STA 4273H: Statistical Machine Learning

Russ Salakhutdinov

Department of Statistics
rsalakhu@utstat.toronto.edu
http://www.cs.toronto.edu/~rsalakhu/

Lecture 6

Three Approaches to Classification

- Construct a discriminant function that directly maps each input vector to a specific class.
- Model the conditional probability distribution $p(\mathcal{C}_k|\mathbf{x})$, and then use this distribution to make optimal decisions.
- There are two approaches:
  - Discriminative Approach: Model $p(\mathcal{C}_k|\mathbf{x})$, directly, for example by representing them as parametric models, and optimize for parameters using the training set (e.g. logistic regression).
  - Generative Approach: Model class conditional densities $p(\mathbf{x}|\mathcal{C}_k)$ together with the prior probabilities $p(\mathcal{C}_k)$ for the classes. Infer posterior probability using Bayes' rule:

$$p(\mathcal{C}_k|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})}.$$

We will consider next.

Fixed Basis Functions

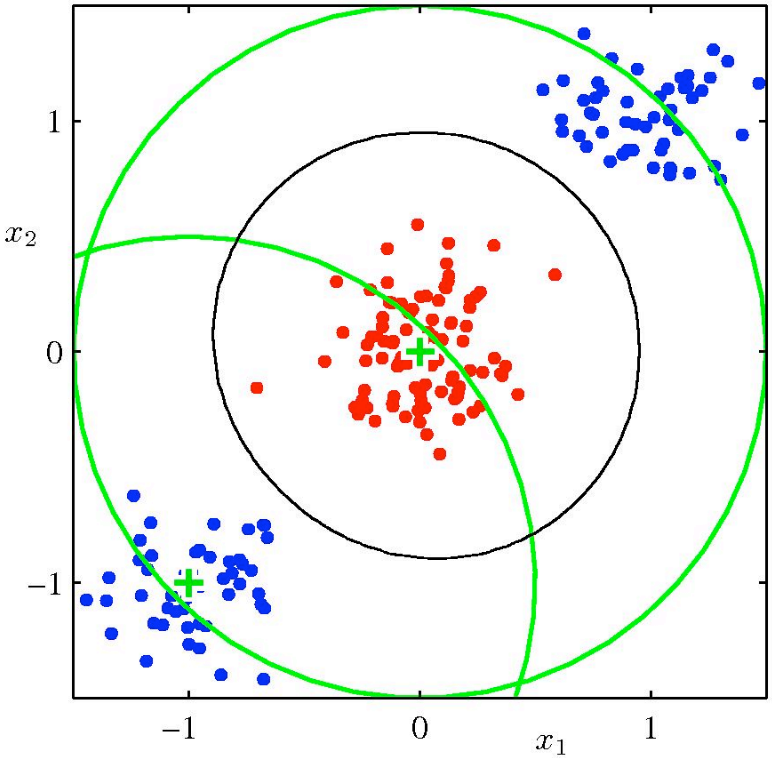
• So far, we have considered classification models that work directly in the input space.

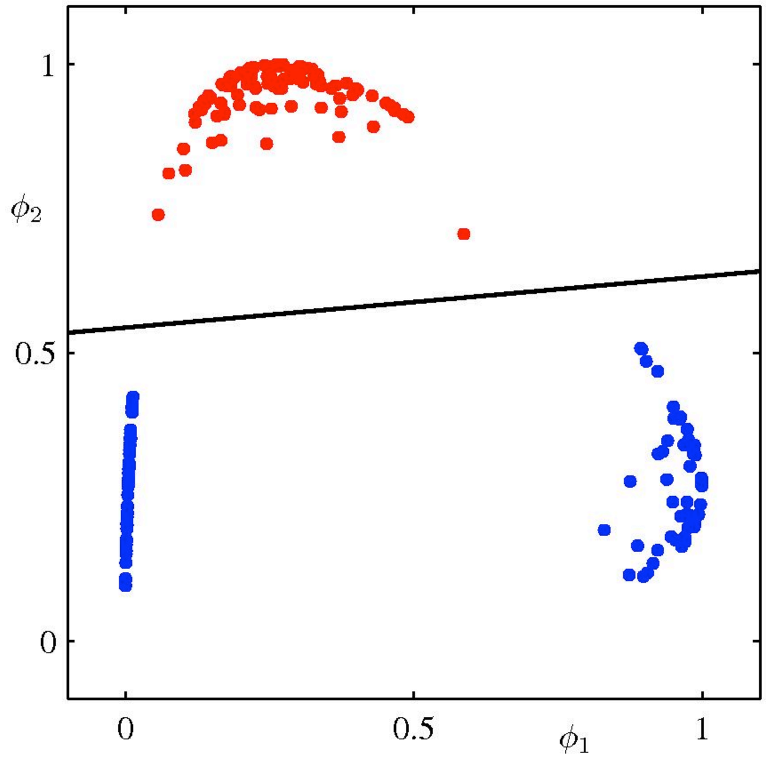
• All considered algorithms are equally applicable if we first make a fixed nonlinear transformation of the input space using vector of basis functions $\phi(\mathbf{x})$.

• Decision boundaries will be linear in the feature space $\phi,$ but would correspond to nonlinear boundaries in the original input space **x**.

• Classes that are linearly separable in the feature space $\phi(\mathbf{x})$ need not be linearly separable in the original input space.

Linear Basis Function Models





• We define two Gaussian basis functions with centers shown by green the crosses, and with contours shown by the green circles.

• Linear decision boundary (right) is obtained using logistic regression, and corresponds to nonlinear decision boundary in the input space (left, black curve).

Logistic Regression

• Consider the problem of two-class classification.

