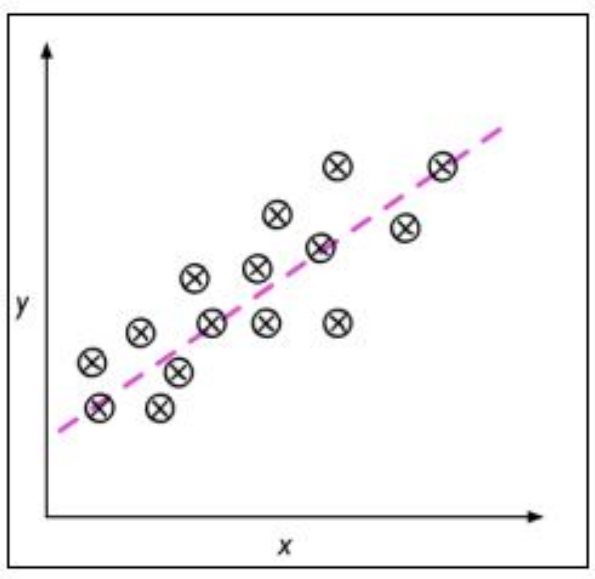
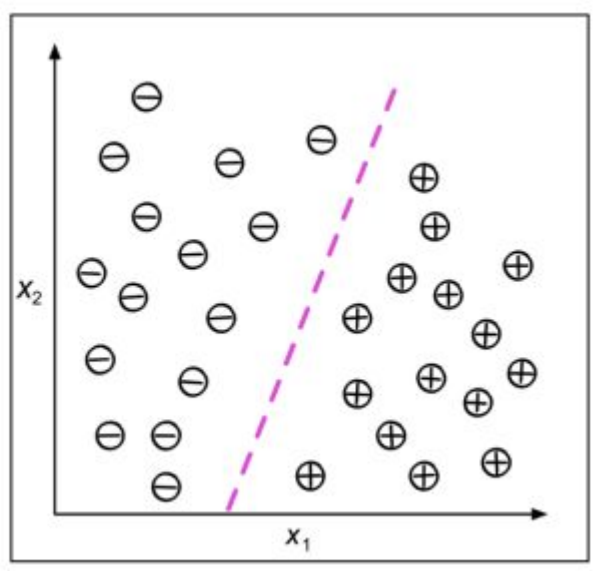
Introduction to Logistic regression

Don't let the name confuse you. For historical reasons it is called 'regression'.

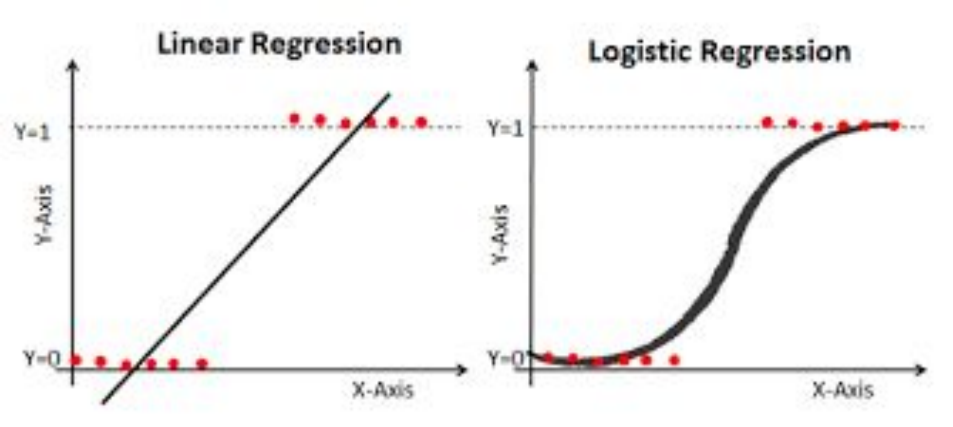
Logistic regression is a *CLASSIFICATION* algorithm!

