

Machine Learning

Bayesian Regression & Classification

Marc Toussaint
University of Stuttgart
Summer 2015

Learning as Inference

- The parameteric view

$$P(\beta|\text{Data}) = \frac{P(\text{Data}|\beta) P(\beta)}{P(\text{Data})}$$

- The function space view

$$P(f|\text{Data}) = \frac{P(\text{Data}|f) P(f)}{P(\text{Data})}$$

- Today:
 - Bayesian (Kernel) Ridge Regression \leftrightarrow Gaussian Process (GP)
 - Bayesian (Kernel) Logistic Regression \leftrightarrow GP classification
 - Bayesian Neural Networks (briefly)

- Beyond learning about specific Bayesian learning methods:

Understand relations between

loss/error \leftrightarrow neg-log likelihood

regularization \leftrightarrow neg-log prior

cost (reg.+loss) \leftrightarrow neg-log posterior

Bayesian (Kernel) Ridge Regression

a.k.a. Gaussian Processes

Ridge regression as Bayesian inference

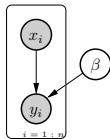
- We have random variables $X_{1:n}, Y_{1:n}, \beta$
- We observe data $D = \{(x_i, y_i)\}_{i=1}^n$ and want to compute $P(\beta | D)$

- Let's assume:

$P(X)$ is arbitrary

$P(\beta)$ is Gaussian: $\beta \sim \mathcal{N}(0, \frac{\sigma^2}{\lambda}) \propto e^{-\frac{\lambda}{2\sigma^2} \|\beta\|^2}$

$P(Y | X, \beta)$ is Gaussian: $y = x^\top \beta + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma^2)$



Ridge regression as Bayesian inference

- Bayes' Theorem:

$$P(\beta | D) = \frac{P(D | \beta) P(\beta)}{P(D)}$$

$$P(\beta | x_{1:n}, y_{1:n}) = \frac{\prod_{i=1}^n P(y_i | \beta, x_i) P(\beta)}{Z}$$

$P(D | \beta)$ is a *product* of independent likelihoods for each observation (x_i, y_i)

Ridge regression as Bayesian inference

- Bayes' Theorem:

$$P(\beta | D) = \frac{P(D | \beta) P(\beta)}{P(D)}$$

$$P(\beta | x_{1:n}, y_{1:n}) = \frac{\prod_{i=1}^n P(y_i | \beta, x_i) P(\beta)}{Z}$$

$P(D | \beta)$ is a *product* of independent likelihoods for each observation (x_i, y_i)

Using the Gaussian expressions:

$$P(\beta | D) = \frac{1}{Z'} \prod_{i=1}^n e^{-\frac{1}{2\sigma^2} (y_i - x_i^\top \beta)^2} e^{-\frac{\lambda}{2\sigma^2} \|\beta\|^2}$$

Ridge regression as Bayesian inference

- Bayes' Theorem:

$$P(\beta | D) = \frac{P(D | \beta) P(\beta)}{P(D)}$$

$$P(\beta | x_{1:n}, y_{1:n}) = \frac{\prod_{i=1}^n P(y_i | \beta, x_i) P(\beta)}{Z}$$

$P(D | \beta)$ is a *product* of independent likelihoods for each observation (x_i, y_i)

Using the Gaussian expressions:

$$P(\beta | D) = \frac{1}{Z'} \prod_{i=1}^n e^{-\frac{1}{2\sigma^2} (y_i - x_i^\top \beta)^2} e^{-\frac{\lambda}{2\sigma^2} \|\beta\|^2}$$

$$-\log P(\beta | D) = \frac{1}{2\sigma^2} \left[\sum_{i=1}^n (y_i - x_i^\top \beta)^2 + \lambda \|\beta\|^2 \right] + \log Z'$$

$$-\log P(\beta | D) \propto L^{\text{ridge}}(\beta)$$

1st insight: The *neg-log posterior* $P(\beta | D)$ is equal to the cost function $L^{\text{ridge}}(\beta)$!

