DS-Bootcamp(Day-1)

Agenda

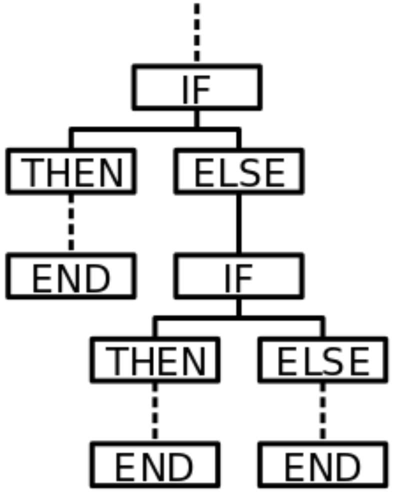
- What is machine learning?
- Who is Data Scientist?
- The three DS core
- ML 101 (Linear regression)
- Theory + Housing data set
- Model Evaluation & Validation

1 What is Machine Learning?

History

Business Expert

Automation Engineer/ Old school intelligent system



Intelligent System

What is Machine Learning

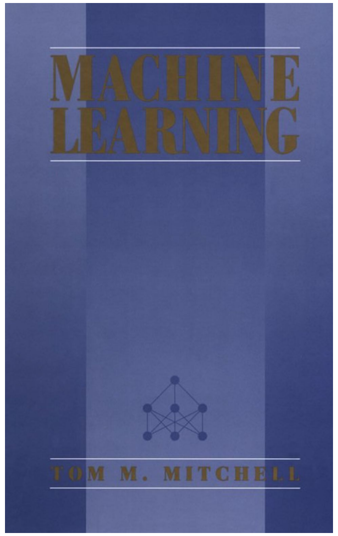
Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

- **Arthur Lee Samuel,**
  Computer Scientist, IBMer, Stanford Professor
  creator of Samuel Checkers-playing Program in 1959

What is Machine Learning

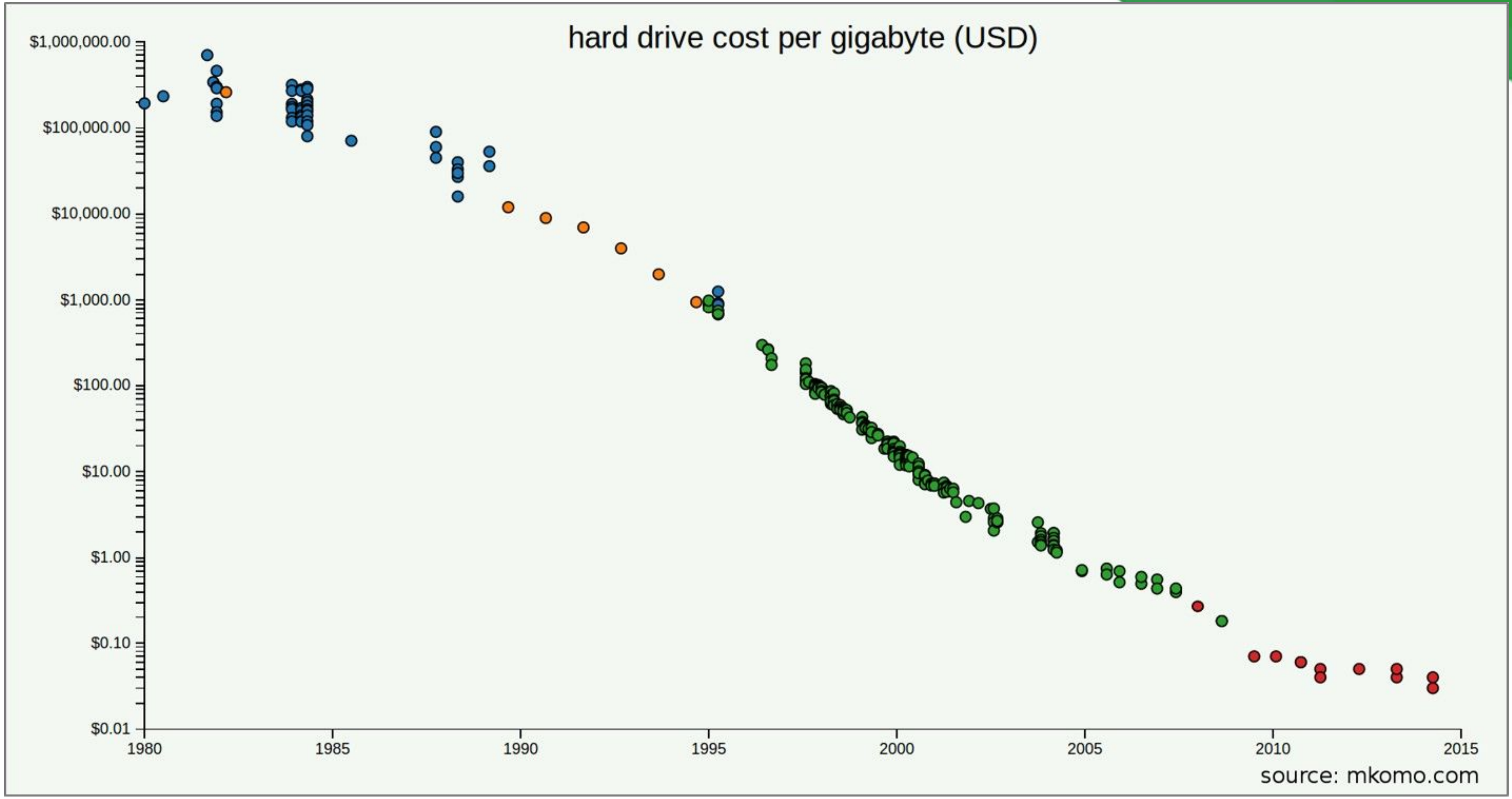
"A machine learns with respect to a particular task T, performance metric P, and type of experience E, if the system reliably improves its performance P at task T, following experience E."

- Tom Mitchell, Professor, Carnegie Mellon University



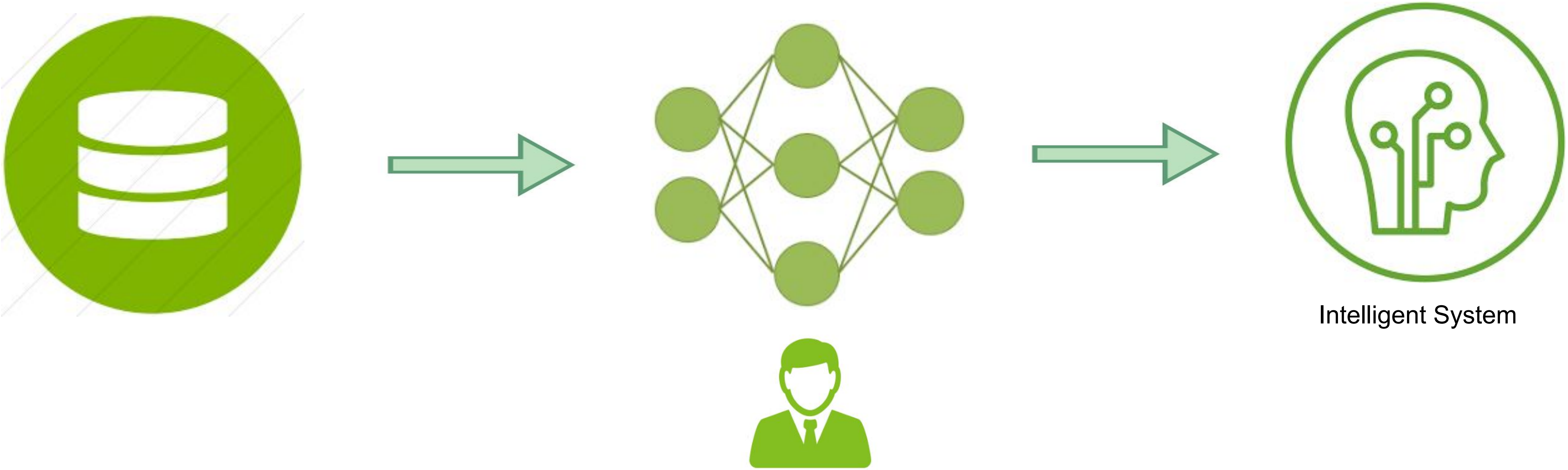
What Changed ?

- **Automation**
- **Storage cost**
- **Compute cost**



https://cloud.withgoogle.com/build/data-analytics/explore-history-machine-learning/

Today



2 Who is Data Scientist?

Who is a Data Scientist?

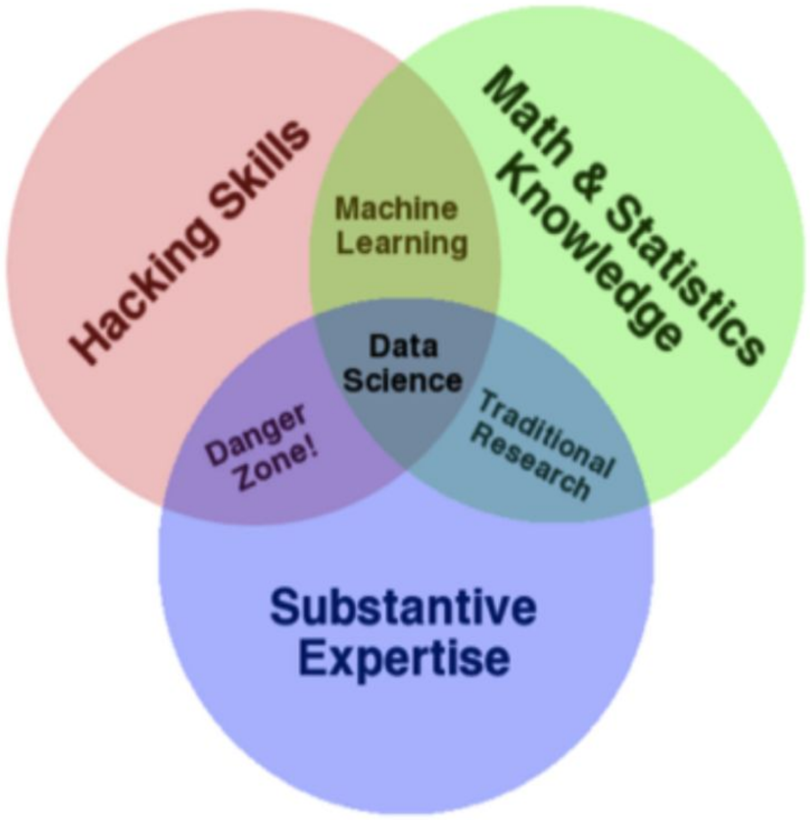
"A DATA SCIENTIST IS THAT UNIQUE BLEND OF SKILLS THAT CAN BOTH UNLOCK THE INSIGHTS OF DATA AND TELL A FANTASTIC STORY VIA THE DATA."

