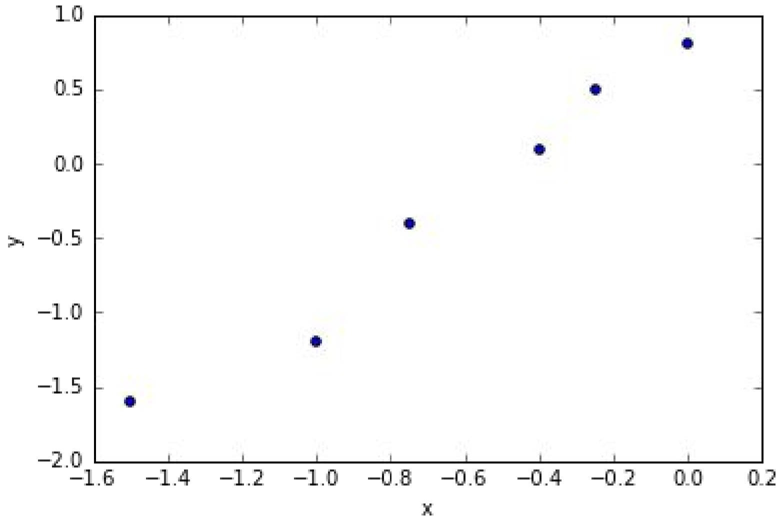
Gaussian Processes and Empirical Bayes

Linear Model

- Dataset

$$D = \{(x_i, y_i) | i = 1, ..., n\}$$



Linear Model

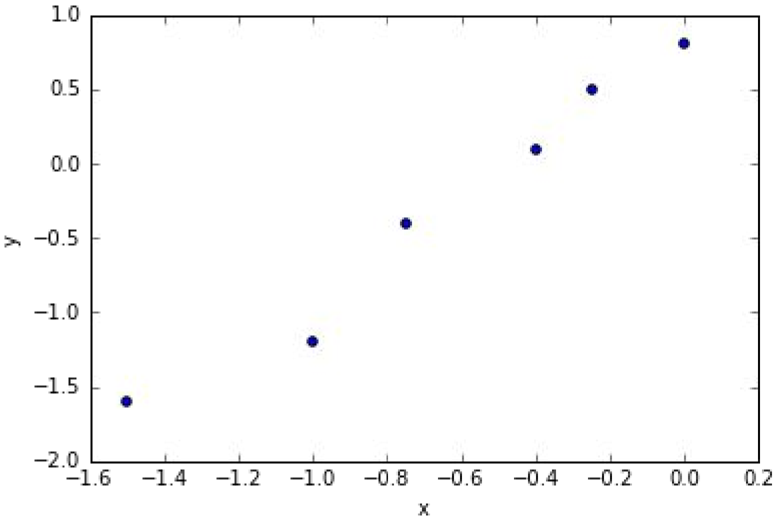
- Dataset

$$D = \{(x_i, y_i) | i = 1, ..., n\}$$

- Fit the data using the standard linear model

$$f(x) = x^T w$$

$$y = f(x) + \epsilon \qquad \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$



Linear Model

- Dataset

$$D = \{(x_i, y_i) | i = 1, ..., n\}$$

- Fit the data using the standard linear model

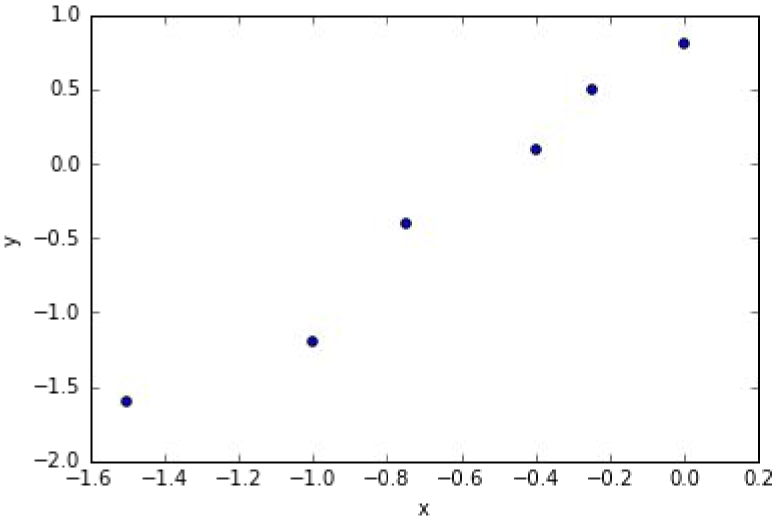
$$f(x) = x^T w$$

function value

$$y = f(x) + \epsilon \qquad \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$

