

A decorative graphic featuring several overlapping circles in teal, lime green, orange, and pink. A large dashed teal circle is centered on the page. Smaller solid and dashed circles in various colors are scattered around the perimeter.

Machine Learning Crash Course

Week 2

The background features a large, faint dashed circle. Scattered around it are various solid-colored circles and arcs in shades of green, yellow, orange, red, and blue. Some circles have smaller circles inside them, creating a nested effect. The overall aesthetic is modern and abstract.

“

“The pace of progress in artificial intelligence (I’m not referring to narrow AI) is incredibly fast. Unless you have direct exposure to groups like Deepmind, you have no idea how fast—it is growing at a pace close to exponential. The risk of something seriously dangerous happening is in the five-year timeframe. 10 years at most.” —Elon Musk

A decorative graphic consisting of various colored circles and rings in shades of pink, orange, teal, yellow, and green, scattered across the slide.

Multivariate Linear Regression

Today's Topics

- Introducing multiple features to the data
- Adapting the Gradient Descent function
- Polynomial Regression
- Normal Equations
- Normal Equations with Non-Invertibility

Housing Price Prediction

Consider the Features
 X_1, X_2, X_3, X_4

Size(feet ²) X_1	Number of bedrooms X_2	Number of floors X_3	Age of house (years) X_4	Price (\$1000) y
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
\vdots	\vdots	\vdots	\vdots	\vdots

M training
samples

n is the number of features

$x^{(i)}$ is the i th training sample

$x_j^{(i)}$ is the j th feature in the i th
training sample

E.g. $n=4$ with the example
above

$$x^{(2)} = \begin{bmatrix} 1416 \\ 3 \\ 2 \\ 40 \end{bmatrix} \quad x_3^{(2)} = 2$$

Hypothesis Function

Original

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1$$

Adapted

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_4$$

Linear Regression

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n$$

$$\underline{X} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad \underline{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix}$$

$$[\theta_0 \ \theta_1 \ \theta_2 \ \dots \ \theta_n] \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$h_{\theta}(x) = \underline{\theta}^T \underline{X}$$

Gradient Descent

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n$$
$$J(\underline{\theta}) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^i) - y^i)^2$$

Algorithm Adaptation

For all ($n \geq 1$)

Simultaneously update Θ_j

Repeat:

$$\Theta_j = \Theta_j - \alpha \underbrace{\frac{1}{m} \sum_{i=1}^m (h_{\Theta}(x^i) - y^i) x_j^{(i)}}_{\frac{\partial}{\partial \Theta_j} J(\Theta)}$$

$$\frac{\partial}{\partial \Theta_j} J(\Theta)$$

Where as for $j=0$

$$\Theta_0 = \Theta_0 - \alpha \frac{\partial}{\partial \Theta_0} J(\Theta)$$

$$\text{w/ } x_0^{(i)} = 1$$

Feature Scaling

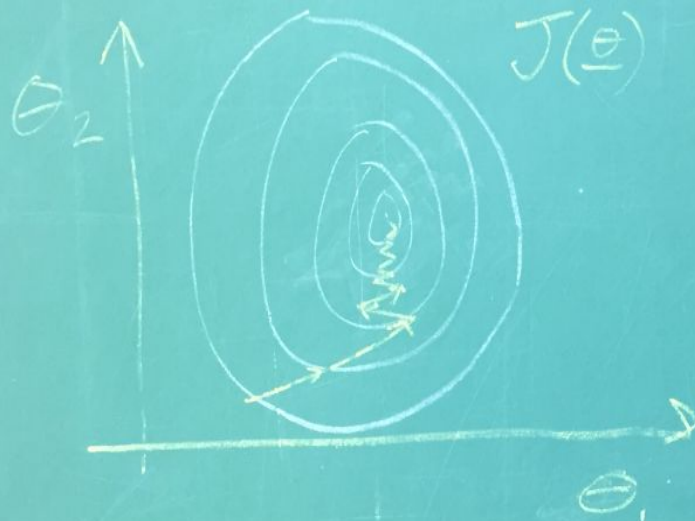
Benefits of proper feature scaling:

