Bayesian Kernel Methods

THE AUSTRALIAN NATIONAL UNIVERSITY

Unit 1: Bayes Rule, Approximate Inference, Hyperparameters

Unit 2: Gaussian Processes, Covariance Function, Kernel

Unit 3: GP: Regression

Unit 4: GP: Classification

Unit 5: Implementation: Laplace Approximation, Low Rank Methods

Unit 6: Implementation: Low Rank Methods, Bayes Committee Machine

Unit 7: Relevance Vector Machine: Priors on Coefficients

Unit 8: Relevance Vector Machine: Efficient Optimization and Extensions

http://mlg.anu.edu.au/~smola/summer2002/

Overview of Unit 1: Bayesics

THE AUSTRALIAN NATIONAL UNIVERSITY

01: Parametric Density Models
02: Maximum Likelihood
03: Example: Mean and Variance
04: Bayes' Rule and Conditional Probabilities
05: Example: Unfair Jury
06: Priors
07: Example: Prior on Function Space
08: Bayesian Inference
09: Confidence Intervals
10: Problems with Exact Inference
11: Maximum a Posteriori Approximation
12: Laplace Approximation for Confidence Intervals
13: Relation to Regularized Risk Functional
14: Hyperparameters
15: MAP2 Approximation
16: To integrate or not to integrate

Parametric Density Models

THE AUSTRALIAN NATIONAL UNIVERSITY

Goal: We want to estimate the density of a random variable, say, $\mathbf{x}$, given a set of observations $X := \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$.

Problem: Without additional knowledge, this is very difficult (and we need lots of data).

"Solution:" Assume **a lot** about $p(\mathbf{x})$ and X.

Assumption 1: The set X has been obtained by drawing **independent identically distributed** samples from $p(\mathbf{x})$.

This assumption will hold throughout the lectures.

$$\text{It follows that } p(X) = p(\mathbf{x}_1, \ldots, \mathbf{x}_m) = \prod_{i=1}^{m} p(\mathbf{x}_i)$$

Assumption 2: The density $p(\mathbf{x})$ can be parameterized by θ, that is $p(\mathbf{x}) = p(\mathbf{x}|\theta)$.

Caution: We should write $p_\theta(\mathbf{x})$ to indicate that $p(\mathbf{x})$ is **parameterized** by θ, rather than the density of $\mathbf{x}$, given θ. But it will be useful later ...

Maximum Likelihood

THE AUSTRALIAN NATIONAL UNIVERSITY

Inference Principle: Find θ such that $p(X|\theta)$ is maximized. This means maximizing

$$p(X|\theta) = \prod_{i=1}^{m} p(\mathbf{x}_i|\theta) \text{ or equivalently } \log p(X|\theta) = \sum_{i=1}^{m} \log p(\mathbf{x}_i|\theta).$$

Likelihood: $p(X|\theta)$ as a **function of** θ is commonly referred to as the likelihood $\mathcal{L}(\theta)$. Thereby we can find the parameter θ that is most plausible given X by maximizing $\mathcal{L}(\theta)$.

Numerical Trick: Typically we minimize $-\log \mathcal{L}$, that is, we minimize $\sum_{i=1}^{m} -\log p(\mathbf{x}_i|\theta)$

Blue: Similarity to training error for regularized risk, here the error per observation corresponds to $-\log(x_i|\theta)$.

Problem 1: The maximum value of $\mathcal{L}$ can be misleading, since $p(\mathbf{x}|\theta)$ may not be the right model (**approximation error**).

Problem 2: We may not have enough data to adjust θ properly, so the maximum value of $\mathcal{L}$ may be misleadingly **high**.

Example: Mean and Variance

THE AUSTRALIAN NATIONAL UNIVERSITY

Normal Distribution: Estimate parameters $\theta := (\mu, \sigma^2)$ for a normal distribution

$$p(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{\sigma^2}\right)$$

Negative Log-Likelihood:

$$-\log \mathcal{L}(\mu, \theta) = -\frac{m}{2}\log 2\pi - m\log\sigma + \frac{1}{2\sigma^2}\sum_{i=1}^{m}(x_i - \mu)^2$$

Optimum for μ: (we assume $\sigma^2 \neq 0$)

$$\partial_\mu - \log \mathcal{L}(\mu, \sigma^2) = \frac{1}{\sigma^2}\sum_{i=1}^{m} x_i - \mu = 0 \iff \mu = \frac{1}{m}\sum_{i=1}^{m} x_i.$$

Optimum for σ^2: (we assume $\sigma^2 \neq 0$)

$$\partial_\sigma - \log \mathcal{L}(\mu, \sigma^2) = -\frac{m}{\sigma} + \frac{1}{\sigma^3}\sum_{i=1}^{m} = 0 \iff \sigma^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_i)^2.$$

Bayes' Rule and Conditional Probabilities

THE AUSTRALIAN NATIONAL UNIVERSITY

Joint Probabilitiy: $\Pr(X, Y)$ is the probability of the events X and Y occurring simultaneously.

Conditional Probabilitiy: $\Pr(X|Y)$ is the probability of the event X, given Y.

Bayes Rule: Joint and Conditional Probability are related by $\Pr(X, Y) = \Pr(X|Y)\Pr(Y)$. We may therefore expand $\Pr(X, Y)$ in X and Y to obtain

$$\Pr(X|Y) = \frac{\Pr(Y|X)\Pr(X)}{\Pr(Y)}$$

Joint Density: $\Pr(\mathbf{x}, \mathbf{y})$ is the density of the events $\mathbf{x}$ and $\mathbf{y}$ occurring simultaneously.

Conditional Density: $\Pr(\mathbf{x}|\mathbf{y})$ is the density of the event $\mathbf{x}$, given $\mathbf{y}$.

Bayes Rule: Joint and Conditional Density are related by

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x}|\mathbf{y})p(\mathbf{y}) \text{ and therefore } p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})}.$$

Examples

AIDS-Test:

We want to find out likely it is that a patient *really* has AIDS (denoted by X) if the test is positive (denoted by Y).

Roughly 0.1% of all Australians are infected ($\Pr(X) = 0.001$). The probability that an AIDS test tells us the wrong result is in the order of 1% ($\Pr(Y|\mathcal{X}\backslash X) = 0.01$) and moreover we assume that it detects all infections ($\Pr(Y|X) = 1$). We have

$$\Pr(X|Y) = \frac{\Pr(Y|X)\Pr(X)}{\Pr(Y)} = \frac{\Pr(Y|X)\Pr(X)}{\Pr(Y|X)\Pr(X) + \Pr(Y|\mathcal{X}\backslash X)\Pr(\mathcal{X}\backslash X)}$$

Hence $\Pr(X|Y) = \frac{1 \cdot 0.001}{1 \cdot 0.001 + 0.01 \cdot 0.999} = 0.091$, i.e. the probability of AIDS is 9.1%!

Reliability of Eye-Witness:

Assume that an eye-witness is 90% sure and that there were 20 people at the crime scene, what is the probability that the guy identified committed the crime?

$$\Pr(X|Y) = \frac{0.9 \cdot 0.05}{0.9 \cdot 0.05 + 0.1 \cdot 0.95} = 0.3213 = 32\% \text{ that's a worry } \ldots$$

Priors

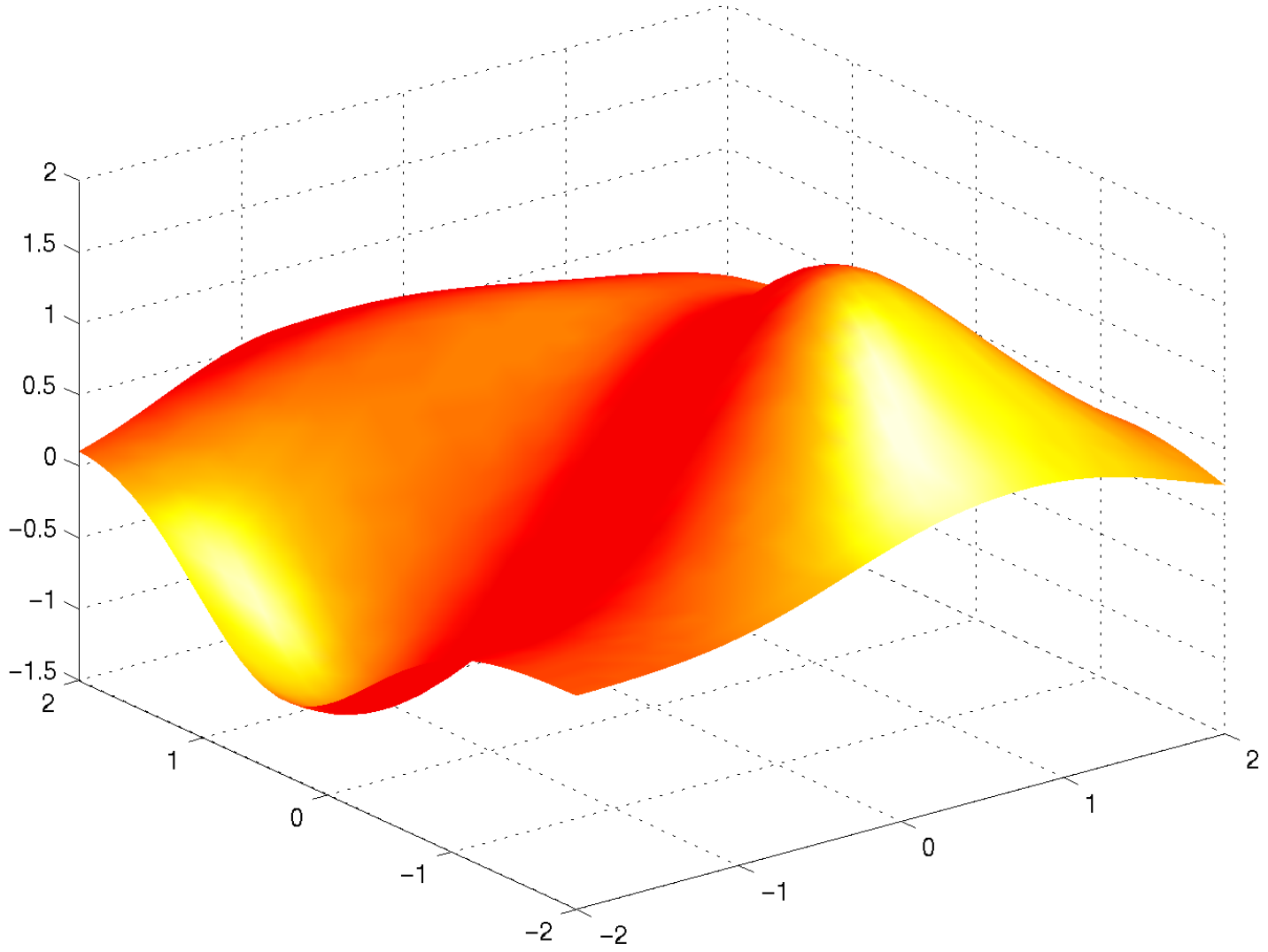
Idea 1

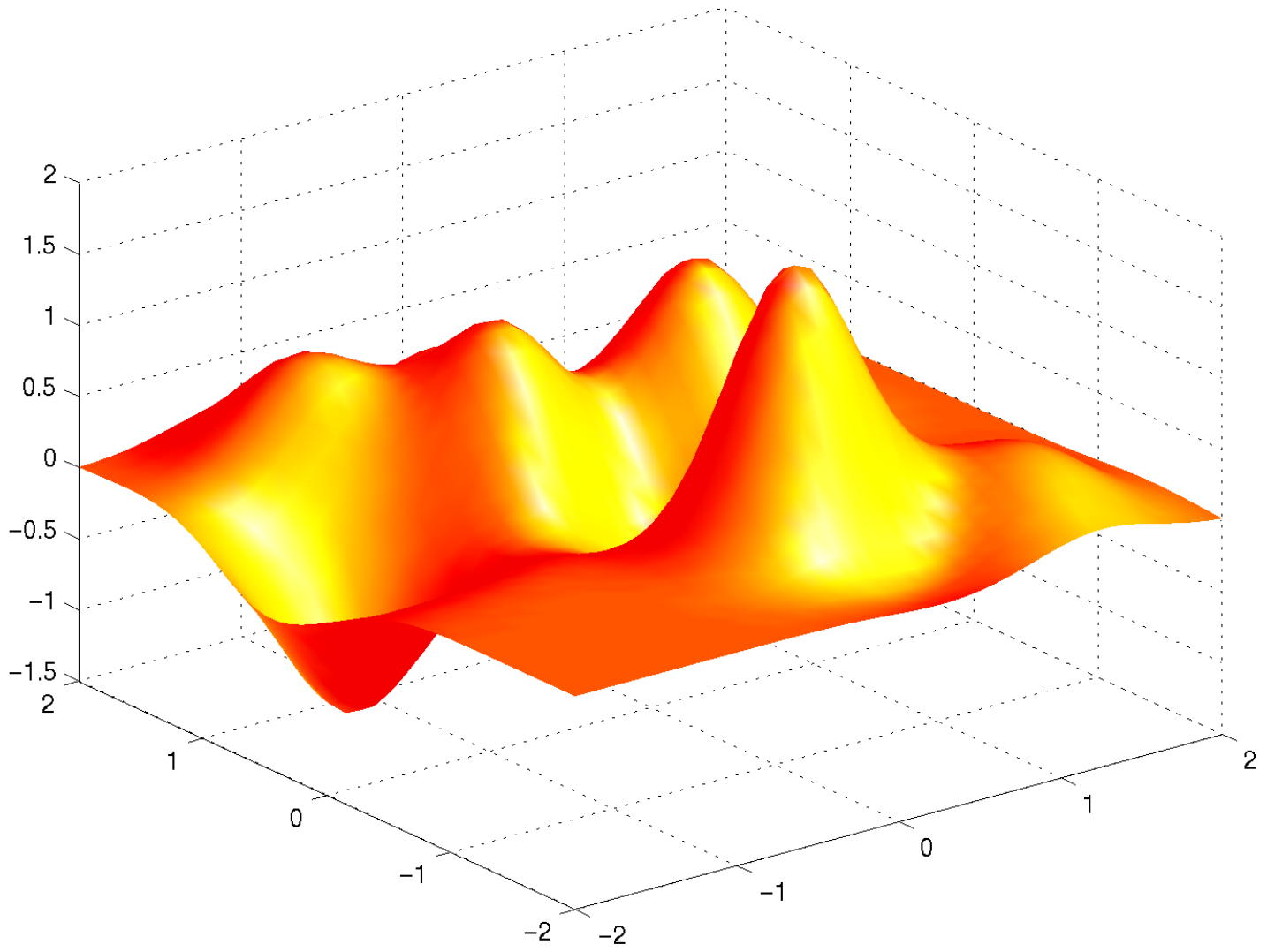
Quite often we have a rough idea of what function we can expect beforehand.

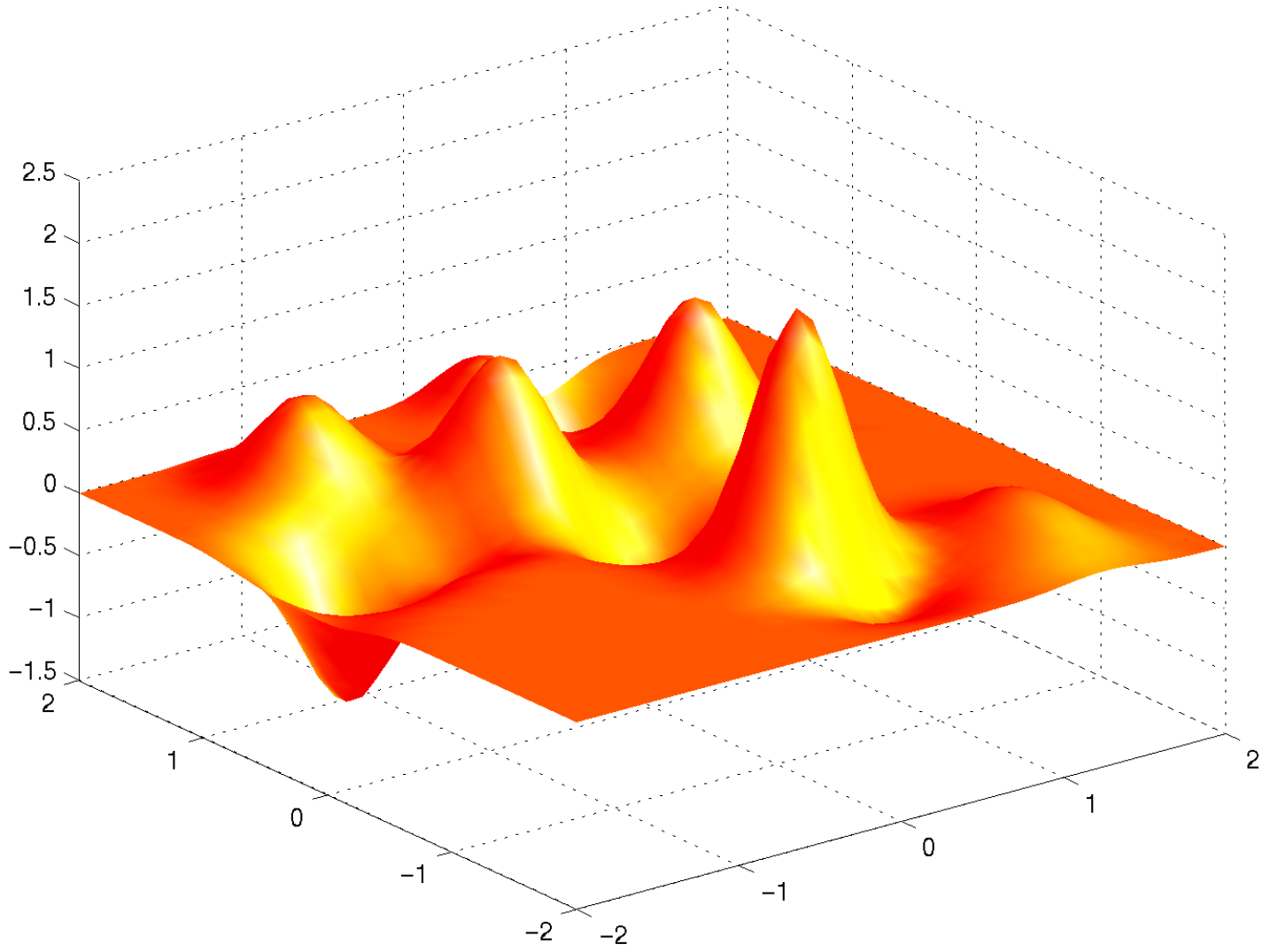
- We observe similar functions in practice.
- We **think** that e.g. smooth functions should be more likely.
- We **would like** a certain type of functions.
- We have **prior knowledge** about specific properties, e.g. vanishing second derivative, etc.

Idea 2

We have to specify somehow, how likely it is to observe a specific function f from an overall class of functions. This is done by assuming some density $p(f)$ describing how likely we are to observe f.


2
1.5
1
0.5
0
−0.5
−1
−1.5
2
1
0
−1
−2
−2
−1
0
1
2


2
1.5
1
0.5
0
−0.5
−1
−1.5
2
1
0
−1
−2
−2
−1
0
1
2


2.5
2
1.5
1
0.5
0
−0.5
−1
−1.5
2
1
0
−1
−2
−2
−1
0
1
2

Example: Prior on Function Space

THE AUSTRALIAN NATIONAL UNIVERSITY

Speech Signal

We know that the signal is bandlimited, hence any signal containing frequency components above 10kHz has density 0.

Parametric Prior

We may know that f is a linear combination of $\sin x, \cos x, \sin 2x,$ and $\cos 2x$ and that the coefficients may be chosen from the interval $[-1, 1]$.

$$p(f) = \begin{cases} \frac{1}{16} & \text{if } f = \alpha_1 \sin x + \alpha_2 \cos x + \alpha_3 \sin 2x + \alpha_4 \cos 2x \text{ with } \alpha_i \in [-1, 1] \\ 0 & \text{otherwise} \end{cases}$$

Prior on Function Values

We assume that there is a correlation between the function values f_i at location $f(\mathbf{x}_i)$. There we have

$$p(f_1, f_2, f_3) = \frac{1}{\sqrt{(2\pi)^3 \det K}} \exp\left(-\frac{1}{2}(f_1, f_2, f_3)^\top K^{-1}(f_1, f_2, f_3)\right).$$

Bayesian Inference

THE AUSTRALIAN NATIONAL UNIVERSITY

Applying Bayes Rule:

We want to infer the probability of f, having observed X, Y. By Bayes' rule we obtain

$$p(f|X,Y) = \frac{p(Y|f,X)p(f|X)}{p(Y|X)} \propto p(Y|f,X)p(f|X).$$

This is also often called the **posterior probability** of observing f, after that the data X, Y arrived.

Usual Assumption:

Typically we assume that X has no influence as to which f we may assume, i.e. $p(f|X) = p(f)$ (X and f are independent random variables).

Prediction: Given $p(f|X,Y)$ we can predict $f(\mathbf{x})$ via

$$\int f(\mathbf{x})p(f|X,Y)df = \frac{1}{Z}\int f(\mathbf{x})p(Y|f,X)dp(f) \text{ where } Z = \int p(Y|f,X)dp(f)$$

Confidence

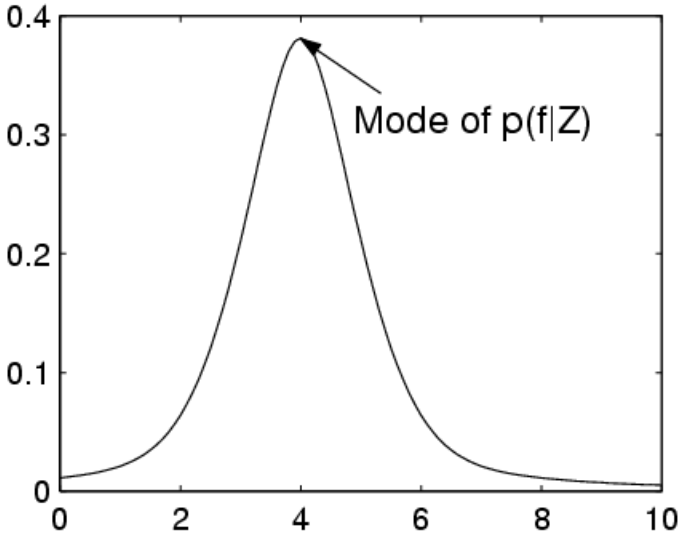
THE AUSTRALIAN NATIONAL UNIVERSITY

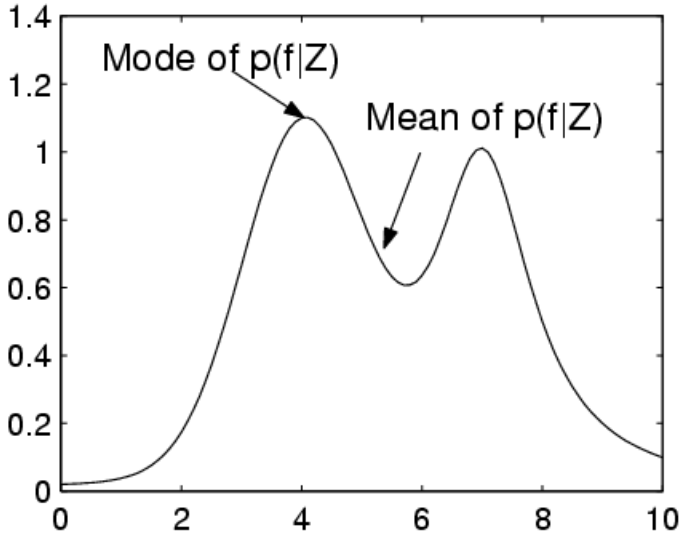
Variance:

Likewise, to infer the predictive variance we compute

$$\mathbf{E}\left[(f(\mathbf{x}) - \mathbf{E}[f(\mathbf{x})])^2\right] = \int (f(\mathbf{x}) - \mathbf{E}[f(\mathbf{x})])^2 \, p(f|X,Y)df$$

This means that we can estimate the variation of $f(\mathbf{x})$, given the data and our prior knowledge about f, as encoded by $p(f)$.