• We have seen that the posterior probability of class C_1 can be written as a logistic sigmoid function:

$$p(\mathcal{C}_1|\mathbf{x}) = \frac{1}{1+\exp(-\mathbf{w}^T\mathbf{x})} = \sigma(\mathbf{w}^T\mathbf{x}),$$

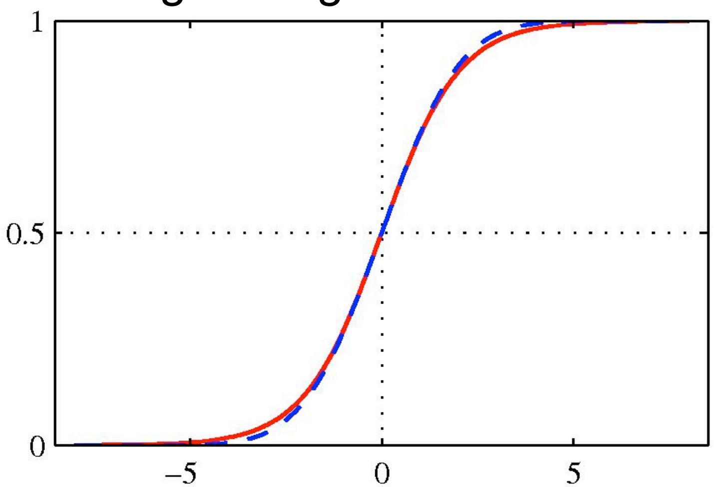
where $p(\mathcal{C}_2|\mathbf{x}) = 1 - p(\mathcal{C}_1|\mathbf{x}),$ and we omit the bias term for clarity.

• This model is known as logistic regression (although this is a model for classification rather than regression).

Note that for generative models, we would first determine the class conditional densities and class-specific priors, and then use Bayes' rule to obtain the posterior probabilities.

Here we model $p(\mathcal{C}_k|\mathbf{x})$ directly.

logistic sigmoid function



ML for Logistic Regression

- We observed a training dataset $\{\mathbf{x}_n, t_n\},\ n = 1, .., N;\ t_n \in \{0, 1\}$.
- Maximize the probability of getting the label right, so the likelihood function takes form:

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}) = \prod_{n=1}^{N} \left[y_n^{t_n} (1 - y_n)^{1-t_n} \right], \quad y_n = \sigma(\mathbf{w}^T \mathbf{x}_n).$$

- Taking the negative log of the likelihood, we can define cross-entropy error function (that we want to minimize):

$$E(\mathbf{w}) = -\ln p(\mathbf{t}|\mathbf{X}, \mathbf{w}) = -\sum_{n=1}^{N} \left[t_n \ln y_n + (1 - t_n) \ln(1 - y_n) \right] = \sum_{n=1}^{N} E_n.$$

- Differentiating and using the chain rule:

$$\frac{\mathrm{d}}{\mathrm{d}y_n} E_n = \frac{y_n - t_n}{y_n(1 - y_n)}, \quad \frac{\mathrm{d}}{\mathrm{d}\mathbf{w}} y_n = y_n(1 - y_n)\mathbf{x}_n, \quad \boxed{\frac{\mathrm{d}}{\mathrm{d}a}\sigma(a) = \sigma(a)(1 - \sigma(a)).}$$

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{w}} E_n = \frac{\mathrm{d}E_n}{\mathrm{d}y_n} \frac{\mathrm{d}y_n}{\mathrm{d}\mathbf{w}} = (y_n - t_n)\mathbf{x}_n.$$

- Note that the factor involving the derivative of the logistic function cancelled.

ML for Logistic Regression

- We therefore obtain:

$$\nabla E(\mathbf{w}) = \sum_{n=1}^{N} (y_n - t_n)\mathbf{x}_n.$$

prediction target

- This takes exactly the same form as the gradient of the sum-of-squares error function for the linear regression model.
- Unlike in linear regression, there is no closed form solution, due to nonlinearity of the logistic sigmoid function.
- The error function is convex and can be optimized using standard gradient-based (or more advanced) optimization techniques.
- Easy to adapt to the online learning setting.

Multiclass Logistic Regression

- For the multiclass case, we represent posterior probabilities by a softmax transformation of linear functions of input variables :

$$p(\mathcal{C}_k|\mathbf{x}) = y_k(\mathbf{x}) = \frac{\exp(\mathbf{w}_k^T \mathbf{x})}{\sum_j \exp(\mathbf{w}_j^T \mathbf{x})}.$$

- Unlike in generative models, here we will use maximum likelihood to determine parameters of this discriminative model directly.

- As usual, we observed a dataset $\{\mathbf{x}_n, t_n\},\ n = 1, .., N,$ where we use 1-of-K encoding for the target vector $\mathbf{t}_n$.

- So if $\mathbf{x}_n$ belongs to class C_k, then **t** is a binary vector of length K containing a single 1 for element k (the correct class) and 0 elsewhere.

- For example, if we have K=5 classes, then an input that belongs to class 2 would be given a target vector:

$$t = (0, 1, 0, 0, 0)^T.$$

Multiclass Logistic Regression

- We can write down the likelihood function:

$$p(\mathbf{T}|\mathbf{X}, \mathbf{w}_1, ..., \mathbf{w}_K) = \prod_{n=1}^{N} \left[\prod_{k=1}^{K} p(\mathcal{C}_k|\mathbf{x}_n)^{t_{nk}} \right] = \prod_{n=1}^{N} \left[\prod_{k=1}^{K} y_{nk}^{t_{nk}} \right]$$

N $\times$ K binary matrix of target variables.

Only one term corresponding to correct class contributes.

where $y_{nk} = p(\mathcal{C}_k|\mathbf{x}_n) = \frac{\exp(\mathbf{w}_k^T \mathbf{x}_n)}{\sum_j \exp(\mathbf{w}_j^T \mathbf{x}_n)}$.

