Artificial Intelligence

Linear Regression
Ch. 18.6
Basics of Linear Regression

• Regression algorithm
• Supervised technique.
• In one dimension:
  – Identify \( y : \mathbb{R} \rightarrow \mathbb{R} \)
• In D-dimensions:
  – Identify \( y : \mathbb{R}^D \rightarrow \mathbb{R} \)
• Given: training data: \( \{ \vec{x}_0, \vec{x}_1, \ldots, \vec{x}_N \} \)
  – And targets: \( \{ t_0, t_1, \ldots, t_N \} \)
Graphical Example of Regression
Graphical Example of Regression
In linear regression, we assume that the model that generates the data involved only a linear combination of input variables.

\[ y(\vec{x}, \vec{w}) = w_0 + w_1 x_1 + \ldots + w_D x_D \]

\[ y(\vec{x}, \vec{w}) = w_0 + \sum_{j=1}^{D} w_j x_j \]

Where \( \vec{w} \) is a vector of weights which define the \( D \) parameters of the model.
Evaluation

• How can we evaluate the performance of a regression solution?

• Error Functions (or Loss functions)
  – Squared Error

\[ E(t_i, y(\vec{x}_i, \vec{w})) = \frac{1}{2} (t_i - y(\vec{x}_i, \vec{w}))^2 \]
Evaluation

• How can we evaluate the performance of a regression solution?
• Error Functions (or Loss functions)
  – Squared Error
  – Linear Error

\[ E(t_i, y(\vec{x}_i, \vec{w})) = |t_i - y(\vec{x}_i, \vec{w})| \]
Regression Error
Empirical Risk

• Empirical risk is the measure of the loss from data.

\[ R_{\text{emp}} = \frac{1}{N} \sum_{i=1}^{N} E(t_i, y(\vec{x}_i, \vec{w})) \]

\[ = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{2} (t_i - y(\vec{x}_i, \vec{w}))^2 \]

• By minimizing risk on the training data, we optimize the fit with respect to the loss function

\[ \nabla_{\vec{w}} R = 0 \]
Two related but distinct ways to look at a model.

1. **Model Likelihood.**
   - “What is the likelihood that a model *generated* the observed data?”

2. **Empirical Risk**
   - “How much *error* does the model have on the training data?”
What is the likelihood that a model with some parameters generated an observed sample?

We assume Gaussian noise:

$$p(t|x, \mathbf{w}, \beta) = N(t; y(x, \mathbf{w}), \beta^{-1})$$

where $\beta = \frac{1}{\sigma^2}$

$$p(t|x', \mathbf{w}, \beta) = \prod_{i=0}^{N-1} N(t_i; y(x_i, \mathbf{w}), \beta^{-1})$$

Assuming Independently and Identically Distributed (iid) data.
Understanding Model Likelihood

\[ p(t | \vec{x}, \vec{w}, \beta) = \prod_{i=0}^{N-1} N(t_i; y(x_i, \vec{w}), \beta^{-1}) \]

\[ p(t | \vec{x}, \vec{w}, \beta) = \prod_{i=0}^{N-1} \sqrt{\beta} \frac{1}{\sqrt{2\pi}} \exp \left\{ \frac{-\beta}{2} (y(x_i, \vec{w}) - t_i)^2 \right\} \]

\[ \ln p(t | \vec{x}, \vec{w}, \beta) = \ln \prod_{i=0}^{N-1} \sqrt{\beta} \frac{1}{\sqrt{2\pi}} \exp \left\{ \frac{-\beta}{2} (y(x_i, \vec{w}) - t_i)^2 \right\} \]

\[ = -\frac{\beta}{2} \sum_{i=0}^{N-1} \left\{ (y(x_i, \vec{w}) - t_i)^2 \right\} + \frac{N}{2} \ln \beta - \frac{N}{2} \ln 2\pi \]

Substitution for the eqn of a gaussian

Apply a log function

Let the log dissolve products into sums
Understanding Model Likelihood

\[
\ln p(t|\vec{x}, \vec{w}, \beta) = -\frac{\beta}{2} \sum_{i=0}^{N-1} \left\{ (y(x_i, \vec{w}) - t_i)^2 \right\} + \frac{N}{2} \ln \beta - \frac{N}{2} \ln 2\pi
\]

\[
\nabla_{\vec{w}} \ln p(t|\vec{x}, \vec{w}, \beta) = \nabla_{\vec{w}} \left( -\frac{\beta}{2} \sum_{i=0}^{N-1} \left\{ (y(x_i, \vec{w}) - t_i)^2 \right\} \right)
\]

\[
\nabla_{\vec{w}} \left( -\frac{\beta}{2} \sum_{i=0}^{N-1} \left\{ (y(x_i, \vec{w}) - t_i)^2 \right\} \right) = 0
\]

\[
R_{emp} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{2} \left( t_i - y(\vec{x}_i, \vec{w}) \right)^2
\]