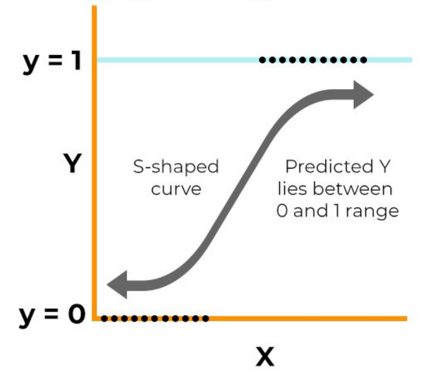
Regression vs Classification



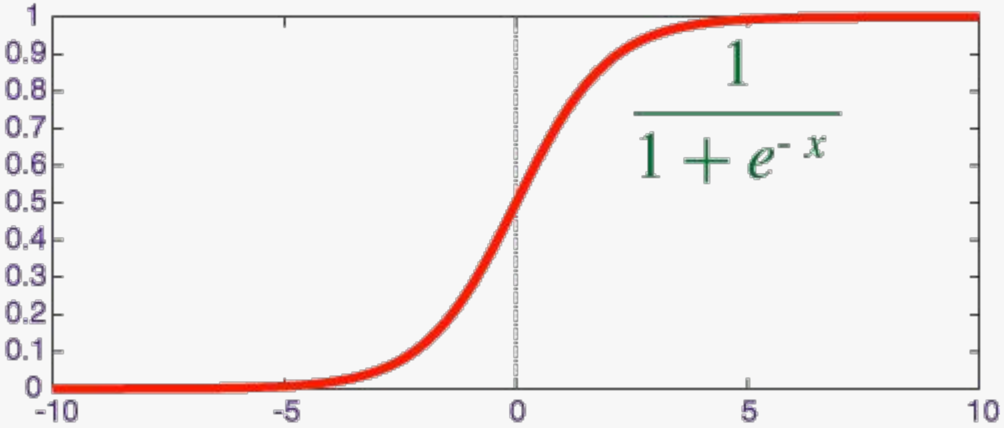


From straight lines to curves

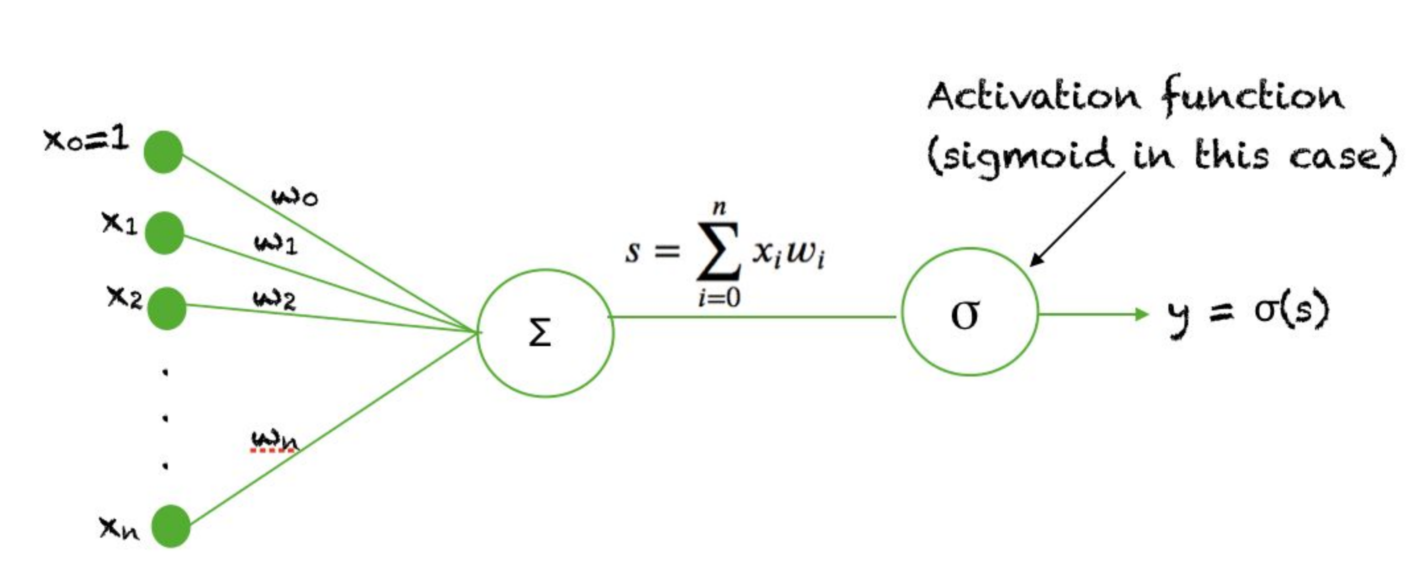




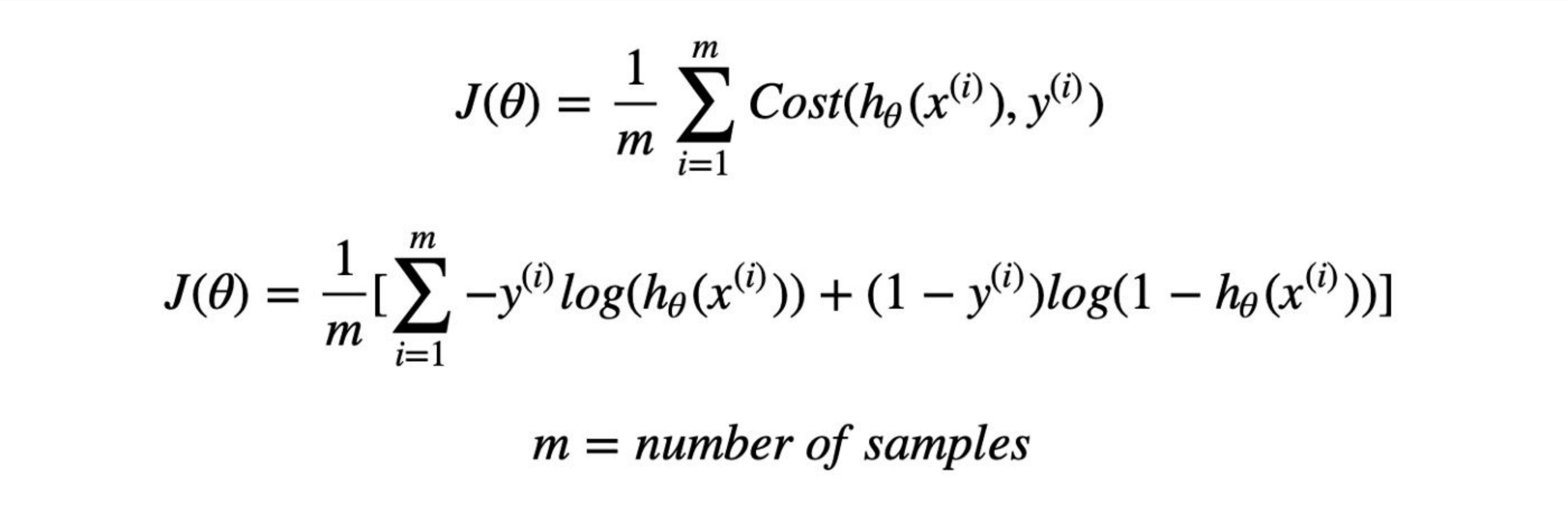
Sigmoid Function



Turning number predictions into class predictions



Cost function for logistic regression

