Exponential Family Distributions

CS772A: Probabilistic Machine Learning

Piyush Rai

Plan today

- PPD for Logistic regression
- Model Selection and Model Averaging
- More on Laplace's Approximation
  - How to make LA scalable when $\boldsymbol{\theta}$ is very high dimensional Exponential family distributions
- Exponential Family Distributions

LR: Posterior Predictive Distribution

- The posterior predictive distribution can be computed as

$$p(y_* = 1|\boldsymbol{x}_*, \boldsymbol{X}, \boldsymbol{y}) = \int p(y_* = 1|\boldsymbol{w}, \boldsymbol{x}_*) p(\boldsymbol{w}|\boldsymbol{X}, \boldsymbol{y}) d\boldsymbol{w}$$

Integral not tractable and must be approximated

sigmoid

Gaussian (if using Laplace approx.)

- Monte-Carlo approximation is one possible way to approximate such integrals
  - Draw M samples $\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_M$, from the posterior $p(\boldsymbol{w}|\boldsymbol{X}, \boldsymbol{y})$
  - Now approximate the PPD as follows

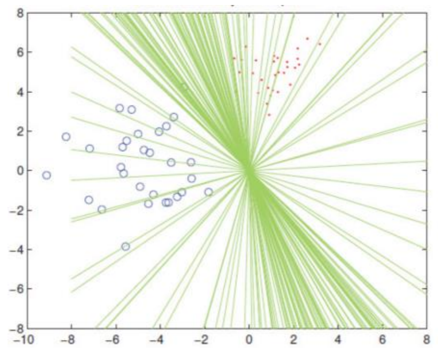
$$p(y_* = 1|\boldsymbol{x}_*, \boldsymbol{X}, \boldsymbol{y}) \approx \frac{1}{M} \sum_{m=1}^{M} p(y_* = 1|\boldsymbol{w}_m, \boldsymbol{x}_*) = \frac{1}{M} \sum_{m=1}^{M} \sigma(\boldsymbol{w}_m^\top \boldsymbol{x}_n)$$

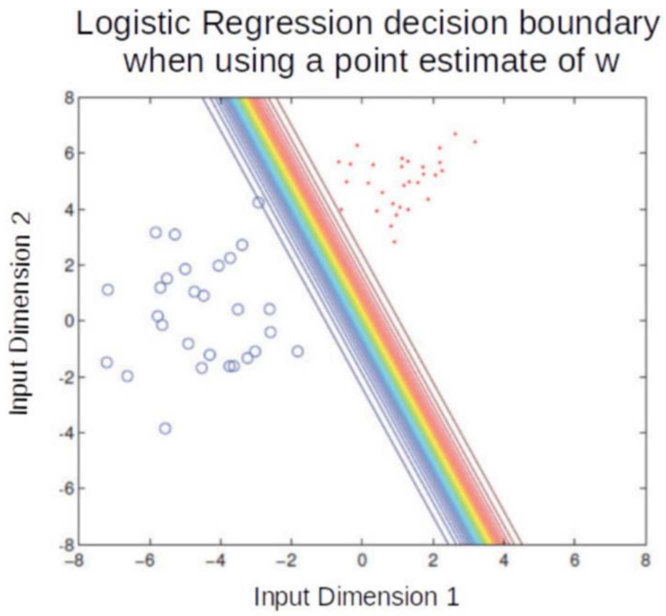
- In contrast, when using MLE/MAP solution $\hat{\boldsymbol{w}}_{opt}$, the plug-in pred. distribution

$$p(y_* = 1|\boldsymbol{x}_*, \boldsymbol{X}, \boldsymbol{y}) = \int p(y_* = 1|\boldsymbol{w}, \boldsymbol{x}_*) p(\boldsymbol{w}|\boldsymbol{X}, \boldsymbol{y}) d\boldsymbol{w}$$
$$\approx p\left(y_* = 1\middle|\hat{\boldsymbol{w}}_{opt}, \boldsymbol{x}_*\right) = \sigma(\hat{\boldsymbol{w}}_{opt}{}^\top \boldsymbol{x}_n)$$

