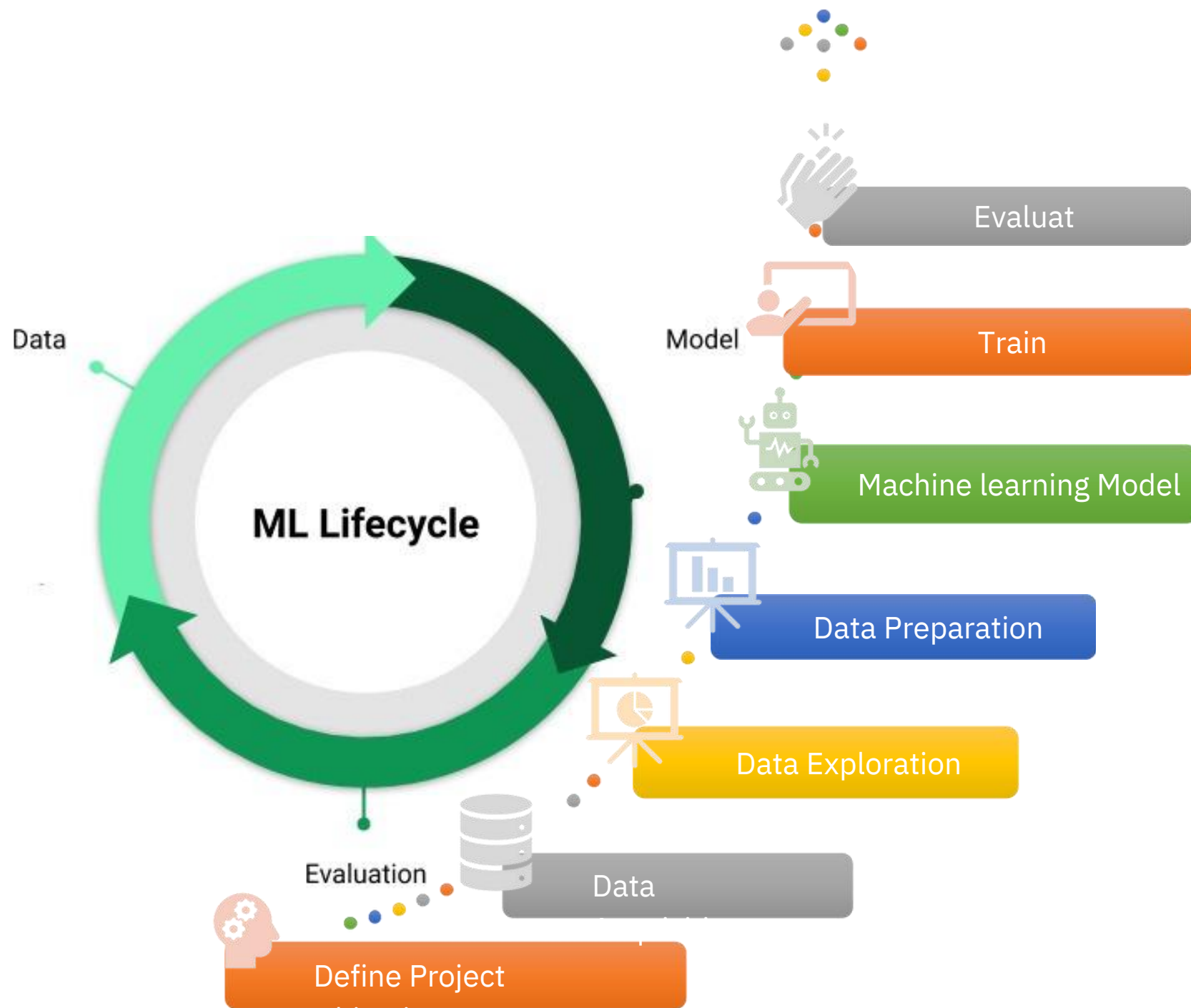


Machine Learning



Supervised Learning

Supervised Learning



Regression



What will be the temperature tomorrow?

84°



Fahrenheit

Classification



Will it be hot or cold tomorrow?

COLD

HOT



Fahrenheit

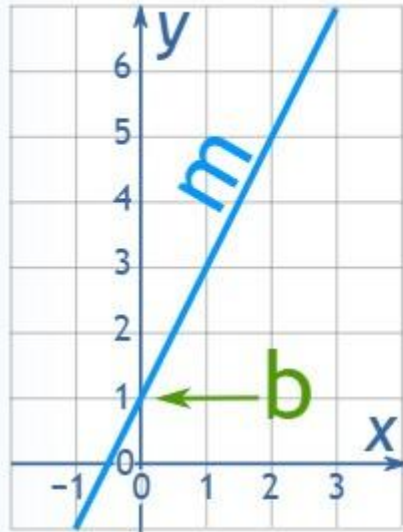
Model

A **Mathematical representation** of a **real world process** in the form of **input-output** relationship.

$$\textit{slices of pizza i'll eat} = 2 * (\textit{hours since last meal}) + 1$$

Lines and Gradient

Equation of a Straight Line



$$y = mx + b$$

Slope or Gradient

y value when **x=0**
(see Y Intercept)

y = how far up

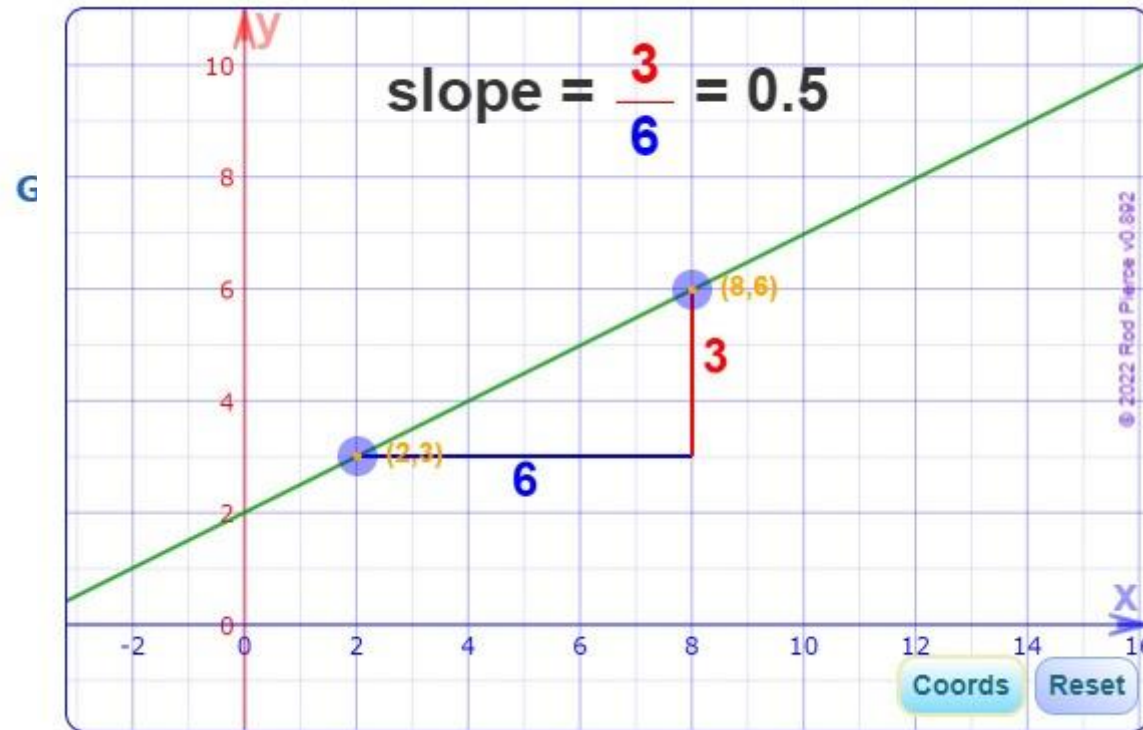
x = how far along

m = Slope or Gradient (how steep the line is)

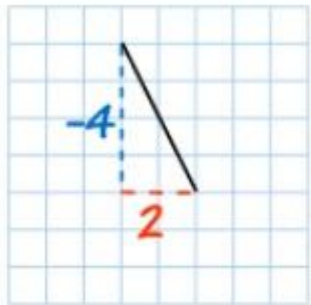
b = value of **y** when **x=0**

Gradient (Slope) of a Straight Line

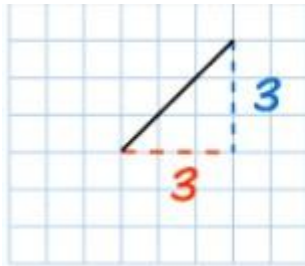
Divide the **change in height** by the **change in horizontal distance**



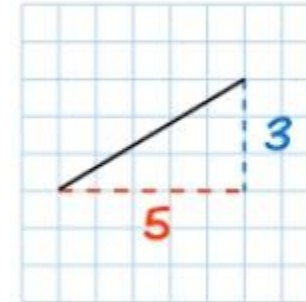
Gradient (Slope) of a Straight Line



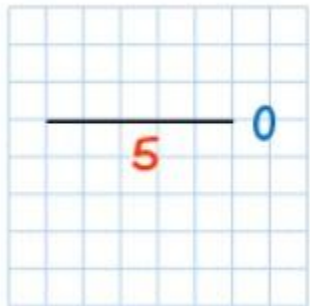
$$\text{Gradient} = \frac{-4}{2} = -2$$



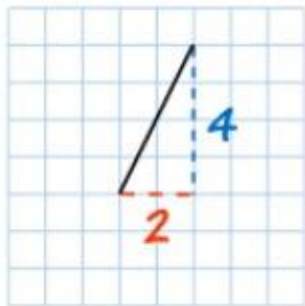
$$\text{The Gradient} = \frac{3}{3} = 1$$



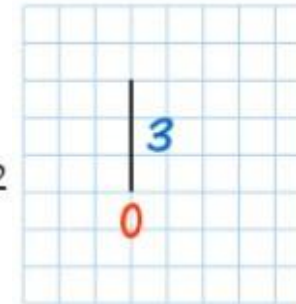
$$\text{The Gradient} = \frac{3}{5} = 0.6$$



$$\text{Gradient} = \frac{0}{5} = 0$$



$$\text{The Gradient} = \frac{4}{2} = 2$$

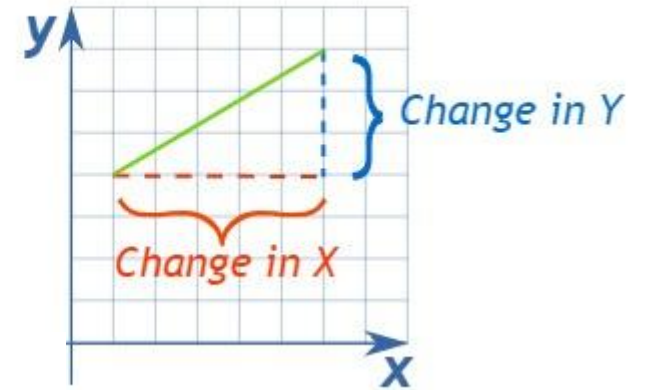


$$\text{Gradient} = \frac{3}{0} = \text{undefined}$$

It is all about slope!

