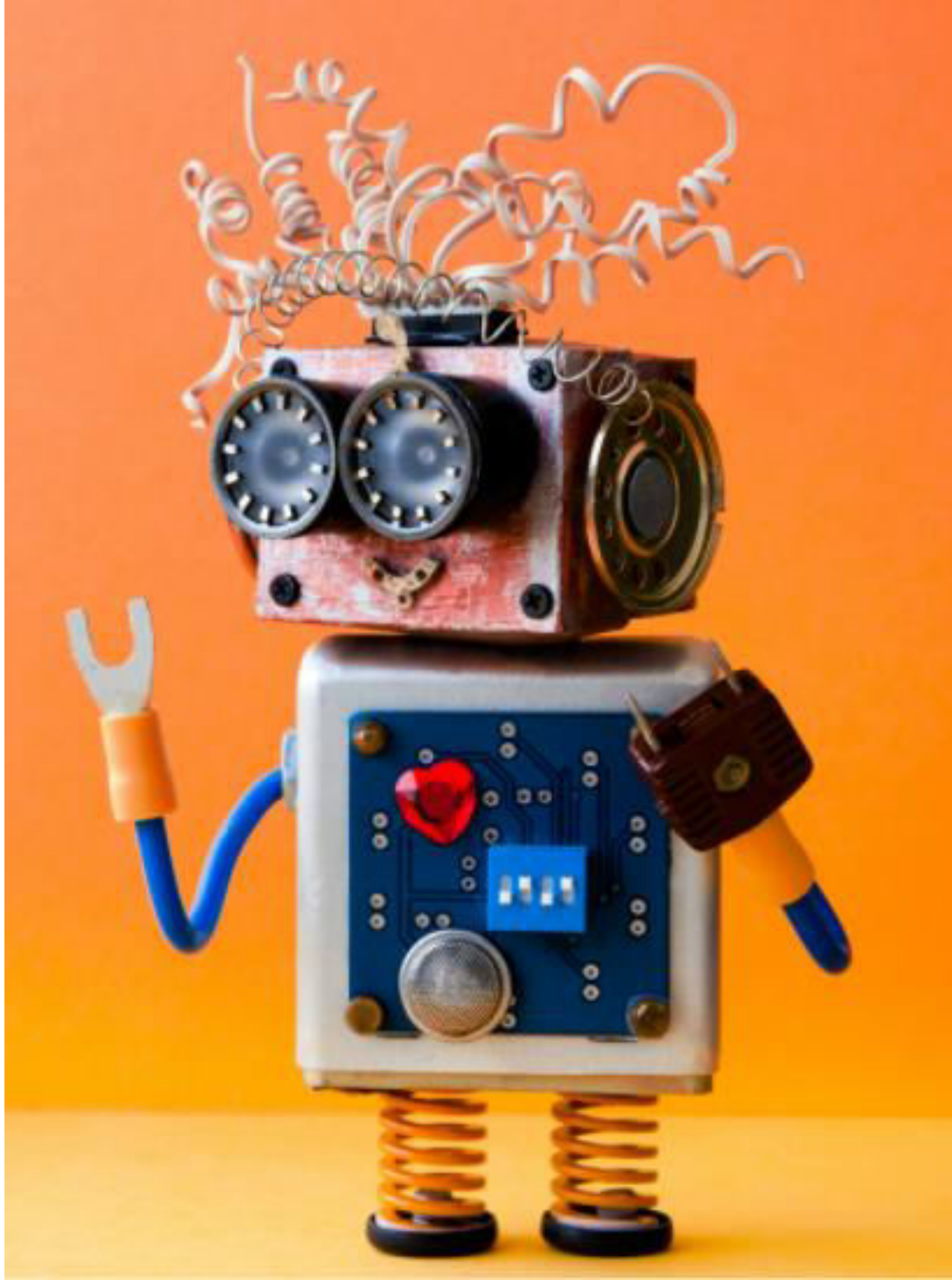
PROJECT OVERVIEW

EASY ADVANCED



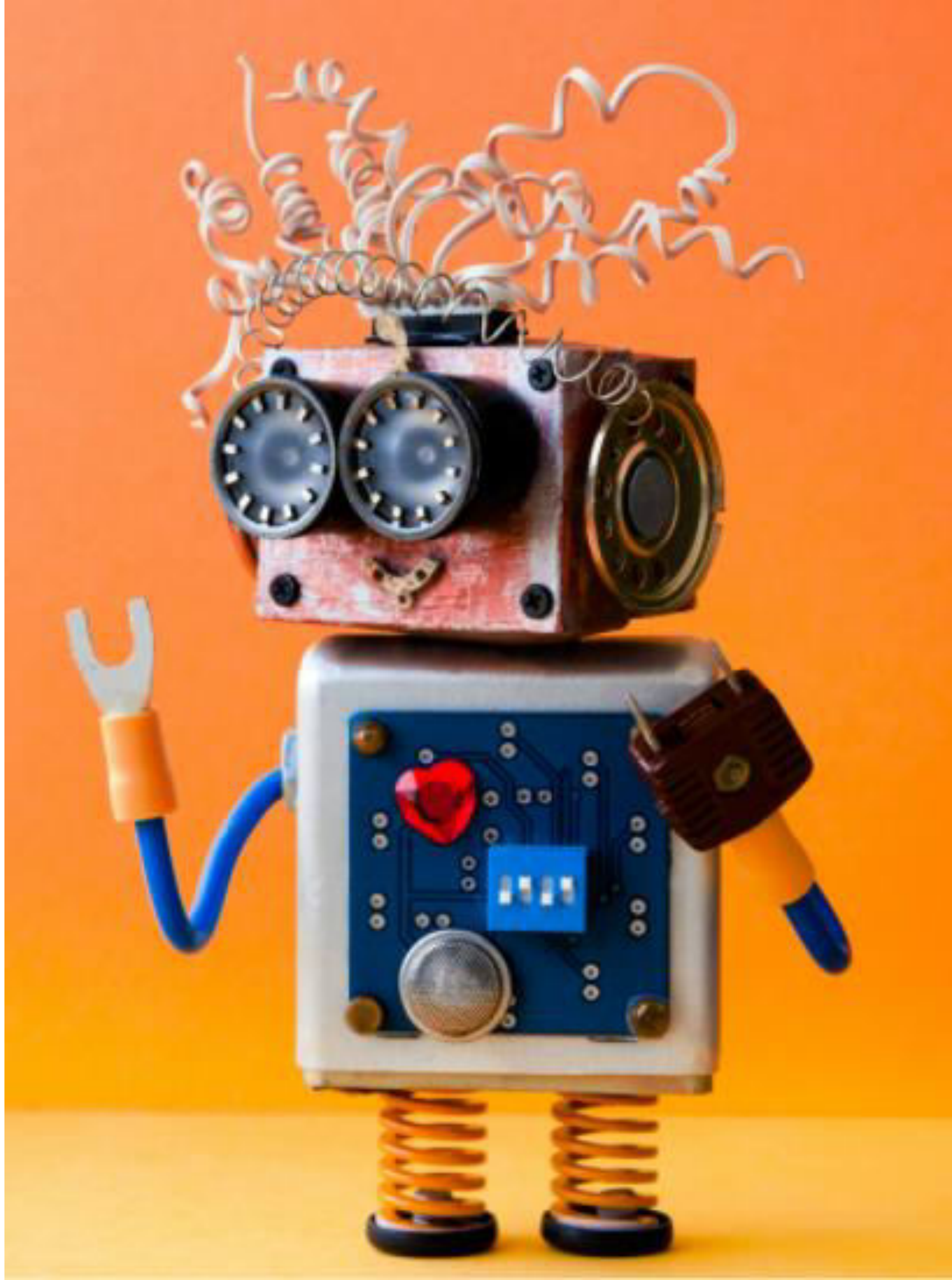
TELECOM CUSTOMERS CHURN PREDICTION

- In this hands-on project, we will train several classification algorithms namely Logistic Regression, Support Vector Machine, K-Nearest Neighbors, and Random Forest Classifier to predict the churn rate of Telecommunication Customers.
- Telecom service providers use customer attrition analysis as one of their key business metrics