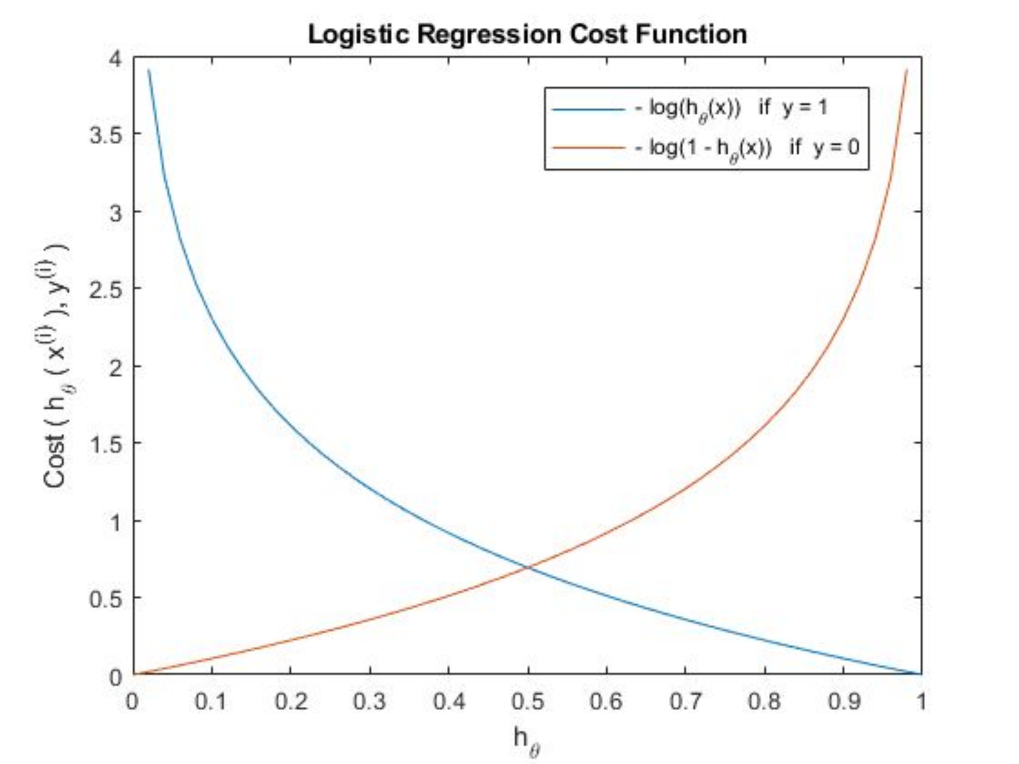


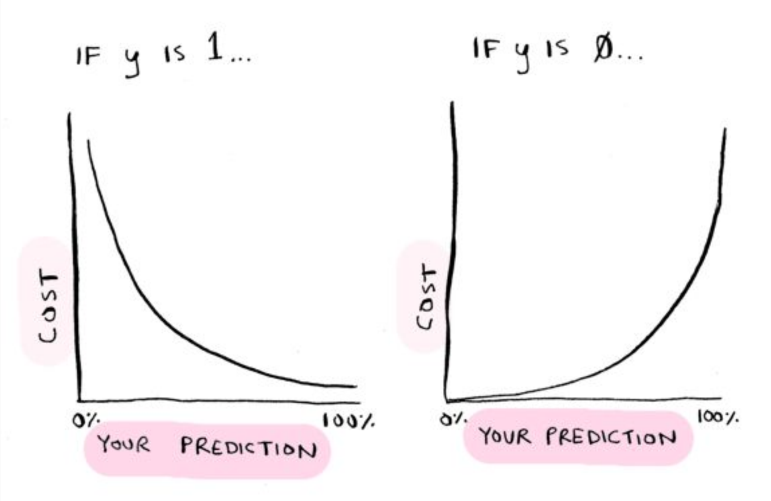
$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(h_\theta(x^{(i)}), y^{(i)})$$

$$J(\theta) = \frac{1}{m}[\sum_{i=1}^{m} -y^{(i)} log(h_\theta(x^{(i)})) + (1 - y^{(i)})log(1 - h_\theta(x^{(i)}))]$$

$$m = number\ of\ samples$$

Why logarithm?



