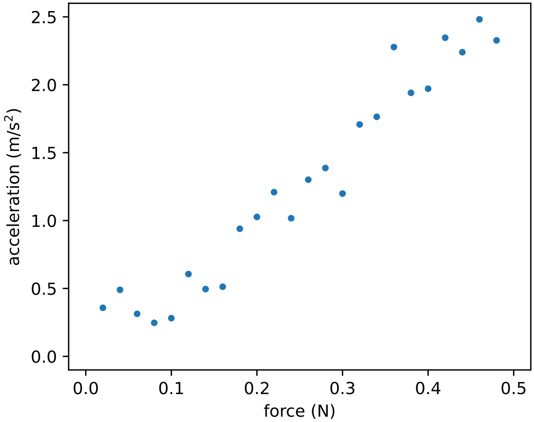
Machine Learning

Lecture 3: Linear Regression

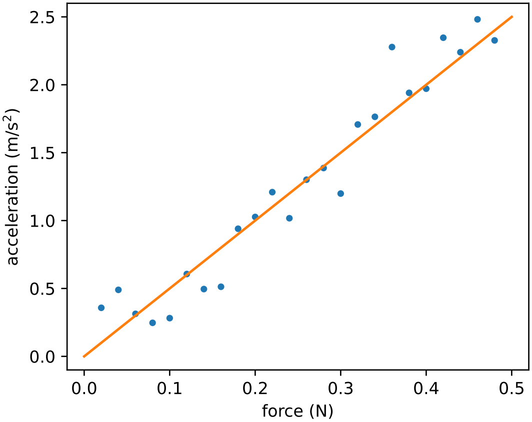
Hao Tang

September 29, 2022

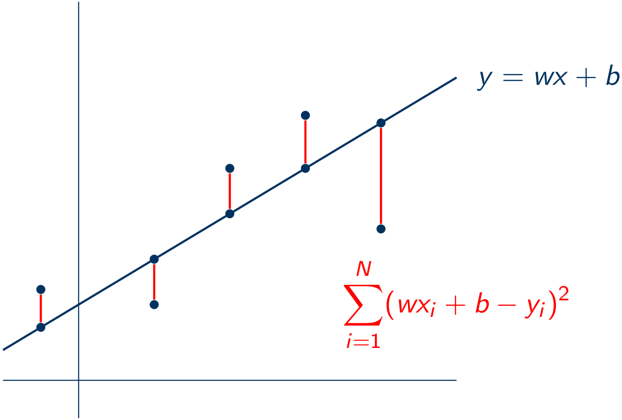
First example



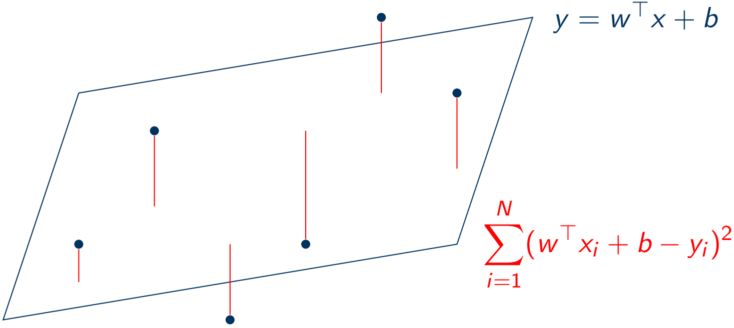
First example



Geometry



Geometry



Linear regression

- $S = \{(x_1, y_1), \ldots, (x_N, y_N)\}$: data set
  - $x = \begin{bmatrix} x[1] & \cdots & x[d] \end{bmatrix}^\top$: input, features
  - y: ground truth, label, gold reference.
- $f(x) = w^\top x + b$: linear predictor, hyperplane
  - $w = \begin{bmatrix} w[1] & \cdots & w[d] \end{bmatrix}^\top$: weights
  - $b \in \mathbb{R}$: bias
  - $\{w, b\}$: parameters

Linear regression

- Given $S = \{(x_1, y_1), \ldots, (x_N, y_N)\}$, find w such that the mean-squared error (MSE)

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

  is minimized.

- The act of finding w is called training.

Linear regression

- The goal of linear regression is to solve

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

- The optimal solution satisfies

$$\frac{\partial L}{\partial b} = 0 \qquad \frac{\partial L}{\partial w} = 0. \tag{3}$$

(Is this optimal? More on this in Lecture 7.)