because the cost of retaining an existing customer is far less than acquiring a new one.
- Machine Learning algorithms help companies analyze customer attrition rate based on several factors which includes various services subscribed by the customers, tenure rate, gender, senior citizen, payment method, etc.

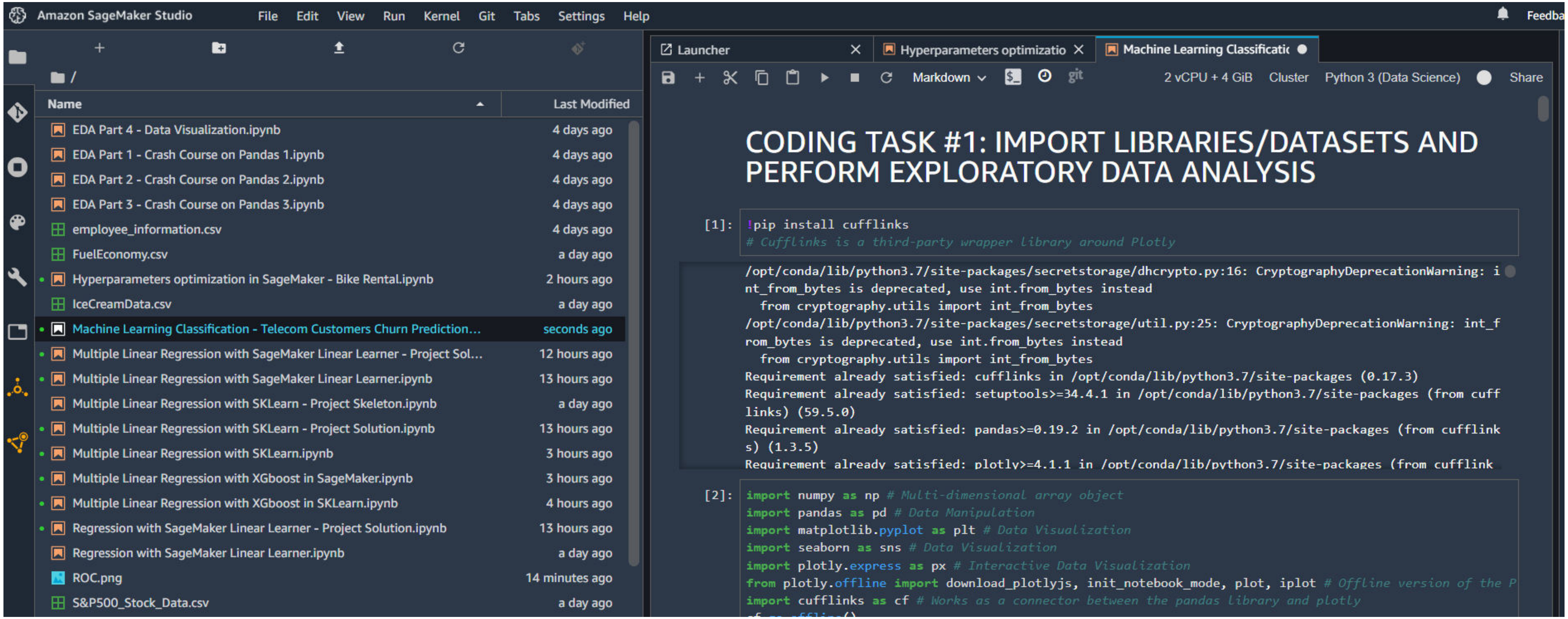
https://pixabay.com/illustrations/family-customer-target-group-ball-563968/
https://www.kaggle.com/blastchar/telco-customer-churn

