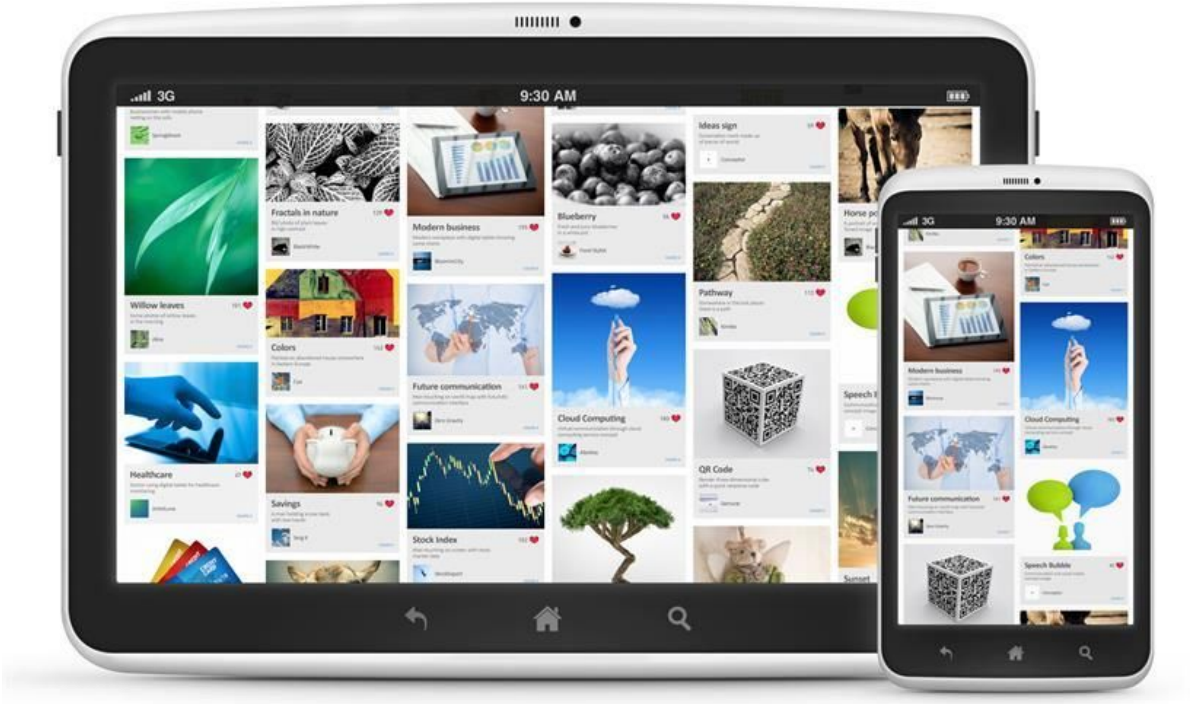
IUT AI Student Chapter

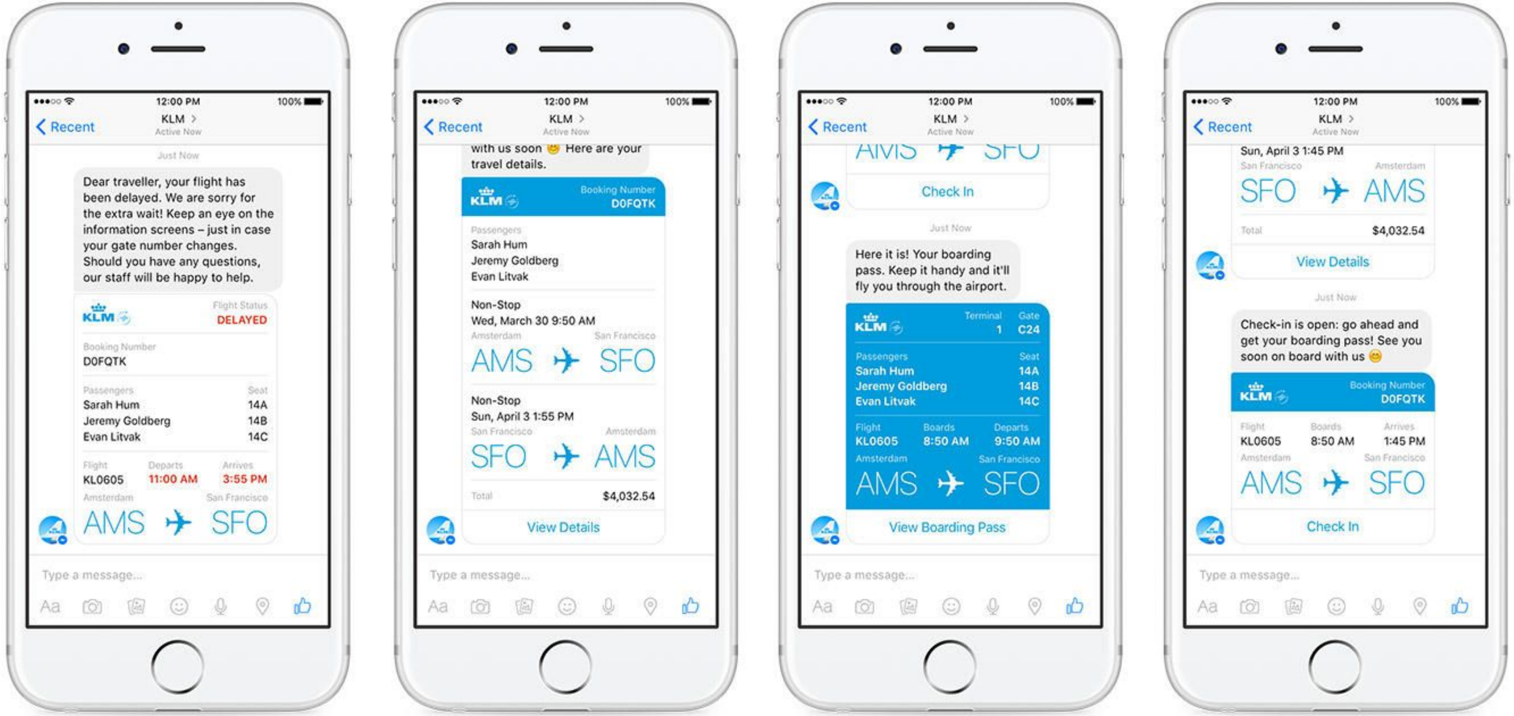
Introduction to Machine Learning

Applications



Pinterest – Improved Content Discovery

Applications (Cont.)



Facebook – Chatbot Army

Applications (Cont.)