- Faster time reaching the global minimum

Feature Scaling

Let $X_2 = \text{size (0-2000 feet}^2\text{)}$

$X_1 = \text{number of bedrooms (1-5)}$



$$X_2 = \frac{\text{size (feet}^2\text{)}}{2000}$$

$$X_1 = \frac{\# \text{ of bedrooms}}{5}$$



Objective Approximately keep the range to $-1 \leq X_i \leq 1$

Mean Normalization

Let $\mu_{\text{size}} = 1000$ and $\mu_{\text{bedrooms}} = 2$

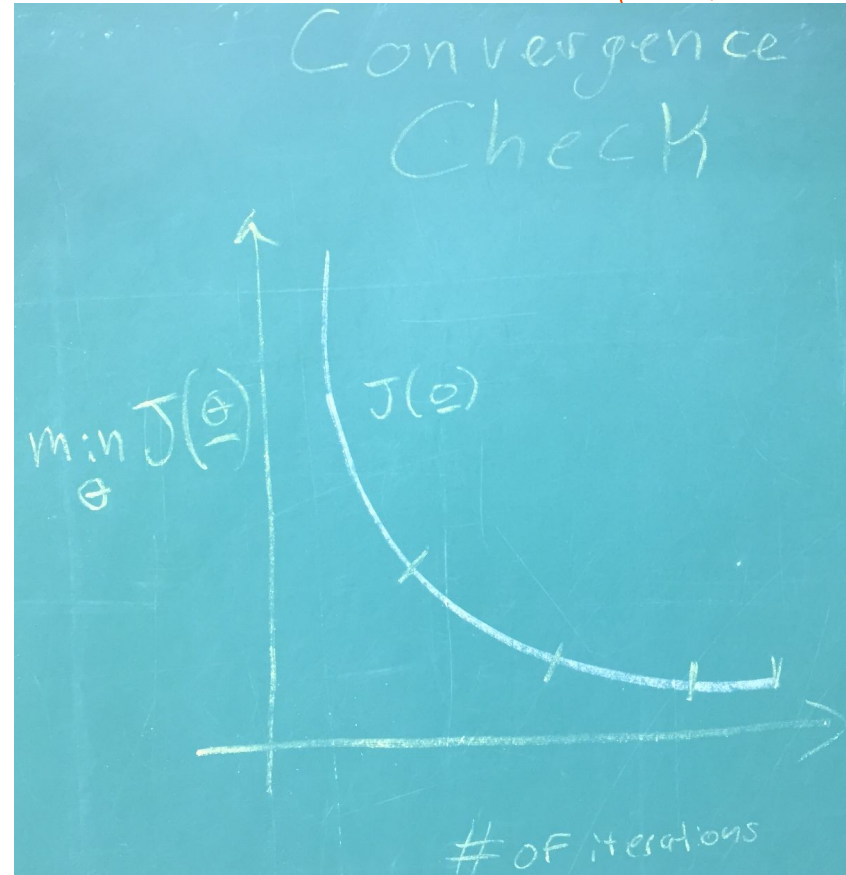
where μ is the ave. value of X_i
and S is the range or Std. dev.