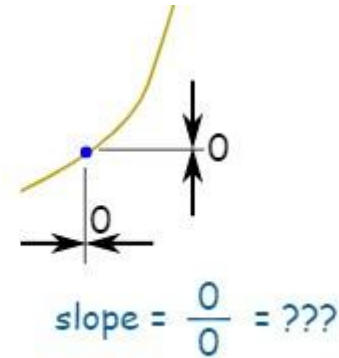
Derivatives

$$\text{Slope} = \frac{\text{Change in Y}}{\text{Change in X}}$$



But how do we find the slope **at a point**?

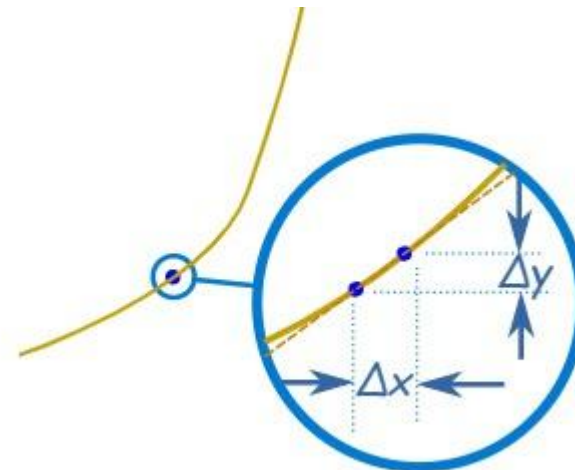
There is nothing to measure!



Sometimes we can't work something out **directly**... but we **can** see what it should be as we get **closer** and **closer**!

But with derivatives we use a small difference ...

... then have it **shrink towards zero**.



Derivatives

To find the derivative of a function $y = f(x)$ we use the slope formula:

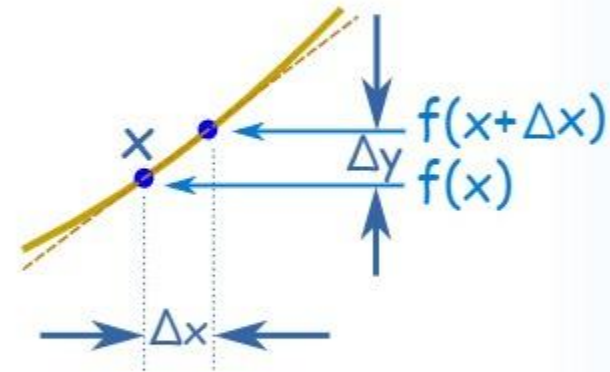
$$\text{Slope} = \frac{\text{Change in Y}}{\text{Change in X}} = \frac{\Delta y}{\Delta x}$$

And (from the diagram) we see that:

x changes from x to $x + \Delta x$
y changes from $f(x)$ to $f(x + \Delta x)$

Now follow these steps:

- Fill in this slope formula: $\frac{\Delta y}{\Delta x} = \frac{f(x + \Delta x) - f(x)}{\Delta x}$
- Simplify it as best we can
- Then make Δx shrink towards zero.



Derivatives

We write **dx** instead of "**Δx heads towards 0**".

And "the derivative of" is commonly written $\frac{d}{dx}$ like this:

$$\frac{d}{dx}x^2 = 2x$$

*"The derivative of x^2 equals $2x$ "
or simply " $d dx$ of x^2 equals $2x$ "*

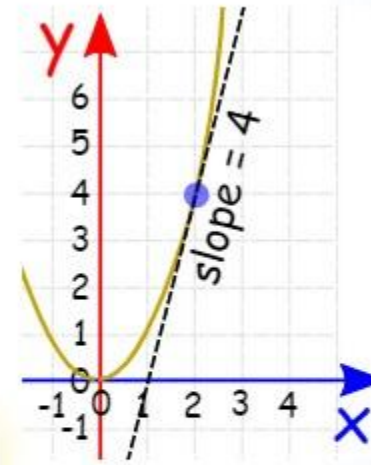
Derivatives

So what does $\frac{d}{dx}x^2 = 2x$ mean?

It means that, for the function x^2 , the slope or "rate of change" at any point is $2x$.

So when $x=2$ the slope is $2x = 4$, as shown here:

Or when $x=5$ the slope is $2x = 10$, and so on.



Note: $f'(x)$ can also be used for "the derivative of":

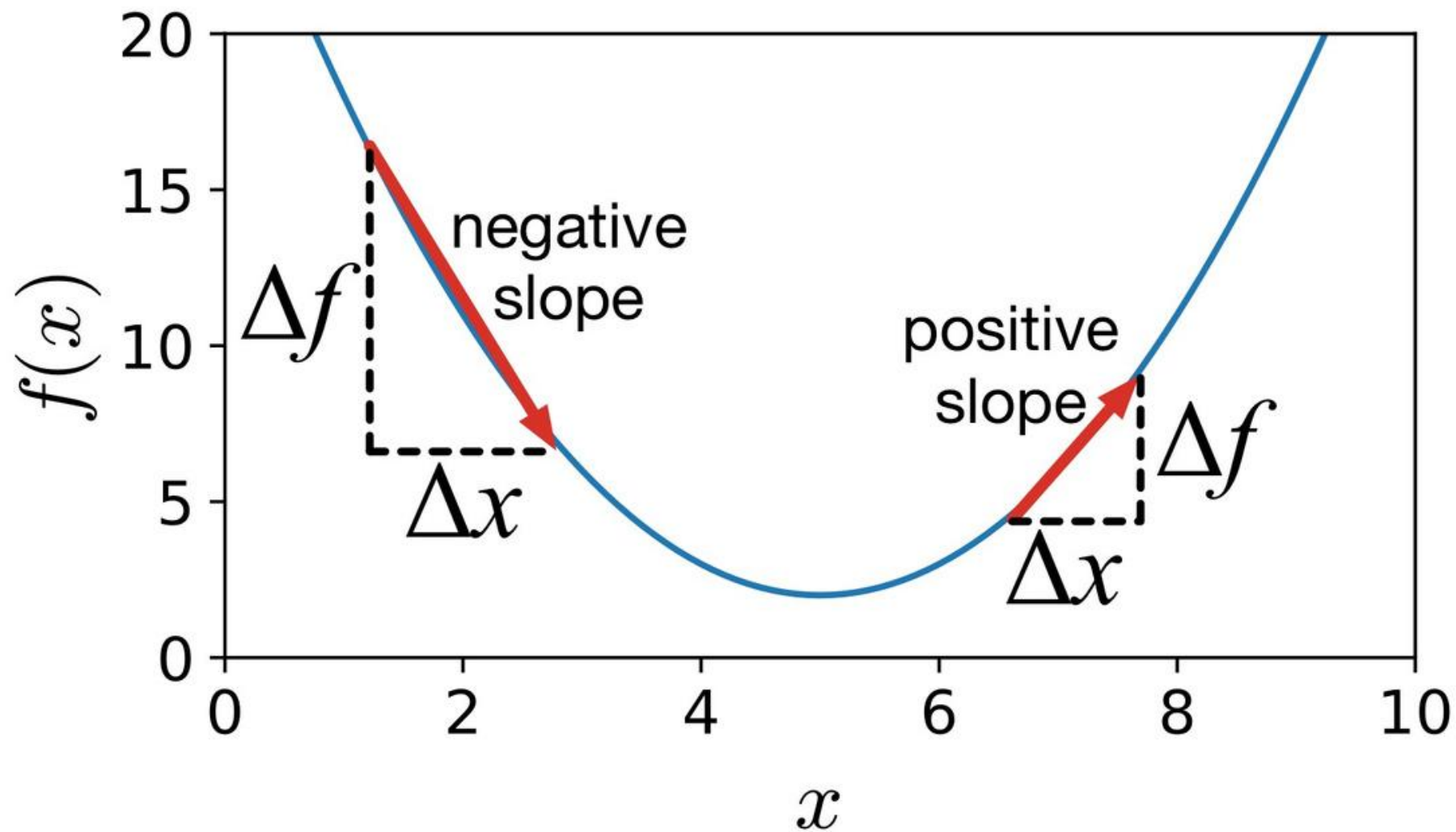
$$f'(x) = 2x$$

"The derivative of $f(x)$ equals $2x$ "
or simply " f -dash of x equals $2x$ "

Derivative Rules

Common Functions	Function	Derivative
Constant	c	0
Line	x	1
	a	a
Square	x^2	$2x$
Square	\sqrt{x}	$(1/2)x^{-1/2}$
Exponential	e^x	e^x
Logarithm	$\ln(ax)$	$1/x$

Derivative

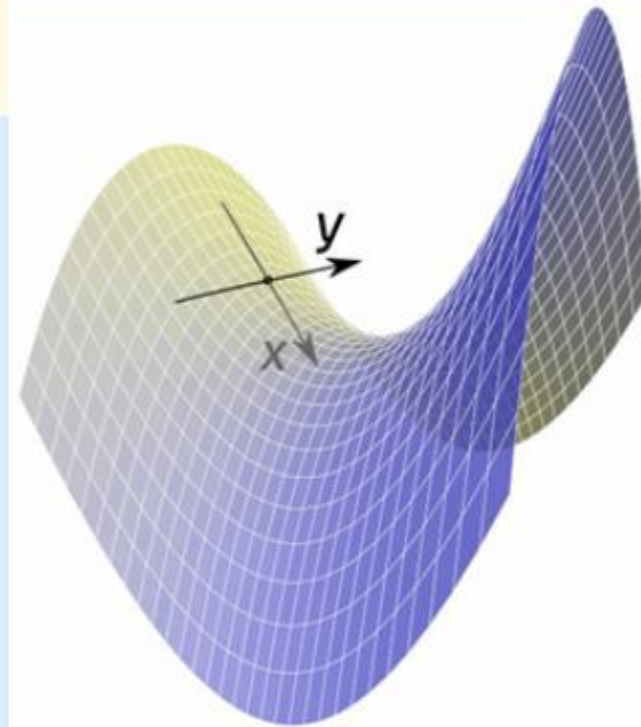


Partial Derivatives

Example: a function for a surface that depends on two variables x and y

When we find the slope in the x direction (while keeping y fixed) we have found a partial derivative.