observed target value

Noise



Linear Model

- Dataset

$$D = \{(\mathrm{x}_i, y_i) | i = 1, ..., n\}$$

- Fit the data using the standard linear model

$$f(x) = x^T w$$

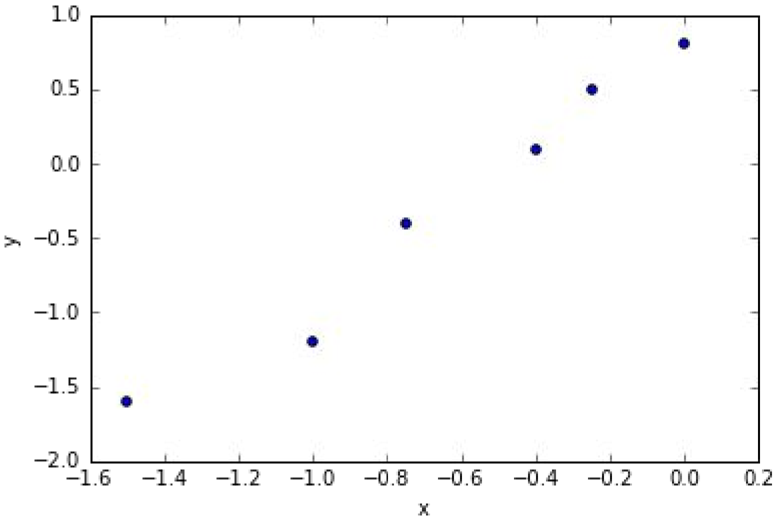
function value

$$y = f(x) + \epsilon$$

observed target value

$$\epsilon \sim \mathcal{N}(0, \sigma_n^2)$$

Noise



- Assumptions
  - *y* differs from *f(x)* by an additive error
  - the error is independent, identically distributed Gaussian distribution

Linear Model with Gaussian Likelihood

- Probability of target value given the data *X* and the parameters *w*

$$p(y|X,w) = \mathcal{N}(X^T w, \sigma_n^2 I)$$

$$= \prod_{i=1}^{n} p(y_i|x_i,w) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_n} \exp(-\frac{(y_i - x_i^T w)^2}{2\sigma_n^2})$$

- The mean is the linear model and the variance is the error
- Notice the simple product - it's due to the observations being assumed independent

Bayesian Linear Model with Gaussian Likelihood

- Specify a prior over the parameters

$$w \sim \mathcal{N}(0, \Sigma_p)$$

- Inference (MAP estimate) in the Bayesian linear model is based on the posterior distribution over the weights

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{marginal likelihood}} \qquad p(\mathbf{w}|\mathbf{y}, X) = \frac{p(\mathbf{y}|X, \mathbf{w})p(\mathbf{w})}{p(\mathbf{y}|X)}$$

- The normalizing constant is the marginal likelihood over *w*

$$p(\mathbf{y}|X) = \int p(\mathbf{y}|X, \mathbf{w})p(\mathbf{w})\, d\mathbf{w}$$

Bayesian Linear Model with Gaussian Likelihood

- Inference in the Bayesian linear model is based on the posterior distribution over the weights

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{marginal likelihood}} \qquad p(\mathbf{w}|\mathbf{y}, X) = \frac{p(\mathbf{y}|X, \mathbf{w})p(\mathbf{w})}{p(\mathbf{y}|X)}$$

- Using proportionality to ignore the normalizing constant we get,

$$\begin{aligned} p(\mathbf{w}|X, \mathbf{y}) &\propto \exp\big(-\frac{1}{2\sigma_n^2}(\mathbf{y} - X^\top \mathbf{w})^\top (\mathbf{y} - X^\top \mathbf{w})\big) \exp\big(-\frac{1}{2}\mathbf{w}^\top \Sigma_p^{-1} \mathbf{w}\big) \\ &\propto \exp\big(-\frac{1}{2}(\mathbf{w} - \bar{\mathbf{w}})^\top \big(\frac{1}{\sigma_n^2} X X^\top + \Sigma_p^{-1}\big)(\mathbf{w} - \bar{\mathbf{w}})\big), \qquad (2.7) \end{aligned}$$