Problems with Exact Inference

THE AUSTRALIAN NATIONAL UNIVERSITY

Problem

Nobody wants to compute integrals, because ...

- Computing integrals is expensive
- No closed form possible
- Not very intuitive for inference

Idea

After all, we are only **averaging**, so replace the mean of the distribution by the mode and hope that it will be ok. This leads to the maximum a posteriori estimate (see next slide).

Problem

Error bars are really hard to obtain.

Idea

Approximate $p(f|X,Y)$ by a normal distribution (Laplace Approximation).

Maximum a Posteriori Approximation

THE AUSTRALIAN NATIONAL UNIVERSITY

Maximizing the Posterior Probability

To find the hypothesis f with the highest posterior probability we have to maximize

$$p(f|X,Y) = \frac{p(Y|f,X)p(f|X)}{p(Y|X)}$$

Lazy Trick

Since we only want f (and $p(Y|X)$ is independent of f), all we have to do is maximize $p(Y|f,X)p(f)$.

Taking Logs

For convenience we get f by minimizing

$$-\log p(Y|f,X)p(f|X) = -\log p(Y|f,X) - \log p(f) = -\log \mathcal{L} - \log p(f)$$

So all we are doing is to **reweight the likelihood** by $-\log p(f)$. This looks suspiciously like the regularization term. We will match up the two terms later.

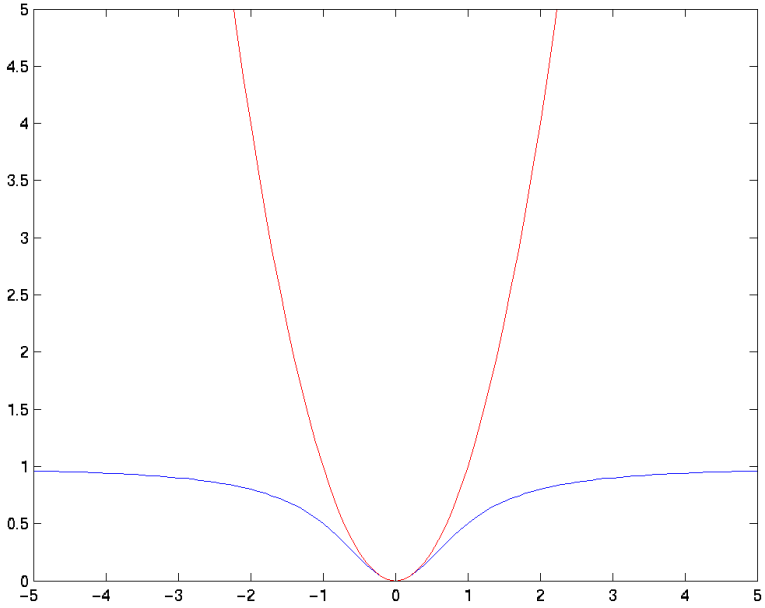
Laplace Approximation for Confidence Intervals

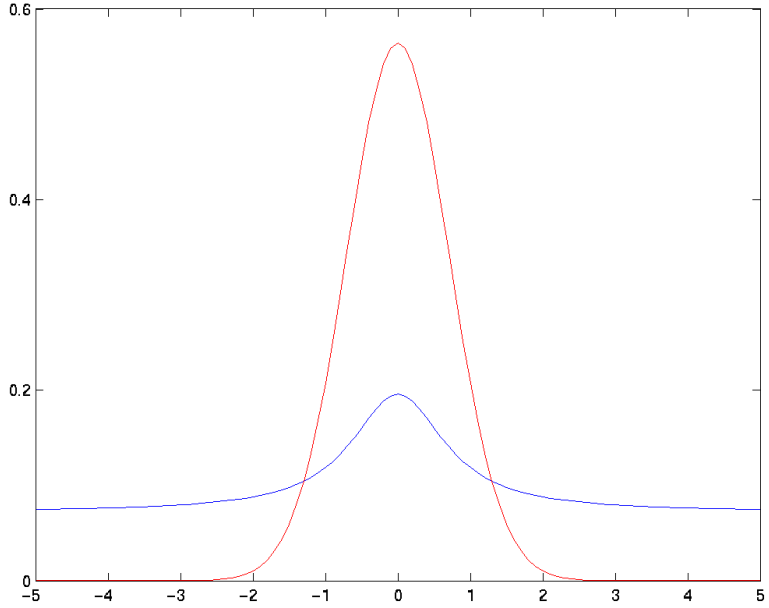
THE AUSTRALIAN NATIONAL UNIVERSITY

Variance

Once we found the **mode** f_0 of the distribution, we might as well approximate the variance by approximating $p(f|X,Y)$ with a normal distribution around f_0.

This is done by computing the second order information at f_0, i.e. $\partial_f^2 - \log p(f|X,Y)$.





Relation to Regularized Risk Functional

THE AUSTRALIAN NATIONAL UNIVERSITY

Recycling of the Likelihood

Match up terms between likelihood and loss function $c(\mathbf{x}, y, f(\mathbf{x}))$. In particular, we recycle these terms:

$$\begin{aligned} c(\mathbf{x}, y, f(\mathbf{x})) &\equiv -\log p(y - f(\mathbf{x})) \\ p(y|f(\mathbf{x}) &\equiv \exp(-c(\mathbf{x}, y, f(\mathbf{x})) \end{aligned}$$

Now all we have to do is take care of the regularizer $m\lambda\Omega[f]$ and $-\log p(f)$.

Regularizer and Prior

The correspondence

$$m\lambda\Omega[f] + c = -\log p(f) \text{ or equivalently } p(f) \propto \exp(-m\lambda\Omega[f])$$

is the link between regularizer $\Omega[f]$ and prior $p(f)$.

Caveat

The translation from regularizer into prior works only to some extent, since the integral over f need not converge.

Hyperparameters

Problem

Sometimes we are not quite sure about the type of prior $p(f)$ we might have, e.g., the variance of some parameters ...

Solution

Put a **prior** on the parameters governing the prior. Instead of $p(f)$ we now have $p(f|\omega)$ and a prior $p(\omega)$ on the **hyperparameter** ω.

Effective Prior: We can obtain the effective prior by integrating out the hyperparameter

$$p(f) = \int p(f|\omega)p(\omega)d\omega$$

Inference

Using the effective prior for $p(f|X,Y)$ (and the assumption $p(f|X) = p(f)$) we obtain $p(f|X,Y) \propto p(Y|f,X)p(f) = p(Y|f,X) \int p(f|X,\omega)p(\omega)d\omega$.

MAP2 Approximation

THE AUSTRALIAN NATIONAL UNIVERSITY

Problem: Nobody wants to compute integrals, because ...

- Computing integrals is expensive
- No closed form possible
- Not very intuitive for inference

Idea

After all, we are only **averaging**, so replace the mean of the distribution by the mode and hope that it will be ok. This leads to the **maximum a posteriori estimate on the hyperparameter**.

Result

$$\underset{f,\omega}{\text{maximize}} \; p(f|X,Y) \propto p(Y|f,X)p(f|\omega)p(\omega)$$

Practical Trick

$$\underset{f,\omega}{\text{minimize}} \; -\log \underbrace{p(Y|f,X)}_{\text{Likelihood}} - \log \underbrace{p(f|\omega)}_{\text{Prior}} - \log \underbrace{p(\omega)}_{\text{Hyperprior}}$$

To integrate or not to integrate

THE AUSTRALIAN NATIONAL UNIVERSITY

Integrate

- This is what you need to do for proper inference
- Fewer Parameters
- $p(f)$ may be of a simpler functional form than $p(f|\omega)p(\omega)$, e.g.,

$$p(a|\omega) = (2\pi\omega^2)^{-\frac{1}{2}} e^{-\frac{a^2}{2\omega^2}} \text{ and } p(\omega) = (2\pi)^{-\frac{1}{2}} e^{-\frac{\omega^2}{2}} \text{ hence } p(a) = \frac{1}{2\pi}\text{BesselK}(0, |a|).$$

To integrate or not to integrate

THE AUSTRALIAN NATIONAL UNIVERSITY

Integrate

- This is what you need to do for proper inference
- Fewer Parameters
- $p(f)$ may be of a simpler functional form than $p(f|\omega)p(\omega)$, e.g.,

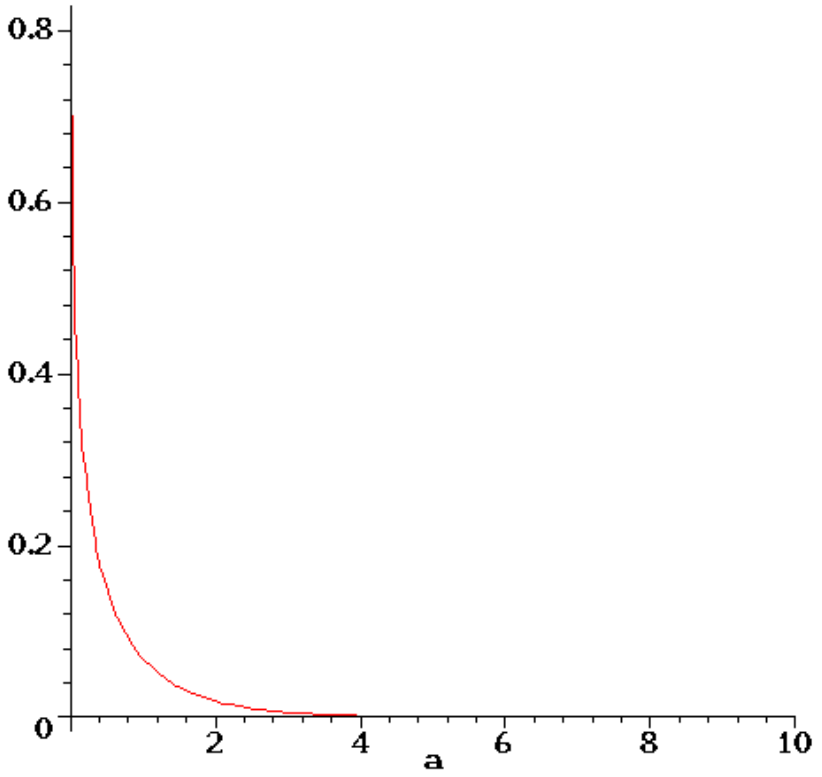
  $p(a|\omega) = (2\pi\omega^2)^{-\frac{1}{2}} e^{-\frac{a^2}{2\omega^2}}$ and $p(\omega) = (2\pi)^{-\frac{1}{2}} e^{-\frac{\omega^2}{2}}$ hence $p(a) = \frac{1}{2\pi}\text{BesselK}(0, |a|)$.

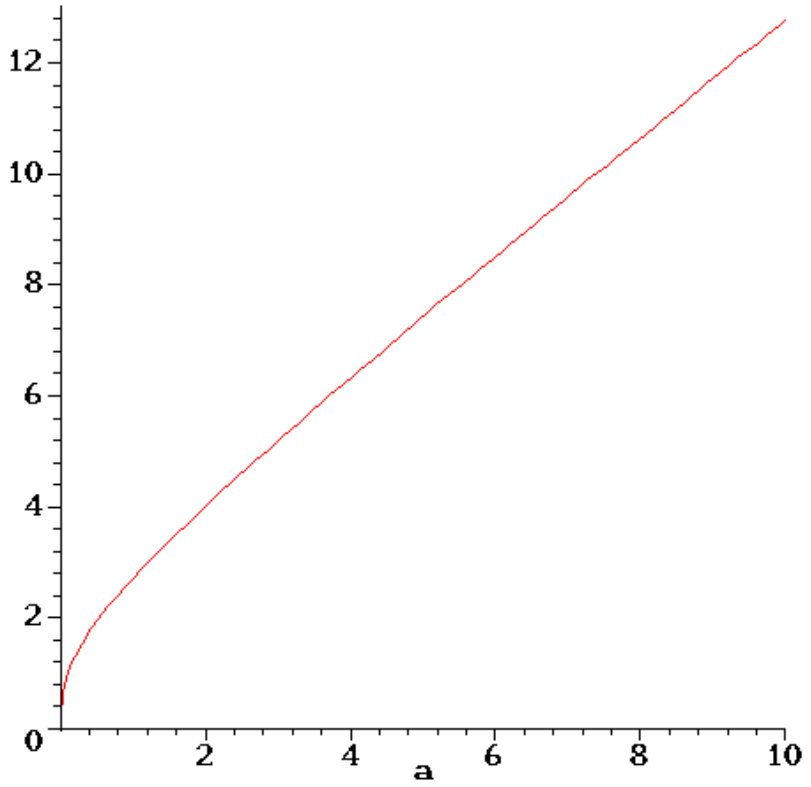
Don't Integrate

- Sometimes easier to optimize (convex optimization problem or simple one-dimensional minimization which can be solved explicitly).
- MAP1 part may become exact (for fixed hyperparameter we have a Gaussian posterior).
- $p(f)$ may be of a simpler functional form than $p(f|\omega)p(\omega)$, e.g., if in the example above $p(\omega) = \frac{1}{2}\exp(-|\omega|)$, then $p(f)$ is really complicated ...

To integrate or not to integrate

THE AUSTRALIAN NATIONAL UNIVERSITY





Overview of Unit 2: Gaussian Processes

01: Gaussian Process
02: Inference with Gaussian Processes
03: Example: Finite Dimensional Regression
04: Linear Model
05: Linear Model: Consequences
06: Parametric Family
07: General Covariance Function
08: Example: Diffusion on a Graph
09: Example: Gaussian RBF Kernel
10: Inference: Posterior Distribution
11: MAP Approximation
12: Confidence Intervals
13: Relation to Regularized Risk Functional
14: Coefficient-based Priors

Gaussian Process

Definition

Denote by $t(x)$ a stochastic process parametrized by $x \in \mathcal{X}$ ($\mathcal{X}$ is an arbitrary index set). Then $t(x)$ is a Gaussian process if for any $m \in \mathbb{N}$ and $\{x_1, \ldots, x_m\} \subset \mathcal{X}$, the random variables $(t(x_1), \ldots, t(x_m))$ are normally distributed.

Covariance Function

We denote by $k(x, x')$ the function generating the covariance matrix

$$K := \operatorname{cov}\{t(x_1), \ldots, t(x_m)\} \text{ where } K_{ij} =: k(x_i, x_j).$$

and by μ the mean of the distribution.

Common Assumption: Set $\mu = 0$.

Density at Observations

We observe t at m locations $x_1, \ldots, x_m$. Then $p(\mathbf{t})$ is given by

$$p(\mathbf{t}) = (2\pi)^{-\frac{m}{2}} |K|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{t} - \mu)^\top K^{-1}(\mathbf{t} - \mu)\right)$$

Inference with Gaussian Processes

THE AUSTRALIAN NATIONAL UNIVERSITY

Goal

After observing $\mathbf{t} := (t(x_1), \ldots, t(x_m))$ we would like to infer the distribution of t at locations $x'_1, \ldots, x'_n$, i.e., we would like to infer about $\mathbf{t}' := (t(x'_1), \ldots, t(x'_n))$.

Conditional Density

We study $p(\mathbf{t}'|\mathbf{t})$. Recall that $p(\mathbf{t}, \mathbf{t}') = p(\mathbf{t}|\mathbf{t}')p(\mathbf{t}')$ and therefore $p(\mathbf{t}|\mathbf{t}')$ can be obtained from $p(\mathbf{t}, \mathbf{t}')$ by **fixing** $\mathbf{t}'$ and **normalizing** by $p(\mathbf{t}') = \int p(\mathbf{t}, \mathbf{t}')d\mathbf{t}$.

Lazy Trick

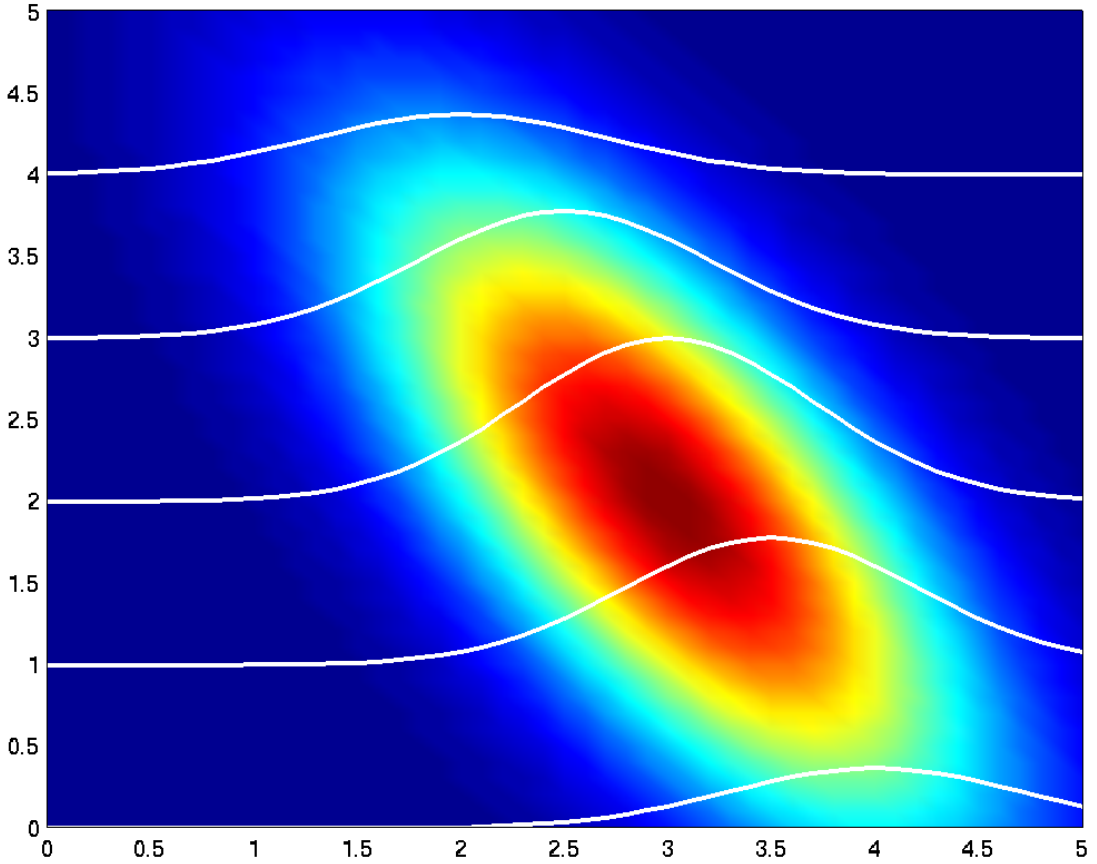
For normal distributions we only need to compute **mean** and **covariance** to determine the density completely (including normalization factors).

Recipe: collect all terms from $p(\mathbf{t}, \mathbf{t}')$ dependent on $\mathbf{t}'$ and ignore the rest.

$$p(\mathbf{t}, \mathbf{t}') \propto \exp\left(-\frac{1}{2}\left(\begin{bmatrix}\mathbf{t}\\ \mathbf{t}'\end{bmatrix} - \begin{bmatrix}\mu\\ \mu'\end{bmatrix}\right)^\top \begin{bmatrix}K_{\mathbf{tt}} & K_{\mathbf{tt}'}\\ K_{\mathbf{t}'\mathbf{t}} & K_{\mathbf{t}'\mathbf{t}'}\end{bmatrix}^{-1} \left(\begin{bmatrix}\mathbf{t}\\ \mathbf{t}'\end{bmatrix} - \begin{bmatrix}\mu\\ \mu'\end{bmatrix}\right)\right)$$

Example: Regression without Noise

THE AUSTRALIAN NATIONAL UNIVERSITY



Example: Regression without Noise

THE AUSTRALIAN NATIONAL UNIVERSITY

Inverting the Covariance Matrix

$$\begin{bmatrix} K_{\mathbf{tt}} & K_{\mathbf{tt}'} \\ K_{\mathbf{tt}'}^\top & K_{\mathbf{t}'\mathbf{t}'} \end{bmatrix}^{-1} = \begin{bmatrix} {K_{\mathbf{tt}}}^{-1} - \left({K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}^\top\right)^\top \chi^{-1} \left({K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}^\top\right) & -\left({K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}^\top\right) \chi^{-1} \\ -\chi^{-1} \left({K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}^\top\right)^\top & \chi^{-1} \end{bmatrix}$$

where $\chi = K_{\mathbf{t}'\mathbf{t}'} - K_{\mathbf{tt}'}^\top {K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}$ (Schur complement).

Reduced Covariance

From the inverse of the covariance matrix we obtain that the only quadratic part in $\mathbf{t}'$ is given by χ. Thus the **variance in $\mathbf{t}'$ is y reduced** from $K_{\mathbf{t}'\mathbf{t}'}$ to $K_{\mathbf{t}'\mathbf{t}'} - K_{\mathbf{tt}'}^\top {K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}$ by observing $\mathbf{t}$.

Predictive Mean

Instead of μ' the mean is shifted to $\mu' + K_{\mathbf{tt}'}^\top K_{\mathbf{tt}}^{-1}(\mathbf{t} - \mu)$.

Linear Model

THE AUSTRALIAN NATIONAL UNIVERSITY

Covariance Function

Assume that $\mathrm{Cov}(t(x), t(x')) = \langle x, x' \rangle$ with $x \in \mathbb{R}^n$, i.e., that we have an n-dimensional Normal distribution, where the covariance between observations is a bilinear function of x and x'.

Density

$$p(\mathbf{t}) = (2\pi)^{-\frac{n}{2}} \left(\det X^\top X\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{t}-\mu)(XX^\top)^*(\mathbf{t}-\mu)\right)$$

where $X = (\mathbf{x}_1, \ldots, \mathbf{x}_m)$ and $(XX^\top)^*$ is the pseudoinverse of $XX^\top$.