- Taking the negative logarithm gives the cross-entropy entropy function for multi-class classification problem:

$$E(\mathbf{w}_1, ..., \mathbf{w}_K) = -\ln p(\mathbf{T}|\mathbf{X}, \mathbf{w}_1, ..., \mathbf{w}_K) = -\sum_{n=1}^{N} \left[\sum_{k=1}^{K} t_{nk} \ln y_{nk} \right].$$

- Taking the gradient:

$$\nabla E_{\mathbf{w}_j}(\mathbf{w}_1, ... \mathbf{w}_K) = \sum_{n=1}^{N} (y_{nj} - t_{nj}) \mathbf{x}_n.$$

Special Case of Softmax

- If we consider a softmax function for two classes:

$$p(\mathcal{C}_1|\mathbf{x}) = \frac{\exp(a_1)}{\exp(a_1) + \exp(a_2)} = \frac{1}{1 + \exp(-(a_1 - a_2))} = \sigma(a_1 - a_2).$$

- So the logistic sigmoid is just a special case of the softmax function that avoids using redundant parameters:
  - Adding the same constant to both a_1 and a_2 has no effect.
  - The over-parameterization of the softmax is because probabilities must add up to one.

Recap

- Generative approach: Determine the class conditional densities and class-specific priors, and then use Bayes' rule to obtain the posterior probabilities.
  - Different models can be trained separately on different machines.
  - It is easy to add a new class without retraining all the other classes.

$$p(\mathcal{C}_k|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)}{p(\mathbf{x})}.$$

- Discriminative approach: Train all of the model parameters to maximize the probability of getting the labels right.

Model $p(\mathcal{C}_k|\mathbf{x})$ directly.

Bayesian Logistic Regression

- We next look at the Bayesian treatment of logistic regression.
- For the two-class problem, the likelihood takes form:

$$p(\mathbf{t}|\mathbf{X},\mathbf{w}) = \prod_{n=1}^{N} \left[y_n^{t_n}(1-y_n)^{1-t_n} \right], \quad y_n = \frac{1}{1+\exp(-\mathbf{w}^T\mathbf{x}_n)} = \sigma(\mathbf{w}^T\mathbf{x}_n).$$

- Similar to Bayesian linear regression, we could start with a Gaussian prior:

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_0, \mathbf{S}_0).$$

- However, the posterior distribution

$$p(\mathbf{w}|\mathbf{X},\mathbf{t}) \propto p(\mathbf{t}|\mathbf{X},\mathbf{w})p(\mathbf{w}).$$

is no longer Gaussian, and we cannot analytically integrate over model parameters **w**.

- We need to introduce some approximations.

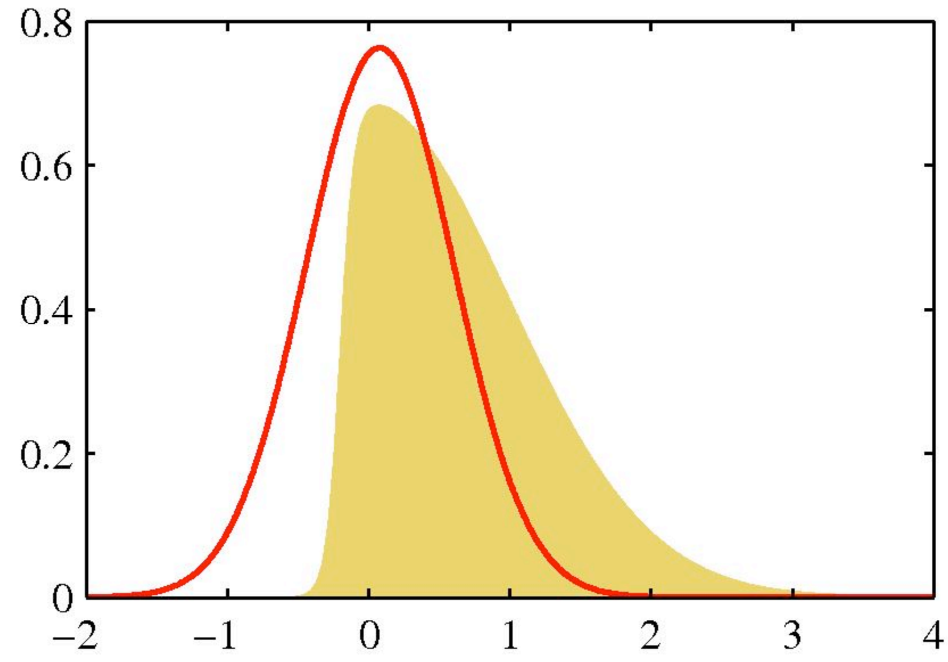
Pictorial illustration

- Consider a simple distribution:

$$p(w) \propto \exp(-w^2)\sigma(20w + 4).$$

- The plot shows the normalized distribution (in yellow), which is not Gaussian.

- The red curve displays the corresponding Gaussian approximation.



Recap: Computational Challenge of Bayesian Framework

Remember: the big challenge is computing the posterior distribution. There are several main approaches:

- Analytical integration: If we use "conjugate" priors, the posterior distribution can be computed analytically (we saw this for Bayesian linear regression).