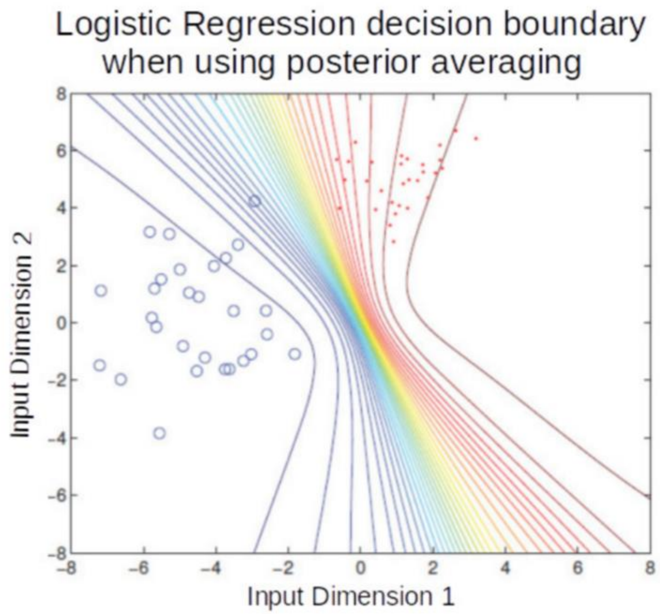
LR: Plug-in Prediction vs Bayesian Averaging

- Plug-in prediction uses a single $\boldsymbol{w}$ (point est) to make prediction
- PPD does an averaging using all possible $\boldsymbol{w}$'s from the posterior





Color transitions (red to blue) in both plots denote how the probability of an input changes from belonging to red class to belonging to blue class. All inputs on a line (or curve on RHS plot) have the same probability of belonging to the red/blue class



Posterior averaging is like using an ensemble of models. In this example, each model is a linear classifier but the ensemble-like effect resulted in nonlinear boundaries

$$p(y_* = 1|\boldsymbol{x}_*, \boldsymbol{X}, \boldsymbol{y}) \approx \sigma(\hat{\boldsymbol{w}}_{opt}{}^\top \boldsymbol{x}_n)$$

$$p(y_* = 1|\boldsymbol{x}_*, \boldsymbol{X}, \boldsymbol{y}) \approx \frac{1}{M}\sum_{m=1}^{M}\sigma(\boldsymbol{w}_m^\top \boldsymbol{x}_n)$$

More on Marginalization

- PPD is a weighted average over all possible parameter values of one model

$$p(y_*|\boldsymbol{x}_*, \mathcal{D}, m) = \int p(y_*|\boldsymbol{x}_*, \theta, m)p(\theta|\mathcal{D}, m)d\theta$$

Note: m is just a model identifier; can ignore when writing

$$\approx \frac{1}{S}\sum_{i=1}^{S} p\left(y_*|\boldsymbol{x}_*, \theta^{(i)}, m\right)$$

Each $\theta^{(i)}$ is drawn i.i.d. from the distribution $p(\theta|\mathcal{D}, m)$

Above integral replaced by a "Monte-Carlo Averaging" (an approximation when PPD integral is intractable)

- PPD marginalization can be done even over several choices of models

Marginalization over all weights of a single model m

$$p(y_*|\boldsymbol{x}_*, \mathcal{D}, m) = \int p(y_*|\boldsymbol{x}_*, \theta, m)p(\theta|\mathcal{D}, m)d\theta$$

Marginalization over all finite choices $m = 1,2, \ldots, M$ of the model

$$p(y_*|\boldsymbol{x}_*, \mathcal{D}) = \sum_{m=1}^{M} p(y_*|\boldsymbol{x}_*, \mathcal{D}, m)p(m|\mathcal{D})$$

For example, deep nets with different architectures

Like a double averaging (over all model choices, and over all weights of each model choice)

Haven't yet told you how to compute this quantity but will see shortly

Model Selection and Model Averaging

- Can use Bayes rule to find the best model from a set of models $m = 1,2,\ldots,M$

$$p(m|\mathbf{X}) = \frac{p(\mathbf{X}|m)p(m)}{p(\mathbf{X})} = \frac{p(\mathbf{X}|m)p(m)}{\sum_{m=1}^{M} p(\mathbf{X}|m)p(m)}$$

Posterior probability of model m

Marginal likelihood of model m

Prior probability of choosing model m

Marginal likelihood over all models

Will discuss later how to compute marginal likelihood

In general, intractable to compute exactly

$$p(\mathbf{X}|m) = \int p(\mathbf{X}|\theta, m)p(\theta|m)d\theta$$

Integrating out all possible parameter values under model m

$$\hat{m} = \arg\max_m p(m|\mathbf{X}) = \arg\max_m p(\mathbf{X}|m)p(m)$$

Best model

- If all models equally likely a priori then $\hat{m} = \arg\max_m p(\mathbf{X}|m)$

