Discussion print(“C”)

Today we will learn:

Today we will learn:

Lina

Regression

But what is Linear Re*GURA*ssion?

In simplified words:

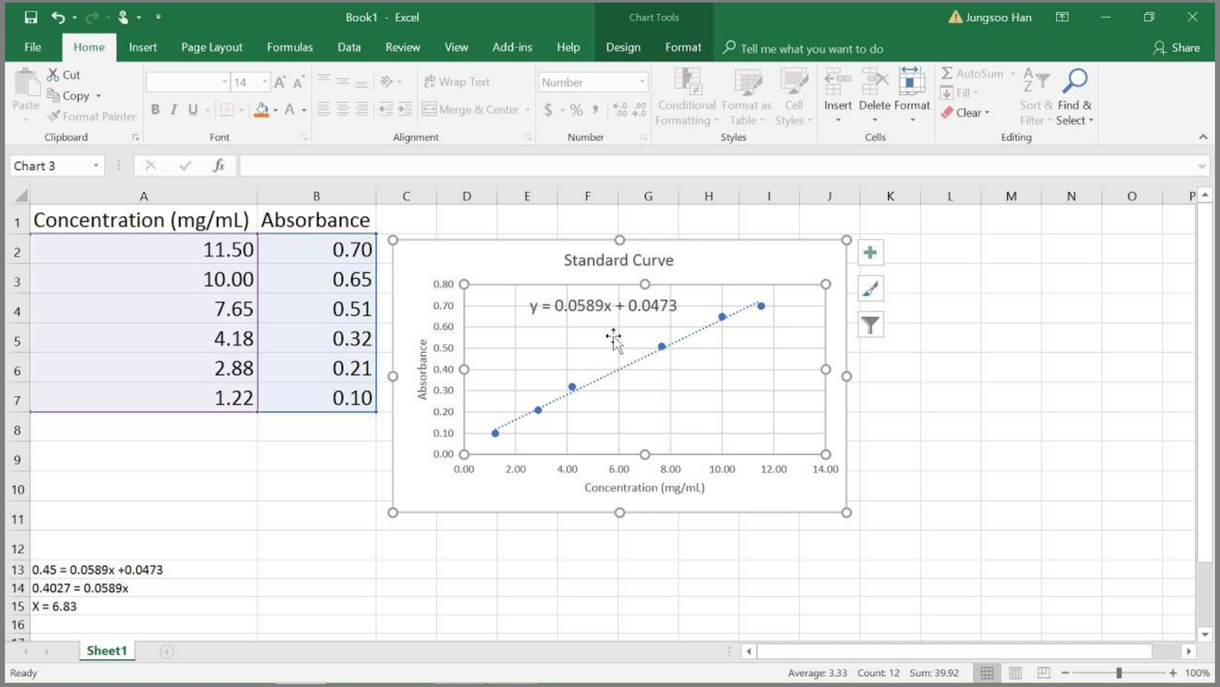
It is a way to capture the trend line of dataset

In longer explanation:

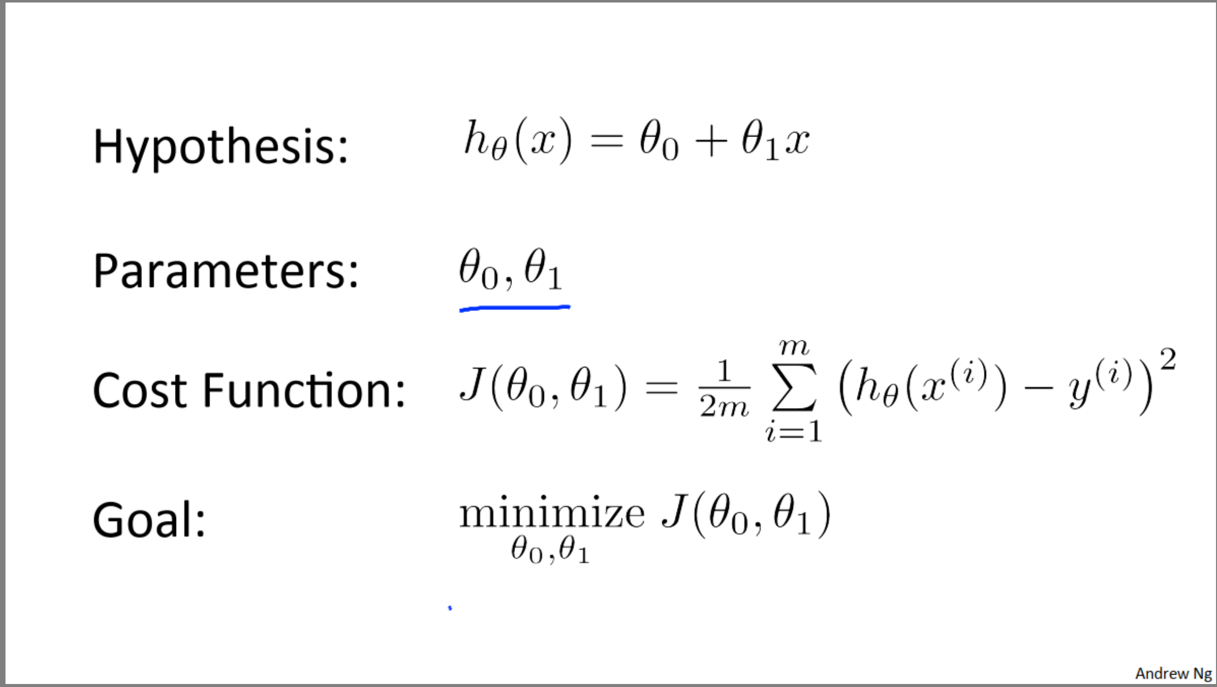
In statistics, linear regression is a linear approach to modelling the relationship between a scalar response and one or more explanatory variables (also known as dependent and independent variables) [wiki]

Let’s look inside the math behind
linear rEVANGELION

Remember this?