Linear regression

$$\frac{\partial}{\partial b} \frac{1}{N} \sum_{i=1}^{N} (w^\top x_i + b - y_i)^2 = \frac{2}{N} \sum_{i=1}^{N} (w^\top x_i + b - y_i) \tag{4}$$

$$= -2b + \frac{2}{N} \sum_{i=1}^{N} (y_i - w^\top x_i) = 0 \tag{5}$$

$$b = \frac{1}{N} \sum_{i=1}^{N} (y_i - w^\top x_i) = \frac{1}{N} \sum_{i=1}^{N} y_i - w^\top \left(\frac{1}{N} \sum_{i=1}^{N} x_i \right) = \bar{y} - w^\top \bar{x} \tag{6}$$

Linear regression

$$\frac{\partial L}{\partial b} = 0 \implies b = \bar{y} - w^\top \bar{x} \tag{7}$$

$$L = \frac{1}{N}\sum_{i=1}^{N}(w^\top x_i + b - y_i)^2 = \frac{1}{N}\sum_{i=1}^{N}[w^\top(x_i - \bar{x}) - (y_i - \bar{y})]^2 \tag{8}$$

$$= \frac{1}{N}\sum_{i=1}^{N}(w^\top x_i' - y_i')^2 \tag{9}$$

Linear regression

$$\frac{\partial}{\partial w}\frac{1}{N}\sum_{i=1}^{N}(w^\top x_i' - y_i')^2 = \frac{2}{N}\sum_{i=1}^{N}(w^\top x_i' - y_i')(x_i') \tag{10}$$

$$= \frac{2}{N}\sum_{i=1}^{N}((w^\top x_i')x_i' - y_i' x_i') \tag{11}$$

Linear regression

$$\frac{\partial}{\partial w}\frac{1}{N}\sum_{i=1}^{N}(w^\top x'_i - y'_i)^2 = \frac{2}{N}\sum_{i=1}^{N}((w^\top x'_i)x'_i - y'_i x'_i) \tag{12}$$

$$= \frac{2}{N}\left(\begin{bmatrix} x'_1 & x'_2 & \cdots & x'_n \end{bmatrix}\begin{bmatrix} w^\top x'_1 \\ w^\top x'_2 \\ \vdots \\ w^\top x'_n \end{bmatrix} - \begin{bmatrix} x'_1 & x'_2 & \cdots & x'_n \end{bmatrix}\begin{bmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_n \end{bmatrix}\right) \tag{13}$$

$$= \frac{2}{N}(XX^\top w - Xy) = 0 \tag{14}$$

$$w = (XX^\top)^{-1}Xy \tag{15}$$

Linear regression

1. Centering

$$y = \begin{bmatrix} y_1 - \bar{y} \\ \vdots \\ y_N - \bar{y} \end{bmatrix} \qquad X = \begin{bmatrix} x_1 - \bar{x} & \cdots & x_N - \bar{x} \end{bmatrix} \tag{16}$$

2. Computing the Moore-Penrose pseudoinverse

$$w = (XX^\top)^{-1}Xy \tag{17}$$

$$b = \bar{y} - w^\top \bar{x} \tag{18}$$

Features

$$y = w^\top x + b = \begin{bmatrix} w^\top & b \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} w \\ b \end{bmatrix}^\top \begin{bmatrix} x \\ 1 \end{bmatrix} = w'^\top x' \tag{19}$$

- Fitting $f(x) = w^\top x + b$ is equivalent to appending 1 to x and fitting $f(x) = w^\top x$.
- The 1 can be seen as a feature independent of the input.

Features

- Suppose we have a data point $x = \begin{bmatrix} x[1] & x[2] & x[3] \end{bmatrix}^\top$.
- The data point after appending 1 becomes

