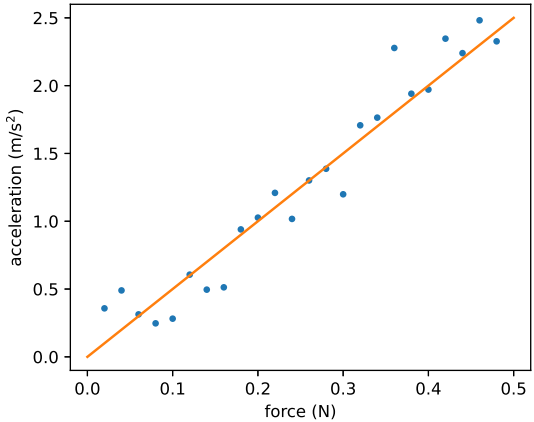
Machine Learning

Linear Regression

Kia Nazarpour

Based on Hao Tang's slides

Motivation

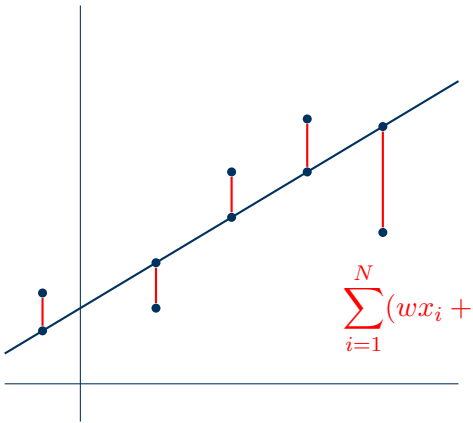


Learning Outcomes

By the end of the session students will be able to

1. Articulate the fundamental concepts of linear regression.
2. Derive the equations for linear regression.
3. Apply linear regression to study real-world data

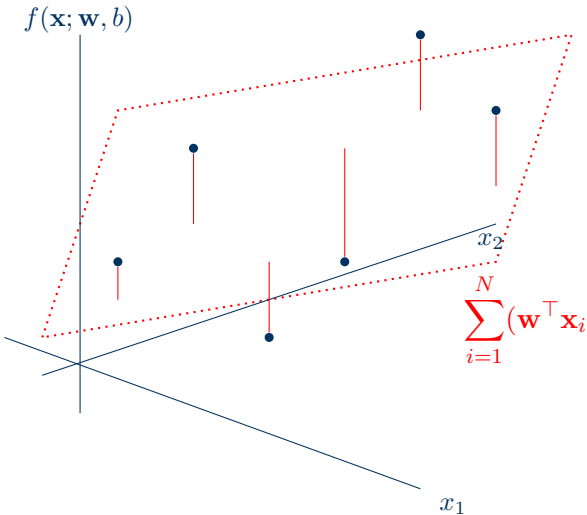
Affine Fucntion - Two dimensional space



$$f(x; w, b) = y = wx + b$$

$$\sum_{i=1}^{N} (wx_i + b - y_i)^2$$

Affine Function - Three dimensional space



$$\sum_{i=1}^{N} (\mathbf{w}^\top \mathbf{x}_i + b - y_i)^2$$

$$\forall \mathbf{x} \in f$$

$$\begin{aligned} f(\mathbf{x}; \mathbf{w}, b) &= \mathbf{w}^\top \mathbf{x} + b \\ &= w_1 x_1 + w_2 x_2 + b \\ &= \mathbf{x}^\top \mathbf{w} + b \\ &= \mathbf{y} \end{aligned}$$

Linear regression

- $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$: data set
  - $\mathbf{x}_i = \begin{bmatrix} x_{i1} & \cdots & x_{id} \end{bmatrix}^\top$: input, features, independent variable
  - $y_i \in \mathbb{R}$: dependent variable ground truth for $\mathbf{x}_i$.
- $f(\mathbf{x}; \mathbf{w}, b) = \mathbf{w}^\top \mathbf{x} + b$: linear predictor, hyperplane
  - $\mathbf{w} = \begin{bmatrix} w_1 & \cdots & w_d \end{bmatrix}^\top$: weights
  - $b \in \mathbb{R}$: bias
  - $\{\mathbf{w}, b\}$: parameters ... $\boldsymbol{\theta} = \begin{bmatrix} b & \mathbf{w}^\top \end{bmatrix}^\top$

Linear regression

- Given S, find $\boldsymbol{\theta}$ such that the mean-squared error (MSE)

$$L = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{w}^\top \mathbf{x}_i + b - y_i)^2$$

  is minimized.

- The act of finding $\mathbf{w}$ is called training.

Linear regression

- The goal of linear regression is to solve

$$\min_{\mathbf{w},b} \quad \frac{1}{N}\sum_{i=1}^{N}(\mathbf{w}^\top \mathbf{x}_i + b - y_i)^2.$$

- The optimal solution satisfies

$$\begin{aligned}\frac{\partial L}{\partial b} &= 0 \\ \frac{\partial L}{\partial \mathbf{w}} &= \begin{bmatrix}\frac{\partial L}{\partial w_1} & \frac{\partial L}{\partial w_2} & \cdots & \frac{\partial L}{\partial w_d}\end{bmatrix} = \mathbf{0}.\end{aligned}$$

1) Differentiate - 2) Solve - 3) Replace - ...
Is this optimal? More on this in Optimisation.