Parameter Transformation

By letting $\mathbf{t} = X\alpha + \mu$ (this is admissible since $p(\mathbf{t})$ only defined a density on an n-dimensional subspace) we see that this is equivalent to

$$p(\alpha) = (2\pi)^{-\frac{n}{2}} \exp\left(-\frac{1}{2}\|\alpha\|^2\right) \text{ where } \mathbf{t} = X\alpha + \mu.$$

see e.g., Box and Tiao, 1973.

Linear Model, Part II

THE AUSTRALIAN NATIONAL UNIVERSITY

Prediction

Since $\mathbf{t} = X\alpha + \mu$, already after observing $m = n$ instances $\{x_1, \ldots, x_n\} \subset \mathbb{R}^n$ we can determine α completely.

Reason: $X\alpha$ spans only an n-dimensional subspace.

Advantage

We only need n observations.

Problem 1

The model breaks if $\mathbf{t} \neq X\alpha + \mu$ for no $\alpha \in \mathbb{R}^n$. We need to modify our statistical model.

Problem 2

We may have an overly simple model, so we cannot learn beyond a certain point.

Parametric Family

THE AUSTRALIAN NATIONAL UNIVERSITY

Extension

Instead of $k(x, x') = \sum_{i=1}^{m} x_i x_i'$ we assume the covariance function

$$k(x, x') = \sum_{i=1}^{N} \phi_i(x)\phi_i(x').$$

where $\phi_i(x)$ are the features.

Reparametrization

As in the linear case reparametrize $\mathbf{t} = \Phi\alpha$, where $\Phi_{ij} = \phi_i(x_j)$. Therefore we have **two equivalent parametrizations** of the prior on $\mathbf{t}$ (assuming $m \geq N$):

$$p(\alpha) = (2\pi)^{-\frac{N}{2}} \exp\left(-\frac{1}{2}\|\alpha\|^2\right) \text{ and } \mathbf{t} = \Phi\alpha + \mu.$$

$$p(\mathbf{t}) = (2\pi)^{-\frac{N}{2}} \left(\det\Phi^\top\Phi\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{t}-\mu)^\top(\Phi\Phi^\top)^*(\mathbf{t}-\mu)\right).$$

See e.g., Fahrmeir and Tutz, 1994.

General Covariance Function

THE AUSTRALIAN NATIONAL UNIVERSITY

Idea

In general, we may not know how many dimensions the function space, or, in other words, the space of observations really has, hence use generic kernel k without further assumptions on the dimensionality of the set of functions $k(x_i, \cdot)$.

Examples

$$
\begin{aligned}
k(x, x') &= \exp\left(-\frac{1}{2\sigma\|x - x'\|}\right) \text{ Laplacian Kernel} \\
k(x, x') &= \exp\left(-\frac{1}{2\sigma^2\|x - x'\|^2}\right) \text{ Gaussian RBF Kernel} \\
k(x, x') &= \left(\langle x, x' \rangle + c\right)^d \text{ with } c \geq 0,\ d \in \mathbb{N} \text{ Polynomial Kernel} \\
k(x, x') &= B_{2n+1}(x - x') \text{ Spline kernel} \\
k(x, x') &= \mathbf{E}_c[p(x|c)p(x'|c)] \text{ Conditional Expectation Kernel}
\end{aligned}
$$

All these kernels correspond to a Gaussian process ... (see Williams 1998, Schölkopf and Smola 2002, Wahba 1990,

Diffusion Process

THE AUSTRALIAN NATIONAL UNIVERSITY

Basic Idea

We have an initial density $p(x, 0)$ of particles, heat, etc., which becomes more spread out over time due a diffusion process. Goal: estimate $p(x, t)$, based on $p(x, 0)$.

Diffusion in $\mathbb{R}$

The change in density is proportional to the second derivative of $p(x, t)$

$$\partial_t p(x, t) = \sigma \partial_x^2 p(x, t)$$

We want to find solutions of the homogeneous PDE.

Extension

More generally we assume a differential equation $\partial_t p(x, t) = Dp(x, t)$ where D is a differential operator whose characteristic polynomial of D satisfies $D(\xi) = D(-\xi)$.

Example

Standard diffusion process: $Dp(x, t) = \sigma \Delta p(x, t)$ and correspondingly $D(\xi) = \xi^2$
Likewise $D = 1 + \partial_x^2 + c\partial_x^4$ and $D(\xi) = 1 + \xi^2 + c\xi^4$.

Diffusion Process, part II

THE AUSTRALIAN NATIONAL UNIVERSITY

Symbolic Solution

We may write $p(x,t) = \exp(Dt)p(x,0)$, which leads to

$$\partial_t p(x,t) = \partial_t \exp(Dt)p(x,0) = D\exp(Dt)p(x,0) = Dp(x,t)$$

Explicit Solution We use the Fourier representation of D and p to obtain

$$\partial_t \mathcal{F}[p](\omega,t) = D(i\omega)\mathcal{F}[p](\omega,t)$$

The homogeneous solution $p(x,t)$ is therefore given by

$$p(x,t) = \left(\mathcal{F}^{-1}[\exp(tD(i\omega))]\right) \circ p(x,0)$$

Example: Diffusion in $\mathbb{R}$

We have $D = \partial_x^2$ and consequently $D(i\omega) = -\omega^2$. This leads to

$$\left(\mathcal{F}^{-1}[\exp(tD(i\omega))]\right) = \left(\mathcal{F}^{-1}[\exp(-t\omega^2)]\right) = \frac{1}{\sqrt{4\pi t}}\exp\left(-\frac{x^2}{4t}\right)$$

See e.g. Kondor 2002, Haken, 1976

Diffusion Process, part III

THE AUSTRALIAN NATIONAL UNIVERSITY

Joint Covariance Function

The function $G_t(x) := \left(\mathcal{F}^{-1}[\exp(tD(i\omega))]\right)(x)$ gives the density of observing a particle at location x, if we started with all the probability mass located at $x = 0$ at time $t = 0$. Hence, the joint probability of observing particles at x, x' is given by

$$p(x, x'|t, x_{\text{start}} = 0) = G_t(x)G_t(x')$$

Uniform Initialization: assuming that at time $t = 0$ the density is uniform, we have

$$\begin{aligned} p(x, x') &= \int G_t(x - \tau)G_t(x' - \tau)d\tau \\ &= (G_t \circ G_t)(x - x')(\text{ Symmetry in } G_t) \\ &= \left(\mathcal{F}^{-1}[\exp(2tD(i\omega))]\right)(x - x') = G_{2t}(x - x')(\text{Fourier-Plancherel}). \end{aligned}$$

Simplifying Conclusion

The logarithm of the Fourier transform of a translation invariant kernel corresponds to the differential operator of the generating diffusion process.

Example: Diffusion on a Graph

THE AUSTRALIAN NATIONAL UNIVERSITY

Connectivity Matrix

Assume an undirected graph with m nodes, then we can represent it by a matrix $C \in \mathbb{R}^{m \times m}$ where $C_{ij} = 1$ if i, j are connected and $C_{ij} = 0$ otherwise.

Next denote by $L := G - \text{diag}(\mathbf{l})$ where $l_i := \sum_j G_{ij}$ the Laplacian of the graph G.

Random Walk on a Graph

Assume that we have a probability distribution on a graph, given by $p \in \mathbb{R}^m$, where $\|p\|_1 = 1$. During time Δt a fraction of $\sigma \cdot \Delta t$ will move from node i to each of the adjacent connected nodes j. This implies that

$$p_i \leftarrow p_i - \sigma \Delta t p_i \sum_j C_{ji} + \sigma \Delta t \sum_j C_{ij} p_j = p_i + \sigma \Delta t [Lp]_i$$

Limiting Case (Kondor, 2002)

After n steps the density p becomes $(1 + \sigma \Delta t L)^n p$. If we now set $\Delta t = \frac{t}{n}$ and let $n \to \infty$, we obtain

$$p = \lim_{n \to \infty} \left(1 + \frac{\sigma t}{n} L\right)^n = \exp(t \sigma L).$$

Inference: Posterior Distribution

THE AUSTRALIAN NATIONAL UNIVERSITY

Recall: Bayes Rule

Given X we want to infer $p(f|X,Y)$. With the usual assumptions (iid data, prior independent of X) this leads to

$$p(f|X,Y) \propto p(Y|f,X)p(f) = \prod_{i=1}^{m} p(y_i|f(x_i),x_i)p(f)$$

GP Assumption

The function values $f(x_i)$ are distributed according to a Gaussian process. The connection to the observations y_i is take care by the noise model $p(y_i|f(x_i),x_i)$. This leads to the following log-posterior

$$-\log p(f|X,Y) = \sum_{i=1}^{m} -\log p(y_i|x_i,f(x_i)) + \frac{1}{2}\log\det K + \frac{1}{2}\mathbf{f}^\top K^{-1}\mathbf{f} + c$$

Inference

We carry out inference by computing e.g., $y = \mathbf{E}_{p(f|X,Y)}[f(x)]$ or $y = \mathbf{E}_{p(f|X,Y)}[f^2(x)]$.

MAP Approximation

THE AUSTRALIAN NATIONAL UNIVERSITY

Problem

Computing integrals is expensive, in particular in high-dimensional spaces.

MAP Solution

Approximate $\mathbf{E}_{p(f|X,Y)}[f] \approx \operatorname{argmax}_{\mathbf{f}} p(f|X,Y)$. In the present case this means that we solve

$$\operatorname*{argmin}_{\mathbf{f}} \sum_{i=1}^{m} -\log p(y_i|x_i, f(x_i)) + \frac{1}{2}\mathbf{f}^\top K^{-1}\mathbf{b} + c$$

Reparametrization

Set $y = K\alpha$. This leads to the optimization problem

$$\operatorname*{argmin}_{\alpha} \sum_{i=1}^{m} -\log p([K\alpha]_i|x_i, f(x_i)) + \frac{1}{2}\alpha^\top K\alpha + c$$

Prediction

Once we obtained α for X, Y, we may predict $f(x')$ as $\sum_{i=1}^{m} k(x_i, x')\alpha_i$.

This assumes that $\alpha' = 0$ is a good estimate.

Latent Variables

THE AUSTRALIAN NATIONAL UNIVERSITY

Problem

$\alpha' = 0$ is often not such a good estimate.

This is especially the case if $-\log p(y|x, f(x))$ does not have a minimum (e.g., loss for classification).

Better Solution

Find f such that the expected log-posterior (with the expectations taken over $y'_1, \ldots y'_{m'}$, and adjusted by themselves to minimize the log-posterior) is minimized.

$$\operatorname*{argmin}_{\mathbf{f}, p(\mathbf{y}')} \sum_{i=1}^{m} -\log p(y_i|x_i, f(x_i)) - \mathbf{E}_{y'_1,\ldots,y'_{m'}} \sum_{i=1}^{m'} \log p(y'_i|x'_i, f(x'_i)) + \frac{1}{2}\mathbf{f}^\top K^{-1}\mathbf{f} + c$$

where K is the covariance matrix over X, X' and likewise $\mathbf{f} \in \mathbb{R}^{m+m'}$.

Algorithm (EM, compare to SVM Transduction)

1) For fixed $p(\mathbf{y}')$ find optimal $\mathbf{f}$ (Maximization).

2) For fixed $\mathbf{f}$, find optimal $p(\mathbf{y}')$ (Expectation).

Confidence Intervals

THE AUSTRALIAN NATIONAL UNIVERSITY

Normal Distribution

If the predictive distribution is a normal distribution, we only need to compute the variance of $y'_1, \ldots y'_{m'}$ to obtain error bars on the prediction (see the reasoning before). Moreover, the MAP approximation is exact.

y'_i have Finite Cardinality

For instance, if we want to predict class labels, we can simply evaluate $p(y = 1|f, x)$ and $p(y = -1|f, x)$ to obtain information about the confidence of the estimate.

General Case: Approximations

Often $p(y|f, x)$ will be none of the above, and, in particular, we will not be able to compute the integrals explicitly, so we have to approximate:

- Quadratic approximation: compute Taylor expansion of $p(f|X, Y)$ at f_{MAP} and use the latter to approximate $p(f|X, Y)$ by a normal distribution.
- Monte Carlo method: sample from $p(f|X, Y)$ (not topic of the lectures here).

Coefficient-based Priors

THE AUSTRALIAN NATIONAL UNIVERSITY

Factorizing Priors

Analogously to a factorizing assumption on the observations we may also assume

$$p(f) = \prod_{i=1}^{m} p(\alpha_i) \text{ where } f = \sum_{i=1}^{m} \alpha_i f_i$$

Motivation

The basis functions f_i correspond to independent "factors" causing the observations, e.g., neurons firing independently but rarely, image elements occurring, etc.

Example: Laplace Prior

Sparse codes are often represented by $p(\alpha_i) = \frac{1}{2}\exp(-|\alpha_i|)$. Often one uses a distribution which is even more peaked at 0 to obtain a posterior with higher sparsity (e.g., the adjoint Bessel function from before).

Example: Normal Prior

Priors such as $p(\alpha_i) = (2\pi)^{-\frac{1}{2}}\exp(-\frac{1}{2}\alpha_i^2)$ lead to Gaussian processes.

Overview of Unit 3: GP Regression

THE AUSTRALIAN NATIONAL UNIVERSITY

01: Noiselsess Case: Joint Covariance
02: Conditioning on Observations
03: Adding two Normal Distributions
04: Regression with Normal Noise
05: Examples
06: Error Bars for GP Regression
07: Examples
08: Hyperparameters
09: Automatic Relevance Determination
10: Generic Noise Models
11: More Noise Models
12: Even More Noise Models
13: Transformation into Lagrange Multipliers
14: MAP Approximation
15: Connection to Support Vectors
16: Scaling Problems

Noiselsess Case: Joint Covariance

THE AUSTRALIAN NATIONAL UNIVERSITY

Recall: Assumptions

Observations $\mathbf{t}$ are samples from a Gaussian process with mean μ and covariance matrix K.

Recall: Goal

After observing $\mathbf{t} := (t(x_1), \ldots, t(x_m))$ we would like to infer the distribution of t at locations $x'_1, \ldots, x'_n$, i.e., we would like to infer about $\mathbf{t}' := (t(x'_1), \ldots, t(x'_n))$.

Lazy Trick

The solution is to study $p(\mathbf{t}'|\mathbf{t})$. For normal distributions we only need to compute **mean** and **covariance** to determine the density completely (including normalization factors). We have

$$p(\mathbf{t}, \mathbf{t}') \propto \exp\left(-\frac{1}{2}\left(\begin{bmatrix}\mathbf{t}\\ \mathbf{t}'\end{bmatrix} - \begin{bmatrix}\mu\\ \mu'\end{bmatrix}\right)^\top \begin{bmatrix}K_{\mathbf{tt}} & K_{\mathbf{tt}'}\\ K_{\mathbf{t}'\mathbf{t}} & K_{\mathbf{t}'\mathbf{t}'}\end{bmatrix}^{-1} \left(\begin{bmatrix}\mathbf{t}\\ \mathbf{t}'\end{bmatrix} - \begin{bmatrix}\mu\\ \mu'\end{bmatrix}\right)\right)$$

Recall: Inverting the Covariance Matrix

THE AUSTRALIAN NATIONAL UNIVERSITY

Inverting the Covariance Matrix

$$\begin{bmatrix} K_{\mathbf{tt}} & K_{\mathbf{tt}'} \\ K_{\mathbf{tt}'}^\top & K_{\mathbf{t}'\mathbf{t}'} \end{bmatrix}^{-1} = \begin{bmatrix} {K_{\mathbf{tt}}}^{-1} - \left({K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}^\top\right)^\top \chi^{-1} \left({K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}^\top\right) & -\left({K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}^\top\right) \chi^{-1} \\ -\chi^{-1} \left({K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}^\top\right)^\top & \chi^{-1} \end{bmatrix}$$

where $\chi = K_{\mathbf{t}'\mathbf{t}'} - K_{\mathbf{tt}'}^\top {K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}$ (Schur complement).

Reduced Covariance

From the inverse of the covariance matrix we obtain that the only quadratic part in $\mathbf{t}'$ is given by χ. Thus the **variance in $\mathbf{t}'$ is y reduced** from $K_{\mathbf{t}'\mathbf{t}'}$ to $K_{\mathbf{t}'\mathbf{t}'} - K_{\mathbf{tt}'}^\top {K_{\mathbf{tt}}}^{-1} K_{\mathbf{tt}'}$ by observing $\mathbf{t}$.

Predictive Mean

Instead of μ' the mean is shifted to $\mu' + K_{\mathbf{tt}'}^\top K_{\mathbf{tt}}^{-1}(\mathbf{t} - \mu)$.

Adding two Normal Distributions

THE AUSTRALIAN NATIONAL UNIVERSITY

Goal

Regression with Gaussian Processes with additive normal noise: here we need to compute the distribution obtained from the sum of two normal distributions.

Theorem (for simplicity only in $\mathbb{R}$)

Denote by ξ, ξ' random variables with $\xi \sim \mathcal{N}(\mu, \sigma^2)$ and $\xi' \sim \mathcal{N}(\mu', \sigma'^2)$. Then $\xi + \xi' \sim \mathcal{N}(\mu + \mu', \sigma^2 + \sigma'^2)$.

Proof

The density arising from the sum of two random variables is given by the convolution of the densities, i.e. $p(\xi + \xi') = (p \circ p')(\xi + \xi')$. The means are clearly given by $\mu + \mu'$. For the rest assume zero mean:

$$p \circ p' = \mathcal{F}^{-1}[\mathcal{F}[p] \cdot \mathcal{F}[p']] \propto \mathcal{F}^{-1}\left[e^{-\frac{\sigma^2}{2}\omega^2} e^{-\frac{\sigma'^2}{2}\omega^2}\right] = \mathcal{F}^{-1}\left[e^{-\frac{\sigma^2+\sigma'^2}{2}\omega^2}\right]$$

Here we see that the covariances add up, hence we obtain $\mathcal{N}(\mu + \mu', \sigma^2 + \sigma'^2)$. The general case can be reduced to $\mathbb{R}$ by simultaneous diagonalization.

Regression with Normal Noise

THE AUSTRALIAN NATIONAL UNIVERSITY

Idea

If we have $y_i = t_i + \xi_i$ where $\mathbf{t} \sim \mathcal{N}(0, K)$ and $\xi_i \sim \mathcal{N}(0, \sigma^2)$, we know that $\mathbf{y}$, being the sum of two normal random variables, satisfies $\mathbf{y} \sim \mathcal{N}(0, K + \sigma^2 \mathbf{1})$.

Posterior Density

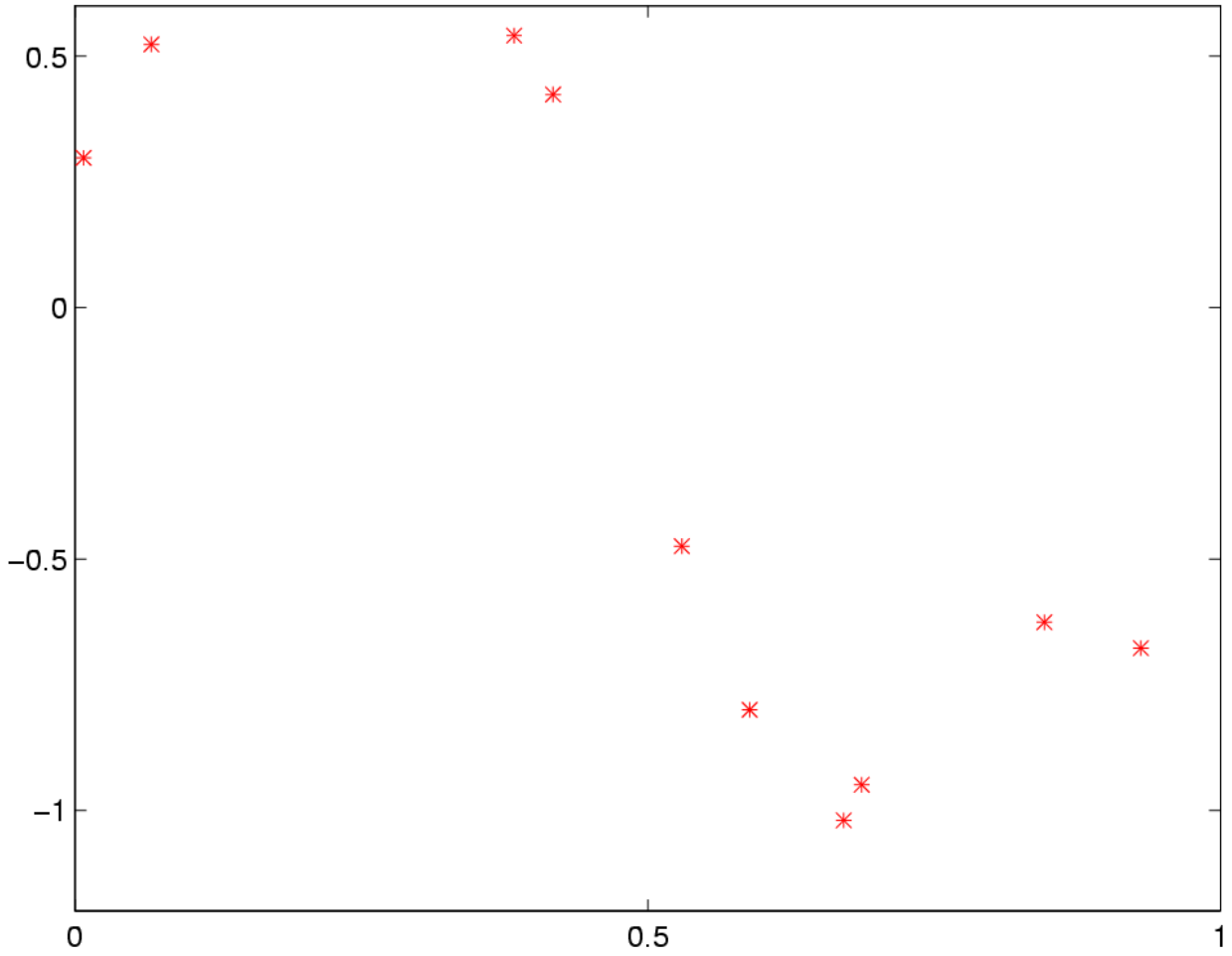
$$p(\mathbf{y}|X) = (2\pi)^{-\frac{n}{2}} (\det(K + \sigma^2 \mathbf{1}))^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\mathbf{y}^\top (K + \sigma^2)^{-1}\mathbf{y}\right)$$

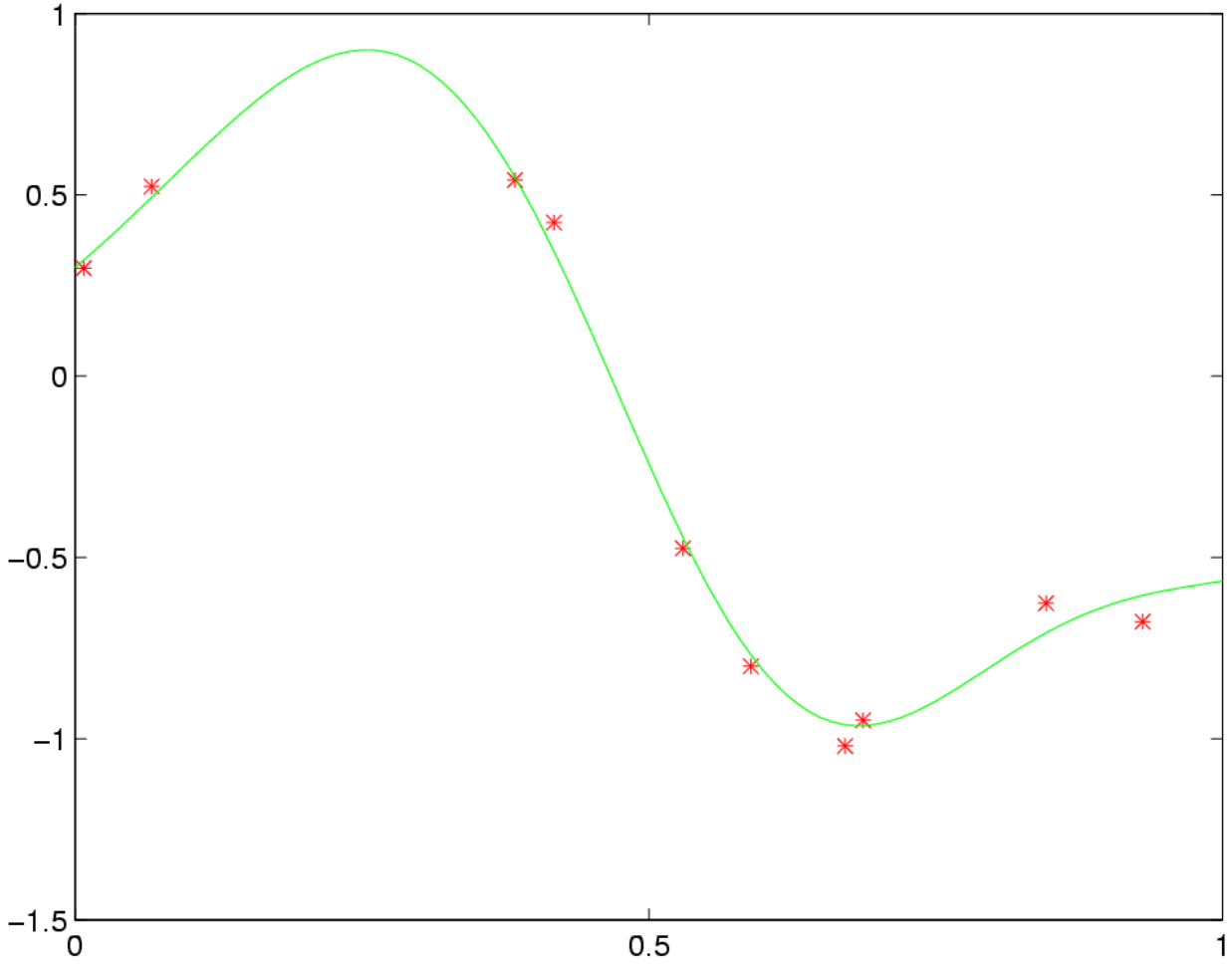
Note that the problem of non-invertibility of the covariance matrix disappeared (similar to regularization to improve the condition of a matrix).

