Chris Bishop’s PRML
Ch. 3: Linear Models of Regression

Mathieu Guillaumin & Radu Horaud

October 25, 2007
An example – polynomial curve fitting – was considered in Ch. 1

A linear combination – regression – of a fixed set of nonlinear functions – basis functions

Supervised learning: $N$ observations $\{x_n\}$ with corresponding target values $\{t_n\}$ are provided. **The goal is to predict $t$ of a new value $x$.**

Construct a function such that $y(x)$ is a prediction of $t$.

Probabilistic perspective: model the predictive distribution $p(t|x)$.
Mathieu Guillaumin & Radu Horaud

Chris Bishop’s PRML Ch. 3: Linear Models of Regression
The chapter section by section

3.1 Linear basis function models
   ▶ Maximum likelihood and least squares
   ▶ Geometry of least squares
   ▶ Sequential learning
   ▶ Regularized least squares

3.2 The bias-variance decomposition

3.3 Bayesian linear regression
   ▶ Parameter distribution
   ▶ Predictive distribution
   ▶ Equivalent kernel

3.4 Bayesian model comparison

3.5 The evidence approximation

3.6 Limitations of fixed basis functions
Linear Basis Function Models

\[ y(x, w) = \sum_{j=0}^{M-1} w_j \phi_j(x) = w^\top \phi(x) \]

where:

- \( w = (w_0, \ldots, w_{M-1})^\top \) and \( \phi = (\phi_0, \ldots, \phi_{M-1})^\top \) with \( \phi_0(x) = 1 \) and \( w_0 = \) bias parameter.
- In general \( x \in \mathbb{R}^D \) but it will be convenient to treat the case \( x \in \mathbb{R} \).
- We observe the set \( X = \{x_1, \ldots, x_n, \ldots, x_N\} \) with corresponding target variables \( t = \{t_n\} \).
Basis function choices

- **Polynomial**
  \[ \phi_j(x) = x^j \]

- **Gaussian**
  \[ \phi_j(x) = \exp \left( -\frac{(x - \mu_j)^2}{2s^2} \right) \]

- **Sigmoidal**
  \[ \phi_j(x) = \sigma \left( \frac{x - \mu_j}{s} \right) \text{ with } \sigma(a) = \frac{1}{1 + e^{-a}} \]

- splines, Fourier, wavelets, etc.
Examples of basis functions
Maximum likelihood and least squares

\[ t = y(x, w) + \epsilon \]

For a i.i.d. data set we have the likelihood function:

\[ p(t|X, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|w^\top \phi(x_n), \beta^{-1}) \]

We can use the machinery of MLE to estimate the parameters \( w \) and the precision \( \beta \):

\[ w_{ML} = (\Phi^\top \Phi)^{-1} \Phi^\top t \text{ with } \Phi_{M \times N} = [\phi_{mn}(x_n)] \]

and:

\[ \beta_{ML}^{-1} = \frac{1}{N} \sum_{n=1}^{N} (t_n - w_{ML}^\top \phi(x_n))^2 \]
Geometry of least squares

\[ S \]

\[ \phi_1 \]

\[ \phi_2 \]

Chris Bishop’s PRML Ch. 3: Linear Models of Regression
Sequential learning

Apply a technique known as **stochastic gradient descent** or **sequential gradient descent**, i.e., replace:

\[
E_D(w) = \frac{1}{2} \sum_{n=1}^{N} (t_n - w^\top \phi(x_n))^2
\]

with (\(\eta\) is a learning rate parameter):

\[
w^{(\tau+1)} = w^{(\tau)} + \eta \left( t_n - w^{(\tau)}^\top \phi(x_n) \right) \frac{\phi(x_n)}{\nabla E_n} \tag{3.23}
\]
Regularized least squares

The total error function:

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

$$w = \left( \lambda I + \Phi^\top \Phi \right)^{-1} \Phi^\top t$$

Regularization has the advantage of limiting the model complexity (the appropriate number of basis functions). This is replaced with the problem of finding a suitable value of the regularization coefficient $\lambda$. 
The Bias-Variance Decomposition

- Over-fitting occurs whenever the number of basis functions is large and with training data sets of limited size.
- Limiting the number of basis functions limits the flexibility of the model.
- Regularization can control over-fitting but raises the question of how to determine $\lambda$.
- The **bias-variance tradeoff** is a frequentist viewpoint of model complexity.
The regression loss-function: \( L(t, y(x)) = (y(x) - t)^2 \)