Or we can find the slope in the y direction (while keeping x fixed).



Partial Derivatives

But what about a function of **two variables** (x and y):

$$f(x, y) = x^2 + y^3$$

We can find its **partial derivative with respect to x** when we treat **y as a constant** (imagine y is a number like 7 or something):

$$f'_x = 2x + 0 = 2x$$

Explanation:

- the derivative of x^2 (with respect to x) is $2x$
- we **treat y as a constant**, so y^3 is also a constant (imagine $y=7$, then $7^3=343$ is also a constant), and the derivative of a constant is 0

To find the partial derivative **with respect to y**, we treat **x as a constant**:

$$f'_y = 0 + 3y^2 = 3y^2$$

$$f(x) = 0.5x^2 + y^2$$

Let's assume we are interested in a gradient at point $p(10,10)$:

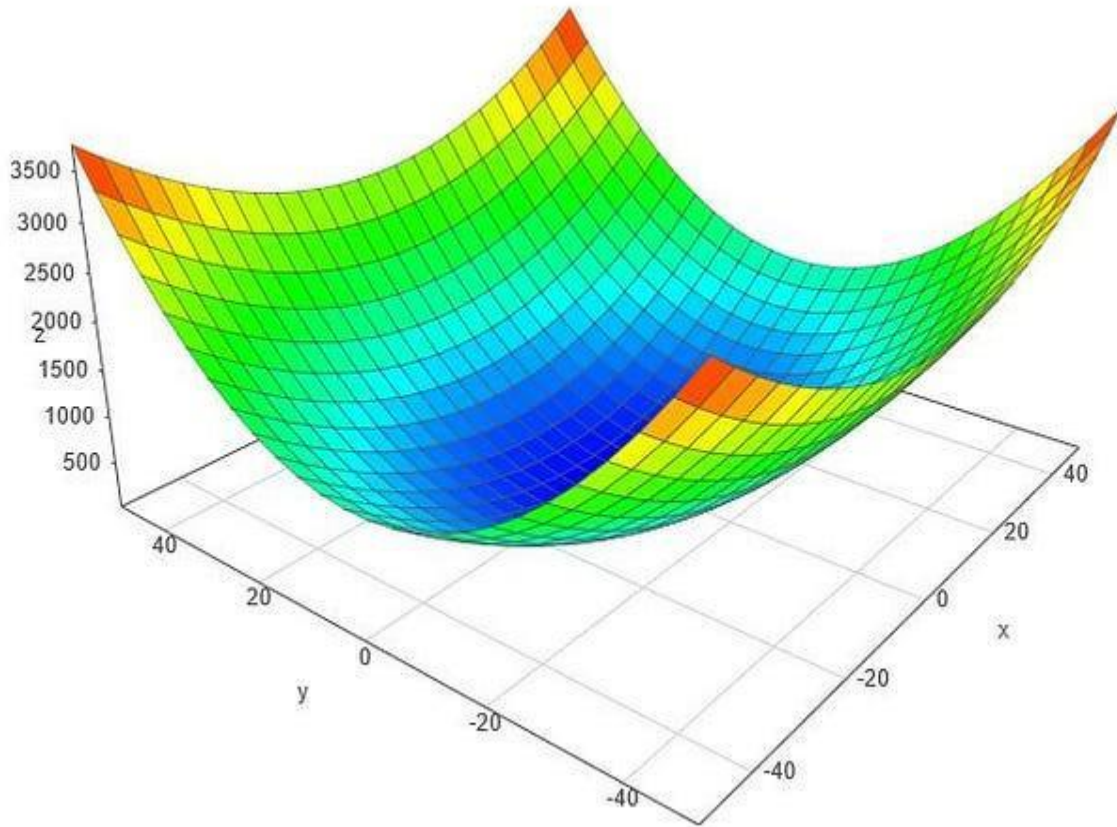
$$\frac{\partial f(x, y)}{\partial x} = x, \quad \frac{\partial f(x, y)}{\partial y} = 2y$$

so consequently:

$$\nabla f(x, y) = \begin{bmatrix} x \\ 2y \end{bmatrix}$$

$$\nabla f(10, 10) = \begin{bmatrix} 10 \\ 20 \end{bmatrix}$$

By looking at these values we conclude that the slope is twice steeper along the y axis.



Finding Maxima and Minima using Derivatives

Where is a function at a high or low point?
Calculus can help!

In a smoothly changing function a maximum or minimum is always where the function **flattens out** (except for a **saddle point**).

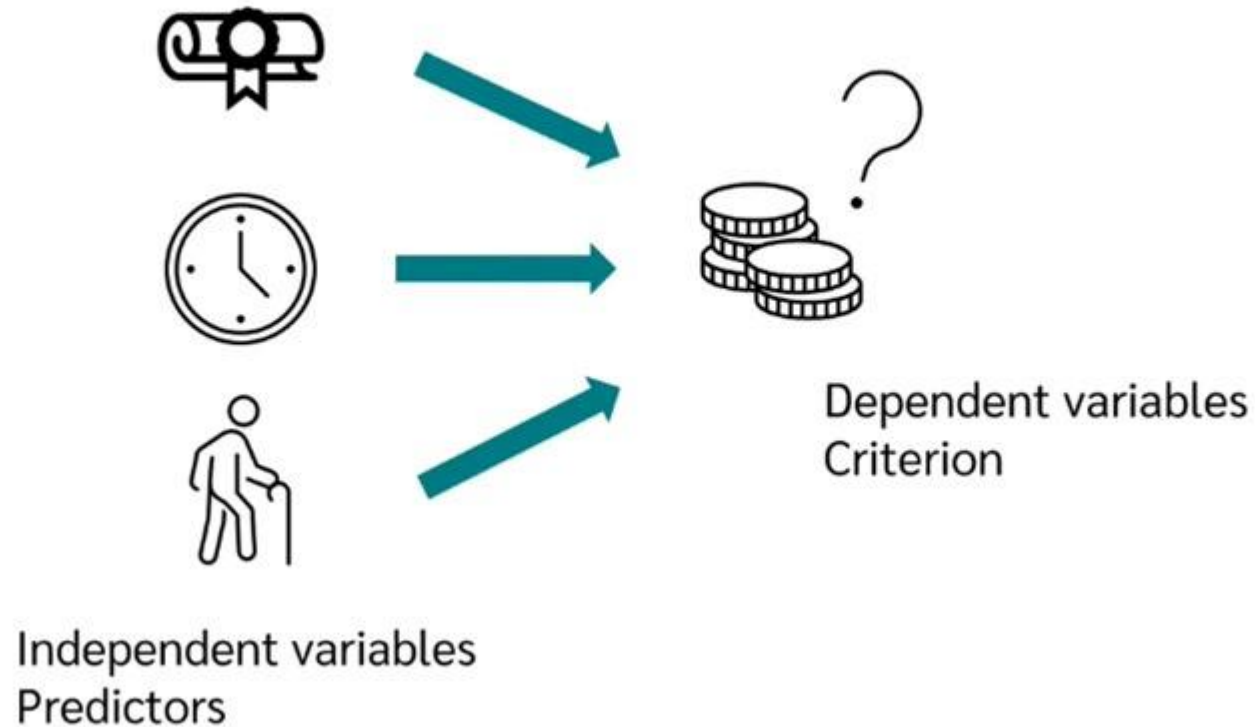


Where does it flatten out? Where the slope is zero. Where is the slope zero? The **Derivative** tells us!

Linear Regression

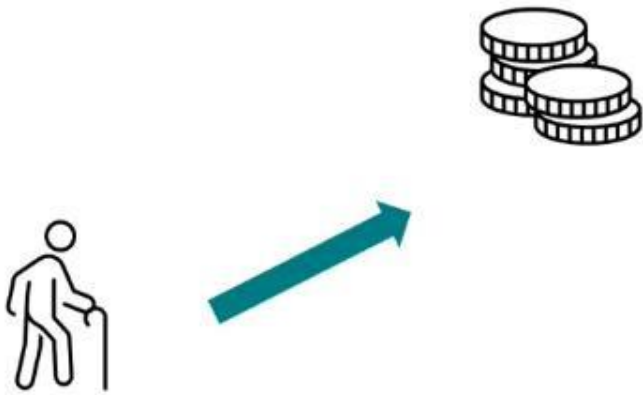
Regression Analysis

A **regression analysis** makes it possible to infer or predict another variable on the basis of one or more variables.

