EFN code docs

Sean Bittner
srb2201@columbia.edu
May 26, 2018

1: Exponential Family Networks

TODO: Take from the NIPS paper and explain what EFN's are.

At some point, I will detail the exponential family models that are built into the code base. These models are manifest as tensorflow classes. There will be instructions on how to use the exponential family model class constructor for user defined exponential families. I will link to the appendix for the parameterization of these built-in classes which will serve as intuitive and demonstrative examples.

A: Built-in exponential family models

A.1 Classic exponential families

Here, we explicitly write out the exponential family parameterizations of some classical distributions in the field of statistics: the multivariate Gaussian, Dirichlet, and inverse-Wishart. Exponential family distributions with parameters θ and natural parameterization $\eta(\theta)$, have the form:

$$p(x \mid \theta) = \frac{h(x)}{A(\eta(\theta))} \exp\left(\eta(\theta)^\top T(x)\right)$$

where h is the base measure, A is the log partition function, and T is a function that maps samples to the vector of sufficient statistics of the exponential family. Alternatively, we can move the base measure and log partition function into the exponentiated term:

$$p(x \mid \theta) = \exp\left(\begin{bmatrix} \eta(\theta) \\ 1 \\ \log(A(\eta(\theta))) \end{bmatrix}^\top \begin{bmatrix} T(x) \\ \log(h(x)) \\ -1 \end{bmatrix} \right)$$

When learning EFN's, we optimize over the evidence lower bound (ELBO), requiring us to evaluate the log-joint $\log(p(x, \theta))$, while allowing us to drop the term $\log(p(\theta)) = \log(A(\eta(\theta)))$.

$$\log(p(x, \theta)) = \begin{bmatrix} \eta(\theta) \\ 1 \end{bmatrix}^\top \begin{bmatrix} T(x) \\ \log(h(x)) \end{bmatrix}$$

exponential family	θ	$\eta(\theta)$	$T(x)$	$\log(h(x))$
multivariate Gaussian	μ, Σ	$\begin{bmatrix} \Sigma^{-1}\mu \\ \frac{1}{2}\Sigma^{-1} \end{bmatrix}$	$\begin{bmatrix} x \\ xx^\top \end{bmatrix}$	$-\frac{D}{2}\log(2\pi)$
Dirichlet	α	α	$\log(x)$	$-\sum_{i=1}^{D} x_i$
inverse-Wishart	Ψ, m	$\begin{bmatrix} -\frac{1}{2}\Psi \\ \frac{-m+p+1}{2} \end{bmatrix}$	$\begin{bmatrix} X^{-1} \\ \log(\mid X \mid) \end{bmatrix}$	0

Sean Bittner
srb2201@columbia.edu

A.2 Posterior inference exponential families

When we wish to do posterior inference with a model that has likelihood that has exponential family form that is closed under sampling, we can learn the posterior inference exponential family model in general using EFNs. We can do this regardless of the form of the prior, but the built-in exponential family models have priors with exponential family form.

$$p_0(z) = \frac{1}{A_0(\alpha)} \exp\left\{\alpha^\top t_0(z)\right\} \quad , \quad p(x_i|z) = \frac{1}{A(z)} \exp\left\{\nu(z)^\top t(x_i)\right\},$$

$$\log(p(z, x_1, ..., x_N)) = \begin{bmatrix} \alpha \\ \log(A_0(\alpha)) \\ \sum_i t(x_i) \\ -N \end{bmatrix}^\top \begin{bmatrix} t_0(z) \\ -1 \\ \nu(z) \\ \log A(z) \end{bmatrix} = \begin{bmatrix} \eta(\theta) \\ 1 \end{bmatrix}^\top \begin{bmatrix} T(z) \\ \log(h(z)) \end{bmatrix}$$

where $\log(h(z)) = 0$.

As a technical note, one can drop the $\log(A_0(\alpha))$ term from the optimization cost evaluation, since it is constant with respect to z.

exponential family	α	$t_0(z)$	$\log(A_0(\alpha))$	$\nu(z)$	$t(x_i)$	$\log(A(z))$
hierarchical Dirichlet	$\alpha_0 - 1$	$\log(z)$	$\log(B(\alpha_0))$	βz	$\log(x_i)$	$\log(B(\beta z))$
Dirichlet-multinomial (N=1)	$\alpha_0 - 1$	$\log(z)$	$\log(B(\alpha_0))$	$\log(z)$	x_i	0
truncated-normal Poisson	$\begin{bmatrix} \Sigma^{-1}\mu \\ \frac{1}{2}\Sigma^{-1} \end{bmatrix}$	$\begin{bmatrix} x \\ xx^\top \end{bmatrix}$	$\frac{1}{2}(\mu^\top \Sigma^{-1} \mu + \log(\|\Sigma\|)$	$\log(z)$	x_i	z