Probabilistic Machine Learning
Lecture 15
Exponential Families

Philipp Hennig
15 June 2020
<table>
<thead>
<tr>
<th>#</th>
<th>date</th>
<th>content</th>
<th>Ex</th>
<th>#</th>
<th>date</th>
<th>content</th>
<th>Ex</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20.04</td>
<td>Introduction</td>
<td>1</td>
<td>14</td>
<td>09.06</td>
<td>Generalized Linear Models</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>21.04</td>
<td>Reasoning under Uncertainty</td>
<td>2</td>
<td>15</td>
<td>15.06</td>
<td>Exponential Families</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>27.04</td>
<td>Continuous Variables</td>
<td>3</td>
<td>16</td>
<td>16.06</td>
<td>Graphical Models</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>28.04</td>
<td>Monte Carlo</td>
<td>4</td>
<td>17</td>
<td>22.06</td>
<td>Factor Graphs</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>04.05</td>
<td>Markov Chain Monte Carlo</td>
<td>5</td>
<td>18</td>
<td>23.06</td>
<td>The Sum-Product Algorithm</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>05.05</td>
<td>Gaussian Distributions</td>
<td>6</td>
<td>19</td>
<td>29.06</td>
<td>Example: Topic Models</td>
<td>10</td>
</tr>
<tr>
<td>7</td>
<td>11.05</td>
<td>Parametric Regression</td>
<td>7</td>
<td>20</td>
<td>30.06</td>
<td>Mixture Models</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>12.05</td>
<td>Learning Representations</td>
<td>8</td>
<td>21</td>
<td>06.07</td>
<td>EM</td>
<td>11</td>
</tr>
<tr>
<td>9</td>
<td>18.05</td>
<td>Gaussian Processes</td>
<td>9</td>
<td>22</td>
<td>07.07</td>
<td>Variational Inference</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>19.05</td>
<td>Understanding Kernels</td>
<td>10</td>
<td>23</td>
<td>13.07</td>
<td>Topics</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>26.05</td>
<td>Gauss-Markov Models</td>
<td>11</td>
<td>25</td>
<td>20.07</td>
<td>Example: Kernel Topic Models</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>25.05</td>
<td>An Example for GP Regression</td>
<td>12</td>
<td>24</td>
<td>14.07</td>
<td>Example: Inferring vision</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>08.06</td>
<td>GP Classification</td>
<td>13</td>
<td>26</td>
<td>21.07</td>
<td>Revision</td>
<td></td>
</tr>
</tbody>
</table>
Why is this hard?

The computational challenge in Bayesian Inference

\[ p(x \mid y) = \frac{p(y \mid x)p(x)}{\int p(y \mid x)p(x) \, dx} \]

- the integral \( \int p(y \mid x)p(x) \, dx \) may be intractable
- thus, also expectations \( \int f(x)p(x \mid y) \, dx \) are hard

Practical probabilistic inference is chiefly a *computational* task.
The Toolbox

Framework:

\[ \int p(x_1, x_2) \, dx_2 = p(x_1) \quad p(x_1, x_2) = p(x_1 | x_2) p(x_2) \quad p(x | y) = \frac{p(y | x) p(x)}{p(y)} \]

Modelling:

- graphical models (conditional independence)
- Gaussian distributions
- Kernels
- Markov Chains
- Exponential Families / Conjugate Priors

Computation:

- Monte Carlo
- Linear algebra / Gaussian inference
- maximum likelihood / MAP
- Laplace approximations
Hierarchical Bayesian Inference

Catch-up from previous lectures

Recall from GP regression: How to set parameters $\theta$? From marginal likelihood $p(Y \mid \theta)$:

$$
\hat{\theta} = \arg \max_{\theta} \mathcal{N}(y; \phi^T_X \mu + b, \phi^T_X \Sigma \phi_X + \Lambda)
$$

$$
= \arg \max_{\theta} \log \mathcal{N}(y; \phi^T_X \mu + b, \phi^T_X \Sigma \phi_X + \Lambda)
$$

$$
= \arg \min_{\theta} - \log \mathcal{N}(y; \phi^T_X \mu + b, \phi^T_X \Sigma \phi_X + \Lambda)
$$

$$
= \arg \min_{\theta} \frac{1}{2} \left( (y - \phi^T_X \mu)^T \left( \phi^T_X \Sigma \phi_X + \Lambda \right)^{-1} (y - \phi^T_X \mu) + \log \left| \phi^T_X \Sigma \phi_X + \Lambda \right| \right) + \frac{N}{2} \log 2\pi
$$

