CSCI 590: Machine Learning

Lecture 8:
Regularized least squares, Bayesian linear regression, Bayesian model comparison
Instructor: Murat Dundar

Acknowledgement:
Some of these slides are taken from course textbook website
http://research.microsoft.com/~cmbishop/prml/
Assume observations from a deterministic function with added Gaussian noise:

\[ t = y(x, w) + \epsilon \quad \text{where} \quad p(\epsilon|\beta) = \mathcal{N}(\epsilon|0, \beta^{-1}) \]

which is the same as saying,

\[ p(t|x, w, \beta) = \mathcal{N}(t|y(x, w), \beta^{-1}). \]

Given observed inputs, \( \mathbf{X} = \{x_1, \ldots, x_N\} \), and targets, \( \mathbf{t} = [t_1, \ldots, t_N]^T \), we obtain the likelihood function

\[ p(\mathbf{t}|\mathbf{X}, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|w^T \phi(x_n), \beta^{-1}). \]
Taking the logarithm, we get

$$\ln p(t | w, \beta) = \sum_{n=1}^{N} \ln \mathcal{N}(t_n | w^T \phi(x_n), \beta^{-1})$$

$$= \frac{N}{2} \ln \beta - \frac{N}{2} \ln(2\pi) - \beta E_D(w)$$

where

$$E_D(w) = \frac{1}{2} \sum_{n=1}^{N} \{ t_n - w^T \phi(x_n) \}^2$$

is the sum-of-squares error.
Maximum Likelihood and Least Squares (3)

Computing the gradient and setting it to zero yields

\[ \nabla_w \ln p(t|w, \beta) = \beta \sum_{n=1}^{N} \left\{ t_n - w^T \phi(x_n) \right\} \phi(x_n)^T = 0. \]

Solving for \( w \), we get

\[ w_{ML} = \left( \Phi^T \Phi \right)^{-1} \Phi^T t \]

where

\[ \Phi = \begin{pmatrix} 
\phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_{M-1}(x_1) \\
\phi_0(x_2) & \phi_1(x_2) & \cdots & \phi_{M-1}(x_2) \\
\vdots & \vdots & \ddots & \vdots \\
\phi_0(x_N) & \phi_1(x_N) & \cdots & \phi_{M-1}(x_N) 
\end{pmatrix}. \]

The Moore-Penrose pseudo-inverse, \( \Phi^+ \).
Consider

\[ y = \Phi w_{\text{ML}} = [\varphi_1, \ldots, \varphi_M] w_{\text{ML}}. \]

\[ y \in S \subseteq T \quad t \in T \]

\( S \) is spanned by \( \varphi_1, \ldots, \varphi_M \).

\( w_{\text{ML}} \) minimizes the distance between \( t \) and its orthogonal projection on \( S \), i.e. \( y \).
Consider the error function:

\[ E_D(w) + \lambda E_W(w) \]

Data term + Regularization term

With the sum-of-squares error function and a quadratic regularizer, we get

\[
\frac{1}{2} \sum_{n=1}^{N} \left( t_n - w^T \phi(x_n) \right)^2 + \frac{\lambda}{2} w^T w
\]

which is minimized by

\[
w = \left( \lambda I + \Phi^T \Phi \right)^{-1} \Phi^T t.
\]

\( \lambda \) is called the regularization coefficient.
Regularized Least Squares (2)

With a more general regularizer, we have

\[ \frac{1}{2} \sum_{n=1}^{N} \{t_n - w^T \phi(x_n)\}^2 + \frac{\lambda}{2} \sum_{j=1}^{M} |w_j|^q \]

\[ q = 0.5 \quad q = 1 \quad q = 2 \quad q = 4 \]

Lasso \hspace{2cm} \text{Quadratic}
Regularized Least Squares (3)

Lasso tends to generate sparser solutions than a quadratic regularizer.
Multiple Outputs (1)

Analogously to the single output case we have:

\[ p(t|x, W, \beta) = \mathcal{N}(t|y(W, x), \beta^{-1}I) \]
\[ = \mathcal{N}(t|W^T \phi(x), \beta^{-1}I). \]