$$\begin{bmatrix} 1 & x[1] & x[2] & x[3] \end{bmatrix}^\top \tag{20}$$

- The data point after appending 1 and quadratic terms becomes

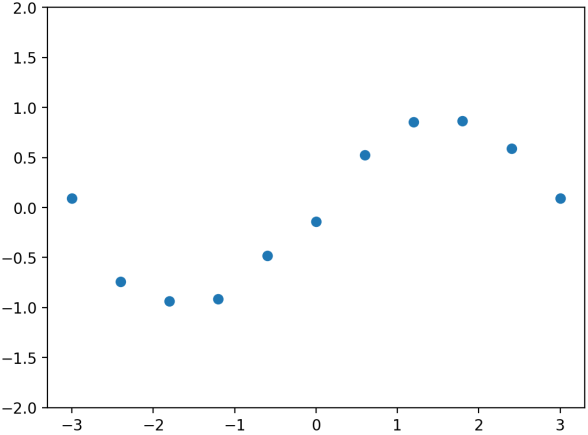
$$\phi(x) = \begin{bmatrix} 1 & x[1] & x[2] & x[3] & x[1]x[2] & x[2]x[3] & x[1]x[3] & x[1]^2 & x[2]^2 & x[3]^2 \end{bmatrix}^\top \tag{21}$$

- The function $f(x) = w^\top \phi(x)$ is a polynomial.

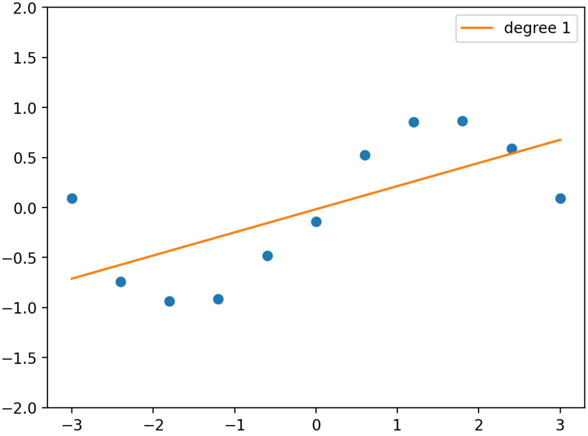
Features

- We call ϕ a feature function.
- In general, ϕ can be any function.
- Instead of $f(x) = w^\top x + b$, we now have $f(x) = w^\top \phi(x)$.
- Instead of $X = \begin{bmatrix} x_1 & x_2 & \cdots & x_N \end{bmatrix}$, we have $\Phi = \begin{bmatrix} \phi(x_1) & \phi(x_2) & \cdots & \phi(x_N) \end{bmatrix}$
- The optimal solution for linear regression becomes $w = (\Phi\Phi^\top)^{-1}\Phi y$.