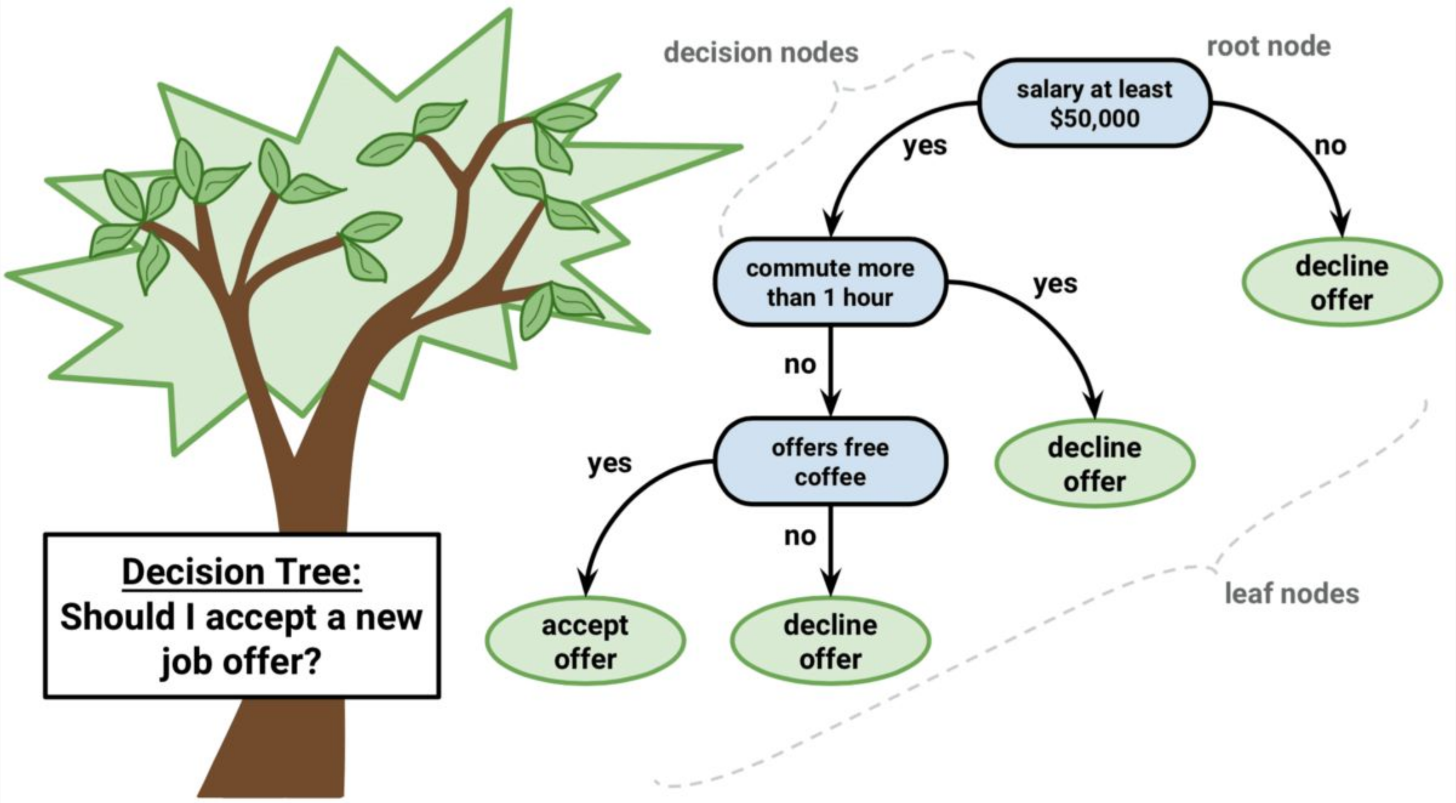


$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost\big(h_\theta(x^{(i)}) - y^{(i)}\big)$$

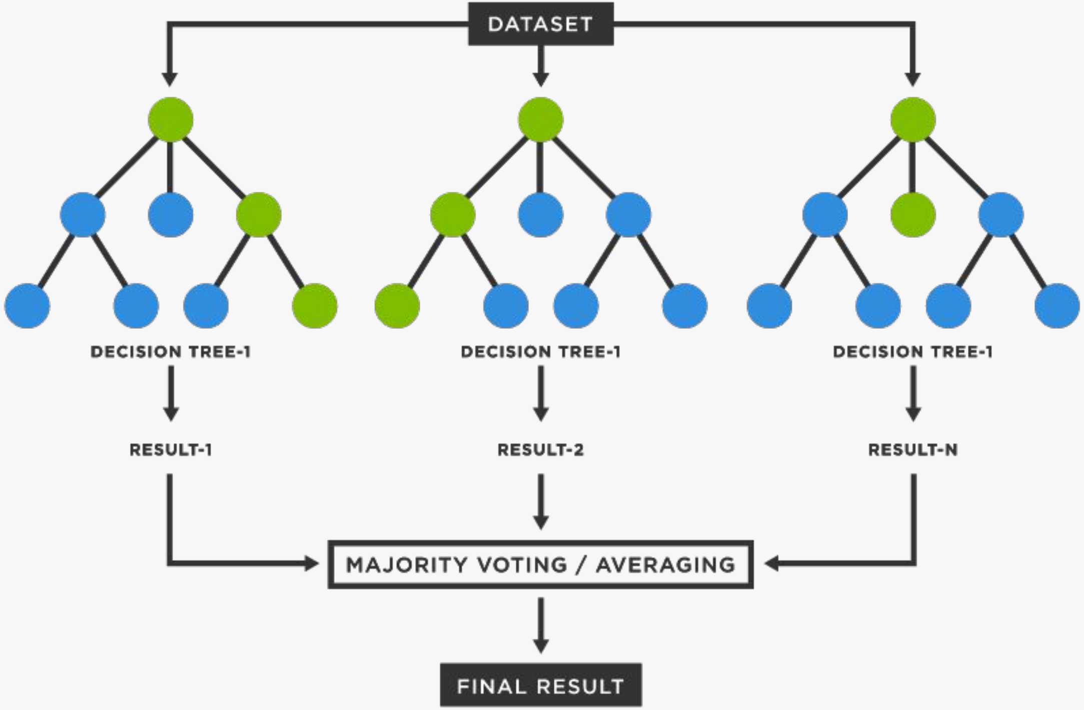
$$\text{Cost}(h_\theta(x), y) = \begin{cases} -\log\left(h_\theta(x)\right) & \text{if y=1} \\ -\log\left(1 - h_\theta(x)\right) & \text{if y=0} \end{cases}$$

Log loss calculation

Non-Parametric models



Ensemble models

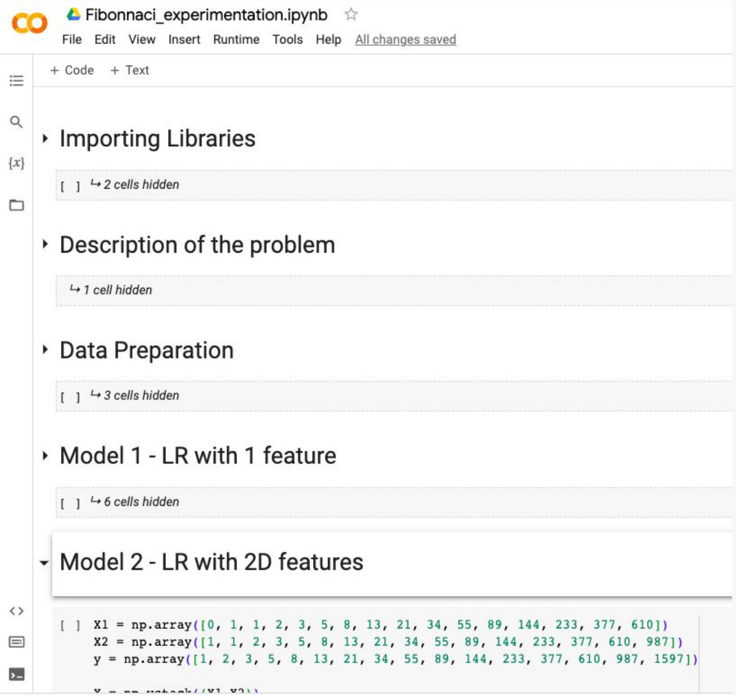


Module 2

Data Science process

Data Engineering Vs Data Science : The number and frequency of experiments

Organizing your experiments



Importing Libraries

[ ] ↳ 2 cells hidden

Description of the problem

↳ 1 cell hidden

Data Preparation

[ ] ↳ 3 cells hidden

Model 1 - LR with 1 feature

[ ] ↳ 6 cells hidden

Model 2 - LR with 2D features

```
[ ] X1 = np.array([0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610])
    X2 = np.array([1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987])
    y = np.array([1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597])

    X = np.vstack((X1,X2))
```