Ridge regression as Bayesian inference

- Let us compute $P(\beta | D)$ explicitly:

$$\begin{aligned}P(\beta | D) &= \frac{1}{Z'} \prod_{i=1}^n e^{-\frac{1}{2\sigma^2} (y_i - x_i^\top \beta)^2} e^{-\frac{\lambda}{2\sigma^2} \|\beta\|^2} \\&= \frac{1}{Z'} e^{-\frac{1}{2\sigma^2} \sum_i (y_i - x_i^\top \beta)^2} e^{-\frac{\lambda}{2\sigma^2} \|\beta\|^2} \\&= \frac{1}{Z'} e^{-\frac{1}{2\sigma^2} [(\mathbf{y} - \mathbf{X}\beta)^\top (\mathbf{y} - \mathbf{X}\beta) + \lambda \beta^\top \beta]} \\&= \frac{1}{Z'} e^{-\frac{1}{2} [\frac{1}{\sigma^2} \mathbf{y}^\top \mathbf{y} + \frac{1}{\sigma^2} \beta^\top (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}) \beta - \frac{2}{\sigma^2} \beta^\top \mathbf{X}^\top \mathbf{y}]} \\&= \mathcal{N}(\beta | \hat{\beta}, \Sigma)\end{aligned}$$

This is a Gaussian with covariance and mean

$$\Sigma = \sigma^2 (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1}, \quad \hat{\beta} = \frac{1}{\sigma^2} \Sigma \mathbf{X}^\top \mathbf{y} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^\top \mathbf{y}$$

- 2nd insight:** The mean $\hat{\beta}$ is exactly the classical $\operatorname{argmin}_{\beta} L^{\text{ridge}}(\beta)$.
- 3rd insight:** The Bayesian inference approach not only gives a mean/optimal $\hat{\beta}$, but also a variance Σ of that estimate!

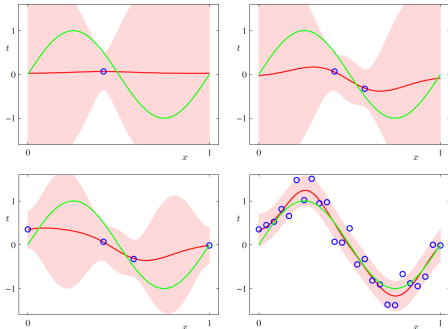
Predicting with an uncertain β

- Suppose we want to make a prediction at x . We can compute the **predictive distribution** over a new observation y^* at x^* :

$$\begin{aligned} P(y^* | x^*, D) &= \int_{\beta} P(y^* | x^*, \beta) P(\beta | D) d\beta \\ &= \int_{\beta} \mathcal{N}(y^* | \phi(x^*)^T \beta, \sigma^2) \mathcal{N}(\beta | \hat{\beta}, \Sigma) d\beta \\ &= \mathcal{N}(y^* | \phi(x^*)^T \hat{\beta}, \sigma^2 + \phi(x^*)^T \Sigma \phi(x^*)) \end{aligned}$$

Note $P(f(x) | D) = \mathcal{N}(f(x) | \phi(x)^T \hat{\beta}, \phi(x)^T \Sigma \phi(x))$ without the σ^2

- So, y^* is Gaussian distributed around the mean prediction $\phi(x^*)^T \hat{\beta}$:



Wrapup of Bayesian Ridge regression

- **1st insight:** The *neg-log posterior* $P(\beta | D)$ is equal to the cost function $L^{\text{ridge}}(\beta)$!

This is a very very common relation: optimization costs correspond to neg-log probabilities; probabilities correspond to exp-neg costs.

- **2nd insight:** The mean $\hat{\beta}$ is exactly the classical $\operatorname{argmin}_{\beta} L^{\text{ridge}}(\beta)$.

More generally, the most likely parameter $\operatorname{argmax}_{\beta} P(\beta | D)$ is also the least-cost parameter $\operatorname{argmin}_{\beta} L(\beta)$. In the Gaussian case, mean and most-likely coincide.

- **3rd insight:** The Bayesian inference approach not only gives a mean/optimal $\hat{\beta}$, but also a variance Σ of that estimate!

This is a core benefit of the Bayesian view: It naturally provides a probability distribution over predictions (“*error bars*”), not only a single prediction.

Kernelized Bayesian Ridge Regression

- As in the classical case, we can consider arbitrary features $\phi(x)$
- .. or directly use a kernel $k(x, x')$:

$$\begin{aligned}P(f(x) | D) &= \mathcal{N}(f(x) | \phi(x)^\top \hat{\beta}, \phi(x)^\top \Sigma \phi(x)) \\ \phi(x)^\top \hat{\beta} &= \phi(x)^\top \mathbf{X}^\top (\mathbf{X} \mathbf{X}^\top + \lambda \mathbf{I})^{-1} \mathbf{y} \\ &= \boldsymbol{\kappa}(x) (\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{y} \\ \phi(x)^\top \Sigma \phi(x) &= \phi(x)^\top \sigma^2 (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1} \phi(x) \\ &= \frac{\sigma^2}{\lambda} \phi(x)^\top \phi(x) - \frac{\sigma^2}{\lambda} \phi(x)^\top \mathbf{X}^\top (\mathbf{X} \mathbf{X}^\top + \lambda \mathbf{I}_k)^{-1} \mathbf{X} \phi(x) \\ &= \frac{\sigma^2}{\lambda} k(x, x) - \frac{\sigma^2}{\lambda} \boldsymbol{\kappa}(x) (\mathbf{K} + \lambda \mathbf{I}_n)^{-1} \boldsymbol{\kappa}(x)^\top\end{aligned}$$