- Therefore,

$$p(\mathbf{w}|X, \mathbf{y}) \sim \mathcal{N}\big(\bar{\mathbf{w}} = \frac{1}{\sigma_n^2} A^{-1} X \mathbf{y},\ A^{-1}\big) \qquad A = \sigma_n^{-2} X X^\top + \Sigma_p^{-1}$$

Relationship between Bayesian Linear Model and Ridge Regression

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{marginal likelihood}} \qquad p(\mathbf{w}|\mathbf{y}, X) = \frac{p(\mathbf{y}|X, \mathbf{w})p(\mathbf{w})}{p(\mathbf{y}|X)}$$

- the penalized maximum likelihood is equivalent to ridge regression

$$\text{posterior} = \frac{\boxed{\text{likelihood} \times \text{prior}}}{\text{marginal likelihood}}$$

- the negative log prior is sometimes thought of as a penalty term
- the likelihood is thought of as the least-squares objective function

- To make prediction over the test data, we average over all possible parameter values, weighted by their posterior probability:

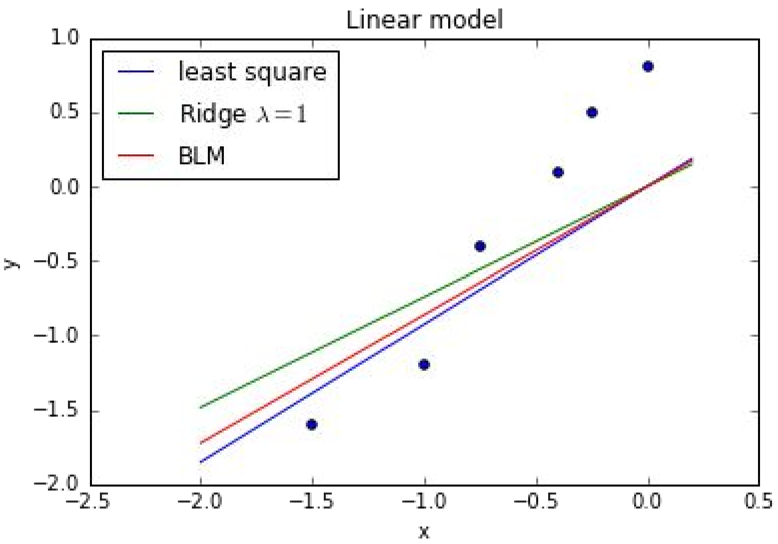
$$p(f_*|\mathbf{x}_*, X, \mathbf{y}) = \int p(f_*|\mathbf{x}_*, \mathbf{w})p(\mathbf{w}|X, \mathbf{y})\, d\mathbf{w}$$
$$= \mathcal{N}\Big(\frac{1}{\sigma_n^2}\mathbf{x}_*^\top A^{-1}X\mathbf{y},\ \mathbf{x}_*^\top A^{-1}\mathbf{x}_*\Big). \qquad A = \sigma_n^{-2}XX^\top + \Sigma_p^{-1}$$

Different Linear Model formulations

- Least square

$$w = (X^T X)^{-1} X^T y$$

$$\hat{y} = x^T w$$



- Ridge Regression

$$w = (X^T X + \lambda I)^{-1} X^T y$$

- Bayesian linear model - the mean of the posterior distribution

$$p(\mathbf{w}|X, \mathbf{y}) \sim \mathcal{N}(\bar{\mathbf{w}} = \frac{1}{\sigma_n^2} A^{-1} X \mathbf{y},\ A^{-1})$$