CODE DEMO

EASY ADVANCED



CODE DEMO

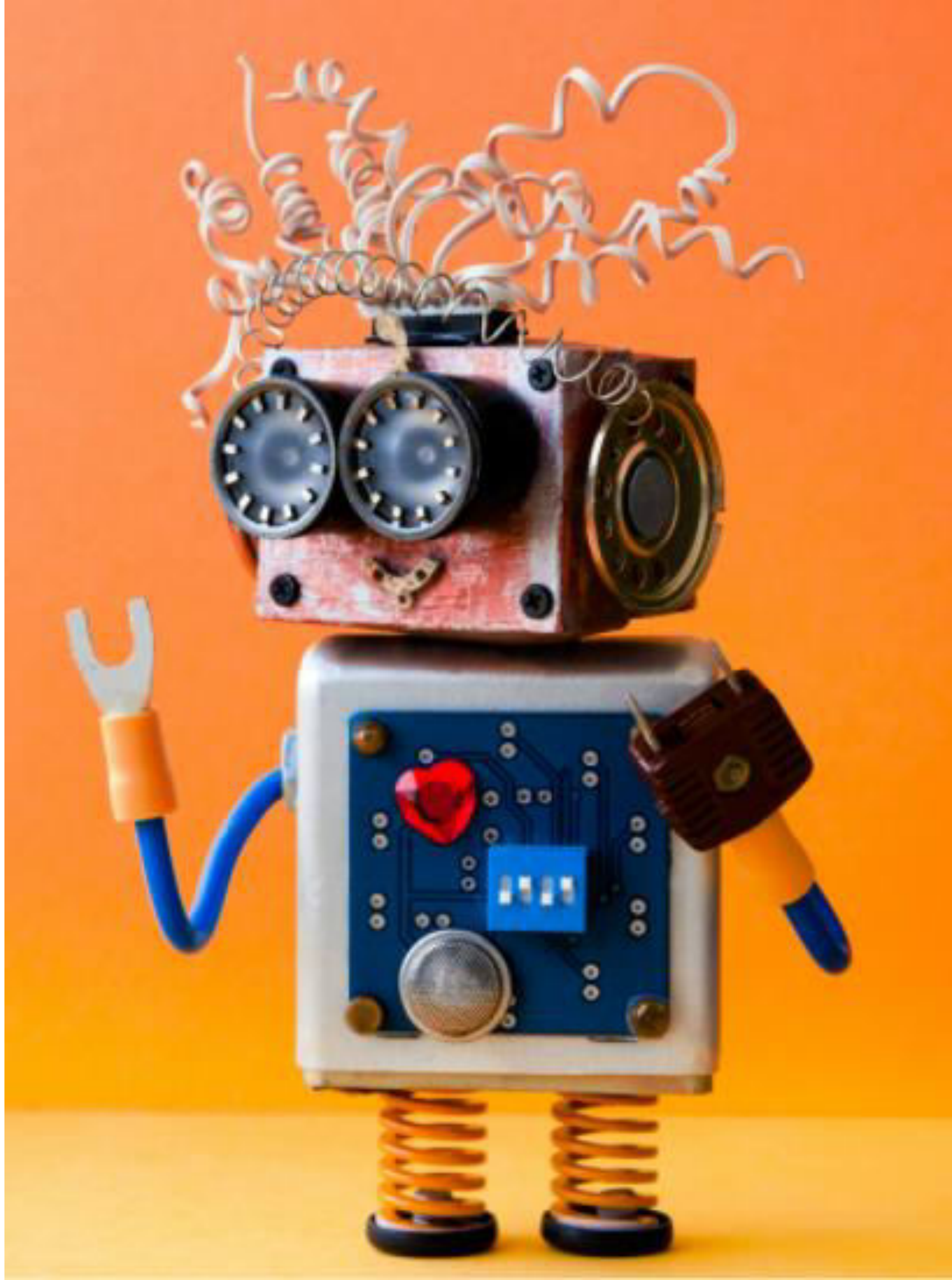


Amazon SageMaker Studio
File
Edit
View
Run
Kernel
Git
Tabs
Settings
Help
Name
Last Modified
EDA Part 4 - Data Visualization.ipynb
4 days ago
EDA Part 1 - Crash Course on Pandas 1.ipynb
4 days ago
EDA Part 2 - Crash Course on Pandas 2.ipynb
4 days ago
EDA Part 3 - Crash Course on Pandas 3.ipynb
4 days ago
employee_information.csv
4 days ago
FuelEconomy.csv
a day ago
Hyperparameters optimization in SageMaker - Bike Rental.ipynb
2 hours ago
IceCreamData.csv
a day ago
Machine Learning Classification - Telecom Customers Churn Prediction...
seconds ago
Multiple Linear Regression with SageMaker Linear Learner - Project Sol...
12 hours ago
Multiple Linear Regression with SageMaker Linear Learner.ipynb
13 hours ago
Multiple Linear Regression with SKLearn - Project Skeleton.ipynb
a day ago
Multiple Linear Regression with SKLearn - Project Solution.ipynb
13 hours ago
Multiple Linear Regression with SKLearn.ipynb
3 hours ago
Multiple Linear Regression with XGboost in SageMaker.ipynb
3 hours ago
Multiple Linear Regression with XGboost in SKLearn.ipynb
4 hours ago
Regression with SageMaker Linear Learner - Project Solution.ipynb
13 hours ago
Regression with SageMaker Linear Learner.ipynb
a day ago
ROC.png
14 minutes ago
S&P500_Stock_Data.csv
a day ago
Launcher
Hyperparameters optimizatio
Machine Learning Classificatic
Markdown
2 vCPU + 4 GiB
Cluster
Python 3 (Data Science)
Share
CODING TASK #1: IMPORT LIBRARIES/DATASETS AND PERFORM EXPLORATORY DATA ANALYSIS
[1]:
!pip install cufflinks
Cufflinks is a third-party wrapper library around Plotly
/opt/conda/lib/python3.7/site-packages/secretstorage/dhcrypto.py:16: CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes instead
from cryptography.utils import int_from_bytes
/opt/conda/lib/python3.7/site-packages/secretstorage/util.py:25: CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes instead
from cryptography.utils import int_from_bytes
Requirement already satisfied: cufflinks in /opt/conda/lib/python3.7/site-packages (0.17.3)
Requirement already satisfied: setuptools>=34.4.1 in /opt/conda/lib/python3.7/site-packages (from cufflinks) (59.5.0)
Requirement already satisfied: pandas>=0.19.2 in /opt/conda/lib/python3.7/site-packages (from cufflinks) (1.3.5)
Requirement already satisfied: plotly>=4.1.1 in /opt/conda/lib/python3.7/site-packages (from cufflink
[2]:
import numpy as np # Multi-dimensional array object
import pandas as pd # Data Manipulation
import matplotlib.pyplot as plt # Data Visualization
import seaborn as sns # Data Visualization
import plotly.express as px # Interactive Data Visualization
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot # Offline version of the P
import cufflinks as cf # Works as a connector between the pandas library and plotly