Given observed inputs, \( X = \{x_1, \ldots, x_N\} \), and targets, \( T = [t_1, \ldots, t_N]^T \), we obtain the log likelihood function

\[
\ln p(T|X, W, \beta) = \sum_{n=1}^{N} \ln \mathcal{N}(t_n|W^T \phi(x_n), \beta^{-1}I)
\]
\[ = \frac{NK}{2} \ln \left(\frac{\beta}{2\pi}\right) - \frac{\beta}{2} \sum_{n=1}^{N} \|t_n - W^T \phi(x_n)\|^2. \]
Maximizing with respect to $\mathbf{W}$, we obtain

$$\mathbf{W}_{\text{ML}} = \left( \Phi^T \Phi \right)^{-1} \Phi^T \mathbf{T}.$$  

If we consider a single target variable, $t_k$, we see that

$$\mathbf{w}_k = \left( \Phi^T \Phi \right)^{-1} \Phi^T t_k = \Phi^\dagger t_k$$

where $t_k = [t_{1k}, \ldots, t_{Nk}]^T$, which is identical with the single output case.
The Bias-Variance Decomposition (1)

Recall the *expected squared loss*,

\[
\mathbb{E}[L] = \int \{y(x) - h(x)\}^2 p(x) \, dx + \int \int \{h(x) - t\}^2 p(x, t) \, dx \, dt
\]

where

\[
h(x) = \mathbb{E}[t \mid x] = \int t p(t \mid x) \, dt.
\]

The second term of \(\mathbb{E}[L]\) corresponds to the noise inherent in the random variable \(t\).

What about the first term?
The Bias-Variance Decomposition (2)

Suppose we were given multiple data sets, each of size $N$. Any particular data set, $\mathcal{D}$, will give a particular function $y(x; \mathcal{D})$. We then have

$$\left\{ y(x; \mathcal{D}) - h(x) \right\}^2$$

$$= \left\{ y(x; \mathcal{D}) - \mathbb{E}_\mathcal{D}[y(x; \mathcal{D})] + \mathbb{E}_\mathcal{D}[y(x; \mathcal{D})] - h(x) \right\}^2$$

$$= \left\{ y(x; \mathcal{D}) - \mathbb{E}_\mathcal{D}[y(x; \mathcal{D})] \right\}^2 + \left\{ \mathbb{E}_\mathcal{D}[y(x; \mathcal{D})] - h(x) \right\}^2$$

$$+ 2\left\{ y(x; \mathcal{D}) - \mathbb{E}_\mathcal{D}[y(x; \mathcal{D})] \right\}\left\{ \mathbb{E}_\mathcal{D}[y(x; \mathcal{D})] - h(x) \right\}. $$
The Bias-Variance Decomposition (3)

Taking the expectation over $\mathcal{D}$ yields

$$
\mathbb{E}_\mathcal{D} \left[ \left\{ y(x; \mathcal{D}) - h(x) \right\}^2 \right] = \left\{ \mathbb{E}_\mathcal{D}[y(x; \mathcal{D})] - h(x) \right\}^2 + \mathbb{E}_\mathcal{D} \left[ \left\{ y(x; \mathcal{D}) - \mathbb{E}_\mathcal{D}[y(x; \mathcal{D})] \right\}^2 \right].
$$

(bias)$^2$ \hspace{1cm} variance
The Bias-Variance Decomposition (4)

Thus we can write

\[
\text{expected loss} = (\text{bias})^2 + \text{variance} + \text{noise}
\]

where

\[
(\text{bias})^2 = \int \left\{ \mathbb{E}_D[y(x; D)] - h(x) \right\}^2 p(x) \, dx
\]

\[
\text{variance} = \int \mathbb{E}_D \left[ \{y(x; D) - \mathbb{E}_D[y(x; D)]\}^2 \right] p(x) \, dx
\]

\[
\text{noise} = \iint \{h(x) - t\}^2 p(x, t) \, dx \, dt
\]
The Bias-Variance Decomposition (5)

Example: 25 data sets from the sinusoidal, varying the degree of regularization, $\lambda$. 

\[ \ln \lambda = 2.6 \]
The Bias-Variance Decomposition (6)

Example: 25 data sets from the sinusoidal, varying the degree of regularization, $\lambda$.

\[ \ln \lambda = -0.31 \]
The Bias-Variance Decomposition (7)

Example: 25 data sets from the sinusoidal, varying the degree of regularization, $\lambda$.
The Bias-Variance Trade-off

From these plots, we note that an over-regularized model (large $\lambda$) will have a high bias, while an under-regularized model (small $\lambda$) will have a high variance.
Bayesian Linear Regression (1)

Define a conjugate prior over $\mathbf{w}$

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

Combining this with the likelihood function and using results for marginal and conditional Gaussian distributions, gives the posterior

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

where

$$\mathbf{m}_N = \mathbf{S}_N \left( \mathbf{S}_0^{-1} \mathbf{m}_0 + \beta \Phi^T \mathbf{t} \right)$$

$$\mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \beta \Phi^T \Phi.$$
Bayesian Linear Regression (2)