In general, hierarchical inference is not analytically tractable. However, there are special cases...
Analytic Hierarchical Bayesian Inference

Inferring the Mean of a Gaussian

\[
p(x \mid \mu) = \prod_{i=1}^{n} \mathcal{N}(x_i; \mu, \Sigma) \quad \text{and} \quad p(\mu \mid \mu_0, \Sigma_0) = \mathcal{N}(\mu; \mu_0, \Sigma_0)
\]

\[
p(\mu \mid x) = \frac{p(x \mid \mu)p(\mu \mid \mu_0, \Sigma_0)}{p(x)} = \mathcal{N}\left(\mu; (\Sigma_0^{-1} + n\Sigma^{-1})^{-1} (\Sigma_0^{-1} \mu_0 + \Sigma^{-1} \sum_i x_i), (\Sigma_0^{-1} + n\Sigma^{-1})^{-1}\right)
\]
Analytic Hierarchical Bayesian Inference
Inferring a Binary Distribution

\[ p(x \mid f) = \prod_{i=1}^{n} f^{x_i} \cdot (1 - f)^{1-x_i} \quad x \in \{0; 1\} \]
\[ = f^{n_1} \cdot (1 - f)^{n_0} \quad n_0 := n - n_1 \]

\[ p(f \mid \alpha, \beta) = \mathcal{B}(\alpha, \beta) = \frac{1}{B(\alpha, \beta)} f^{\alpha-1} (1 - f)^{\beta-1} \]

\[ p(f \mid x) = \mathcal{B}(\alpha + n_1, \beta + n_0) = \frac{1}{B(\alpha + n_1, \beta + n_0)} f^{\alpha+n_1-1} (1 - f)^{\beta+n_0-1} \]

Pierre Simon, marquis de Laplace, 1749–1827
Analytic Hierarchical Bayesian Inference

Inferring a Categorical Distribution

\[ p(x) = \prod_{i=1}^{n} f_{x_i} \quad x \in \{0; \ldots , K\} \]

\[ = \prod_{k=1}^{K} f_{n_k}^{n_k} \quad n_k := |\{x_i \mid x_i = k\}| \]

\[ p(f \mid \alpha) = D(\alpha) = \frac{1}{B(\alpha)} \prod_{k=1}^{K} f_{\alpha_k - 1}^{\alpha_k - 1} \]

\[ p(f \mid x) = D(\alpha + n) \]

Peter Gustav Lejeune Dirichlet (1805–1859)
Analytic Hierarchical Bayesian Inference
Inferring the (Co-) Variance of a Gaussian

\[ p(x \mid \sigma) = \prod_{i=1}^{n} \mathcal{N}(x_i; \mu, \sigma^2) \]

\[ p(\sigma) = ? \]
Analytic Hierarchical Bayesian Inference

Inferring the (Co-) Variance of a Gaussian

\[
p(x | \sigma) = \prod_{i=1}^{n} \mathcal{N}(x_i; \mu, \sigma^2)
\]

\[
p(\sigma) = ?
\]

\[
\log p(x | \sigma) = -\frac{1}{2} \log \sigma^2 - \frac{1}{2} (x - \mu)^2 \cdot \frac{1}{\sigma^2} - \frac{1}{2} \log 2\pi
\]
Analytic Hierarchical Bayesian Inference