The decision problem = minimize the expected loss:

\[
E[L] = \int \int (y(x) - t)^2 p(x, t) \, dx \, dt
\]

Solution: \( y(x) = \int t p(t|x) \, dt = E_t[t|x] \)

this is known as the regression function

conditional average of \( t \) conditioned on \( x \), e.g., figure 1.28, page 47

Another expression for the expectation of the loss function:

\[
E[L] = \int (y(x) - E[t|x])^2 p(x) \, dx + \int (E[t|x] - t)^2 p(x) \, dx.
\]

\( 1.90 \)
The optimal prediction is obtained by minimization of the expected squared loss function:

\[
h(x) = E[t|x] = \int tp(t|x)dt
\]  

(3.36)

The expected squared loss can be decomposed into two terms:

\[
E[L] = \int (y(x) - h(x))^2 p(x)dx + \int (h(x) - t)^2 p(x, t)dxdt.
\]  

(3.37)

The theoretical minimum of the first term is zero for an appropriate choice of the function \( y(x) \) (for unlimited data and unlimited computing power).

The second term arises from noise in the data and it represents the minimum achievable value of the expected squared loss.
An ensemble of data sets

- For any given data set $D$ we obtain a prediction function $y(x, D)$.
- The performance of a particular algorithm is assessed by taking the average over all these data sets, namely $E_D[L]$. This expands into the following terms:

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

- There is a tradeoff between bias and variance:
  - flexible models have low bias and high variance
  - rigid models have high bias and low variance
- The bias-variance decomposition provides interesting insights in model complexity, it is of limited practical value because several data sets are needed.
Example: $L=100$, $N=25$, $M=25$, Gaussian basis

\[
egin{bmatrix}
0 & 1 \\
-1 & 0 \\
0 & 1 \\
\end{bmatrix}
\]

$\ln \lambda = 2.6$

$\ln \lambda = -2.4$
Bayesian Linear Regression (1/5)

Assume additive gaussian noise with known precision $\beta$.

The likelihood function $p(t|w)$ is the exponential of a quadratic function of $w$, its conjugate prior is Gaussian:

$$p(w) = \mathcal{N}(w|m_0, S_0)$$

Its posterior is also Gaussian (2.116):

$$p(w|t) = \mathcal{N}(w|m_N, S_N) \propto p(t|w)p(w)$$

where

$$m_N = S_N(S_0^{-1}m_0 + \beta\Phi^Tt)$$

$$S_N^{-1} = S_0^{-1} + \beta\Phi^T\Phi$$

Note how this fits a sequential learning framework

The max of a Gaussian is at its mean: $w_{MAP} = m_N$
Bayesian Linear Regression (2/5)

Assume \( p(\mathbf{w}) \) is governed by a hyperparameter \( \alpha \) following a Gaussian law of scalar covariance (i.e. \( \mathbf{m}_0 = 0 \) and \( \mathbf{S}_0 = \alpha^{-1} \mathbf{I} \ )):

\[
p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|0, \alpha^{-1} \mathbf{I}) \quad (3.52)
\]

then

\[
\begin{align*}
\mathbf{m}_N &= \beta \mathbf{S}_N \Phi^T \mathbf{t} \\
\mathbf{S}_N^{-1} &= \alpha \mathbf{I} + \beta \Phi^T \Phi 
\end{align*} \quad (3.53/3.54)
\]

➤ Note \( \alpha \to 0 \) implies \( \mathbf{m}_N \to \mathbf{w}_{ML} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{t} \quad (3.35) \)

Log of posterior is sum of log of likelihood and log of prior:

\[
\ln p(\mathbf{w}|\mathbf{t}) = -\frac{\beta}{2} \sum_{n=1}^{N} \left( t_n - \mathbf{w}^T \phi(\mathbf{x}_n) \right)^2 - \frac{\alpha}{2} \mathbf{w}^T \mathbf{w} + \text{const} \quad (3.55)
\]

which is equivalent to a quadratic regularizer with coeff. \( \alpha/\beta \).
Bayesian Linear Regression (3/5)

In practice, we want to make predictions of \( t \) for new values of \( x \):

\[
p(t|t, \alpha, \beta) = \int p(t|w, \beta)p(w|t, \alpha, \beta)dw
\]  

(3.57)

- Conditional distribution: \( p(t|w, \beta) = \mathcal{N}(t|y(x, w), \beta^{-1}) \)  
- Posterior: \( p(w|t, \alpha, \beta) = \mathcal{N}(w|m_N, S_N) \)  

(3.49) (3.8)