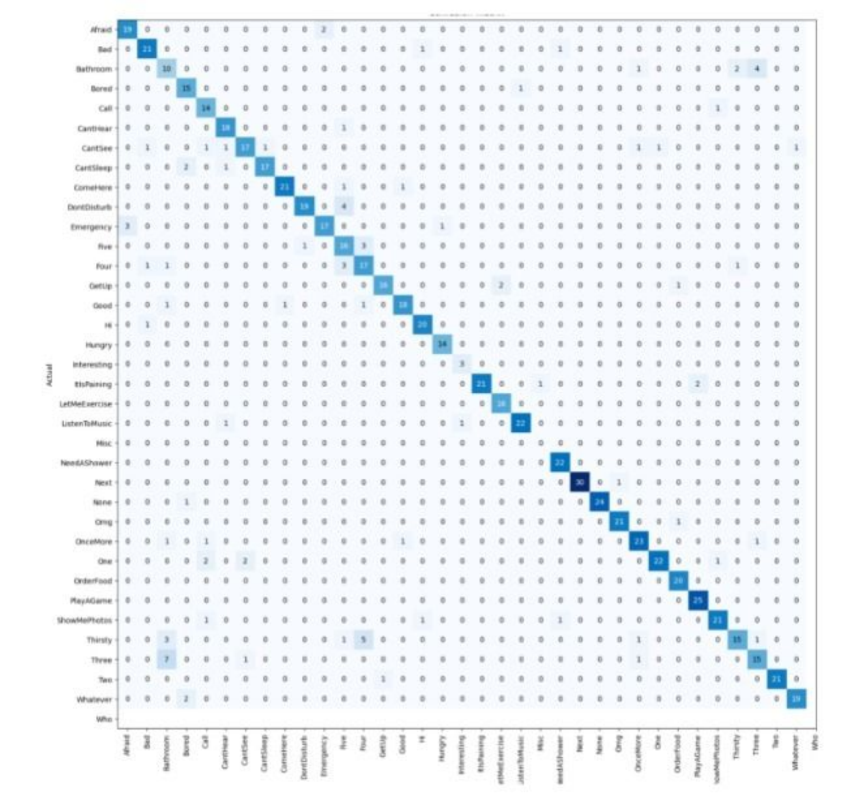
Introducing ML communities

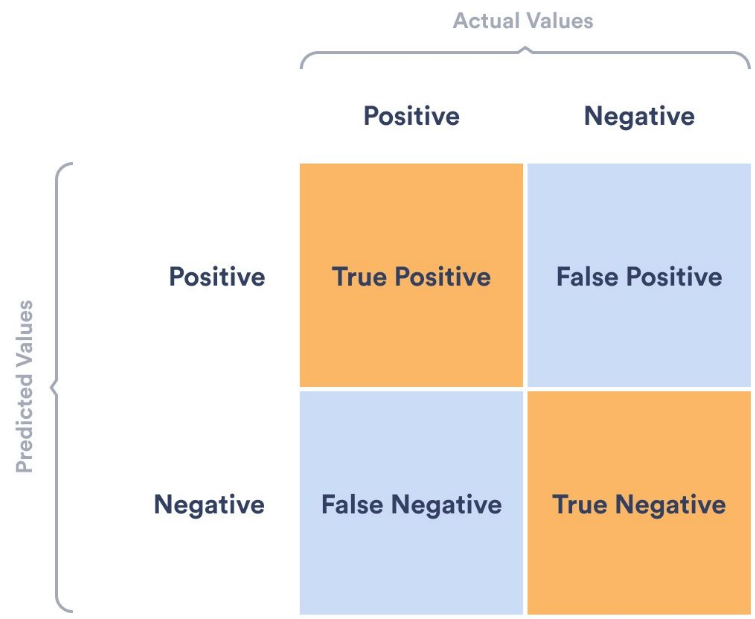
- Kaggle
- Huggingface 🤗

How is my model doing?

