Kernel Exponential Family Estimation via Doubly Dual Embedding

Bo Dai\textsuperscript{1} \hspace{5mm} Hanjun Dai\textsuperscript{2} \hspace{5mm} Arthur Gretton\textsuperscript{3} \hspace{5mm} Le Song\textsuperscript{2} \hspace{5mm} Dale Schuurmans\textsuperscript{1} \hspace{5mm} Niao He\textsuperscript{4}

\textsuperscript{1}Google Brain \hspace{5mm} \textsuperscript{2}Georgia Tech \hspace{5mm} \textsuperscript{3}UCL \hspace{5mm} \textsuperscript{4}UIUC

Abstract

We investigate penalized maximum log-likelihood estimation for exponential family distributions whose natural parameter resides in a reproducing kernel Hilbert space. Key to our approach is a novel technique, doubly dual embedding, that avoids computation of the partition function. This technique also allows the development of a flexible sampling strategy that amortizes the cost of Monte-Carlo sampling in the inference stage. The resulting estimator can be easily generalized to kernel conditional exponential families. We establish a connection between kernel exponential family estimation and MMD-GANs, revealing a new perspective for understanding GANs. Compared to the score matching based estimators, the proposed method improves both memory and time efficiency while enjoying stronger statistical properties, such as fully capturing smoothness in its statistical convergence rate while the score matching estimator appears to saturate. Finally, we show that the proposed estimator empirically outperforms state-of-the-art methods in both kernel exponential family estimation and its conditional extension.

1 Introduction

The exponential family is one of the most important classes of distributions in statistics and machine learning. The exponential family possesses a number of useful properties (Brown, 1986) and includes many commonly used distributions with finite-dimensional natural parameters, such as the Gaussian, Poisson and multinomial distributions, to name a few. It is natural to consider generalizing the richness and flexibility of the exponential family to an infinite-dimensional parameterization via reproducing kernel Hilbert spaces (RKHS) (Canu and Smola, 2006).

Maximum log-likelihood estimation (MLE) has already been well-studied in the case of finite-dimensional exponential families, where desirable statistical properties such as asymptotic unbiasedness, consistency and asymptotic normality have been established. However, it is difficult to extend MLE to the infinite-dimensional case. Beyond the intractability of evaluating the partition function for a general exponential family, the necessary conditions for maximizing log-likelihood might not be feasible; that is, there may exist no solution to the KKT conditions in the infinite-dimensional case (Pistone and Rogantin, 1999; Fukumizu, 2009). To address this issue, Barron and Shen (1991), Gu and Qiu (1993), Fukumizu (2009) considered several ways to regularize the function space by constructing (a series of) finite-dimensional spaces that approximate the original RKHS, yielding a tractable estimator in the restricted finite-dimensional space. However, as Sriperumbudur et al. (2017) note, even with the finite-dimension approximation, these algorithms are still expensive as every update requires Monte-Carlo sampling from a current model to compute the partition function.

An alternative score matching based estimator has recently been introduced by Sriperumbudur et al. (2017). This approach replaces the Kullback-Leibler (KL) with the Fisher divergence, defined by the expected squared distance between the score of the model (i.e., the derivative of the log-density) and the target distribution score (Hyvärinen, 2005). By minimizing the Tikhonov-regularized Fisher divergence, Sriperumbudur et al. (2017) develop a computable estimator for the infinite-dimensional exponential family that also obtains a consistency guarantee. Recently, this method has been generalized to conditional infinite-dimensional exponential family estimation (Arbel and Gretton, 2017). Although score matching avoids computing the generally intractable integral, it requires computing and saving the first- and second-order derivatives of the reproducing kernel for each
Kernel Exponential Family Estimation via Doubly Dual Embedding

dimension on each sample. For \( n \) samples with \( d \) features, this results in \( \mathcal{O}(n^2d^2) \) memory and \( \mathcal{O}(n^3d^3) \) time cost respectively, which becomes prohibitive for datasets with moderate sizes of \( n \) and \( d \). To alleviate this cost, Sutherland et al. (2017) utilize the \( m \)-rank Nystöm approximation to the kernel matrix, reducing memory and time complexity to \( \mathcal{O}(nmd^2) \) and \( \mathcal{O}(m^2d^3) \) respectively. Although this reduces the cost dependence on sample size, the dependence on \( d \) is unaffected, hence the estimator remains unsuitable for high-dimensional data. Estimating a general exponential family model by either MLE or score matching also generally requires some form of Monte-Carlo sampling, such as MCMC or HMC, to perform inference, which significantly adds to the computation cost.

In summary, both the MLE and score matching based estimators for the infinite-dimensional exponential family incur significant computational overhead, particularly in high-dimensional applications.

In this paper, we revisit penalized MLE for the kernel exponential family and propose a new estimation strategy. Instead of solving the log-likelihood equation directly, as in existing MLE methods, we exploit a doubly dual embedding technique that leads to a novel saddle-point reformulation for the MLE (along with its conditional distribution generalization) in Section 3. We then propose a stochastic algorithm for the new view of penalized MLE in Section 4. Since the proposed estimator is based on penalized MLE, it does not require the first- and second-order derivatives of the kernel, as in score matching, greatly reducing the memory and time cost. Moreover, since the saddle point reformulation of penalized MLE avoids the intractable integral, the need for a Monte-Carlo step is also bypassed, thus accelerating the learning procedure. This approach also learns a flexibly parameterized sampler simultaneously, therefore, further reducing inference cost. We present the consistency and rate of the new estimation strategy in the well-specified case, i.e., the true density belongs to the kernel exponential family, and an algorithm convergence guarantee in Section 5. We demonstrate the empirical advantages of the proposed algorithm in Section 6, comparing to state-of-the-art estimators for both the kernel exponential family and its conditional extension.

2 Preliminaries

We first provide a preliminary introduction to the exponential family and Fenchel duality, which will play vital roles in the derivation of the new estimator.