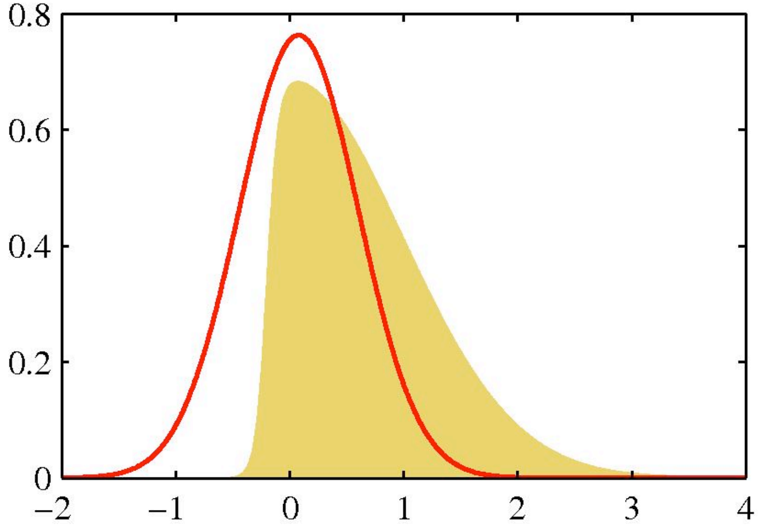
We will consider Laplace approximation next.

- Gaussian (Laplace) approximation: Approximate the posterior distribution with a Gaussian. Works well when there is a lot of data compared to the model complexity (as posterior is close to Gaussian).

- Monte Carlo integration: The dominant current approach is Markov Chain Monte Carlo (MCMC) -- simulate a Markov chain that converges to the posterior distribution. It can be applied to a wide variety of problems.

- Variational approximation: A cleverer way to approximate the posterior. It often works much faster, but not as general as MCMC.

Laplace Approximation



• We will use the following notation:

$$p(\mathbf{z}) = \frac{\tilde{p}(\mathbf{z})}{Z}, \quad Z = \int \tilde{p}(\mathbf{z})\mathrm{d}\mathbf{z}.$$

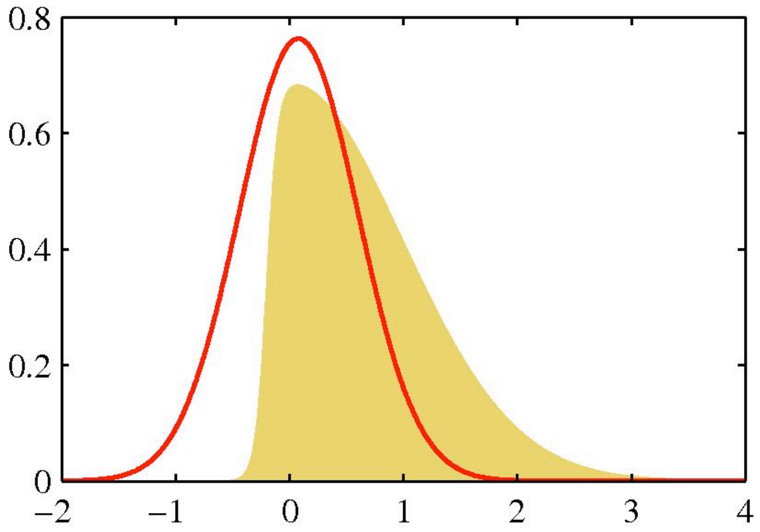
• We can evaluate $\tilde{p}(\mathbf{z})$ point-wise but cannot evaluate Z.

• For example

$$p(\mathbf{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{p(\mathcal{D})}.$$

• Goal: Find a Gaussian approximation q(z) which is centered on a mode of the distribution p(z).

Laplace Approximation



- We will use the following notation:

$$p(\mathbf{z}) = \frac{\tilde{p}(\mathbf{z})}{\mathcal{Z}}, \quad \mathcal{Z} = \int \tilde{p}(\mathbf{z}) \mathrm{d}\mathbf{z}.$$

- At the stationary point $\mathbf{z}_0$, the gradient $\bigtriangledown \tilde{p}(\mathbf{z}_0)$ vanishes.

- Consider a Taylor approximation $\ln \tilde{p}(\mathbf{z})$ around $\mathbf{z}_0$.

$$\ln \tilde{p}(\mathbf{z}) \approx \ln \tilde{p}(\mathbf{z}_0) - \frac{1}{2}(\mathbf{z} - \mathbf{z}_0)^T A (\mathbf{z} - \mathbf{z}_0),$$

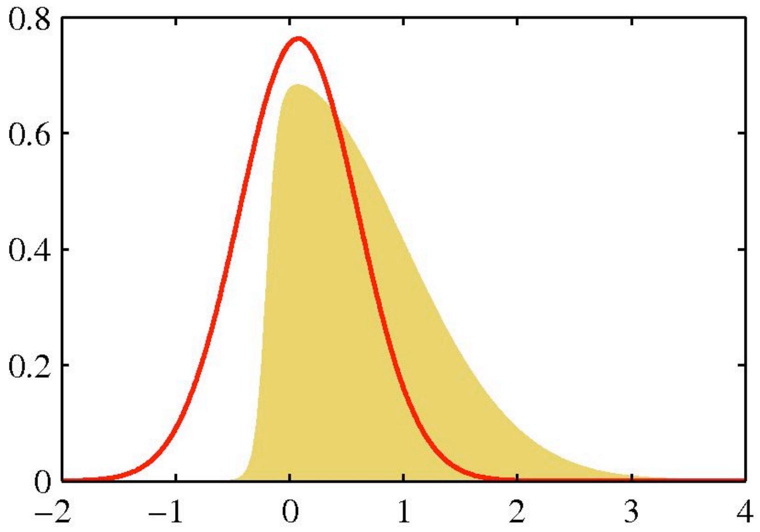
where A is a Hessian matrix:

$$A = - \bigtriangledown \bigtriangledown \ln \tilde{p}(\mathbf{z})|_{\mathbf{z}=\mathbf{z}_0}.$$

- Exponentiating both sides:

$$\tilde{p}(\mathbf{z}) \approx \tilde{p}(\mathbf{z}_0) \exp\left(-\frac{1}{2}(\mathbf{z} - \mathbf{z}_0)^T A (\mathbf{z} - \mathbf{z}_0)\right).$$