Metrics notebook demo

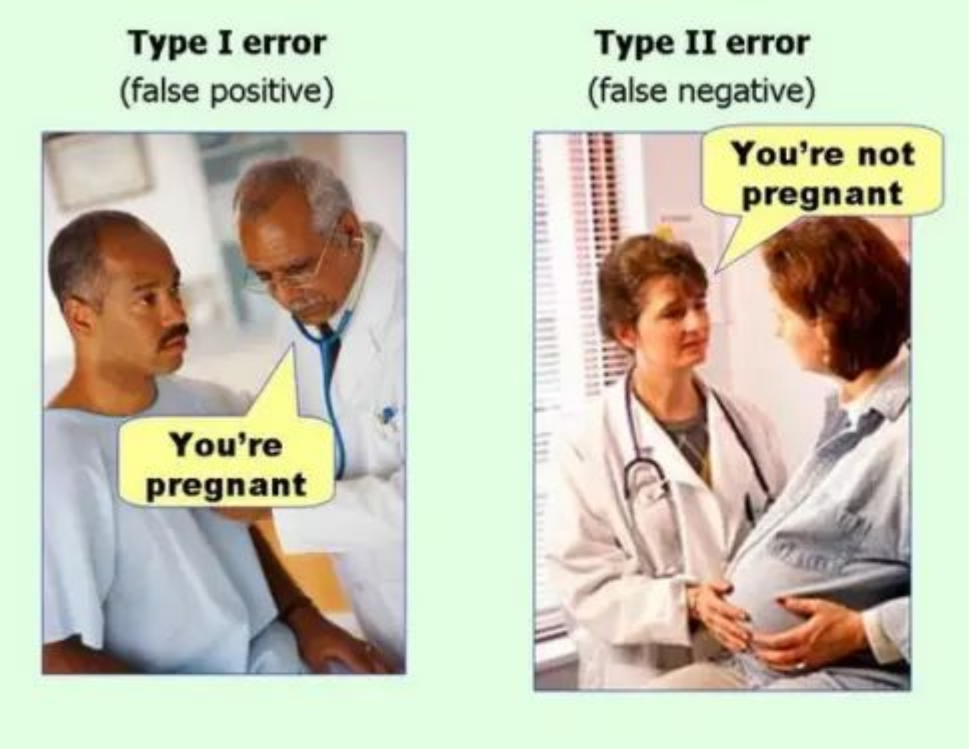
Prediction errors - Confusion Matrix





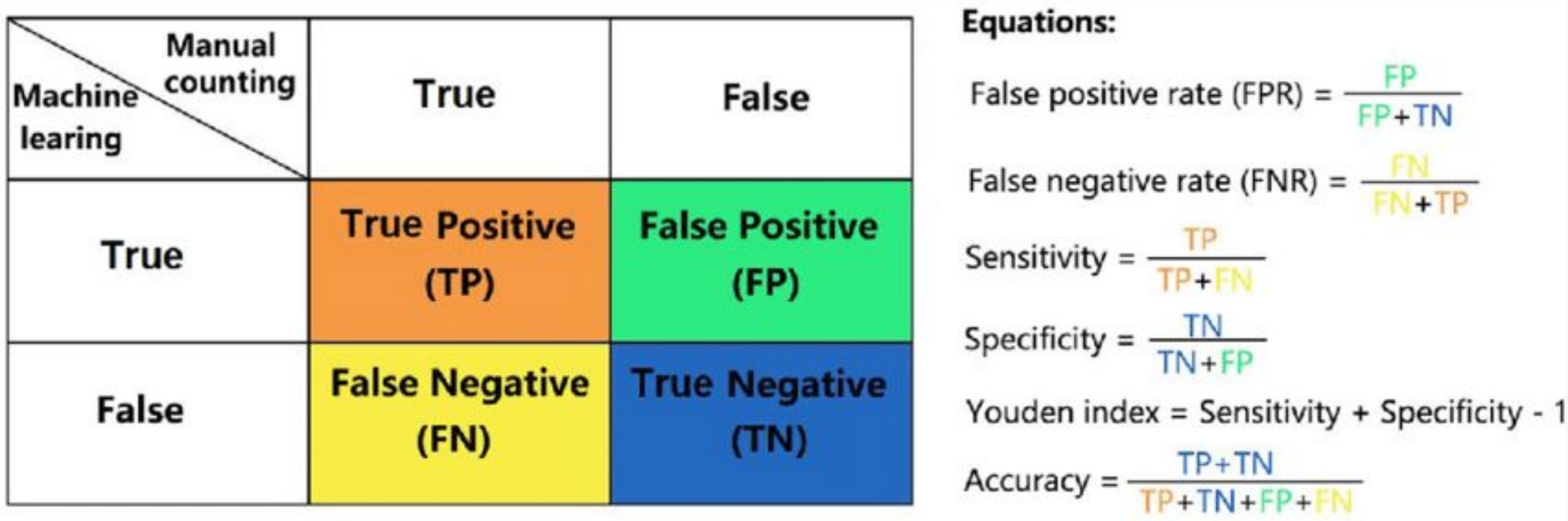
	Actual Values	
Predicted Values	Positive	Negative
Positive	True Positive	False Positive
Negative	False Negative	True Negative

Metrics - Measuring model performance



Going beyond *accuracy*

Accuracy is a limited metric, because models can cheat!!



Manual counting Machine learing	True	False
True	True Positive (TP)	False Positive (FP)
False	False Negative (FN)	True Negative (TN)

Equations:

$$\text{False positive rate (FPR)} = \frac{FP}{FP+TN}$$

$$\text{False negative rate (FNR)} = \frac{FN}{FN+TP}$$

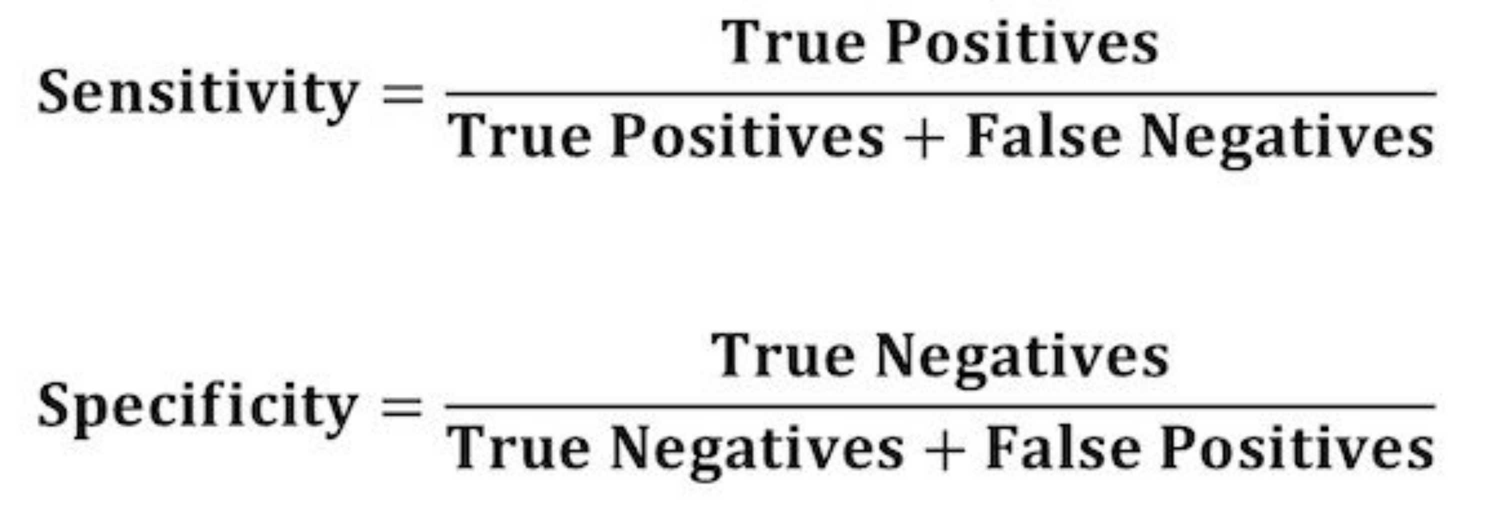
$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

$$\text{Specificity} = \frac{TN}{TN+FP}$$

$$\text{Youden index} = \text{Sensitivity} + \text{Specificity} - 1$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