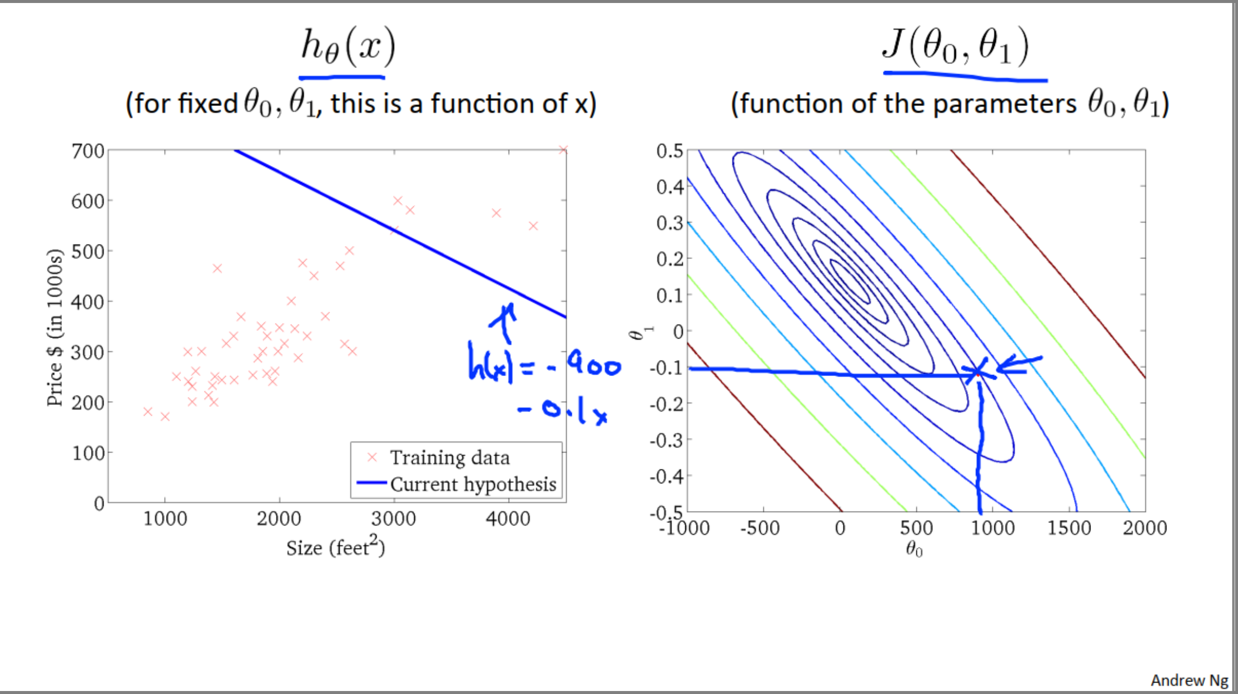
This is him now



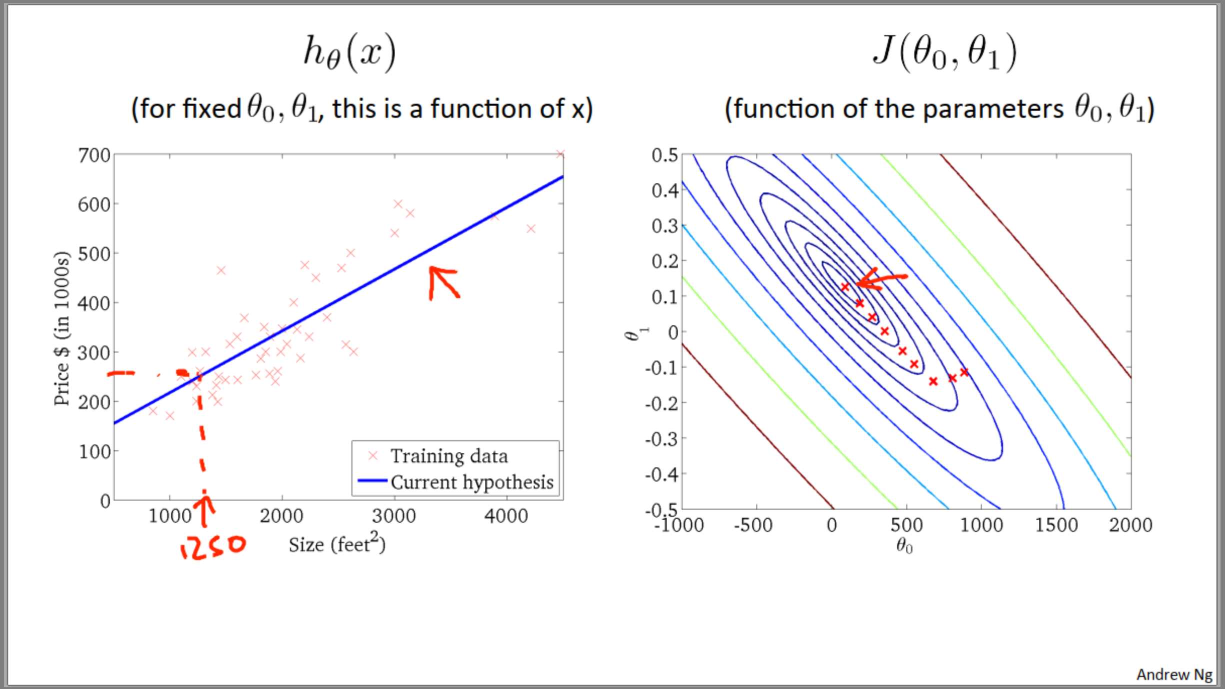
Credit: Andrew Ng

Feels old yet?