Twitter – Curated Timelines

“

***Machine Learning** is a subfield of computer science that gives “computers the ability to learn without being explicitly programmed.”*

“

A more modern definition:

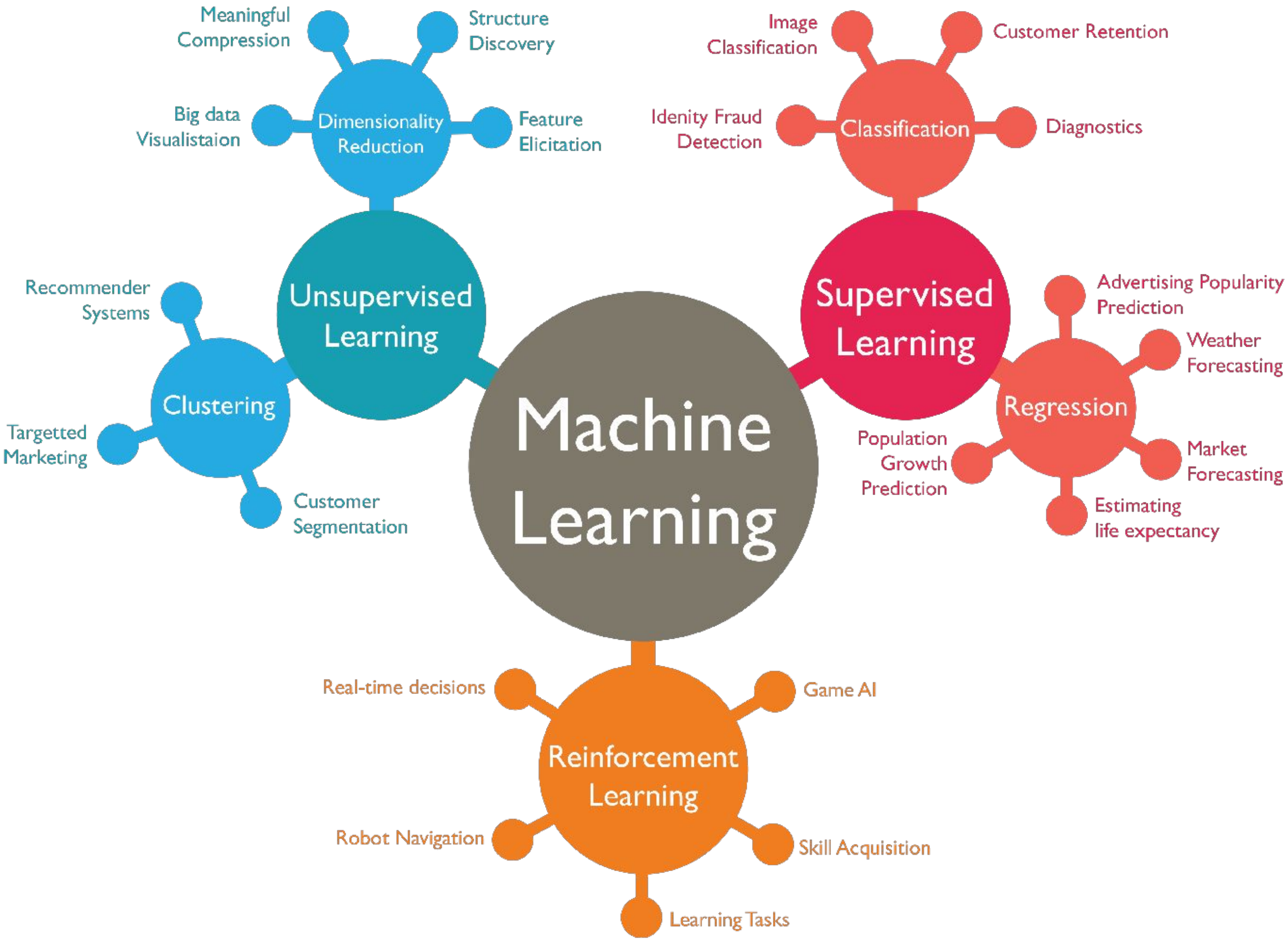
A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

Playing Chess

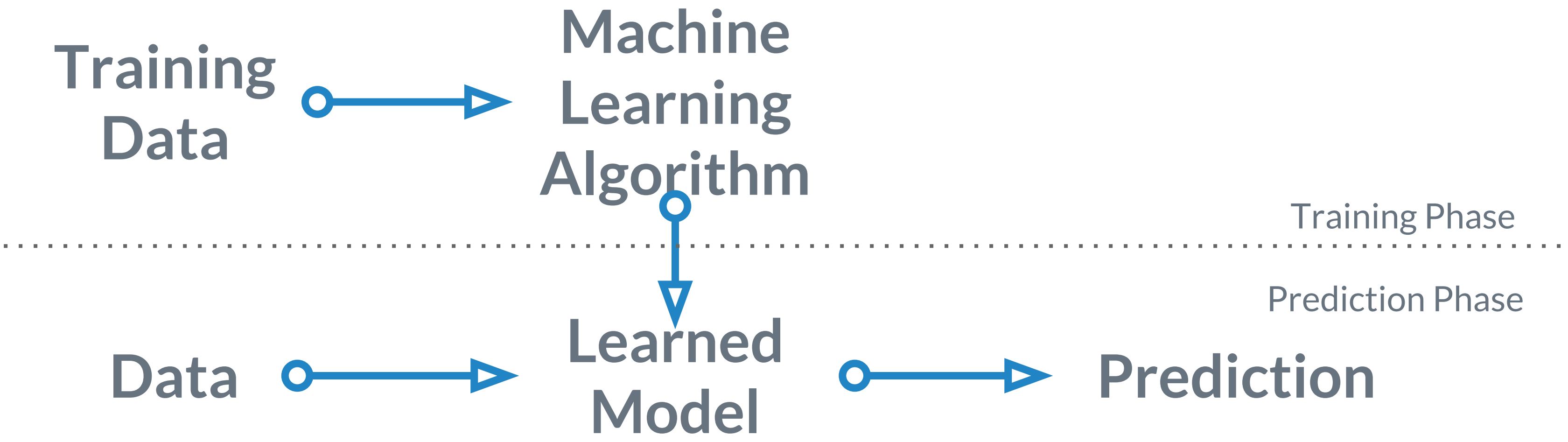
E = the experience of playing many games of chess

T = the task of playing chess.

P = the probability that the program will win the next game.