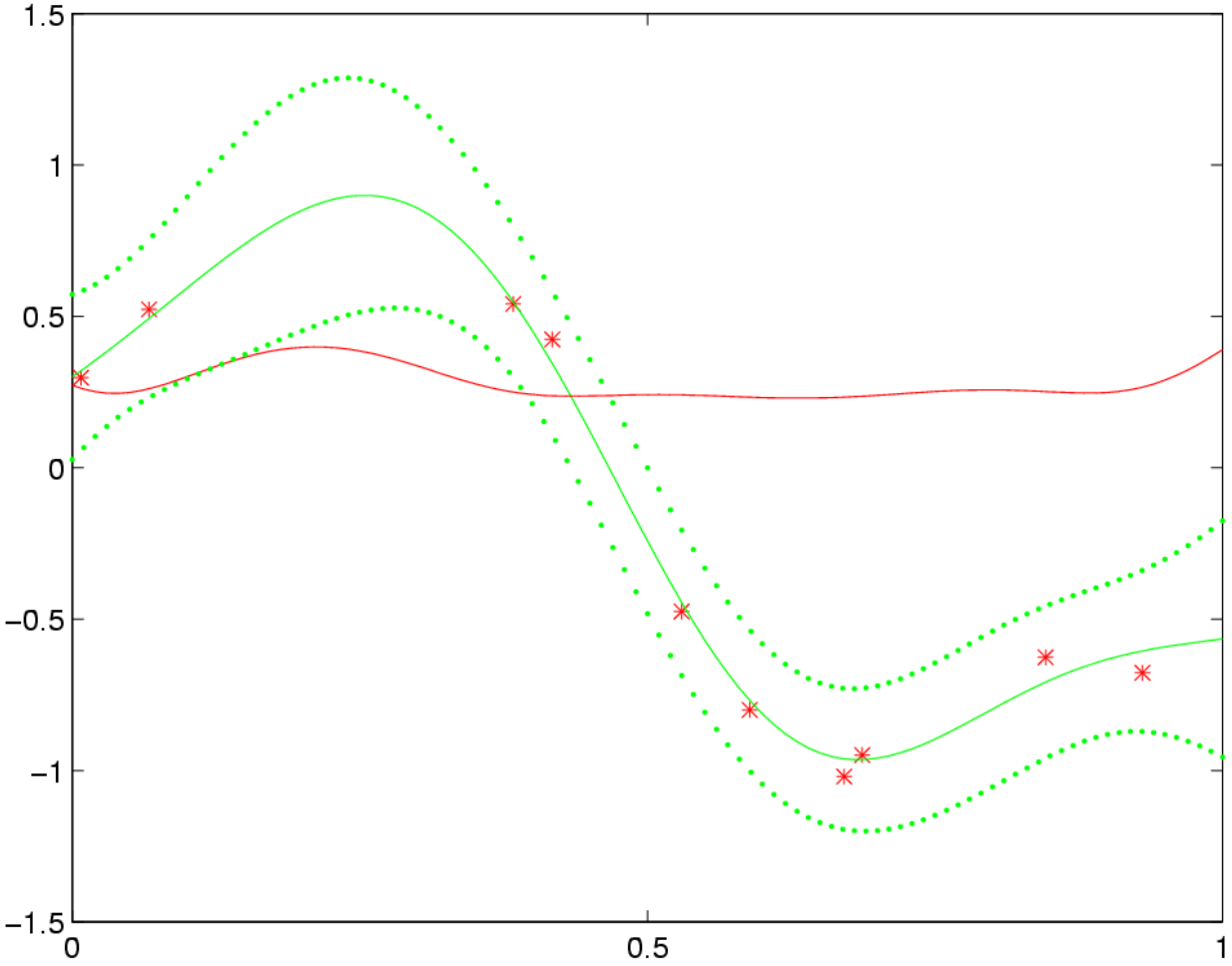
Inference

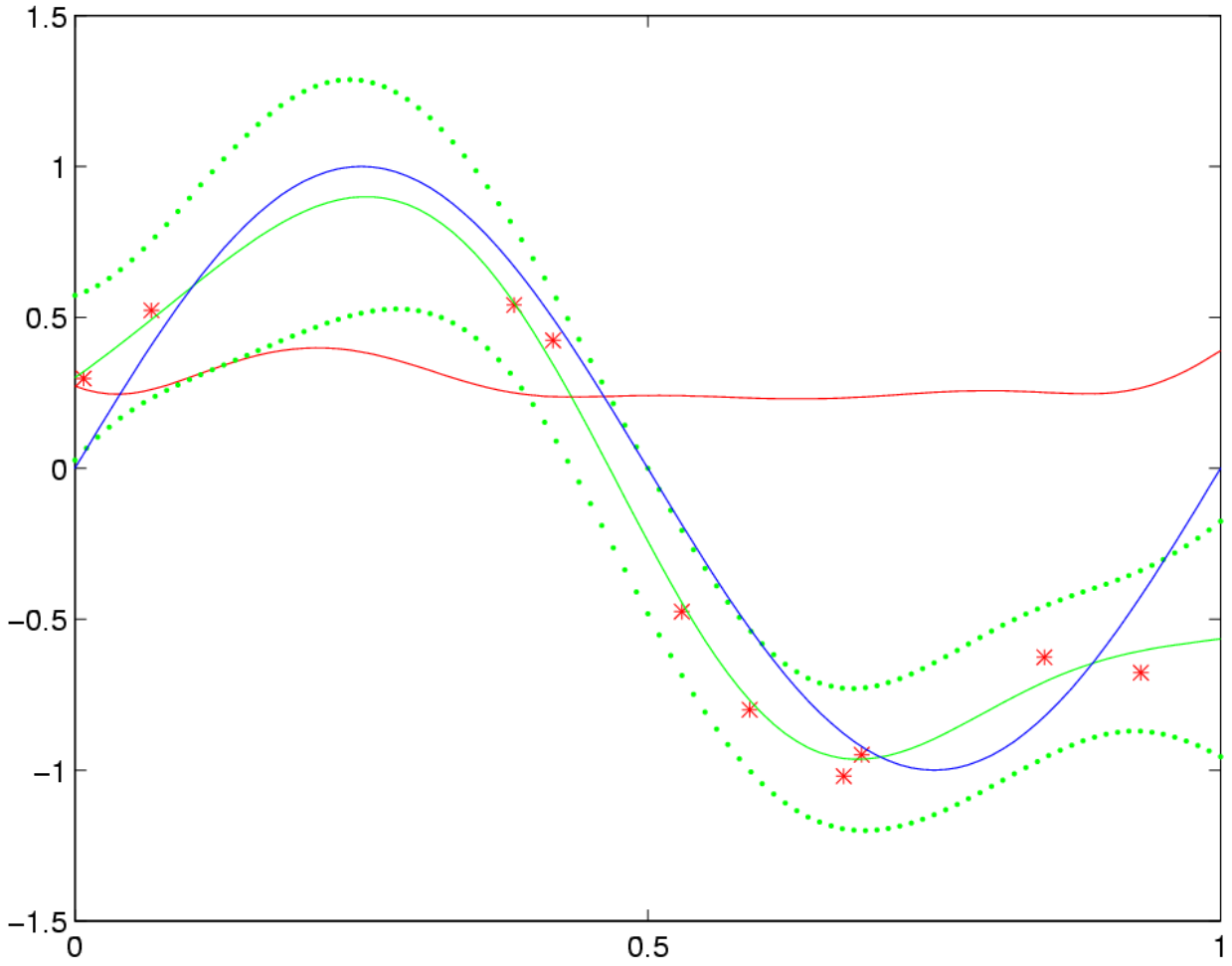
We can simply re-use the results from inference without noise and obtain (for inferring $\mathbf{y}'$ after observing $\mathbf{y}, X, X'$): $\mathbf{y}' \sim \mathcal{N}(\mu_y, \Sigma_y)$where

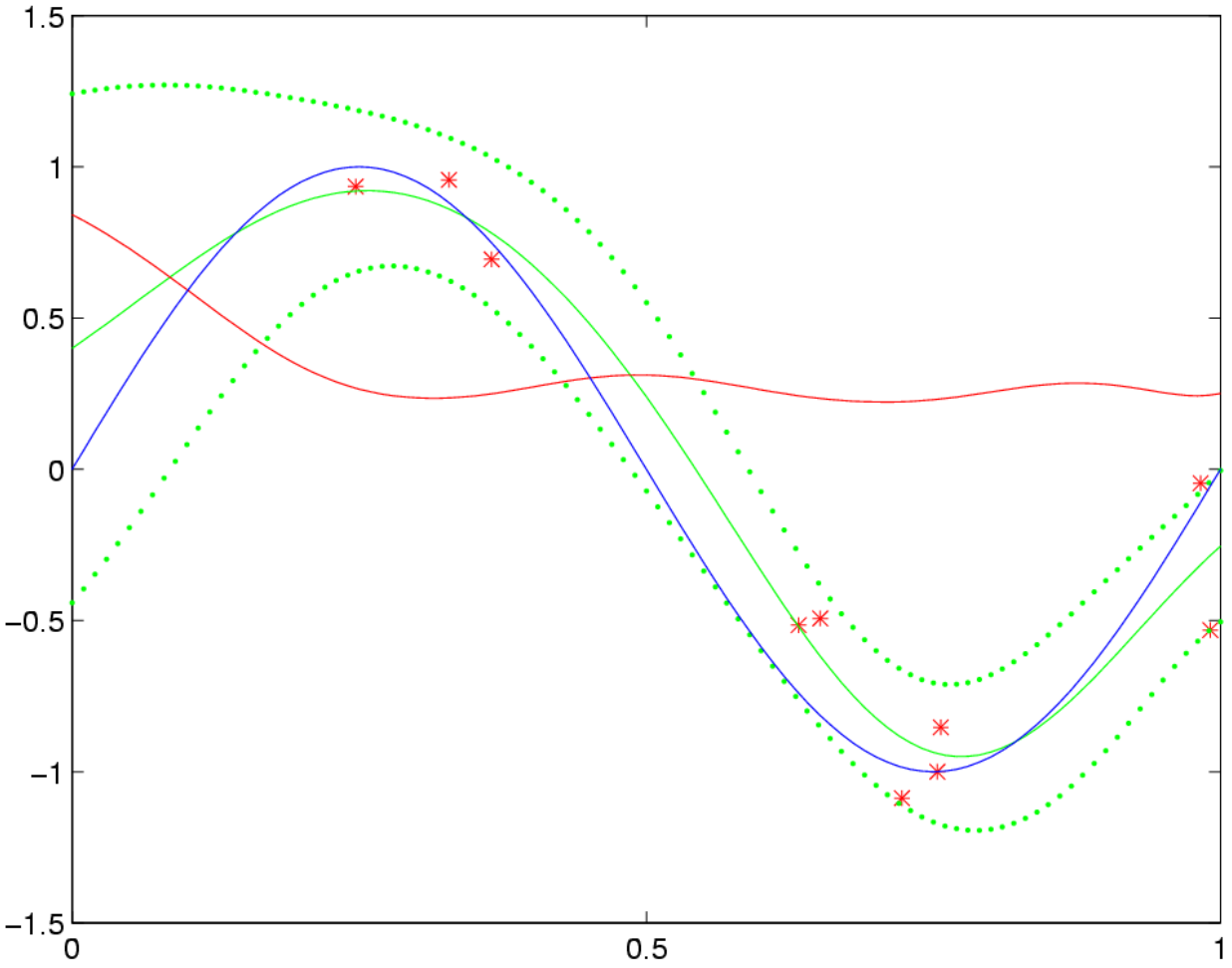
$$\mu_y = K_{\mathbf{tt}'}^\top (K_{\mathbf{tt}} + \sigma^2 \mathbf{1})^{-1}\mathbf{y} \text{ and } \Sigma_y = K_{\mathbf{t}'\mathbf{t}'} + \sigma^2 \mathbf{1} - K_{\mathbf{tt}'}^\top (K_{\mathbf{tt}} + \sigma^2 \mathbf{1})^{-1} K_{\mathbf{tt}'}$$


0.5
0
−0.5
−1
0
0.5
1


1
0.5
0
−0.5
−1
−1.5
0
0.5
1


1.5
1
0.5
0
−0.5
−1
−1.5
0
0.5
1


1.5
1
0.5
0
−0.5
−1
−1.5
0
0.5
1


1.5
1
0.5
0
−0.5
−1
−1.5
0
0.5
1

Hyperparameters

Problem

We do not know the exact values of σ, the correlation width ω of the kernel (for Gaussian RBF), etc., so we have to avoid making too specific guesses.

Solution

Treat σ, ω as hyperparameters and put a prior on the distribution of them. For simplicity, we only study σ:

$$p(f|X,Y) = \int p(f|X,Y,\sigma)p(\sigma)d\sigma$$

MAP2 approximation leads to $\operatorname{argmax}_{f,\sigma} p(Y|f,X,\sigma)p(f)p(\sigma)$.

Regression with Normal Noise

We can take advantage of the fact that $\mathbf{y}$ is taken from a normal distribution. So the problem of finding an appropriate value of σ reduces to

$$\operatorname*{argmax}_{\sigma} \frac{1}{2}\log\det(K+\sigma^2\mathbf{1}) + f^\top(K+\sigma^2\mathbf{1})^{-1}f$$

Matrix Magic

Derivatives of the Inverse

We need to compute $\partial_{\sigma^2} f^\top (K + \sigma^2 \mathbf{1})^{-1} f$.

$$0 = \partial_t (A^{-1} A) = \partial_t A^{-1} A + A^{-1} \partial_t A \text{ hence } \partial_t A^{-1} = A^{-1} (\partial_t A) A^{-1}$$

This leads to

$$\partial_{\sigma^2} f^\top (K + \sigma^2 \mathbf{1})^{-1} f = \| (K + \sigma^{-2} \mathbf{1})^{-1} \|^2$$

Derivatives of the Log-Determinant

To compute $\partial_{\sigma^2} \log \det (K + \sigma^2 \mathbf{1})$ note that $\frac{d}{dA} \log \det A = A^*$. The latter can be seen as follows:

$$\partial_{A_{ij}} \log \det A = \frac{1}{\det A} \partial_{A_{ij}} \det A = \frac{1}{\det A} \partial_{A_{ij}} \det \bar{A}_{ij}$$

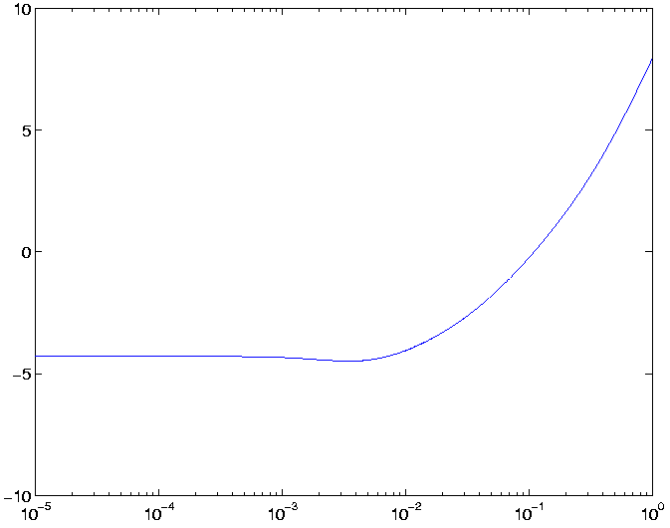
where $\bar{A}$ is the matrix of cofactors of A. This yields

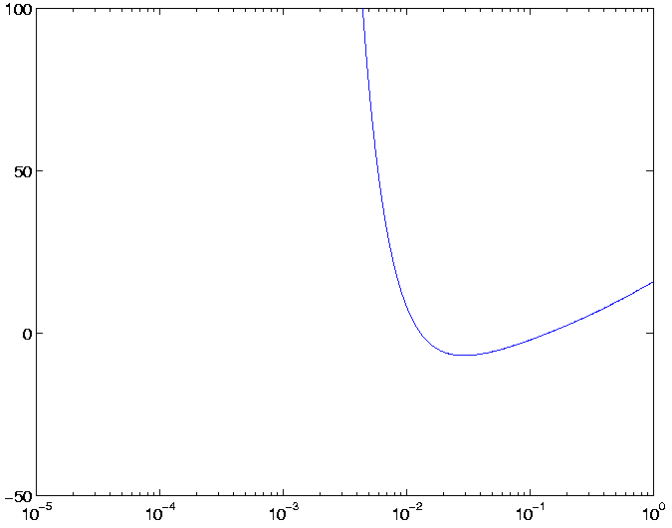
$$\partial_{\sigma^2} \log \det (K + \sigma^2 \mathbf{1}) = \text{tr } \left((K + \sigma^2 \mathbf{1})^{-1} \partial_{\sigma^2} (K + \sigma^2 \mathbf{1}) \right) = \text{tr } (K + \sigma^2 \mathbf{1})^{-1}.$$

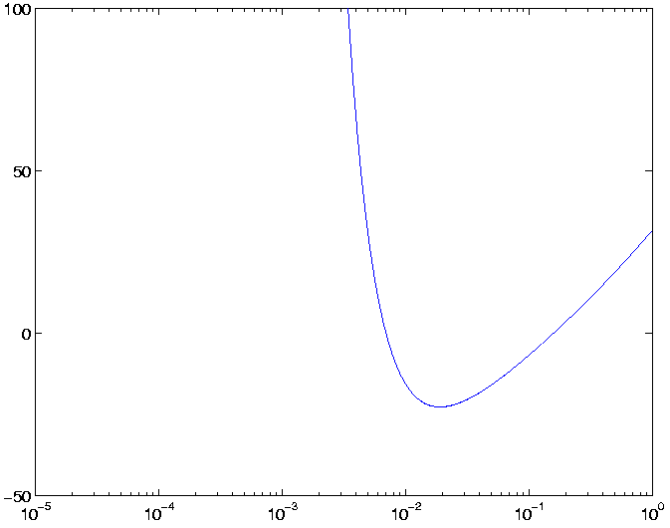
This allows us to compute the gradient wrt. σ^2 and optimize.

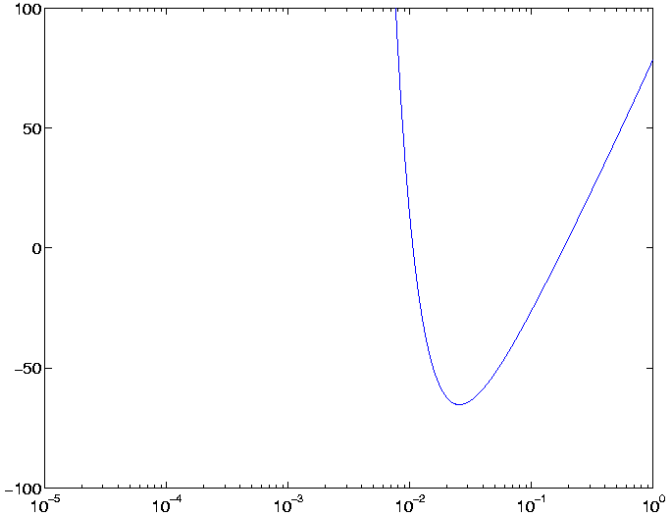
Example: Adjusting σ $(m = 5, 10, 20, 50)$

THE AUSTRALIAN NATIONAL UNIVERSITY









Automatic Relevance Determination

Problem

Which is the proper scale of the data (some inputs more important than others)? Which inputs are relevant?