then $X_1 = \frac{X_1 - \mu_{\text{size}}}{S_1}$ and

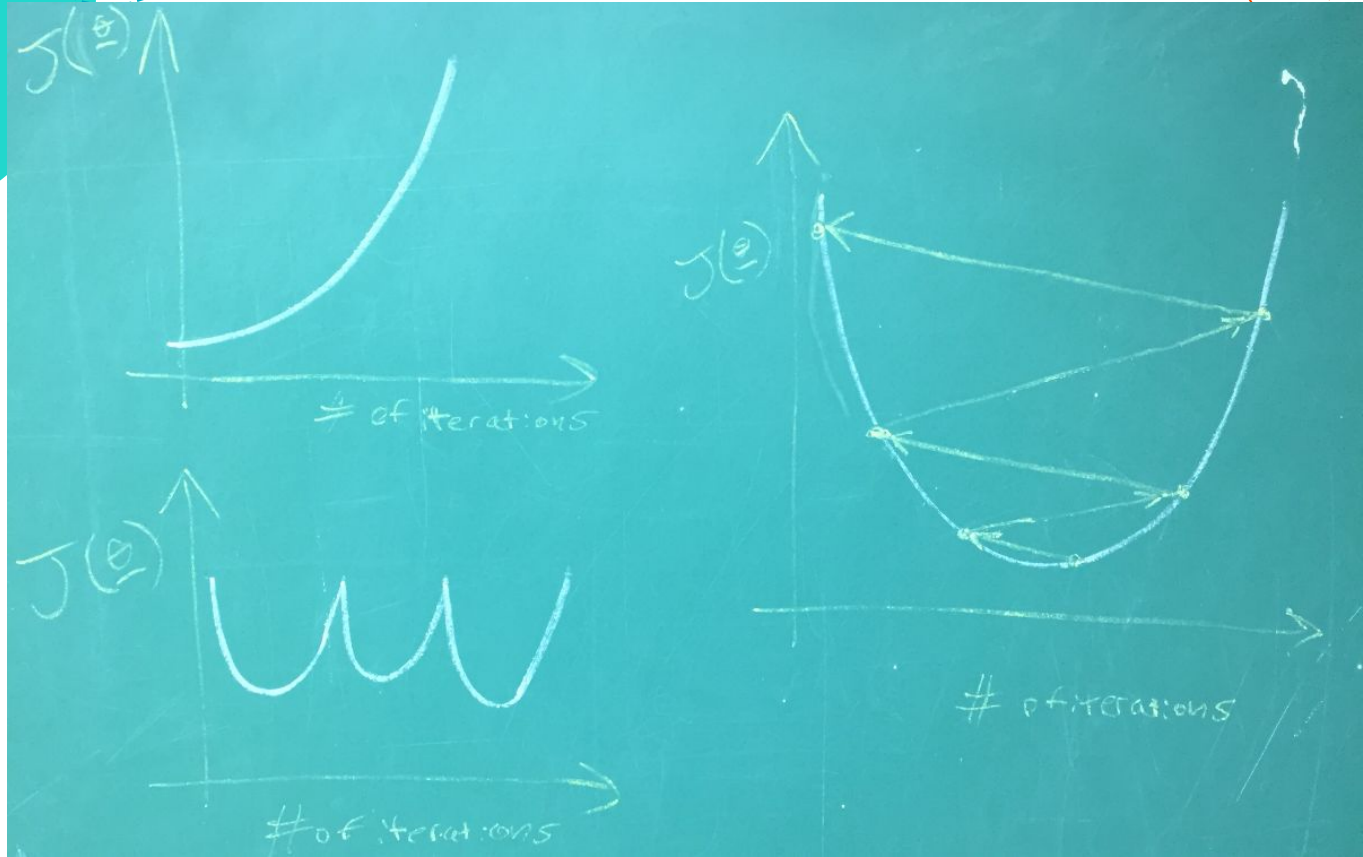
$$X_2 = \frac{X_2 - \mu_{\text{bedrooms}}}{S_2}$$