3rd line: As on slide 02:24

last lines: Woodbury identity $(A + UBV)^{-1} = A^{-1} - A^{-1}U(B^{-1} + VA^{-1}U)^{-1}VA^{-1}$
with $A = \lambda \mathbf{I}$

- In standard conventions $\lambda = \sigma^2$, i.e. $P(\beta) = \mathcal{N}(\beta|0, 1)$
 - Regularization: scale the covariance function (or features)

Kernelized Bayesian Ridge Regression

is equivalent to Gaussian Processes

(see also Welling: “Kernel Ridge Regression” Lecture Notes; Rasmussen & Williams sections 2.1 & 6.2; Bishop sections 3.3.3 & 6)

- As we have the equations already, I skip further math details. (See Rasmussen & Williams)

Gaussian Processes

- The function space view

$$P(f|\text{Data}) = \frac{P(\text{Data}|f) P(f)}{P(\text{Data})}$$

- Gaussian Processes define a probability distribution over functions:
 - A function is an infinite dimensional thing – how could we define a Gaussian distribution over functions?
 - For every finite set $\{x_1, \dots, x_M\}$, the function values $f(x_1), \dots, f(x_M)$ are Gaussian distributed with mean and cov.

$$\langle f(x_i) \rangle = \mu(x_i) \quad (\text{often zero})$$

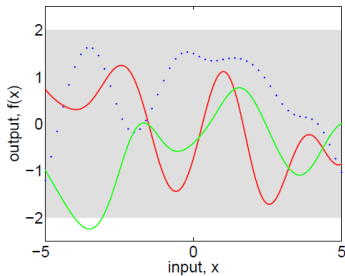
$$\langle [f(x_i) - \mu(x_i)][f(x_j) - \mu(x_j)] \rangle = k(x_i, x_j)$$

Here, $k(\cdot, \cdot)$ is called **covariance function**

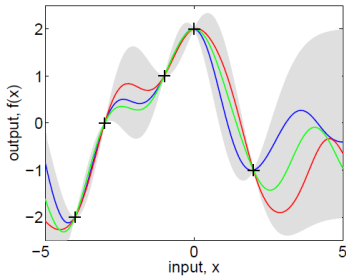
- Second, Gaussian Processes define an observation probability

$$P(y|x, f) = \mathcal{N}(y|f(x), \sigma^2)$$

Gaussian Processes



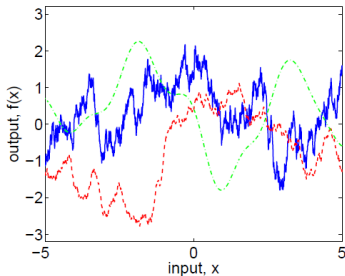
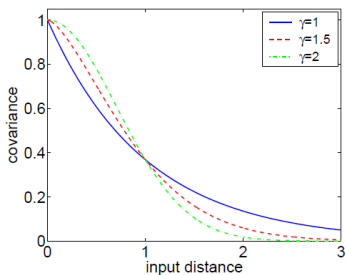
(a), prior



(b), posterior

(from Rasmussen & Williams)

GP: different covariance functions

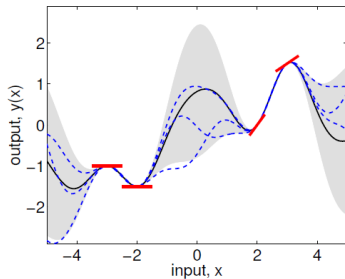
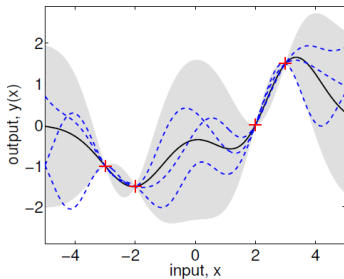


(from Rasmussen & Williams)

- These are examples from the γ -exponential covariance function

$$k(x, x') = \exp\{-|(x - x')/l|^\gamma\}$$

GP: derivative observations



(from Rasmussen & Williams)

- Bayesian Kernel Ridge Regression = Gaussian Process
- GPs have become a standard regression method
- If exact GP is not efficient enough, many approximations exist, e.g. sparse and pseudo-input GPs