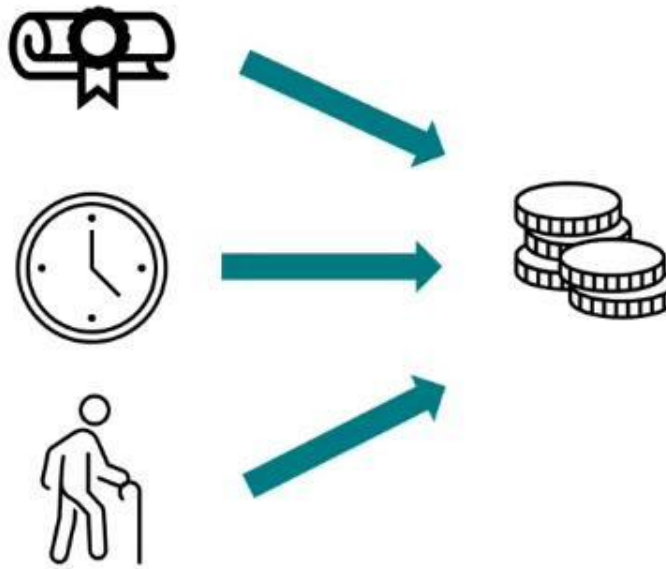


Forms of Regression Analysis

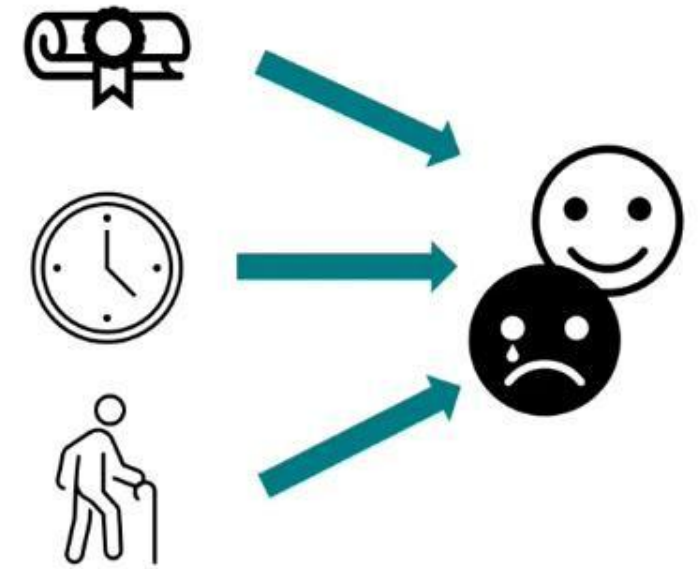
Simple linear Regression



Multiple linear Regression

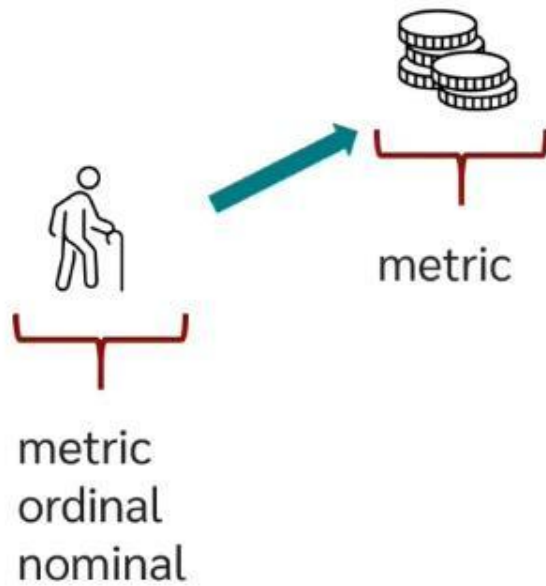


Logistic Regression

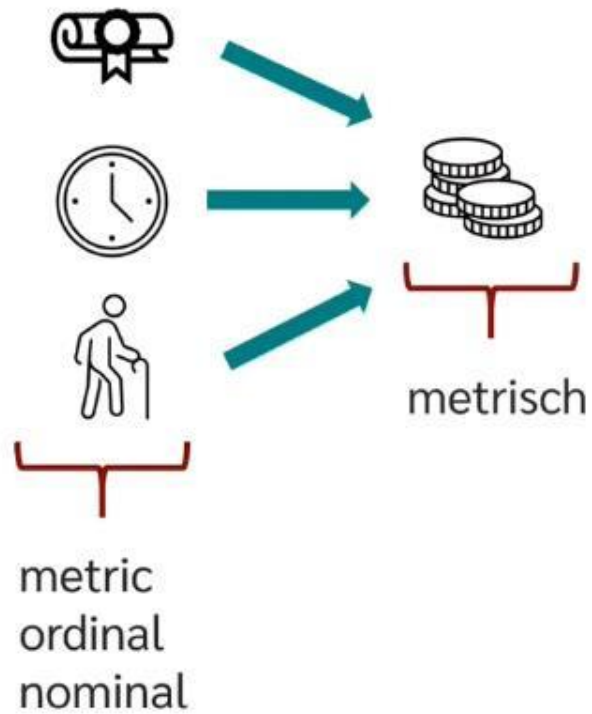


Forms of Regression Analysis

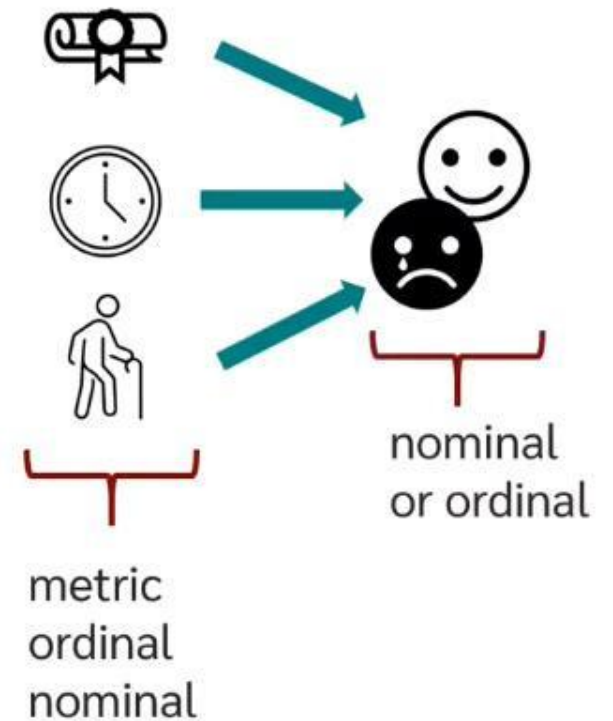
Simple linear Regression



Multiple linear Regression



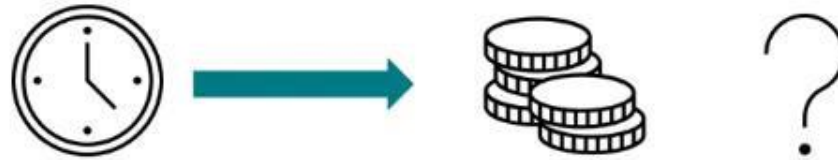
Logistic Regression



Forms of Regression Analysis

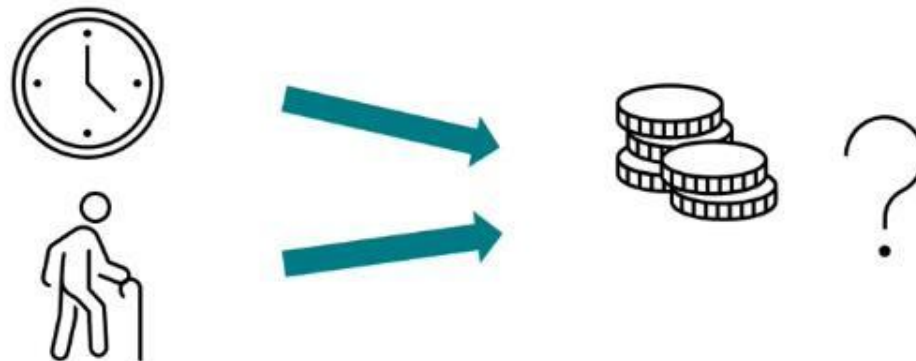
Simple linear regression

Does **the weekly working time** have an influence on the **hourly salary** of employees?

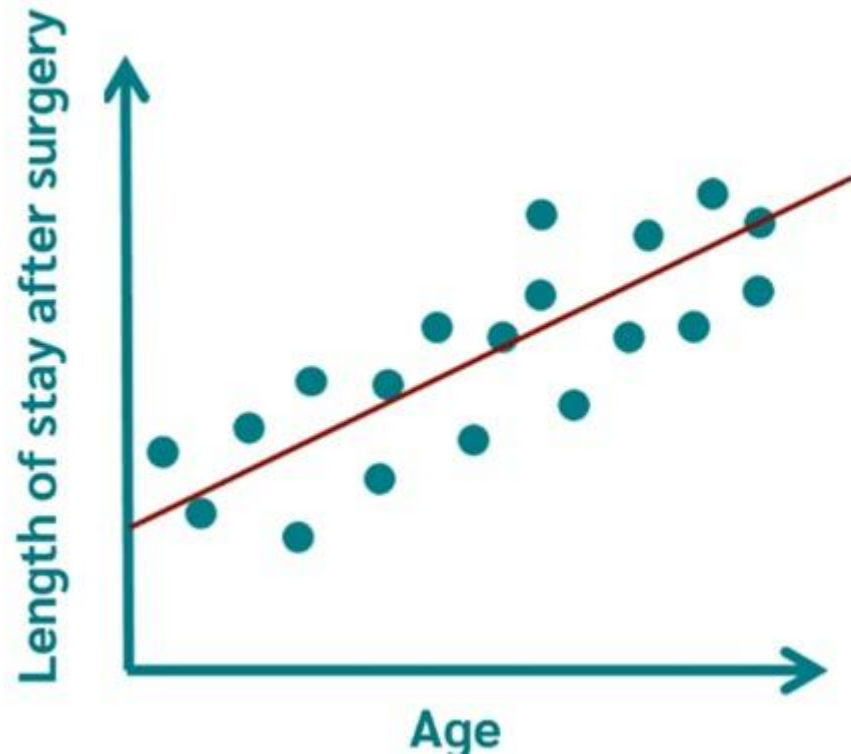


Multiple linear regression

Do the **weekly working hours** and the **age** of employees have an influence on their **hourly salary**?



Simple Linear Regression



Estimated length
of stay

Age

$$\hat{y} = b \cdot x + a$$

$$\hat{y} = 0.14 \cdot x + 1.2$$

$$5.82 = 0.14 \cdot 33 + 1.2$$

The model

A simple linear regression model predicts the output as a linear function of the input feature x :

$$f(x; w_0, w_1) = w_0 + w_1 x$$



We refer to w_0 and w_1 as the *parameters* of the model.