2.1 Exponential family

The natural form of the exponential family over \( \Omega \) with the sufficient statistics \( f \in \mathcal{F} \) is defined as

\[
p_f(x) = p_0(x) \exp(\lambda f(x) - A(\lambda f)),
\]

where \( A(\lambda f) := \log \int_{\Omega} \exp(\lambda f(x)) p_0(x) dx, x \in \Omega \subset \mathbb{R}^d, \lambda \in \mathbb{R}, \) and \( \mathcal{F} := \{ f \in \mathcal{H} : \exp(A(\lambda f)) < \infty \}. \) In this paper, we mainly focus on the case where \( \mathcal{H} \) is a reproducing Hilbert kernel space (RKHS) with kernel \( k(x, x') \), such that \( f(x) = \langle f, k(x, \cdot) \rangle \). However, we emphasize that the proposed algorithm in Section 4 can be easily applied to arbitrary differentiable function parametrizations, such as deep neural networks.

Given samples \( D = \{ x_i \}_{i=1}^N \), a model with a finite-dimensional parameterization can be learned via maximum log-likelihood estimation (MLE),

\[
\max_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^N \log p_f(x_i) = \hat{E}_D[\lambda f(x) + \log p_0(x)] - A(\lambda f),
\]

where \( \hat{E}_D[\cdot] \) denotes the empirical expectation over \( D \). The MLE (2) is well-studied and has nice properties. However, the MLE is known to be “ill-posed” in the infinite-dimensional case, since the optimal solution might not be exactly achievable in the representable space. Therefore, penalized MLE has been introduced for the finite (Dudık et al., 2007) and infinite dimensional Gu and Qiu (1993); Altun and Smola (2006) exponential families respectively. Such regularization essentially relaxes the moment matching constraints in terms of some norm, as shown later, guaranteeing the existence of a solution. In this paper, we will also focus on MLE with RKHS norm regularization.

One useful theoretical property of MLE for the exponential family is convexity w.r.t. \( f \). With convex regularization, e.g., \( \| f \|_H^2 \), one can use stochastic gradient descent to recover a unique global optimum. Let \( L(f) \) denote RKHS norm penalized log-likelihood, i.e.,

\[
L(f) := \frac{1}{N} \sum_{i=1}^N \log p_f(x_i) - \eta \frac{1}{2} \| f \|^2_H.
\]

where \( \eta > 0 \) denotes the regularization parameter. The gradient of \( L(f) \) w.r.t. \( f \) can be computed as

\[
\nabla_f L = \hat{E}_D[\lambda \nabla_f f(x)] - \nabla_f A(\lambda f) - \eta f.
\]

To calculate the \( \nabla_f A(\lambda f) \) in (3), we denote \( Z(\lambda f) = \int_{\Omega} \exp(\lambda f(x)) p_0(x) dx \) and expand the partition function by definition,

\[
\nabla_f A(\lambda f) = \frac{1}{Z(\lambda f)} \nabla_f \int_{\Omega} \exp(\lambda f(x)) p_0(x) dx = \int_{\Omega} p_0(x) \exp(\lambda f(x)) \nabla_f \lambda f(x) dx = E_{p_0(x)}[\nabla_f \lambda f(x)].
\]

One can approximate the \( E_{p_0(x)}[\nabla_f \lambda f(x)] \) by AIS/MCMC samples (Vembu et al., 2009), which leads to the Contrastive Divergence (CD) algorithm (Hinton, 2002).

To avoid costly MCMC sampling in estimating the gradient, Srivastava et al. (2017) construct an esti-
mator based on score matching instead of MLE, which minimizes the penalized Fisher divergence. Plugging the kernel exponential family into the empirical Fisher divergence, the optimization reduces to
\[ J(f) := \frac{\lambda^2}{2} \langle \hat{C} f, f \rangle_{\mathcal{H}} + \lambda \langle f, \hat{\delta} \rangle_{\mathcal{H}} + \frac{\eta}{2} \| f \|^2_{\mathcal{H}}, \]  
where
\[ \hat{C} := \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{d} \partial_i \log p_0(x_i) \odot \partial_j \log p_0(x_i), \]
\[ \hat{\delta} := \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{d} \partial_i \log p_0(x_i) \left( \partial_j \log p_0(x_i) + \partial^2_j \log p_0(x_i) \right). \]

As we can see, the score matching objective is convex and does not involve the intractable integral in \( A(\lambda f) \). However, such an estimator requires the computation of first- and second-order of derivatives of kernel for each dimension on each data, leading to memory and time cost of \( O(n^2d^2) \) and \( O(n^3d^3) \) respectively. This quickly becomes prohibitive for even modest \( n \) and \( d \).

The same difficulty also appears in the score matching based estimator in (Arbel and Gretton 2017) for the conditional exponential family, which is defined as
\[ p(y|x) = p_0(y) \exp(\lambda f(x, y) - A_\lambda(f(x,y))), \quad f \in \mathcal{F} \]  
where \( y \in \mathcal{Y}_y \subset \mathbb{R}^p, x \in \mathcal{X}_x \subset \mathbb{R}^d, \lambda \in \mathbb{R}, \) and \( A_\lambda(f) := \log \int_{\mathcal{Y}_y} p_0(y) \exp(\lambda f(x, y)) dy \), \( \mathcal{F} \) is defined as \( \{ f \in \mathcal{H}_y : \exp(\lambda f(x,y)) < \infty \} \). We consider \( f : \mathcal{X}_x \times \mathcal{Y}_y \rightarrow \mathbb{R} \) such that \( f(x,\cdot) \) is in RKHS \( \mathcal{H}_y \) for \( \forall x \in \mathcal{X}_x \). Denoting \( \mathcal{T} \in \mathcal{H}_{\mathcal{T}_x} : \mathcal{X}_x \rightarrow \mathcal{H}_y \) such that \( \mathcal{T}_x(y) = f(x,y) \), we can derive its kernel function following (Micchelli and Pontil 2005; Arbel and Gretton 2017). By the Riesz representation theorem, \( \forall x \in \mathcal{X}_x \) and \( h \in \mathcal{H}_y \), there exists a linear operator \( \Gamma_x : \mathcal{H}_y \rightarrow \mathcal{H}_y \) such that
\[ \langle h, \Gamma_x h \rangle_{\mathcal{H}_y} = \langle \mathcal{T}_x, h \rangle_{\mathcal{H}_{\mathcal{T}_x}}, \quad \forall \mathcal{T} \in \mathcal{H}_{\mathcal{T}_x}. \]