Gradient Descent Checking

- Converges if the cost decreases by less than 10^{-3} between 2 iterations



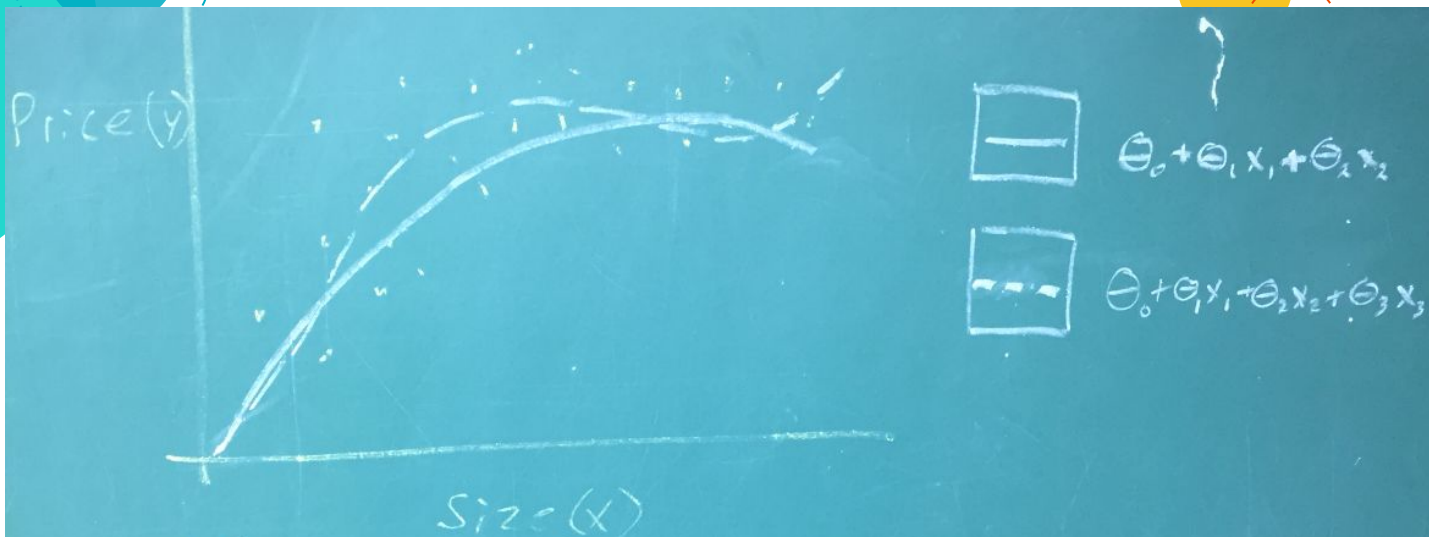
Learning Rate



Polynomial Regression

- Pick a function to closely characterize the feature pattern within the data

Polynomial Regression



E.G.

$$h_{\theta}(x) = \theta_0 + \theta_1 X_1 + \theta_2 X_2 + \theta_3 X_3$$

whereas $X_1 = \text{size}$

$$X_2 = \text{size}^2$$

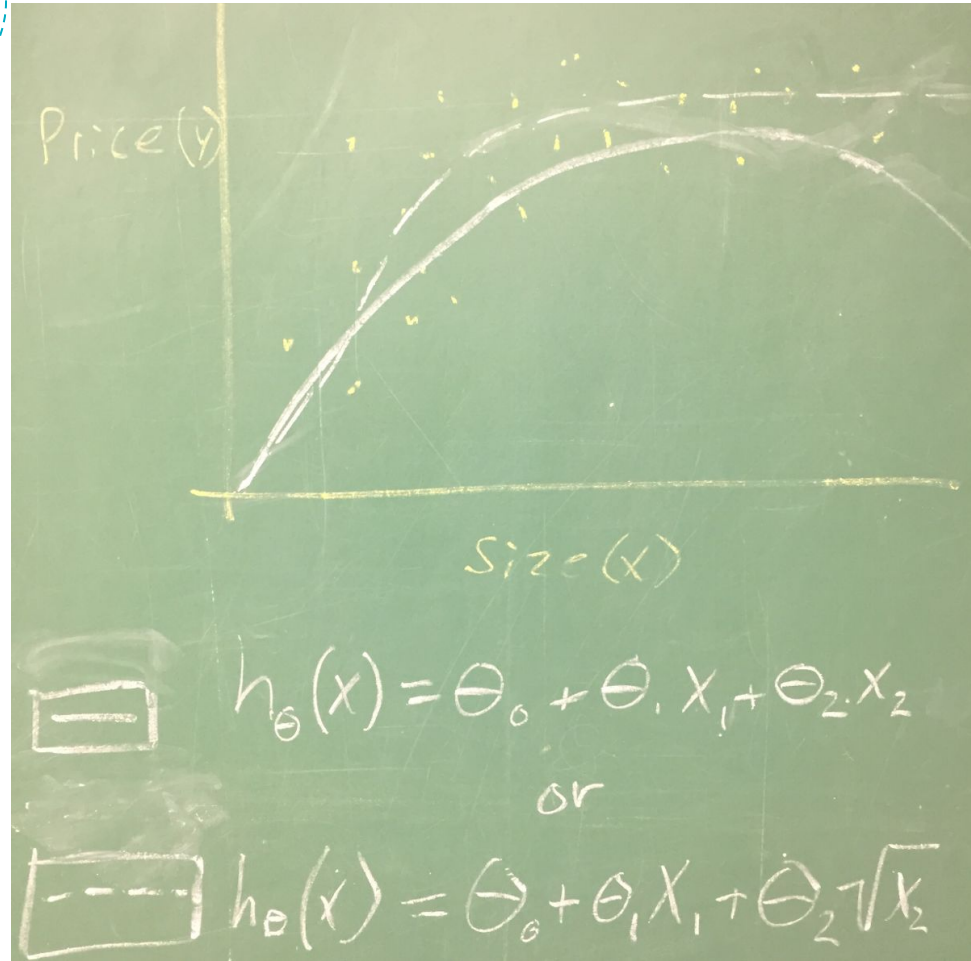
$$X_3 = \text{size}^3$$

$$1 - 1,000$$

$$1 - 1,000,000$$

$$1 - 10^9$$

Polynomial Regression



Normal Equation

- Analytically solve for theta using matrices
 - If theta is a real number
 - Set the derivative to 0 and solve
 - Else set the partial derivative to 0 and solve