Linear regression - finding the bias b

$$\begin{aligned}\frac{\partial}{\partial b}\frac{1}{N}\sum_{i=1}^{N}(\mathbf{w}^\top\mathbf{x}_i + b - y_i)^2 &= \frac{2}{N}\sum_{i=1}^{N}(\mathbf{w}^\top\mathbf{x}_i + b - y_i) \\ &= 2b - \frac{2}{N}\sum_{i=1}^{N}(y_i - \mathbf{w}^\top\mathbf{x}_i) = 0\end{aligned}$$

$$b = \frac{1}{N}\sum_{i=1}^{N}(y_i - \mathbf{w}^\top\mathbf{x}_i) = \frac{1}{N}\sum_{i=1}^{N} y_i - \mathbf{w}^\top\left(\frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i\right) = \bar{y} - \mathbf{w}^\top\bar{\mathbf{x}}$$

$$\boxed{\frac{\partial L}{\partial b} = 0 \implies b = \bar{y} - \mathbf{w}^\top\bar{\mathbf{x}}}$$

Linear regression - data centring (mean normalisation)

$$\frac{\partial L}{\partial b} = 0 \implies b = \bar{y} - \mathbf{w}^\top \bar{\mathbf{x}}$$

$$\begin{aligned}
L &= \frac{1}{N}\sum_{i=1}^{N}(\mathbf{w}^\top \mathbf{x}_i + b - y_i)^2 \\
&= \frac{1}{N}\sum_{i=1}^{N}(\mathbf{w}^\top \mathbf{x}_i + \bar{y} - \mathbf{w}^\top \bar{\mathbf{x}} - y_i)^2 \\
&= \frac{1}{N}\sum_{i=1}^{N}[\mathbf{w}^\top \underbrace{(\mathbf{x}_i - \bar{\mathbf{x}})}_{\dot{\mathbf{x}}_i} - \underbrace{(y_i - \bar{y})}_{\dot{y}_i}]^2 \\
&= \frac{1}{N}\sum_{i=1}^{N}(\mathbf{w}^\top \dot{\mathbf{x}}_i - \dot{y}_i)^2
\end{aligned}$$

Linear regression - finding the weights $\mathbf{w}$

$$\begin{aligned}\frac{\partial}{\partial \mathbf{w}} \frac{1}{N} \sum_{i=1}^{N} (\mathbf{w}^\top \dot{\mathbf{x}}_i - \dot{y}_i)^2 &= \frac{2}{N} \sum_{i=1}^{N} (\mathbf{w}^\top \dot{\mathbf{x}}_i - \dot{y}_i)(\dot{\mathbf{x}}_i) \\ &= \frac{2}{N} \sum_{i=1}^{N} ((\mathbf{w}^\top \dot{\mathbf{x}}_i)\dot{\mathbf{x}}_i - \dot{y}_i \dot{\mathbf{x}}_i)\end{aligned}$$

Linear regression - finding the weights $\mathbf{w}$

To express the result in matrix form, we use vectorised notation. Let:

- $\mathbf{X} \in \mathbb{R}^{N \times d}$ be the matrix with rows $\dot{\mathbf{x}}_i^\top$,
- $\dot{\mathbf{y}} \in \mathbb{R}^N$ be the vector of targets $\dot{y}_i$,
- $\mathbf{w} \in \mathbb{R}^d$ be the vector of weights.

We can them compute the gradient in vectorised form as

$$\frac{\partial L}{\partial \mathbf{w}} = \frac{2}{N} \mathbf{X}^\top (\mathbf{X}\mathbf{w} - \dot{\mathbf{y}}).$$

Linear regression - finding the weights $\mathbf{w}$

Now that we have the gradient, let's solve for $\mathbf{w}$ by setting the gradient to zero

$$\frac{2}{N}\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \dot{\mathbf{y}}) = \mathbf{0} \quad \rightarrow \quad \mathbf{X}^\top(\mathbf{X}\mathbf{w} - \dot{\mathbf{y}}) = \mathbf{0}$$

Solving for $\mathbf{w}$ we will have:

$$\boxed{\mathbf{w} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\dot{\mathbf{y}}}$$

Linear regression - Summary

1. Centering

$$\dot{\mathbf{y}} = \begin{bmatrix} y_1 - \bar{y} \\ y_2 - \bar{y} \\ \vdots \\ y_N - \bar{y} \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} \dot{\mathbf{x}}_1^\top \\ \dot{\mathbf{x}}_2^\top \\ \vdots \\ \dot{\mathbf{x}}_N^\top \end{bmatrix}$$

2. Computing the **Moore-Penrose pseudoinverse**

$$\begin{aligned} \mathbf{w} &= (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \dot{\mathbf{y}} \\ b &= \bar{y} - \mathbf{w}^\top \bar{\mathbf{x}} \end{aligned}$$

Linear regression - Example

Let's say you're trying to figure out if studying more hours actually prepares your better for the exam, or if it's just the coffee. You decide to collect some data on how many hours you studied and what mark you got.

Hours Studied (x)	Test Score (y)
1	55.9
2	59.7
3	66.2
4	73.0
5	74.5
6	79.5
7	88.1

Fit a linear regression model to this data in the form $y = b + w_1 x$ where

- b is the intercept (the score when no hours are studied),
- w_1 is the slope (the increase in score for each additional hour studied).