Laplace Approximation



- We will use the following notation:

$$p(\mathbf{z}) = \frac{\tilde{p}(\mathbf{z})}{\mathcal{Z}}, \quad \mathcal{Z} = \int \tilde{p}(\mathbf{z}) \mathrm{d}\mathbf{z}.$$

- Using Taylor approximation, we get:

$$\tilde{p}(\mathbf{z}) \approx \tilde{p}(\mathbf{z}_0) \exp\left(-\frac{1}{2}(\mathbf{z} - \mathbf{z}_0)^T A (\mathbf{z} - \mathbf{z}_0)\right).$$

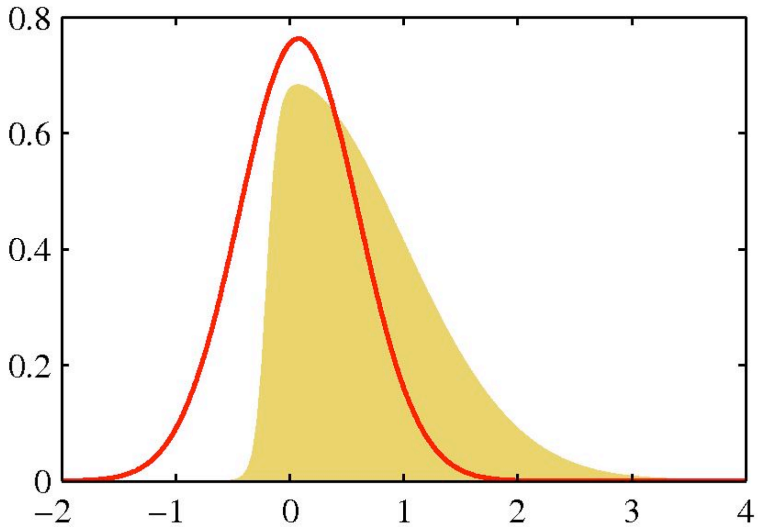
- Hence a Gaussian approximation for $p(\mathbf{z})$ is:

$$q(\mathbf{z}) = \frac{|A|^{1/2}}{(2\pi)^{D/2}} \exp\left(-\frac{1}{2}(\mathbf{z} - \mathbf{z}_0)^T A (\mathbf{z} - \mathbf{z}_0)\right),$$

where $\mathbf{z}_0$ is the mode of $p(\mathbf{z})$, and A is the Hessian:

$$A = - \nabla \nabla \ln \tilde{p}(\mathbf{z})|_{\mathbf{z}=\mathbf{z}_0}.$$

Laplace Approximation



- We will use the following notation:

$$p(\mathbf{z}) = \frac{\tilde{p}(\mathbf{z})}{\mathcal{Z}}, \quad \mathcal{Z} = \int \tilde{p}(\mathbf{z}) \mathrm{d}\mathbf{z}.$$

- Using Taylor approximation, we get:

$$\tilde{p}(\mathbf{z}) \approx \tilde{p}(\mathbf{z}_0) \exp\left(-\frac{1}{2} (\mathbf{z} - \mathbf{z}_0)^T A (\mathbf{z} - \mathbf{z}_0) \right).$$

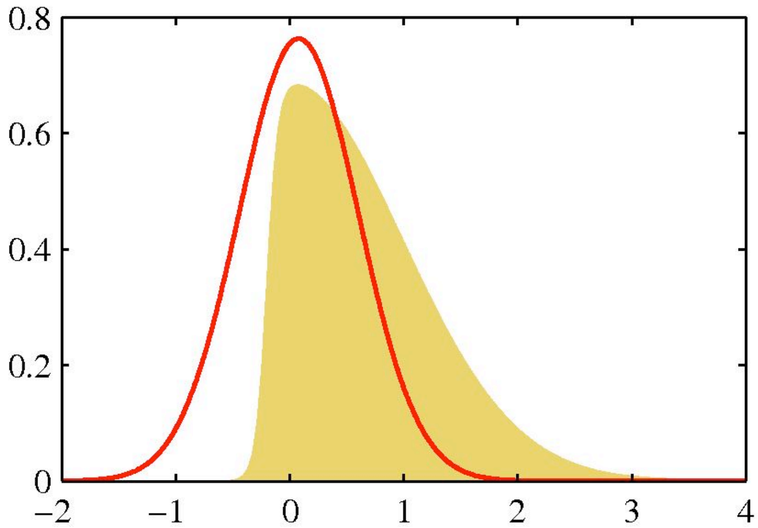
- Bayesian inference: $p(\mathbf{w}|\mathcal{D}) = \dfrac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{p(\mathcal{D})}.$

- Identify: $\tilde{p}(\theta|\mathcal{D}) = p(\mathcal{D}|\theta)p(\theta), \quad \mathcal{Z} = \displaystyle\int p(\mathcal{D}|\theta)p(\theta)\mathrm{d}\theta.$

- The posterior is approximately Gaussian around the MAP estimate:

$$p(\theta|\mathcal{D}) \approx \frac{|A|^{1/2}}{(2\pi)^{D/2}} \exp\left(-\frac{1}{2} (\theta - \theta_{\mathrm{MAP}})^T A (\theta - \theta_{\mathrm{MAP}}) \right).$$

Laplace Approximation



- We will use the following notation:

$$p(\mathbf{z}) = \frac{\tilde{p}(\mathbf{z})}{\mathcal{Z}}, \quad \mathcal{Z} = \int \tilde{p}(\mathbf{z}) \mathrm{d}\mathbf{z}.$$