But how do you get from this

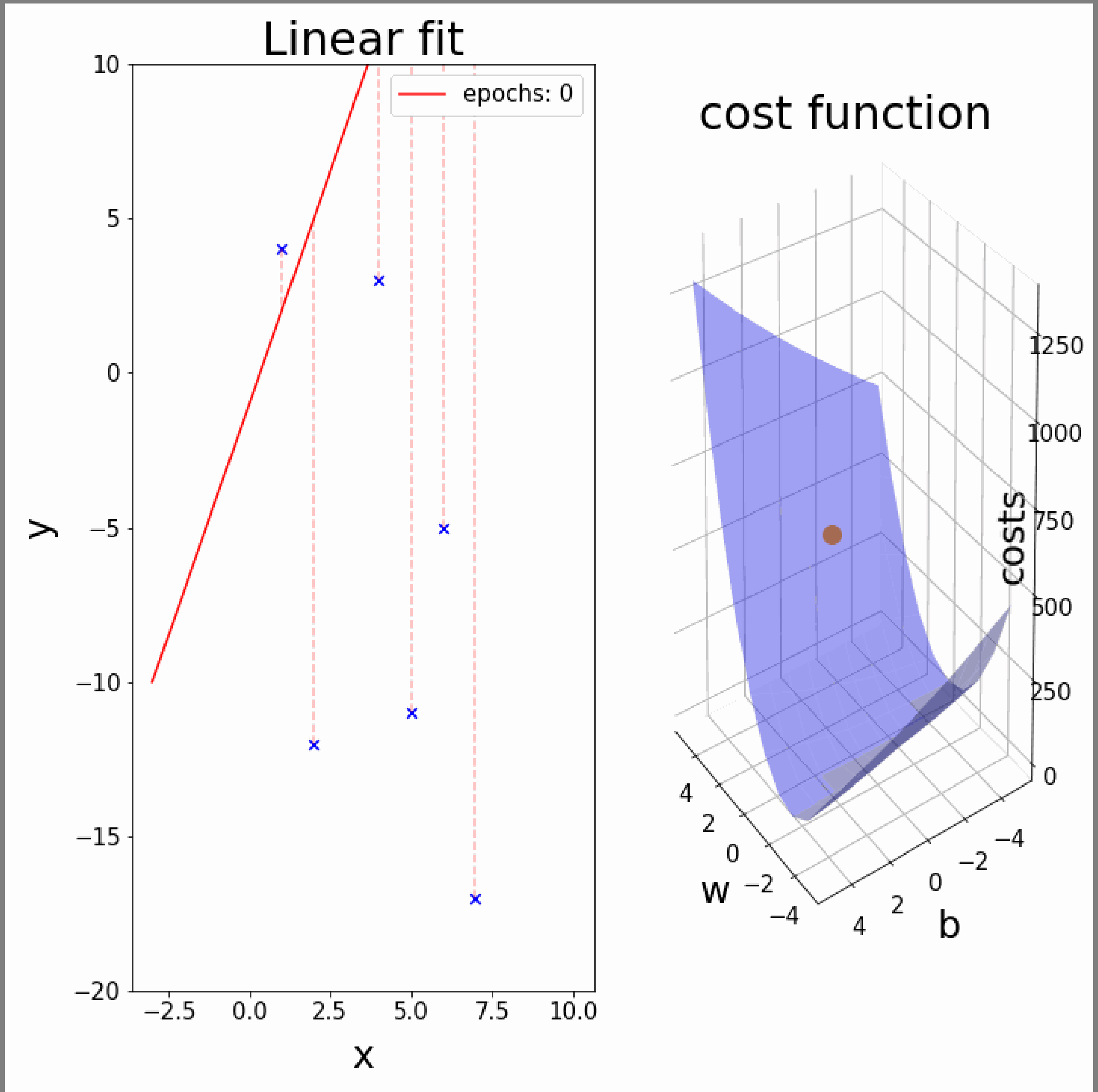


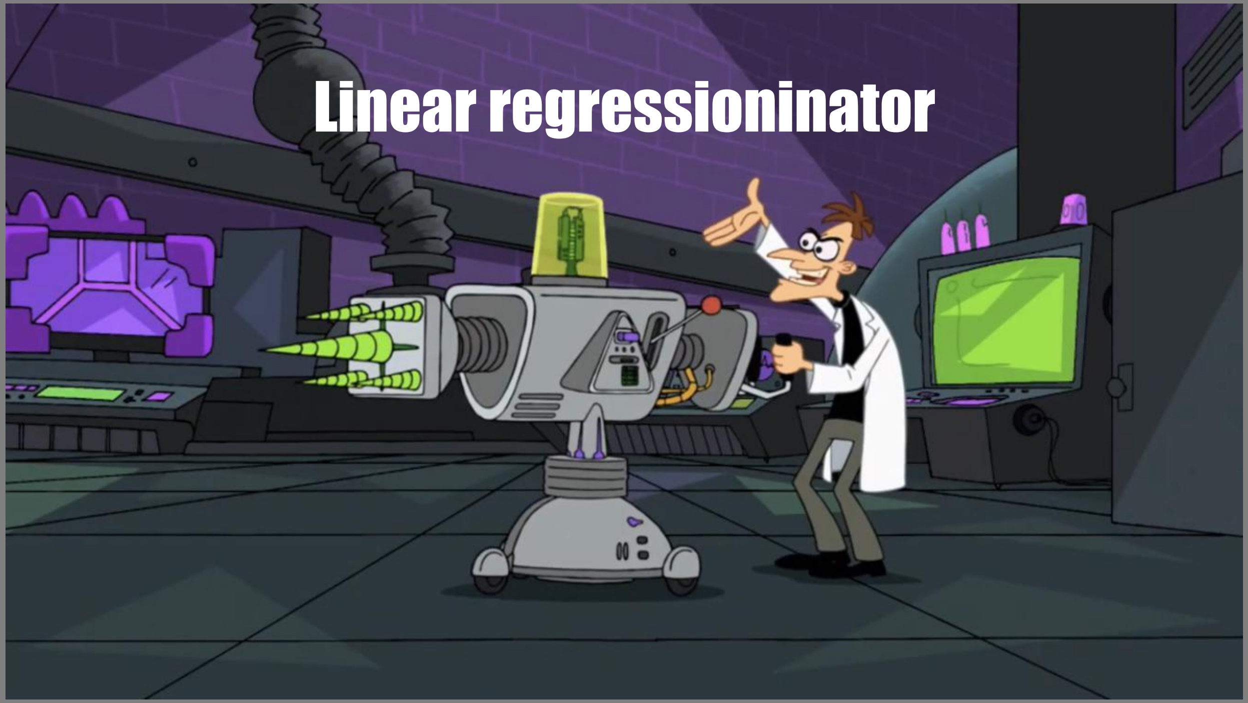
To this



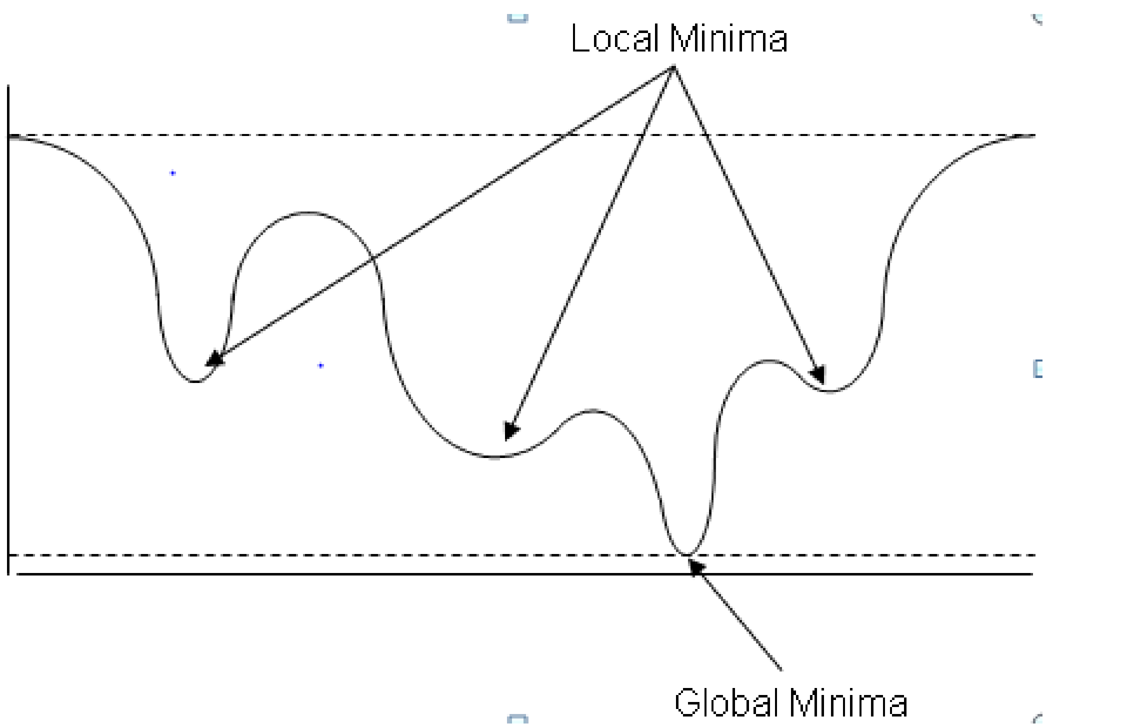
Credit: Andrew Ng

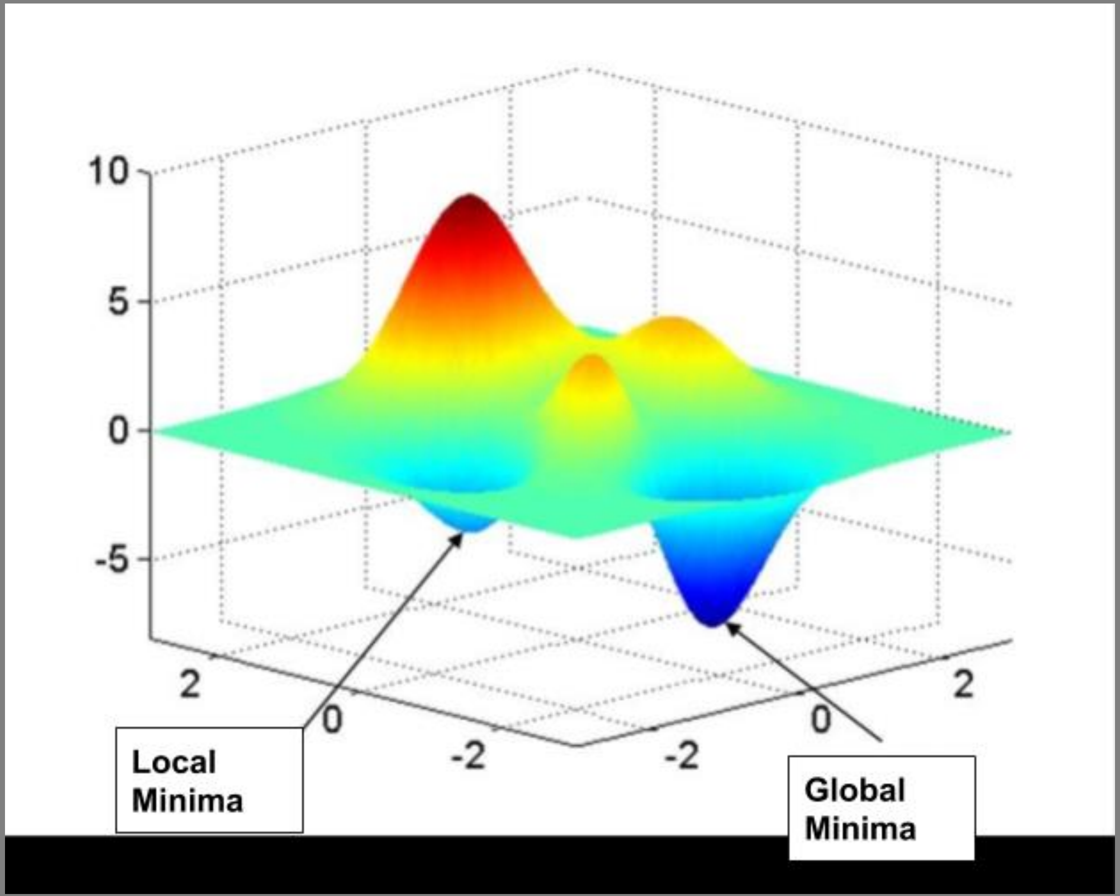
Introducing: Gradient Descent

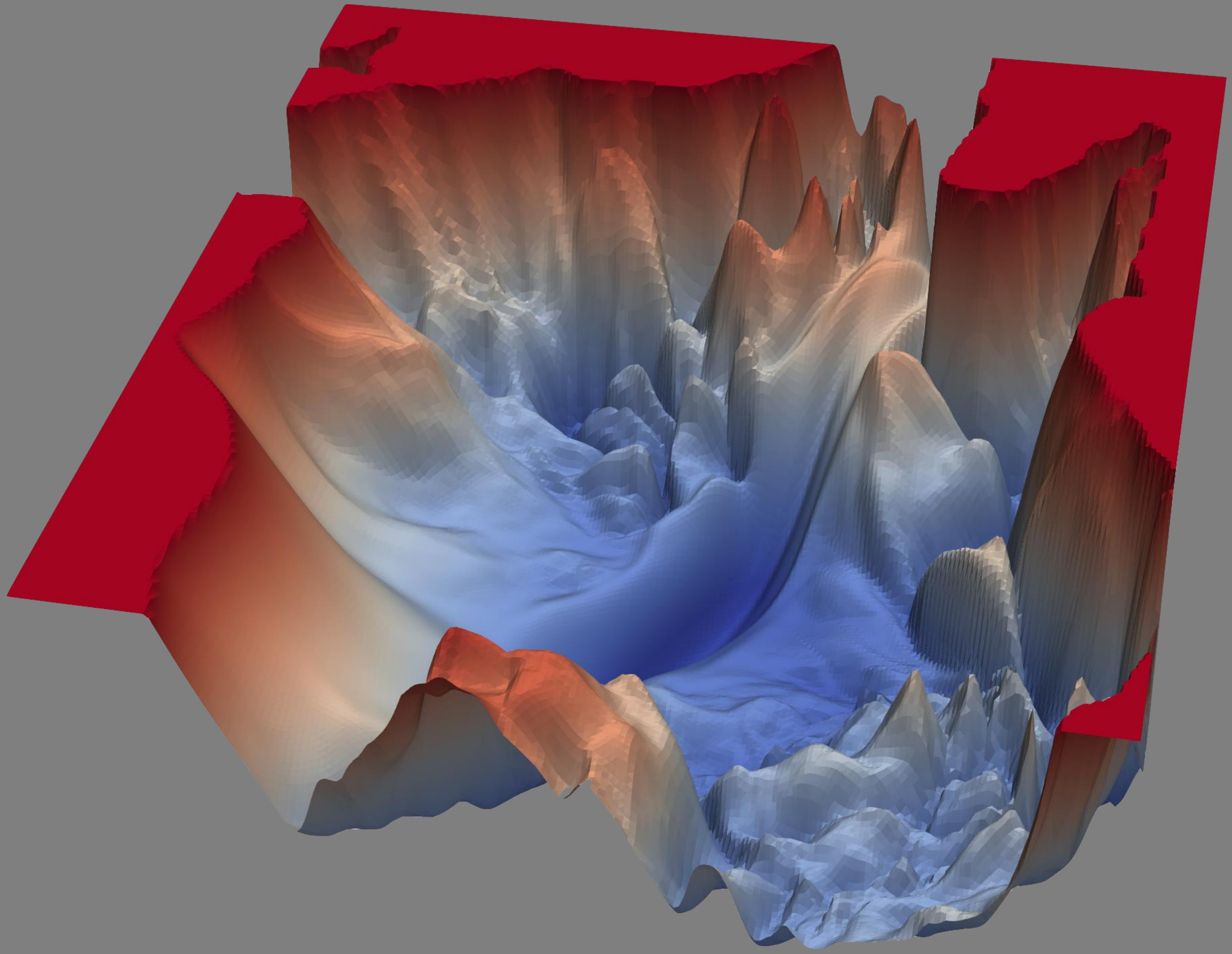

Linear fit
epochs: 0
y
x
10
5
0
−5
−10
−15
−20
−2.5
0.0
2.5
5.0
7.5
10.0
cost function
costs
1250
1000
750
500
250
0
w
b
4
2
0
−2