To choose w_0 and w_1 , we are given a data set of previous input-output measurements:

$$\left\{ (x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots (x^{(N)}, y^{(N)}) \right\}$$

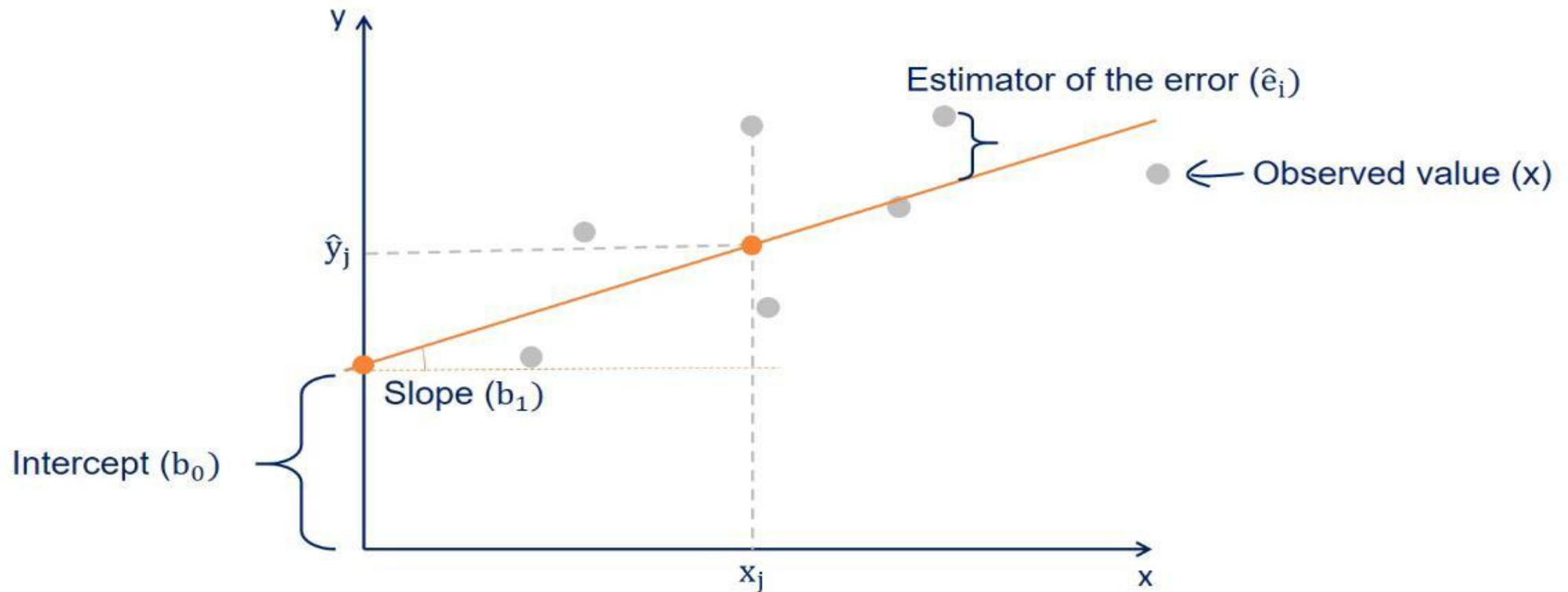
I will sometimes just write this as:

$$\left\{ (x^{(n)}, y^{(n)}) \right\}_{n=1}^N$$

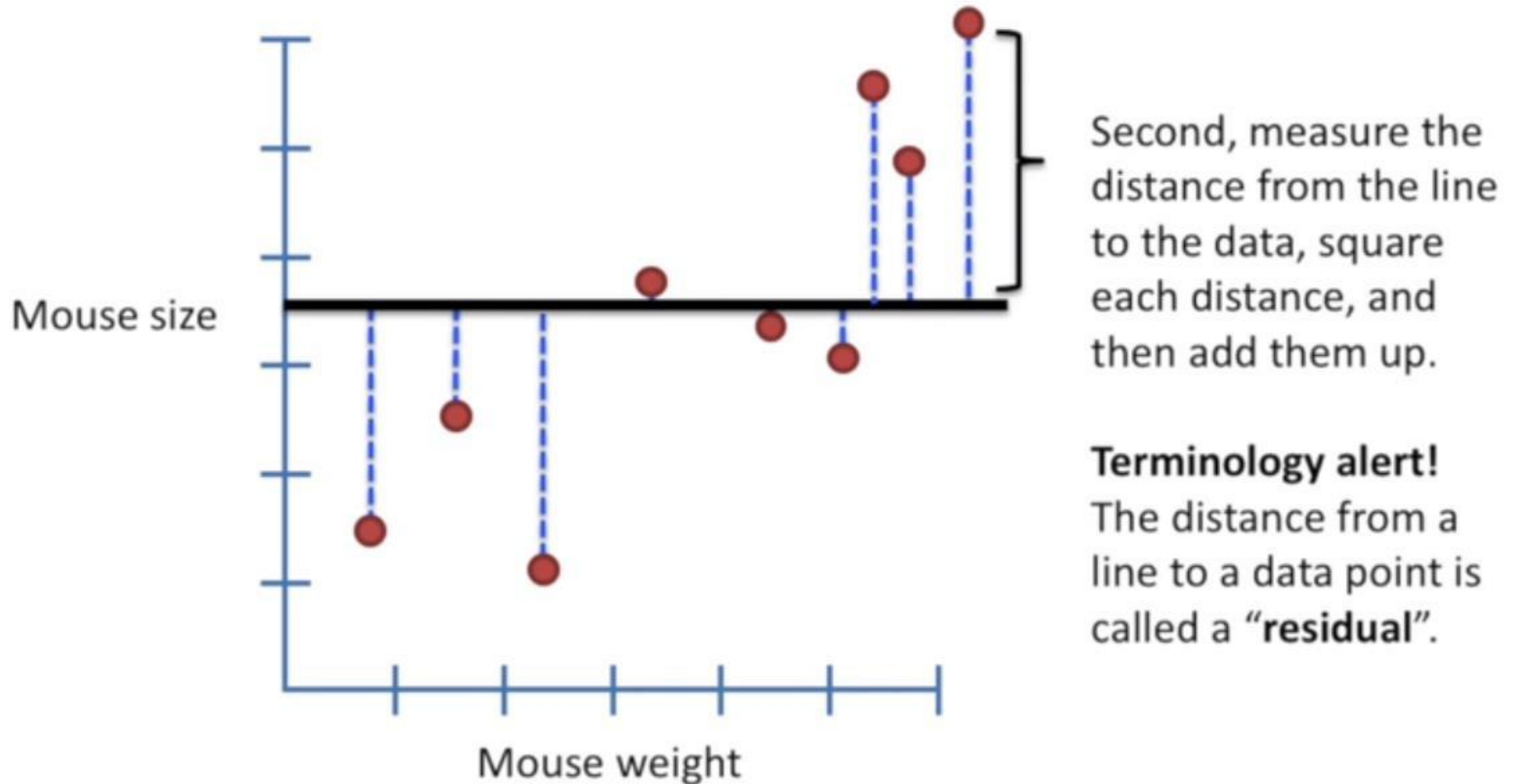
How do we choose w_0 and w_1 based on the data? We need some way to measure the “goodness” or “badness” of the parameters, given the data.

Simple Linear Regression

$$\hat{y}_i = b_0 + b_1 x_i$$

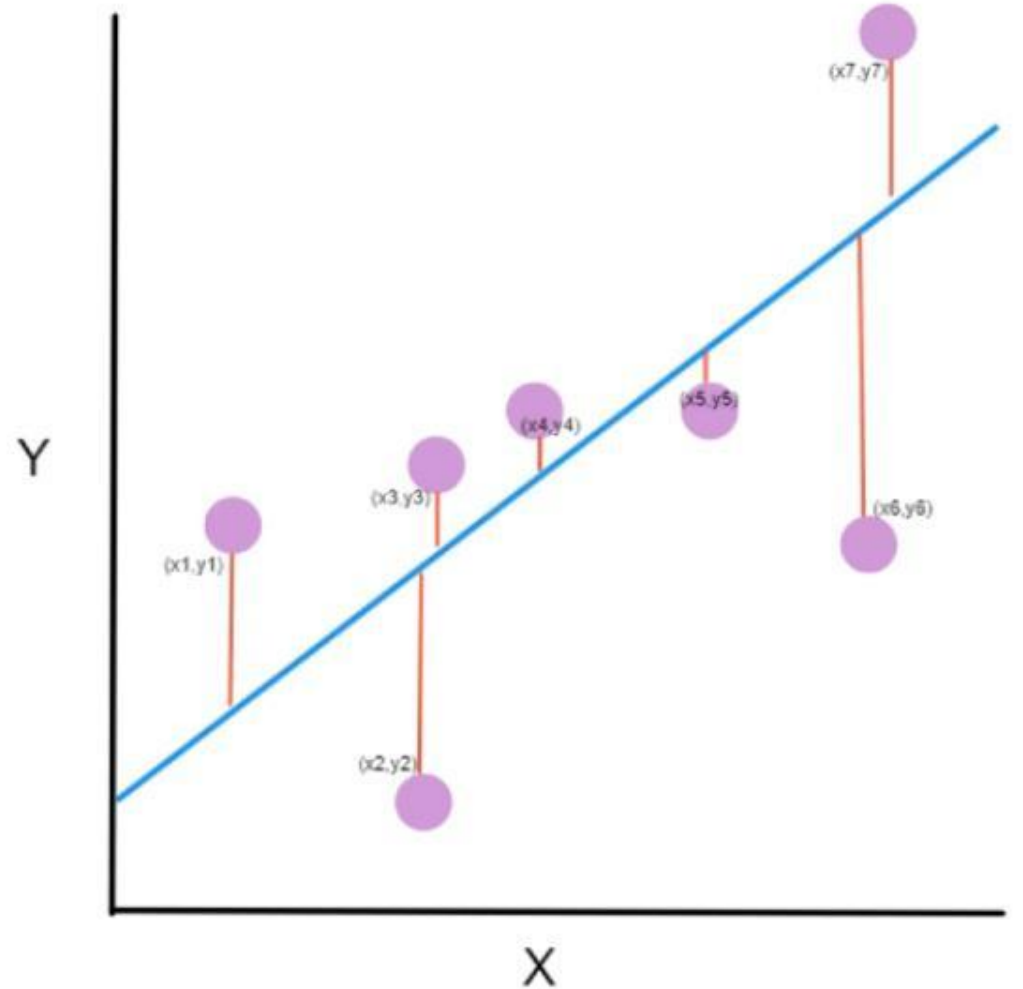


Simple Linear Regression



Error/Loss/Cost Function

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



Multiple Linear Regression

- Unlike simple linear regression, multiple linear regression can include two or more independent variables.
- The goal is to estimate one variable based on several other variables.
- The variable to be estimated is called the **dependent variable (criterion)**. The variables that are used for prediction are called **independent variables (predictors)**.
- Multiple linear regression is often used in empirical social research as well as in market research. In both areas it is of interest to find out what influence different factors have on a variable.