LOGISTIC REGRESSION CLASSIFIER MODEL

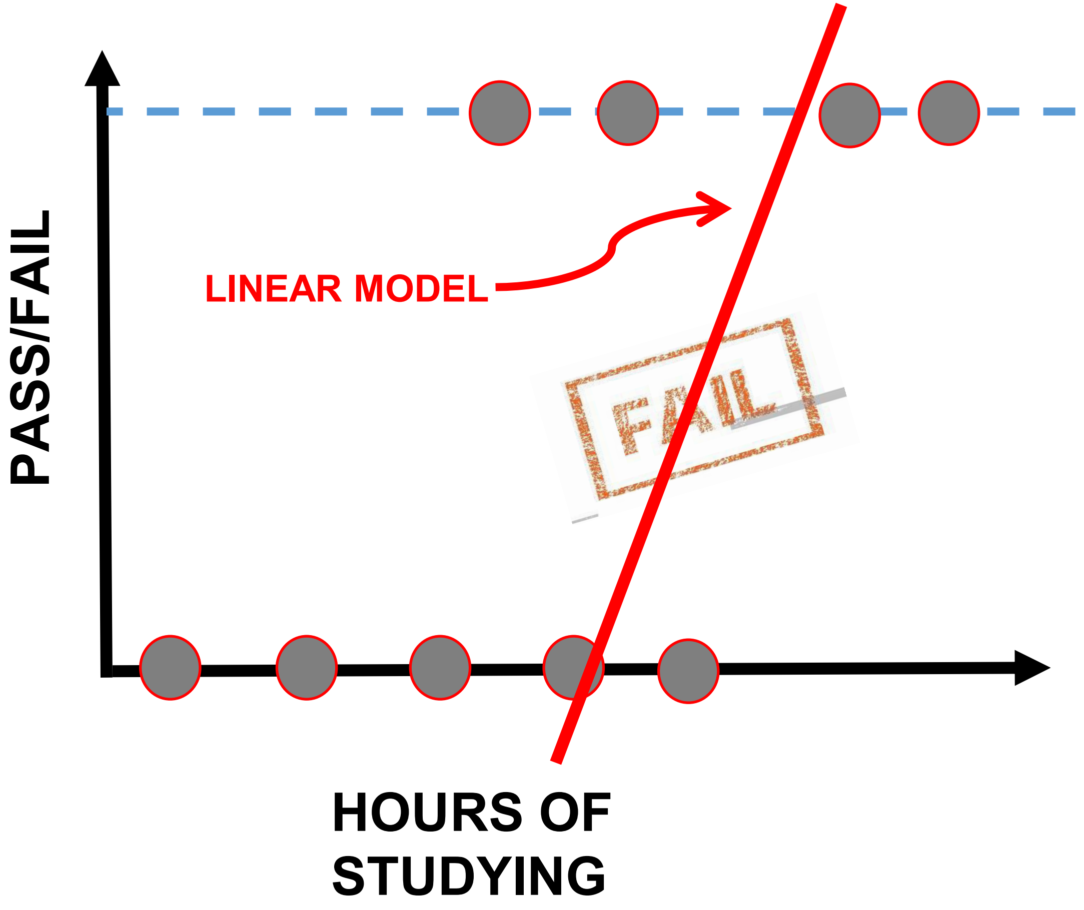
EASY

ADVANCED



LOGISTIC REGRESSION: INTUITION

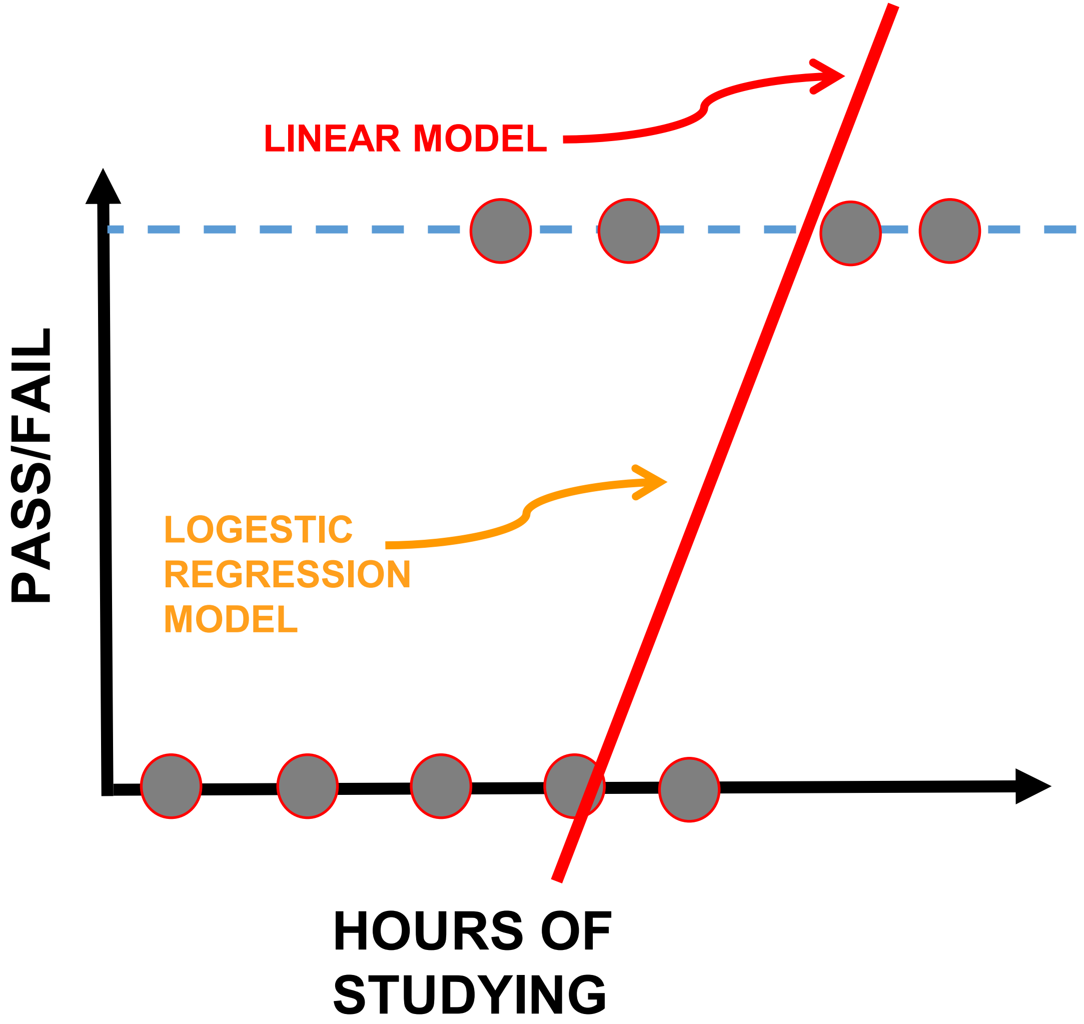
- Linear Regression is used to predict outputs on a continuous spectrum.
- Example: Predicting revenue based on the outside air temperature.
- Logistic Regression is used to predict binary outputs with 2 possible values (0 or 1).
- Example: Logistic model output can be one of two classes: pass/fail, win/lose, healthy/sick



Hours Studying	Pass/Fail
1	0
1.5	0
2	0
3	1
3.25	0
4	1
5	1

LOGISTIC REGRESSION: MATH

- Linear Regression is not suitable for classification problem.
- Linear Regression is unbounded, so Logistic Regression will be a better candidate in which the output value ranges from 0 to 1.