Then, the kernel can be defined by composing \( \Gamma_x \) with its dual, i.e., \( k(x',x) = \Gamma_x^* \Gamma_x \) and the function \( f(x,y) = \mathcal{T}_x(y) = \langle \mathcal{T}_x, \Gamma_x k(y,\cdot) \rangle \). We follow such assumptions for kernel conditional exponential family.

2.2 Convex conjugate and Fenchel duality

Denote \( h(\cdot) \) as a function \( \mathbb{R}^d \rightarrow \mathbb{R} \), then its convex conjugate function is defined as
\[ h^*(u) = \sup_{v \in \mathbb{R}^d} \{ u^T v - h(v) \}. \]

If \( h(\cdot) \) is proper, convex and lower semicontinuous, the conjugate function, \( h^*(\cdot) \), is also proper, convex and lower semicontinuous. Moreover, \( h \) and \( h^* \) are dual to each other, i.e., \( (h^*)^* = h \). Such a relationship is known as Fenchel duality (Rockafellar 1970; Hiriart-Urruty and Lemaréchal 2012). By the conjugate function, we can represent the \( h \) by as,
\[ h(v) = \sup_{u \in \mathbb{R}^d} \{ u^T v - h^*(u) \}. \]

The supremum achieves if \( v \in \partial h^*(u) \), or equivalently \( u \in \partial h(v) \).

3 A Saddle-Point Formulation of Penalized MLE

As discussed in Section 2, the penalized MLE for the exponential family involves computing the log-partition functions, \( A(\lambda f) \) and \( A_x(\lambda f) \), which are intractable in general. In this section, we first introduce a saddle-point reformulation of the penalized MLE of the exponential family, using Fenchel duality to bypass the computation of the intractable log-partition function. This approach can also be generalized to the conditional exponential family. First, observe that we can rewrite the log-partition function \( A(\lambda f) \) via Fenchel duality as follows.

**Theorem 1 (Fenchel dual of log-partition)**
\[ A(\lambda f) = \max_{q \in \mathcal{P}} \lambda \langle q(x), f(x) \rangle - KL(q\|p_0), \quad (7) \]
\[ p_f(x) = \arg \max_{q \in \mathcal{P}} \lambda \langle q(x), f(x) \rangle - KL(q\|p_0), \quad (8) \]
where \( \langle q, g \rangle := \int \omega(x) q(x) dx \), \( \mathcal{P} \) denotes the space of distributions and \( KL(q\|p_0) := \int \omega(x) \log \frac{q(x)}{p_0(x)} dx \).

**Proof** Denote \( l(q) := \lambda \langle q(x), f(x) \rangle - KL(q\|p_0) \), which is strongly concave w.r.t. \( q \in \mathcal{P} \), the optimal \( \hat{q}^* \) can be obtained by setting the
\[ \log q^*(x) \propto \lambda f(x) + \log p_0(x). \]

Since \( \hat{q}^* \in \mathcal{P} \), we have
\[ q^*(x) = p_0(x) \exp(\lambda f(x) - A(\lambda f)) = p_f(x), \]
which leads to (9). Plugging \( q^* \) to \( l(q) \), we obtain the maximum as \( \log \int \omega(x) \exp(\lambda f(x)) p_0(x) dx \), which is exactly \( A(\lambda f) \), leading to (7).

Therefore, invoking the Fenchel dual of \( A(\lambda f) \) into the penalized MLE, we achieve a saddle-point optimization, i.e.,
\[ \max_{f \in \mathcal{F}} L(f) \propto \min_{q \in \mathcal{P}} \mathbb{E}_{P_f} [f(x) - \mathbb{E}_q[f(x)] - \frac{\eta}{2} \| f \|^2_{\mathcal{H}} + \frac{1}{\lambda} KL(q\|p_0)]. \]

The saddle-point formulation of the penalized MLE in (9) resembles the optimization in MMD GAN (Liu et al. 2017; Binkowski et al. 2018). In fact, the dual problem of the penalized MLE of the exponential family is a \( KL \)-regularized MMD GAN with a special design of the kernel family. Alternatively, if \( f \) is a Wasserstein-1 function, the optimization resembles the Wasserstein GAN (Arjovsky et al. 2017).

Next, we consider the duality properties.

**Theorem 2 (weak and strong duality)** The weak duality holds in general, i.e.,
\[ \max_{f \in \mathcal{F}} \min_{q \in \mathcal{P}} \ell(f, q) \leq \min_{q \in \mathcal{P}} \max_{f \in \mathcal{F}} \ell(f, q). \]

The strong duality holds when \( F \) is a closed RKHS and
\( \mathcal{P} \) is the distributions with bounded \( L_2 \) norm, i.e.,
\[
\max_{f \in \mathcal{P}} \min_{q \in \mathcal{P}} \ell(f, q) = \min_{q \in \mathcal{P}} \max_{f \in \mathcal{P}} \ell(f, q).
\]
(10)

Theorem 2 can be obtained by directly applying the minimax theorem [Ekeland and Temam 1999][Proposition 2.1]. We refer to the max-min problem in (10) as the primal problem, while the min-max form as the its dual form.

