dx = 1$, where $C \subseteq \mathcal{R}^n$ is convex. The last condition follows from the definition of convexity and the fact that any sum of two real numbers is a real number. Then as any non negative weighting of a convex set preserves convexity,

$$\int_C p(x)x\,dx \in C \tag{42}$$

if the integral exists.

Application to Maximum Entropy

We wish to maximise the entropic functional $\mathcal{S}(p) = -\int p(x)\log p(x)dx$ under certain moment constraints $\int p(x)x^m dx = \mu_m$. This can be written as,

$$\begin{aligned} &\text{minimize } f_0[p(x)] = \int p(x)\log p(x)dx \\ &\text{subject to } h_i[p(x)] = \int p(x)x^i dx - \mu_i = 0,\ i = 1...p. \end{aligned} \tag{43}$$

Given that the negative entropy is a convex objective and that the moment equality constraints are affine in the variable being optimised over $p(x)$ by strong duality we have an equivalence between the solution of the dual and that of the primal.

It is also clear that the domain defined as the intersection of the constraint sets can never increase upon the addition of an extra constraint. Hence,

$$\inf_{x\in\mathcal{D}=\bigcap_{i=0}^{m} f_i} L(x,\lambda,\nu) \leq \inf_{x\in\mathcal{D}=\bigcap_{i=0}^{m+1} f_i} L(x,\lambda,\nu) \tag{44}$$

and thus the entropy can only decrease when adding an extra constraint. Hence by adding more moment constraints, we always reduce the entropy and given equations (**??**) and (**??**) we necessarily reduce $\mathcal{D}_{kl}(p(x)\|q(x))$, where $p(x), q(x)$ define the true eigenvalue and MaxEnt proposal distributions respectively

KL as a measure of divergence

Using Pinsker's inequality, which is tight up to constant factors, we can relate the KL divergence to both the total variation distance and the total variation norm Cover and Thomas (2012):

$$\delta(P,Q) \leq \sqrt{\frac{1}{2}\mathcal{D}_{kl}(P\|Q)}, \tag{45}$$

where the total variation distance is defined as

$$\delta(P,Q) = sup\{|P(A) - Q(A)|\} \text{where} \in \Sigma. \tag{46}$$

The total variation norm between P and Q can be written as,

$$|P-Q| \leq \sqrt{2\mathcal{D}_{kl}(P\|Q)}. \tag{47}$$

This follows as $2\delta(P,Q) = |P-Q|_1$ where the 1 relates to the $L1$ norm.

Bound on Error of MaxEnt

To calculate the log determinant of the matrix in question, once we have the proposal eigenvalue distribution $q(x)$ we calculate the mean value of $\log(x)$ under the distribution $q(x)$, i.e $\int q(x)\log(x)dx$. We can write the error of our MaxEnt estimate as

$$\epsilon = \left|\int_{x\in\chi}[p(x)-q(x)]\log(x)dx\right| \tag{48}$$

Where $p(x)$ is the true eigenvalue distribution. $\forall p(x), q(x) \geq 0$ it is hence true that,

$$\left|\int_{x\in\chi}(p(x)-q(x))\log(x)dx\right| \leq \int_{x\in\chi}|p(x)-q(x)||\log(x)|dx. \tag{49}$$

From the monotonicity of the function $\log(x)$, we have that $\log(x) \leq \max[|\log(x_{max})|, |\log(x_{min})|$ and rewriting the total variational norm in terms of the KL divergence we have:

$$\epsilon \leq \max[|\log(x_{max})|, |\log(x_{min})|]\sqrt{2\mathcal{D}_{kl}(P\|Q)}. \quad (50)$$

We thus note that by reducing the self entropy of the proposal distribution $q(x)$ we necessarily reduce the maximum possible error of the log determinant estimation. However, given that we do not have an analytic form of $p(x)$, we cannot explicitly calculate $\mathcal{D}_{kl}(P\|Q)$ and hence the bound in its current form is not inherently practical. We leave the estimation of this term and derivation of estimate uncertainty to future work.

Acknowledgements

References

Bandyopadhyay, K.; Bhattacharya, A. K.; Biswas, P.; and Drabold, D. 2005. Maximum entropy and the problem of moments: A stable algorithm. *Physical Review E* 71(5):057701.