−4


Linear regressioninator

Issue: what if we cannot get the lowest cost function



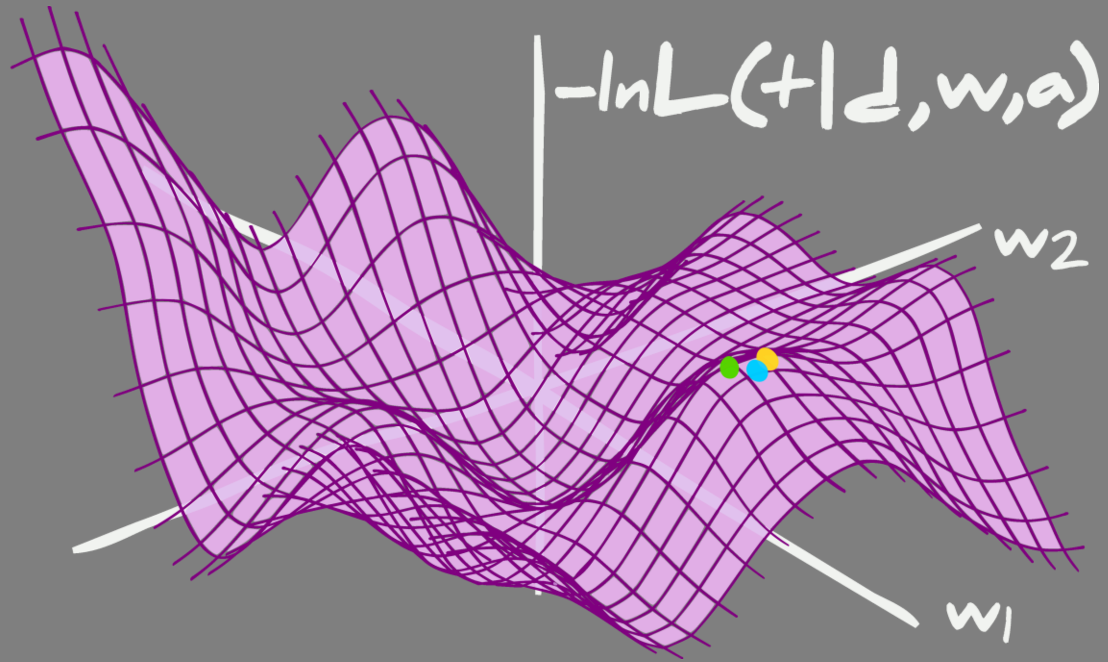




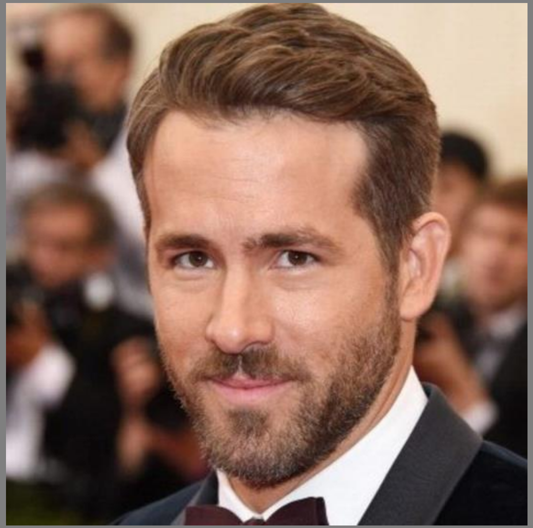
A Complicated Loss Landscape *Image Credits: https://www.cs.umd.edu/~tomg/projects/landscapes/*

- Adjust learning rate (difficult to find the right learning rate)

- Repeat the trial while changing starting point every time (Stochastic Gradient Descent)