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Machine Learning



IUT AI Student Chapter

Supervised Learning

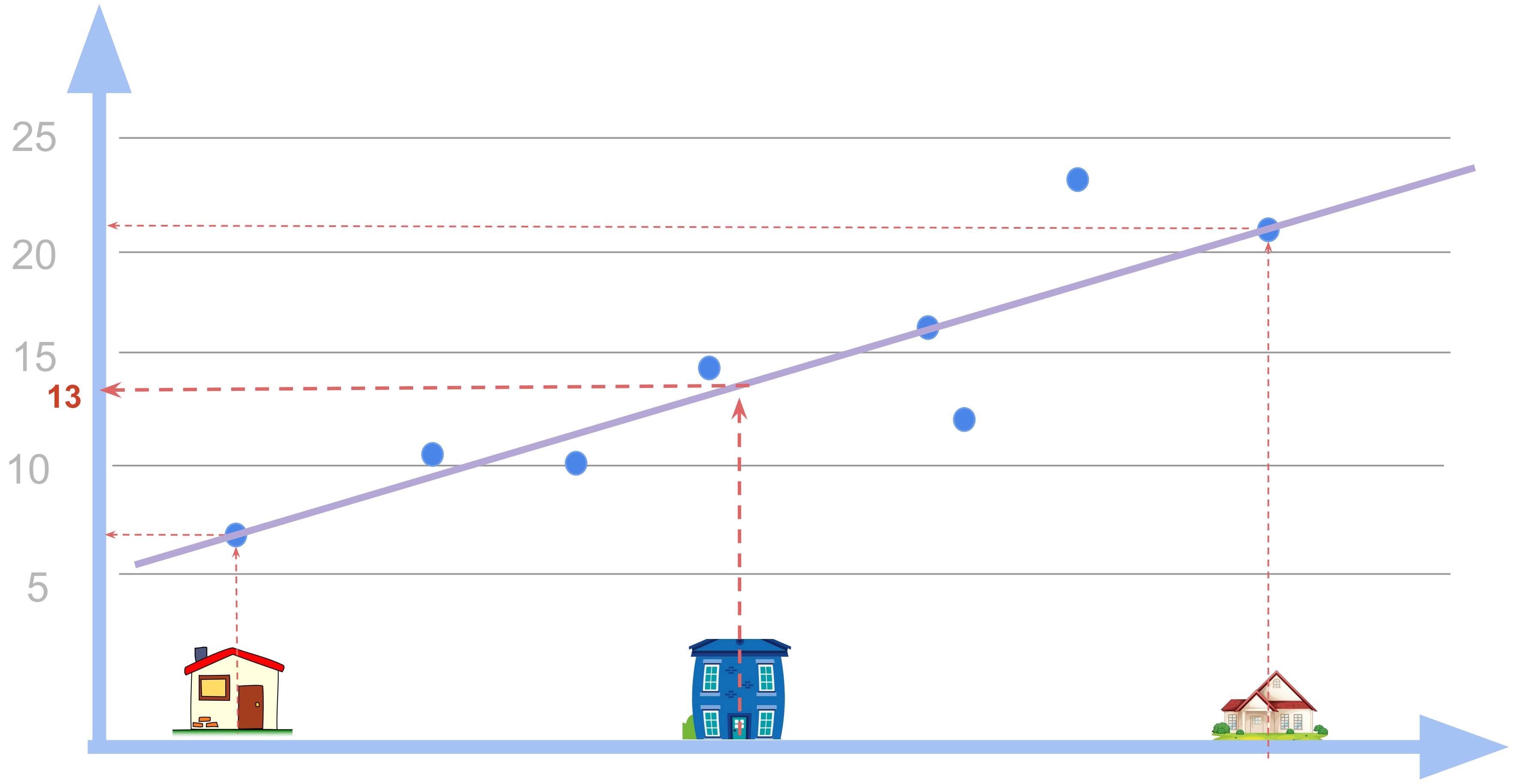
Linear Regression

Pricing Houses

70000 $

?

210000 $



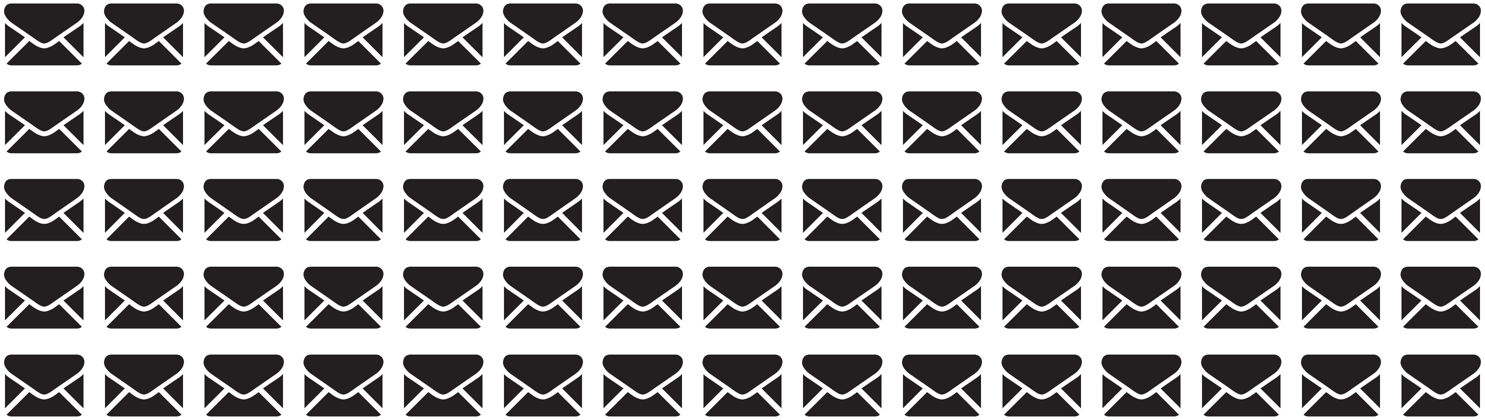
25
20
15
13
10
5

IUT AI Student Chapter

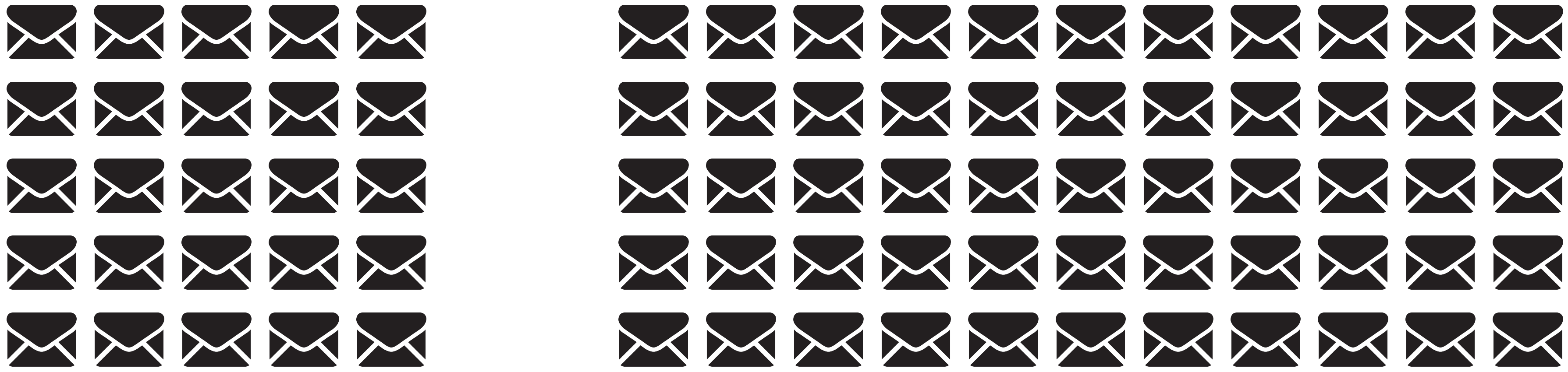
Supervised Learning

Classification

Mail Spam/not Spam Example



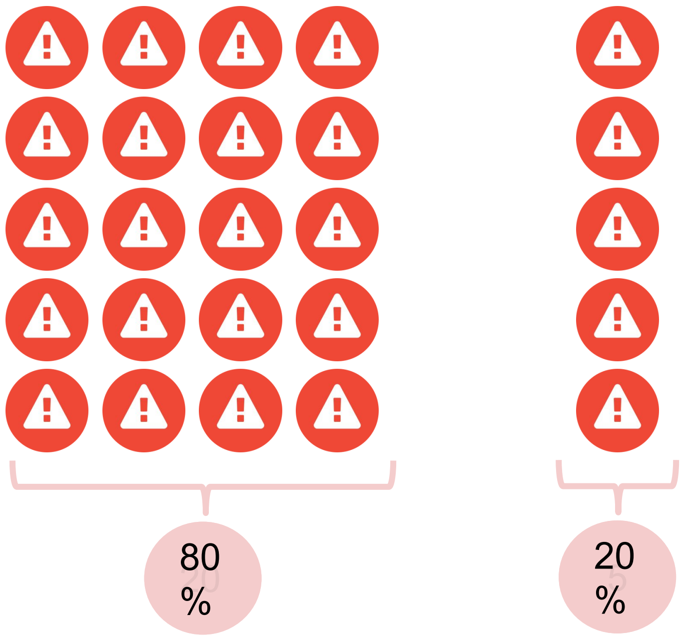
Detecting Spam Email



Detecting Spam Email

Detecting Spam Email

“cheap”



