Linear regression

Outline

Linear Regression
- Linear model
- Loss (error) function based on the least squares fit
- Parameter estimation.
- Gradient methods.
- On-line regression techniques.
- Linear additive models
- Statistical model of linear regression
**Supervised learning**

**Data:** \( D = \{ D_1, D_2, ..., D_n \} \) a set of \( n \) examples

- \( D_i = \langle x_i, y_i \rangle \)
- \( x_i = (x_{i,1}, x_{i,2}, \cdots x_{i,d}) \) is an input vector of size \( d \)
- \( y_i \) is the desired output (given by a teacher)

**Objective:** learn the mapping \( f : X \rightarrow Y \)

\[ y_i \approx f(x_i) \text{ for all } i = 1, ..., n \]

- **Regression:** \( Y \) is **continuous**
  - Example: earnings, product orders \( \rightarrow \) company stock price
- **Classification:** \( Y \) is **discrete**
  - Example: handwritten digit in binary form \( \rightarrow \) digit label

---

**Linear regression**

- **Function** \( f : X \rightarrow Y \) is a linear combination of input components

\[ f(x) = w_0 + w_1 x_1 + w_2 x_2 + \cdots w_d x_d = w_0 + \sum_{j=1}^{d} w_j x_j \]

\( w_0, w_1, \ldots, w_k \) - parameters (weights)

Bias term \( 1 \)

Input vector \( x \)

\[ x_1, x_2, \ldots, x_d \]

- Diagram showing the linear regression function and the bias term.
Linear regression

• **Shorter (vector) definition of the model**
  – Include bias constant in the input vector

\[
x = (1, x_1, x_2, \ldots, x_d)
\]

\[
f(x) = w_0 x_0 + w_1 x_1 + w_2 x_2 + \ldots w_d x_d = \mathbf{w}^T \mathbf{x}
\]

\[w_0, w_1, \ldots, w_k\] - parameters (weights)

\[\sum_{i=1}^{1} f(x, w) = 1 \rightarrow \sum_{i=1}^{1} \sum_{x} \sum_{w} f(x, w)
\]

Input vector

**x**

\[
\begin{cases}
  1 & w_0 \\
  x_1 & w_1 \\
  x_2 & w_2 \\
  \vdots & w_d \\
  x_d & \end{cases}
\]

Linear regression. Error.

• Data: \( D_i =< x_i, y_i > \)
• Function: \( x_i \rightarrow f(x_i) \)
• We would like to have \( y_i \approx f(x_i) \) for all \( i = 1, \ldots, n \)

• **Error function**
  – measures how much our predictions deviate from the desired answers

Mean-squared error

\[
J_n = \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2
\]

• Learning:
  We want to find the weights minimizing the error!
Linear regression. Example

• 1 dimensional input $x = (x_1)$

Linear regression. Example.

• 2 dimensional input $x = (x_1, x_2)$
**Linear regression. Optimization.**

- We want the **weights minimizing the error**
  \[ J_n = \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2 = \frac{1}{n} \sum_{i=1}^{n} (y_i - w^T x_i)^2 \]

- For the optimal set of parameters, derivatives of the error with respect to each parameter must be 0
  \[ \frac{\partial}{\partial w_j} J_n(w) = -\frac{2}{n} \sum_{i=1}^{n} (y_i - w_0 x_{i,0} - w_1 x_{i,1} - \ldots - w_d x_{i,d}) x_{i,j} = 0 \]

- **Vector of derivatives:**
  \[ \text{grad}_w (J_n(w)) = \nabla_w (J_n(w)) = -\frac{2}{n} \sum_{i=1}^{n} (y_i - w^T x_i) x_i = 0 \]

---

**Linear regression. Optimization.**

- \( \text{grad}_w (J_n(w)) = 0 \) defines a set of equations in \( w \)
  \[ \frac{\partial}{\partial w_0} J_n(w) = -\frac{2}{n} \sum_{i=1}^{n} (y_i - w_0 x_{i,0} - w_1 x_{i,1} - \ldots - w_d x_{i,d}) = 0 \]
  \[ \frac{\partial}{\partial w_1} J_n(w) = -\frac{2}{n} \sum_{i=1}^{n} (y_i - w_0 x_{i,0} - w_1 x_{i,1} - \ldots - w_d x_{i,d}) x_{i,1} = 0 \]
  \[ \ldots \]
  \[ \frac{\partial}{\partial w_d} J_n(w) = -\frac{2}{n} \sum_{i=1}^{n} (y_i - w_0 x_{i,0} - w_1 x_{i,1} - \ldots - w_d x_{i,d}) x_{i,d} = 0 \]
Solving linear regression

\[ \frac{\partial}{\partial w_j} J_n(w) = -\frac{2}{n} \sum_{i=1}^{n} (y_i - w_0 x_{i,0} - w_1 x_{i,1} - \ldots - w_d x_{i,d}) x_{i,j} = 0 \]

By rearranging the terms we get a system of linear equations with \( d+1 \) unknowns:

\[ Aw = b \]

\[
\begin{align*}
    w_0 \sum_{i=1}^{n} x_{i,0} &+ w_1 \sum_{i=1}^{n} x_{i,1} + \ldots + w_d \sum_{i=1}^{n} x_{i,d} = \sum_{i=1}^{n} y_i \\
    w_0 \sum_{i=1}^{n} x_{i,0} x_{i,1} + w_1 \sum_{i=1}^{n} x_{i,1} x_{i,1} + \ldots + w_d \sum_{i=1}^{n} x_{i,d} x_{i,1} = \sum_{i=1}^{n} y_i x_{i,1} \\
    \vdots \\
    w_0 \sum_{i=1}^{n} x_{i,0} x_{i,j} + w_1 \sum_{i=1}^{n} x_{i,1} x_{i,j} + \ldots + w_d \sum_{i=1}^{n} x_{i,d} x_{i,j} = \sum_{i=1}^{n} y_i x_{i,j} 
\end{align*}
\]

Solving linear regression

- The optimal set of weights satisfies:
  \[ \nabla_w (J_n(w)) = -\frac{2}{n} \sum_{i=1}^{n} (y_i - w^T x_i) x_i = 0 \]

Leads to a system of linear equations (SLE) with \( d+1 \) unknowns of the form

\[ Aw = b \]

Solution to SLE: ?
Solving linear regression

- The optimal set of weights satisfies:
  \[ \nabla_w (J_n (w)) = - \frac{2}{n} \sum_{i=1}^{n} (y_i - w^T x_i)x_i = 0 \]

  Leads to a system of linear equations (SLE) with \( d+1 \) unknowns of the form
  \[ Aw = b \]

  \[
  w_0 \sum_{i=1}^{n} x_{i,0}x_{i,j} + w_1 \sum_{i=1}^{n} x_{i,1}x_{i,j} + \ldots + w_j \sum_{i=1}^{n} x_{i,j}x_{i,j} + \ldots + w_d \sum_{i=1}^{n} x_{i,d}x_{i,j} = \sum_{i=1}^{n} y_i x_{i,j}
  \]

  Solution to SLE:
  \[ w = A^{-1}b \]

- matrix inversion