Bhatia, R. 2013. *Matrix analysis*, volume 169. Springer Science & Business Media.

Boutsidis, C.; Drineas, P.; Kambadur, P.; Kontopoulou, E.-M.; and Zouzias, A. 2017. A randomized algorithm for approximating the log determinant of a symmetric positive definite matrix. *Linear Algebra and its Applications* 533:95–117.

Boyd, S. P., and Vandenberghe, L. 2009. *Convex optimization*. Cambridge University Press.

Capocci, A.; Servedio, V. D.; Caldarelli, G.; and Colaiori, F. 2005. Detecting communities in large networks. *Physica A: Statistical Mechanics and its Applications* 352(2-4):669–676.

Caticha, A. 2012. Entropic inference and the foundations of physics (monograph commissioned by the 11th brazilian meeting on Bayesian statistics–ebeb-2012.

Chung, F. R. 1997. *Spectral graph theory*. Number 92. American Mathematical Soc.

Cover, T. M., and Thomas, J. A. 2012. *Elements of information theory*. John Wiley & Sons.

Cucuringu, M.; Koutis, I.; Chawla, S.; Miller, G.; and Peng, R. 2016. Simple and scalable constrained clustering: a generalized spectral method. In *Artificial Intelligence and Statistics*, 445–454.

Davis, J. V.; Kulis, B.; Jain, P.; Sra, S.; and Dhillon, I. S. 2007. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, 209–216. ACM.

De Finetti, B. 1974. Teoria delle probabilita. einaudi, turin, 1970. *English translation:[51]*.

Dong, K.; Eriksson, D.; Nickisch, H.; Bindel, D.; and Wilson, A. G. 2017. Scalable log determinants for gaussian process kernel learning. In *Advances in Neural Information Processing Systems*, 6330–6340.

Fitzsimons, J.; Granziol, D.; Cutajar, K.; Osborne, M.; Filippone, M.; and Roberts, S. 2017. Entropic trace estimates for log determinants.

Fox, C. W., and Roberts, S. J. 2012. A tutorial on variational bayesian inference. *Artificial intelligence review* 38(2):85–95.

Gershgorin, S. A. 1931. Über die Abgrenzung der Eigenwerte einer Matrix. *Izvestija Akademii Nauk SSSR, Serija Matematika* (6):749–754.

Giffin, A.; Cafaro, C.; and Ali, S. A. 2016. Application of the Maximum Relative Entropy method to the Physics of Ferromagnetic Materials. *Physica A: Statistical Mechanics and its Applications* 455:11 – 26.

Gong, B.; Chao, W.-L.; Grauman, K.; and Sha, F. 2014. Diverse sequential subset selection for supervised video summarization. In *Advances in Neural Information Processing Systems*, 2069–2077.

Granziol, D., and Roberts, S. 2017a. An Information and Field Theoretic approach to the Grand Canonical Ensemble.

Granziol, D., and Roberts, S. J. 2017b. Entropic determinants of massive matrices. In *2017 IEEE International Conference on Big Data, BigData 2017, Boston, MA, USA, December 11-14, 2017*, 88–93.

Han, I.; Kambadur, P.; Park, K.; and Shin, J. 2017. Faster greedy map inference for determinantal point processes. *arXiv preprint arXiv:1703.03389*.

Han, I.; Malioutov, D.; and Shin, J. 2015. Large-scale log-determinant computation through stochastic Chebyshev expansions. In *International Conference on Machine Learning*, 908–917.

Jaynes, E. T. 1957a. Information theory and statistical mechanics. *Phys. Rev.* 106:620–630.

Jaynes, E. T. 1957b. Information theory and statistical mechanics. *Phys. Rev.* 106:620–630.

Jaynes, E. T. 1982. On the rationale of maximum-entropy methods. *Proceedings of the IEEE* 70(9):939–952.

Kang, U.; Meeder, B.; and Faloutsos, C. 2011. Spectral analysis for billion-scale graphs: Discoveries and implementation. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 13–25. Springer.

Kang, B. 2013. Fast determinantal point process sampling with application to clustering. In *Advances in Neural Information Processing Systems*, 2319–2327.