- Using Taylor approximation, we get:

$$\tilde{p}(\mathbf{z}) \approx \tilde{p}(\mathbf{z}_0) \exp\left(-\frac{1}{2}(\mathbf{z}-\mathbf{z}_0)^T A(\mathbf{z}-\mathbf{z}_0)\right).$$

$$\mathcal{Z} = \int \tilde{p}(\mathbf{z})\mathrm{d}\mathbf{z} \approx \tilde{p}(\mathbf{z}_0) \int \exp\left(-\frac{1}{2}(\mathbf{z}-\mathbf{z}_0)^T A(\mathbf{z}-\mathbf{z}_0)\right) = \tilde{p}(\mathbf{z}_0)\frac{(2\pi)^{D/2}}{|A|^{1/2}}.$$

- We can approximate Model Evidence: $p(\mathcal{D}) = \int p(\mathcal{D}|\theta)P(\theta)\mathrm{d}\theta,$ using Laplace approximation:

$$\ln p(\mathcal{D}) \approx \underbrace{\ln p(\mathcal{D}|\theta_{\mathrm{MAP}})}_{\text{Data fit}} + \underbrace{\ln P(\theta_{\mathrm{MAP}}) + \frac{D}{2}\ln 2\pi - \frac{1}{2}\ln|A|}_{\text{Occam factor: penalize model complexity}}.$$

Bayesian Information Criterion

- BIC can be obtained from the Laplace approximation:

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|\theta_{\mathrm{MAP}}) + \ln P(\theta_{\mathrm{MAP}}) + \frac{D}{2}\ln 2\pi - \frac{1}{2}\ln|A|,$$

by taking the large sample limit (N $\to \infty$) where N is the number of data points.

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|\theta_{\mathrm{MAP}}) - \frac{1}{2}D\ln N.$$

- Quick and easy, does not depend on the prior.
- Can use maximum likelihood estimate instead of the MAP estimate.
- D denotes the number of well-determined parameters.
- **Danger**: Counting parameters can be tricky (e.g. infinite models).

Bayesian Logistic Regression

- Remember the likelihood:

$$p(\mathbf{t}|\mathbf{X},\mathbf{w}) = \prod_{n=1}^{N} \left[y_n^{t_n} (1-y_n)^{1-t_n} \right], \quad y_n = \frac{1}{1+\exp(-\mathbf{w}^T\mathbf{x}_n)} = \sigma(\mathbf{w}^T\mathbf{x}_n).$$

- And the prior: $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_0, \mathbf{S}_0)$.

- The log of the posterior takes form:

Log-prior term

Log-likelihood term

$$\ln p(\mathbf{w}|\mathbf{X},\mathbf{t}) = -\frac{1}{2}(\mathbf{w}-\mathbf{m}_0)^T \mathbf{S}_0^{-1}(\mathbf{w}-\mathbf{m}_0) + \sum_{n=1}^{N} \left[t_n \ln y_n + (1-t_n)\ln(1-t_n) \right] + \text{const.}$$

- We first maximize the log-posterior to get the MAP estimate: $\mathbf{w}_{\text{MAP}}$.
- The inverse of covariance is given by the matrix of second derivatives:

$$\mathbf{S}_N^{-1} = -\bigtriangledown\bigtriangledown \ln p(\mathbf{w}|\mathbf{X},\mathbf{t}) = S_0^{-1} + \sum_n y_n(1-y_n)\mathbf{x}_n\mathbf{x}_n^T.$$

- The Gaussian approximation to the posterior distribution is given by:

$$q(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{w}_{\text{MAP}}, \mathbf{S}_N).$$

Predictive Distribution

• The predictive distribution for class C_1, given a new input $\mathbf{x}^*$ is given by marginalizing with respect to posterior distribution $p(\mathbf{w}|\mathbf{X}, \mathbf{t})$, which is itself approximated by a Gaussian distribution:

$$\begin{aligned} p(\mathcal{C}_1|\mathbf{x}^*, \mathbf{t}, \mathbf{X}) &= \int p(\mathcal{C}_1|\mathbf{x}^*, \mathbf{w}) p(\mathbf{w}|\mathbf{t}, \mathbf{X}) \mathrm{d}\mathbf{w} \\ &\approx \int \sigma(\mathbf{w}^T \mathbf{x}^*) q(\mathbf{w}) \mathrm{d}\mathbf{w}, \end{aligned}$$

Still not tractable.

with the corresponding probability for class C_2 given by:

$$p(\mathcal{C}_1|\mathbf{x}^*, \mathbf{t}, \mathbf{X}) = 1 - p(\mathcal{C}_1|\mathbf{x}^*, \mathbf{t}, \mathbf{X}).$$

• The convolution of Gaussian with logistic sigmoid cannot be evaluated analytically.

Predictive Distribution

$$p(\mathcal{C}_1|\mathbf{x}^*, \mathbf{X}, \mathbf{t}) \approx \int \sigma(\mathbf{w}^T\mathbf{x}^*)q(\mathbf{w})\mathrm{d}\mathbf{w}.$$

• Note that the logistic function depends on **w** only through its projection onto **x***. Denoting $a = \mathbf{w}^T\mathbf{x}^*,$ we have:

$$\sigma(\mathbf{w}^T\mathbf{x}^*) = \int \delta(a - \mathbf{w}^T\mathbf{x}^*)\sigma(a)\mathrm{d}a,$$

where δ is the Dirac delta function. Hence

$$\int \sigma(\mathbf{w}^T\mathbf{x}^*)q(\mathbf{w})\mathrm{d}\mathbf{w} = \int \sigma(a)p(a)\mathrm{d}a, \text{ where } p(a) = \int \delta(a - \mathbf{w}^T\mathbf{x}^*)q(\mathbf{w})\mathrm{d}\mathbf{w}.$$

1-dimensional integral.

• Let us characterize p(a).