– **DJ Patil**,

The White House former Chief of Data Scientist.

3 Three DS Cores

The three DS core



Wide variance in terms of skillsets: many job descriptions are more appropriate for a **team of data scientists!**

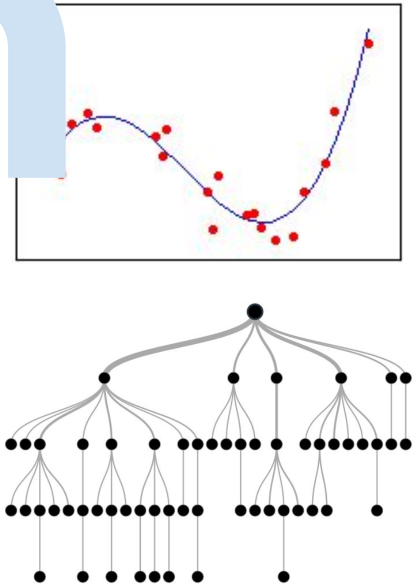
Data Science Workflow

Keep in mind that data science is an end to end process.

iterate

Understand the Business

Know your data



Apply Machine Learning

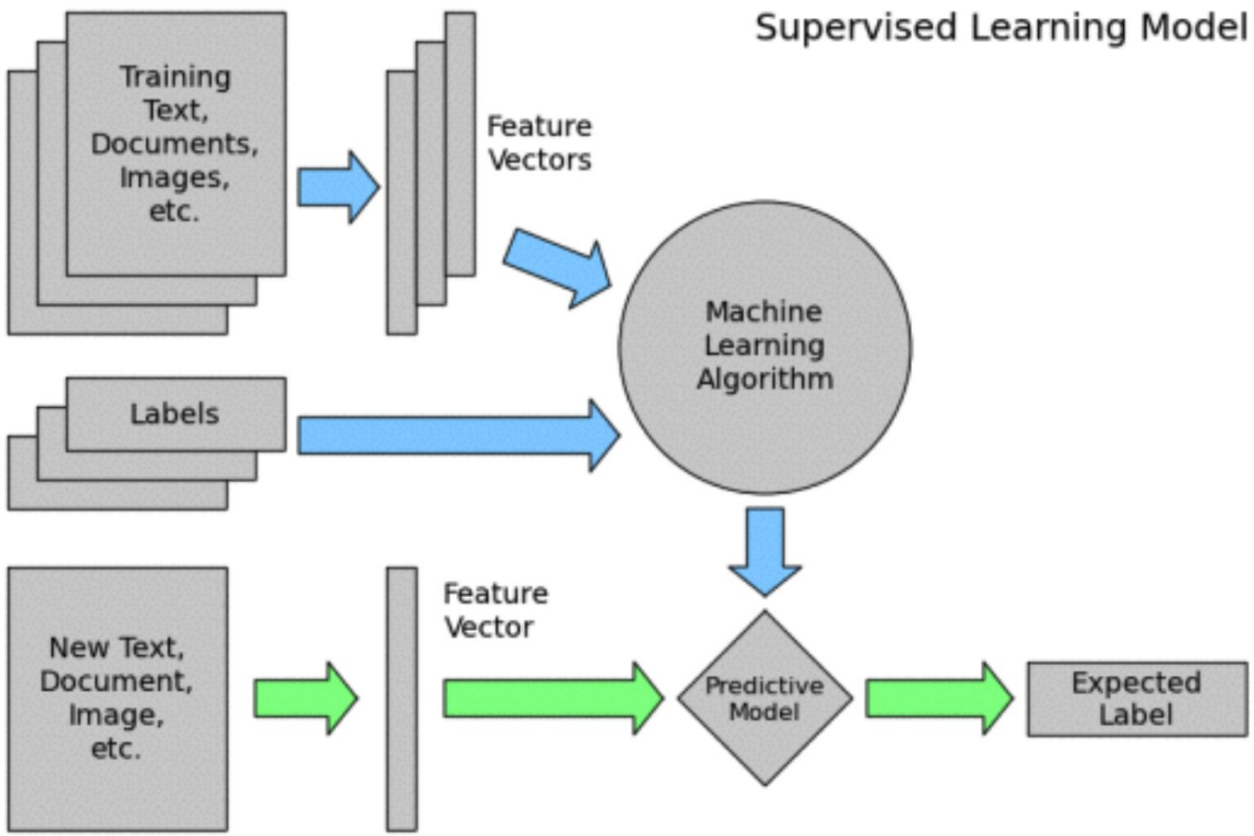
Communicate your result

4 ML 101

Machine Learning Problems

	Supervised Learning	***Unsupervised Learning***
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

Supervised Learning



Supervised Learning

150 observations *(n = 150)*

Fisher's *Iris* Data

Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	*I. setosa*
4.9	3.0	1.4	0.2	*I. setosa*
4.7	3.2	1.3	0.2	*I. setosa*
4.6	3.1	1.5	0.2	*I. setosa*
5.0	3.6	1.4	0.2	*I. setosa*
5.4	3.9	1.7	0.4	*I. setosa*
4.6	3.4	1.4	0.3	*I. setosa*
5.0	3.4	1.5	0.2	*I. setosa*

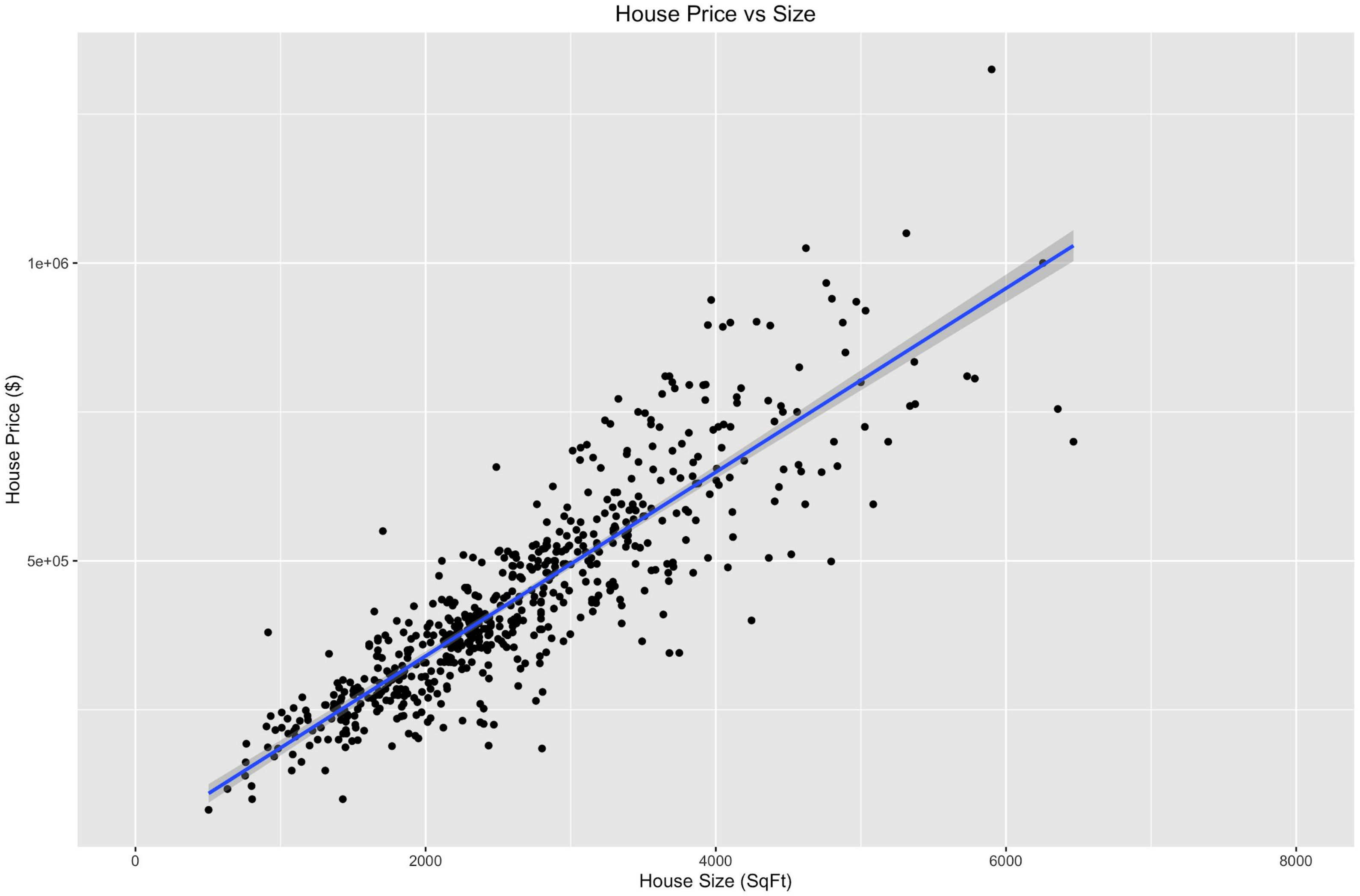
response

4 predictors (p = 4)