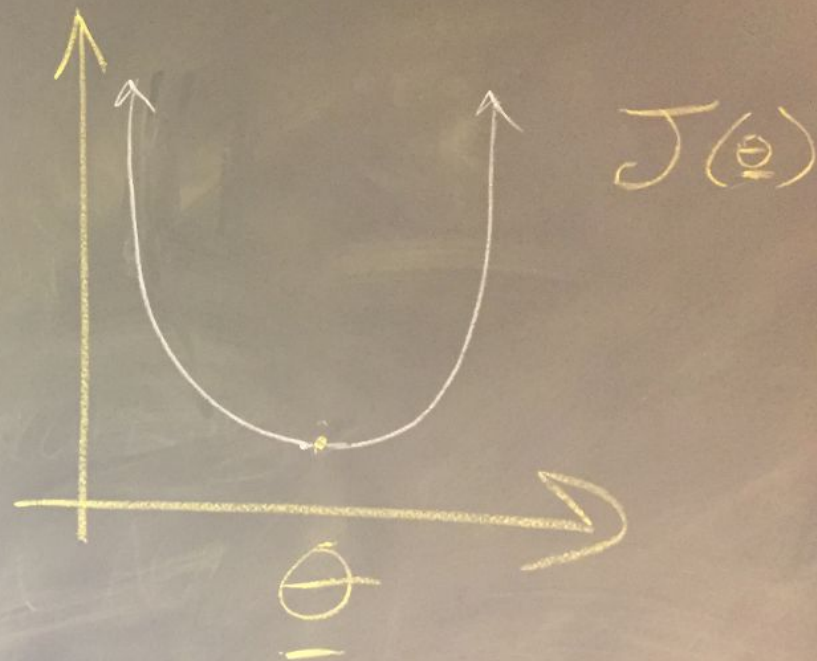
Normal Equation

If 1D ($\theta \in \mathbb{R}$)

$$J(\theta) = a\theta^2 + b\theta + c$$

$$\frac{d}{d\theta} J(\theta) = \dots = 0$$

Solve for θ



Normal Equation

$$J(\underline{\theta}) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

$\frac{\partial}{\partial \theta_j} J(\underline{\theta}) = 0$ and solve
at every j

Example

Let $m=4$ and $X_0=1$

X_0	Size (feet ²) X_1	Number of bedrooms X_2	Number of Floors X_3	Age of House (years) X_4	Price (\$1000) y
1	2104	5	1	45	460
1	1416	3	2	40	232
1	1534	3	2	30	315
1	852	2	1	36	178

Example (continued)

$$\underline{X} = \begin{bmatrix} 1 & 2104 & 5 & 1 & 45 \\ 1 & 1416 & 3 & 2 & 40 \\ 1 & 1534 & 3 & 2 & 30 \\ 1 & 852 & 2 & 1 & 36 \end{bmatrix} \quad \underline{y} = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix}$$
$$\underline{\theta} = (\underline{X}^T \underline{X})^{-1} \underline{X}^T \underline{y}$$

Comparisons

Gradient Descent

- Chosen alpha component
- Many Iterations
- Works well with large n (10,000)
- Works well with data of features > 1000

Normal Equations

- No chosen alpha component
- No iterations
- Slow when n is large 1,000
- Works well with the number of features < 1000



Non-Invertibility

If some matrix X times its transpose is invertible the strategy is:

- Delete some features (especially ones that are redundant)
- Use regularization
- Generalized Inverse



Sources

- © <https://www.ritchieng.com/>
- © <https://towardsdatascience.com/super-simple-machine-learning-by-me-multiple-linear-regression-part-1-447800e8b624>
- © http://www.dmi.unict.it/farinella/SMM/Lectures/25_Nov2015_2.pdf