• The delta function imposes a linear constraint on **w**. It forms a marginal distribution from the joint q(**w**) by marginalizing out all directions orthogonal to **x***.

• Since q(**w**) is Gaussian, the marginal is also Gaussian.

Predictive Distribution

$$\int \sigma(\mathbf{w}^T\mathbf{x}^*)q(\mathbf{w})\mathrm{d}\mathbf{w} = \int \sigma(a)p(a)\mathrm{d}a, \text{ where } p(a) = \int \delta(a - \mathbf{w}^T\mathbf{x}^*)q(\mathbf{w})\mathrm{d}\mathbf{w}.$$

- We can evaluate the mean and variance of the marginal p(a).

$$\mu_a = \mathbb{E}[a] = \int ap(a)\mathrm{d}a = \int \mathbf{w}^T\mathbf{x}^*q(\mathbf{w})\mathrm{d}\mathbf{w} = \mathbf{w}_{\mathrm{MAP}}^T\mathbf{x}^*.$$

$$\sigma_a^2 = \mathrm{var}[a] = \int p(a)\left[a^2 - \mathbb{E}[a]^2\right] =$$

$$= \int \left[(\mathbf{w}^T\mathbf{x}^*)^2 - (\mathbf{w}_{\mathrm{MAP}}^T\mathbf{x}^*)^2\right]q(\mathbf{w})\mathrm{d}\mathbf{w} = \mathbf{x}^{*T}\mathbf{S}_N\mathbf{x}^*.$$

Same form as the predictive distribution for the Bayesian linear regression model.

- Hence we obtain approximate predictive:

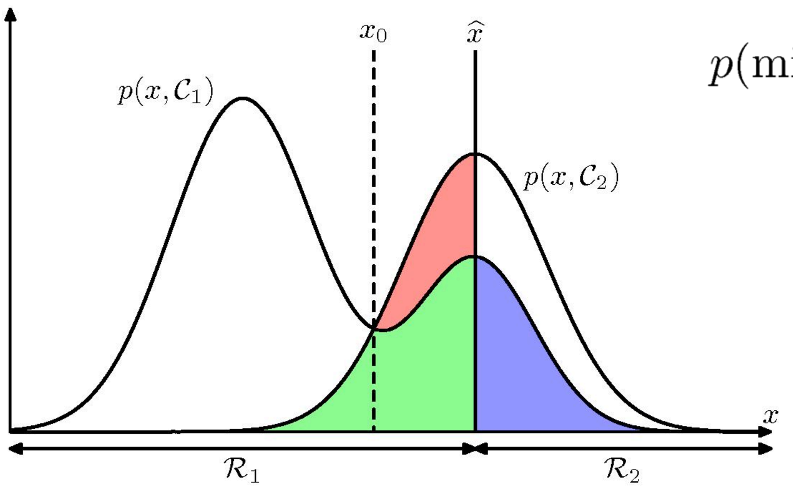
$$p(\mathcal{C}_1|\mathbf{x}^*, \mathbf{X}, \mathbf{t}) \approx \int \sigma(\mathbf{w}^T\mathbf{x}^*)q(\mathbf{w})\mathrm{d}\mathbf{w} = \int \sigma(a)\mathcal{N}(a|\mu_a, \sigma_a^2).$$

- The integral is 1-dimensional and can further be approximated via:

$$\int \sigma(a)\mathcal{N}(a|\mu_a, \sigma_a^2) \approx \sigma(k\mu_a), \quad \text{where } k = (1 + \pi\sigma_a^2/8)^{-1/2}.$$

Midterm Review

- Polynomial curve fitting – generalization, overfitting
- Decision theory:
  - Minimizing misclassification rate / Minimizing the expected loss



$$\begin{aligned} p(\text{mistake}) &= p(\mathbf{x} \in \mathcal{R}_1, \mathcal{C}_2) + p(\mathbf{x} \in \mathcal{R}_2, \mathcal{C}_1) \\ &= \int_{\mathcal{R}_1} p(\mathbf{x}, \mathcal{C}_2)\,\mathrm{d}\mathbf{x} + \int_{\mathcal{R}_2} p(\mathbf{x}, \mathcal{C}_1)\,\mathrm{d}\mathbf{x}. \end{aligned}$$

  - Loss functions for regression

$$\mathbb{E}[L] = \int\int \left(t - y(\mathbf{x})\right)^2 p(\mathbf{x}, t)\mathrm{d}\mathbf{x}\mathrm{d}t.$$

Midterm Review

- Bernoulli, Multinomial random variables (mean, variances)
- Multivariate Gaussian distribution (form, mean, covariance)
- Maximum likelihood estimation for these distributions.

- Exponential family / Maximum likelihood estimation / sufficient statistics for exponential family.

$$p(\mathbf{x}|\boldsymbol{\eta}) = h(\mathbf{x})g(\boldsymbol{\eta}) \exp\left\{\boldsymbol{\eta}^{\mathrm{T}}\mathbf{u}(\mathbf{x})\right\}$$

- Linear basis function models / maximum likelihood and least squares:

$$\ln p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \sum_{i=1}^{N} \ln \mathcal{N}(t_n|\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n), \beta)$$

$$= -\frac{\beta}{2}\sum_{n=1}^{N}\left(t_n - \mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n)\right)^2 + \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi).$$

$$\mathbf{w}_{\mathrm{ML}} = \left(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$$