Good models require multiple regressions, in order to address the higher complexity of problems



$$\text{College GPA} = 0.275 + 0.0017 * \text{SAT}$$



MULTIPLE REGRESSION EQUATION

$$\hat{y} = b_0 + b_1x_1 + b_2x_2 + \cdots + b_kx_k$$



independent
variable



independent
variable



independent
variable

Multiple Linear Regression

Simple linear
Regression

$$\hat{y} = b \cdot x + a$$



Multiple linear
Regression

$$\hat{y} = b_1 \cdot x_1 + b_2 \cdot x_2 + \dots + b_k \cdot x_k + a$$

- The coefficients are interpreted similarly to the linear regression equation.
- If all independent variables are 0, the value a is obtained.
- If an independent variable changes by one unit, the associated coefficient b indicates by how much the dependent variable changes.



$$f_{w,b}(x) = 0.1x_1 + 4x_2 + 10x_3 - 2x_4 + 80$$

size bedrooms floors years base price



Metrics

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

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

less robust to outliers

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

less robust to outliers

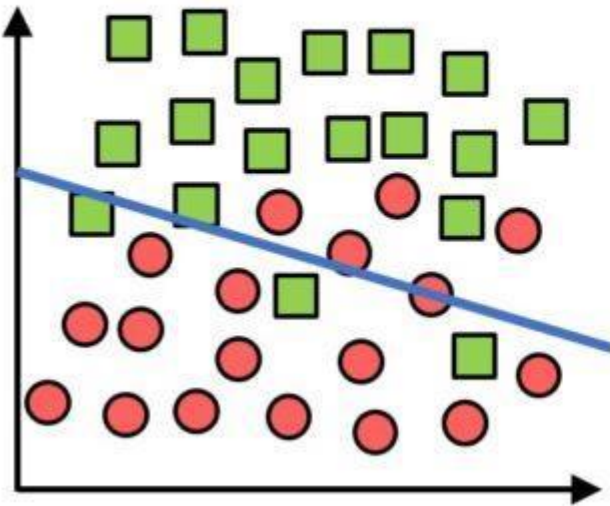
If you want to train a model which focuses on **reducing large outlier errors** then **MSE** is the better choice, whereas if this **isn't important** and you would prefer greater interpretability then **MAE** would be **better**.

So a robust system or metric must be **less affected by outliers**. In this scenario it is easy to conclude that MSE may be less robust than **MAE**, since the **squaring** of the **errors** will **enforce a higher importance on outliers**.