- For PPD, can use either the best model $\hat{m}$ or can average over all models

$$p(x_*|\mathbf{X}) \approx p(x_*|\mathbf{X}, \hat{m}) \quad \text{OR} \quad p(x_*|\mathbf{X}) = \sum_{m=1}^{M} p(x_*|\mathbf{X}, m)p(m|\mathbf{X})$$

Test data

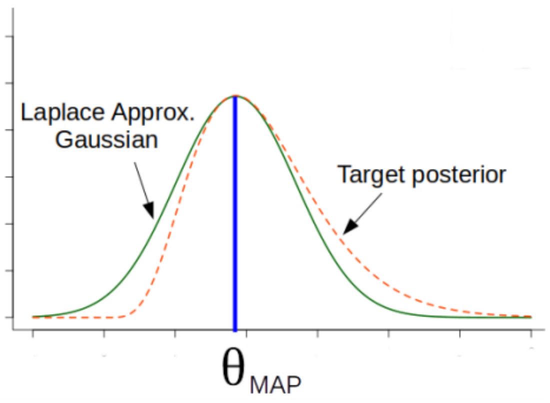
Training data

Recap: Laplace's Approximation

- Consider a posterior distribution that is intractable to compute

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D},\theta)}{p(\mathcal{D})} = \frac{p(\mathcal{D}|\theta)p(\theta)}{p(\mathcal{D})}$$

- Laplace approximation approximates the above using a Gaussian distribution



$$p(\theta|\mathcal{D}) \approx \mathcal{N}(\theta|\theta_{MAP}, \mathbf{\Lambda}^{-1})$$

Tells us about the space (curvature) of the true posterior around θ_{MAP}

Related to the Fisher Information Matrix (FIM); will see shortly

$$\theta_{MAP} = \operatorname{argmax}_\theta \log p(\theta|\mathcal{D})$$

Negative of the Hessian, i.e., the second derivative of the log joint, at θ_{MAP}

$$\mathbf{\Lambda} = -\nabla^2_\theta \log p(\theta|\mathcal{D})\Big|_{\theta=\theta_{MAP}} = -\nabla^2_\theta \log p(\mathcal{D},\theta)\Big|_{\theta=\theta_{MAP}}$$

- Laplace's approx. is based on a second-order Taylor approx. of the posterior

Detour: Hessian and Fisher Information Matrix

- Hessian is related to the Fisher Information Matrix (FIM)
- Gradient of the log likelihood is also called score function: $s(\theta) = \nabla_\theta \log p(y|\theta)$
  - Note: At some places (some generative models) $\nabla_y \log p(y|\theta)$ also called score function
- Expectation of score function is zero: $\mathbb{E}_{p(y|\theta)}[s(\theta)] = 0$ (exercise)
- Fisher Information Matrix (FIM) is covariance matrix of score function

$$\mathbf{F} = \mathbb{E}_{p(y|\theta)}[(s(\theta) - 0)(s(\theta) - 0)^\top] = \mathbb{E}_{p(y|\theta)}[\nabla_\theta \log p(y|\theta) \nabla_\theta \log p(y|\theta)^\top]$$

Note: If we have a prior $p(\theta)$ too, then also add the second derivative of $\log p(\theta)$

- $\mathbf{F} = -\mathbb{E}_{p(y|\theta)}\left[\nabla_\theta^2 \log p(y|\theta)\right]$, i.e., negative of expected Hessian (exercise)
- Each entry F_{ij} tells us how "sensitive" the model is w.r.t. the pair (θ_i, θ_j)
  - Each diagonal entry $F_{ii} = (\nabla_{\theta_i} \log p(y|\theta))^2$ tells "important" θ_i is by itself
- Can compute empirical FIM using data: $\hat{\mathbf{F}} = \frac{1}{N}\sum_{n=1}^{N}[\nabla_\theta \log p(y_n|\theta) \nabla_\theta \log p(y_n|\theta)^\top]$