Linear Equation:

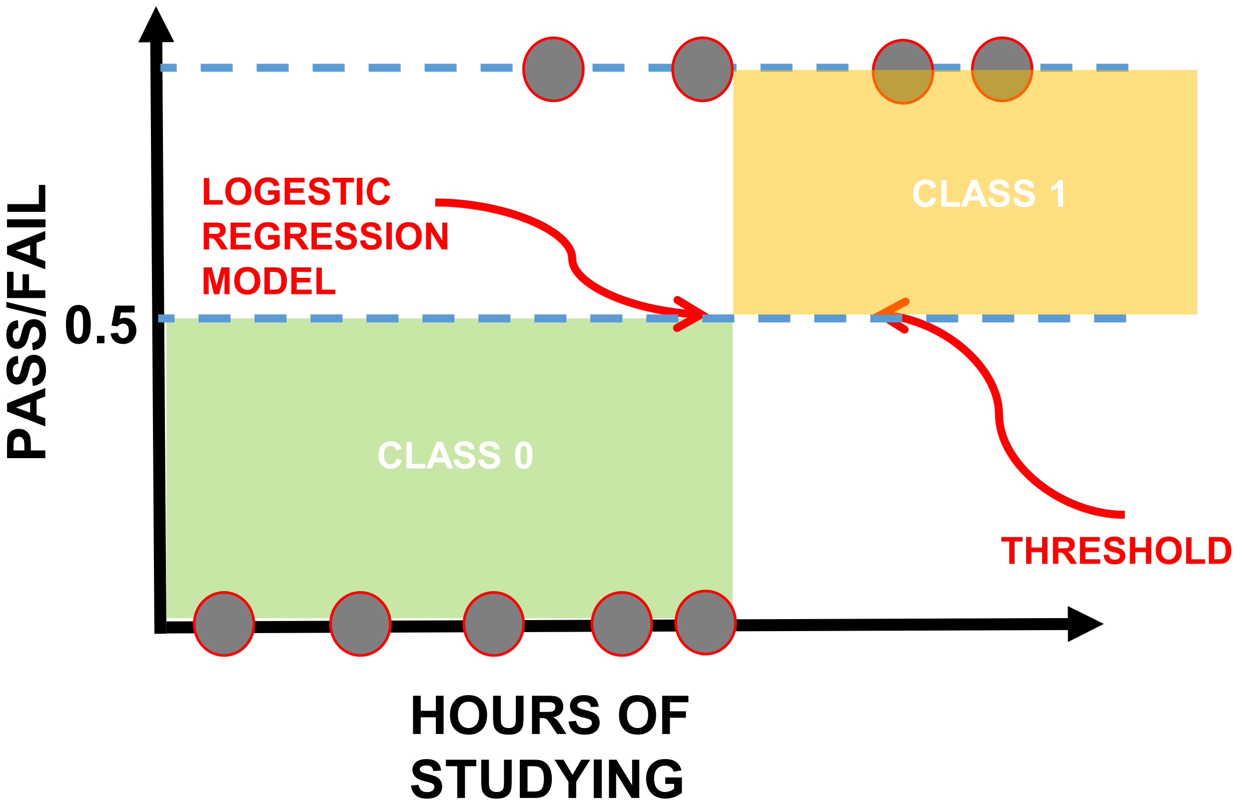
- $y = b_o + b_1 * x$

Apply Sigmoid Function:

- $P(x) = \text{sigmoid}(y)$
- $P(x) = 1/1+e^{-y}$
- $P(x) = 1/1+e^{-(b_0 + b_1 * x)}$

LOGISTIC REGRESSION: FROM PROBABILITY TO CLASS

- Now we need to convert from a probability to a class value which is “0” or “1”



Linear Equation:

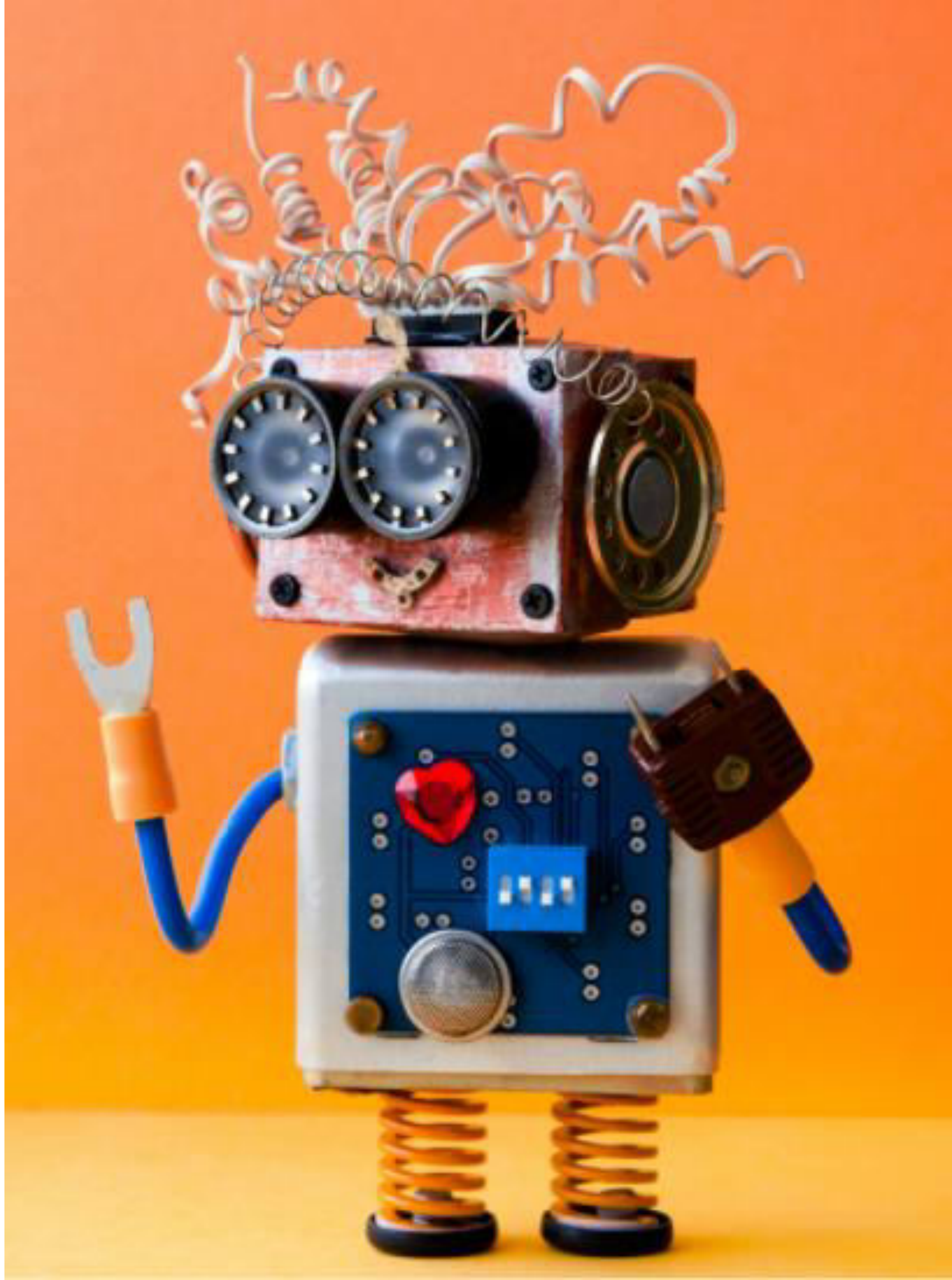
- $y = b_o + b_1 * x$

Apply Sigmoid Function:

- $P(x) = \text{sigmoid}(y)$