Inferring the (Co-) Variance of a Gaussian

\[
p(x \mid \sigma) = \prod_{i=1}^{n} \mathcal{N}(x_i; \mu, \sigma^2)
\]

\[
p(\sigma) = \ ?
\]

\[
\log p(x \mid \sigma) = -\frac{1}{2} \log \sigma^2 - \frac{1}{2} (x - \mu)^2 \cdot \frac{1}{\sigma^2} - \frac{1}{2} \log 2\pi
\]

\[
\log p(\sigma \mid \alpha, \beta) = (\alpha + 1) \log \sigma^{-2} - \beta \cdot \frac{1}{\sigma^2} - Z(\alpha, \beta)
\]

\[
p(\sigma \mid \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} (\sigma^{-2})^{\alpha+1} e^{-\beta \sigma^{-2}} := \mathcal{G}(\sigma^{-2}; \alpha, \beta)
\]

\[
p(\sigma \mid \alpha, \beta, x) = \mathcal{G} \left( \sigma^{-2}; \alpha + \frac{n}{2}, \beta + \frac{1}{2} \sum_i (x_i - \mu)^2 \right)
\]

Daniel Bernoulli (1700–1782)
Analytic Hierarchical Bayesian Inference

Inferring Mean and Co-Variance of a Gaussian

\[ p(x \mid \mu, \sigma) = \prod_{i=1}^{n} \mathcal{N}(x_i; \mu, \sigma^2) \]

\[ p(\mu, \sigma \mid \mu_0, \nu, \alpha, \beta) = \mathcal{N} \left( \mu; \mu_0, \frac{\sigma^2}{\nu} \right) G(\sigma^{-2}; \alpha, \beta) \]

\[ p(\mu, \sigma \mid x, \mu_0, \nu, \alpha, \beta) = \mathcal{N} \left( \mu; \frac{\nu \mu_0 + n\bar{x}}{\nu + n}, \frac{\sigma^2}{\nu + n} \right). \]

\[ G \left( \sigma^{-2}; \alpha + \frac{n}{2}, \beta + \frac{1}{2} \sum_{i=1}^{n} (x_i - \bar{x})^2 + \frac{n\nu}{2(n + \nu)}(\bar{x} - \mu_0)^2 \right) \]

where \[ \bar{x} := \frac{1}{n} \sum_{i=1}^{n} x_i \]
Analytic Hierarchical Bayesian Inference

Inferring Mean and Co-Variance of a Gaussian

\[
p(x \mid \mu, \sigma) = \prod_{i=1}^{n} \mathcal{N}(x_i; \mu, \sigma^2)
\]

\[
p(\mu, \sigma \mid \mu_0, \nu, \alpha, \beta) = \mathcal{N} \left( \mu; \frac{\sigma^2}{\nu} \right) \mathcal{G} \left( \sigma^{-2}; \alpha, \beta \right)
\]

\[
p(\mu, \sigma \mid x, \mu_0, \nu, \alpha, \beta) = \mathcal{N} \left( \mu; \frac{\nu \mu_0 + n\bar{x}}{\nu + n}, \frac{\sigma^2}{\nu + n} \right).
\]

\[
\mathcal{G} \left( \sigma^{-2}; \alpha + \frac{n}{2}, \beta + \frac{1}{2} \sum_{i=1}^{n} (x_i - \bar{x})^2 + \frac{n\nu}{2(n + \nu)} (\bar{x} - \mu_0)^2 \right)
\]

where \( \bar{x} := \frac{1}{n} \sum_{i=1}^{n} x_i \)
**Definition (Conjugate Prior)**

Let $D$ and $x$ be a data-set and a variable to be inferred, respectively, connected by the likelihood $p(D \mid x) = \ell(D; x)$. A **conjugate prior to** $\ell$ **for** $x$ is a probability measure with pdf $p(x) = \pi(x; \theta)$ of functional form $\pi$, such that

$$p(x \mid D) = \frac{\ell(D; x)\pi(x; \theta)}{\int \ell(D; x)\pi(x; \theta) \, dx} = \pi(x; \theta').$$

That is, such that the posterior arising from $\ell$ is of the same functional form as the prior, with updated parameters.