Bayesian (Kernel) Logistic Regression

a.k.a. GP classification

Bayesian Logistic Regression

- f now defines a discriminative function:

$$P(X) = \text{arbitrary}$$

$$P(\beta) = \mathcal{N}(\beta|0, \frac{2}{\lambda}) \propto \exp\{-\lambda\|\beta\|^2\}$$

$$P(Y=1 | X, \beta) = \sigma(\beta^\top \phi(x))$$

- Recall

$$L^{\text{logistic}}(\beta) = - \sum_{i=1}^n \log p(y_i | x_i) + \lambda\|\beta\|^2$$

- Again, the parameter posterior is

$$P(\beta|D) \propto P(D | \beta) P(\beta) \propto \exp\{-L^{\text{logistic}}(\beta)\}$$

Bayesian Logistic Regression

- Use **Laplace approximation** (2nd order Taylor for L) at $\beta^* = \operatorname{argmin}_{\beta} L(\beta)$:

$$L(\beta) \approx L(\beta^*) + \bar{\beta}^{\top} \nabla + \frac{1}{2} \bar{\beta}^{\top} H \bar{\beta}, \quad \bar{\beta} = \beta - \beta^*$$

At β^* the gradient $\nabla = 0$ and $L(\beta^*) = \text{const.}$ Therefore

$$\begin{aligned} P(\beta|D) &\propto \exp\{-\bar{\beta}^{\top} \nabla - \frac{1}{2} \bar{\beta}^{\top} H \bar{\beta}\} \\ &= \mathcal{N}(\beta|\beta^*, H^{-1}) \end{aligned}$$

- Then the predictive distribution of the *discriminative function* is also Gaussian!

$$\begin{aligned} P(f(x) | D) &= \int_{\beta} P(f(x) | \beta) P(\beta | D) d\beta \\ &= \int_{\beta} \mathcal{N}(f(x) | \phi(x)^{\top} \beta, 0) \mathcal{N}(\beta | \beta^*, H^{-1}) d\beta \\ &= \mathcal{N}(f(x) | \phi(x)^{\top} \beta^*, \phi(x)^{\top} H^{-1} \phi(x)) =: \mathcal{N}(f(x) | f^*, s^2) \end{aligned}$$

- The predictive distribution over the label $y \in \{0, 1\}$:

$$\begin{aligned} P(y(x)=1 | D) &= \int_{f(x)} \sigma(f(x)) P(f(x)|D) df \\ &\approx \sigma((1 + s^2 \pi/8)^{-\frac{1}{2}} f^*) \end{aligned}$$

which uses a probit approximation of the convolution.

→ The variance s^2 pushes predictive class probabilities towards 0.5.

Kernelized Bayesian Logistic Regression

- As with Kernel Logistic Regression, the MAP discriminative function f^* can be found iterating the Newton method \leftrightarrow iterating GP estimation on a *re-weighted* data set.
- The rest is as above.

Kernel Bayesian Logistic Regression

is equivalent to Gaussian Process Classification

- GP classification became a standard classification method, if the prediction needs to be a meaningful probability that takes the *model uncertainty* into account.

Bayesian Neural Networks

General non-linear models

- Above we always assumed $f(x) = \phi(x)^\top \beta$ (or kernelized)
- Bayesian Learning also works for non-linear function models $f(x, \beta)$
- Regression case:

$P(X)$ is arbitrary.

$P(\beta)$ is Gaussian: $\beta \sim \mathcal{N}(0, \frac{\sigma^2}{\lambda}) \propto e^{-\frac{\lambda}{2\sigma^2} \|\beta\|^2}$

$P(Y | X, \beta)$ is Gaussian: $y = f(x, \beta) + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma^2)$

General non-linear models

- To compute $P(\beta|D)$ we first compute the most likely

$$\beta^* = \underset{\beta}{\operatorname{argmin}} L(\beta) = \underset{\beta}{\operatorname{argmax}} P(\beta|D)$$

- Use Laplace approximation around β^* : 2nd-order Taylor of $f(x, \beta)$ and then of $L(\beta)$ to estimate a Gaussian $P(\beta|D)$
- Neural Networks:
 - The Gaussian prior $P(\beta) = \mathcal{N}(\beta|0, \frac{\sigma^2}{\lambda})$ is called **weight decay**
 - This pushes “sigmoids to be in the linear region”.

Conclusions

- Probabilistic inference is a very powerful concept!
 - Inferring about the world given data
 - Learning, decision making, reasoning can view viewed as forms of (probabilistic) inference
- We introduced Bayes' Theorem as the fundamental form of probabilistic inference
- Marrying Bayes with (Kernel) Ridge (Logistic) regression yields
 - Gaussian Processes
 - Gaussian Process classification