Remark (connections to MMD GAN): Consider the dual problem with the kernel learning, i.e.,
\[
\min_{q \in \mathcal{P}} \max_{f \in \mathcal{P}, \phi} \mathbb{E}_q[f(x)] - \mathbb{E}_q[f(x)] - \frac{\eta}{2} \|f\|_H^2 + \frac{1}{\lambda} KL(q\|p_0).
\]
(11)
where we involve the parameters of the kernel \( \phi \) to be learned in the optimization. By setting the gradient \( \nabla f \ell(f, q) = 0 \), for fixed \( q \), we obtain the optimal witness function in the RKHS, \( f_q = \frac{1}{\eta} \left( \mathbb{E}_q[k_\phi(x, \cdot)] - \mathbb{E}_q[k_\phi(x, \cdot)] \right) \), which leads to
\[
\min_{q \in \mathcal{P}} \max_{\phi} \left[ \sum_{i=1}^{MMD_\phi(D, q)} \right] - 2\mathbb{E}_q[k_\phi(x, x')] + \mathbb{E}_q[k_\phi(x, x')]
\]
(12)
This can be regarded as the \( KL \)-divergence regularized MMD GAN. Thus, with \( KL \)-divergence regularization, the MMD GAN learns an infinite-dimension exponential family in an adaptive RKHS. Such a novel perspective bridges GAN and exponential family estimation, which appears to be of independent interest and potentially brings a new connection to the GAN literature for further theoretical development.

\[
\min_{q \in \mathcal{P}} KL(q\|p_0)
\]
s.t.
\[
\left\| \mathbb{E}_q[k(x, \cdot)] - \mathbb{E}_q[k(x, \cdot)] \right\|_{H_k} \leq \frac{\eta'}{2},
\]
whose dual problem will be reduced to the penalized MLE [2] with proper choice of \( \eta' \) [Altun and Smola 2006][Lemma 6]. Interestingly, the proposed saddle-point formulation [Dai et al. 2016] shares the solution to (13). From the maximum entropy view, the penalty \( \|f\|_H^2 \) relaxes the moment matching constraints. However, the algorithms provided in [Altun and Smola 2006; Dudik et al. 2007] simply ignore the difficulty in computing the expectation in \( A(\lambda f) \), which is not practical, especially when \( f \) is infinite-dimensional.

Similar to Theorem 2, we can also represent \( A_x(\lambda f) \) by its Fenchel dual,
\[
A_x(\lambda f) = \max_{q \in \mathcal{P}} \lambda \langle q(y|x), f(x, y) \rangle - KL(q\|p_0).
\]
(14)
\[
p_f(y|x) = \arg\max_{q \in \mathcal{P}} \lambda \langle q(x), f(x) \rangle - KL(q\|p_0).
\]
(15)
Then, we can recover the penalized MLE for the conditional exponential family as
\[
\max_{f \in \mathcal{P}} \min_{q \in \mathcal{P}} \mathbb{E}_D[f(x, y)] - \mathbb{E}_q(y|x) f(x, y)
\]
\[
- \frac{\eta}{2} \|f\|_H^2 + \frac{1}{\lambda} KL(q\|p_0).
\]
(16)
The saddle-point reformulations of penalized MLE in [1] and [10] bypass the difficulty in the partition function. Therefore, it is very natural to consider learning the exponential family by solving the saddle-point problem [2] with an appropriate parametrized dual distribution \( q(x) \). This approach is referred to as the “dual embedding” technique in [Dai et al. 2016], which requires:

i) the parametrization family should be flexible enough to reduce the extra approximation error;
ii) the parametrized representation should be able to provide density value.

As we will see in Section 5 the flexibility of the parametrization of the dual distribution will have a significant effect on the consistency of the estimator.

One can of course use the kernel density estimator (KDE) as the dual distribution parametrization, which preserves convex-concavity. However, KDE will easily fail when approximating high-dimensional data. Applying the reparametrization trick with a suitable class of probability distributions [Kingma and Welling 2013; Rezende et al. 2014] is another alternative to parametrize the dual distribution. However, the class of such parametrized distributions is typically restricted to simple known distributions, which might not be able to approximate the true solution, potentially leading to a huge approximation bias. At the other end of the spectrum, the distribution family generated by transport mapping is sufficiently flexible to model smooth distributions [Goodfellow et al. 2014; Arjovsky et al. 2017]. However, the density value of such distributions, i.e., \( q(x) \), is not available for \( KL \)-divergence computation, and thus, is not applicable for parameterizing the dual distribution. Recently, flow-based parametrized density functions [Rezende and Mohamed 2015; Kingma et al. 2016; Dinh et al. 2016] have been proposed for a trade-off between the flexibility and tractability. However, the expressive power of existing flow-based models remains restrictive even in our synthetic example and cannot be directly applied to conditional models.

3.1 Doubly Dual Embedding for MLE
Transport mapping is very flexible for generating smooth distributions. However, a major difficulty is that it lacks the ability to obtain the density value \( q(x) \), making the computation of the \( KL \)-divergence impossible. To retain the flexibility of transport mapping and avoid the calculation of \( q(x) \), we introduce
Algorithm 1 Doubly Dual Embedding-SGD for saddle-point reformulation of MLE [19]

1: for \( l = 1, \ldots, L \) do
2: Compute \((f, \nu)\) by Algorithm \[2\]
3: Sample \( \{\xi_b \sim p(\xi)\}_{b=1}^B \)
4: Generate \( x_b = gw_{\xi_b}(\xi) \) for \( b = 1, \ldots, B \).
5: Compute stochastic approximation \( \hat{\nabla}_{w_y} \hat{L}(w_g) \)
6: Update \( w_g^{l+1} = w_g^l - p_l \hat{\nabla}_{w_y} \hat{L}(w_g) \).
7: end for
8: Output \( w_g \) and \( f \).