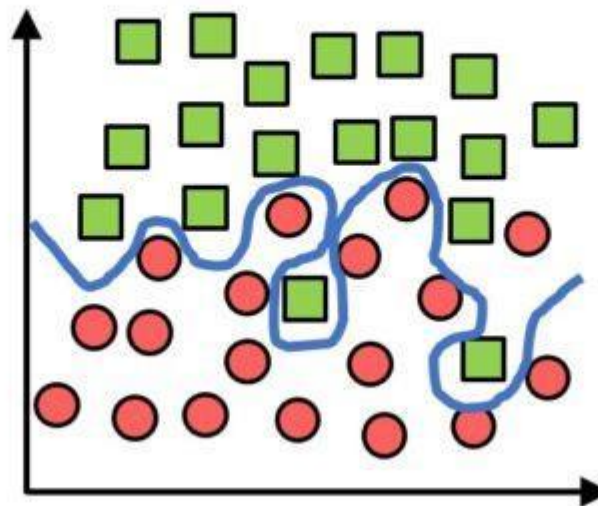
ML Concepts

Generalization, Overfitting, and Underfitting

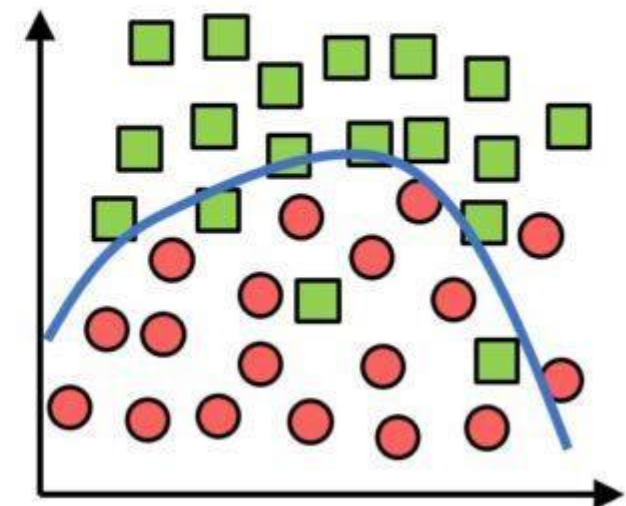
Underfitting



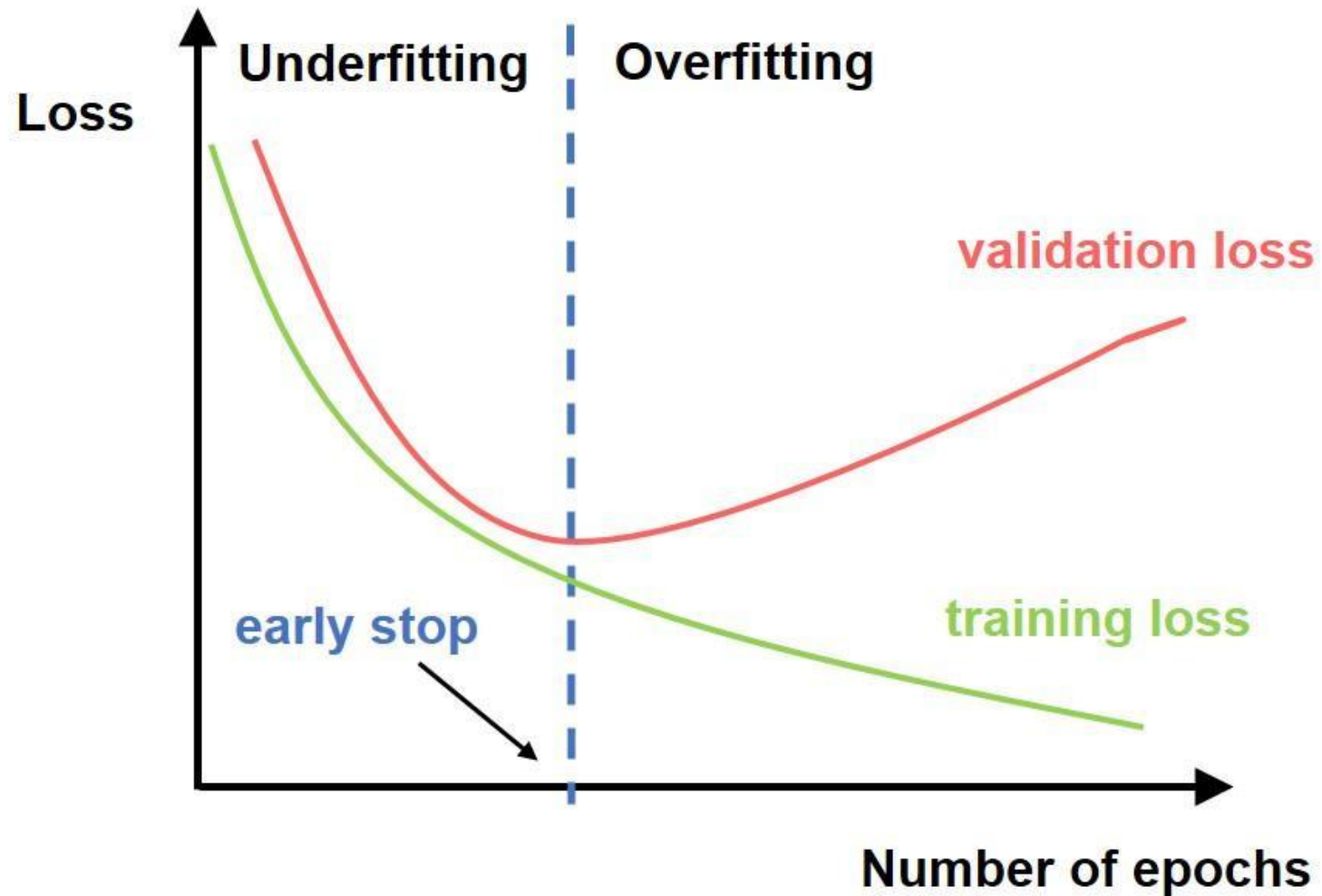
Overfitting

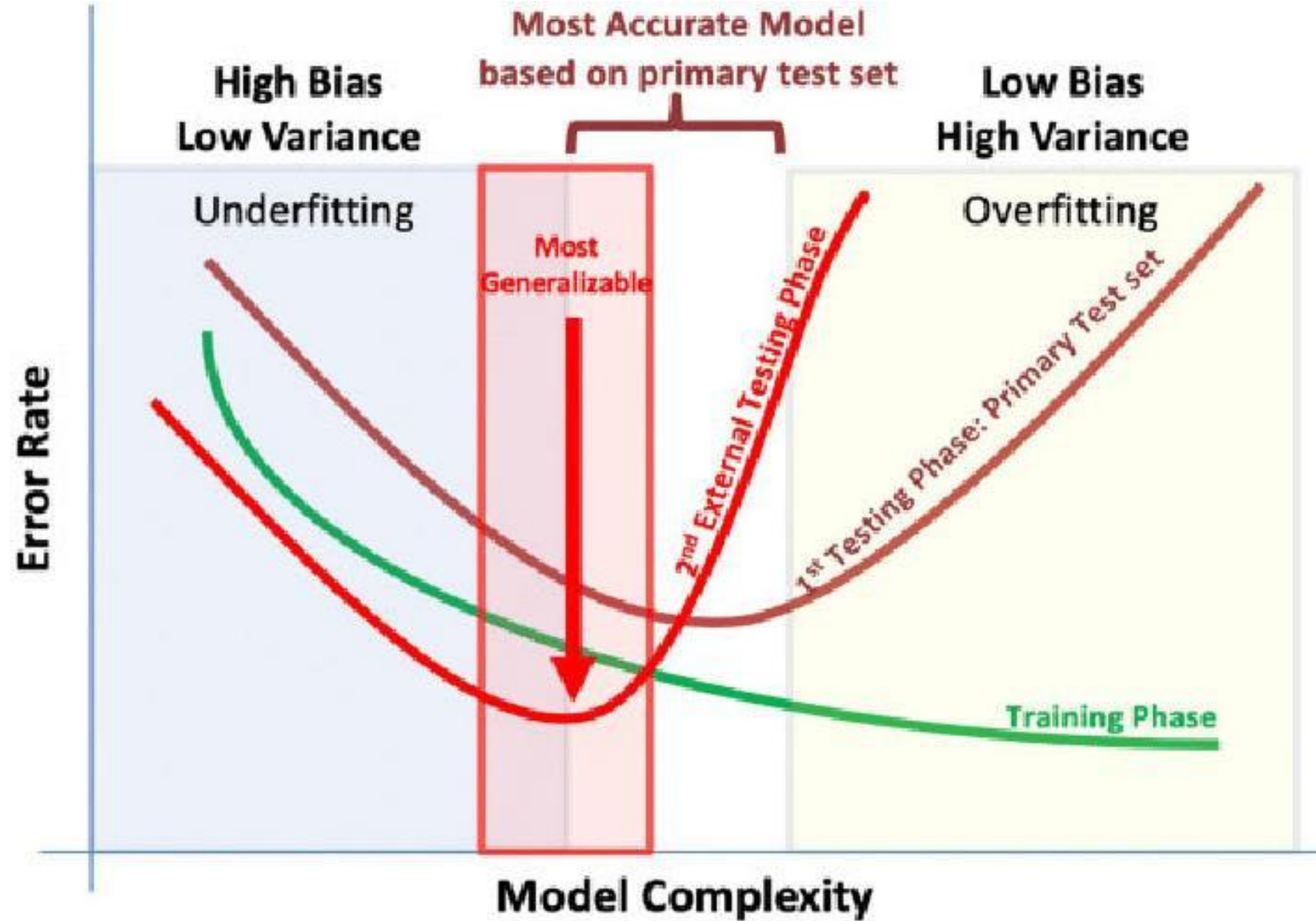


Optimal



Generalization, Overfitting, and Underfitting





How to avoid Overfitting

- **Simplifying The Model Early Stopping**
- **Adding More Data To The Training Set**
- **Cross-validation Remove unnecessary**
- **features Regularization**
-
-

Regularisation Techniques

Handle **Overfitting**

With

Regularization



dataaspirant.com

L1 Regularization

- This adds a penalty equal to the **L1 norm** of the weights **vector** (sum of the absolute value of the coefficients). It will shrink some parameters to **zero**.
- Hence some variables will not play any role in the model.
- L1 regression can be seen as a way to **select features** in a model.

$$\text{L1: } R(\theta) = \|\theta\|_1 = \sum_{i=1}^n |\theta_i|$$

$$\text{LossFunction} = \frac{1}{N} \sum_{i=1}^N (\hat{Y} - Y)^2 + \lambda \sum_{i=1}^N |\theta_i|$$

L2 Regularization

- This adds a **penalty** equal to the **L2 norm** of the weights vector (sum of the squared values of the coefficients).
- It will force the parameters to be **relatively small**.

$$\text{L1: } R(\theta) = \|\theta\|_1 = \sum_{i=1}^n |\theta_i|$$

$$\text{L2: } R(\theta) = \|\theta\|_2^2 = \sum_{i=1}^n \theta_i^2$$

$$\text{LossFunction} = \frac{1}{N} \sum_{i=1}^N (\hat{Y} - Y)^2 + \lambda \sum_{i=1}^N \theta_i^2$$

$$J_{\text{test}}(\theta) = \frac{1}{2m_{\text{test}}} \sum_{i=1}^{m_{\text{test}}} (\hat{y}_i - y_i)^2$$

where

- $h_{\theta}(x(i))$ is the predicted value of some datapoint $x(i)$
- $y(i)$ is original

The penalized cost function looks like this

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m \left(h_{\theta}(x^{(i)}) - y^{(i)} \right)^2 + \underbrace{\lambda \sum_{j=1}^n \theta_j^2}_{\text{Regularization Term}}$$

start at θ_1

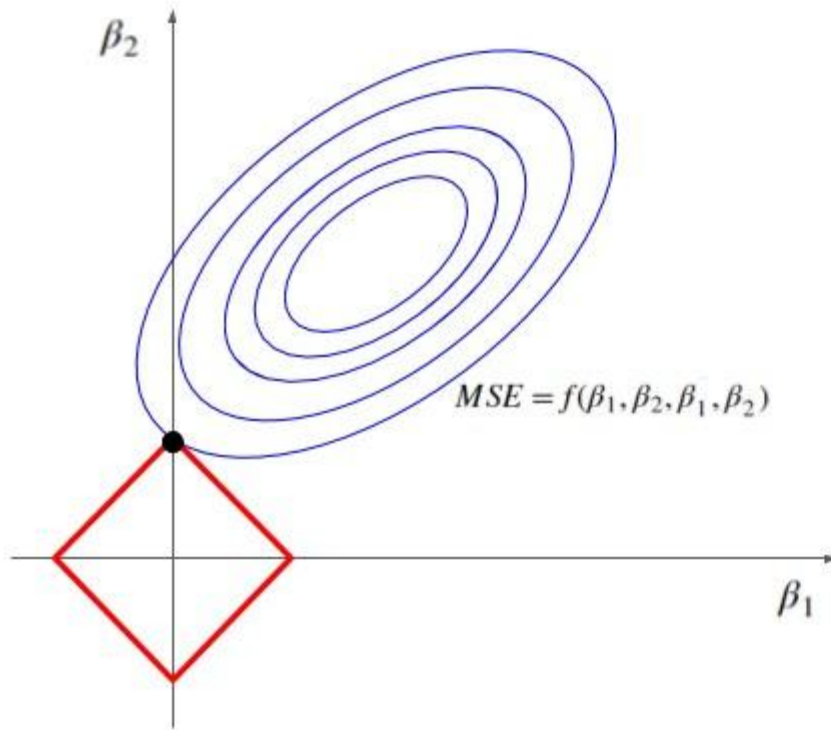
Regularization Parameter

where-

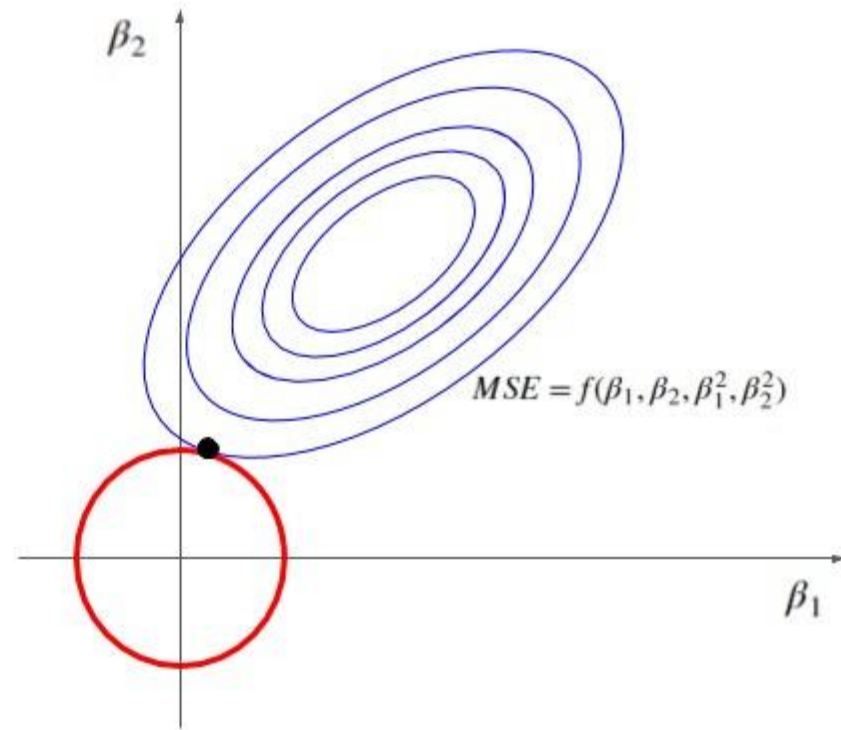
- λ is the tuning parameter that decides how much we want to penalize the flexibility of our model. It can be tuned using cross-validation.

The difference is that : while shrinking the quota, L1 tends to cut off some factors by turning their coefficients to zero, while L2 tends to shrinking these coefficients to a tiny number (none zero) , keep some of their influence on Y.

Find β_1 β_2 to minimize MSE with restriction on β_1 β_2



L1 regularization



L2 regularization

<i>Comparison of L1 and L2 regularization</i>	
<i>L1 regularization</i>	<i>L2 regularization</i>
Sum of absolute value of weights	Sum of square of weights
Sparse solution	Non-sparse solution
Multiple solutions	One solution
Built-in feature selection	No feature selection
Robust to outliers	Not robust to outliers (due to the square term)

By this I mean the number of solutions to arrive at one point. L1 regularization uses Manhattan distances to arrive at a single point, so there are many routes that can be taken to arrive at a point. L2 regularization uses Euclidean distances, which will tell you the fastest way to get to a point. This means the L2 norm only has 1 possible solution.

Which solution is less Computationally expensive? L2

Classification Metrics

What is Confusion Matrix?