Midterm Review

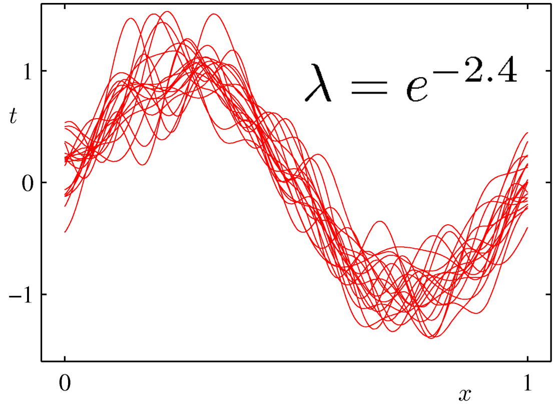
Ridge regression

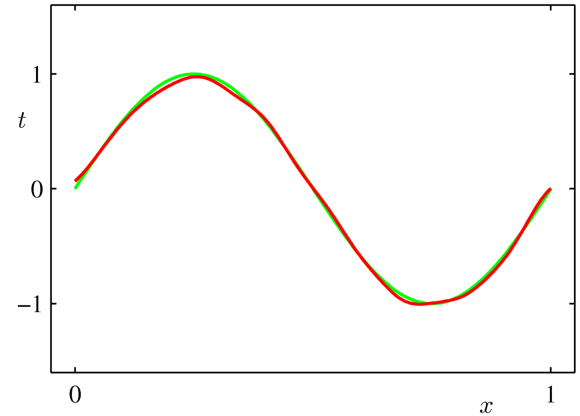
- Regularized least squares:

$$\frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n)\}^2 + \frac{\lambda}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w} \qquad \mathbf{w} = \left(\lambda\mathbf{I} + \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}.$$

- Bias-variance decomposition.

High variance





Low bias

Midterm Review

- Bayesian Inference: likelihood, prior, posterior:

$$p(\mathbf{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbf{w})P(\mathbf{w})}{P(\mathcal{D})}$$

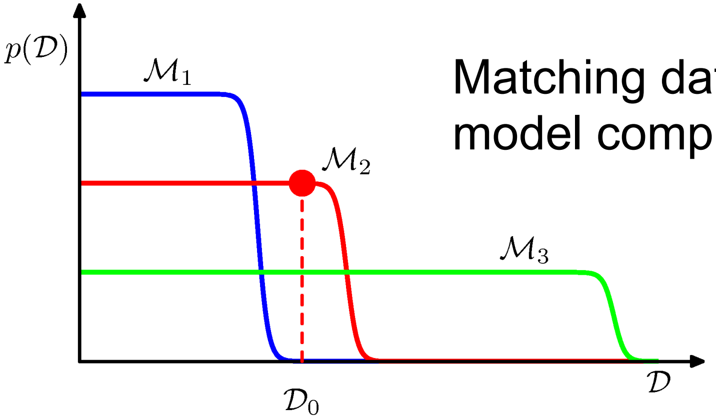
Marginal likelihood (normalizing constant):

- Marginal likelihood / predictive distribution.

$$P(\mathcal{D}) = \int p(\mathcal{D}|\mathbf{w})P(\mathbf{w})\mathrm{d}\mathbf{w}$$

- Bayesian linear regression / parameter estimation / posterior distribution / predictive distribution

- Bayesian model comparison / Evidence approximation



Matching data and model complexity

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|\mathbf{w}_{\mathrm{MAP}}) + M \ln\left(\frac{\Delta w_{\mathrm{posterior}}}{\Delta w_{\mathrm{prior}}}\right).$$

Midterm Review

- Classification models:
  - Discriminant functions
  - Fisher's linear discriminant
  - Perceptron algorithm

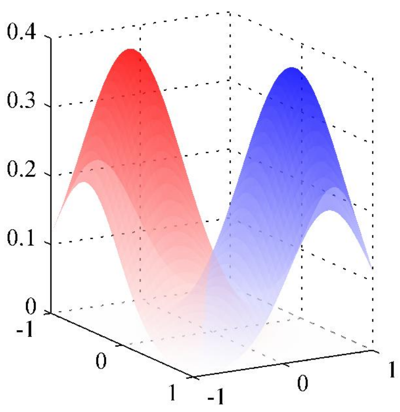
- Probabilistic Generative Models / Gaussian class conditionals / Maximum likelihood estimation:

$$p(\mathbf{x}|\mathcal{C}_k) = \frac{1}{(2\pi)^{D/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)\right).$$

$$p(\mathcal{C}_k|\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x} + w_0),$$

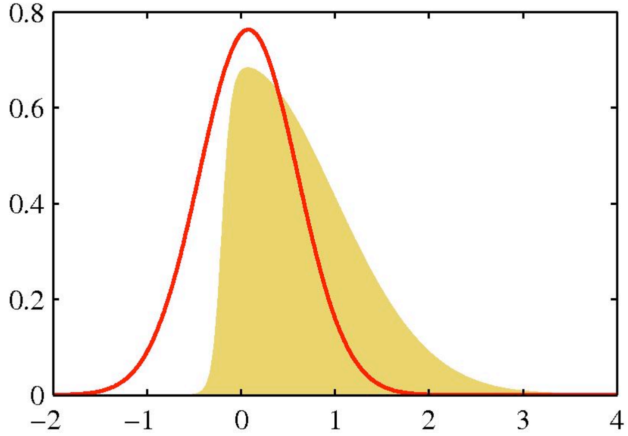
$$\mathbf{w} = \mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2),$$

$$w_0 = -\frac{1}{2}\boldsymbol{\mu}_1^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_2^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_2 + \ln\frac{p(\mathcal{C}_1)}{p(\mathcal{C}_2)}.$$



Midterm Review

- Discriminative Models / Logistic regression / maximum likelihood estimation



- Laplace approximation

- BIC

$$\ln p(\mathcal{D}) \approx \ln p(\mathcal{D}|\theta_{\text{MAP}}) + \ln P(\theta_{\text{MAP}}) + \frac{D}{2}\ln 2\pi - \frac{1}{2}\ln|A|,$$

- Bayesian logistic regression / predictive distribution