**doubly dual embedding**, which achieves a delicate balance between flexibility and tractability.

First, noting that KL-divergence is also a convex function, we consider the Fenchel dual of KL-divergence [Nguyen et al., 2008], i.e.,

\[
KL(q\|p_0) = \max_{\nu} \mathbb{E}_q[\nu(x)] - \mathbb{E}_{p_0}[\exp(\nu(x))] + 1, \quad (17)
\]

\[
\log q(x) = \arg\max_{\nu} \mathbb{E}_q[\nu(x)] - \mathbb{E}_{p_0}[\exp(\nu(x))] + 1, \quad (18)
\]

with \( \nu(\cdot) : \Omega \to \mathbb{R} \). One can see in the dual representation of the KL-divergence that the introduction of the auxiliary optimization variable \( \nu \) eliminates the explicit appearance of \( q(x) \) in (17), which makes the transport mapping parametrization for \( q(x) \) applicable. Since there is no extra restriction on \( \nu \), we can use arbitrary smooth function approximation for \( \nu(\cdot) \), such as kernel functions or neural networks.

Applying the dual representation of KL-divergence into the dual problem of the saddle-point reformulation [19], we obtain the ultimate saddle-point optimization for the estimation of the kernel exponential family,

\[
\min_{q \in \mathcal{P}} \max_{f, \nu \in \mathcal{F}} \hat{\ell}(f, \nu, q) := \mathbb{E}_D[f] - \mathbb{E}_q[f] - \frac{\eta}{2} \|f\|_H^2 + \frac{1}{\lambda} \left( \mathbb{E}_q[\nu] - \mathbb{E}_{p_0}[\exp(\nu)] \right). \quad (19)
\]

Note that several other work also applied Fenchel duality to KL-divergence [Nguyen et al., 2008; Nowozin et al., 2010], but the approach taken here is different as it employs duals of \( KL(q\|p_0) \), rather than \( KL(q\|D) \).

**Remark (Extension for kernel conditional exponential family):** Similarly, we can apply the doubly dual embedding technique to the penalized MLE of the kernel conditional exponential family, which leads to

\[
\min_{q \in \mathcal{P}, f, \nu \in \mathcal{F}} \mathbb{E}_D[f] - \mathbb{E}_{q_g(x), x \sim D}[f] - \frac{\eta}{2} \|f\|_H^2 + \frac{1}{\lambda} \left( \mathbb{E}_q[\nu(x), x \sim D][\nu] - \mathbb{E}_{p_0}(\nu) [\exp(\nu)] \right). \quad (20)
\]

Algorithm 2 Stochastic Functional Gradients for \( f \) and \( \nu \)

1: for \( k = 1, \ldots, K \) do
2: Sample \( \xi \sim p(\xi) \).
3: Generate \( x = g(\xi) \).
4: Sample \( x' \sim p_0(x) \).
5: Compute stochastic function gradient w.r.t. \( f \) and \( \nu \) with (21) and (21).
6: Update \( f_k \) and \( \nu_k \) with (22) and (22).
7: end for
8: Output \( (f_K, \nu_K) \).

## 4 Practical Algorithm

In this section, we introduce the transport mapping parametrization for the dual distribution, \( q(x) \), and apply the stochastic gradient descent for solving the optimization problems in (19) and (20). For simplicity of exposition, we only illustrate the algorithm for (19). The algorithm can be easily applied to (20).

Denote the parameters in the transport mapping for \( q(x) \) as \( w_q \), such that \( \xi \sim p(\xi) \) and \( x = g_{w_q}(\xi) \). We illustrate the algorithm with a kernel parametrize \( \nu(\cdot) \).

There are many alternative choices of parametrization for \( \nu \), e.g., neural networks—the proposed algorithm is still applicable to the parametrization as long as it is differentiable. We abuse notation somewhat by using \( \hat{\ell}(f, \nu, w_q) \) as \( \hat{\ell}(f, \nu, q) \) in (19), with such parametrization, the (19) is an upper bound of the penalty MLE in general by Theorem 2.

With the kernel parametrized \((\nu, \nu)\), the inner maximization over \( f \) and \( \nu \) is a standard concave optimization. We can solve it using existing algorithms to achieve the global optimal solution. Due to the existence of the expectation, we will use the stochastic functional gradient descent for scalability. Given \((f, \nu) \in \mathcal{H} \), following the definition of functional gradients [Kivinen et al., 2004; Dai et al., 2014], we have

\[
\zeta_f(\cdot) := \nabla_f \hat{\ell}(f, \nu, w_q) = \mathbb{E}_D[k(x, \cdot)] - \mathbb{E}_x[k(g_{w_q}(\xi), \cdot)] - \eta f(\cdot). \quad (20)
\]

\[
\zeta_{\nu}(\cdot) := \nabla_{\nu} \hat{\ell}(f, \nu, w_q) = \frac{1}{\lambda} \left( \mathbb{E}_x[k(g_{w_q}(\xi), \cdot)] - \mathbb{E}_x[k(g_{w_q}(\xi), \cdot)] \right). \quad (21)
\]

In the \( k \)-th iteration, given sample \( x \sim D, \xi \sim p(\xi) \), and \( x' \sim p_0(x) \), the update rule for \( f \) and \( \nu \) will be

\[
f_{k+1}(\cdot) = (1 - \eta \tau_k) f_k(\cdot) + \tau_k (k(x, \cdot) - k(g_{w_q}(\xi), \cdot)), \quad \nu_{k+1}(\cdot) = \nu_k(\cdot) + \frac{\tau_k}{\lambda} (k(g_{w_q}(\xi), \cdot) - \exp(\nu_k(x')) k(x', \cdot)), \quad (22)
\]

where \( \tau_k \) denotes the step-size.

Then, we consider the update rule for the parameters in dual transport mapping embedding.