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error
	Negative	False Positive (FP) Type I Error	True Negative (TN)

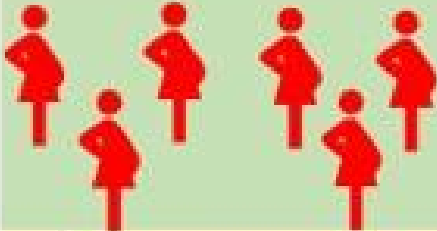
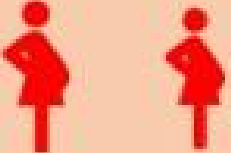
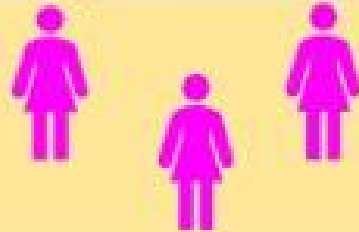
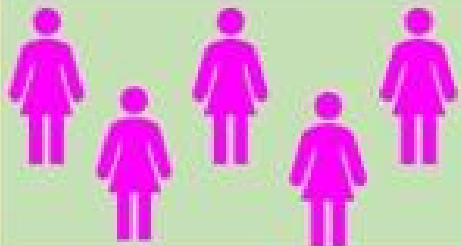
What is Confusion Matrix?

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error
	Negative	False Positive (FP) Type I Error	True Negative (TN)

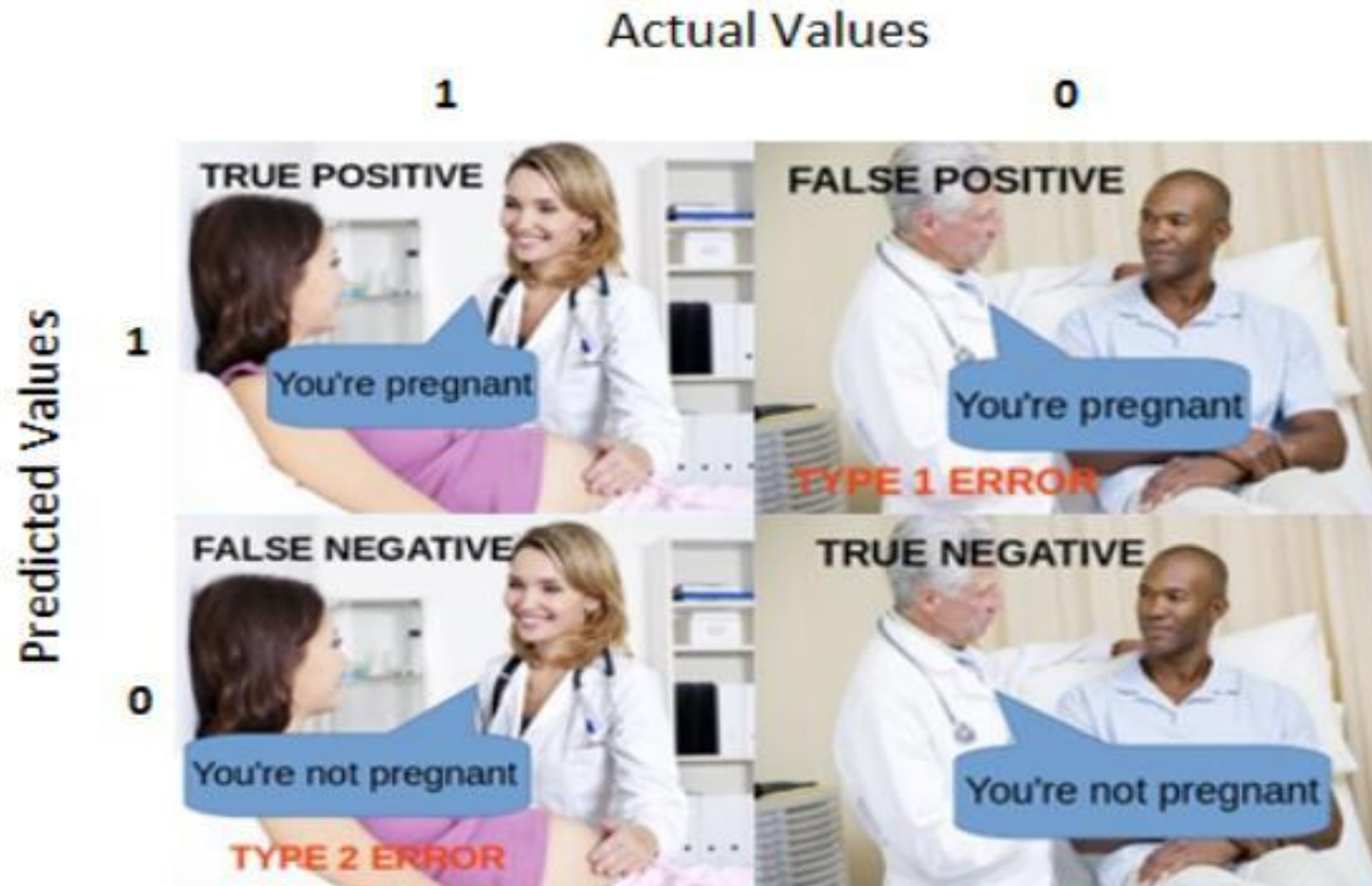
- True Positive (TP) — When the model says that the **patient has cancer** and the patient **actually has it**
 - False Positive (FP) — When the model says that the **patient has cancer** but the patient **doesn't have it**
 - True Negative (TN) — When the model says that the **patient does not have cancer** and the patient **actually doesn't have it**
 - False Negative (FN) — When the model says that the **patient doesn't have cancer** but the patient **actually has it**.
- We don't want this, do we?*

We don't

What is Confusion Matrix?

		PREDICTED VALUES	
		Pregnant	Not Pregnant
ACTUAL VALUES	Pregnant		
	Not Pregnant		

What is Confusion Matrix?



Performance evaluation Measures

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error
	Negative	False Positive (FP) Type I Error	True Negative (TN)

$$\text{Accuracy} = (TP+TN) / (TP+FP+FN+TN)$$

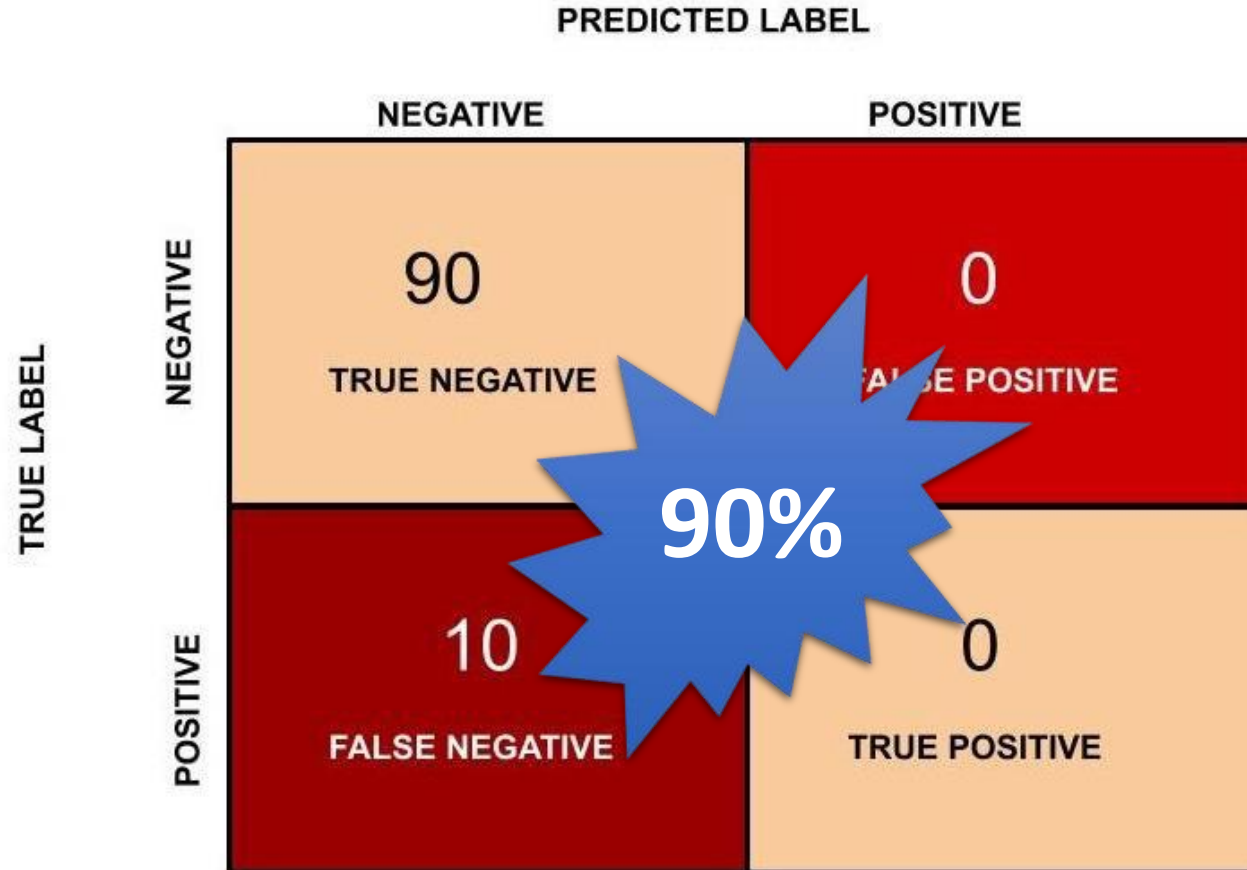
Exercise

- We select 100 people which includes pregnant women, not pregnant women and men with fat belly. Let us assume out of this 100 people 40 are pregnant and the remaining 60 people include not pregnant women and men with fat belly. We now use a machine learning algorithm to predict the outcome.
- Out of 40 pregnant women 30 pregnant women are classified correctly and the remaining 10 pregnant women are classified as not pregnant by the machine learning algorithm.
- On the other hand, out of 60 people in the not pregnant category, 55 are as not pregnant and the remaining 5 are classified as pregnant.

Example

		PREDICTED LABEL	
		NEGATIVE	POSITIVE
TRUE LABEL	NEGATIVE	55 TRUE NEGATIVE	5 FALSE POSITIVE
	POSITIVE	10 FALSE NEGATIVE	30 TRUE POSITIVE

Is accuracy the best measure?



$$\text{Accuracy} = (TP+TN) / (TP+FP+FN+TN)$$

Performance evaluation Measures

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error
	Negative	False Positive (FP) Type I Error	True Negative (TN)

$$\text{Accuracy} = (TP+TN) / (TP+FP+FN+TN)$$

$$\text{Precision} = TP / (TP+FP)$$

$$\text{Recall} = TP / TP+FN$$

$$\text{F1 Score} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

MultiClass Classification