Scaling of Data

Rescale inputs $\mathbf{x}$ by scaling matrix Ω, i.e. $x \rightarrow \Omega x$ (typically we use a diagonal matrix, as it has fewer parameters). Assume hyperprior on Ω and repeat MAP2 procedure. This leads to

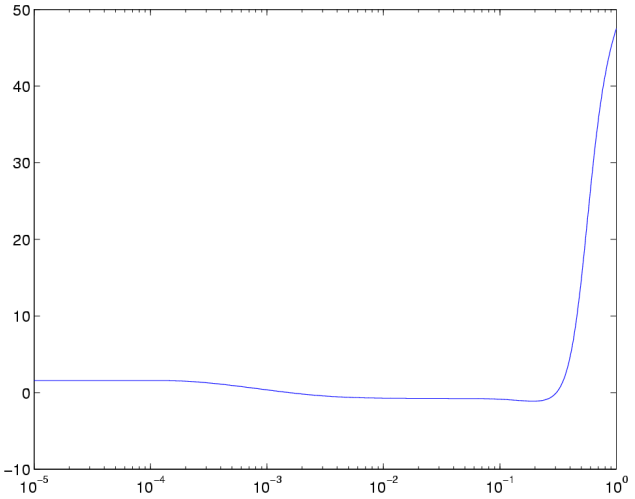
$$p(f|X,Y) \propto \int p(Y|\Omega X, f)p(f)p(\Omega)d\Omega$$

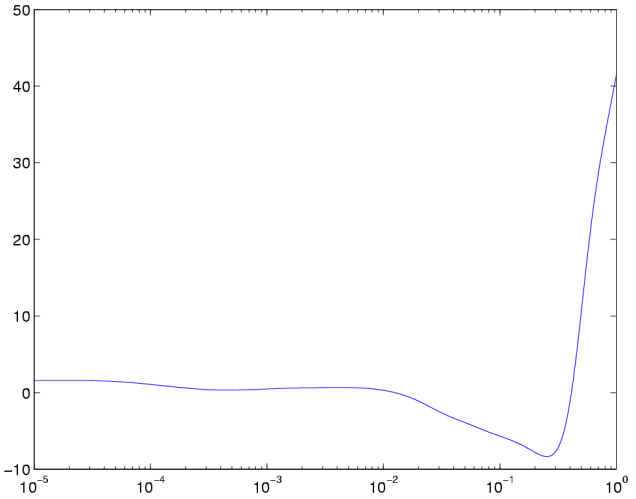
Improper Prior

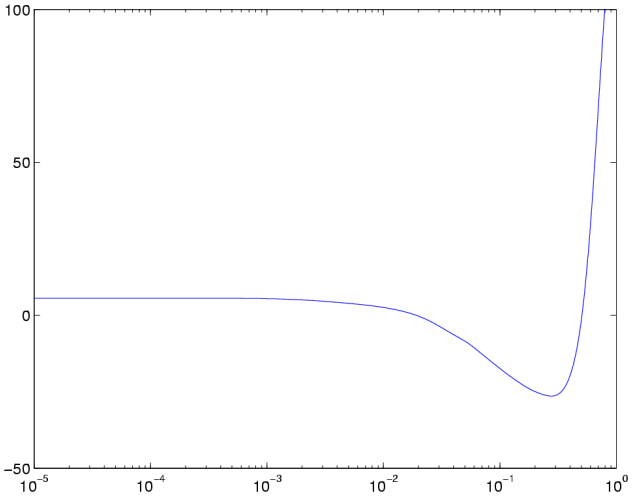
Often it is convenient to use a function $p(\omega)$ (not only for ARD, though), that does not correspond to a finite measure, often called an improper prior (since there $\int p(f|\sigma)p(\sigma)d\sigma$ is not defined). **Note: the MAP2 procedure works regardless.**

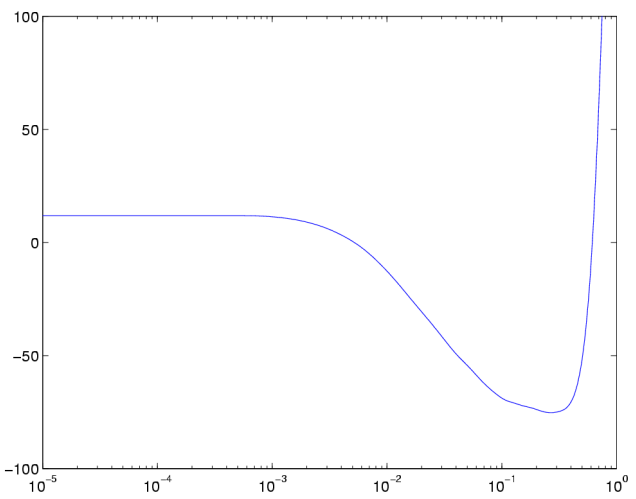
Example: Adjusting ω $(m = 5, 10, 20, 50)$

THE AUSTRALIAN NATIONAL UNIVERSITY









Additive Noise Models

THE AUSTRALIAN NATIONAL UNIVERSITY

Additive Noise: Often, we have an underlying effect, say $f(x)$, which is corrupted by additive noise ξ such that we observe $y = f(x) + \xi$.

Simplifying Assumptions

Typically we assume that the random variables ξ are **uncorrelated and have zero mean**, i.e. $\mathbf{E}\xi = 0$ and $\mathbf{E}\xi\xi' = 0$ for all ξ, ξ'.

Furthermore we typically assume that ξ **is independent of** x (no heteroscedasticity). This means that there exists one density $p(\xi)$ governing the whole noise process. Under the iid assumption the posterior can now be written as

$$p(f|X, Y) \propto p(f) \prod_{i=1}^{m} p(y_i - f(x_i))$$

Note

There are many cases where the noise depends on the size of f itself, such as measurements which provide only relative precision. We are treating only a **very special** case (which works very well in practice, though).

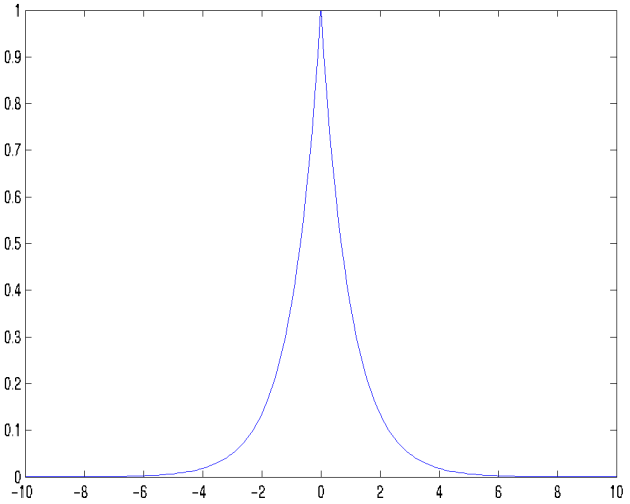
Laplacian Noise

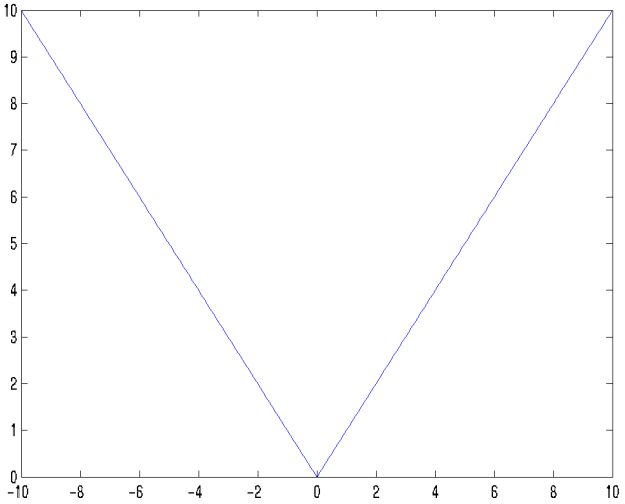
Noise Model

$$p(\xi) = \frac{\sigma}{2} \exp(-\sigma|\xi|)$$

This is a very long-tailed distribution. It occurs, e.g., in the decay of atoms: at any time, the probability that a given fraction of atoms will decay is constant. Result: even after 1000s of years there's still some C^{14} left.

Density and Log-Likelihood





Huber's Robust Density

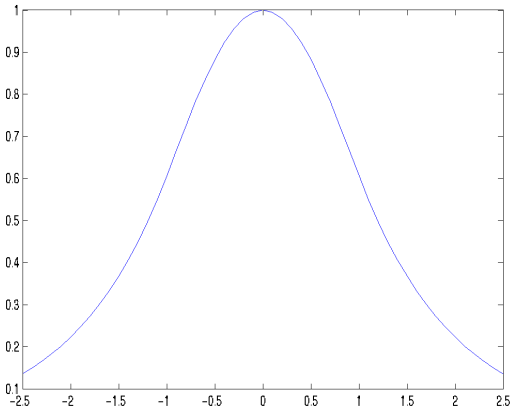
THE AUSTRALIAN NATIONAL UNIVERSITY

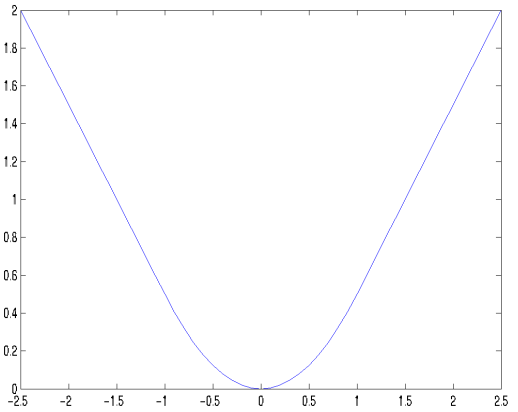
Problem: Sometimes we may not know what the additive density model of the likelihood is, in particular, how long-tailed the distribution may be.

Idea: Use the "worst" distribution as a reference. For distributions composed of a known (in our case Gaussian) part plus up to ε of an unknown part, we have the robust noise model

$$p(\xi) = \begin{cases} \frac{1}{2\sigma}\xi^2 & \text{if } |\xi| \leq \sigma \\ |\xi| - \frac{\sigma}{2} & \text{otherwise} \end{cases}$$

Density and Log-Likelihood





Coordinate Transformation

THE AUSTRALIAN NATIONAL UNIVERSITY

Problem

Minimization in terms of $\mathbf{t}$, the latent variables, is expensive, since it involves dealing with $\log p(\mathbf{t}) \propto \mathbf{y}^\top K^{-1}\mathbf{y}$, for which every calculation costs a matrix inversion.

Idea

Variable substitution from $\mathbf{t}$ to $\mathbf{t} = K\alpha$, which leads to $\alpha^\top K\alpha$.

Posterior for α

For the likelihood term we need $y_i = \xi_i + [K\alpha]_i$, hence

$$p(\alpha|X,Y) \propto \left[\prod_{i=1}^{m} p\left(y_i - [K\alpha]_i\right)\right] |K|^{\frac{1}{2}} \exp\left(-\frac{1}{2}\alpha^\top K\alpha\right)$$

Now the posterior looks similar to one for a generalized linear model, where the functions $k(x_i, \cdot)$ are the terms into which we expand the estimate.

MAP Approximation

THE AUSTRALIAN NATIONAL UNIVERSITY

Well Known Problem

Integrals are expensive, so we need an approximation.

Well Known Solution

Compute the maximum of the posterior and assume a known parametric distribution around the maximum (typically we choose a normal distribution).

Result

$$\text{minimize} \quad -\log p(\alpha|X,Y) = \sum_{i=1}^{m} -\log p(y_i - [K\alpha]_i) + \frac{1}{2}\alpha^\top K\alpha + \text{const.}.$$

Optimality Condition

$$K(c'(y_1 - [K\alpha]_1), \ldots, c'(y_m - [K\alpha]_m)) + K\alpha = 0$$

where $c(\xi) := -\log p(\xi)$. This looks very much like a loss function (see Bernhard's talk).

Connection to Support Vectors

THE AUSTRALIAN NATIONAL UNIVERSITY

Regularized Risk Functional

Here we minimize the loss on the training set, i.e.,

$$R_{\text{emp}}[f, X, Y] := \sum_{i=1}^{m} c(x_i, y_i, f(x_i))$$

plus a regularization term $\lambda\Omega[f]$, which typically is chosen to be $\Omega[f] = \frac{1}{2}\|f\|^2_{\mathcal{H}}$. In summary, we minimize

$$R_{\text{reg}}[f, X, Y] = R_{\text{emp}}[f, X, Y] + \lambda\Omega[f] = \sum_{i=1}^{m} c(x_i, y_i, f(x_i)) + \frac{\lambda}{2}\|f\|^2_{\mathcal{H}}$$

Empirical Risk — Log-Likelihood

Match up $-\log p(y_i|x_i, t_i)$ and $c(x_i, y_i, f(x_i))$, e.g., squared loss $\frac{1}{2}(y_i - t_i)^2$.

Regularization — Prior

Match up $-\log p(\alpha) = \frac{1}{2}\alpha^\top K\alpha + \text{cont.}$ and $\Omega[f] = \frac{1}{2}\|f\|_{\mathcal{H}} = \frac{1}{2}\alpha^\top K\alpha$.

Scaling Problems

Storage

We have to store the covariance matrix $K \in \mathbb{R}^{m \times m}$. On workstations this becomes a problem for $m > 10^4$.

Prediction

We have to sum up up to m kernel functions $k(x_i, x)$ to predict at x (covariances between training data and new test point). This becomes a problem for $m > 10^5$.

Training

Typically training involves at least one factorization of a matrix of size K. This is usually of order $O(m^3)$. On workstations we get problems if $m > 10^4$.

Solution

Approximate K by an object of lower rank. More on this later.

Overview of Unit 4: GP Classification

THE AUSTRALIAN NATIONAL UNIVERSITY

01: Estimating Probabilities
02: Logistic Regression
03: Multiclass Logistic Regression
04: Probit Model
05: Label Noise
06: Discriminant Analysis
07: MAP Approximation
08: Optimization Problems (Why Logit is good for you)
09: Laplace Approximation and Error Bars
10: Examples
11: Hyperparameters
12: Soft Margin Loss
13: How to fix it
14: Platt's Trick
15: Why all is well (Proof by Graph)
16: Scaling Problems

Estimating Probabilities

THE AUSTRALIAN NATIONAL UNIVERSITY

Classification Problem

Unlike in regression we have $y_i \in \mathcal{Y}$ with $|\mathcal{Y}| \in \mathbb{N}$, in other words, we have only a finite number of possible outcomes. Again, the goal is to estimate $p(y|x_i)$.

Special Case

Consider the binary classification problem where $\mathcal{Y} = \{\pm 1\}$.

Problem

It is easy to build estimators generating unconstrained functions $f(x)$, yet we need some tricks to make sure that p is normalized, i.e., $\sum_u p(y|x) = 1$.

Solution

We use a link function $l(y, f(x), x)$ connecting a real valued function f and $p(y|x, f) = l(y, f(x), x)$.

Logistic Regression

THE AUSTRALIAN NATIONAL UNIVERSITY

Basic Idea

For classification purposes we are mainly interested in the ratio between $p(y = 1|x)$ and $p(y = -1|x)$, since this tells us the Bayes optimal classifier (i.e., the classifier with minimal error).

Making the Problem Symmetric

Estimating $\frac{p(y=1|x)}{p(y=-1|x)}$ would help us find a classifier, but it isn't symmetric with respect to y. So we attempt to find f with

$$f(x) = \log \frac{p(y = 1|x)}{p(y = -1|x)} \Rightarrow p(y = 1|x) = \frac{1}{1 + \exp(-f(x))}.$$

Likewise $p(y = -1|x) = \frac{1}{1+\exp(f(x))}$,

Likelihood

For the likelihood we obtain

$$p(Y|X, f) = \prod_{i=1}^{m} \frac{1}{1 + \exp(-y_i f(x_i))} \Rightarrow -\log p(Y|X, f) = \sum_{i=1}^{m} \log(1+\exp(-y_i f(x_i))).$$

Multiclass Logistic Regression

THE AUSTRALIAN NATIONAL UNIVERSITY

Observation

We may write $p(y|x, f(x))$ as follows

$$p(y = 1|x, f(x)) = \frac{\exp(\frac{1}{2}f(x))}{\exp(\frac{1}{2}f(x)) + \exp(-\frac{1}{2}f(x))}$$

$$p(y = -1|x, f(x)) = \frac{\exp(-\frac{1}{2}f(x))}{\exp(\frac{1}{2}f(x)) + \exp(-\frac{1}{2}f(x))}$$

Idea

For more than two classes, estimate one function $f_j(x)$ per class and compute probabilities $p(y_j|x, f)$ via

$$p(y_j|x, f) = \frac{\exp(f_j(x))}{\sum_{i=1}^{N} \exp(f_i(x))}$$

Posterior

$$p(f|X, Y) \propto \prod_{i=1}^{m} \frac{\exp(f_{y_i}(x_i))}{\sum_{i=1}^{N} \exp(f_i(x_i))} \prod_{j=1}^{N} p(f_j)$$

Probit Model

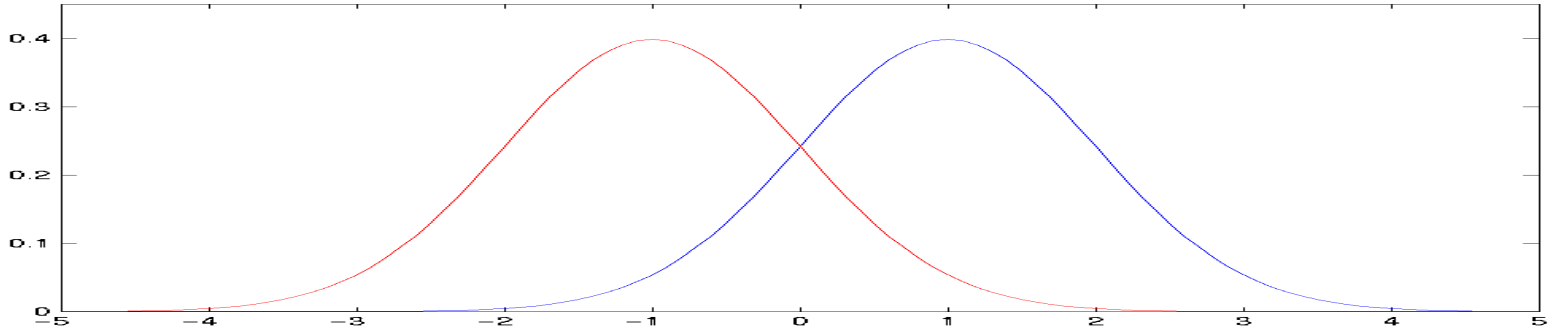
THE AUSTRALIAN NATIONAL UNIVERSITY

Basic Idea

We may assume that y is given by the sign of f, but corrupted by Gaussian noise; thus, $y = \operatorname{sgn}(f(x) + \xi)$ where $\xi \sim \mathcal{N}(0, \sigma)$. In this case, we have

$$\begin{aligned} p(y|f(x)) &= \int \frac{\operatorname{sgn}(yf(x) + \xi) + 1}{2} p(\xi) d\xi \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-yf(x)}^{\infty} \exp\left(-\frac{\xi^2}{2\sigma^2}\right) d\xi = \Phi\left(\frac{yf(x)}{\sigma}\right). \end{aligned}$$

Here Φ is the distribution function of the normal distribution.



Label Noise

Basic Idea

We want to perform classification in the presence of random label noise (in addition to the noise model $p_0(y|t)$ discussed previously).

Here, a label is randomly *assigned* to observations with probability 2η (note that this is the same as randomly *flipping* with probability η). We then write

$$p(y|f(x)) = \eta + (1 - 2\eta)p_0(y|f(x)).$$

Consequence

The influence of $p_0(y|f(x))$ on the posterior is descreased, hence η has a "regularizing" effect on the estimate.

Discriminant Analysis

Basic Idea

Assume that the classes to be separated (we assume $N = 2$ for simplicity) correspond to **Normal distributions** in some space, and that $f(x)$ are **projections** from this space onto a line.

Result

Projections on a real line yield normal distributions. Hence we can model the probability $p(y|x, f(x))$ by

$$p(y|x, f(x)) \propto \exp\left(-\frac{1}{2}(y - f(x))^2\right).$$

Algorithmic Result

This is essentially **regression on the labels**, which can be done very cheaply.

Problem: often the assumption of a normal distribution is not so well satisfied.

MAP Approximation

Log-Posterior

Instead of integrating over $p(f|X,Y)$ we minimize the negative log-posterior. To make matters simpler, we reparameterize $f = K\alpha$.

$$-\log p(f|X,Y) = \sum_{i=1}^{m} -\log l(y_i, x_i, [K\alpha]_i) + \frac{1}{2}\alpha^\top K\alpha.$$

Practical Issues

- Convex loss functions lead to optimization problems with a **global minimum**. Proof: assume two (local) minima at, say $\mathbf{t}_1, \mathbf{t}_2$, then for all arguments $\lambda\mathbf{t}_1 + (1-\lambda)\mathbf{t}_2$ the values will be less or equal to the linear interpolation. This, however, is a contradiction.
- Choice of link function determines whether the optimization problem.
- Morale of the story: choose link function according to data **and** numerical considerations.