**Theorem 3 (Dual gradient)** Denoting
\[ \left( f^*_w, \nu^*_w \right) = \argmax_{(f, \nu) \in \mathcal{H}} \hat{L}(f, \nu, w) \quad \text{and} \]

\[ L(w_g) = \hat{L}(f^*_w, \nu^*_w, w_g), \]

we have

\[
\nabla_{w_g} \hat{L}(w_g) = -\mathbb{E}_\xi \left[ \nabla_{w_g} f^*_w \left( g_{w_g}(\xi) \right) \right] + \frac{1}{\lambda} \mathbb{E}_\xi \left[ \nabla_{w_g} \nu^*_w \left( g_{w_g}(\xi) \right) \right].
\]

Proof details are given in Appendix A. With this gradient estimator, we can apply stochastic gradient descent to update \( w_g \) iteratively. We summarize the updates for \( (f, \nu) \) and \( w_g \) in Algorithm 1 and 2.

Compared to score matching based estimators (Sriperumbudur et al., 2017; Sutherland et al., 2017), although the convexity no longer holds for the saddle-point estimator, the doubly dual embedding estimator avoids representing \( f \) with derivatives of the kernel, thus avoiding the memory cost dependence on the square of dimension. In particular, the proposed estimator for \( f \) based on (13) reduces the memory cost from \( O(n^2d^2) \) to \( O(K^2) \) where \( K \) denotes the number of iterations in Algorithm 2. In terms of time cost, we exploit the stochastic update, which is naturally suitable for large-scale datasets and avoids the matrix inverse computation in score matching estimator whose cost is \( O(n^4d^3) \). We also learn the dual distribution simultaneously, which can be easily used to generate samples from the exponential family for inference, thus saving the cost of Monte-Carlo sampling in the inference stage. For detailed discussion about the computation cost, please refer to Appendix D.

**Remark (random feature extension):** Memory cost is a well-known bottleneck for applying kernel methods to large-scale problems. When we set \( K = n \), the memory cost will be \( O(n^2) \), which is prohibitive for millions of data points. Random feature approximation (Rahimi and Recht, 2008; Dai et al., 2014; Bach, 2015) can be utilized for scaling up kernel methods. The proposed Algorithm 1 and Algorithm 2 are also compatible with random feature approximation, and hence, applicable to large-scale problems in the same way. With \( r \) random features, we can further reduce the memory cost of storing \( f \) to \( O(rd^2) \). However, even with a random feature approximation, the score matching based estimator will still require \( O(rd^2) \) memory. One can also learn random features by back-propagation, which leads to the neural networks extension. Due to space limitations, details of this variant of Algorithm 1 are given in Appendix D.

## 5 Theoretical Analysis

In this section, we will first provide the analysis of consistency and the sample complexity of the proposed estimator based on the saddle-point reformulation of the penalized MLE in the well-specified case, where the true density is assumed to be in the kernel (conditional) exponential family, following Sutherland et al. (2017). Then, we consider the convergence property of the proposed algorithm.

### 5.1 Statistical Consistency

We explicitly consider approximation error from the dual embedding in the consistency of the proposed estimator. We first establish some notation that will be used in the analysis. For simplicity, we set \( \lambda = 1 \) and \( p_0(x) = 1 \) improperly in the exponential family expression (1). We denote \( p^*(x) = \exp(f^*(x) - A(f^*)) \) as the ground-truer distribution with its true potential function \( f^* \) and \( \hat{f}(\hat{q}, \hat{v}) \) as the optimal primal and dual solution to the saddle point reformulation of the penalized MLE (1). The parametrized dual space is denoted as \( \mathcal{P}_w \). We denote \( p_f := \exp\left( \lambda^T - A(\lambda^T f) \right) \) as the exponential family generated by \( f \). We have the consistency results as

**Theorem 4** Assume the spectrum of kernel \( k(\cdot, \cdot) \) decays sufficiently homogeneous in rate \( l^{-r} \). With some other mild assumptions listed in Appendix A.2, we have as \( n \to 0 \) and \( \eta \to 0 \),

\[
KL\left( p^* || p_f \right) + KL\left( p_f || p^* \right) = O_{\text{p}}\left( n^{-1} \eta^{-\frac{1}{2}} + \eta + \epsilon^{2}_{\text{approx}} \right),
\]

where \( \epsilon_{\text{approx}} := \sup_{f \in \mathcal{F}} \inf_{q \in \mathcal{P}_w} \| p_f - q \|_{p^*} \). Therefore, when setting \( \eta = O\left( n^{-\frac{1}{2r}} \right) \), \( p_f \) converges to \( p^* \) in terms of Jensen-Shannon divergence at rate \( O_{\text{p}}\left( n^{-\frac{1}{2r}} + \epsilon^{2}_{\text{approx}} \right) \).

For the details of the assumptions and the proof, please refer to Appendix A.2. Recall the connection between the proposed model and MMD GAN as discussed in Section 3. Theorem 1 also provides a learnability guarantee for a class of GAN models as a byproduct. The most significant difference of the bound provided above, compared to Gu and Qiu (1993); Altun and Smola (2006), is the explicit consideration of the bias from the dual parametrization. Moreover, instead of the Rademacher complexity used in the sample complexity results of Altun and Smola (2006), our result exploits the spectral decay of the kernel, which is more directly connected to properties of the RKHS.

From Theorem 3 we can clearly see the effect of the parametrization of the dual distribution: if the parametric family of the dual distribution is simple, the optimization for the saddle-point problem may become easier, however, \( \epsilon_{\text{approx}} \) will dominate the error. The other extreme case is to also use the kernel exponential family to parametrize the dual distribution, then, \( \epsilon_{\text{approx}} \) will reduce to 0, however, the optimization will be difficult to handle. The saddle-point reformulation...
provides us the opportunity to balance the difficulty of the optimization with approximation error.

The statistical consistency rate of our estimator and the score matching estimator in [Sriperumbudur et al., 2017] are derived under different assumptions, therefore, they are not directly comparable. However, since the smoothness is not fully captured by the score matching based estimator in [Sriperumbudur et al., 2017], it only achieves $O\left( n^{-\frac{3}{2}} \right)$ even if $f$ is infinitely smooth. While under the case that dual distribution parametrization is relative flexible, i.e., $\epsilon_{\text{approx}}$ is negligible, and the the spectrum of the kernel decay rate $r \rightarrow \infty$, the proposed estimator will converge in rate $O\left( n^{-1} \right)$, which is significantly more efficient than the score matching method.