Optimize the weights.
(Maximum Likelihood Estimation)

Log Likelihood

Empirical Risk w/ Squared Loss Function
Maximizing Log Likelihood (1-D)

- Find the optimal settings of $w$.

$$\vec{w} = \begin{bmatrix} w_0 & w_1 \end{bmatrix}^T$$

$$\nabla_{\vec{w}} R = \vec{0} \quad \begin{bmatrix} \frac{\partial R}{\partial w_0} \\ \frac{\partial R}{\partial w_1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$R(\vec{w}) = \frac{1}{2N} \sum_{i=0}^{N-1} (t_i - w_1 x_i - w_0)^2$$
Maximizing Log Likelihood

\[ \nabla_{\vec{w}} R(\vec{w}) = \frac{1}{2N} \sum_{i=0}^{N-1} (t_i - w_1 x_i - w_0)^2 \]

\[ \frac{\partial R}{\partial w_0} = \frac{1}{N} \sum_{i=0}^{N-1} (t_i - w_1 x_i - w_0)(-1) \]

\[ \frac{1}{N} \sum_{i=0}^{N-1} (t_i - w_1 x_i - w_0)(-1) = 0 \]

\[ \frac{1}{N} \sum_{i=0}^{N-1} w_0 = \frac{1}{N} \sum_{i=0}^{N-1} (t_i - w_1 x_i) \]

\[ w_0 = \frac{1}{N} \sum_{i=0}^{N-1} (t_i - w_1 x_i) \]

\[ w_0 = \frac{1}{N} \sum_{i=0}^{N-1} t_i - w_1 \frac{1}{N} \sum_{i=0}^{N-1} x_i \]
Maximizing Log Likelihood

\[ \nabla_{\vec{w}} R(\vec{w}) = \frac{1}{2N} \sum_{i=0}^{N-1} (t_i - w_1 x_i - w_0)^2 \]

\[ \frac{\partial R}{\partial w_1} = \frac{1}{N} \sum_{i=0}^{N-1} (t_i - w_1 x_i - w_0)(-x_i) \]

Partial derivative

Set to zero

\[ \frac{1}{N} \sum_{i=0}^{N-1} (t_i - w_1 x_i - w_0)(-x_i) = 0 \]

\[ \frac{1}{N} \sum_{i=0}^{N-1} -(t_i x_i - w_1 x_i^2 - w_0 x_i) = 0 \]

Separate the sum to isolate \(w_0\)

\[ \frac{1}{N} \sum_{i=0}^{N-1} w_1 x_i^2 = \frac{1}{N} \sum_{i=0}^{N-1} t_i x_i - \frac{1}{N} \sum_{i=0}^{N-1} w_0 x_i \]

\[ w_1 \sum_{i=0}^{N-1} x_i^2 = \sum_{i=0}^{N-1} t_i x_i - w_0 \sum_{i=0}^{N-1} x_i \]
Maximizing Log Likelihood

\[ w_0^* = \frac{1}{N} \sum_{i=0}^{N-1} t_i - w_1 \frac{1}{N} \sum_{i=0}^{N-1} x_i \]

\[ w_1 \sum_{i=0}^{N-1} x_i^2 = \sum_{i=0}^{N-1} t_i x_i - w_0 \sum_{i=0}^{N-1} x_i \]

From previous partial

From prev. slide

Substitute

Isolate \( w_1 \)

\[ w_1 \left( \sum_{i=0}^{N-1} x_i^2 - \frac{1}{N} \sum_{i=0}^{N-1} x_i \sum_{i=0}^{N-1} x_i \right) = \sum_{i=0}^{N-1} t_i x_i - \frac{1}{N} \sum_{i=0}^{N-1} t_i \sum_{i=0}^{N-1} x_i \]

\[ w_1^* = \frac{\sum_{i=0}^{N-1} t_i x_i - \frac{1}{N} \sum_{i=0}^{N-1} t_i \sum_{i=0}^{N-1} x_i}{\sum_{i=0}^{N-1} x_i^2 - \frac{1}{N} \sum_{i=0}^{N-1} x_i \sum_{i=0}^{N-1} x_i} \]
Maximizing Log Likelihood

- Clean and easy.

\[
\begin{bmatrix}
    w_0^* \\
    w_1^*
\end{bmatrix}
= \left[
\frac{1}{N} \sum_{i=0}^{N-1} t_i - w_1^* \frac{1}{N} \sum_{i=0}^{N-1} x_i \\
\frac{1}{N} \sum_{i=0}^{N-1} t_i x_i - \frac{1}{N} \sum_{i=0}^{N-1} t_i \sum_{i=0}^{N-1} x_i
\right]
\left[
\sum_{i=0}^{N-1} x_i - \frac{1}{N} \sum_{i=0}^{N-1} x_i \sum_{i=0}^{N-1} x_i
\right]
\]