- $P(x) = 1/1+e^{-y}$
- $P(x) = 1/1+e^{-(b_0 + b_1 * x)}$

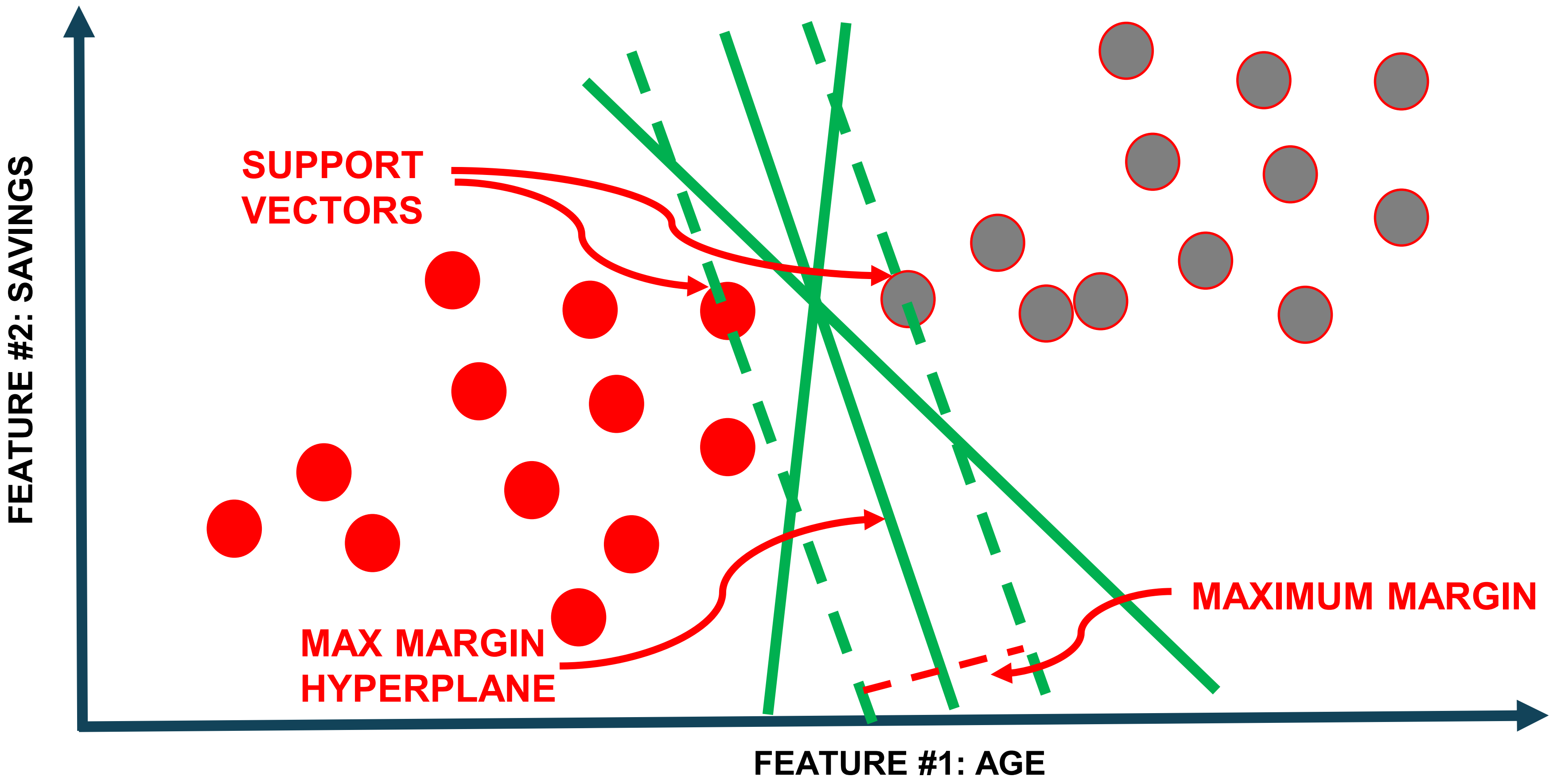
SUPPORT VECTOR MACHINES (SVM)