5.2 Algorithm Convergence

It is well-known that the stochastic gradient descent converges for saddle-point problem with convex-concave property [Nemirovski et al., 2009]. However, for better the dual parametrization to reduce $\epsilon_{\text{approx}}$ in Theorem 3, we parameterize the dual distribution with the nonlinear transport mapping, which breaks the convexity. In fact, by Theorem 3, we obtain the unbiased gradient w.r.t. $w_g$. Therefore, the proposed Algorithm 4 can be understood as applying the stochastic gradient descent for the non-convex dual minimization problem, i.e., $\text{min}_{w_g} \hat{L}(w_g) := \mathbb{E} \left[ L(w_g, \nu_{w_g}, y) \right]$. From such a view, we can prove the sublinearly convergence rate to a stationary point when stepsize is diminishing following [Ghadimi and Lan, 2013, Dai et al., 2017]. We list the result below for completeness.

**Theorem 5** Assume that the parametrized objective $\hat{L}(w_g)$ is $C$-Lipschitz and variance of its stochastic gradient is bounded by $\sigma^2$. Let the algorithm run for $L$ iterations with stepsize $\rho_t = \min \{ t^{-\frac{1}{4}}, (D')^{-\frac{1}{2}} \}$ for some $D' > 0$ and output $w_g^1, \ldots, w_g^L$. Setting the candidate solution to be $\tilde{w}_g$ randomly chosen from $w_g^1, \ldots, w_g^L$ such that $P(w = w_g^1) = \frac{2\rho_1 - C \rho^2}{L}$, then it holds that $E \left[ \left\| \nabla \hat{L}(\tilde{w}_g) \right\|^2 \right] \leq \frac{C^2}{L} + (D' + \frac{D}{L}) \frac{\sigma^2}{\sqrt{L}}$ where

$D := \sqrt{2(\hat{L}(w_g^1) - \min \hat{L}(w_g)) / L}$ represents the distance of the initial solution to the optimal solution.

The above result implies that under the choice of the parametrization of $f, \nu$ and $g$, the proposed Algorithm 4 converges sublinearly to a stationary point, whose rate will depend on the smoothing parameter.

6 Experiments

In this section, we compare the proposed doubly dual embedding (DDE) with the current state-of-the-art score matching estimators for kernel exponential family (KEF) [Sutherland et al., 2017] and its conditional extension (KECEF) [Arbel and Gretton, 2017], respectively, as well as several competitors. We test the proposed estimator empirically following their setting. We use Gaussian RBF kernel for both exponential family and its conditional extension, $k(x, x') = \exp\left(-\|x - x'\|^2 / \sigma^2\right)$, with the bandwidth $\sigma$ set by median-trick [Dai et al., 2014]. For a fair comparison, we follow Sutherland et al. [2017] and Arbel and Gretton [2017] to set the $p_0(x)$ for kernel exponential family and its conditional extension, respectively. The dual variables are parametrized by MLP with 5 layers. More implementation details can be found in Appendix C and the code repository which is available at https://github.com/MichaelArbel/KCEF.

**Density estimation** We evaluate the DDE on the synthetic datasets, including ring, grid and two moons, where the first two are used in Sutherland et al. [2017], and the last one is from Rezende and Mohamed [2015]. The ring dataset contains the points uniformly sampled along three circles with radii $(1, 3, 5) \in \mathbb{R}^2$ and $N(0, 0.1^2)$ noise in the radial direction and extra dimensions. The d-dim grid dataset contains samples from mixture of $d$ Gaussians. Each center lies on one dimension in the $d$-dimension hypercube. The two moons dataset is sampled from the exponential family with potential function as $\frac{1}{2} \left( \frac{\|x\|^2}{\sigma} - 1 \right)^2 - 2 \log \left( \exp\left(-\frac{1}{2} \left( \frac{x_0^2}{\sigma} \right) + \exp\left(-\frac{1}{2} \left( \frac{x_0^2}{\sigma} \right) 1 \right) \right) \right)$. We use 500 samples for training, and for testing 1500 (grid) or 5000 (ring, two moons) samples, following Sutherland et al. [2017].

We visualize the samples generated by the learned sampler, and compare it with the training datasets in the first row in Figure 1. The learned $f$ is also plotted in the second row in Figure 1. The DDE learned models generate samples that cover the training data,

---

1https://github.com/karlnapf/nystrom-kexpfam
2https://github.com/MichaelArbel/KCEF

---

Figure 1: The blue points in the figures in first row show the generated samples by learned models, and the red points are the training samples. The learned $f$ are illustrated in the second row.
The data and randomly split the datasets into a train-
Sugiyama et al., 2010). We centered and normalized
for studying these methods (Arbel and Gretton, 2017;
conditional distribution
experiment, models are trained to estimate the condi-
Conditional model extension.

Then, we demonstrate the convergence of the DDE in Figure 2 on rings dataset. We initialize with a ran-
distribution. As the algorithm iterates, the
convergence guarantees. More results on 2-dimensional grid and two moons can be found in Figure 3 in Appendix C. The DDE algorithm behaves similarly on these two datasets.

Finally, we compare the MMD between the gener-
ated samples from the learned model to the train-
ing data with the current state-of-the-art method for
KEF (Sutherland et al., 2017), which performs better
than KDE and other alternatives (Strathmann et al.
2015). We use HMC to generate the samples from the
KEF learned model, while in the DDE, we can bypass
the HMC step by the learned sampler. The computa-
tion time for inference is listed in Table 1. It shows the
DDE improves the inference efficiency in orders. The
MMD comparison is listed in Table 1. The proposed
DDE estimator performs significantly better than the
best performances by KEF in terms of MMD.