Laplace Approx. for High-Dimensional Problems

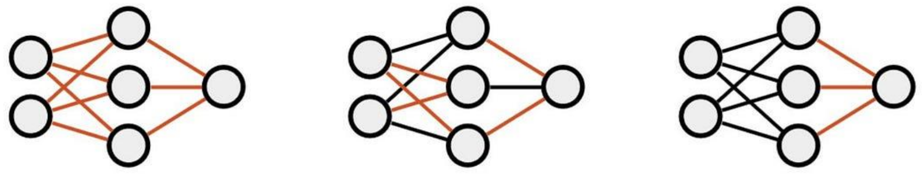
- For high-dim $\boldsymbol{\theta}$, Laplace's approx $p(\theta|\mathcal{D}) \approx \mathcal{N}(\theta|\theta_{MAP}, \boldsymbol{\Lambda}^{-1})$ can be expensive
- Many methods to address this, e.g.,
  - Use a diagonal of (empirical) Fisher as the precision

$$\boldsymbol{\Lambda} \approx \mathrm{diag}(\mathbf{F})$$

  - Use a block-diagonal approximation* of $\boldsymbol{\Lambda}$ (better than diagonal approx)

Diagonal approximation assumes that the weights are all independent whereas block-diagonal assumes that the weights within each block may have correlations

- For deep nets, use LA only for some weights + point estimates for others
  - Option 1: Use LA only for last layer weights - "last layer Laplace's approximation" (LLLA)
  - Option 2: Use LA for weights from an identified "subnetwork"



(a) All **(b)** Subnetwork **(c)** Last-Layer

  - See the "Laplace Redux" paper for more options and discussion on scalability of LA

*KFAC paper: "A Scalable Laplace Approximation for Neural Networks" (Ritter et al, ICLR 2018) *Laplace Redux -- Effortless Bayesian Deep Learning " (Daxberger et al, NeurIPS 2021)

Exp. Family (Pitman, Darmois, Koopman, 1930s)

- Defines a class of distributions. An Exponential Family distribution is of the form

$$p(\mathbf{x}|\theta) = \frac{1}{Z(\theta)} h(\mathbf{x}) \exp[\theta^\top \phi(\mathbf{x})] \quad = \quad h(\mathbf{x}) \exp[\theta^\top \phi(\mathbf{x}) - A(\theta)]$$

- $\boldsymbol{x} \in \mathcal{X}^m$ is the r.v. being modeled ($\mathcal{X}$ denotes some space, e.g., $\mathbb{R}$ or $\{0,1\}$)
- $\theta \in \mathbb{R}^d$: Natural parameters or canonical parameters defining the distribution
- $\phi(\boldsymbol{x}) \in \mathbb{R}^d$: Sufficient statistics (another random variable)
    - Knowing this quantity suffices to estimate parameter $\boldsymbol{\theta}$ from $\boldsymbol{x}$
- $Z(\theta) = \int h(\boldsymbol{x}) \exp[\theta^\top \phi(\boldsymbol{x})] d\boldsymbol{x}$: Partition Function
- $A(\theta) = \log Z(\theta)$: Log-partition function (also called cumulant function)
- $h(\boldsymbol{x})$: A constant (doesn't depend on $\boldsymbol{\theta}$)

Expressing a Distribution in Exp. Family Form

- Recall the form of exp-fam distribution $p(x|\theta) = h(x)\exp[\theta^\top \phi(x) - A(\theta)]$
- To write any exp-fam dist $p()$ in the above form, write it as $\exp(\log p())$

$$
\begin{aligned}
\exp\left(\log \text{Binomial}(x|N,\mu)\right) &= \exp\left(\log \binom{N}{x}\mu^x(1-\mu)^{N-x}\right) \\
&= \exp\left(\log \binom{N}{x} + x\log\mu + (N-x)\log(1-\mu)\right) \\
&= \binom{N}{x}\exp\left(x\log\frac{\mu}{1-\mu} - N\log(1-\mu)\right)
\end{aligned}
$$

- Now compare the resulting expression with the exponential family form

$$p(x|\theta) = h(x)\exp[\theta^\top \phi(x) - A(\theta)]$$

.. to identify the natural parameters, sufficient statistics, log-partition function, etc.