$$\bar{\mathbf{w}} = \frac{1}{\sigma_n^2} A^{-1} X \mathbf{y}$$

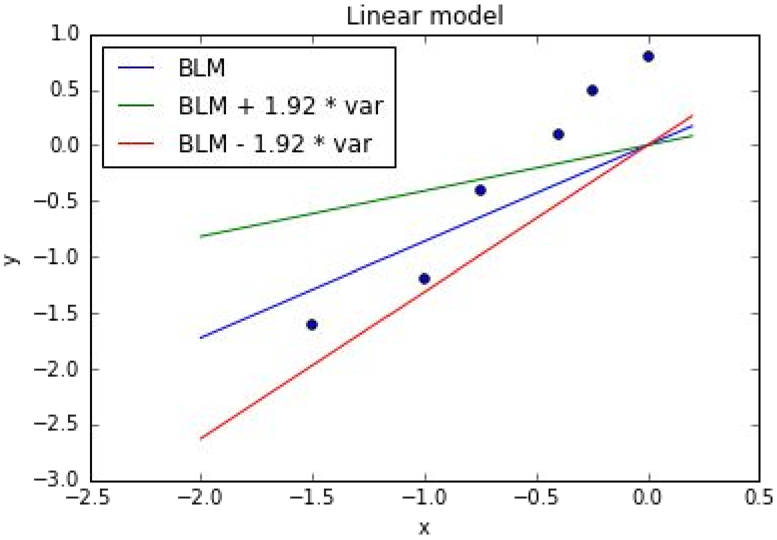
$$A = \sigma_n^{-2} X X^\top + \Sigma_p^{-1}$$

Different Linear Model formulations

- Least square

$$w = (X^T X)^{-1} X^T y$$

$$\hat{y} = x^T w$$



- Ridge Regression

$$w = (X^T X + \lambda I)^{-1} X^T y$$

- Bayesian linear model - the mean of the posterior distribution

$$p(\mathbf{w}|X, \mathbf{y}) \sim \mathcal{N}(\bar{\mathbf{w}} = \frac{1}{\sigma_n^2} A^{-1} X \mathbf{y},\ A^{-1})$$

$$\bar{\mathbf{w}} = \frac{1}{\sigma_n^2} A^{-1} X \mathbf{y}$$

$$A = \sigma_n^{-2} X X^\top + \Sigma_p^{-1}$$

Feature-space interpretation

- Linear model suffer from limited expressiveness - assumes data is linearly separable

- To resolve this,
    1. project the inputs into some high dimensional space using a set of basis functions (e.g. polynomial)

$$\phi(x) = (1, x, x^2, x^3, \ldots)^\top$$

    2. fit a linear model in this new space

$$f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w}$$

- Prediction over the test data thus becomes,

$$f_* | \mathbf{x}_*, X, \mathbf{y} \sim \mathcal{N}\left(\frac{1}{\sigma_n^2} \phi(\mathbf{x}_*)^\top A^{-1} \Phi \mathbf{y},\ \phi(\mathbf{x}_*)^\top A^{-1} \phi(\mathbf{x}_*)\right)$$

$$A = \sigma_n^{-2} \Phi \Phi^\top + \Sigma_p^{-1}$$

Feature-space interpretation

- Prediction over the test data thus becomes,

$$f_*|\mathbf{x}_*, X, \mathbf{y} \sim \mathcal{N}\left(\frac{1}{\sigma_n^2}\phi(\mathbf{x}_*)^\top A^{-1}\Phi\mathbf{y},\ \phi(\mathbf{x}_*)^\top A^{-1}\phi(\mathbf{x}_*)\right)$$

$$A = \sigma_n^{-2}\Phi\Phi^\top + \Sigma_p^{-1}$$

- An alternative formulation is the following (helps with the kernel trick)

$$\begin{aligned} f_*|\mathbf{x}_*, X, \mathbf{y} \sim \mathcal{N}\big(&\phi_*^\top\Sigma_p\Phi(K+\sigma_n^2 I)^{-1}\mathbf{y},\\ &\phi_*^\top\Sigma_p\phi_* - \phi_*^\top\Sigma_p\Phi(K+\sigma_n^2 I)^{-1}\Phi^\top\Sigma_p\phi_*\big), \end{aligned}$$

The Kernel trick

- Transforming the feature space into higher dimensional space can be computationally and memory extensive
- Consider the following formulation

$$f_*|\mathbf{x}_*, X, \mathbf{y} \sim \mathcal{N}(\boldsymbol{\phi}_*^\top \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \mathbf{y},$$
$$\boldsymbol{\phi}_*^\top \Sigma_p \boldsymbol{\phi}_* - \boldsymbol{\phi}_*^\top \Sigma_p \Phi (K + \sigma_n^2 I)^{-1} \Phi^\top \Sigma_p \boldsymbol{\phi}_*),$$