---


- **Conjugate priors** allow analytic Bayesian inference
- How can we construct them in general?
Definition (Exponential Family, simplified form)

Consider a random variable $X$ taking values $x \in \mathbb{X} \subset \mathbb{R}^n$. A probability distribution for $X$ with pdf of the functional form

$$p_w(x) = h(x) \exp \left[ \phi(x)^\top w - \log Z(w) \right] = \frac{h(x)}{Z(w)} e^{\phi(x)^\top w} = p(x \mid w)$$

is called an *exponential family* of probability measures. The function $\phi : \mathbb{X} \rightarrow \mathbb{R}^d$ is called the *sufficient statistics*. The parameters $w \in \mathbb{R}^d$ are the *natural parameters* of $p_w$. The normalization constant $Z(w) : \mathbb{R}^d \rightarrow \mathbb{R}$ is the *partition function*. The function $h(x) : \mathbb{X} \rightarrow \mathbb{R}_+$ is the *base measure*. 
The Bernoulli Distribution

a quick tour of exponential families

\[ p(k \mid q) = \binom{n}{k} q^k \cdot (1 - q)^{n-k} \quad \text{(nb: treating } n \text{ as fixed)} \]

\[ = \binom{n}{k} \exp(k \log q + (n - k) \log(1 - q)) \]

\[ = \binom{n}{k} \exp \left( \frac{k}{\phi(k)} \frac{q}{1 - q} + n \log(1 - q) \right) \]

\[ \log Z(w) = n \log(1 + e^w) \]
The Beta Distribution

a quick tour of exponential families

\[ p(q \mid \alpha, \beta) = \frac{1}{B(\alpha, \beta)} q^{\alpha-1} (1 - q)^{\beta-1} \]

\[ = \exp \left( \frac{\log q}{\log(1 - q)} \right)^T \begin{bmatrix} \alpha - 1 \\ \beta - 1 \end{bmatrix} - \log B(\alpha, \beta) \]

\[ = \frac{1}{q(1 - q)} \exp \left( \frac{\log q}{\log(1 - q)} \right)^T \begin{bmatrix} \alpha \\ \beta \end{bmatrix} - \log B(\alpha, \beta) \]

sufficient statistics \( \phi \), natural parameters \( w \) and base measure \( h \) are not uniquely defined.
### A Family Meeting

**incomplete list of exponential families**

<table>
<thead>
<tr>
<th>Name</th>
<th>sufficient stats</th>
<th>domain</th>
<th>use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bernoulli</td>
<td>$\phi(x) = [x]$</td>
<td>$X = {0; 1}$</td>
<td>coin toss</td>
</tr>
<tr>
<td>Poisson</td>
<td>$\phi(x) = [x]$</td>
<td>$X = \mathbb{R}_+$</td>
<td>emails per day</td>
</tr>
<tr>
<td>Laplace</td>
<td>$\phi(x) = [1, x]^\top$</td>
<td>$X = \mathbb{R}$</td>
<td>floods</td>
</tr>
<tr>
<td>Helmert ($\chi^2$)</td>
<td>$\phi(x) = [x, -\log x]$</td>
<td>$X = \mathbb{R}$</td>
<td>variances</td>
</tr>
<tr>
<td>Dirichlet</td>
<td>$\phi(x) = [\log x]$</td>
<td>$X = \mathbb{R}_+$</td>
<td>class probabilities</td>
</tr>
<tr>
<td>Euler ($\Gamma$)</td>
<td>$\phi(x) = [x, \log x]$</td>
<td>$X = \mathbb{R}_+$</td>
<td>variances</td>
</tr>
<tr>
<td>Wishart</td>
<td>$\phi(X) = [X, \log</td>
<td>X</td>
<td>]$</td>
</tr>
<tr>
<td>Gauss</td>
<td>$\phi(X) = [X, XX^\top]$</td>
<td>$X = \mathbb{R}^N$</td>
<td>functions</td>
</tr>
<tr>
<td>Boltzmann</td>
<td>$\phi(X) = [X, \text{triag}(XX^\top)]$</td>
<td>$X = {0; 1}^N$</td>
<td>thermodynamics</td>
</tr>
</tbody>
</table>
Consider the exponential family \( p_w(x \mid w) = h(x) \exp \left[ \phi(x)^T w - \log Z(w) \right] \)

its conjugate prior is the exponential family

\[
F(\alpha, \nu) = \int \exp(\alpha^T w - \nu \log Z(w)) \, dw
\]

because \( p_\alpha(w \mid \alpha, \nu) \prod_{i=1}^{n} p_w(x_i \mid w) \propto p_\alpha \left( w \mid \alpha + \sum_i \phi(x_i), \nu + n \right) \)

and the predictive is

\[
p(x) = \int p_w(x \mid w) p_\alpha(w \mid \alpha, \nu) \, dw = h(x) \int e^{(\phi(x) + \alpha)^T w + (\nu + 1) \log Z(w) - \log F(\alpha, \nu)} \, dw
\]

\[
= h(x) \frac{F(\phi(x) + \alpha, \nu + 1)}{F(\alpha, \nu)}
\]