4.1

ML 101:
LINEAR REGRESSION

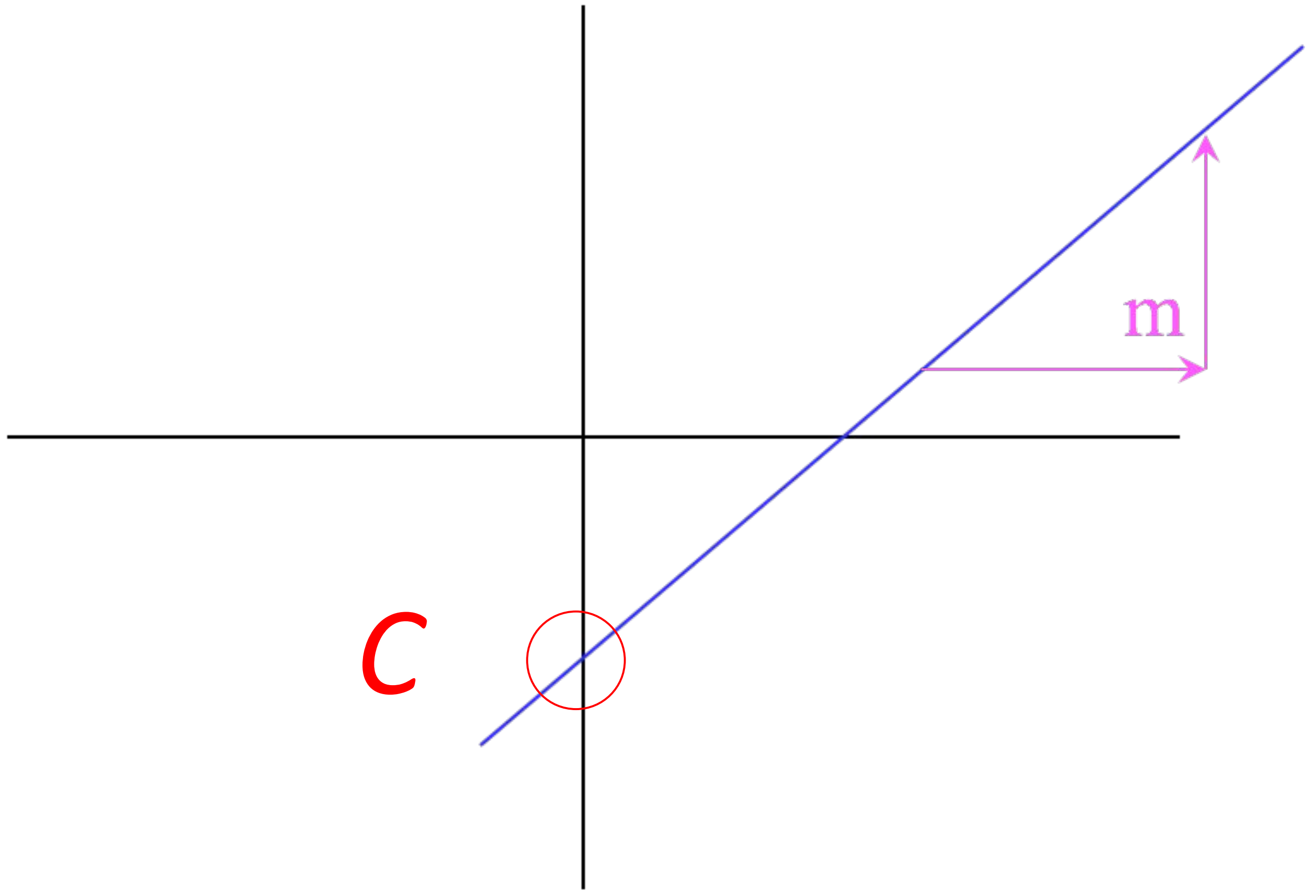
Linear regression



Linear regression

- Remember this ?

Y=mX+C?



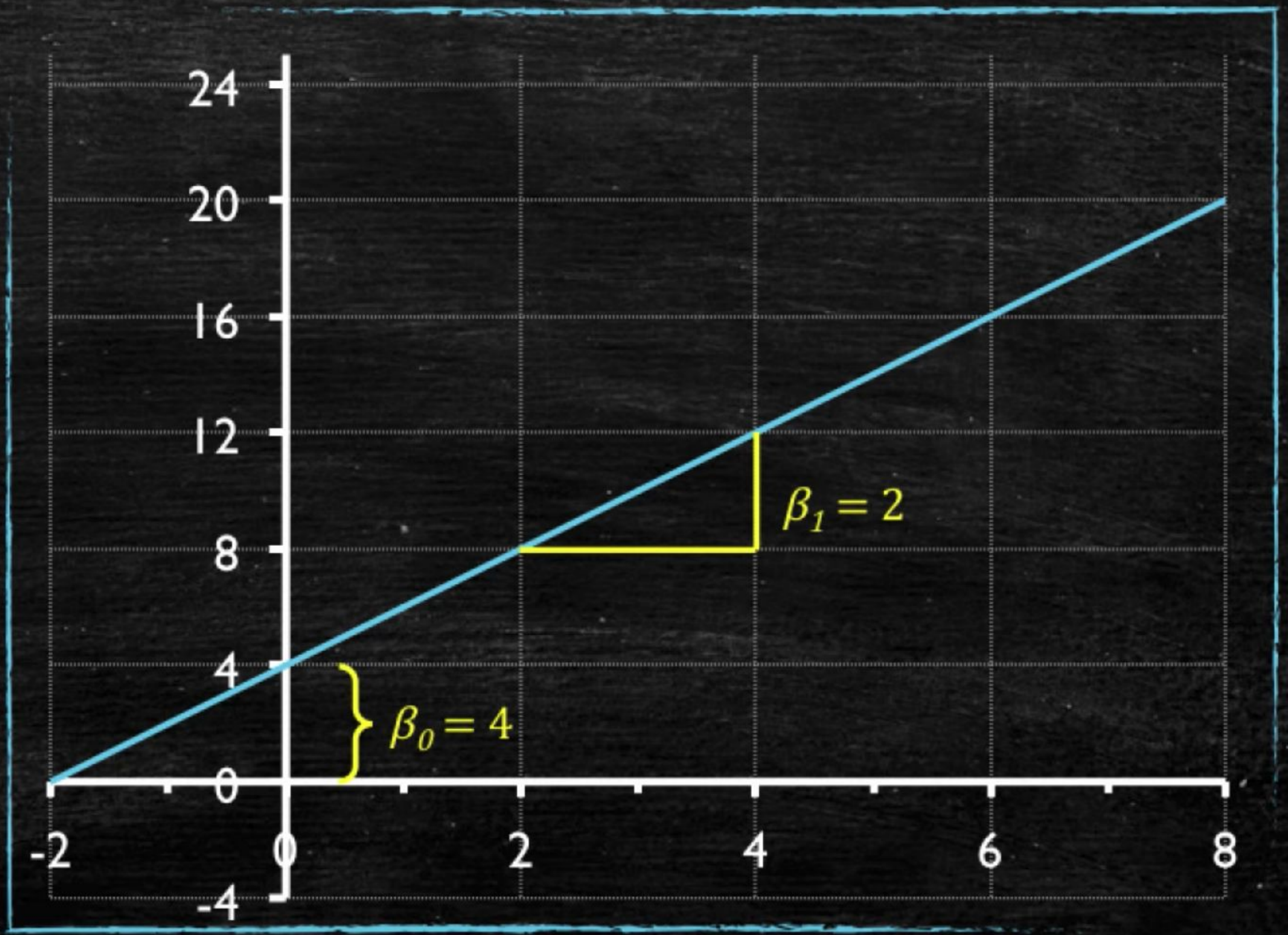
Linear regression

$$y = \beta_0 + \beta_1 x$$

Linear Equation Example

$$y = \beta_0 + \beta_1 x$$

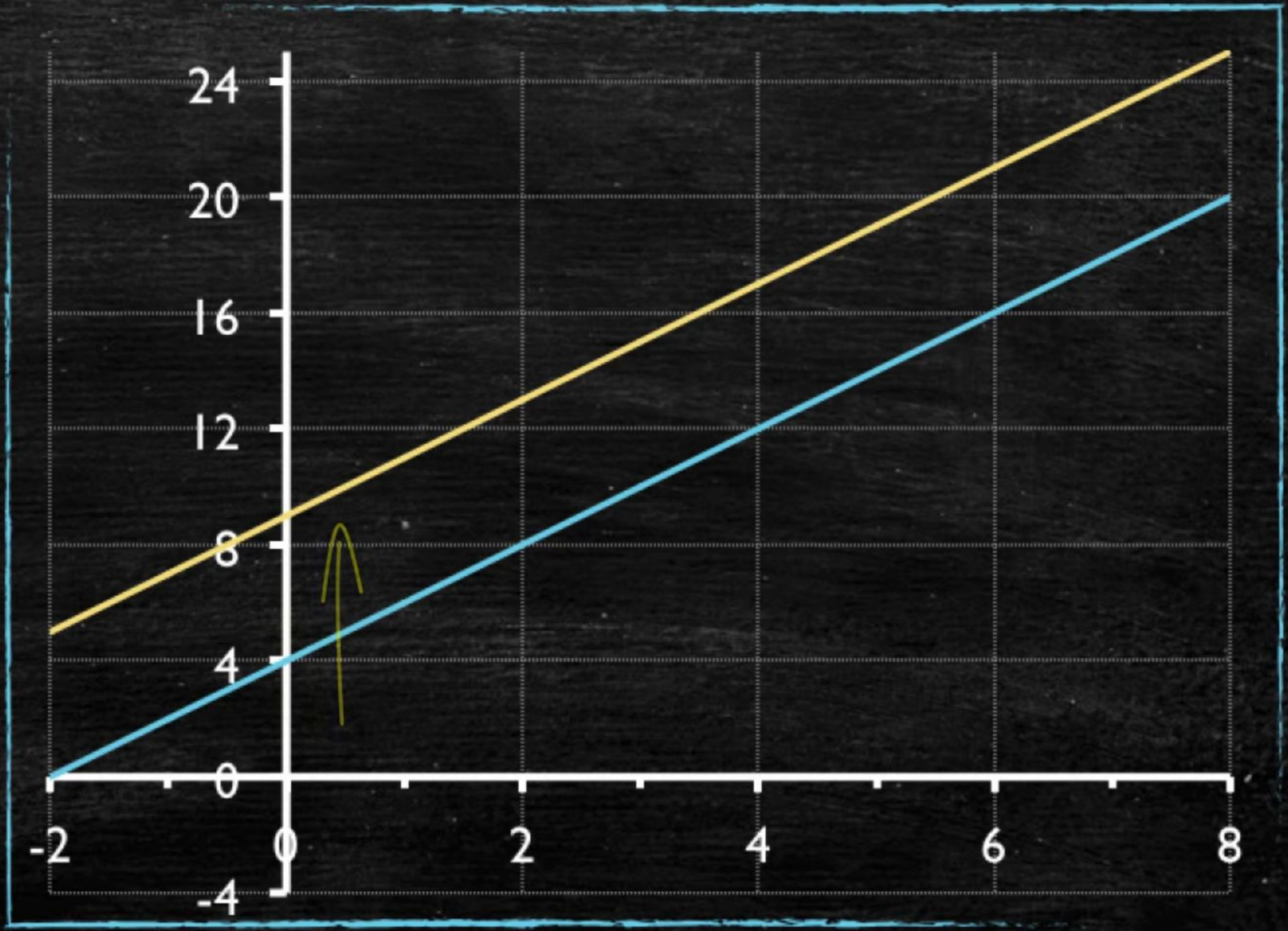
$$y = 4 + 2x$$



What happens is we change the intercept?