- Notice that the feature space are in these forms,

$$\Phi^\top \Sigma_p \Phi, \ \boldsymbol{\phi}_*^\top \Sigma_p \Phi, \text{ or } \boldsymbol{\phi}_*^\top \Sigma_p \boldsymbol{\phi}_*$$

- We can replace these terms by the kernel function defined as:

$$k(\mathbf{x}, \mathbf{x}') = \boldsymbol{\phi}(\mathbf{x})^\top \Sigma_p \boldsymbol{\phi}(\mathbf{x}') = \boldsymbol{\psi}(\mathbf{x}) \cdot \boldsymbol{\psi}(\mathbf{x}') \qquad \boldsymbol{\psi}(\mathbf{x}) = \Sigma_p^{1/2} \boldsymbol{\phi}(\mathbf{x})$$

- This computes the inner products between pairs in the dataset (implicitly using higher order features) instead of explicitly computing the new features in the higher dimensional space - this is known as the kernel trick

The Kernel trick

- Polynomial kernel: https://en.wikipedia.org/wiki/Polynomial_kernel

Building models with Gaussians

- Under the bayesian context we often work with integrations for computing marginals
- The normal distribution is easy to work with

$$p(y \mid m, \Sigma) = (2\pi)^{-k/2}|\Sigma|^{-1/2} \exp\left\{-\frac{1}{2}(y-\mu)'\Sigma^{-1}(y-\mu)\right\}$$

- Marginals of the normal distribution are normally distributed

$$p(x,y) = \mathcal{N}\left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{xy}^T & \Sigma_y \end{bmatrix}\right)$$

$$p(x) = \int p(x,y)dy = \mathcal{N}(\mu_x, \Sigma_x)$$

- conditionals of multivariate normals are normal

$$p(x|y) = \mathcal{N}(\mu_x + \Sigma_{xy}\Sigma_y^{-1}(y-\mu_y), \Sigma_x - \Sigma_{xy}\Sigma_y^{-1}\Sigma_{xy}^T)$$

Gaussian processes

- A Gaussian process is completely specified by its mean function and covariance function

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})], \\ k(\mathbf{x},\mathbf{x}') &= \mathbb{E}[(f(\mathbf{x})-m(\mathbf{x}))(f(\mathbf{x}')-m(\mathbf{x}'))], \end{aligned} \tag{2.13}$$

$$f(\mathbf{x}) \sim \mathcal{GP}\big(m(\mathbf{x}), k(\mathbf{x},\mathbf{x}')\big). \tag{2.14}$$

- We can derive a simple Gaussian process from the bayesian regression model

$$f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w} \quad \text{with prior } \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \Sigma_p)$$

- The function values of two samples ***x*** and ***x'*** are jointly Gaussian with zero mean and covariance $\phi(\mathbf{x})^\top \Sigma_p \phi(\mathbf{x}')$. This is due to the fact that,

$$\begin{aligned} \mathbb{E}[f(\mathbf{x})] &= \phi(\mathbf{x})^\top \mathbb{E}[\mathbf{w}] = 0, \\ \mathbb{E}[f(\mathbf{x})f(\mathbf{x}')] &= \phi(\mathbf{x})^\top \mathbb{E}[\mathbf{w}\mathbf{w}^\top]\phi(\mathbf{x}') = \phi(\mathbf{x})^\top \Sigma_p \phi(\mathbf{x}'). \end{aligned} \tag{2.15}$$

Gaussian processes

- A Gaussian process is completely specified by its mean function and covariance function

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})], \\ k(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))], \end{aligned} \quad (2.13)$$

$$f(\mathbf{x}) \sim \mathcal{GP}\big(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\big). \quad (2.14)$$

- We can derive a simple Gaussian process from the bayesian regression model

$$f(\mathbf{x}) = \phi(\mathbf{x})^\top \mathbf{w} \ \text{ with prior } \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \Sigma_p)$$

- The function values of two samples ***x*** and ***x'*** are jointly Gaussian with zero mean and covariance $\phi(\mathbf{x})^\top \Sigma_p \phi(\mathbf{x}')$. This is due to the fact that,