Linear REYgression with multiple features



Simple Linear Regression

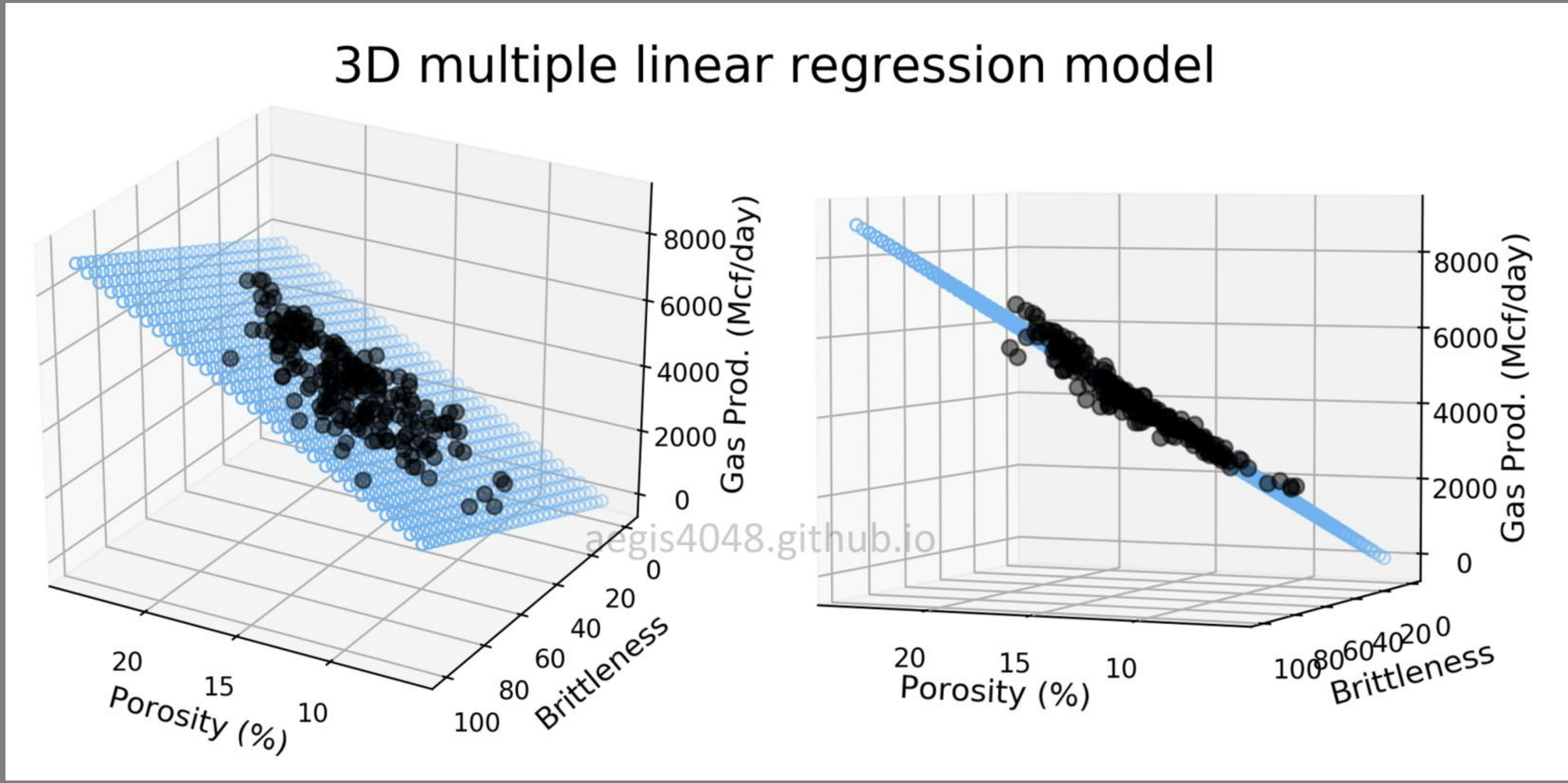
$$y = b_0 + b_1{}^*x_1$$

Multiple Linear Regression

Dependent variable (DV) Independent variables (IVs)

$$y = b_0 + b_1{}^*x_1 + b_2{}^*x_2 + \ldots + b_n{}^*x_n$$

Constant Coefficients



If we have this….

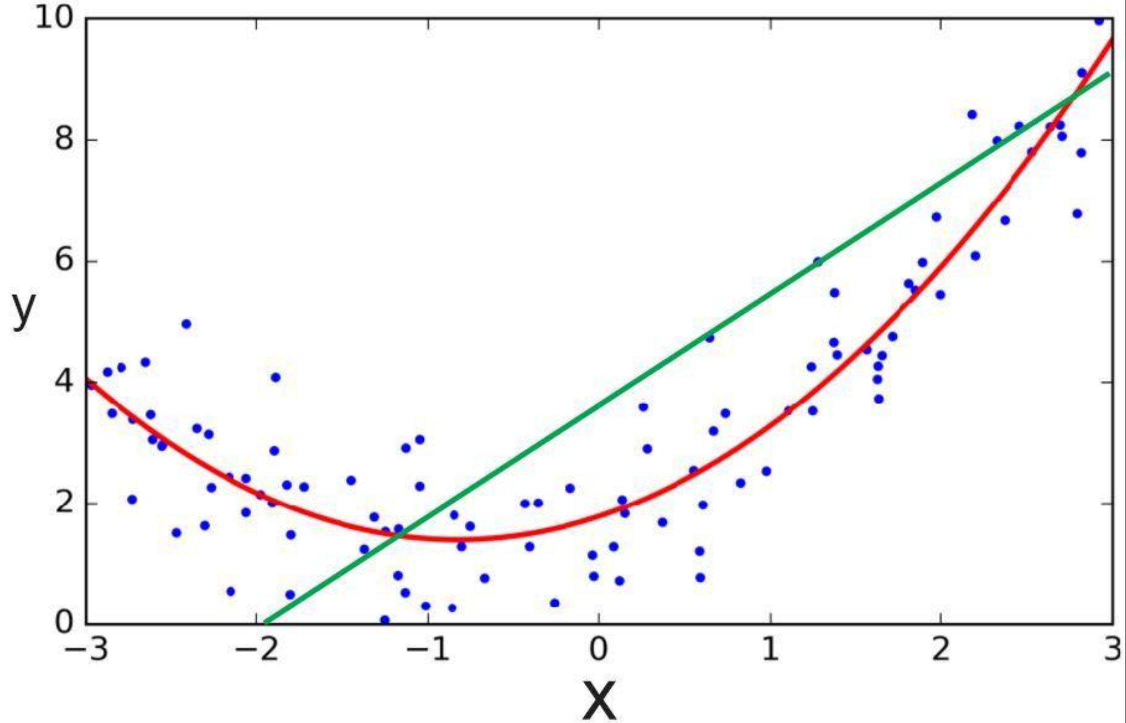
Multiple Linear Regression

$$y = b_0 + b_1x_1 + b_2x_2 + \ldots + b_nx_n$$

We can also have this!

Polynomial Linear Regression

$$y = b_0 + b_1x_1 + b_2x_1^2 + \ldots + b_nx_1^n$$



Same cost(loss) function and gradient descent

repeat until convergence: {

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

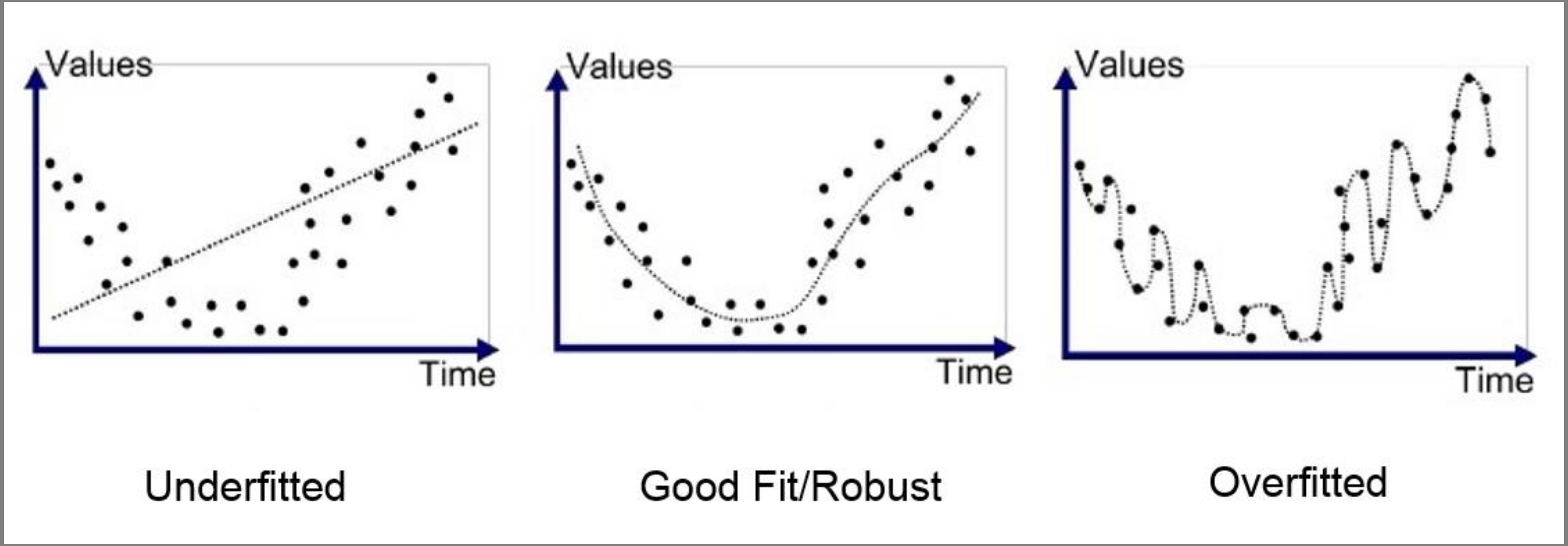
$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_1^{(i)}$$