Examples

Penalized Logistic Regression

We use the logistic link function, which leads to the following minimization problem:

$$\text{minimize} \sum_{i=1}^{m} \log\left(1 + \exp\left(-y_i \sum_{j=1}^{m} k(x_i, x_j)\alpha_j\right)\right) + \frac{1}{2}\alpha^\top K\alpha$$

where $f = K\alpha$

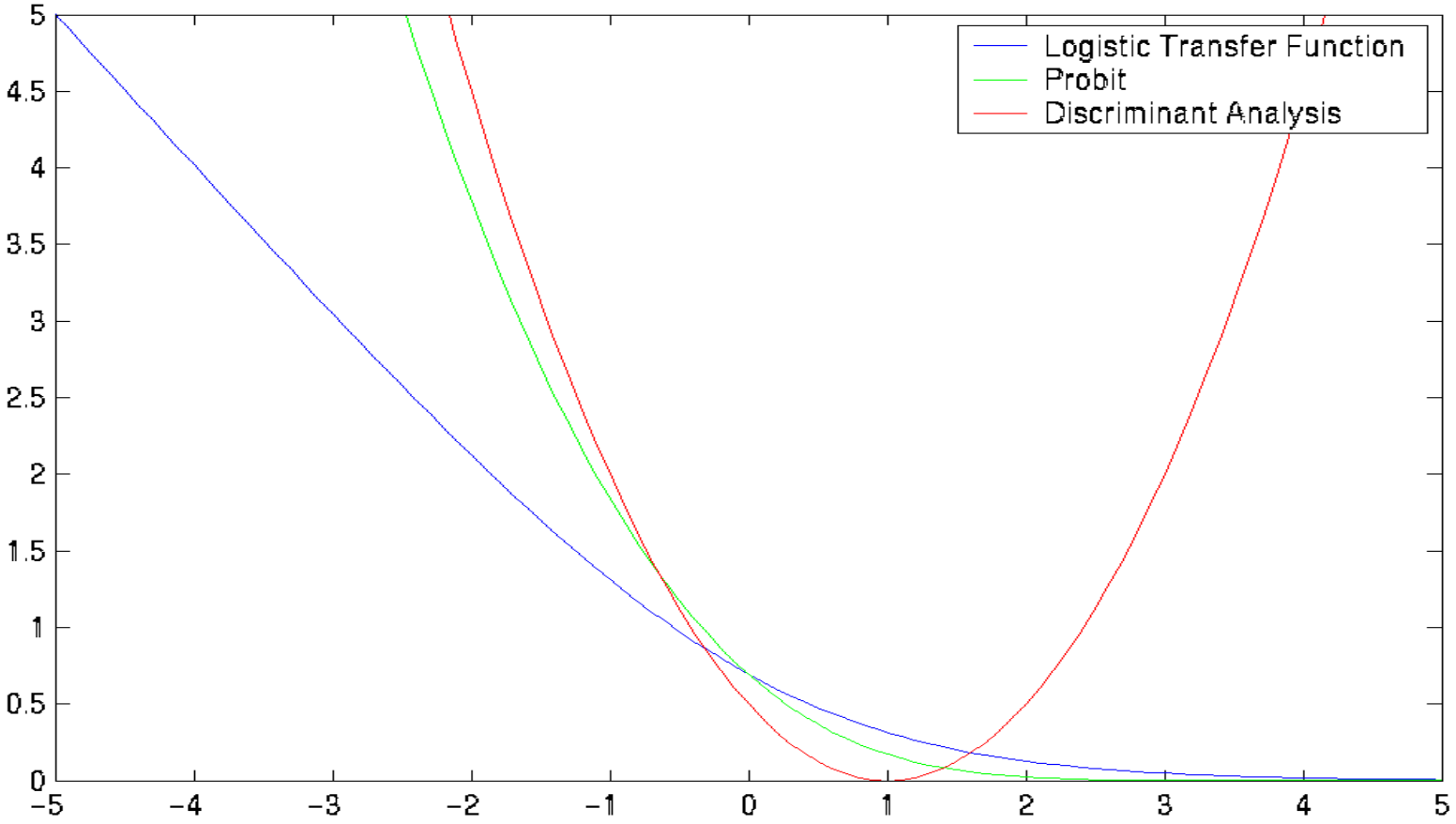
Prediction

For a new instance we obtain $f(x) = \sum_{i=1}^{m} \alpha_i k(x_i, x)$ and subsequently predict $y = 1$ if $f(x) > 0$ and $y = -1$ otherwise.

Confidence Ratings

For each observation we get $p(y = 1|x, y) = \frac{1}{1+\exp(f(x))}$.

Link Functions



Examples

Penalized Logistic Regression

We use the logistic link function, which leads to the following minimization problem:

$$\text{minimize} \sum_{i=1}^{m} \log\left(1 + \exp\left(-y_i \sum_{j=1}^{m} k(x_i, x_j)\alpha_j\right)\right) + \frac{1}{2}\alpha^\top K\alpha$$

where $f = K\alpha$

Prediction

For a new instance we obtain $f(x) = \sum_{i=1}^{m} \alpha_i k(x_i, x)$ and subsequently predict $y = 1$ if $f(x) > 0$ and $y = -1$ otherwise.

Confidence Ratings

For each observation we get $p(y = 1|x, y) = \frac{1}{1+\exp(f(x))}$.

Soft Margin Loss

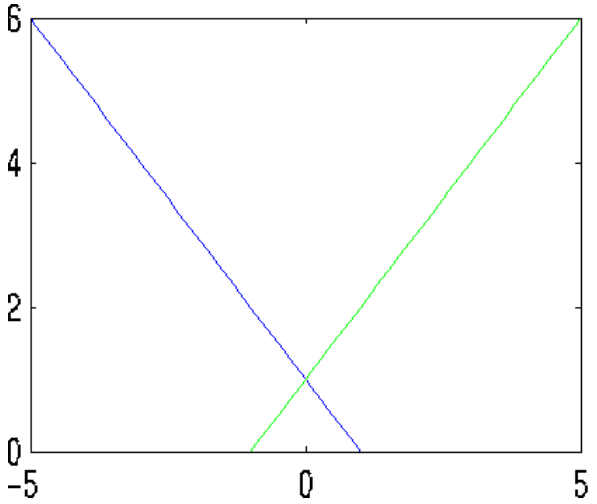
Support Vector Loss Function: In SVM one uses as a loss function

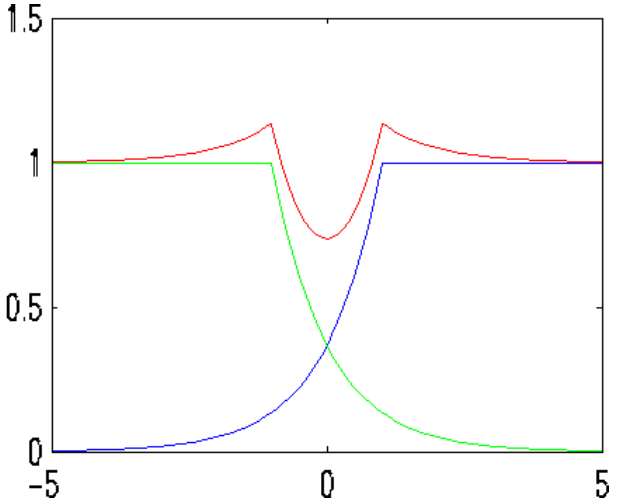
$$c(x, y, f(x)) = \max(0, 1 - yf(x))$$

Using the correspondence between loss functions and log-likelihood, we would get

$$p(y|x, y, f(x)) = \exp(-\max(0, 1 - yf(x))) = \min(1, \exp(yf(x) - 1))$$

Problem: Probabilities don't sum up to 1.





How to fix it

Idea 1

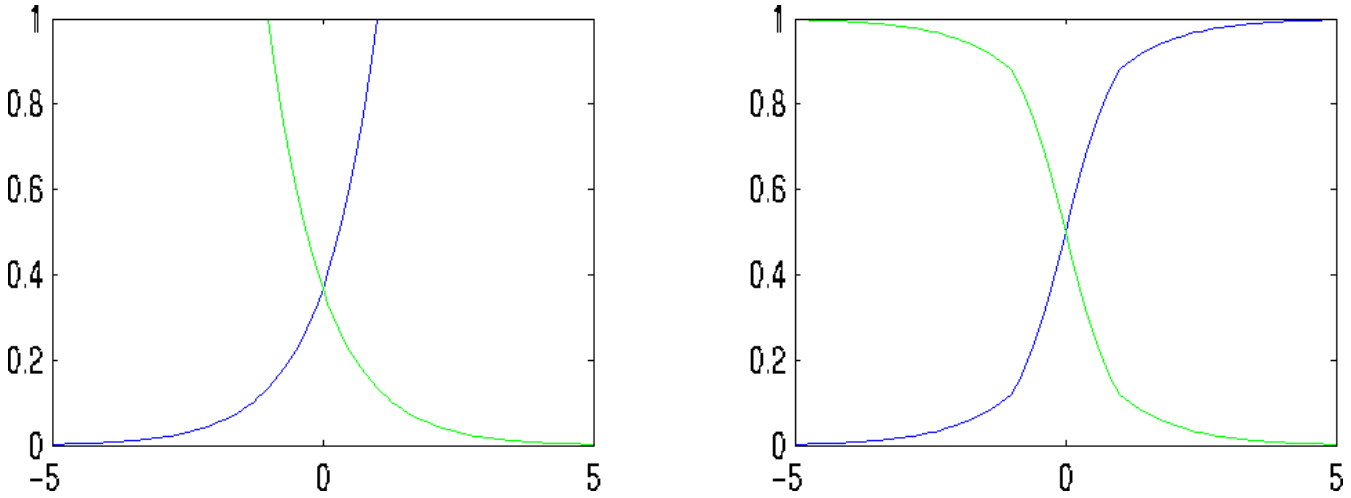
Introduce a "Don't Know" class. This makes sense inside the margin, since we may not be sure which label we have ...

Problem

The "Don't Know" class increases again for large $|f(x)|$. This does not make sense.

Idea 2

Ignore all don't know elements and re-normalize to 1.



Platt's Trick

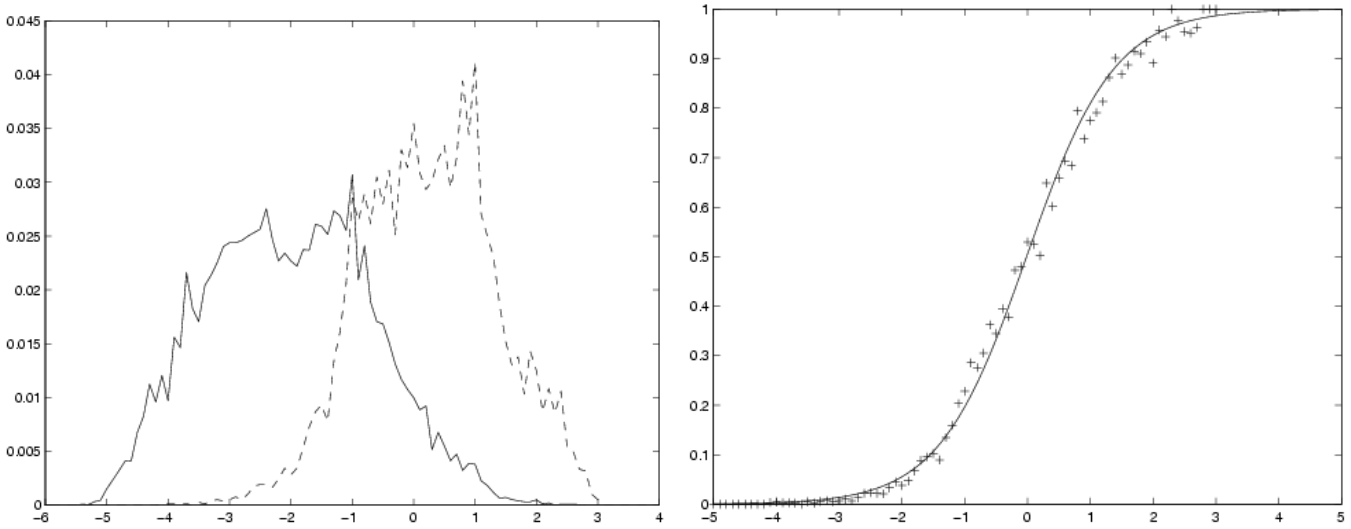
THE AUSTRALIAN NATIONAL UNIVERSITY

Problem

After obtaining an estimator with a Support Vector Machine we would like to have probabilities (of course, we could have trained a GP estimator straight away) ...

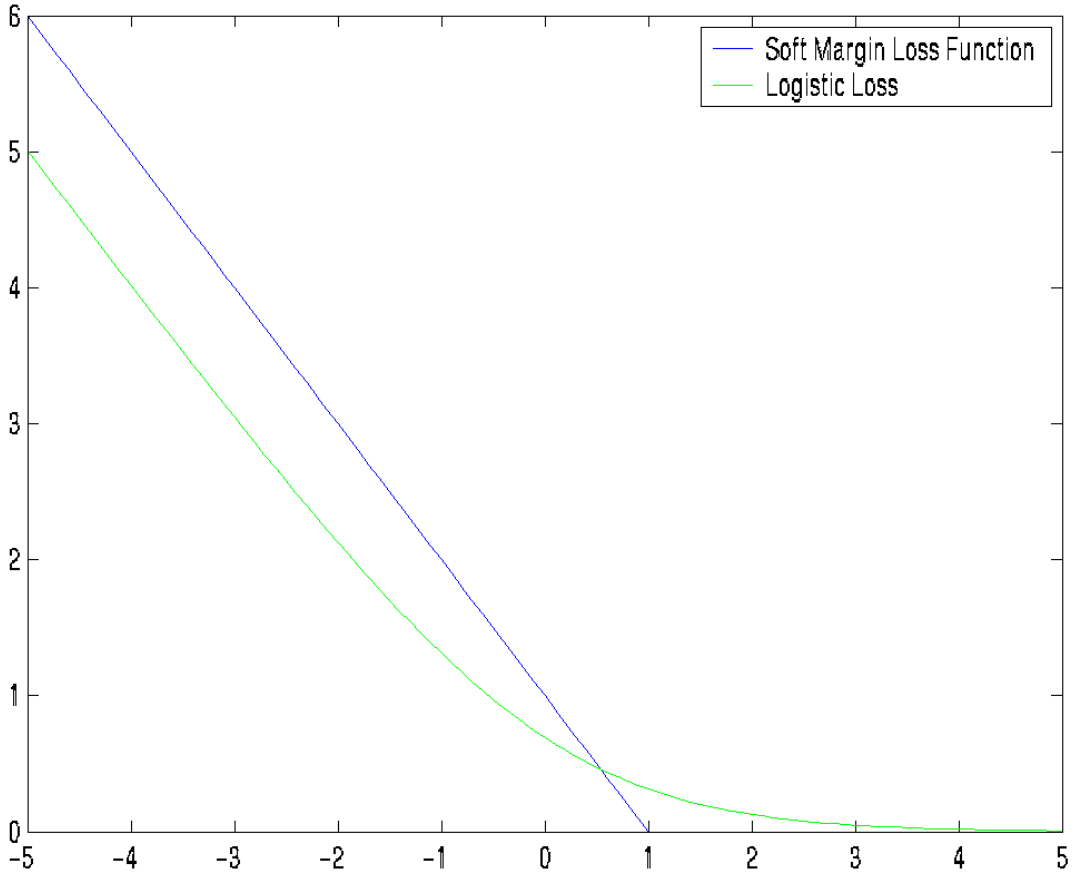
Solution Fit a logistic model to the function values $f(x)$, i.e., we

$$\underset{a,b}{\text{maximize}}\, p(Y|f,X) = \prod_{i=1}^{m} \frac{1}{1+\exp(-ay_i f(x_i)+b)}$$



Why all is well (Proof by Graph)

THE AUSTRALIAN NATIONAL UNIVERSITY



Overview of Unit 5: Low Rank Methods

THE AUSTRALIAN NATIONAL UNIVERSITY

01: Spectrum of Covariance Matrix
02: Equations for GP Regression
03: A Bounding Theorem
04: Proof
05: Approximation by PCA
06: Example
07: Projection on Subsets
08: Sparse Greedy Methods
09: A Subset Trick
10: Example
11: A Gradient Lemma
12: Coordinate Descent and Convergence
13: Proof
14: Selection Rule
15: Algorithm
16: Example

A Simple Implementation

THE AUSTRALIAN NATIONAL UNIVERSITY

Idea

Minimize the negative log-likelihood with the Newton method.

Basic Algorithm

To minimize a function $\mathcal{L}(f)$ which is twice differentiable in f approximate

$$\mathcal{L}(f + \Delta f) \approx \mathcal{L}(f) + \Delta f \mathcal{L}'(f) + \frac{1}{2}\Delta f^\top \mathcal{L}''(f)\Delta f$$

Hence we may approximately compute the minimum via

$$f \leftarrow f - (\mathcal{L}''(f))^{-1}\mathcal{L}'(f)$$

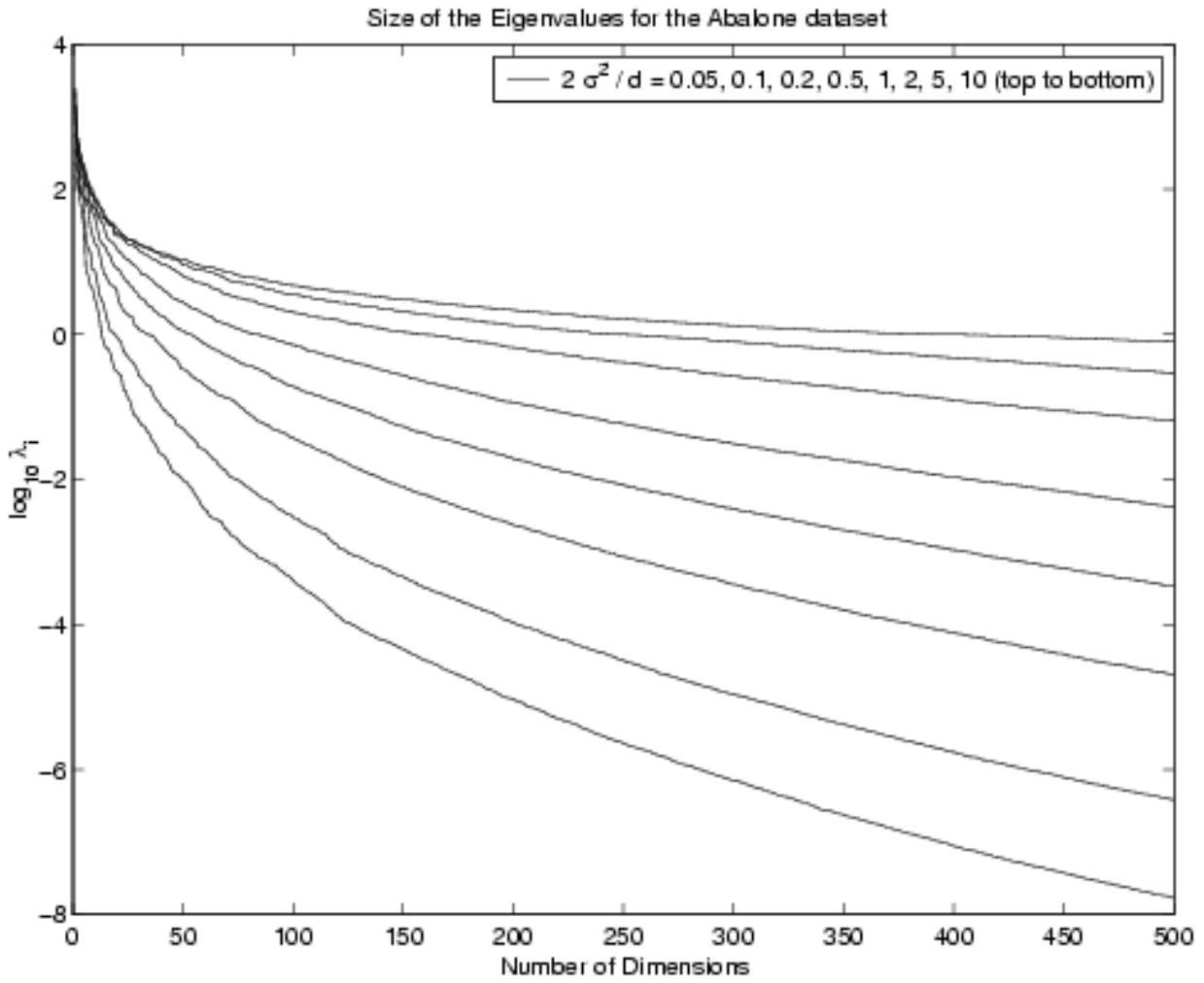
Practical Consequence

From $\mathcal{L}(f) = \sum_{i=1}^m -\log p(y_i|[K\alpha]_i, x_i) + \frac{1}{2}\alpha^\top K\alpha$ (with the usual parameterization $f = K\alpha$) we obtain

$$\alpha \leftarrow \alpha - (K + K^\top C'' K)^{-1} K c'$$

where $c'_i = \partial^1_{[K\alpha]_i} - \log p(y_i|[K\alpha]_i, x_i)$ and $C''_{ii} = \partial^2_{[K\alpha]_i} - \log p(y_i|[K\alpha]_i, x_i)$.

Spectrum of Covariance Matrix



Practical Consequences

Ill conditioned matrix

Inverting K or products thereof is numerically unstable procedure.

Observation

Removing the smallest eigenvalues/eigenvectors, we obtain almost the same solution.

Computational Speed

Smaller matrices mean that we can solve each Newton step more efficiently (in a nutshell, from $O(m^3)$ cost we go to $O(mn^2)$)

Prediction

If we **could** compute the functions corresponding to the eigensystem of K directly, this **would** speed prediction up from $O(m)$ to $O(n)$.

Plan (for today)

Replace the PCA with something more efficient, where we only need to compute n covariance functions $k(x_i, \cdot)$.

Recall: Gaussian Process Regression

THE AUSTRALIAN NATIONAL UNIVERSITY

Goal

Find distribution of y at location x (i.e. **mean** and **variance** of the normal distribution) by integrating out the normal distribution in the rest.

Solution: Denote by $\mathbf{k} = (k(x_1, x), \ldots, k(x_m, x))$. Then we have

$$\boxed{\mathbf{E}[y] = \mathbf{k}^\top (K + \sigma^2 \mathbf{1})^{-1} \mathbf{y}} \quad \text{and} \quad \boxed{\mathrm{Var}[y] = k(x,x) + \sigma^2 - \mathbf{k}^\top (K + \sigma^2 \mathbf{1})^{-1} \mathbf{k}}$$

Modified Solution

If we have to predict at several points it pays to compute $\alpha^* := (K + \sigma^2 \mathbf{1})^{-1} \mathbf{y}$ and predict the mean of y by $\mathbf{k}^\top \alpha$.

Idea: Find α and $\mathbf{k}^\top (K + \sigma^2 \mathbf{1})^{-1} \mathbf{k}$ by minimizing quadratic forms:

$$\begin{aligned} \alpha^* &= \operatorname*{argmin}_\alpha \left[-\mathbf{y}^\top K \alpha + \frac{1}{2} \alpha^\top (K^\top K + \sigma^2 K) \alpha \right] \\ \mathbf{k}^\top (K + \sigma^2 \mathbf{1})^{-1} \mathbf{k} &= 2 \cdot \min_\alpha \left[-\mathbf{k}^\top \alpha + \frac{1}{2} \alpha^\top (K + \sigma^2 \mathbf{1}) \alpha \right] \end{aligned}$$

Approximating Quadratic Forms

THE AUSTRALIAN NATIONAL UNIVERSITY

Theorem

Denote by $K \in \mathbb{R}^{m \times m}$ a positive semidefinite matrix, $\mathbf{y}, \alpha \in \mathbb{R}^m$ and define the two quadratic forms

$$Q(\alpha) := -\mathbf{y}^\top K\alpha + \frac{1}{2}\alpha^\top(\sigma^2 K + K^\top K)\alpha,$$
$$Q^*(\alpha) := -\mathbf{y}^\top \alpha + \frac{1}{2}\alpha^\top(\sigma^2 \mathbf{1} + K)\alpha.$$

Suppose Q and Q^* have minima $Q_{\min}$ and $Q^*_{\min}$. Then for all $\alpha, \alpha^* \in \mathbb{R}^m$

$$Q(\alpha) \geq Q_{\min} \geq -\frac{1}{2}\|\mathbf{y}\|^2 - \sigma^2 Q^*(\alpha^*),$$
$$Q^*(\alpha^*) \geq Q^*_{\min} \geq \sigma^{-2}\left(-\frac{1}{2}\|\mathbf{y}\|^2 - Q(\alpha)\right),$$

with equalities throughout when $Q(\alpha) = Q_{\min}$ and $Q^*(\alpha^*) = Q^*_{\min}$.

