Effective and principled score estimation with Nyström kernel exponential families

Dougal J. Sutherland*, Heiko Strathmann*, Michael Arbel, Arthur Gretton

Gatsby Computational Neuroscience Unit, University College London

Problem: Unnormalized density estimation

- Given samples \( \{X_n\}_{n=1}^{id} \sim p_0 \), \( X_n \in \mathbb{R}^d \)
- Want computationally efficient estimator \( p \) so that \( p(x)/Z \approx p_0(x) \)
- Don’t especially care about \( Z \): often difficult, not needed for finding modes / sampling (with MCMC) / use in approximate HMC / …
- Want to avoid strong (parametric) assumptions about \( p_0 \)

Exponential families

- Many classic densities on \( \mathbb{R}^d \) are of the form:
  \[ p(x) = \exp\left( \eta \left( T(x) \right) - A(\eta) \right) \begin{pmatrix} \text{log-determinant} \\ \text{measure} \end{pmatrix} q_\eta(x) \]
- Gaussian: \( T(x) = \langle x, x^T \rangle \); Gamma: \( T(x) = \langle x, log(x) \rangle \)
- Density is on \( T(x) \), \( s \)-dimensional “features”; can we make this richer?

Kernel exponential families [1]

- Use an RKHS \( \mathcal{H} \), with kernel \( k(x, y) = \langle k_x, k_y \rangle \).
  - Parameter \( \eta \in \mathcal{H} \), sufficient statistic: \( T(x) = k_x \) gives
    \[ p(x) = \exp(f(x) - A(f)) \]
  - Includes standard exponential family: \( k(x, y) = T(x) \cdot T(y) \)
  - But \( T \) can be infinite-dimensional, e.g. \( k(x, y) = \exp\left( -\frac{1}{2} ||x - y||^2 \right) \)
  - Class very rich: dense in anything with smooth log-density, tails like \( q_\eta \) [3]
  - But \( A(f) \) is hard to compute: maximum likelihood estimate intractable

Score matching-based estimator [3]

- Score matching approach here: minimize regularized Fisher divergence
  \[ J_\lambda(f) = \frac{1}{2} \int p_0(x) \left( ||\nabla_x \log p_0(x) - \nabla_x \log p_i(x) ||^2 \right) dx + \lambda \int \|f\|_H^2 \]
  \[ = \int p_0(x) \sum_{i=1}^n \left[ \frac{1}{2} \partial_i f(x) - \frac{1}{2} \partial_i d(f(x))^2 \right] dx + C(p_0, q_\eta) + \lambda \int \|f\|_H^2 \]
  where we used integration by parts, some mild assumptions

- Estimate integral with simple Monte Carlo
- Representer theorem: best solution \( f_{X,n} = \text{argmin}_{f \in \mathcal{H}} J_\lambda(f) \) is
  \[ f_{X,n}(x) = \sum_{a=1}^d \sum_{i=1}^n \left( \beta_{a,i} \log q_\eta(X_i) \right) \partial_a k(X_{a,i}, x) - \frac{1}{2} \partial_a^2 k(X_{a,i}, x) \]
  where \( \beta \) is the solution to an \( nd \times nd \) linear system: \( \mathcal{O}(n^3 d^3) \) time!

Nyström approximation

- Instead of minimizing \( f \) over \( \mathcal{H} \), minimize over subspace
  \[ \mathcal{H}_M = \text{span}\{q_\eta \}_{\|\eta\|^2 = 1} \subset \mathcal{H} \]
- Full solution \( f_{X,n} \) has \( y_{a,(i,2)} = \partial_a k_{X_i} \); \( M = 2nd \)

- “Nyström”: pick \( m \) points at random, \( y_{a,(i,1)} = \partial_a k_{X_i} ; M = md \)

- “Lite” [4]: pick \( m \) points at random, \( y_a = k_{X_i} ; M = m \)

Computing the Nyström approximation

- Minimizer of \( J_\lambda \) in \( \mathcal{H}_M \): \( f_{X,n}(x) = \sum_{a=1}^d \beta_a y_a \)
  \[ \beta = \left( \sum_{M \times M} B_{MM} + \lambda G_{MM} \right)^{-1} b_Y \]
  \[ (B_{XY})_{a,b} := \langle \partial_a k_{X_{a,i}}, \partial_b k_{X_{b,i}} \rangle \mathcal{H} \]

- “Nyström”: \( \mathcal{O}(nm^2d^2) \) time; “lite”: \( \mathcal{O}(nm^2d) \) time

Theory

- Assume \( p_0 = p_{f_0} \) for some \( f_0 \in \mathcal{H} \); technical assumptions on \( \mathcal{H}, f_0 \)
- \( \theta \) a parameter depending on problem smoothness: worst case \( \frac{1}{2} \), best \( \frac{1}{4} \)
- If we use “Nyström” with \( m = \Omega(n^\delta \log n) \), \( \lambda = \frac{n}{m^2} \)
- “Easy” problems: same convergence in \( J, \mathcal{H}, L_r, KL, \text{Hellinger as [3]} \)
- “Hard” problems: same \( J \) convergence, others saturate slightly sooner
- Proof uses ideas from [2] for regression, but different decomposition:
  \( f_X = \text{argmin}_{f \in \mathcal{H}} J_\lambda(f) \)

Synthetic experiments

- Target: Gaussians centered on \( d \) vertices of \( d \)-dimensional hypercube
- Evaluate Fisher divergence \( J(f) \):

Approximate Hamiltonian Monte Carlo

- HMC uses \( \nabla_x \log p(x) \), often more efficient
- Sometimes we can’t get these gradients
  - e.g. marginalizing out hyperparameter choice for a GP classifier
- Kernel Adaptive HMC [4]:
  - Start with random walk MCMC
  - Estimate \( \nabla_x \log p(x) \) from chain so far
  - Propose HMC trajectories with estimate
  - Metropolis rejection step accounts for errors in the proposed trajectories

Takeaways

- Flexible density modeling with kernel exponential families
- Nyström approximation: faster algorithm (\( n^3 \) to \( n^2 \)) with same statistical guarantees as full-data fit (\( n^2 \))
- Kernel Conditional Exponential Family: less-smooth densities
- Open questions: kernel choice, theory for “lite” basis, misspecified case

References