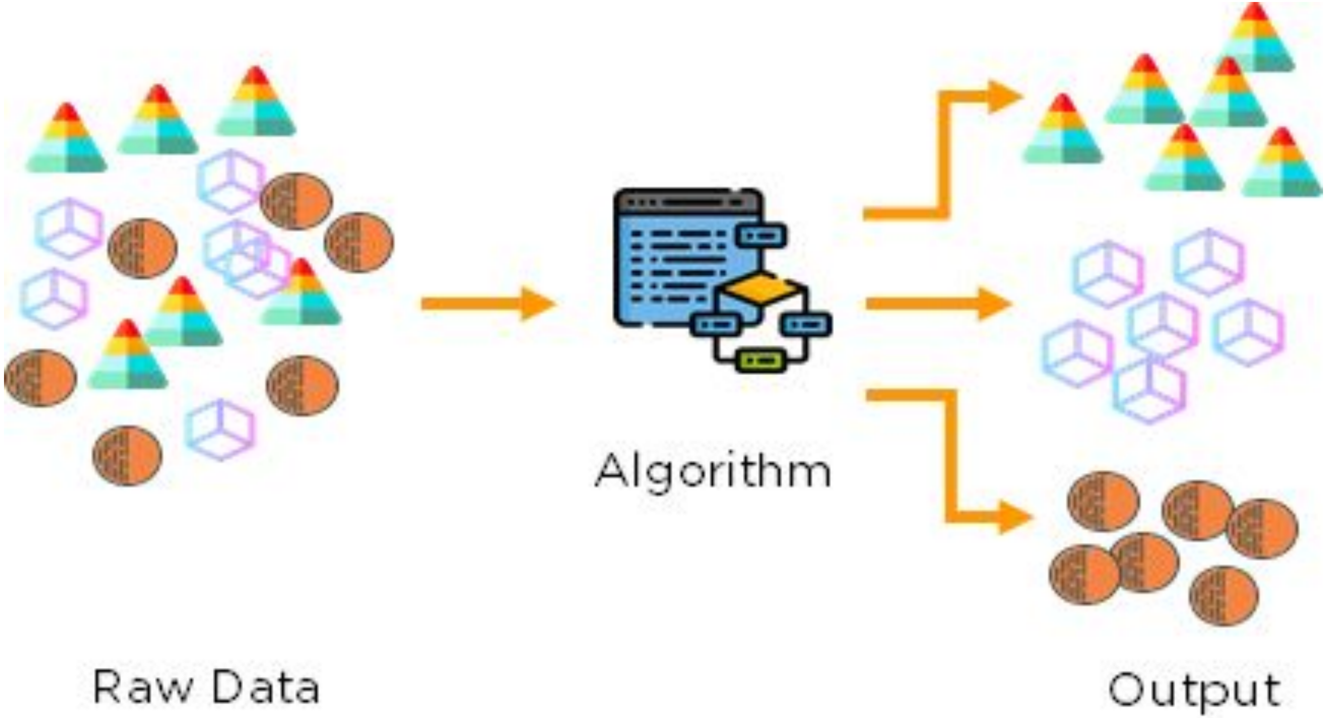
IF an email contain the word “cheap” , what is the probability of it being spam?

conclusion:
If an email contain the word “cheap” The probability of it being spam is 80 %