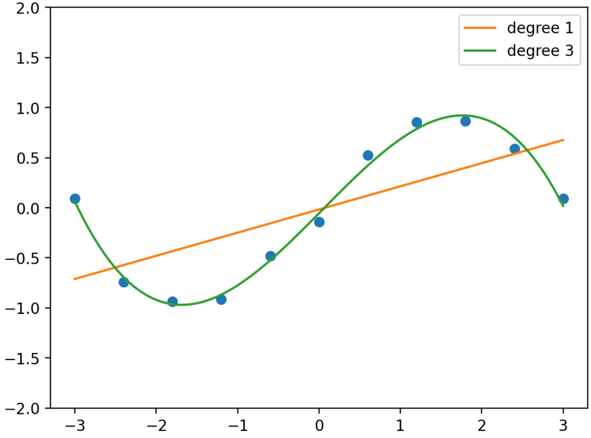
Examples



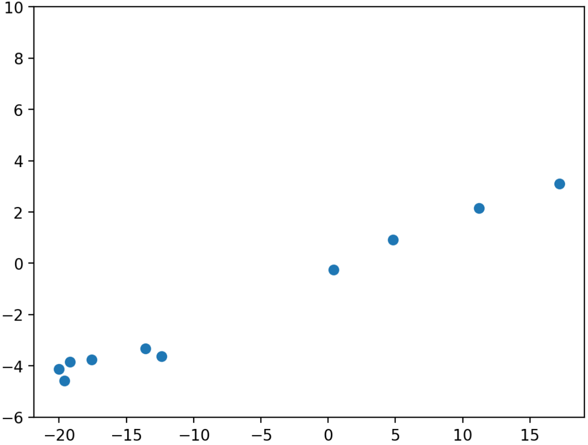
Examples



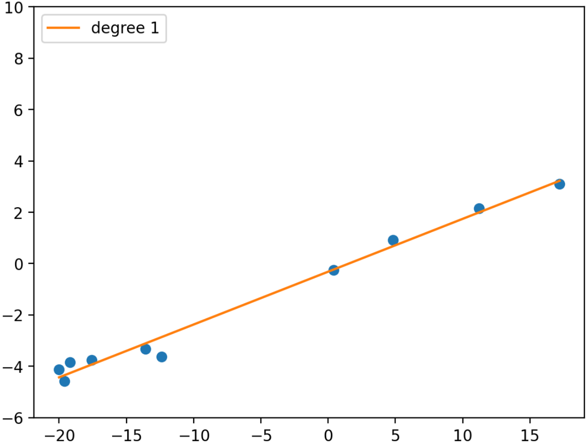
Examples



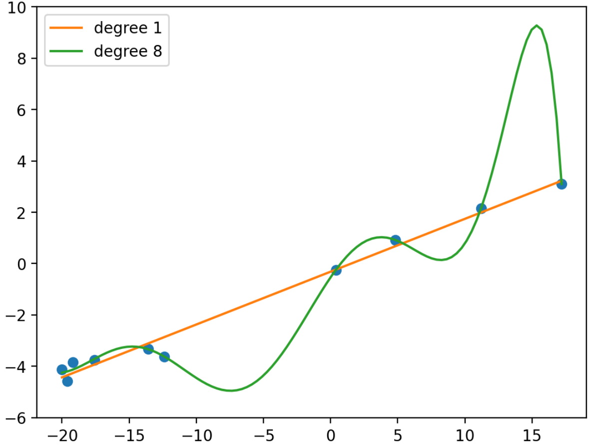
Examples



Examples



Examples



Linear regression

- A "linear" regression model is linear in the parameters w, **not** the features.
- A linear regression model can fit an arbitrary nonlinear function.
- What are the "right" features?
- What does it mean for the program $w^\top \phi(x)$ we write with data to be "correct"?

A probabilistic interpretation

- Assume we cannot get a perfect fit because of noise.
- In particular, we assume the noise is additive and Gussian.
- In other words, $y_i = w^\top \phi(x_i) + \epsilon_i$, where $\epsilon_i \sim \mathcal{N}(0, 1)$.
- If $\epsilon_i \sim \mathcal{N}(0, 1)$, then $y_i \sim \mathcal{N}(w^\top \phi(x_i), 1)$.
- The log-likelihood of w is

$$\log \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(y_i - w^\top \phi(x_i))^2\right) \tag{22}$$

A probabilistic interpretation

- Log-likelihood of w

$$\sum_{i=1}^{N}\left[-\frac{1}{2}\log(2\pi)-\frac{1}{2}(y_i-w^\top\phi(x_i))^2\right] \tag{23}$$

- Mean-squared error

$$\frac{1}{N}\sum_{i=1}^{N}(y_i-w^\top\phi(x_i))^2 \tag{24}$$

- The maximum likelihood estimator is the optimal solution for MSE.

Linear regression

- The complexity of computing $(\Phi\Phi^\top)\Phi y$ is $O(N^3)$, where N is the number of samples.
- The runtime is not particularly suitable for large data sets.
- Instead of solving $\min_w L$ exactly, could we find an approximate solution?
- In exchange, could we have an algorithm that scales better than $O(N^3)$?
- Not all problems have closed-form solutions for $\frac{\partial L}{\partial w}$ anyways.

Linear regression

- We write a program $f(x) = w^\top \phi(x)$ with $w = (\Phi\Phi^\top)^{-1}\Phi y$.
- In what sense is this program "correct"?

Linear regression using matrix calculus

- The mean-squared error can be written compactly as

$$L = \|\Phi^\top w - y\|_2^2. \tag{25}$$

- We can expand the mean-squared error as

$$L = \|\Phi^\top w - y\|_2^2 = (\Phi^\top w - y)^\top(\Phi^\top w - y) = w^\top \Phi\Phi^\top w - 2y^\top \Phi^\top w + y^\top y. \tag{26}$$

- Solving the optimal solution gives

$$\frac{\partial L}{\partial w} = (\Phi\Phi^\top + (\Phi\Phi^\top)^\top)w - 2\Phi y = 0 \implies w = (\Phi\Phi^\top)^{-1}\Phi y. \tag{27}$$

Check your understanding

- What is mean-squared error?
- Given a data set, what is the optimal solution for mean-squared error?
- How can we include polynomial features in regression?
- Can linear regression fit nonlinear functions?
- What is the likelihood of a hyperplane under Gaussian noise given a data set?