(Univariate) Gaussian as Exponential Family

- Let's try to write a univariate Gaussian in the exponential family form

$$p(\mathbf{x}|\theta) = h(\mathbf{x}) \exp[\theta^\top \phi(\mathbf{x}) - A(\theta)]$$

- Recall the PDF of a univar Gaussian (already has exp, so less work needed :))

$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right] = \frac{1}{\sqrt{2\pi}} \exp\left[\frac{\mu}{\sigma^2}x - \frac{1}{2\sigma^2}x^2 - \frac{\mu^2}{2\sigma^2} - \log\sigma\right]$$

$$= \frac{1}{\sqrt{2\pi}} \exp\left[\begin{bmatrix} \frac{\mu}{\sigma^2} \\ -\frac{1}{2\sigma^2} \end{bmatrix}^\top \begin{bmatrix} x \\ x^2 \end{bmatrix} - \left(\frac{\mu^2}{2\sigma^2} + \log\sigma\right)\right]$$

$$\theta = \begin{bmatrix} \frac{\mu}{\sigma^2} \\ -\frac{1}{2\sigma^2} \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \qquad \phi(x) = \begin{bmatrix} x \\ x^2 \end{bmatrix} \quad \text{, and} \quad \begin{bmatrix} \mu \\ \sigma^2 \end{bmatrix} = \begin{bmatrix} -\frac{\theta_1}{2\theta_2} \\ -\frac{1}{2\theta_2} \end{bmatrix}$$

$$h(x) = \frac{1}{\sqrt{2\pi}} \qquad A(\theta) = \frac{\mu^2}{2\sigma^2} + \log\sigma = \frac{-\theta_1^2}{4\theta_2} - \frac{1}{2}\log(-2\theta_2) - \frac{1}{2}\log(2\pi)$$

Other Examples

- Many other distribution belong to the exponential family
  - Bernoulli
  - Beta
  - Gamma
  - Multinoulli/Multinomial
  - Dirichlet
  - Multivariate Gaussian
  - .. and many more (https://en.wikipedia.org/wiki/Exponential_family)
- Note: Not all distributions belong to the exponential family, e.g.,
  - Uniform distribution ($x \sim \text{Unif}(a, b)$)
  - Student-t distribution
  - Mixture distributions (e.g., mixture of Gaussians)

Log-Partition Function

- The log-partition function is $A(\theta) = \log Z(\theta) = \log \int h(\boldsymbol{x}) \exp[\theta^\top \phi(\boldsymbol{x})] d\boldsymbol{x}$
- $A(\boldsymbol{\theta})$ is also called the cumulant function
- Derivatives of $A(\boldsymbol{\theta})$ can be used to generate the cumulants of the sufficient statistics
- Exercise: Assume $\boldsymbol{\theta}$ to be a scalar (thus $\boldsymbol{\phi}(x)$ is also scalar). Show that the first and the second derivatives of $A(\boldsymbol{\theta})$ are

$$\frac{dA}{d\theta} = \mathbb{E}_{p(\boldsymbol{x}|\theta)}[\phi(\boldsymbol{x})]$$

$$\frac{d^2A}{d\theta^2} = \mathbb{E}_{p(\boldsymbol{x}|\theta)}[\phi^2(\boldsymbol{x})] - \left[\mathbb{E}_{p(\boldsymbol{x}|\theta)}[\phi(\boldsymbol{x})]\right]^2 = \mathrm{var}[\phi(\boldsymbol{x})]$$

- Above result also holds when $\boldsymbol{\theta}$ and $\boldsymbol{\phi}(x)$ are vector-valued (the "var" will be "covar")
- Important: $A(\boldsymbol{\theta})$ is a convex function of $\boldsymbol{\theta}$. Why?

MLE for Exponential Family Distributions

- Assume data $\mathcal{D} = \{x_1, \ldots, x_N\}$ drawn i.i.d. from an exp. family distribution

$$p(x|\theta) = h(x)\exp[\theta^\top \phi(x) - A(\theta)]$$

- To do MLE, we need the overall likelihood -- a product of the individual likelihoods

$$p(\mathcal{D}|\theta) = \prod_{i=1}^{N} p(\mathbf{x}_i|\theta) = \left[\prod_{i=1}^{N} h(\mathbf{x}_i)\right] \exp\left[\theta^\top \sum_{i=1}^{N} \phi(\mathbf{x}_i) - NA(\theta)\right] = \left[\prod_{i=1}^{N} h(\mathbf{x}_i)\right] \exp\left[\theta^\top \phi(\mathcal{D}) - NA(\theta)\right]$$