$$y = 4 + 2x$$

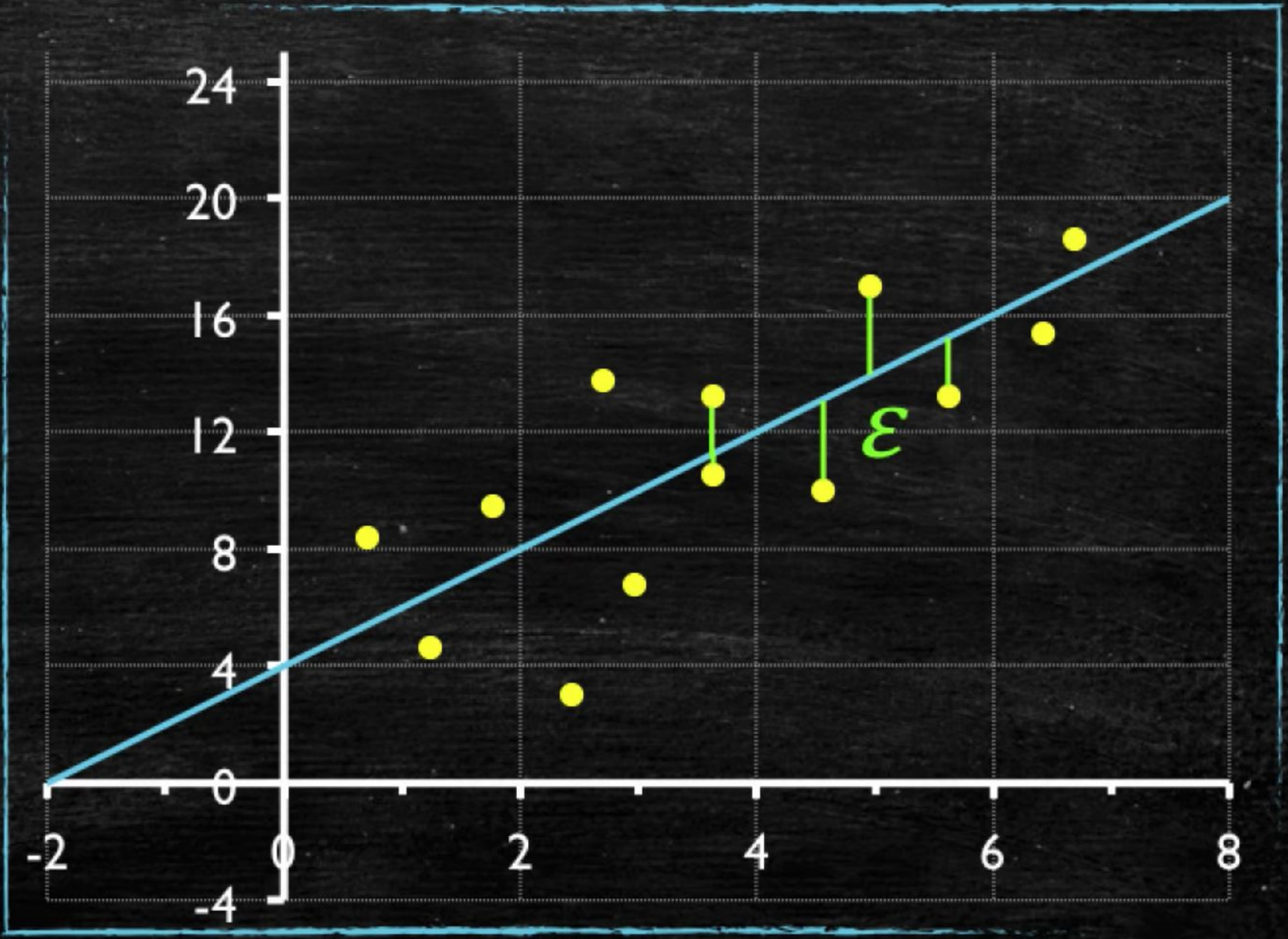
$$y = 9 + 2x$$



But the world is not linear!

$y = 4 + 2x$

True Values



Simple Linear Regression Model

$$y = \beta_0 + \beta_1 x + \varepsilon$$

- y is the dependent variable
- x is the independent variable
- β_0 is the constant or intercept
- β_1 is x's slope or coefficient
- ε is the error term

But how does the machine learn ?