Computing \( F(\alpha, \nu) \) can be tricky. In general, this is the challenge when constructing an EF.
Consider the exponential family

\[ p_w(x \mid w) = \exp \left[ \phi(x)^\top w - \log Z(w) \right] \]

for iid data:

\[ p_w(x_1, x_2, \ldots, x_n \mid w) = \prod_{i} p_w(x_i \mid w) = \exp \left( \sum_{i} \phi(x_i)^\top w - n \log Z(w) \right) \]

to find the maximum likelihood estimate for \( w \), set

\[ \nabla_w \log p(x \mid w) = 0 \quad \Rightarrow \quad \nabla_w \log Z(w) = \frac{1}{n} \sum_i \phi(x_i) \]

hence, collect statistics of \( \phi \), compute \( \nabla_w \log Z(w) \) and solve the above for \( w \).
Re-phrased from above: because \( \int_X dp_w(x) = 1 \), we have

\[
\nabla_w \int p_w(x \mid w) \, dx = \int \nabla_w p_w(x \mid w) \, dx = \int \phi(x) \, dp_w(x \mid w) - \nabla_w \log Z(w) \int dp_w(x \mid w) = \nabla_w 1 = 0
\]

\[\Rightarrow \mathbb{E}_{p_w}(\phi(x)) = \nabla_w \log Z(w)\]

hence, if we should need to compute \( \mathbb{E}_{p_w}(\phi(x)) \), we can do so by differentiating \( \log Z \) wrt. \( w \) instead of integrating \( p \) over \( x \). (actually, we're efficiently re-using someone else's integral)

Note that an exponential family forms a *Abelian semigroup* on \( w \):

\[
p_w(x \mid w_1) \cdot p_w(x \mid w_2) \propto p(x \mid w_1 + w_2)
\]

Thus, combining information about \( x \) from independent \( p_w \)-sources can be done by floating point addition. In this sense, exponential families map inference to addition.
Exponential Families

- have conjugate priors
- allow maximum likelihood inference on their parameters from \( N \) observations in \( \mathcal{O}(N) \), because doing so requires only the sufficient statistics \( \phi \).
- allow computation of the integrals \( \mathbb{E}_{p_w}(\phi(x)) = \nabla_w \log Z(w) \)

All of this hinges on the fact that \( \log Z(w) \) is (analytically) known.

Can we use exponential families \( p_w(x) = e^{\phi(x)^T w} / Z(w) \) to learn distributions, just like we used linear forms \( f(x) = \phi(x)^T w \) to learn functions?

Yes! In fact, we can even do Bayesian distribution regression. It is called conjugate prior inference.
Recap: Regression on Functions

The $\ell_2$ loss

- Recall previous lectures: **regression** on **real functions**:
  Given $(y_i, x_i)_{i=1,...,n}$, and assume $p(y_i \mid f(x_i)) = \mathcal{N}(y_i; f(x_i), \sigma^2)$ and $f(x) = \phi(x)^T w$. Notice conjugate Gaussian prior $p(w) = \mathcal{N}(w; \mu, \Sigma)$, get Gaussian posterior $p(w \mid y) = \mathcal{N}(\ldots)$
  - statistical analysis: interpret negative log posterior as empirical risk $L_2(w) \propto -\log p(w \mid y)$.