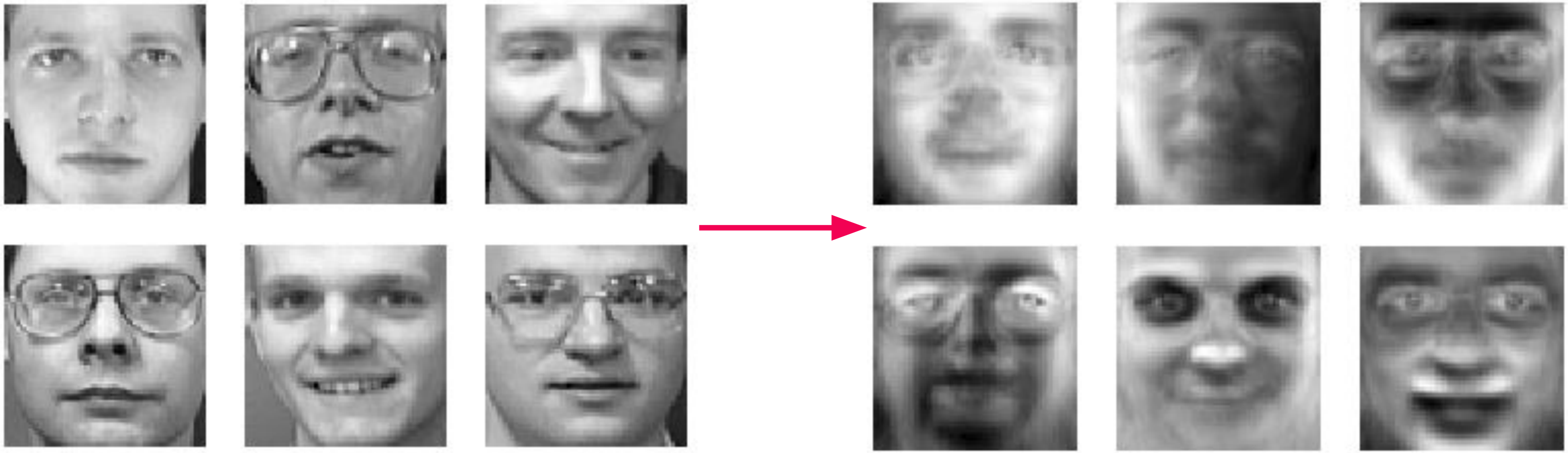
IUT AI Student Chapter

Unupervised Learning

Clustering



Dimensionality Reduction



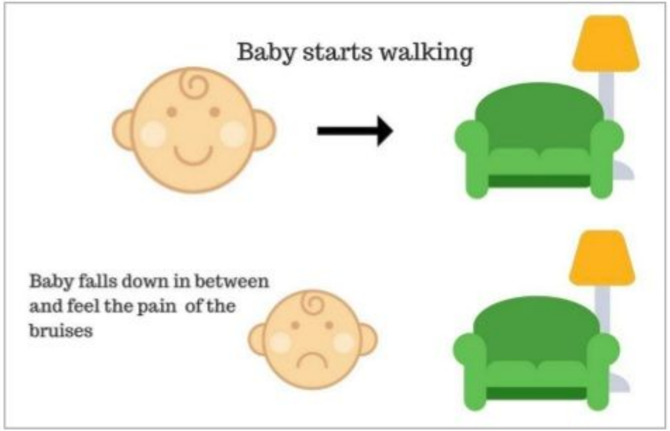
IUT AI Student Chapter

Reinforcement Learning

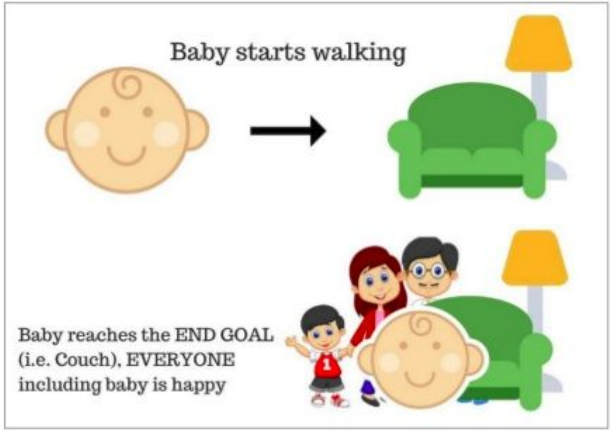
Reinforcement vs Supervised

Supervised learning is "teach by **example**":
Here's some examples, now learn patterns in these example.

Reinforcement learning is "teach by **experience**":
Here's a world, now learn patterns by exploring it.



Failure



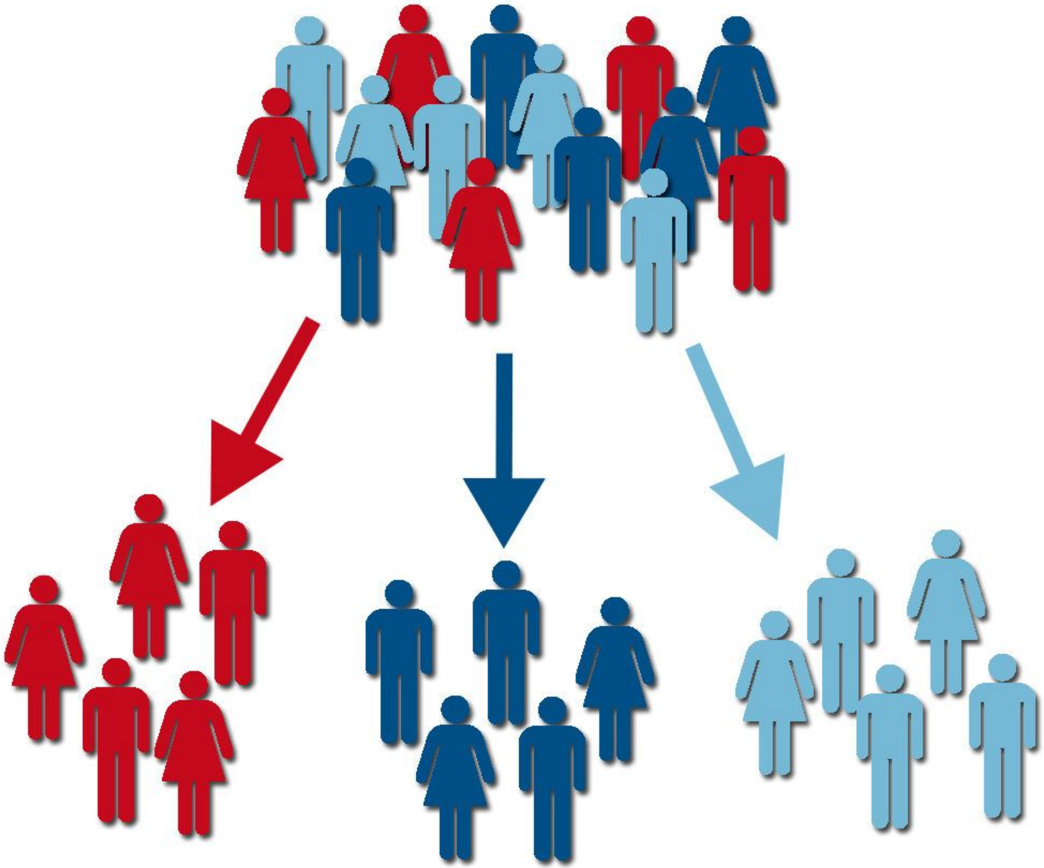
Success

Now, It's your turn :)

- What types of problems are these?

Weather Prediction

Disease Prediction



Customer Segmentation

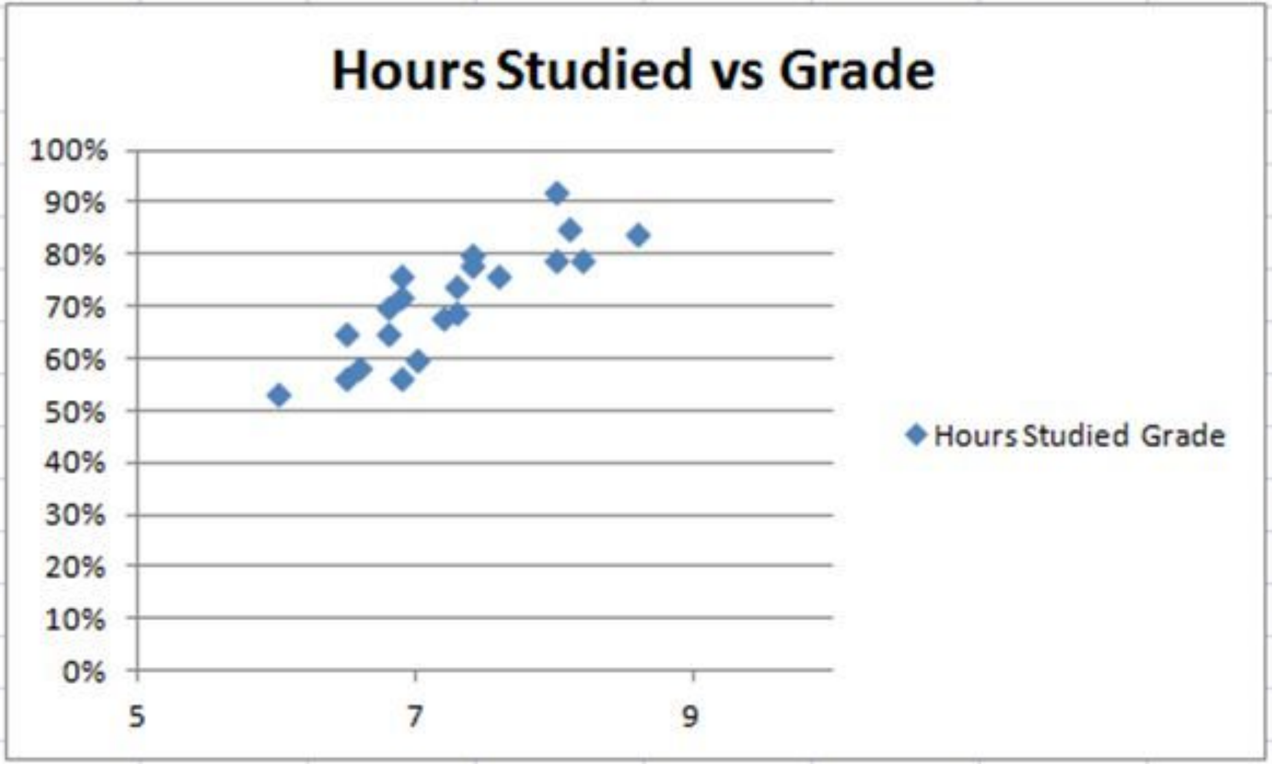
IUT AI Student Chapter

Linear Regression

Regression

- How can we predict the grade of a student?
    - We need to know the features
    - Also the outcome