Our Hypothesis given X

$$h_\theta(x) = \theta_0 + \theta_1 x$$

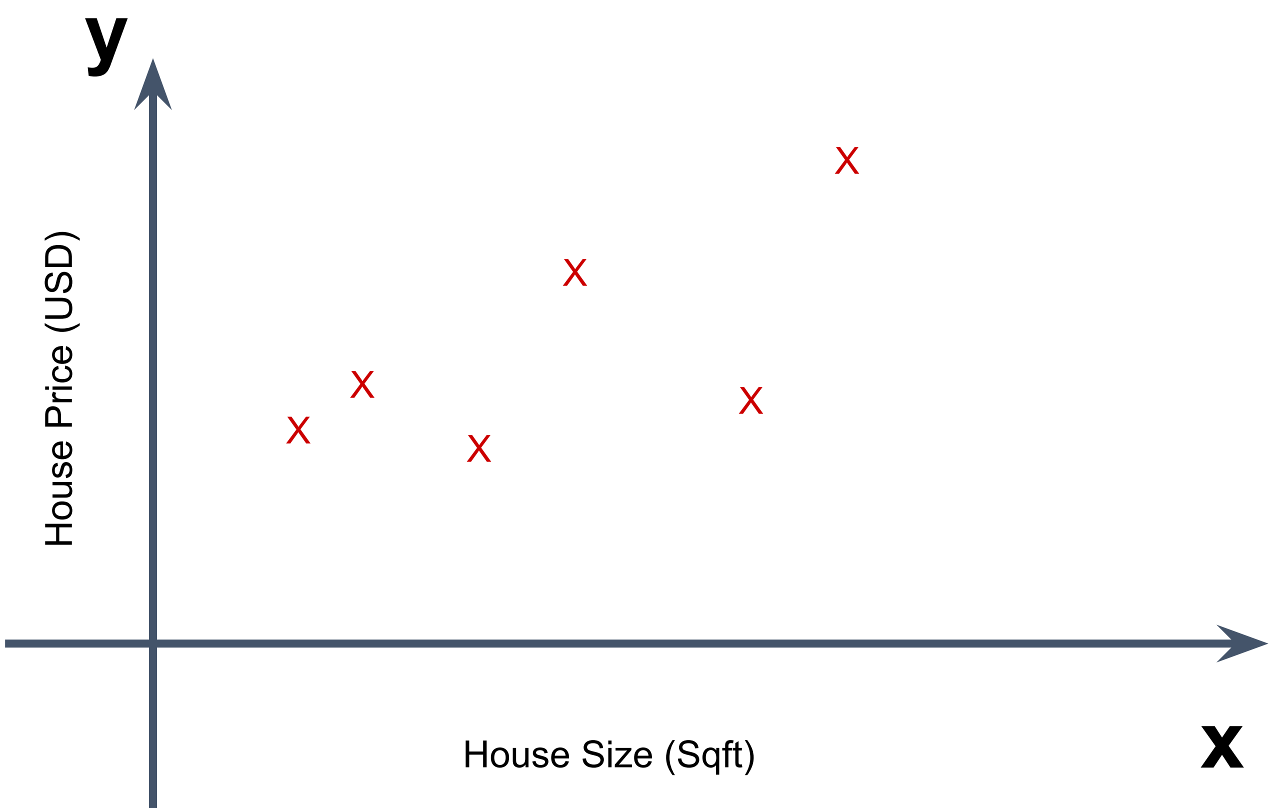
Error Function

error = Our hypothesis - real value

$$error = h_\theta(x) - y$$

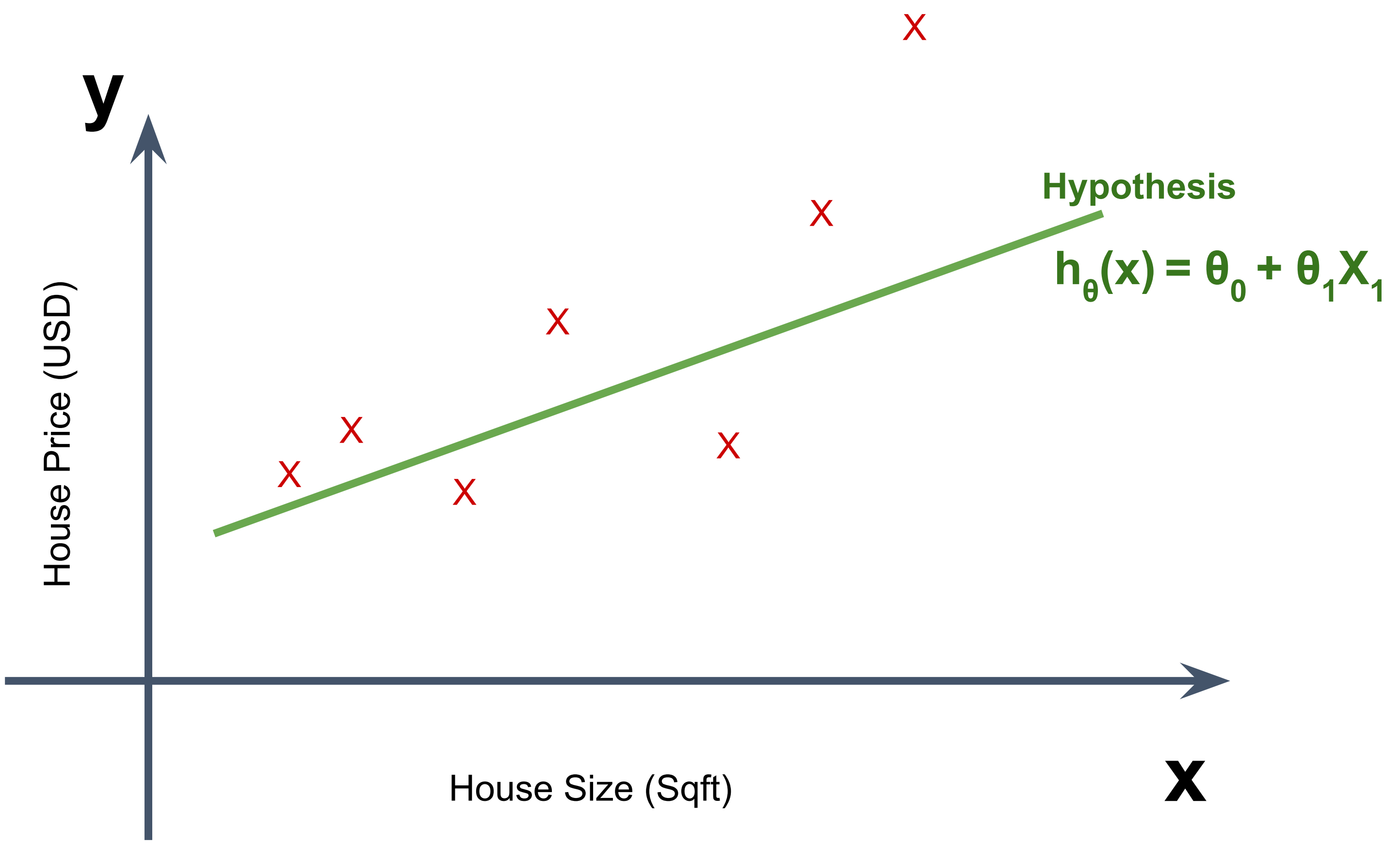
Error Function

$$error = h_\theta(x) - y$$



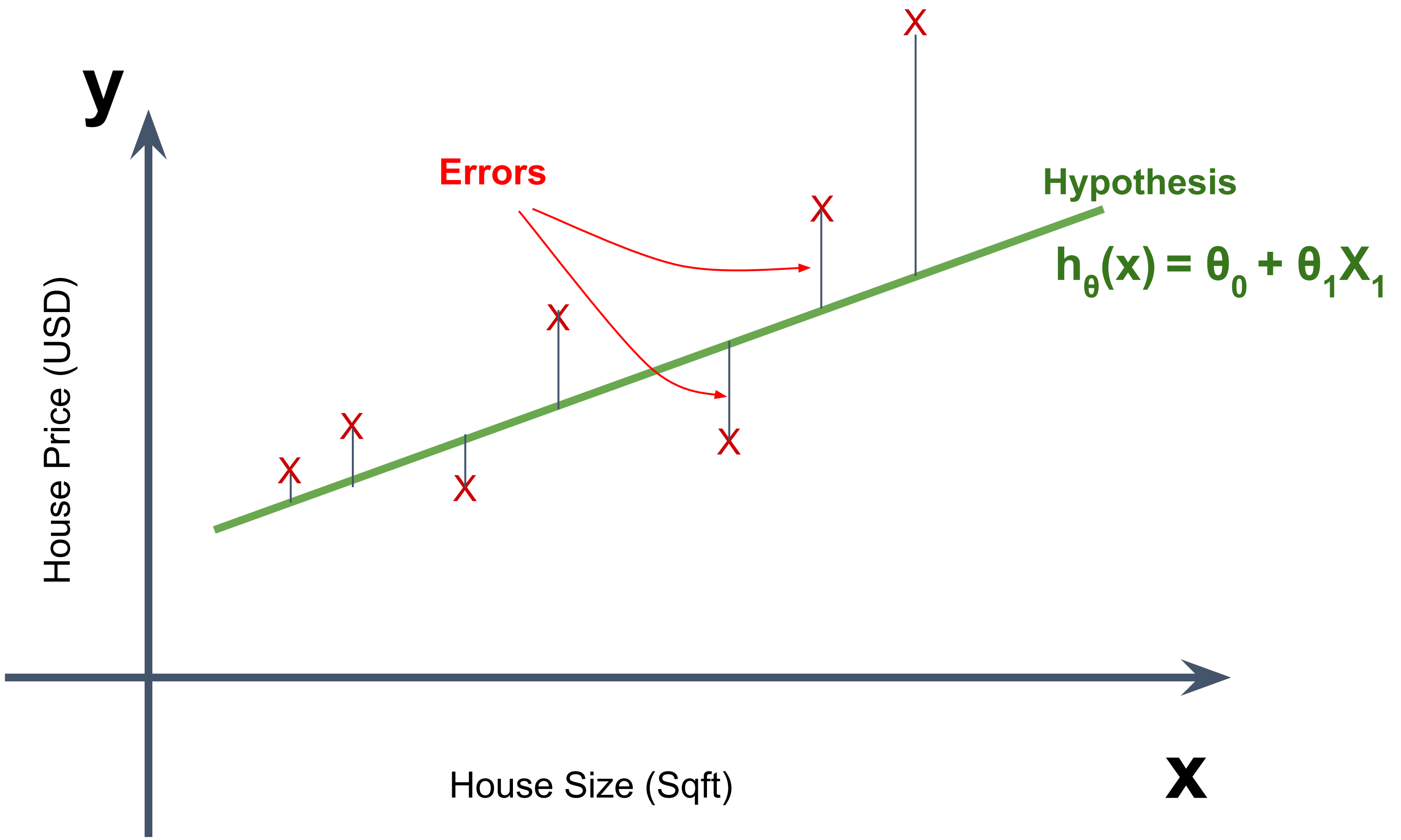
Error Function

$$error = h_\theta(x) - y$$



Error Function

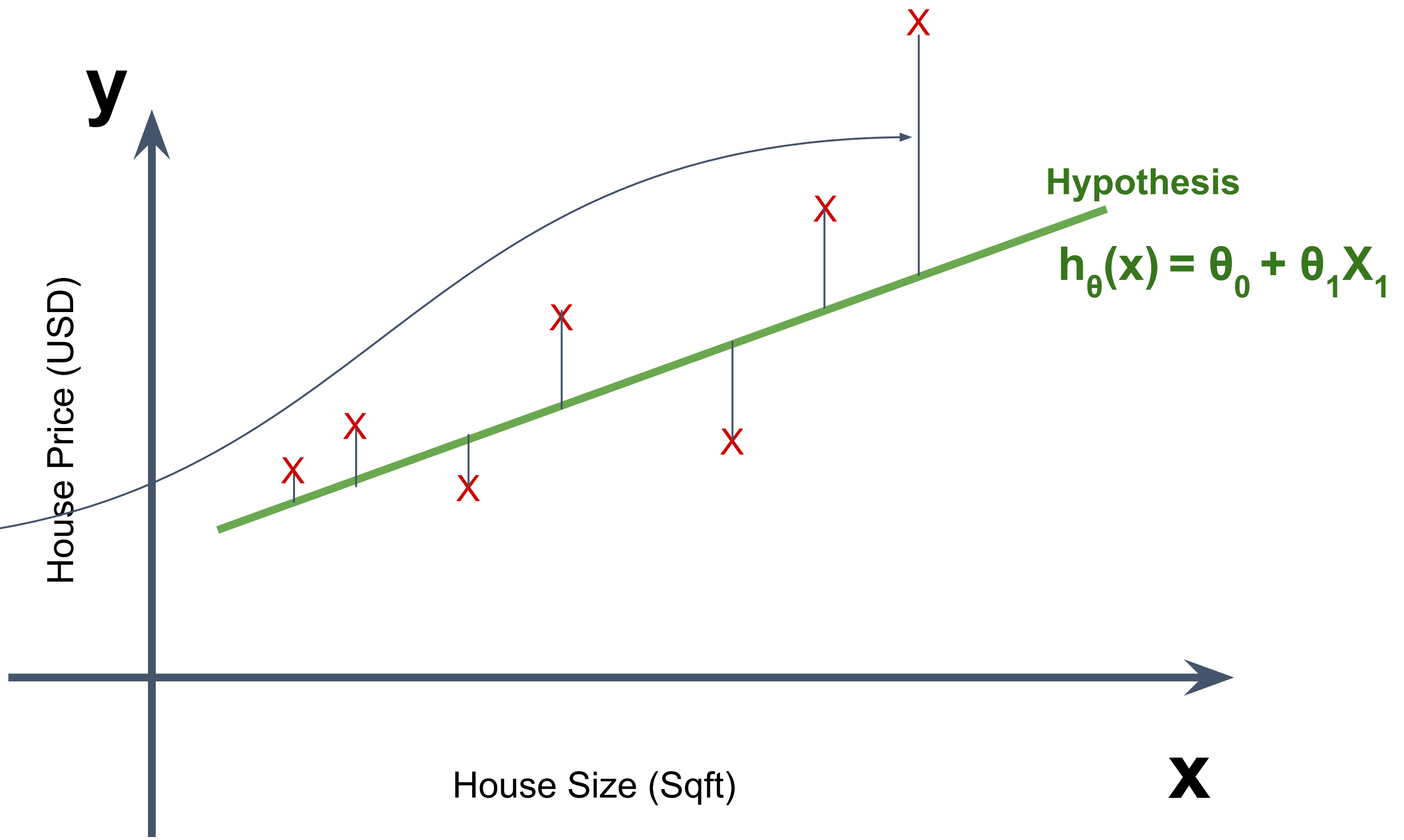
$$error = h_\theta(x) - y$$



Absolute Error

$$\sum |h_\theta(x) - y|$$

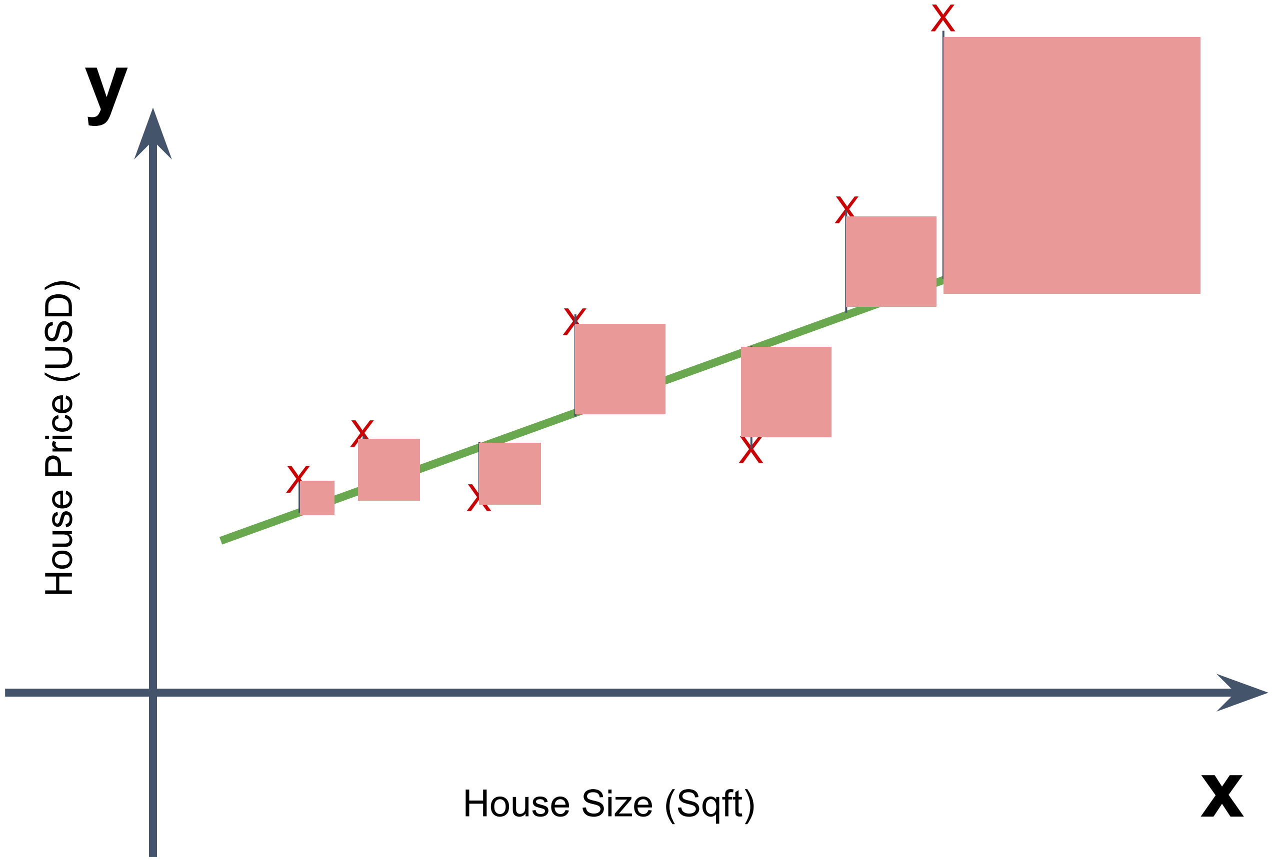
Idea:
Total length of these lines



Squared Error

$$\sum (h_\theta(x) - y)^2$$