Proof

Minimum of $Q(\alpha)$

The minimum of $Q(\alpha)$ is obtained for $\alpha_{\text{opt}} = (K + \sigma^2 \mathbf{1})^{-1}\mathbf{y}$ (which also minimizes Q^*), hence

$$Q_{\min} = -\frac{1}{2}\mathbf{y}^\top K(K + \sigma^2 \mathbf{1})^{-1}\mathbf{y} \text{ and } Q^*_{\min} = -\frac{1}{2}\mathbf{y}^\top (K + \sigma^2 \mathbf{1})^{-1}\mathbf{y}.$$

Combining Q and Q^*

This allows us to combine the minima to

$$Q_{\min} + \sigma^2 Q^*_{\min} = -\frac{1}{2}\|\mathbf{y}\|^2.$$

Minimum Property of Q, Q^*

Since by definition $Q(\alpha) \geq Q_{\min}$ for all α (and likewise $Q^*(\alpha^*) \geq Q^*_{\min}$ for all α^*), we may solve $Q_{\min} + \sigma^2 Q^*_{\min}$ for either Q or Q^* to obtain lower bounds for each of the two quantities.

Decomposition and Update

THE AUSTRALIAN NATIONAL UNIVERSITY

Recall: Objective Functions

$$\begin{aligned} Q(\alpha) &:= -\mathbf{y}^\top K\alpha + \frac{1}{2}\alpha^\top(\sigma^2 K + K^\top K)\alpha, \\ Q^*(\alpha) &:= -\mathbf{y}^\top \alpha + \frac{1}{2}\alpha^\top(\sigma^2 \mathbf{1} + K)\alpha. \end{aligned}$$

Ansatz

Use $P \in \mathbb{R}^{m\times n}$ (as an **extension** matrix) to approximate α by $P\beta$. In particular, P contains only one nonzero entry per column.

Optimal solution in β

$$\begin{aligned} \beta_{\text{opt}} &= \left(P^\top\left(\sigma^2 K + K^\top K\right)P\right)^{-1} P^\top K^\top \mathbf{y} \\ \beta^*_{\text{opt}} &= \left(P^\top\left(\sigma^2 \mathbf{1} + K\right)P\right)^{-1} P^\top \mathbf{k} \end{aligned}$$

Decomposition and Update

THE AUSTRALIAN NATIONAL UNIVERSITY

Idea

We can obtain the inverse matrices by a rank 1 update at $O(mn)$ cost if we know the inverse for P_{old} where $P = [P_{\text{old}}, \mathbf{e}_j]$).

$$P^\top K^\top \mathbf{y} = [P_{\text{old}}, \mathbf{e}_i]^\top K^\top \mathbf{y} = (P_{\text{old}}^\top K^\top \mathbf{y}, \mathbf{k}_i^\top \mathbf{y})$$

$$P^\top \left(K^\top K + \sigma^2 K\right) P = \begin{bmatrix} P_{\text{old}}^\top \left(K^\top K + \sigma^2 K\right) P_{\text{old}} & P_{\text{old}}^\top \left(K^\top + \sigma^2 \mathbf{1}\right) \mathbf{k}_i \\ \mathbf{k}_i^\top (K + \sigma^2 \mathbf{1}) P_{\text{old}} & \mathbf{k}_i^\top \mathbf{k}_i + \sigma^2 K_{ii} \end{bmatrix}$$

Strategy

Try out several new randomly chosen basis functions at each iteration and pick the one which minimizes the objective function most.

Performance Guarantee

With high probability we will find one of the best basis functions (e.g., with a subset of 59 we'll get a 95% guarantee).

Why do random subsets work?

THE AUSTRALIAN NATIONAL UNIVERSITY

Theorem

Given a random variable ξ with cumulative distribution function $F(\xi)$, then for n instances $\xi_1, \ldots \xi_m$ of ξ and $\xi_i \sim \partial_\xi F(\xi)$

$$\zeta := \max\{\xi_1, \ldots, \xi_n\} \text{ we have } F(\zeta) = F^n(\xi).$$

Corollary

The cumulative distribution of percentiles χ (i.e. fraction of samples larger than χ) for ζ is bounded from below by $F(\chi) = \chi^n$.

Practical Consequence

We only need at most $\left\lceil \frac{\log \delta}{\log(1-\eta)} \right\rceil$ samples in order to obtain a sample among the best δ with $1 - \eta$ confidence.

In particular 59 samples suffice to obtain with 95% probability a sample that is better than 95% of the rest.

Comparison with Other Methods

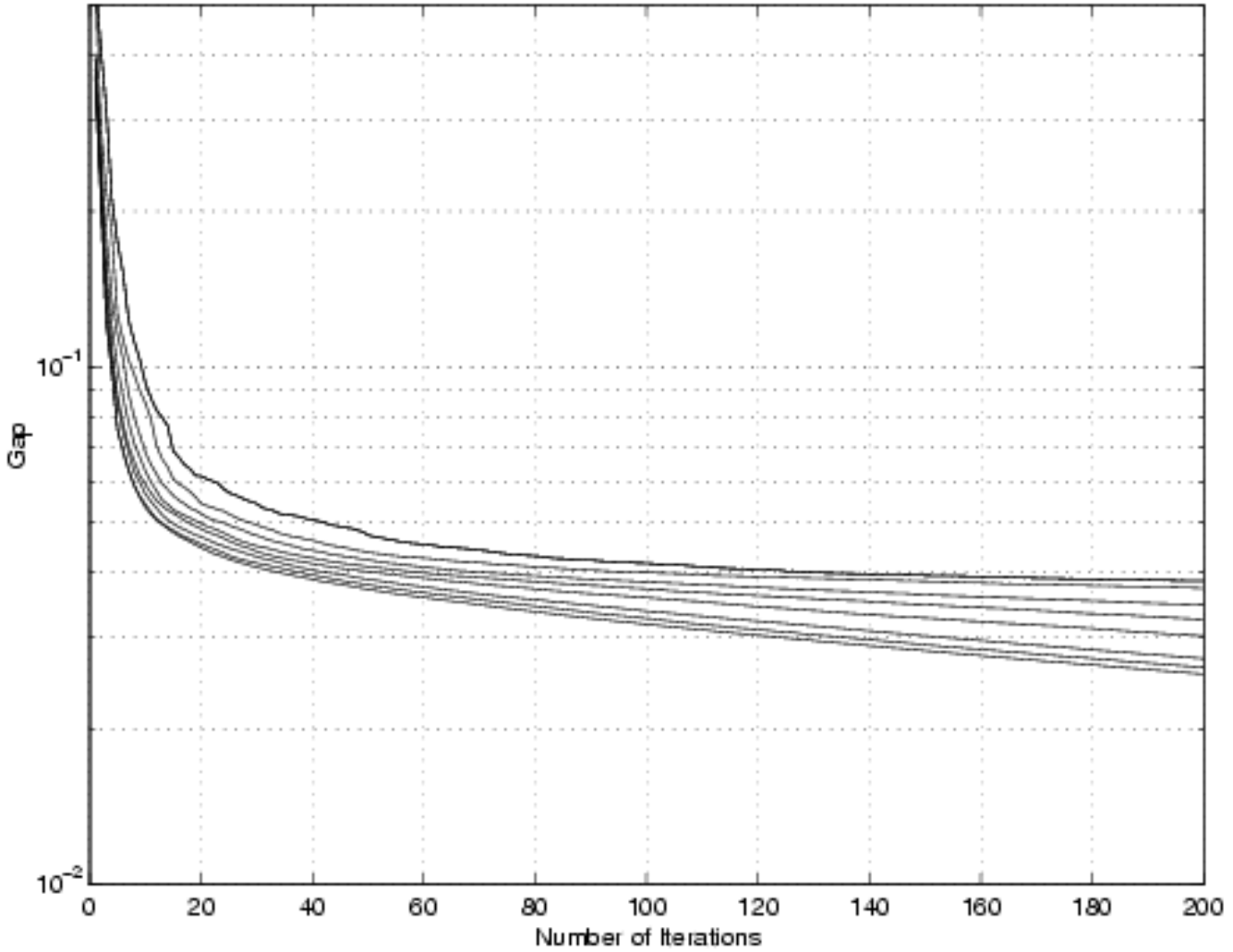
	Exact Solution	Conjugate Gradient	Sparse Decomposition	Sparse Greedy Approximation
Memory	$O(m^2)$	$O(m^2)$	$O(nm)$	$O(nm)$
Initialization	$O(m^3)$	$O(nm^2)$	$O(n^2m)$	$O(\kappa n^2 m)$
Prediction:				
Mean	$O(m)$	$O(m)$	$O(n)$	$O(n)$
Error Bars	$O(m^2)$	$O(nm^2)$	$O(n^2m)$ or $O(n^2)$	$O(\kappa n^2 m)$ or $O(n^2)$

Optimal Rate

The sparse decomposition rates would be optimal but can only be obtained after an NP hard search for the best basis.

Note that $n \ll m$ and that the n used in CG, SD, and SGA methods will differ, with $n_{\text{CG}} \leq n_{\text{SD}} \leq n_{\text{SGA}}$ since the search spaces are more restricted.

Speed of Convergence



Size of the gap between upper and lower bound of the log posterior, i.e. $Q(\alpha)$ for the first 4000 samples from the Abalone dataset. From top to bottom: subsets of size 1, 2, 5, 10, 20, 50, 100, 200.

Basis Functions and Performance

THE AUSTRALIAN NATIONAL UNIVERSITY

Generalization Performance of Greedy Gaussian Processes

	Generalization Error	Log Posterior
Optimal Solution	1.782 ± 0.33	$-1.571 \cdot 10^5(1 \pm 0.005)$
Sparse Greedy Approximation	1.785 ± 0.32	$-1.572 \cdot 10^5(1 \pm 0.005)$

Kernels needed to minimize the log posterior, depending on the width of the Gaussian kernel ω. Also, number of basis functions required to approximate $\mathbf{k}^\top(K + \sigma^2\mathbf{1})^{-1}\mathbf{k}$ which is needed to compute the error bars.

Kernel width $2\omega^2$	1	2	5	10	20	50
Kernels for log-posterior	373	287	255	257	251	270
Kernels for error bars	79±61	49±43	26±27	17±16	12±9	8±5

Projections on Subspace

THE AUSTRALIAN NATIONAL UNIVERSITY

Basic Idea

Even for arbitrary posteriors, using only a subset of coefficients, i.e., $P\beta$ instead of α, will allow us to find rather good approximations. We then minimize

$$-\log \mathcal{L}(P\beta, X, Y) = \sum_{i=1}^{m} -\log p(y_i|x_i, [KP\beta]_i) + \frac{1}{2}\beta^\top P^\top K P \beta$$

Now we can minimize a smaller optimization problem which costs $O(mn^2)$ (details on this later).

Parameter Transformation

We now switch to a parameter space in which the GP prior will become **diagonal**.

Without loss of generality assume that P picks the first n coefficients: $P = \begin{bmatrix} \mathbf{1} \\ \mathbf{0} \end{bmatrix}$.

Note: in numerical mathematics this process arises from Gauss elimination of the the rows of the covariance matrix .

Projections on Subspace, Part II

THE AUSTRALIAN NATIONAL UNIVERSITY

Gauss Elimination

Transform $K = \begin{bmatrix} K^{nn} & K^{mn} \\ (K^{mn})^\top & K^{mm} \end{bmatrix}$ into $\tilde{K} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & K^{mm} - (K^{mn})^\top (K^{nn})^{-1} K^{mn} \end{bmatrix}$ by $\begin{bmatrix} (K^{nn})^{-\frac{1}{2}} & -(K^{nn})^{-1} K^{mn} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}$.

The term $\tilde{K} := K^{mm} - (K^{mn})^\top (K^{nn})^{-1} K^{mn}$ is often referred to as the Schur complement.

Terms of the Optimization Problem

Reparameterizing by $\alpha = \begin{bmatrix} (K^{nn})^{-\frac{1}{2}} & -(K^{nn})^{-1} K^{mn} \\ & \mathbf{1} \end{bmatrix} \begin{bmatrix} \beta_n \\ \beta_m \end{bmatrix}$ yields

$$\alpha^\top K \alpha \to \|\beta_n\|^2 + \beta_m^\top \tilde{K} \beta_m \text{ and } K\alpha \to \begin{bmatrix} (K^{nn})^{\frac{1}{2}} \\ K^{mn}(K^{nn})^{-\frac{1}{2}} \end{bmatrix} \beta_n + \begin{bmatrix} \mathbf{0} \\ \tilde{K} \end{bmatrix} \beta_m$$

Projections on Subspace, Part III

THE AUSTRALIAN NATIONAL UNIVERSITY

Gradients of Log-Posterior

$$\partial_{\beta_n} - \log \mathcal{L} = \begin{bmatrix} (K^{nn})^{\frac{1}{2}} \\ K^{mn}(K^{nn})^{-\frac{1}{2}} \end{bmatrix} \mathbf{c}' + \beta_n$$

$$\partial_{\beta_m} - \log \mathcal{L} = \begin{bmatrix} \mathbf{0} \\ \tilde{K} \end{bmatrix} \mathbf{c}' + \tilde{K}\beta_m$$

Hessian

$$\partial^2_{\beta_n} - \log \mathcal{L} = \begin{bmatrix} (K^{nn})^{\frac{1}{2}} \\ K^{mn}(K^{nn})^{-\frac{1}{2}} \end{bmatrix}^\top \mathbf{c}'' \begin{bmatrix} (K^{nn})^{\frac{1}{2}} \\ K^{mn}(K^{nn})^{-\frac{1}{2}} \end{bmatrix} + \mathbf{1}$$

$$\partial^2_{\beta_m} - \log \mathcal{L} = \begin{bmatrix} \mathbf{0} \\ \tilde{K} \end{bmatrix}^\top \mathbf{c}'' \begin{bmatrix} \mathbf{0} \\ \tilde{K} \end{bmatrix} + \tilde{K}$$

where $c_i = -\log p(y_i|x_i, f(x_i)$ and the derivatives are taken wrt. $f(x_i)$.

Newton Method

THE AUSTRALIAN NATIONAL UNIVERSITY

Recall

We have updates $f \leftarrow f - (\mathcal{L}''(f))^{-1}\mathcal{L}'(f)$.

Updates in β_n

To optimize over the subspace spanned by the first n covariance functions, we only need to compute

$$\beta_n \leftarrow \beta_n - (Z\mathbf{c}''Z^\top)^{-1}(Z\mathbf{c}' + \beta_n) \text{ where } Z := \begin{bmatrix} (K^{nn})^{\frac{1}{2}} \\ K^{mn}(K^{nn})^{-\frac{1}{2}} \end{bmatrix}.$$

Computational Cost

Storage requirement is $O(mn)$ for Z and $O(n^2)$ for K^{nn}. CPU cost per inversion is $O(mn^2)$ to compute $(Z\mathbf{c}''Z^\top)$, plus $O(n^3)$ for the inversion. That is, if the space is spanned by a small number of basis functions, the estimation process is **linear** in the number of observations.

A Gradient Lemma

Problem

We need to know when to stop the optimization. For this purpose we use a bound in terms of the gradient of the log likelihood.

Lemma

Denote by $\mathcal{P}(\beta)$ a differentiable convex functions with $\mathcal{P}(\beta) = \mathcal{L}(\beta) + \frac{1}{2}\beta^\top M\beta$. Then we have

$$\min_\beta \mathcal{P}(\beta) \geq \mathcal{P}(\tilde{\beta}) - \frac{1}{2}\left[\partial_\beta \mathcal{P}(\tilde{\beta})\right]^\top M^{-1}\left[\partial_\beta \mathcal{P}(\tilde{\beta})\right].$$

Proof Idea

A linear approximation of $\mathcal{L}(\beta)$ at $\mathcal{L}(\tilde{\beta})$ is a lower bound on $\mathcal{L}(\beta)$. This allows us to compute lower bound the minimum of $\mathcal{P}(\beta)$.

Selection Rule

Application of the Bound

If the gradients and the Hessian in β factorize as in the previous case, we obtain

$$\Delta\left[-\log p(\beta|X,Y)\right] \leq \frac{1}{2}\|Z\mathbf{c}' + \beta_n\|^2 + \frac{1}{2}(\mathbf{c}'_m + \beta_m)^\top \tilde{K}(\mathbf{c}'_m + \beta_m).$$

Here $\mathbf{c}'_m$ is the part of $\mathbf{c}'$ corresponding to β_m.

Problem

Which basis function to add to β_n (after the gradient on β_n vanishes)?

Approximate Solution

Since $\beta_m = 0$ we can rewrite the β_m term as $\frac{1}{2}(\mathbf{c}'_m)^\top \tilde{K}\mathbf{c}'_m$. Computing this is **expensive**, the diagonal terms, however, are cheap. We bound

$$\sqrt{(\mathbf{c}'_m)^\top \tilde{K}\mathbf{c}'_m} \leq \sum_{i=n+1}^{m} \sqrt{\tilde{K}_{ii}}|c'_i|$$

Hence, pivoting for i with large $\tilde{K}_{ii}(c'_i)^2$ is a good idea.

Overview of Unit 6: Bayes Committee Machine

THE AUSTRALIAN NATIONAL UNIVERSITY

01: Splitting the Data
02: Bayes Committee Machine
03: Joining the Posterior
04: Proof
05: Sherman-Morrison-Woodbury
06: Predicting for Small Test Set
07: Generalized BCM

Splitting the Data

Idea

If we have too much data to minimize the log-posterior directly, we could simply use the following strategy:

- split into chunks
- optimize over each of the chunks independently
- average over the results

Problems

- how to average
- how to improve confidence ratings
- what is the form of the optimization problem on the chunks
- connection to the exact solution

Bayes Committee Machine (Tresp)

THE AUSTRALIAN NATIONAL UNIVERSITY

Basic Idea

Split data D into N chunks $D_1, \ldots, D_N$. By Bayes' rule we have

$$p(f|D_i, D_{i-1}) \propto p(D_i|f, D_{i-1})p(f|D_{i-1})$$

Approximation To be able to expand $p(f|D_1, \ldots, D_N)$ into terms of $p(f|D_i)$ we approximate

$$p(D_i|f, D_{i-1}) \approx p(D|f)$$

This would be true for function generating the data (given the underlying hypothesis, the individual data blocks are independent), in our case it is just an approximation.

Result

$$p(f|D_i) \propto \left(\prod_{i=1}^{N} p(D_i|f)\right) p(f) = \frac{\prod_{i=1}^{N} p(f|D_i)p(f)}{p^{N-1}(f)}$$

Now we may approximate each of the $p(f|D_i)p(f)$ and combine the results.

Joining the Posterior

Laplace Approximation

We approximate each $p(f|D_i)p(f)$ by a normal distribution.

Combining Normal Distributions

Taking products of normal distributions with means μ_i and covariances Σ_i leads to an overall normal distribution with

$$\Sigma^{-1} = \sum_{i=1}^{N} \Sigma_i^{-1} \text{ and } \mu = \Sigma \sum_{i=1}^{N} \Sigma_i^{-1} \mu_i$$

For quotients the signs are reversed.

Combined Posterior

Given the GP prior $p(f)$ with covariance matrix Σ_G we obtain

$$\Sigma^{-1} = (1-N)\Sigma_G + \sum_{i=1}^{N} \Sigma_i^{-1} \text{ and } \mu = \Sigma \sum_{i=1}^{N} \Sigma_i^{-1} \mu_i$$

GP Regression

Estimate on Subset

For regression with normal additive noise we have

$$\mathbf{f}_i = K^{mn}(K^{nn} + \sigma^2\mathbf{1})^{-1}\mathbf{y} \text{ and } \Sigma_i = K^{mm} - K^{mn}(K^{nn} + \sigma^2\mathbf{1})^{-1}\left(K^{mn}\right)^\top$$

where we labelled all the predictive part with m and the given part with n.

Combining Individual Predictions

$$\begin{aligned}\text{Covar } \Sigma^{-1} &= (1-N)K^{mm} + \sum_{i=1}^{N}\left(K^{mm} - K_i^{mn}(K_i^{nn} + \sigma^2\mathbf{1})^{-1}\left(K_i^{mn}\right)^\top\right) \\ &= \Sigma\sum_{i=1}^{N}\Sigma_i^{-1}\mathbf{f}_i\end{aligned}$$

Why Does It Work?

THE AUSTRALIAN NATIONAL UNIVERSITY

A Simple Idea

Given functions $g_0, g_1, \ldots, g_N : \mathbb{R}^n \to \mathbb{R}$ we want to minimize

$$g(\alpha) := g_0(\alpha) + \sum_{i=1}^{N} g_i(\alpha).$$

Instead, we **minimize each $\tilde{g}_i := g_0 + g_i$ separately**, compute a quadratic approximation q_i of $\tilde{g}_i$ at its minimum, and subsequently minimize $q := \sum_{i=1}^{N} q_i - (N-1)g_0$.

Clearly, if all g_i are quadratic functions, this procedure is exact. Otherwise it is a good first iteration.

Maximizing the Posterior

We want to minimize the negative log posterior. For GP regression with Normal noise this is a **quadratic function**. For each of the partial negative log-posteriors the approximation is exact, hence the overall estimate is exact.

Iterative Extension

A Simple Idea

Use the quadratic approximations q_i to improve the estimates at the next iteration:

- Find initial approximations q_i by minimizing $g_i + g_0$.
- Repeat
- minimize $g_i + \sum_{j=1, j \neq i}^{N} q_j$
- compute quadratic approximation at minimum
- Until converged

When to use

- If we have a simple minimization algorithm which cannot deal with $g = \sum_i f_i$ simultaneously (too much data).
- If we have a ready-made optimizer for the subproblems.
- Otherwise, Newton method should be better (after all, we need an algorithm to minimize each of the auxiliary functions).

Online-BCM

Idea

If we observe a new instance (x_{m+1}, y_{m+1}), we can make the approximation

$$p(f|X, Y, (x_{m+1}, y_{m+1})) \approx p(X, Y|f)p(f|(x_i, y_i))$$

and simply update mean and covariance according to the combination strategy.

Advantage

We only need to store mean and covariance for updates. No need to remember the training data (for GP regression exact, since mean and variance are **sufficient statistics** of a Normal distribution).

Kalman Filter

Update equations are identical (again, propagating a Normal distribution in time).

Overview of Unit 7: Relvance Vector Machine

THE AUSTRALIAN NATIONAL UNIVERSITY

01: Data-Dependent Priors
02: Applications
03: Recall: Coefficient Priors
04: Example: Neurons
05: Example: Independent Sources
06: Example: Kernel Expansions
07: Convergence to Gaussian Processes
08: Proof
09: Posterior
10: The RVM Idea
11: Example: Gamma-Hyperprior
12: Example: Normal-Hyperprior
13: General Hyperpriors
14: More Examples
15: Practical Problem: Inference

Data-Dependent Priors

Problem

We are wasting information if we ignore the training patterns in specifying our prior.

Solution: Revisiting Bayes' Rule

$$P(f|X,Y) = \frac{P(Y|f,X)P(f|X)}{P(Y|X)}$$

This means that we already have a **data dependent prior**. The problem with data-independence only arose from the standard approximation $p(f|X) = p(f)$.

Note: the same connection applies to densities.