A common choice for the prior is

\[ p(w) = \mathcal{N}(w | 0, \alpha^{-1}I) \]

for which

\[ m_N = \beta S_N \Phi^T t \]
\[ S_N^{-1} = \alpha I + \beta \Phi^T \Phi. \]

Next we consider an example ...
Bayesian Linear Regression (3)

0 data points observed

Prior

Data Space
Bayesian Linear Regression (4)

1 data point observed
Bayesian Linear Regression (5)

2 data points observed

Likelihood

Posterior

Data Space
Bayesian Linear Regression (6)

20 data points observed
Predictive Distribution (1)

Predict $t$ for new values of $x$ by integrating over $w$:

$$p(t|\mathbf{t}, \alpha, \beta) = \int p(t|\mathbf{w}, \beta)p(\mathbf{w}|\mathbf{t}, \alpha, \beta) \, d\mathbf{w}$$

$$= \mathcal{N}(t|\mathbf{m}_N^T \phi(x), \sigma^2_N(x))$$

where

$$\sigma^2_N(x) = \frac{1}{\beta} + \phi(x)^T \mathbf{S}_N \phi(x).$$
Predictive Distribution (2)

Example: Sinusoidal data, 9 Gaussian basis functions, 1 data point
Predictive Distribution (3)

Example: Sinusoidal data, 9 Gaussian basis functions, 2 data points
Predictive Distribution (4)

Example: Sinusoidal data, 9 Gaussian basis functions, 4 data points
Predictive Distribution (5)

Example: Sinusoidal data, 9 Gaussian basis functions, 25 data points
Equivalent Kernel (1)

The predictive mean can be written

\[
y(x, m_N) = m_N^T \phi(x) = \beta \phi(x)^T S_N \Phi^T t
\]

\[
= \sum_{n=1}^{N} \beta \phi(x)^T S_N \phi(x_n) t_n
\]

\[
= \sum_{n=1}^{N} k(x, x_n) t_n. \quad \text{Equivalent kernel or smoother matrix.}
\]

This is a weighted sum of the training data target values, \( t_n \).
Equivalent Kernel (2)

Weight of $t_n$ depends on distance between $x$ and $x_n$; nearby $x_n$ carry more weight.
Non-local basis functions have local equivalent kernels:

- Polynomial
- Sigmoidal
Equivalent Kernel (4)

The kernel as a covariance function: consider

$$\text{cov}[y(x), y(x')] = \text{cov}[\phi(x)^T w, w^T \phi(x')]$$

$$= \phi(x)^T S_N \phi(x') = \beta^{-1} k(x, x').$$

We can avoid the use of basis functions and define the kernel function directly, leading to *Gaussian Processes* (Chapter 6).
Equivalent Kernel (5)

\[
\sum_{n=1}^{N} k(x, x_n) = 1
\]

for all values of \( x \); however, the equivalent kernel may be negative for some values of \( x \).

Like all kernel functions, the equivalent kernel can be expressed as an inner product:

\[
k(x, z) = \psi(x)^T \psi(z)
\]

where \( \psi(x) = \beta^{1/2} S_N^{1/2} \phi(x) \).
Bayesian Model Comparison (1)

How do we choose the ‘right’ model?
Assume we want to compare models $M_i$, $i=1, \ldots, L$, using data $D$; this requires computing

$$p(M_i | D) \propto p(M_i) p(D | M_i).$$

Posterior Prior $\text{Model evidence or marginal likelihood}$

Bayes Factor: ratio of evidence for two models

$$\frac{p(D | M_i)}{p(D | M_j)}$$
Bayesian Model Comparison (2)

For a model with parameters \( w \), we get the model evidence by marginalizing over \( w \)

\[
p(D|M_i) = \int p(D|w, M_i) p(w|M_i) \, dw.
\]

Note that

\[
p(w|D, M_i) = \frac{p(D|w, M_i)p(w|M_i)}{p(D|M_i)}
\]
Bayesian Model Comparison (4)

For a given model with a single parameter, $w$, consider the approximation

$$p(D) = \int p(D|w)p(w)\,dw$$

$$\simeq p(D|w_{MAP}) \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}}$$

where the posterior is assumed to be sharply peaked.
Bayesian Model Comparison (5)

Taking logarithms, we obtain

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|w_{\text{MAP}}) + \ln \left( \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}} \right).$$

With $M$ parameters, all assumed to have the same ratio $\Delta w_{\text{posterior}}/\Delta w_{\text{prior}},$ we get

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

Negative and linear in $M.$