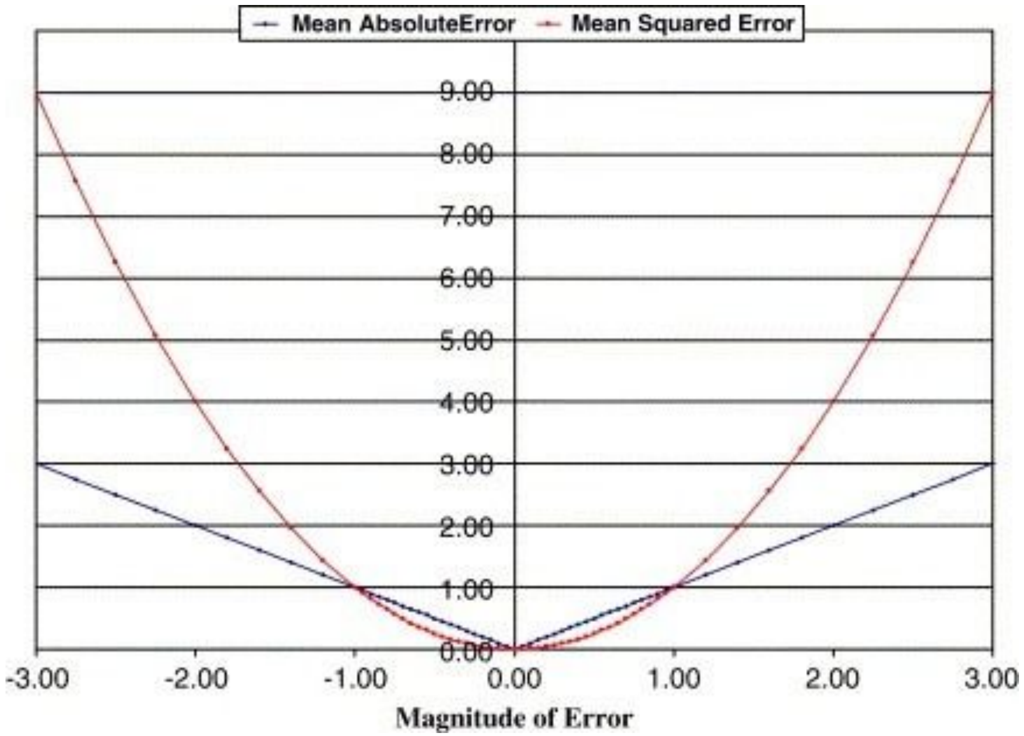
Idea:
Total Area of these squares



Error Function

Why?
Because we have positive and negative errors. So, the **squared loss error** will:

- summing each other instead of negating each other
- But why not absolute error? **Sensitive to large errors**
- Has mathematical advantages such as derivatives



source: https://www.sciencedirect.com/science/article/pii/S0165410104000837?via%3Dihub

Cost Function:
Squared Loss Error

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x) - y)^2$$

Our goal is to minimize this function

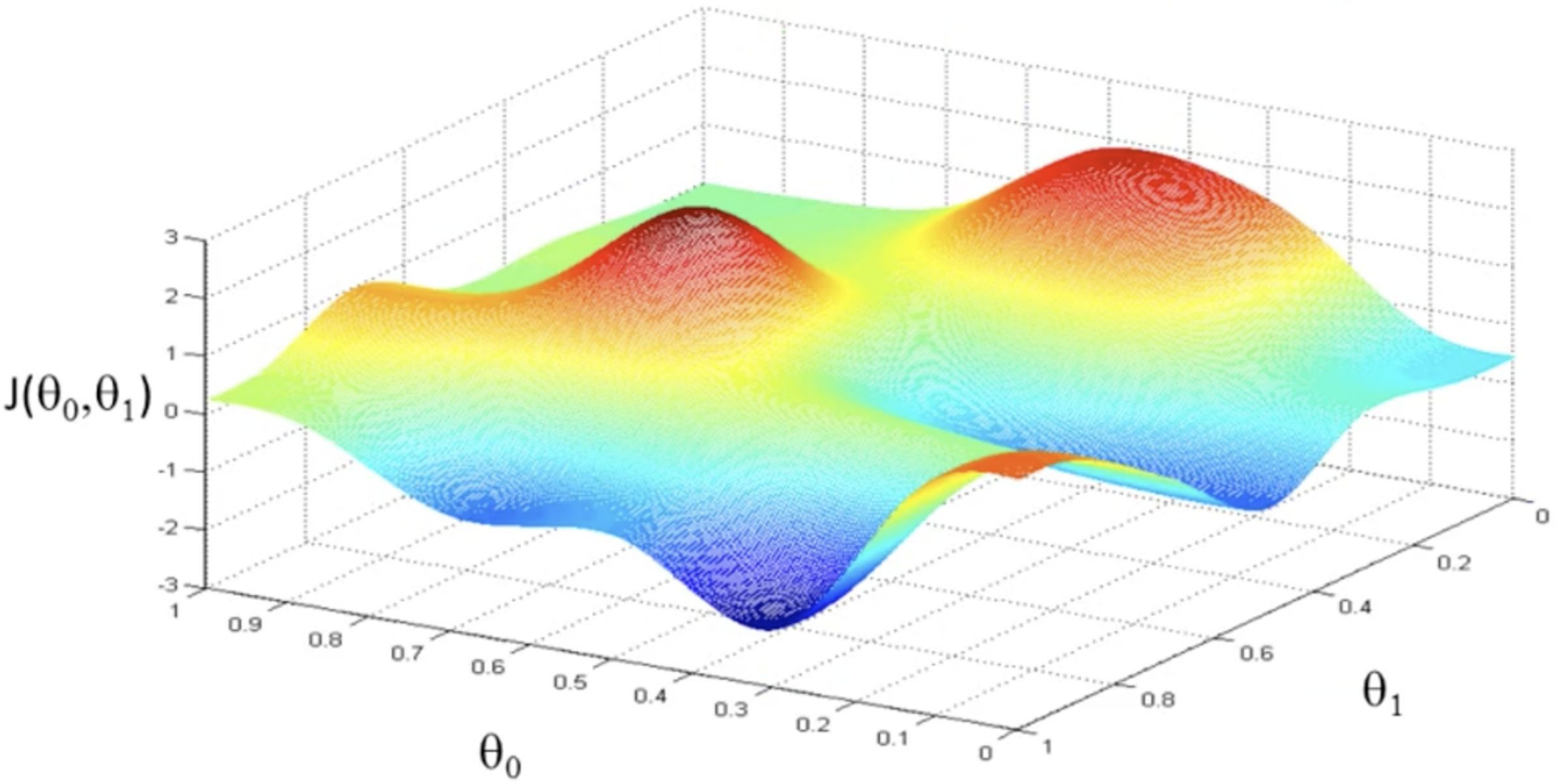
Gradient Descent

Goal:

- Minimize $J(\theta_0, \theta_1)$

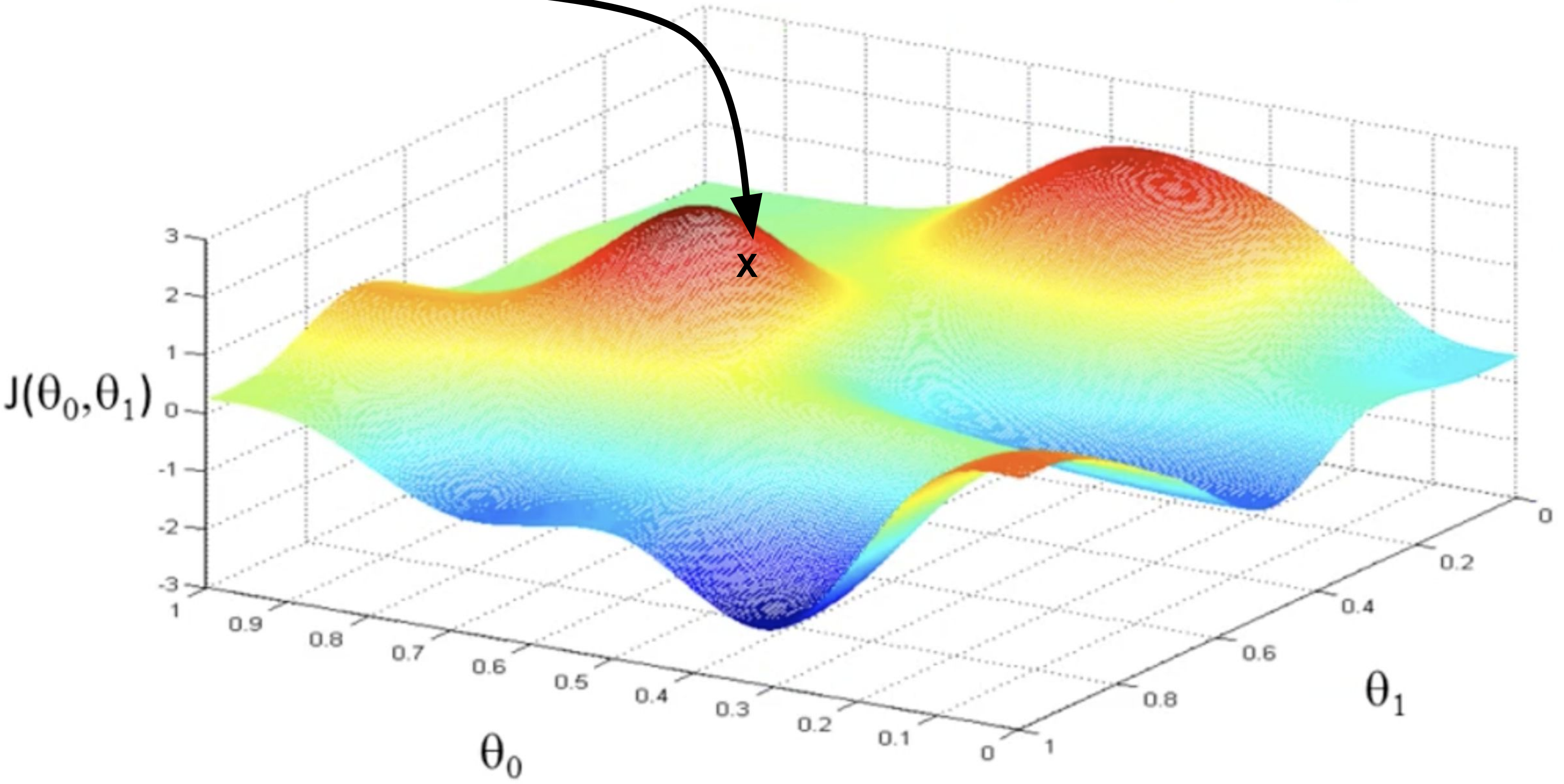
How?

- Start with some (θ_0, θ_1)
- Keep adjusting (θ_0, θ_1) until converge



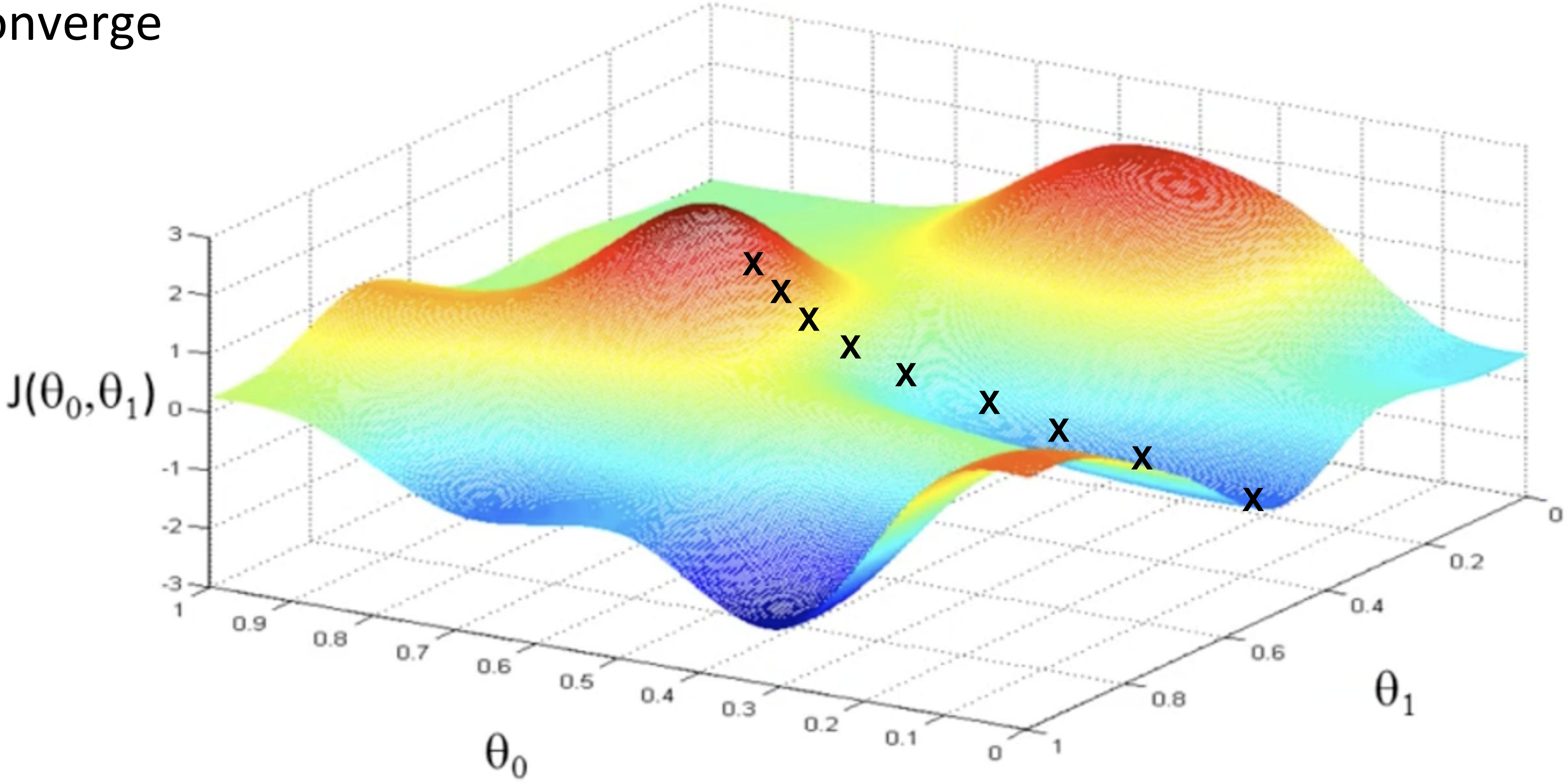
source: Andrew Ng's Machine Learning

Start at some (θ_0, θ_1)



source: Andrew Ng's Machine Learning

Keep adjusting (θ_0, θ_1) until converge



source: Andrew Ng's Machine Learning

Model Evaluation

Mean Absolute Error (MAE) is the mean of the absolute value of the errors:

$$\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Mean Squared Error (MSE) is the mean of the squared errors:

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

Root Mean Squared Error (RMSE) is the square root of the mean of the squared errors:

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Model Evaluation

Mean Absolute Error (MAE) is the mean of the absolute value of the errors:

$$\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Mean Squared Error (MSE) is the mean of the squared errors:

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

punishes larger errors

Root Mean Squared Error (RMSE) is the square root of the mean of the squared errors:

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Model Evaluation

Mean Absolute Error (MAE) is the mean of the absolute value of the errors:

$$\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Mean Squared Error (MSE) is the mean of the squared errors:

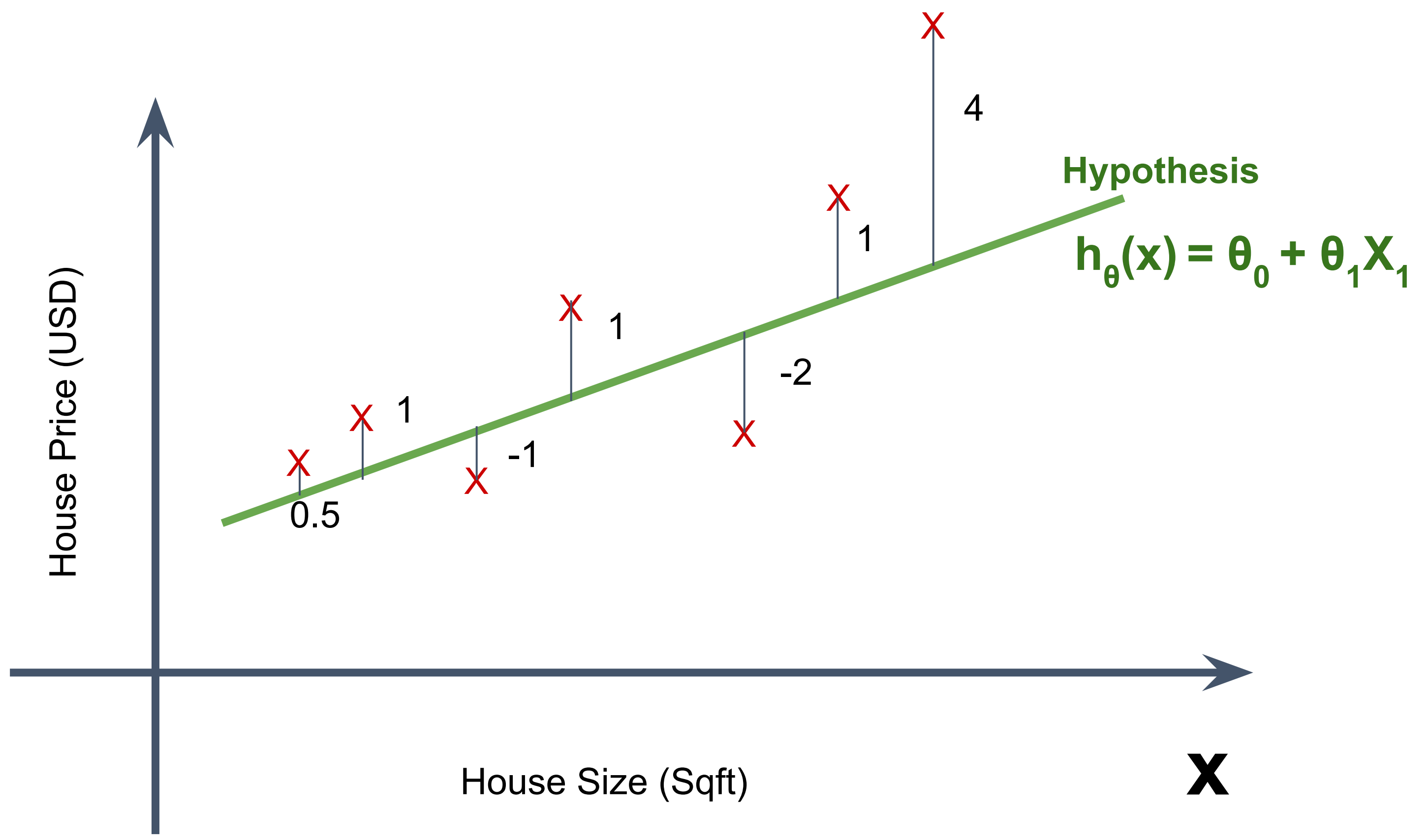
$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Root Mean Squared Error (RMSE) is the square root of the mean of the squared errors:

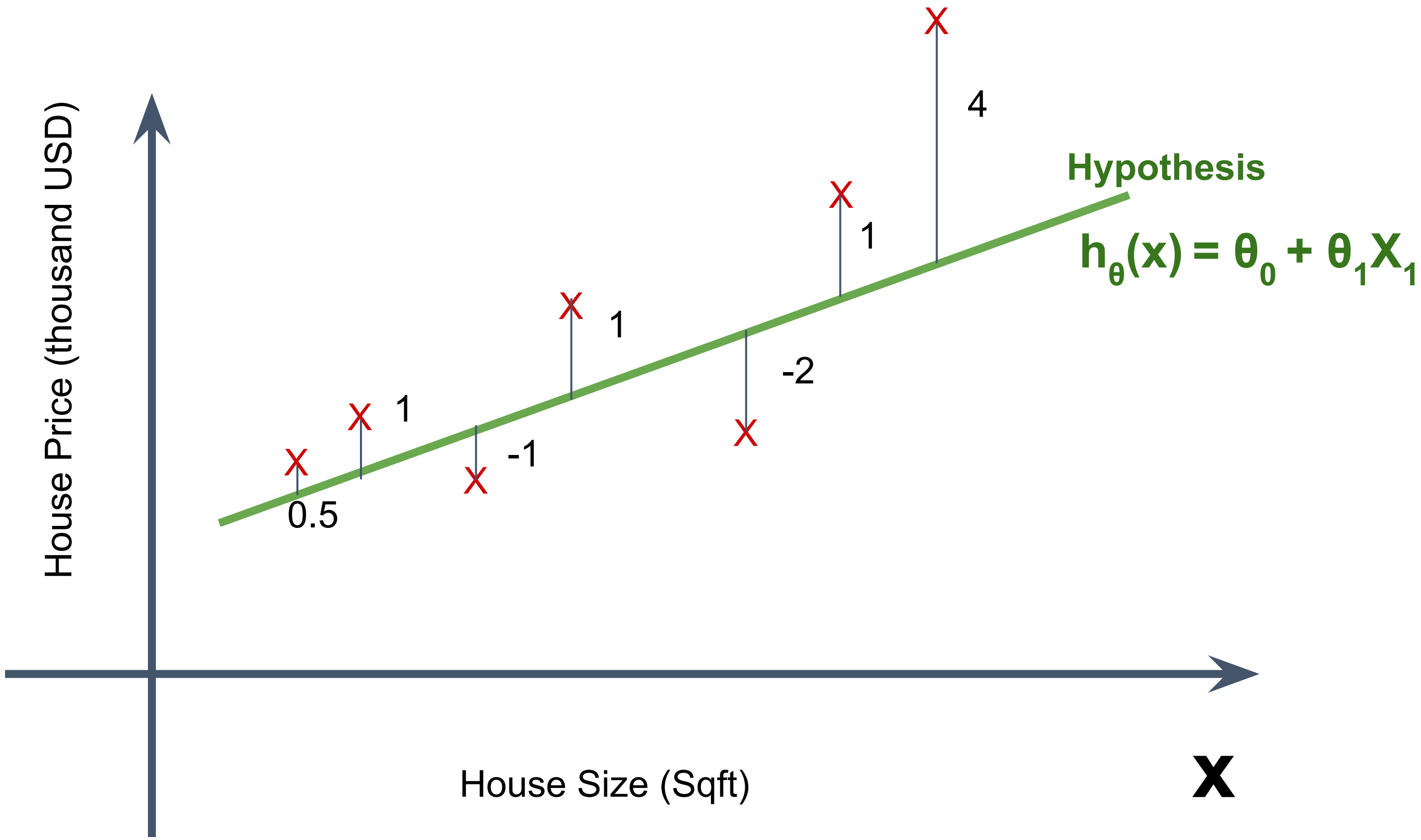
$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

More interpretable in y units

Model Evaluation



Model Evaluation



Mean Absolute Error:
1/7 * (0.5 + 1 + 1 + 1 + 2 + 1 + 4)
= 10.5 /7
= 1.5 (thousand USD)

MSE
1/7 * 0.25 + 1 + 1 + 1 + 4 + 1 + 16
= 24.25 / 7
= 3.46 (million USD^2)

RMSE
(1/7 * 0.25 + 1 + 1 + 1 + 4 + 1 + 16)^(0.5)
= 1.86 (thousand USD)

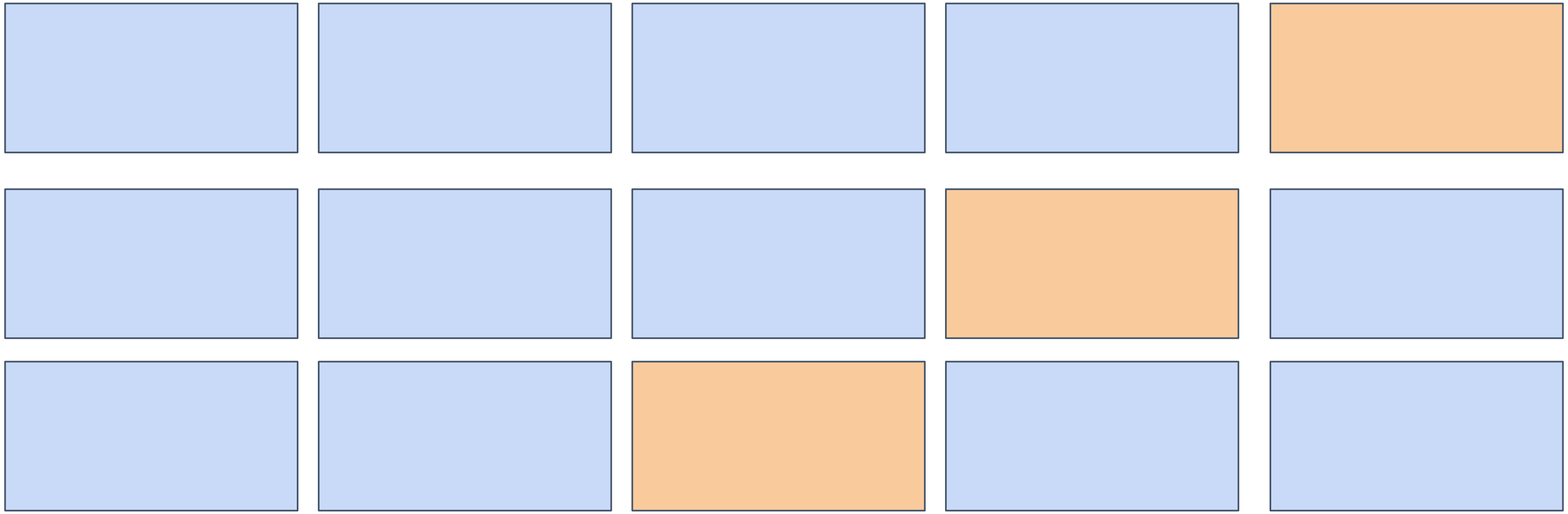
Model Validation

Using Hold - Out test data

using 80% data as training set, 20% as testing
But How do we know the testing set is not randomly the easy part to predict?

Model Validation

Cross Validation (with 5 fold)



Each partition takes turn to be the training & testing data set.
Important to know the real performance of the model

Questions ?

dipta@go-jek.com