The kernel function

$$\begin{aligned} \mathbb{E}[f(\mathbf{x})] &= \phi(\mathbf{x})^\top \mathbb{E}[\mathbf{w}] = 0, \\ \mathbb{E}[f(\mathbf{x})f(\mathbf{x}')] &= \phi(\mathbf{x})^\top \mathbb{E}[\mathbf{w}\mathbf{w}^\top]\phi(\mathbf{x}') = \phi(\mathbf{x})^\top \Sigma_p \phi(\mathbf{x}'). \end{aligned} \quad (2.15)$$

Gaussian processes

- Therefore, the distribution over a set of function values is given as,

$$\mathbf{f}_* \sim \mathcal{N}\big(\mathbf{0}, K(X_*, X_*)\big)$$

Squared exponential kernel

$$k(x, x') = \theta_1 \exp\left(-\frac{\theta_2}{2}(x - x')^2\right)$$

- Given a training set f and a testing set f*, their joint distribution is according to the following prior,

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X, X) & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}\right).$$

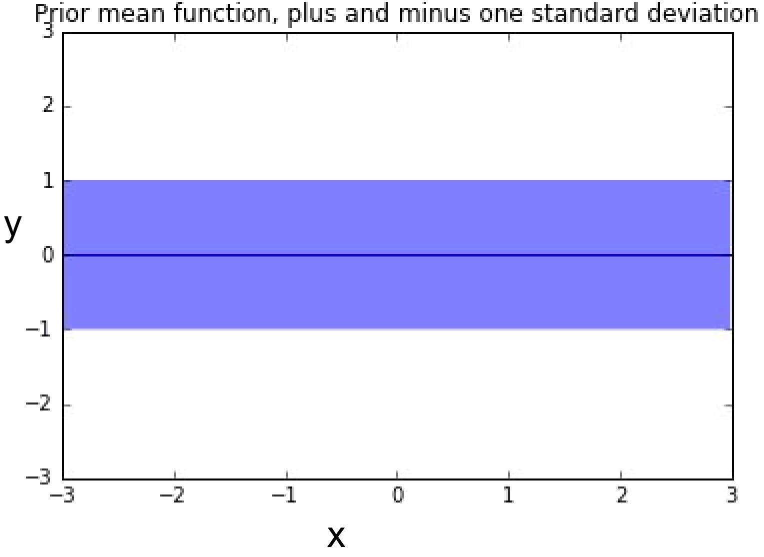
- Conditioning the joint Gaussian prior distribution on the observations gets us the following posterior

$$\mathbf{f}_* | X_*, X, \mathbf{f} \sim \mathcal{N}\big(K(X_*, X)K(X, X)^{-1}\mathbf{f},\ K(X_*, X_*) - K(X_*, X)K(X, X)^{-1}K(X, X_*)\big).$$

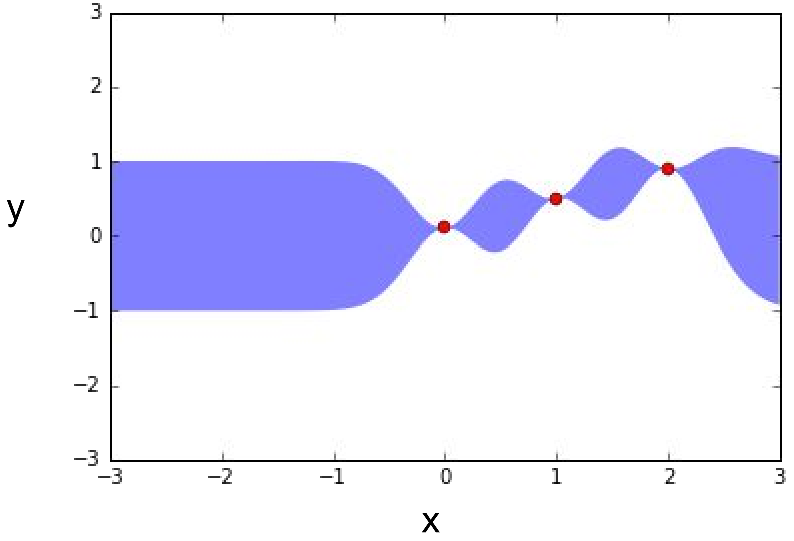
Empirical Gaussian processes

$$\mathbf{f}_* \sim \mathcal{N}\big(\mathbf{0}, K(X_*, X_*)\big)$$

$$\mathbf{f}_* | X_*, X, \mathbf{f} \sim \mathcal{N}\big(K(X_*, X)K(X, X)^{-1}\mathbf{f},\\ K(X_*, X_*) - K(X_*, X)K(X, X)^{-1}K(X, X_*)\big).$$