Grade prediction



Jargon

- IQ, hours studied, … are *Features*
- Grade is called *Label*
- The dataset is called training set
- Features : ‘x’
- Labels : ‘y’
- Predictions : ‘$\hat{y}$’

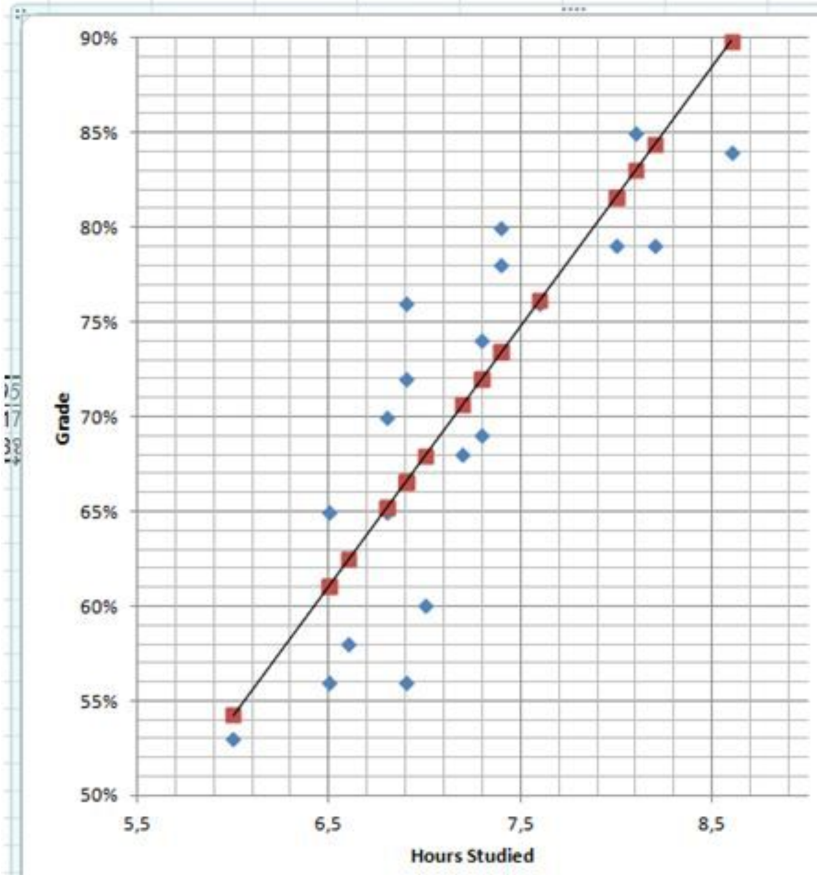
Hypothesis

$$H_\theta(X) = \theta_1 + \theta_2 X_i$$

Prediction :

$$\hat{y}_i = \sum_{i=1}^{m} x_{ij}\theta_j$$

- But how to find the best parameters?



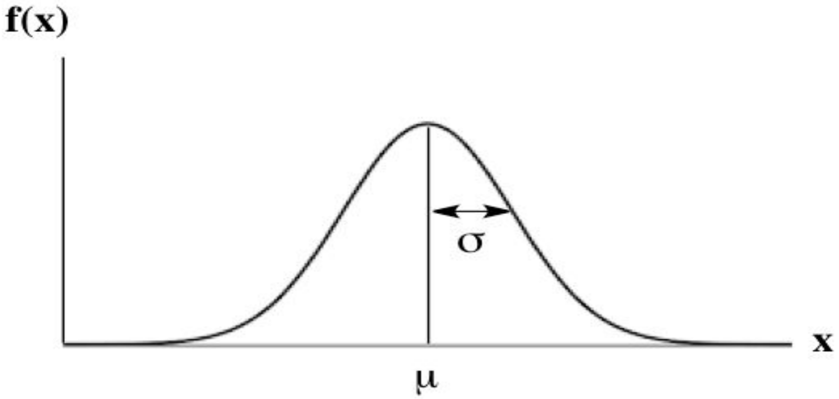
Loss Function

- a.k.a Loss, Objective, Error, Cost, Energy
- Mean squared error

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$$

Where it comes from?

- P(X=x) : Probability Distribution
- Let's work with the Gaussian

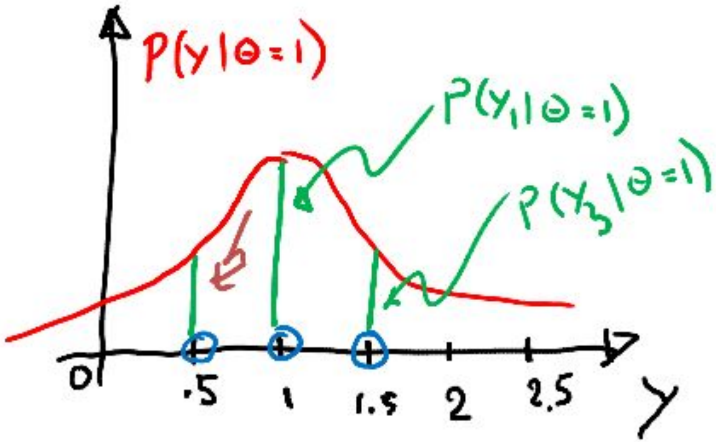


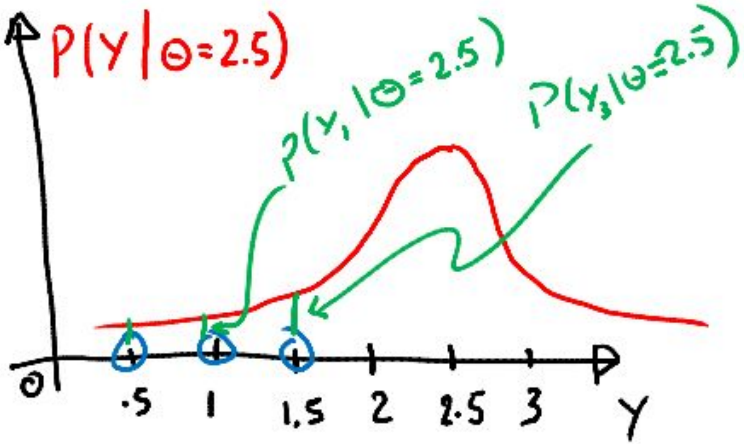
$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$$

likelihood

- Consider we have three data points y= 1 , 0.5, 1.5 and known variance of 1
- Let's guess Theta = 1 or Theta = 2.5?