  MAP estimate at
  $\hat{f}(x) = \arg \min_{w \in \mathbb{R}^d} \sum_{i=1}^n \|y_i - \phi(x_i)^T w\|^2 + \frac{\sigma^2}{n} \|w\|_\Sigma^2 =: \arg \min_{w \in \mathbb{R}^d} L_2(w)$

- assume $x_i \sim p(x)$, then the Loss approximates an expected log posterior
  $\hat{f} \approx \arg \min_{w \in \mathbb{R}^d} \int \|f(x) - \phi(x)^T w\|^2 dp(x) + \frac{\sigma^2}{n} \|w\|_\Sigma^2$

- thus, for $n \to \infty$, find function $\hat{f}$ that minimizes the **expected square** risk to $f$ in $\mathcal{H}_\phi = \{f : \mathbb{X} \to \mathbb{R} \mid f(x) = \phi(x)^T w\}$. 
Interlude: KL divergence

The most mis-spelled names in statistics

Definition (Kullback-Leibler divergence)

Let $P$ and $Q$ be probability distributions over $\mathbb{X}$ with pdf’s $p(x)$ and $q(x)$, respectively. The **KL-divergence from $Q$ to $P$** is defined as

$$D_{\text{KL}}(P \parallel Q) := \int \log \left( \frac{p(x)}{q(x)} \right) \, dp(x)$$

(I will often write $D_{\text{KL}}(p \parallel q)$ instead)

Some properties:

- $D_{\text{KL}}(P \parallel Q) \neq D_{\text{KL}}(Q \parallel P)$
- $D_{\text{KL}}(P \parallel Q) \geq 0$, $\forall P, Q$ (**Gibbs’ inequality**), and
- $D_{\text{KL}}(P \parallel Q) = 0 \iff p \equiv q$ almost everywhere
Given $[x_i]_{i=1,...,n}$ with $x_i \sim p(x)$, assume

$$p(x) \approx \hat{p}(x \mid w) = \exp(\phi(x)^T w - \log Z(w))$$

to find $\hat{w}$, consider

$$\hat{w} = \arg\min_{w \in \mathbb{R}^d} D_{KL}(p(x) \| \hat{p}(x \mid w)) = \arg\min_{w \in \mathbb{R}^d} \int [\log p(x) - \log \hat{p}(x \mid w)] dp(x)$$

$$= \arg\min_{w \in \mathbb{R}^d} \int \log p(x) dp(x) + \mathbb{E}_p(\phi(x))^T w - \log Z(w) = \arg\min_{w \in \mathbb{R}^d} \mathcal{L}_{\log}(w)$$

Find minimum at $\nabla_w \mathcal{L}_{\log}(w) = 0$, where

$$\mathbb{E}_p(\phi(x)) \approx \frac{1}{n} \sum_{i=1}^{n} \phi(x_i) = \nabla_w \log Z(w) = \mathbb{E}_{\hat{p}}(\phi(x))$$
MAP Regression on Distributions!

Fitting distributions with exponential families

Given \([x_i]_{i=1,\ldots,n}\) with \(x_i \sim p(x)\), assume

\[ p(x) \approx \hat{p}(x \mid w) = \exp(\phi(x)^{\top}w - \log Z(w)) \]

to find \(\hat{w}\), consider (to regularize, include the conjugate prior. No need to know its normalizer!)

\[
\hat{w} = \arg \min_{w \in \mathbb{R}^d} D_{KL}(p(x)\|\hat{p}(x, w)) = \arg \min_{w \in \mathbb{R}^d} \int [\log p(x) - \log \hat{p}(x \mid w)] dp(x) + \alpha^{\top}w - \nu \log Z(w)
\]

\[
= \arg \min_{w \in \mathbb{R}^d} \int \log p(x) dp(x) + \mathbb{E}_p(\phi(x))^{\top}w - \log Z(w) + \alpha^{\top}w - \nu \log Z(w) = \arg \min_{w \in \mathbb{R}^d} \tilde{L}_{\log}(w)
\]