X = [0 , 1, 2] y = [0.1, 0.5, 0.9]



Empirical Bayes

- Observations $y = \{y_1, y_2, ..., y_n\}$
- Assume that $y_i \sim \mathcal{N}(w^T x_i, \sigma_i^2)$
- Prior on ***w*** : $w_j \sim \mathcal{N}(0, \lambda_j^{-1})$

- Type I maximum likelihood: $\text{argmax}_w \; p(y \mid w, X)$
- Type I MAP estimate: $\text{argmax}_w \, p(y \mid X, w)p(w)$

- Type II maximum likelihood: $\text{argmax}_\lambda \; p(y \mid X, \lambda)$
- Type II MAP estimate: $\text{argmax}_\lambda \, p(y \mid X, \lambda)p(\lambda)$

$$\text{argmax}_w \; p(y \mid X, \lambda) = \int p(y, w \mid \lambda, X)dw = \int p(y \mid \lambda, X, w)p(w \mid \lambda)dw$$

Empirical Bayes

- Observations $y = \{y_1, y_2, ..., y_n\}$
- Assume that $y_i \sim \mathcal{N}(w^T x_i, \sigma_i^2)$
- Prior on ***w*** : $w_j \sim \mathcal{N}(0, \lambda_j^{-1})$

- Type II maximum likelihood: $\text{argmax}_\lambda \; p(y \mid X, \lambda)$

$$\text{argmax}_\lambda \;\; p(y \mid X, \lambda) = \int p(y, w \mid \lambda, X) dw = \int p(y \mid \lambda, X, w) p(w \mid \lambda) dw$$

$$= \text{argmax}_\lambda \log |\Sigma_y| + y^T \Sigma_y^{-1} y$$

- Optimize the objective function using gradient descent, MCMC, coordinate descent etc.

Empirical Bayes

$f(x) = x^T w$

function value

$y = f(x) + \epsilon$ — observed target value

$\epsilon \sim \mathcal{N}(0, \sigma_n^2)$ — Noise

- Observations $y = \{y_1, y_2, ..., y_n\}$
- Assume that $y_i \sim \mathcal{N}(w^T x_i, \sigma_i^2)$
- Prior on ***w*** : $w_j \sim \mathcal{N}(0, \lambda_j^{-1})$

- Type II maximum likelihood: $\text{argmax}_\lambda \; p(y \mid X, \lambda)$

$$\text{argmax}_\lambda \; p(y \mid X, \lambda) = \int p(y, w \mid \lambda, X) dw = \int p(y \mid \lambda, X, w) p(w \mid \lambda) dw$$

$$= \text{argmax}_\lambda \log |\Sigma_y| + y^T \Sigma_y^{-1} y$$

- Optimize the objective function using gradient descent, MCMC, coordinate descent etc.

- Also known as Automatic relevance determination, similar to the L1 regularization term, leads to sparse solutions

Automatic Relevance Determination

- Consider the following kernel

$$k(\mathbf{x}_p, \mathbf{x}_q) \;=\; \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x}_p - \mathbf{x}_q)^\top M (\mathbf{x}_p - \mathbf{x}_q)\right)$$

where,

$$M \;= \mathrm{diag}(\boldsymbol{\ell})^{-2} \qquad \boldsymbol{\ell} = \ell_1, \ldots, \ell_D$$

- $\boldsymbol{\ell}$ defines the length-scale - a measure of how far you need to move (along a particular axis) in input space for the function values to become uncorrelated
- the inverse of the length-scale determines how relevant an input is: if the length-scale has a very large value the covariance will become almost independent of that input, effectively removing it from the inference
- Equivalent to L1-Regularization but generates more sparse solutions

Demonstration

- Consider the following kernel

$$k(\mathbf{x}_p, \mathbf{x}_q) \;=\; \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x}_p - \mathbf{x}_q)^\top M (\mathbf{x}_p - \mathbf{x}_q)\right)$$

where, $M = \operatorname{diag}(\boldsymbol{\ell})^{-2} \quad \boldsymbol{\ell} = \ell_1, \ldots, \ell_D$