The convolution is a Gaussian (2.115):

\[
p(t|x, t, \alpha, \beta) = \mathcal{N}(t|m_N^T\Phi(x), \sigma^2_N(x))
\]  

(3.58)

where

\[
\sigma^2_N(x) = \beta^{-1} + \Phi(x)^T S_N \Phi(x)
\]

(3.59)
Chris Bishop’s PRML Ch. 3: Linear Models of Regression
Mathieu Guillaumin & Radu Horaud

Chris Bishop’s PRML Ch. 3: Linear Models of Regression
Bayesian Linear Regression (4/5)

\[ y(x, m_N) \text{ rewrites as } \sum_{n=1}^{N} k(x, x_n) t_n \text{ where } \]

\[ k(x, x') = \beta \Phi(x)^T S_N \Phi(x') \]

(3.61-3.62)

Smother matrix, equivalent kernel, linear smoother

The kernel works as a similarity or closeness measure, giving more weight to evidence that is close to the point where we want to make the prediction

- Basis functions \( \leftrightarrow \) kernel duality
- With \( \Psi(x) = \beta^{-1/2} S_N^{1/2} \Phi(x) \), \( k(x, x') = \Psi(x)^T \Psi(x') \) (3.65)
- The kernel sums to one (over the training set)
- \( \text{cov}(y(x), y(x')) = \beta^{-1} k(x, x') \) (3.63)
Bayesian Linear Regression (5/5)

Kernel from Gaussian basis functions

Kernels at $x = 0$ for kernels corresponding (left) to the polynomial basis functions and (right) to the sigmoidal basis functions.
Bayesian Model Comparison (1/2)

The overfitting that appears in ML can be avoided by marginalizing over the model parameters.

- Cross-validation is no more useful
- We can use all the data for better training the model
- We can compare models based on training data alone

\[
p(M_i|D) \propto p(M_i)p(D|M_i) \tag{3.66}
\]

\(p(D|M_i)\): model evidence or marginal likelihood.

Using model selection and assuming the posterior \(p(w|D, M_i)\) is sharply peaked at \(w_{\text{MAP}}\) (single parameter case):

\[
p(D) = \int p(D|w)p(w)dw \approx p(D|w_{\text{MAP}}) \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}} \tag{3.70}
\]
Back to multiple parameters, assuming they share the same $\Delta w$ ratio, the complexity penalty is linear in $M$:

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

(3.72)

About $p(D | M_i)$:

- if $M_i$ is too simple, bad fitting of the data
- if $M_i$ is too complex/powerful, the probability of generating the observed data is washed out
The evidence approximation (1/2)

Fully bayesian treatment would imply marginalizing over hyperparameters and parameters, but this is intractable:

$$p(t|\mathbf{t}) = \int \int \int p(t|\mathbf{w}, \beta)p(\mathbf{w}|\mathbf{t}, \alpha, \beta)p(\alpha, \beta|\mathbf{t})d\mathbf{w}d\alpha d\beta$$  \hspace{1cm} (3.74)

An approximation is found by maximizing the marginal likelihood function $p(\alpha, \beta|\mathbf{t}) \propto p(\mathbf{t}|\alpha, \beta)p(\alpha, \beta)$ to get $(\hat{\alpha}, \hat{\beta})$ (empirical Bayes).

$$\ln p(\mathbf{t}|\alpha, \beta) = \frac{M}{2} \ln \alpha + \frac{N}{2} \ln \beta - E(\mathbf{m}_N) - \frac{1}{2} \ln |\mathbf{S}_N^{-1}| - \frac{N}{2} \ln(2\pi)$$  \hspace{1cm} (3.77 \rightarrow 3.86)

Assuming $p(\alpha, \beta|\mathbf{t})$ is highly peaked at $(\hat{\alpha}, \hat{\beta})$:

$$p(t|\mathbf{t}) \simeq p(t|\mathbf{t}, \hat{\alpha}, \hat{\beta}) = \int p(t|\mathbf{w}, \hat{\beta})p(\mathbf{w}, \hat{\alpha}, \hat{\beta})d\mathbf{w}$$  \hspace{1cm} (3.75)
The evidence approximation (2/2)

Plot of the model evidence $\ln p(t|\alpha, \beta)$ versus $M$, the model complexity, for the polynomial regression of the synthetic sinusoidal example (with fixed $\alpha$).

The computation for $(\hat{\alpha}, \hat{\beta})$ give rise to $\gamma = \alpha m_N^T m_N$ (3.90)

$\gamma$ has the nice interpretation of being the effective number of parameters