- To estimate θ (as we'll see shortly), we only need $\phi(\mathcal{D}) = \sum_{i=1}^{N} \phi(\mathbf{x}_i)$ and N
- Size of $\phi(\mathcal{D}) = \sum_{i=1}^{N} \phi(x_i)$ does not grow with N (same as the size of each $\phi(x_i)$)
- Only exponential family distributions have finite-sized sufficient statistics
  - No need to store all the data; can simply update the sufficient statistics as data comes
  - Useful in probabilistic inference with large-scale data sets and "online" parameter estimation

Bayesian Inference for Expon. Family Distributions

- Already saw that the total likelihood given N i.i.d. observations $\mathcal{D} = \{x_1, \ldots, x_N\}$

$$p(\mathcal{D}|\theta) \propto \exp\left[\theta^\top \phi(\mathcal{D}) - NA(\theta)\right] \quad \text{where} \quad \phi(\mathcal{D}) = \sum_{i=1}^{N} \phi(\boldsymbol{x}_i)$$

- Let's choose the following prior (note: looks similar in terms of $\boldsymbol{\theta}$ within exp)

$$p(\theta|\nu_0, \boldsymbol{\tau}_0) = h(\theta) \exp\left[\theta^\top \tau_0 - \nu_0 A(\theta) - A_c(\nu_0, \boldsymbol{\tau}_0)\right]$$

- Ignoring the prior's log-partition function $A_c(\nu_0, \boldsymbol{\tau}_0) = \log \int_\theta h(\theta) \exp\left[\theta^\top \tau_0 - \nu_0 A(\theta)\right] d\theta$

$$p(\theta|\nu_0, \boldsymbol{\tau}_0) \propto h(\theta) \exp\left[\theta^\top \tau_0 - \nu_0 A(\theta)\right]$$

- Comparing the prior's form with the likelihood, note that
  - ν_0 is like the number of "pseudo-observations" coming from the prior
  - τ_0 is the total sufficient statistics of the pseudo-observations (τ_0 / ν_0 per pseudo-obs)

The Posterior

- The likelihood and prior were

$$p(\mathcal{D}|\theta) \propto \exp\left[\theta^\top \phi(\mathcal{D}) - NA(\theta)\right] \quad \text{where} \quad \phi(\mathcal{D}) = \sum_{i=1}^{N} \phi(\boldsymbol{x}_i)$$

$$p(\theta|\nu_0, \boldsymbol{\tau}_0) \propto h(\theta) \exp\left[\theta^\top \tau_0 - \nu_0 A(\theta)\right]$$

Assume its log partition function denoted as $A_c(\nu_0, \tau_0)$

- The posterior $p(\theta|\mathcal{D}) \propto p(\theta)p(\mathcal{D}|\theta)$ therefore will be

$$p(\theta|\mathcal{D}) \propto h(\theta) \exp\left[\theta^\top (\tau_0 + \phi(\mathcal{D})) - (\nu_0 + N)A(\theta)\right]$$

Its log partition function will be $A_c(\nu_0 + N, \tau_0 + \phi(\mathcal{D}))$

Posterior is also from the same family as the prior

Happens when the prior is conjugate to the likelihood

- Every exp family likelihood has a conjugate prior having the form above
- Posterior's hyperparams τ_0', ν_0' obtained by adding "stuff" to prior's hyperparams

$$\nu_0' \leftarrow \nu_0 + N$$

$$\tau_0' \leftarrow \tau_0 + \phi(\mathcal{D})$$

Number of pseudo-observations plus number of actual observations

Suff-stats of pseudo-obervations plus suff-stats of actual observations

Another equivalent form

$$p(\theta|\mathcal{D}) \propto h(\theta) \exp\left[\theta^\top (\nu_0 + N) \frac{\nu_0 \bar{\tau}_0 + \phi(\mathcal{D})}{\nu_0 + N} - (\nu_0 + N)A(\theta)\right]$$

$\bar{\tau}_0 = \tau_0 / \nu_0$

$$\nu_0' \leftarrow \nu_0 + N$$

$$\bar{\tau}_0' \leftarrow \frac{\nu_0 \bar{\tau}_0 + N\bar{\phi}}{\nu_0 + N}$$

$\bar{\phi} = \frac{\phi(D)}{N}$

Convex comb of avg suff-stats of pseudo obs and actual obs

Posterior Predictive Distribution

- Assume some training data $\mathcal{D} = \{x_1, \ldots, x_N\}$ from some exp-fam distribution
- Assume some test data $\mathcal{D}' = \{\tilde{x}_1, \ldots, \tilde{x}_{N'}\}$ from the same distribution
- The posterior pred. distr. of $\mathcal{D}'$

