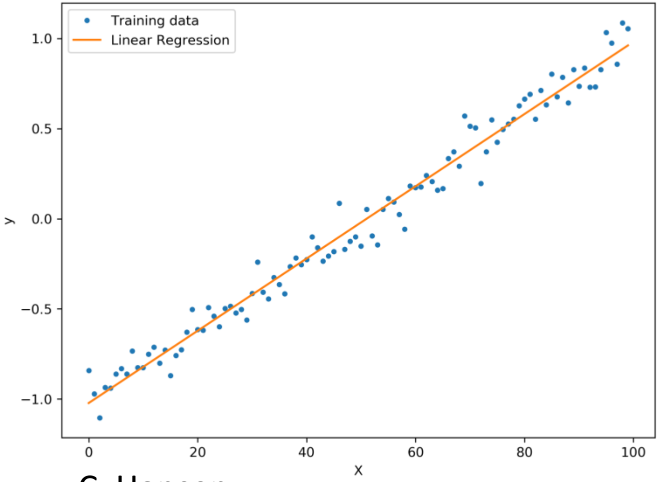
CS 540 Introduction to Artificial Intelligence
Linear Regression Bonus Slides

Fred Sala
University of Wisconsin-Madison

March 2, 2021

Linear Regression

Simplest type of regression problem.



C. Hansen

- **Inputs**: $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)$
  - *x*'s are vectors, *y*'s are scalars.
  - "**Linear**": predict a linear combination of x components + intercept

$$f(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_d x_d = \theta_0 + x^T \theta$$

- **Want**: parameters θ

Linear Regression Setup

Problem Setup

- Goal: figure out how to minimize square loss
- Let's organize it. Train set $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)$
  - Since $f(x) = \theta_0 + x^T\theta$, wrap intercept: $f(x) = x^T\theta$
  - Take train data and make it a matrix/vector: $X = \begin{bmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{bmatrix}$
  - Then, square loss is

$$\frac{1}{n}\sum_{i=1}^{n}(x_i^T\theta - y_i)^2 = \frac{1}{n}\|X^T\theta - y\|^2$$

Finding The Optimal Parameters

Have our loss: $\frac{1}{n}\|X^T\theta - y\|^2$

- Could optimize it with SGD, etc…
- No need: minimum has a solution (easy with vector calculus)

$$\hat{\theta} = (X^T X)^{-1} X^T y$$

Hat: indicates an estimate

Not always invertible…

“Normal Equations”

How Good are the Optimal Parameters?

Now we have parameters $\hat{\theta} = (X^T X)^{-1} X^T y$

- How good are they?
- Predictions are $f(x_i) = \hat{\theta}^T x_i = ((X^T X)^{-1} X^T y)^T x_i$
- Errors ("**residuals**")

$$|y_i - f(x_i)| = |y_i - \hat{\theta}^T x_i| = |y_i - ((X^T X)^{-1} X^T y)^T x_i|$$

- If data is linear, residuals are 0. Almost never the case!

Train/Test for Linear Regression?

So far, residuals measure error on **train** set

- Sometimes that's all we care about (**Fixed Design LR**)
  - Data is deterministic.
  - Goal: find best linear relationship on dataset

- Or, create a test set and check (**Random Design LR**)
  - Common: assume data is $y = \theta^T x + \varepsilon$
  - The more noise, the less linear

0-mean
Gaussian noise

Solving With Gradient Descent

What if we don't know the exact solution?

- Use one of the iterative algorithms to do $\min_{\theta} \ell(\theta)$
- Among the most popular: **gradient descent**
- Basic idea: start at $\theta^{(0)}$
  - Next step: do $\theta^{(j+1)} = \theta^{(j)} - \gamma \nabla \ell(\theta^{(j)})$

Gradient of the loss, evaluated at current sol.

Next solution

Current solution

Learning Rate (a constant)

  - Run till convergence. (You'll implement this in HW5!)

Linear Regression → Classification?

What if we want the same idea, but *y* is 0 or 1?

- Need to convert the $\theta^T x$ to a probability in [0,1]

$$p(y=1|x) = \frac{1}{1+\exp(-\theta^T x)}$$

Logistic function

Why does this work?

- If $\theta^T x$ is really big, $\exp(-\theta^T x)$ is really small → *p* close to 1
- If really negative exp is huge → *p* close to 0

"Logistic Regression"