$$p(y1, y2, y3|\theta) = p(y1|\theta)p(y2|\theta)p(y3|\theta)$$





likelihood

$$p(y|X,\theta,\sigma) = \prod_{i=1}^{n} p(y_i|x_i,\theta,\sigma)$$

$$\rightarrow \prod_{i=1}^{n} (2\pi\sigma^2)^{-1/2}\ e^{-\frac{1}{2\sigma^2}(y_i - x_i^T\theta)^2}$$

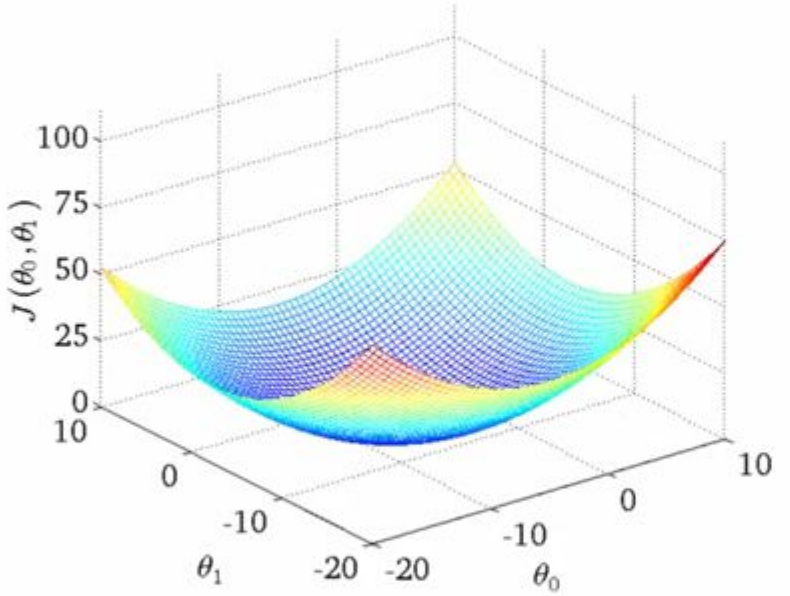
$$\rightarrow (2\pi\sigma^2)^{-n/2}\ e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - x_i^T\theta)^2}$$

- Objective = to maximize the likelihood
- What if we take negative of logarithm?

Optimization

Goal: Minimize J(θ)

Finding the exact answer can be infeasible when number of parameters increase



Gradient Descent

- Start with some initial value for Thetas
- Keep changing them to reduce J(θ)

→ Take steps proportional to the negative of the gradient

Gradient Descent

$$
\begin{aligned}
\frac{\partial}{\partial \theta_j} J(\theta) &= \frac{\partial}{\partial \theta_j} \frac{1}{2} \left(h_\theta(x) - y\right)^2 \\
&= 2 \cdot \frac{1}{2} \left(h_\theta(x) - y\right) \cdot \frac{\partial}{\partial \theta_j} \left(h_\theta(x) - y\right) \\
&= \left(h_\theta(x) - y\right) \cdot \frac{\partial}{\partial \theta_j} \left(\sum_{i=0}^{n} \theta_i x_i - y \right) \\
&= \left(h_\theta(x) - y\right) x_j
\end{aligned}
$$

repeat until convergence: {

$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_0^{(i)}$$

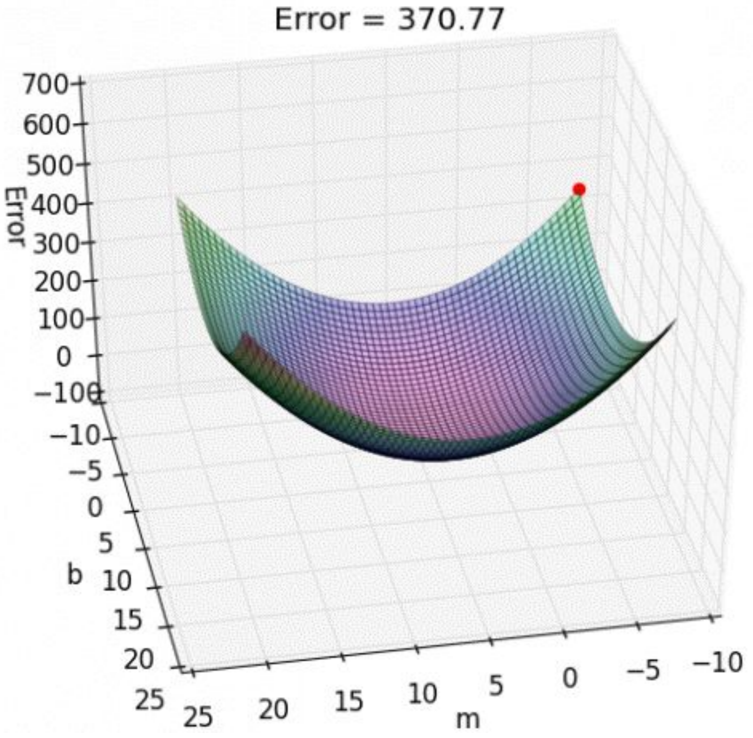
$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_1^{(i)}$$

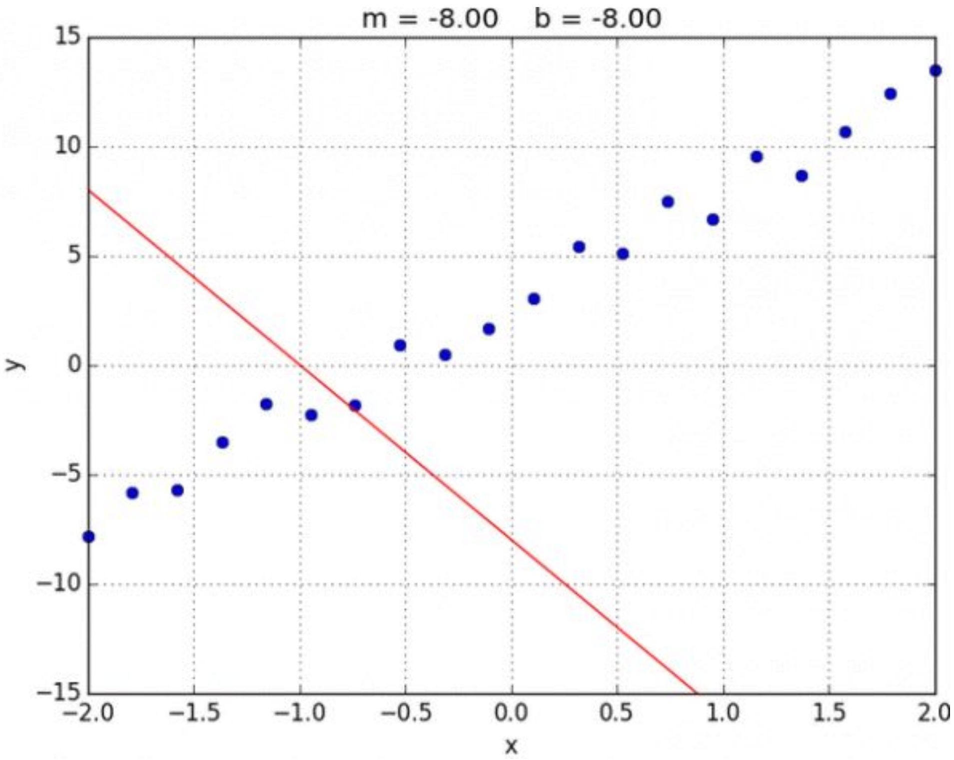
$$\theta_2 := \theta_2 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_2^{(i)}$$

...