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

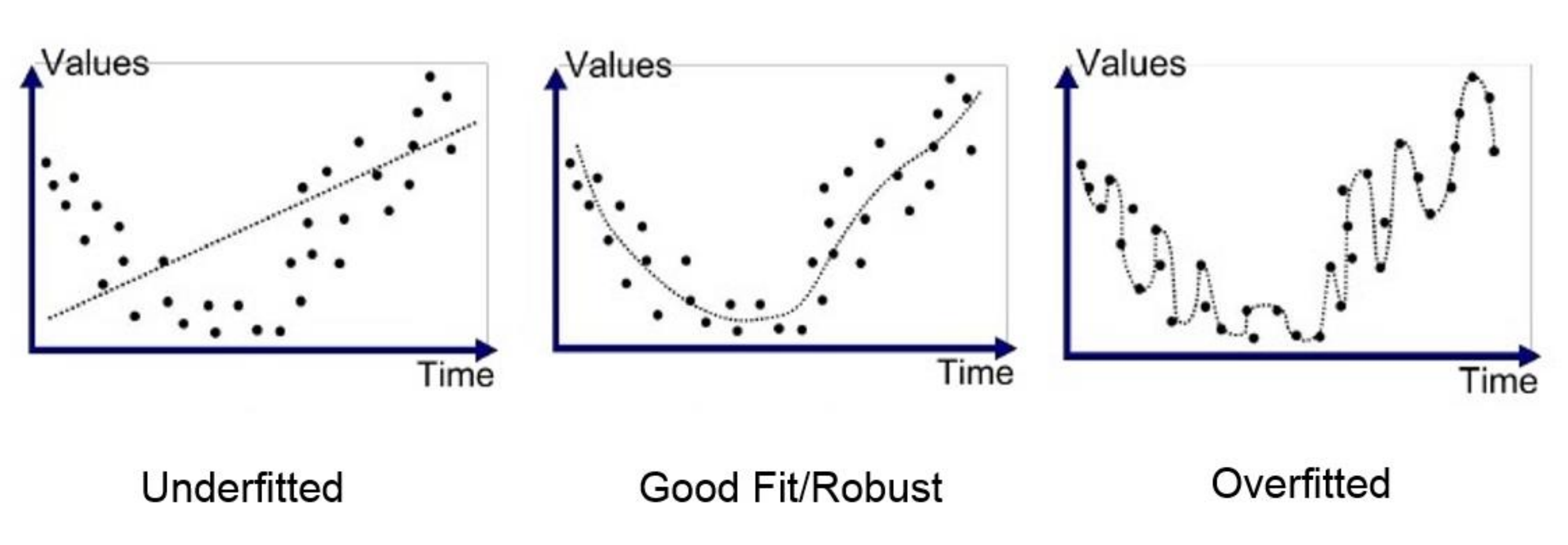
$\cdots$

}

The good, the bad, and the ugly



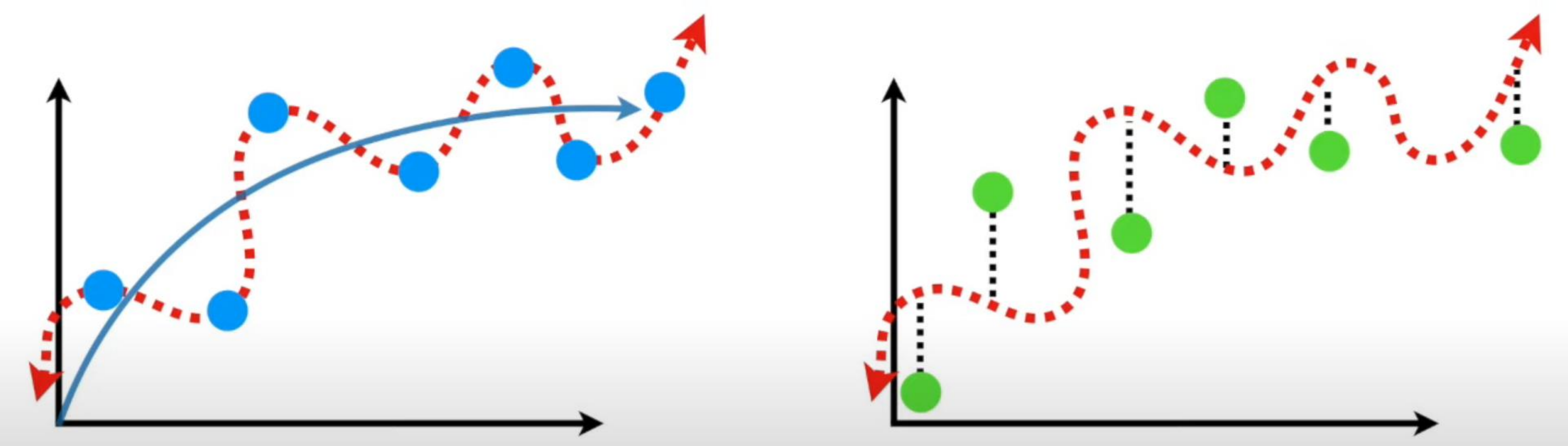
Terminology and Tips



We use the term “bias” and “variance” as another way to explain how well the trend line(or plane if 3D) captures data

Bias: Inability to capture the true relationship
Variance: The difference in cost function between train dataset and test dataset

Bias: Inability to capture the true relationship
Variance: The difference in cost function between train dataset and test dataset