Exp. Fam. likelihood w.r.t. test data

Posterior (same form as the prior due to conjugacy)

$$p(\mathcal{D}'|\mathcal{D}) = \int p(\mathcal{D}'|\theta)p(\theta|\mathcal{D})d\theta$$

$$= \int \underbrace{\left[\prod_{i=1}^{N'} h(\tilde{\mathbf{x}}_i)\right]}_{\text{constant w.r.t. } \theta} \exp\left[\theta^\top \phi(\mathcal{D}') - N' A(\theta)\right] h(\theta) \exp\left[\theta^\top(\tau_0 + \phi(\mathcal{D})) - (\nu_0 + N)A(\theta) - \underbrace{A_c(\nu_0 + N, \boldsymbol{\tau}_0 + \phi(\mathcal{D}))}_{\text{constant w.r.t. } \theta}\right] d\theta$$

- This gets further simplified into

$$p(\mathcal{D}'|\mathcal{D}) = \left[\prod_{i=1}^{N'} h(\tilde{\mathbf{x}}_i)\right] \frac{\int h(\theta) \exp\left[\theta^\top(\tau_0 + \phi(\mathcal{D}) + \phi(\mathcal{D}')) - (\nu_0 + N + N')A(\theta)\right] d\theta}{\exp\left[A_c(\nu_0 + N, \boldsymbol{\tau}_0 + \phi(\mathcal{D}))\right]}$$

$$= \left[\prod_{i=1}^{N'} h(\tilde{\mathbf{x}}_i)\right] \frac{Z_c(\nu_0 + N + N', \boldsymbol{\tau}_0 + \phi(\mathcal{D}) + \phi(\mathcal{D}'))}{\exp\left[A_c(\nu_0 + N, \boldsymbol{\tau}_0 + \phi(\mathcal{D}))\right]}$$

Posterior Predictive Distribution

> Thus PPD as well as marginal likelihood has closed form expression when working with exp-family distributions

- Since $A_c = \log Z_c$ or $Z_c = \exp(A_c)$, we can write the PPD as

$$p(\mathcal{D}'|\mathcal{D}) = \left[\prod_{i=1}^{N'} h(\tilde{\mathbf{x}}_i)\right] \frac{Z_c(\nu_0 + N + N', \boldsymbol{\tau}_0 + \phi(\mathcal{D}) + \phi(\mathcal{D}'))}{Z_c(\nu_0 + N, \boldsymbol{\tau}_0 + \phi(\mathcal{D}))}$$

$$= \left[\prod_{i=1}^{N'} h(\tilde{\mathbf{x}}_i)\right] \exp\left[A_c(\nu_0 + N + N', \boldsymbol{\tau}_0 + \phi(\mathcal{D}) + \phi(\mathcal{D}')) - A_c(\nu_0 + N, \boldsymbol{\tau}_0 + \phi(\mathcal{D}))\right]$$

- Therefore the posterior predictive is proportional to
  - Ratio of two partition functions of two "posterior distributions" (one with $N + N'$ examples and the other with N examples)
  - Exponential of the difference of the corresponding log-partition functions
- Note that the form of Z_c (and A_c) will simply depend on the chosen conjugate prior
- Very useful result. Also holds for $N = 0$
  - In this case $p(\mathcal{D}') = \int p(\mathcal{D}'|\theta)p(\theta)d\theta$ is simply the marginal likelihood of test data $\mathcal{D}'$

Summary

- Exp. family distributions are very useful for modeling diverse types of data/parameters
- Conjugate priors to exp. family distributions make parameter updates very simple
- Other quantities such as posterior predictive can be computed in closed form
- Useful in designing generative classification models. Choosing class-conditional from exponential family with conjugate priors helps in parameter estimation
- Useful in designing generative models for unsupervised learning
- Used in designing Generalized Linear Models: Model $p(y|x)$ using exp. fam distribution
  - Linear regression (with Gaussian likelihood) and logistic regression are GLMs
- Will see several use cases when we discuss approx inference algorithms (e.g., Gibbs sampling, and especially variational inference)