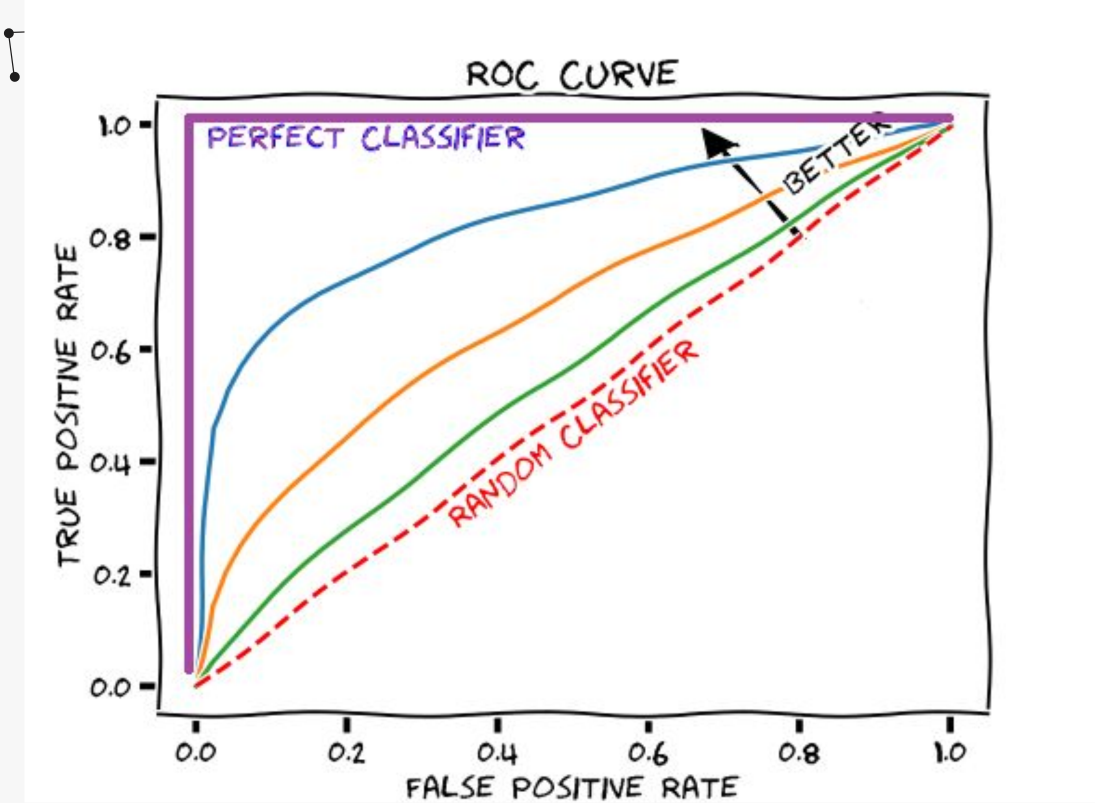
Going beyond *accuracy*



$$\text{Sensitivity} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

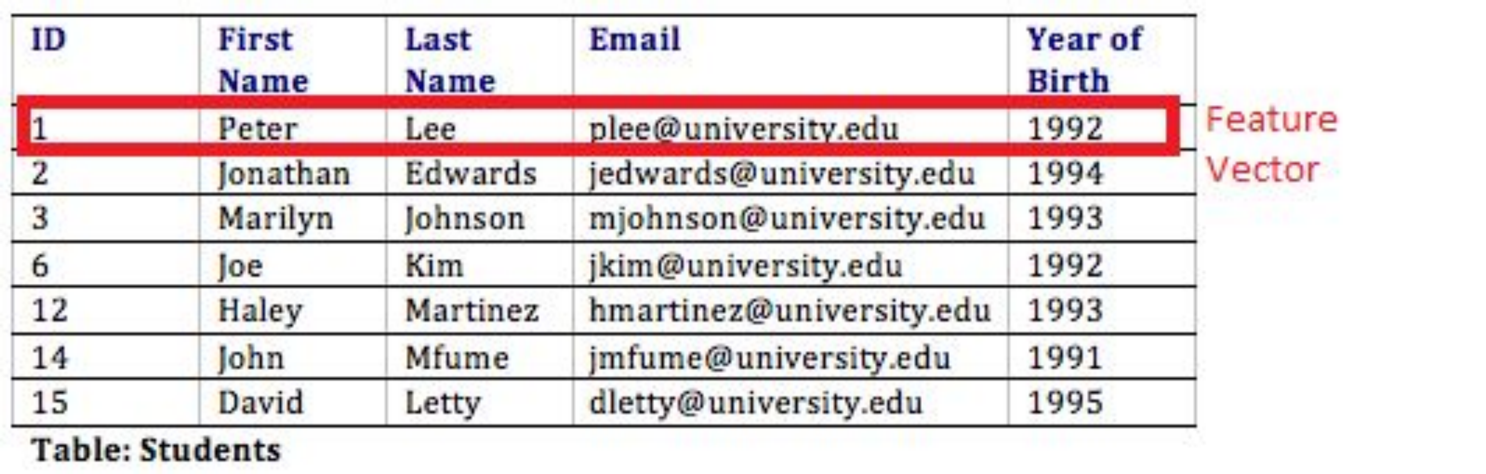
AUROC curve



Receiver operating characteristic curve.

Introduction to Data Science!

How should we store features?