Kolmogorov, A. N. 1950. On logical foundations of probability theory. *Lecture Notes in Mathematics Probability Theory and Mathematical Statistics* 1–5.

Kulesza, A. 2012. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning* 5(2-3):123–286.

Leskovec, J., and Krevl, A. 2014. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`.

Li, C.; Jegelka, S.; and Sra, S. 2016. Fast dpp sampling for nystr\" om with application to kernel methods. *arXiv preprint arXiv:1603.06052*.

Lin, L.; Saad, Y.; and Yang, C. 2016. Approximating spectral densities of large matrices. *SIAM Review* 58(1):34–65.

Liu, J.; Wang, C.; Danilevsky, M.; and Han, J. 2013. Large-scale spectral clustering on graphs. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 1486–1492. AAAI Press.

Macchi, O. 1975. The coincidence approach to stochastic point processes. *Advances in Applied Probability* 7(1):83–122.

MacKay, D. J. 1992. *Bayesian methods for adaptive models*. Ph.D. Dissertation, California Institute of Technology.

MacKay, D. J. 2003. *Information theory, inference and learning algorithms*. Cambridge university press.

McGraw, P. N., and Menzinger, M. 2008. Laplacian spectra as a diagnostic tool for network structure and dynamics. *Physical Review E* 77(3):031102.

Mead, L. R., and Papanicolaou, N. 1984. Maximum entropy in the problem of moments. *Journal of Mathematical Physics* 25(8):2404–2417.

Neri, C., and Schneider, L. 2012. Maximum Entropy Distributions inferred from Option Portfolios on an Asset. *Finance and Stochastics* 16(2):293–318.

Newman, M. E. 2006. Finding community structure in networks using the eigenvectors of matrices. *Physical review E* 74(3):036104.

Pressé, S.; Ghosh, K.; Lee, J.; and Dill, K. A. 2013. Principles of Maximum Entropy and Maximum Caliber in Statistical Physics. *Reviews of Modern Physics* 85:1115–1141.

Rasmussen, C. E., and Williams, C. 2006. *Gaussian Processes for Machine Learning*. MIT Press.

Rasmussen, C. E. 2006. Gaussian processes for machine learning.

Rue, H., and Held, L. 2005. *Gaussian Markov random fields: theory and applications*. CRC press.

Shore, J., and Johnson, R. 1980. Axiomatic Derivation of the Principle of Maximum Entropy and the Principle of Minimum Cross-Entropy. *IEEE Transactions on information theory* 26(1):26–37.

Ubaru, S., and Saad, Y. Applications of trace estimation techniques.

Ubaru, S.; Chen, J.; and Saad, Y. 2016. Fast Estimation of tr (f (a)) via Stochastic Lanczos Quadrature.

Ubaru, S.; Chen, J.; and Saad, Y. 2017. Fast estimation of tr(f(a)) via stochastic Lanczos quadrature. *SIAM Journal on Matrix Analysis and Applications* 38(4):1075–1099.

Van Aelst, S., and Rousseeuw, P. 2009. Minimum volume ellipsoid. *Wiley Interdisciplinary Reviews: Computational Statistics* 1(1):71–82.

Von Luxburg, U. 2007. A tutorial on spectral clustering. *Statistics and computing* 17(4):395–416.

Wachinger, C., and Golland, P. 2015. Sampling from determinantal point processes for scalable manifold learning. In *International Conference on Information Processing in Medical Imaging*, 687–698. Springer.

Wainwright, M. J., and Jordan, M. I. 2006. Log-determinant relaxation for approximate inference in discrete markov random fields. *IEEE transactions on signal processing* 54(6):2099–2109.

Walley, P. 1991a. Statistical reasoning with imprecise probabilities.

Walley, P. 1991b. *Statistical reasoning with imprecise probabilities*. Chapman and Hall.

Yap, P.; Raveendran, P.; and Ong, S. 2001. Chebyshev moments as a new set of moments for image reconstruction. In *Neural Networks, 2001. Proceedings. IJCNN'01. International Joint Conference on*, volume 4, 2856–2860. IEEE.

Zhang, Y., and Leithead, W. E. 2007. Approximate implementation of the logarithm of the matrix determinant in gaussian process regression. *Journal of Statistical Computation and Simulation* 77(4):329–348.