**Table 2: The negative log-likelihood comparison on benchmarks.** Mean and std are calculated on 20 runs of different train/test splits. KCEF gets numerically unstable on two datasets, where we mark as N/A.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>MMD time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>two moons</td>
<td>DDE KCEF KCEF-sample KCEF+HMC</td>
</tr>
<tr>
<td>ring</td>
<td>0.11 1.32 0.07 13.04</td>
</tr>
<tr>
<td>grid</td>
<td>1.53 3.19 0.07 14.12</td>
</tr>
</tbody>
</table>

Figure 2: The DDE estimators on rings dataset in each iteration. The blue points are sampled from the learned model. With the algorithm proceeds, the learned distribution converges to the ground-truth.

Table 1: Quantitative evaluation on synthetic data. MMD in $\times 10^{-3}$ scale on test set is reported.

**Table 2: The negative log-likelihood comparison on benchmarks.** Mean and std are calculated on 20 runs of different train/test splits. KCEF gets numerically unstable on two datasets, where we mark as N/A.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>DDE</th>
<th>KCEF</th>
<th>$\epsilon$-KDE</th>
<th>LSCDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>geyser</td>
<td>0.55 ± 0.07</td>
<td>1.21 ± 0.04</td>
<td>1.11 ± 0.02</td>
<td>0.7 ± 0.01</td>
</tr>
<tr>
<td>caution</td>
<td>0.95 ± 0.19</td>
<td>0.99 ± 0.01</td>
<td>1.25 ± 0.19</td>
<td>1.19 ± 0.02</td>
</tr>
<tr>
<td>floodsimnow</td>
<td>1.49 ± 0.14</td>
<td>1.46 ± 0.00</td>
<td>1.53 ± 0.05</td>
<td>1.56 ± 0.01</td>
</tr>
<tr>
<td>snowgrees</td>
<td>0.42 ± 0.31</td>
<td>0.72 ± 0.02</td>
<td>1.35 ± 0.17</td>
<td>1.39 ± 0.05</td>
</tr>
<tr>
<td>GAGurine</td>
<td>0.43 ± 0.15</td>
<td>0.46 ± 0.00</td>
<td>0.92 ± 0.05</td>
<td>0.7 ± 0.01</td>
</tr>
<tr>
<td>topo</td>
<td>1.02 ± 0.31</td>
<td>0.67 ± 0.01</td>
<td>1.18 ± 0.09</td>
<td>0.83 ± 0.00</td>
</tr>
<tr>
<td>CohartOre</td>
<td>1.45 ± 0.23</td>
<td>3.42 ± 0.03</td>
<td>1.65 ± 0.09</td>
<td>1.61 ± 0.02</td>
</tr>
<tr>
<td>mcycle</td>
<td>0.60 ± 0.15</td>
<td>0.56 ± 0.01</td>
<td>1.25 ± 0.23</td>
<td>0.93 ± 0.01</td>
</tr>
<tr>
<td>BigMac2003</td>
<td>0.47 ± 0.36</td>
<td>0.59 ± 0.01</td>
<td>1.29 ± 0.14</td>
<td>1.63 ± 0.03</td>
</tr>
<tr>
<td>cups</td>
<td>-0.63 ± 0.77</td>
<td>N/A</td>
<td>1.01 ± 0.10</td>
<td>1.04 ± 0.07</td>
</tr>
<tr>
<td>crabs</td>
<td>-0.60 ± 0.26</td>
<td>N/A</td>
<td>0.99 ± 0.09</td>
<td>-0.07 ± 0.11</td>
</tr>
<tr>
<td>birthwt</td>
<td>1.22 ± 0.15</td>
<td>1.18 ± 0.13</td>
<td>1.48 ± 0.01</td>
<td>1.43 ± 0.01</td>
</tr>
<tr>
<td>gilpigs</td>
<td>0.61 ± 0.10</td>
<td>0.65 ± 0.08</td>
<td>1.35 ± 0.03</td>
<td>0.73 ± 0.05</td>
</tr>
<tr>
<td>UN3</td>
<td>1.03 ± 0.09</td>
<td>1.15 ± 0.21</td>
<td>1.78 ± 0.14</td>
<td>1.42 ± 0.12</td>
</tr>
<tr>
<td>ucf</td>
<td>1.03 ± 0.10</td>
<td>0.90 ± 0.14</td>
<td>1.40 ± 0.02</td>
<td>1.03 ± 0.01</td>
</tr>
</tbody>
</table>

Although these datasets are low-dimensional with few samples and the KCEF uses the anisotropic RBF kernel (i.e., different bandwidth in each dimension, making the experiments preferable to the KCEF), the proposed DDE still outperforms the competitors on six datasets significantly, and achieves comparable performance on the rest, even though it uses a simple isotropic RBF kernel. This further demonstrates the statistical power of the proposed DDE, comparing to the score matching estimator.

### 7 Conclusion

In this paper, we exploit the doubly dual embedding to reformulate the penalized MLE to a novel saddle-point optimization, which bypasses the intractable integration and provides flexibility in parameterizing the dual distribution. The saddle point view reveals a unique understanding of GANs and leads to a practical algorithm, which achieves state-of-the-art performance. We also establish the statistical consistency and algorithm convergence guarantee for the proposed algorithm. Although the transport mapping parametrization is flexible enough, it requires extra optimization for the $KL$-divergence estimation. For the future work, we will explore dynamic-based sampling methods to design new parametrization, which shares both flexibility and density tractability.

**Acknowledgements**

We thank Michael Arbel and the anonymous reviewers for their insightful comments and suggestions. NH is supported in part by NSF-CRII-1755829, NSF-CMMI-1761699, and NCVA Faculty Fellowship.

---

**Conditional model extension.** In this part of the experiment, models are trained to estimate the conditional distribution $p(y|x)$ on the benchmark datasets for studying these methods (Arbel and Gretton, 2017; Sugiyama et al., 2010). We centered and normalized the data and randomly split the datasets into a train-
References