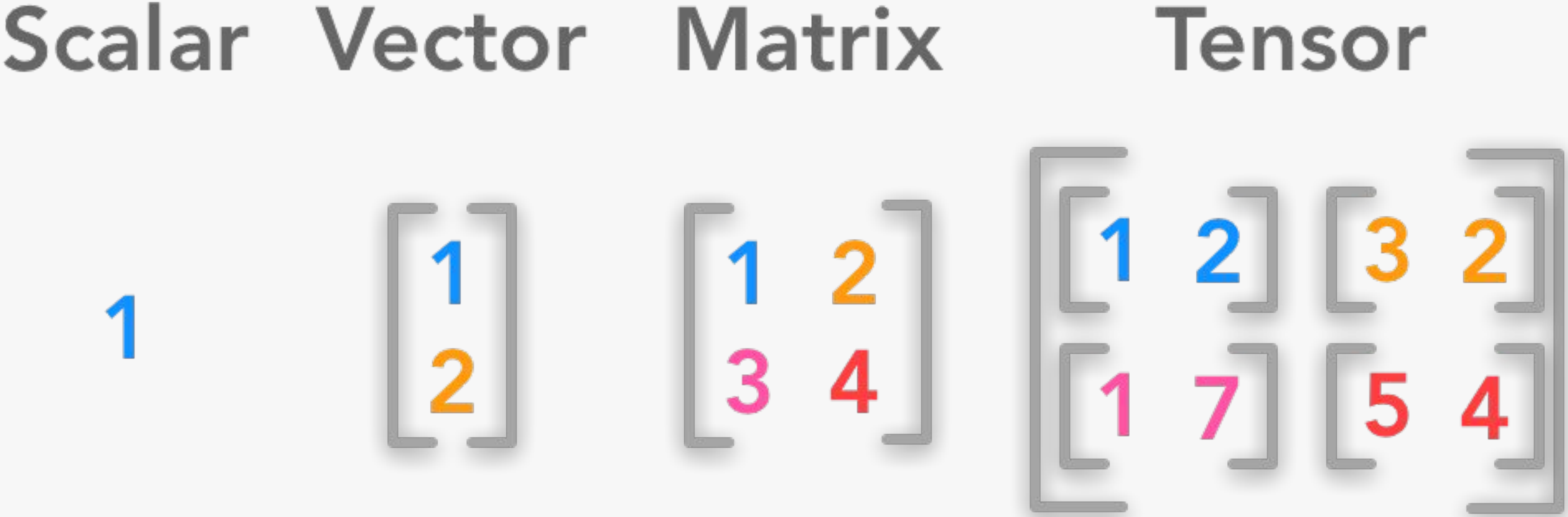
ID	First Name	Last Name	Email	Year of Birth
1	Peter	Lee	plee@university.edu	1992
2	Jonathan	Edwards	jedwards@university.edu	1994
3	Marilyn	Johnson	mjohnson@university.edu	1993
6	Joe	Kim	jkim@university.edu	1992
12	Haley	Martinez	hmartinez@university.edu	1993
14	John	Mfume	jmfume@university.edu	1991
15	David	Letty	dletty@university.edu	1995

Table: Students

Feature Vector

Arrays or vectors!

Vectors and Tensors!

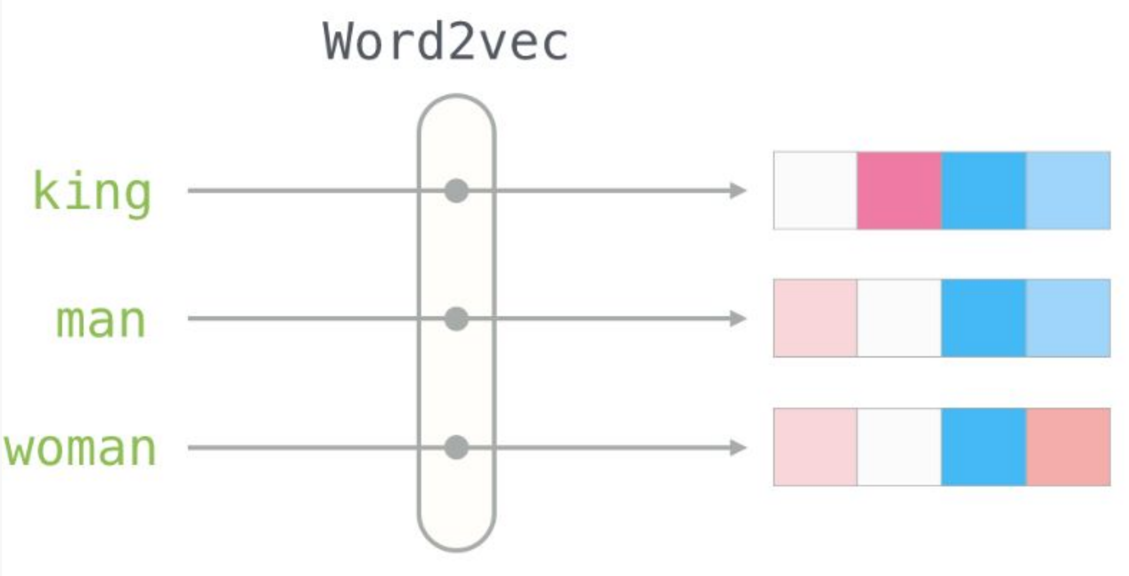


$$1 \quad \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad \begin{bmatrix} \begin{bmatrix} 1 & 2 \end{bmatrix} & \begin{bmatrix} 3 & 2 \end{bmatrix} \\ \begin{bmatrix} 1 & 7 \end{bmatrix} & \begin{bmatrix} 5 & 4 \end{bmatrix} \end{bmatrix}$$

Source: Hadrien Jean’s blog post

Vectors Continued...

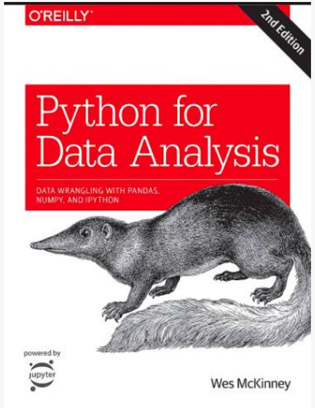
Remember the idea of 'Similarity' ?



Source: Jay Alammar's blog post

Open Source Heroes

Pandas: Wes Mckinney



Python: Guido Van Rossum

Jupyter: Fernando Perez

NumPy: Travis Olyphant