Consequence

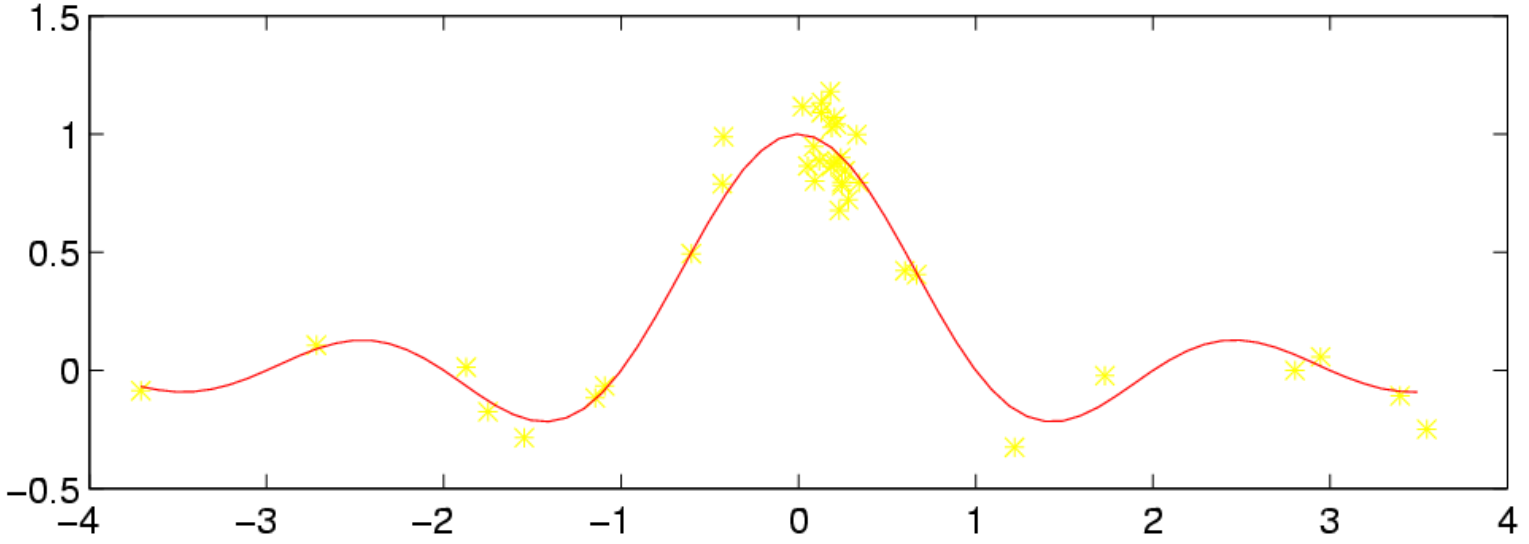
We need to find suitable data-dependent priors which correspond to useful priors over function spaces. If we know $p(X)$, we obviously have

$$p(f) = \int p(f|X)p(X)dX.$$

Examples

Density Dependent Capacity

We can allow for a higher complexity function where we have a large amount of data.



Different Regimes

Data might come from N different sources, which can be distinguished solely based on $x_1, \ldots, x_m$. So, depending on which source, we will switch between priors $p_1(f), \ldots, p_N(f)$.

Recall: Coefficient Priors

THE AUSTRALIAN NATIONAL UNIVERSITY

Function Expansion

Assume that f can be expanded into a linear model of type

$$f(x) = \sum_{i=1}^{M} \alpha_i f_i(x)$$

where $\{f_i(x)\}$ is a suitable set of functions. This could, e.g., be a kernel, i.e., $m = M$ and $f_i = k(x_i, x)$. Note: k **is arbitrary**, e.g., we do not require positivity.

Factorizing Priors

Analogously to a factorizing assumption on the observations we may also assume

$$p(f) = \prod_{i=1}^{m} p(\alpha_i) \text{ where } f = \sum_{i=1}^{m} \alpha_i f_i$$

Motivation

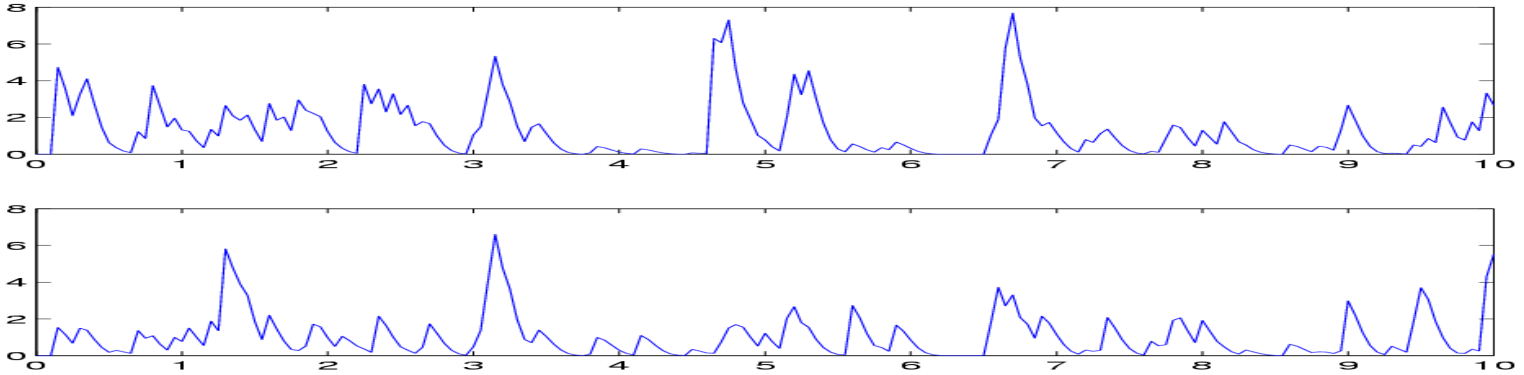
The basis functions f_i correspond to independent "factors" causing the observations, e.g., neurons firing independently but rarely, image elements occurring, etc.

Examples

THE AUSTRALIAN NATIONAL UNIVERSITY

Brain Signals (gross oversimplification)

Neurons fire independently, and very rarely, however, we only observe the signal from several neurons at the same time, possibly several observations with different linear combinations thereof.



Cocktail Party Problem

Assume many speakers, talking (not necessarily to each other) independently. We have many microphones, what is the signal we receive on each microphone? What were the underlying signals?

Example: Kernel Expansions

THE AUSTRALIAN NATIONAL UNIVERSITY

Expansion

$$f(x) = \sum_{i=1}^{m} \alpha_i k(x_i, x) \text{ and } p(f) = \prod_{i=1}^{m} p(\alpha_i)$$

Rationale

- Convenient way of specifying data-dependent prior
- Increases capacity automatically where much data occurs
- Easy to optimize
- Easy to explain (linear model)
- Nice theoretical properties

Examples

$p(\alpha) \propto \exp(-|\alpha|^p)$, $p(\alpha) = \text{BesselK}(0, |\alpha|)$, $p(\alpha) = \frac{1}{s_i}$, ...

Convergence to Gaussian Processes

THE AUSTRALIAN NATIONAL UNIVERSITY

Theorem

- Denote by α_i independent random variables (we do not require identical distributions on α_i) with **unit variance and zero mean**.
- Assume that there exists a **distribution** $p(x)$ **on** $\mathcal{X}$ according to which a sample $\{x_1, \ldots, x_m\}$ is drawn.
- Assume that $k(x, x')$ is **bounded** on $\mathcal{X} \times \mathcal{X}$.

Then the random variable $y(x)$ given by

$$y(x) = \frac{1}{m} \sum_{i=1}^{m} \alpha_i k(x_i, x)$$

converges for $m \to \infty$ to a Gaussian process with zero mean and covariance function

$$\tilde{k}(x, x') = \int_{\mathcal{X}} k(x, \bar{x}) k(x', \bar{x}) p(\bar{x}) d\bar{x}.$$

Proof

Normal Distribution of Linear Combinations

We need only check is that $y(x)$ and any linear combination $\sum_j y(x_j)$ (for arbitrary $x'_j \in \mathcal{X}$) converge to a normal distribution. By application of a theorem of Cramér, this is sufficient to prove that $y(x)$ is distributed according to a Gaussian Process.

Computing $y(x)$

The random variable $y(x)$ is a sum of m independent random variables with bounded variance (since $k(x, x')$ is bounded on $\mathcal{X} \times \mathcal{X}$). Therefore in the limit $m \to \infty$, by virtue of the Central Limit Theorem, we have

$$y(x) \sim \mathcal{N}(0, \sigma^2(x)) \text{ for some } \sigma^2(x) \in \mathbb{R}$$

Linear Combinations For arbitrary $x'_j \in \mathcal{X}$, linear combinations of $y(x'_j)$ also have Gaussian distributions since

$$\sum_{j=1}^{n} \beta_i y(x'_j) = \frac{1}{\sqrt{m}} \sum_{i=1}^{m} \alpha_i \sum_{j=1}^{n} \beta_i k(x_i, x'_j).$$

Proof, Part II

THE AUSTRALIAN NATIONAL UNIVERSITY

Central Limit Theorem on Linear Combination

We may apply the Central Limit Theorem to the sum since the inner sum $\sum_{j=1}^{n} \beta_i k(x_i, x'_j)$ is bounded for any x_i. This also implies that $\sum_{j=1}^{n} \beta_j y(x'_j) \sim \mathcal{N}(0, \sigma^2)$ for $m \to \infty$ and some $\sigma^2 \in \mathbb{R}$, which proves that $y(x)$ is distributed according to a Gaussian Process.

Computing an equivalent Gaussian Process

Note that $y(x)$ has zero mean. Thus the covariance function for finite m can be found as expectation with respect to the random variables α_i,

$$E[y(x)y(x')] = E\left[\frac{1}{m}\sum_{i,j=1}^{m} \alpha_i \alpha_j k(x_i, x) k(x_j, x')\right] = \frac{1}{m}\sum_{i=1}^{m} k(x_i, x) k(x_j, x'),$$

since the α_i are independent and have zero mean. This converges to the Riemann

integral over $\mathcal{X}$ with the density $p(x)$ as $m \to \infty$. Thus

$$E[y(x)y(x')] \underset{m\to\infty}{\longrightarrow} \int_{\mathcal{X}} k(x,\bar{x})k(x',\bar{x})p(\bar{x})d\bar{x}.$$

Effective Kernels

THE AUSTRALIAN NATIONAL UNIVERSITY

Example: Linear Kernel

$k(x, x') = \langle x, x' \rangle$ and coefficient-based prior. Here we have

$$\tilde{k}(x, x') = \int_{\mathcal{X}} k(x, \bar{x}) k(x', \bar{x}) p(\bar{x}) d\bar{x} = x^\top \left(\int \bar{x}\bar{x}^\top p(\bar{x}) d\bar{x} \right) x' = x^\top \left(\mathrm{Cov}[x] \right) x'.$$

Example: Gaussian Kernel

For a kernel $k(x, x') = \exp(-\frac{1}{2}\|x - x'\|^2)$ and $p(x) = (2\pi)^{-\frac{1}{2}} \exp(-\frac{1}{2}x^2)$ we obtain for $\tilde{k}$

$$\tilde{k}(x, x') = \frac{1}{\sqrt{5}} \exp\left(-\frac{3}{5}(x - x')^2 \right) \exp\left(-\frac{2}{5}\langle x, x' \rangle \right)$$

Note

The specific form of $p(\alpha_i)$ is irrelevant for $\tilde{k}$, as long as the variance is bounded (of course, this holds only in the limit).

Consequence

We can look for priors which allow for many zero coefficients α_i.

The RVM Idea

Posterior

For a kernel expansion, the posterior can be found as

$$p(\alpha|X,Y) = \prod_{i=1}^{m} p(y_i|f(x_i))p(\alpha_i) \text{ where } f(x) = \sum_{i=1}^{m} \alpha_i k(x_i, x).$$

Problem

For rather arbitrary priors, this is a difficult optimization problem. We would rather like to have a Gaussian prior ...

Idea

Rewrite $p(\alpha)$ as $\int p(\alpha|s)p(s)ds$, i.e., by means of a Hyperparameter (and optimize via MAP2).

Result

$$p(\alpha|X,Y) = \int_{\mathbb{R}^m} \prod_{i=1}^{m} p(y_i|f(x_i))p(\alpha_i|s_i)ds_1 \dots ds_m$$

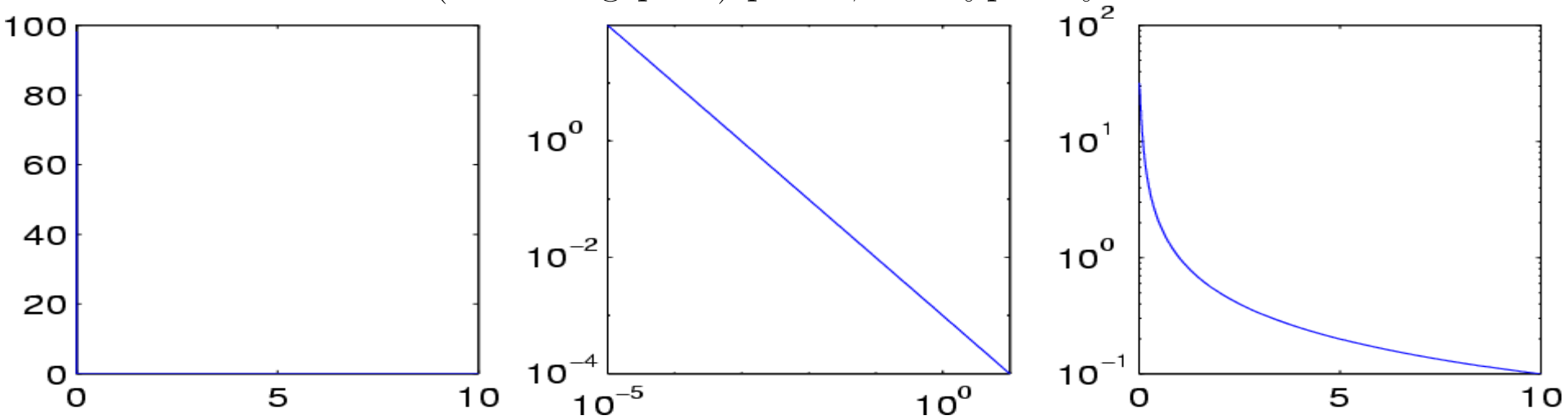
Example: Gamma-Hyperprior

THE AUSTRALIAN NATIONAL UNIVERSITY

Gamma Distribution

$$p(s) = \Gamma(s|a,b) := \frac{s^{a-1}b^a \exp(-sb)}{\Gamma(a)} \text{ for } s_i > 0.$$

For non-informative (flat in logspace) priors, one typically chooses $a = b = 10^{-4}$.



Effective Prior

For the normal prior $p(\alpha|s) = \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{1}{2s^2}\alpha^2\right)$ we have

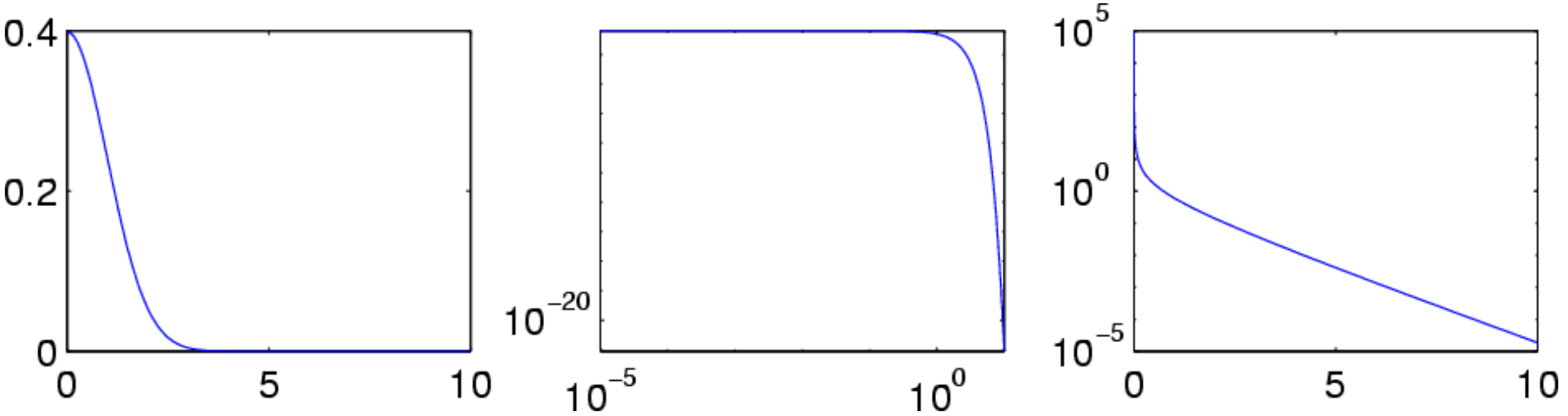
$$p(\alpha) = \int \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{1}{2s^2}\alpha^2\right) \frac{s_i^{a-1}b^a \exp(-s_i b)}{\Gamma(a)} ds = \exp\left(-(a+1/2)\ln\left(b+\frac{\alpha^2}{2}\right)\right)$$

Example: Normal-Hyperprior

THE AUSTRALIAN NATIONAL UNIVERSITY

Gamma Distribution

$$p(s) = \frac{1}{\sqrt{2\pi\omega^2}} \exp\left(-\frac{1}{2\omega^2}s^2\right)$$



Effective Prior

For the normal prior $p(\alpha|s) = \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{1}{2s^2}\alpha^2\right)$ we have

$$p(\alpha) = \int \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{1}{2s^2}\alpha^2\right) \frac{1}{\sqrt{2\pi\omega^2}} \exp\left(-\frac{1}{2\omega^2}s^2\right) ds \propto \text{BesselK}(0, |\omega|).$$

General Hyperpriors

THE AUSTRALIAN NATIONAL UNIVERSITY

Problem

How can we find a suitable hyperprior $p(s)$ for a given $p(\alpha)$ such that

$$p(\alpha) = \int p(\alpha|s)p(s)ds = \int \frac{1}{\sqrt{2\pi s^2}} \exp\left(\frac{1}{2s^2}\alpha^2\right) p(s)ds$$

Solution (after Girosi, 1991)

Parameter transformation $\beta = \frac{1}{2\omega^2}$ leads to

$$p(\alpha) = \int \exp\left(-\beta\alpha\right) \left[\frac{1}{\sqrt{8\pi}\beta} p\left(\frac{1}{\sqrt{2\beta}}\right)\right] d\beta$$

That is, $p(\alpha)$ is the **Laplace Transform** of $\left[\frac{1}{\sqrt{8\pi}\beta} p\left(\frac{1}{\sqrt{2\beta}}\right)\right]$.

Strategy

Given $p(\alpha)$ we only need to find its **inverse** Laplace Transform $\mathcal{L}^{-1}p$ to obtain $p(s)$.

More Examples

Polynomial Priors

For $p(\alpha) = \exp(-|\alpha|^{-a})$ for $a > 1$ we have

$$\left[\mathcal{L}^{-1}p\right](s) = \frac{s^{a-1}}{\Gamma(a)} \text{ hence } p(s) = \sqrt{2\pi}\,\frac{2^{1-a}}{\Gamma(a)}\omega^{-2a}$$

Consequence

- We can deal quite conveniently with priors which do not lead to a lower-bounded optimization problem.
- Large a leads to priors highly peaked at 0 (hence a very sparse code).
- For $a > 1.5$ the variance of s is bounded, hence we get a limiting Gaussian Process.
- For more examples see Bronstein & Semendjajev, Abramovitz & Stegun, etc.

Practical Problem: Inference

THE AUSTRALIAN NATIONAL UNIVERSITY

MAP2 Approximation

Instead of computing the integral over m hyperparameters, we approximate by maximizing

$$p(\alpha, s|X, Y) \propto \prod_{i=1}^{m} \underbrace{p(y_i|f(x_i))}_{\text{Likelihood}} \underbrace{p(\alpha_i|s_i)}_{\text{Prior}} \underbrace{p(s_i)}_{\text{Hyperprior}}$$

Simple Coordinate Descent Algorithm

Step 1 For fixed s minimize $p(\alpha, s|X, Y)$ with respect to α.

Step 2 For fixed α minimize $p(\alpha, s|X, Y)$ with respect to s.

Repeat until a (local) minimum has been obtained.

Confidence

For fixed s we study $p(s|X, Y)$, which gives the error bars for regressions (for classification we already have conditional probabilities).

Regression with Gaussian Noise

THE AUSTRALIAN NATIONAL UNIVERSITY

Likelihood

For fixed s, we have additive normal noise in the observations, i.e., We assume that $p(y_i|f(x_i)) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(y_i - [K\alpha]_i)^2\right)$.

Prior

Furthermore we have $\alpha \sim \mathcal{N}(0, S)$, where $S = \mathrm{diag}(s_1^2, \ldots, s_m^2)$.

Posterior

Since both prior and likelihood are normal, we may find $p(\alpha|X, Y, s)$ as

$$\alpha \sim \mathcal{N}(\mu, \Sigma) \text{ where } \Sigma = (\sigma^{-2}K^\top K + A)^{-1} \text{ and } \mu = \sigma^{-2}\Sigma K^\top \mathbf{y}$$

Prediction

By assumption we have $f(x) = \sum_{i=1}^m \alpha_i k(x_i, x) + \xi = \mathbf{k}^\top \alpha + \xi$. This leads to

$$y(x) \sim \mathcal{N}(\mu^*, \sigma^{*2}) \text{ where } \mu^* = \mathbf{k}^\top \alpha \text{ and } \sigma^{*2} = \sigma^* + \mathbf{k}^\top \Sigma \mathbf{k}.$$

Optimization

Effective Likelihood

By integrating out α we can contract the posterior into $p(Y|X,s)p(s)$, where

$$p(Y|X,s) = \int p(Y|X,\alpha)p(\alpha|s).$$

Since we have only normal distributions, this leads to

$$\mathbf{y} \sim \mathcal{N}(0, (\sigma^2\mathbf{1} + KS^{-1}K^\top))$$

MAP Approximation

Maximize $p(Y|X,s)p(s)$ with respect to s, σ^2:

$$\underset{s,\sigma^2}{\text{maximize}}\ (2\pi)^{\frac{m}{2}}|\sigma^2\mathbf{1} + KS^{-1}K|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\mathbf{y}^\top(\sigma^2\mathbf{1} + KS^{-1}K)^{-1}\mathbf{y}\right) p(s).$$

To find the optimal solution, we take derivatives with respect to s, σ^2 and minimize (details are tedious and omitted, see Tipping 2001).

General Case

Non-Gaussian Likelihood

Minimization of the negative log-posterior cannot be carried out explicitly any more.

Laplace Approximation

A quadratic approximation at the minimum can be used to obtain approximate confidence intervals (we approximate three times: MAP, MAP2, Laplace Approximation).

Practical Solution

Newton method or Fisher Scoring (compute the expectation of the Hessian) leads to rapid convergence.

Classification

Completely analogous to GP Classification.