- Or not…

- Apply some linear algebra.
Likelihood using linear algebra

- Representing the linear regression function in terms of vectors.

\[ y = w_0 + w_1 x_1 + w_2 x_2 + \ldots + w_{N-1} x_{N-1} \]

\[ \vec{x} = \begin{bmatrix} 1 & x_1 & x_2 & \ldots & x_{N-1} \end{bmatrix}^T \]

\[ \vec{w} = \begin{bmatrix} w_0 & w_1 & w_2 & \ldots & w_{N-1} \end{bmatrix}^T \]

\[ y = \vec{x}^T \vec{w} \]
Likelihood using linear algebra

- Stack $x^T$ into a matrix of data points, $X$.

\[
R_{emp}(\vec{w}) = \frac{1}{2N} \sum_{i=0}^{N-1} (t_i - w_1 x_i - w_0)^2
\]

\[
= \frac{1}{2N} \sum_{i=0}^{N-1} \left( t_i - \begin{bmatrix} 1 & x_i \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} \right)^2
\]

\[
= \frac{1}{2N} \left\| \begin{bmatrix} t_0 \\ t_1 \\ \vdots \\ t_{N-1} \end{bmatrix} - \begin{bmatrix} 1 & x_0 \\ 1 & x_1 \\ \vdots \\ 1 & x_{N-1} \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} \right\|^2
\]

\[
= \frac{1}{2N} \left\| \vec{t} - \vec{Xw} \right\|^2
\]
Likelihood in multiple dimensions

- This representation of risk has no inherent dimensionality.

\[
R_{emp}(\vec{w}) = \frac{1}{2N} \| \vec{t} - \vec{X} \vec{w} \|^2 \\
\nabla_{\vec{w}} R_{emp}(\vec{w}) = 0 \\
\nabla_{\vec{w}} \left( \frac{1}{2N} \| \vec{t} - \vec{X} \vec{w} \|^2 \right) = 0
\]
**Maximum Likelihood Estimation redux**

\[
\nabla_{\vec{w}} R_{emp}(\vec{w}) = 0
\]

\[
\nabla_{\vec{w}} \left( \frac{1}{2N} \left\| \vec{t} - \vec{X} \vec{w} \right\|^2 \right) = 0
\]

\[
\frac{1}{2N} \nabla_{\vec{w}} \left( (\vec{t} - \vec{X} \vec{w})^T (\vec{t} - \vec{X} \vec{w}) \right) = 0
\]

\[
\frac{1}{2N} \nabla_{\vec{w}} \left( (\vec{t}^T \vec{t} - \vec{t}^T \vec{X} \vec{w} - \vec{w}^T \vec{X}^T \vec{t} + \vec{w}^T \vec{X}^T \vec{X} \vec{w}) \right) = 0
\]

\[
\frac{1}{2N} \left( -\vec{X}^T \vec{t} - \vec{X}^T \vec{t} + 2\vec{X}^T \vec{X} \vec{w}^* \right) = 0
\]

\[
\frac{1}{2N} \left( -2\vec{X}^T \vec{t} + 2\vec{X}^T \vec{X} \vec{w}^* \right) = 0
\]

\[
\vec{X}^T \vec{X} \vec{w}^* = \vec{X}^T \vec{t}
\]

\[
\vec{w}^* = (\vec{X}^T \vec{X})^{-1} \vec{X}^T \vec{t}
\]

Decompose the norm

FOIL – linear algebra style

Differentiate

Combine terms

Isolate \( \vec{w} \)
Extension to polynomial regression
Extension to polynomial regression

\[ y = c_0 + c_1 x_1 + c_2 x_2 \]

\[ y = c_0 + c_1 x + c_2 x^2 \]

- Polynomial regression is the same as linear regression in D dimensions
Training data vs. Testing Data

- Evaluating the performance of a classifier on training data only is meaningless.
- With enough parameters, a model can simply memorize (encode) every training point.
- To evaluate performance, data is divided into training and testing (or evaluation) data.
  - Training data is used to learn model parameters
  - Testing data is used to evaluate performance
Overfitting
Overfitting

\[ M = 3 \]

\[ M = 9 \]
Overfitting performance
Definition of overfitting

• When the model describes the noise, rather than the signal.

• How can you tell the difference between overfitting, and a bad model?
Possible detection of overfitting

• Stability
  – An appropriately fit model is stable under different samples of the training data
  – An overfit model generates inconsistent performance

• Performance
  – A good model has low test error
  – A bad model has high test error
What is the optimal model size?

- The best model size generalizes to unseen data the best.
- Approximate this by testing error.
- One way to optimize parameters is to minimize testing error.
  - This operation uses testing data as tuning or development data
- Sacrifices training data in favor of parameter optimization