Find minimum at \(\nabla_w \tilde{L}_{\log}(w) = 0\), where

\[
\mathbb{E}_p(\phi(x)) \approx \frac{1}{n} \sum_{i=1}^{n} \phi(x_i) = \frac{n}{n + \nu} \nabla_w \log Z(w) - \frac{1}{n} \alpha
\]
Given \( \{x_i\}_{i=1}^{n} \) with \( x_i \sim p(x) \), assume

\[
p(x) \approx p_w(x \mid w) = \exp(\phi(x)^T w - \log Z(w)) \quad \text{and} \quad p_F(w \mid \alpha, \nu) = \exp(w^T \alpha - \nu \log Z(w) - \log F(\alpha, \nu))
\]

compute the posterior on \( w \), using the conjugate prior

\[
p(w \mid x, \alpha, \nu) = \frac{\prod_{i=1}^{n} p_w(x_i \mid w)p_F(w \mid \alpha, \nu)}{\int p(x \mid w)p(w \mid \alpha, \nu) \, dx} = p_F \left( w \mid \alpha + \sum_{i} \phi(x_i), \nu + n \right)
\]

note that \( \nabla \nabla p_F(w \mid \alpha, \nu)|_{w_*} = \arg \max p(w \mid \alpha, \nu) = -\nu p(w_* \mid \alpha, \nu) \nabla_w \nabla_w^T \log Z(w_*) \)

In the limit \( n \to \infty \), posterior concentrates at \( w_* \) with

\[
\nabla_w \log Z(w_*) = \frac{\alpha}{n} + \frac{1}{n} \sum_{i=1}^{n} \phi(x_i) = \mathbb{E}_p(\phi(x)) \quad \text{thus} \quad p_w(x \mid w_*) = \arg \min_w D_{KL}(p(x) \parallel p_w(x \mid w))
\]
Learning probability distributions with exponential families

- Given data $x_1, \ldots, x_N$ drawn iid. from unknown $p(x)$, consider approximating $p(x) \approx p_w(x \mid w)$ with an EF
- The maximum likelihood and MAP estimates for $w$ can be computed in $O(N)$
- If the conjugate prior to $p_w$ (which is an EF) is tractable, it allows full Bayesian inference
- Asymptotically, the posterior concentrates around the maximum likelihood estimate, which is the minimizer of the KL-divergence $D_{KL}(p \parallel p_w)$ within the exponential family.
Wouldn’t you want to join this club?
Build your own exponential family!
1. choose features (come up with grand motivation: attraction/repulsion)

   \[ \phi(x) = \begin{bmatrix} -x^2 \\ -x^{-2} \end{bmatrix} \]

2. solve integral (the hard bit)

   \[ Z(w) = \int_0^\infty \exp(-w_1 x^2 - w_2 / x^2) \, dx = \sqrt{\frac{\pi}{w_1}} e^{-2\sqrt{w_1 w_2}} \]

3. profit! The bagel-distribution!

   \[ H(x; w) = \sqrt{\frac{w_1}{\pi}} e^{2\sqrt{w_1 w_2}} e^{-w_1 x^2 - w_2 / x^2} \]

4. don’t know the conjugate prior, though. :(
Let’s fit a distribution!

collecting sufficient statistics

We need

\[
\log Z(w) = -2(w_1 w_2)^{1/2} - \frac{1}{2} \log w_1 + \frac{1}{2} \log \pi
\]

\[
-\nabla_w \log Z(w) = \begin{bmatrix}
\sqrt{\frac{w_2}{w_1} + \frac{1}{2w_1}} \\
\sqrt{\frac{w_1}{w_2}}
\end{bmatrix} \overset{!}{=} -\frac{1}{n} \sum_i \begin{bmatrix}
x_i^2 \\
x_i^{-2}
\end{bmatrix} =: \begin{bmatrix}
\bar{\mu} \\
\bar{\omega}
\end{bmatrix}
\]

\[
\Rightarrow \hat{w}_1 = \frac{1}{2(\bar{\mu} - \bar{\omega})} \quad \hat{w}_2 = \frac{\hat{w}_1}{\bar{\omega}^2}
\]
Summary:

▶ Conjugate Priors allow analytic inference of “nuisance parameters” in probabilistic models
▶ Exponential Families
  ▶ guarantee the existence of conjugate priors, although not always tractable ones
  ▶ allow analytic MAP inference from only a finite set of sufficient statistics

Conjugate prior inference with exponential families is a form of Bayesian regression on distributions. Gaussian process inference, in this sense, is inference on the unknown mean of a Gaussian distribution.

▶ The hardest part is finding the normalization constant. In fact, finding the normalization constant is the only hard part.
▶ Exponential families are a way to turn someone else's integral into an inference algorithm!