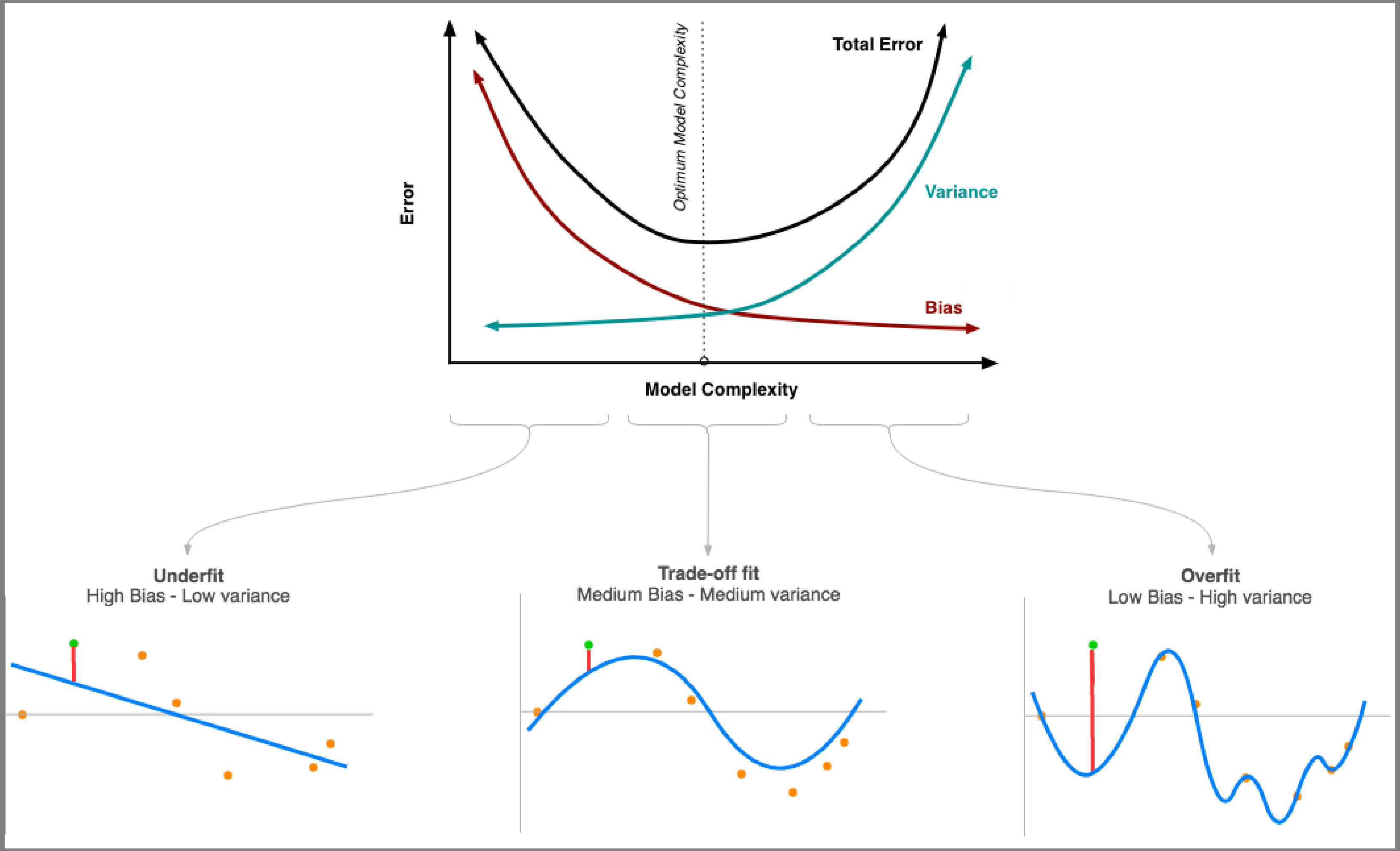
Train dataset

Test dataset

Ideally: low bias and low variance

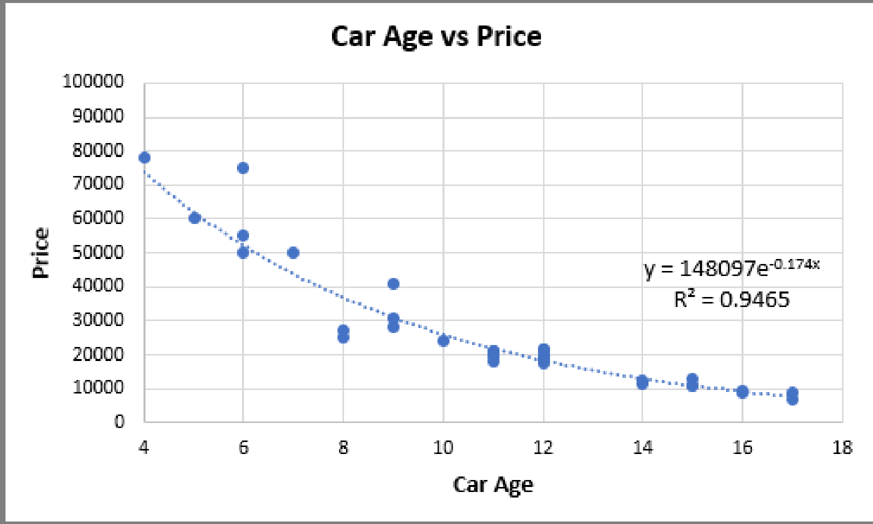
Bias: Inability to capture the true relationship
Variance: The difference in cost function between train dataset and test dataset