}

GD visualization

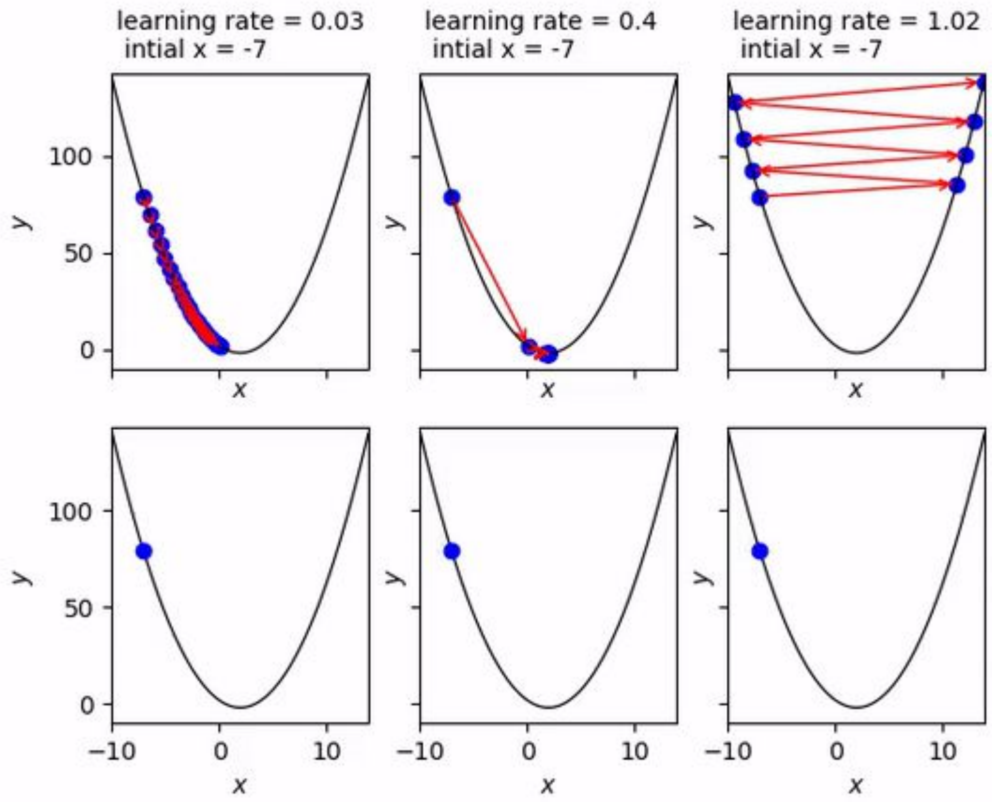




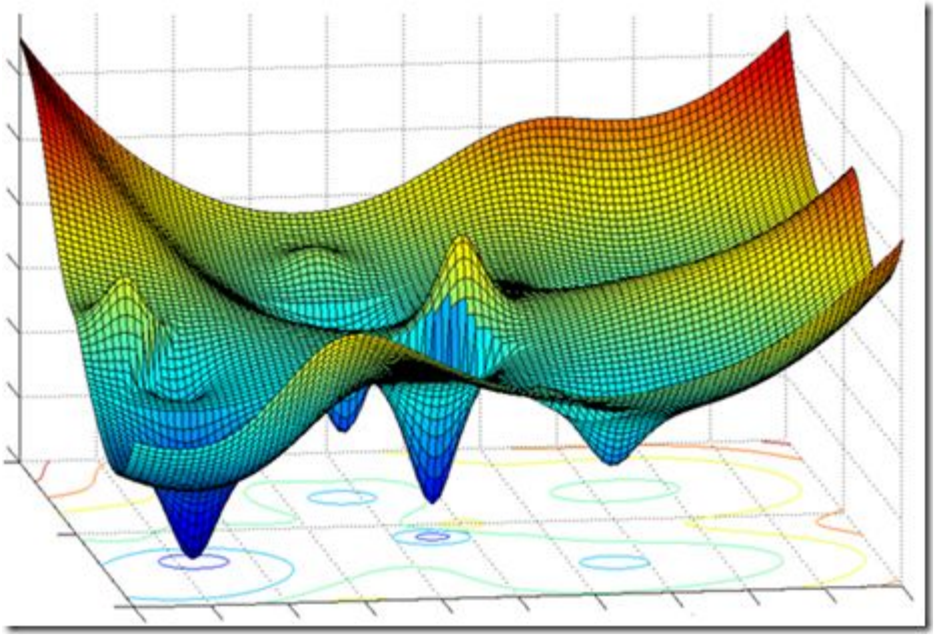
What was **α**?

Learning rate : how big the steps are.

(changing mind more quickly!)



When it fails?

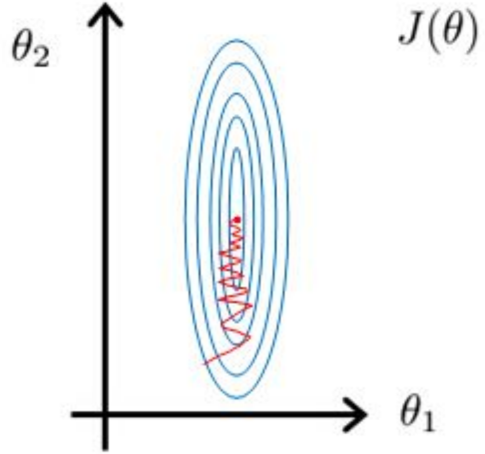


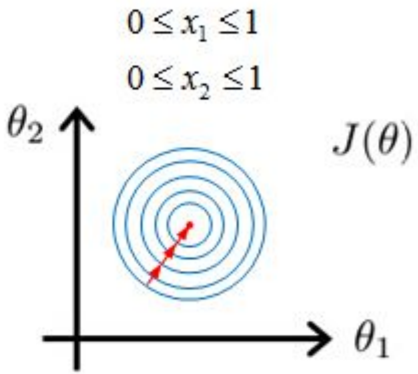
Feature scaling

Consider we have:

X1 = Area [0, 1000]

X2 = #rooms [0, 6]





- Normalization: Divide each feature by max of the feature column.
- Mean Normalization: $\frac{x_i - \mu_i}{x_{max}}$

Vectorization

Recall : $\hat{y}_i = \sum_{i=1}^{m} x_{ij}\theta_j$

Instead of using loops : $\hat{y} = X\Theta$

$$J(\theta) = \frac{1}{2m}(X\Theta - y)^T(X\Theta - y)$$

$$\frac{\partial J(\theta)}{\partial \theta} = \frac{1}{m}X^T(X\theta - y)$$

Faster, Simpler!