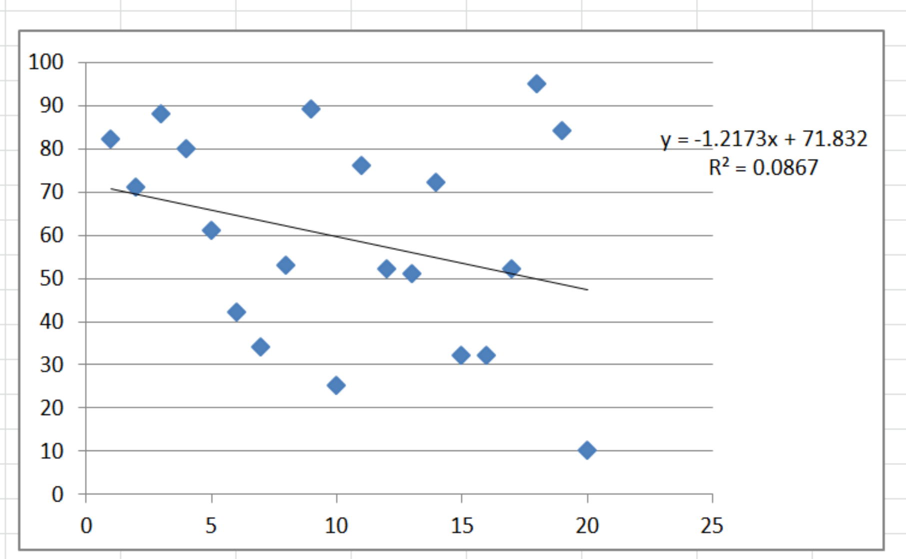
Ideally: low bias and low variance



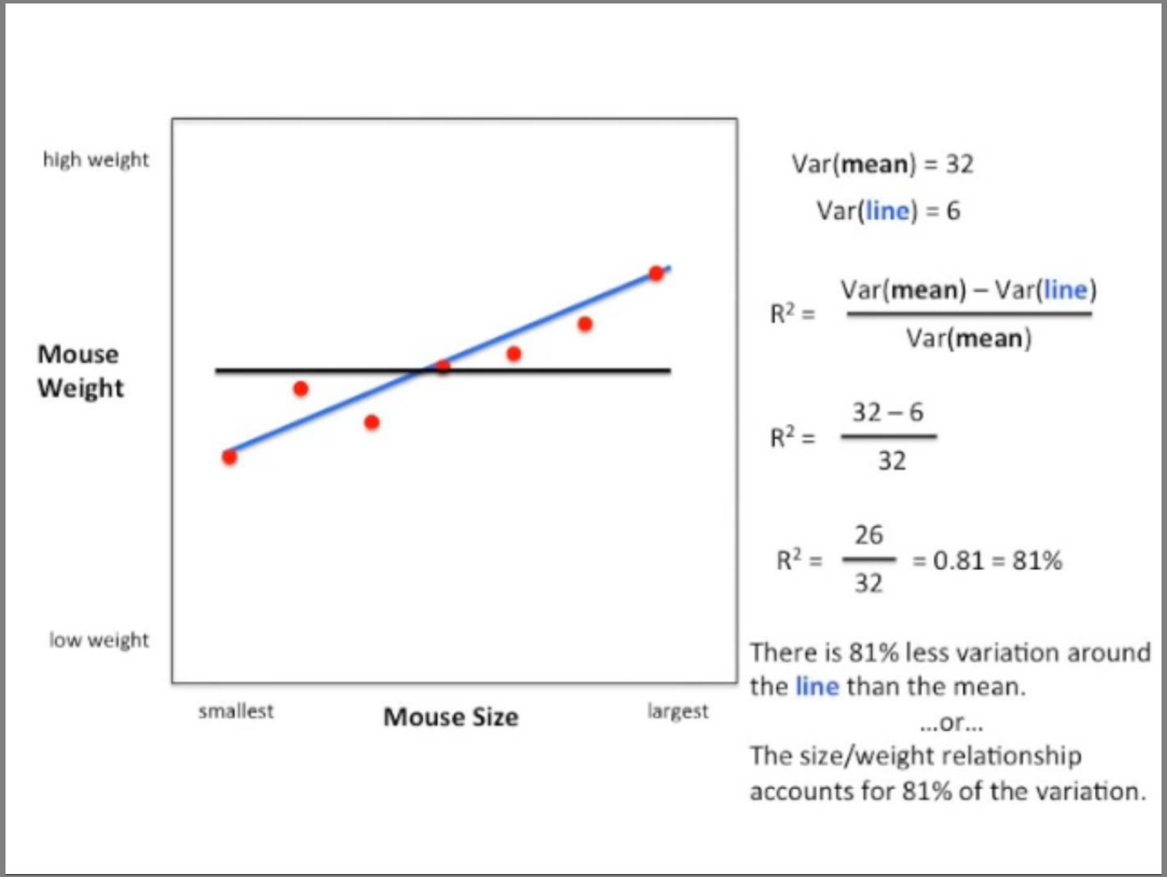
R-squared

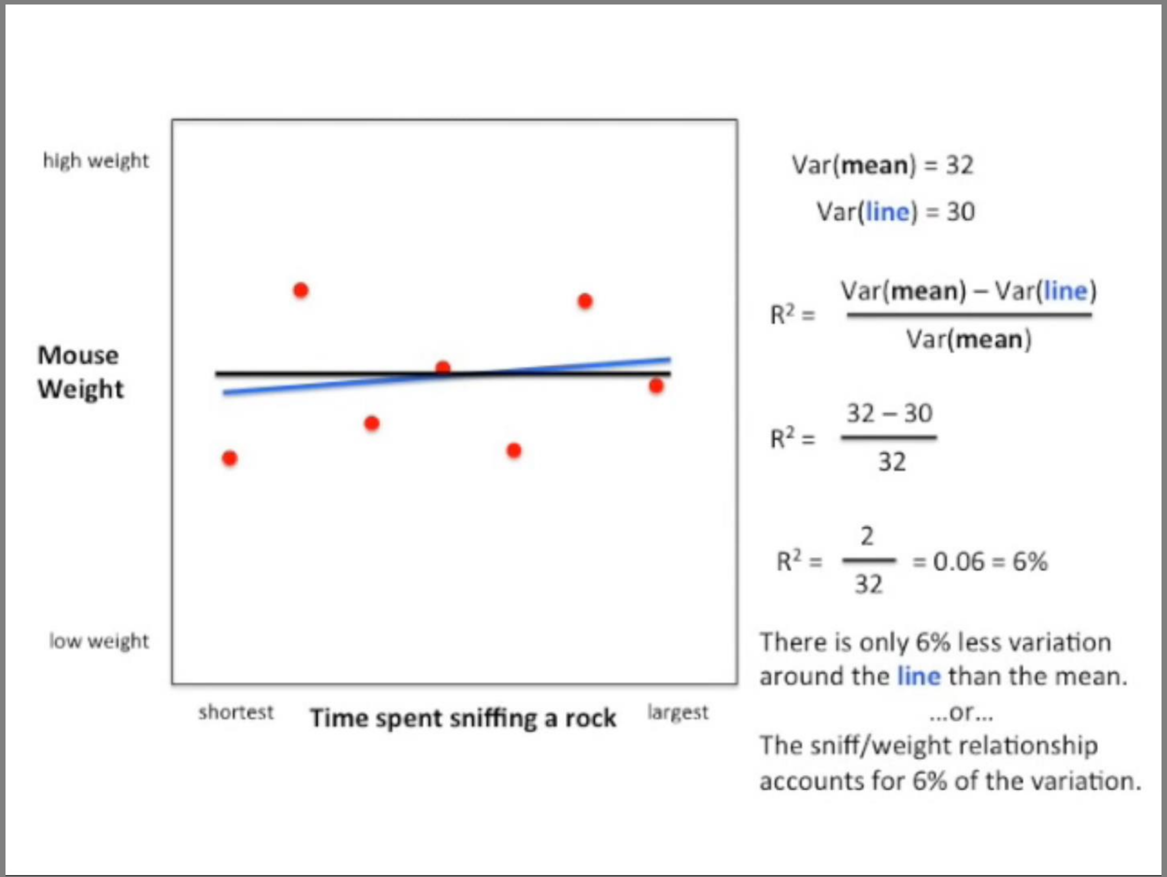
- R-squared is a goodness-of-fit measure for linear regression models
- It tells how much two variables are correlated. $R^2 = 1$ means two variables are perfectly correlated. $R^2 = 0$ means that two variables are not correlated





R-squared





BUT! Be careful! High R-squared can also mean the model overfits

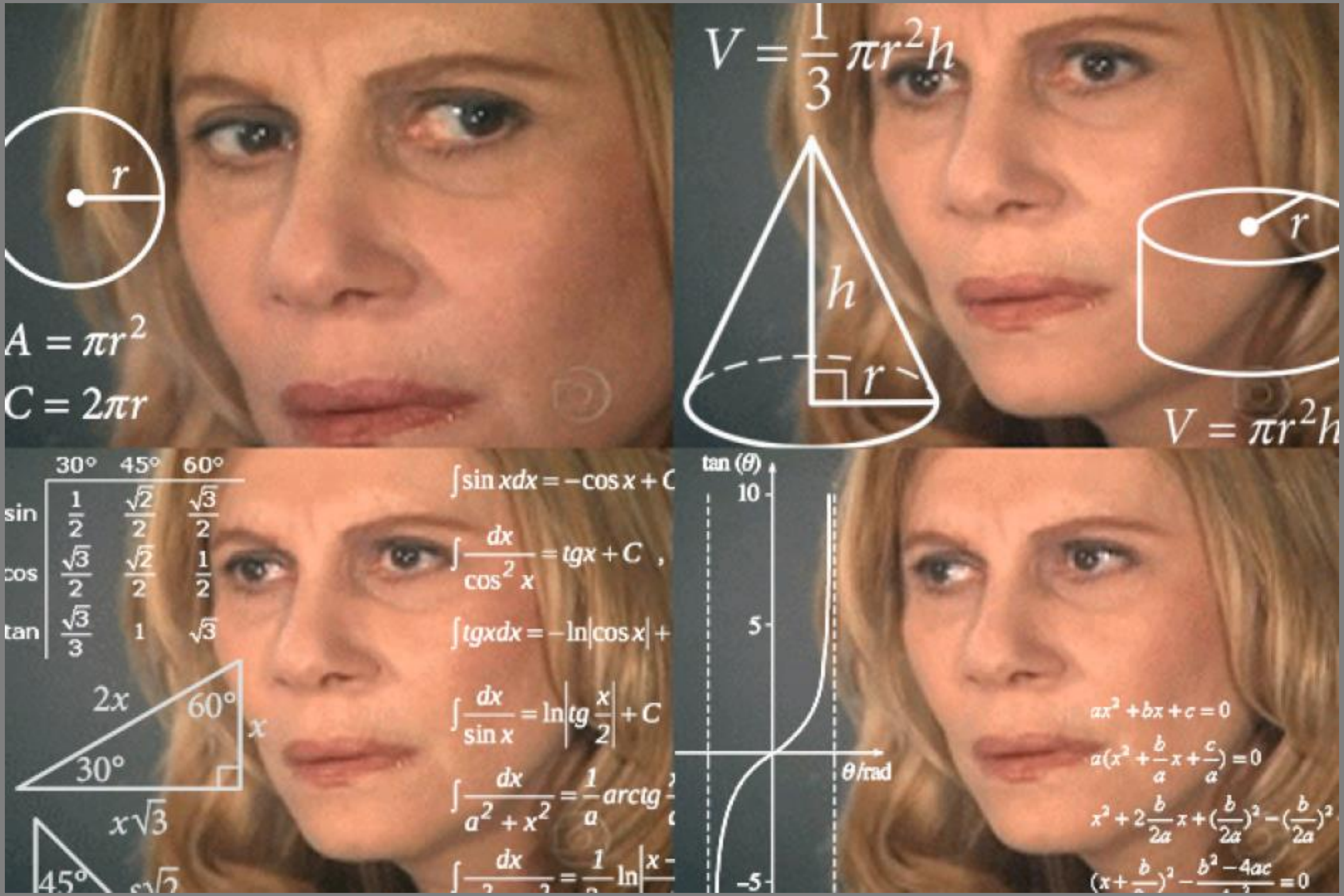
```
[16]: correlation = airbnb_housing.corr()
      correlation["price"].sort_values(ascending=False)

[16]: price                             1.000000
      availability_365                  0.081829
      calculated_host_listings_count    0.057472
      minimum_nights                    0.042799
      latitude                          0.033939
      host_id                           0.015309
      id                                0.010619
      reviews_per_month                -0.030608
      number_of_reviews                -0.047954
      longitude                        -0.150019
      Name: price, dtype: float64
```

Correlation between housing price and other features. Note: this is correlation (R), not R^2.

Basically: when working with dataset, consider features correlation. It's up to you to drop a certain feature if you believe it does not contribute to the prediction model

// R-squared is for the predicting line correlation to all feature, while R is for correlation between 2 features


r
$A = \pi r^2$
$C = 2\pi r$
$V = \frac{1}{3}\pi r^2 h$
h
r
r
$V = \pi r^2 h$
30° 45° 60°
sin $\frac{1}{2}$ $\frac{\sqrt{2}}{2}$ $\frac{\sqrt{3}}{2}$
cos $\frac{\sqrt{3}}{2}$ $\frac{\sqrt{2}}{2}$ $\frac{1}{2}$
tan $\frac{\sqrt{3}}{3}$ 1 $\sqrt{3}$
$2x$
60°
x
30°
$x\sqrt{3}$
45°
$\int \sin x dx = -\cos x +$
$\int \frac{dx}{\cos^2 x} = tgx + C$
$\int tgx dx = -\ln|\cos x| +$
$\int \frac{dx}{\sin x} = \ln\left|tg\frac{x}{2}\right| + C$
$\int \frac{dx}{a^2 + x^2} = \frac{1}{a} arctg$
tan (θ)
10
5
−5
θ/rad
$ax^2 + bx + c = 0$
$a(x^2 + \frac{b}{a}x + \frac{c}{a}) = 0$
$x^2 + 2\frac{b}{2a}x + (\frac{b}{2a})^2 - (\frac{b}{2a})^2$

But you know, I learned something today

- Linear regression can find the trend line so we can make a prediction
- The model should not be too overfitted or underfitted
- R-square scored is a way to find model’s accuracy

https://colab.research.google.com/drive/1YxHMiHZn
EiwnHRFJh2SL83Bj-6eHJzQR?usp=sharing