EASY ADVANCED

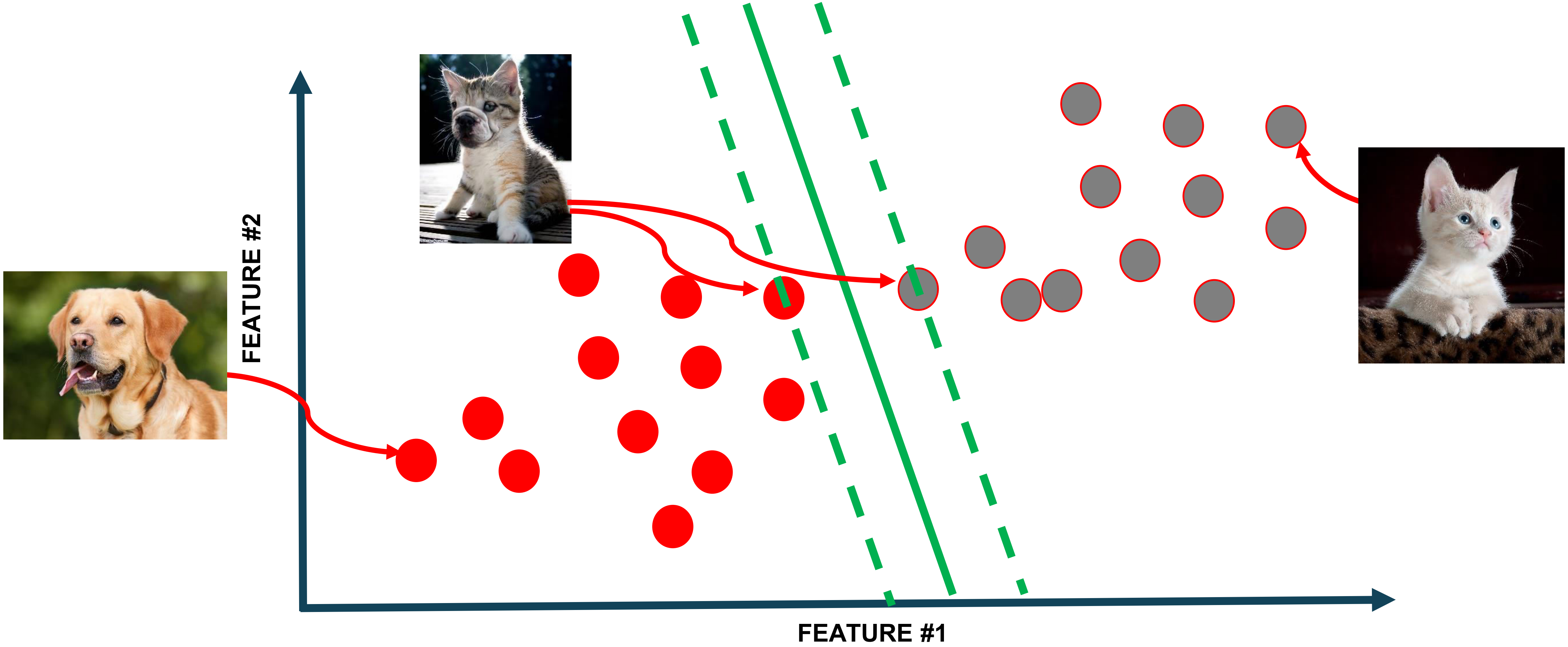


SUPPORT VECTOR MACHINES: INTUITION

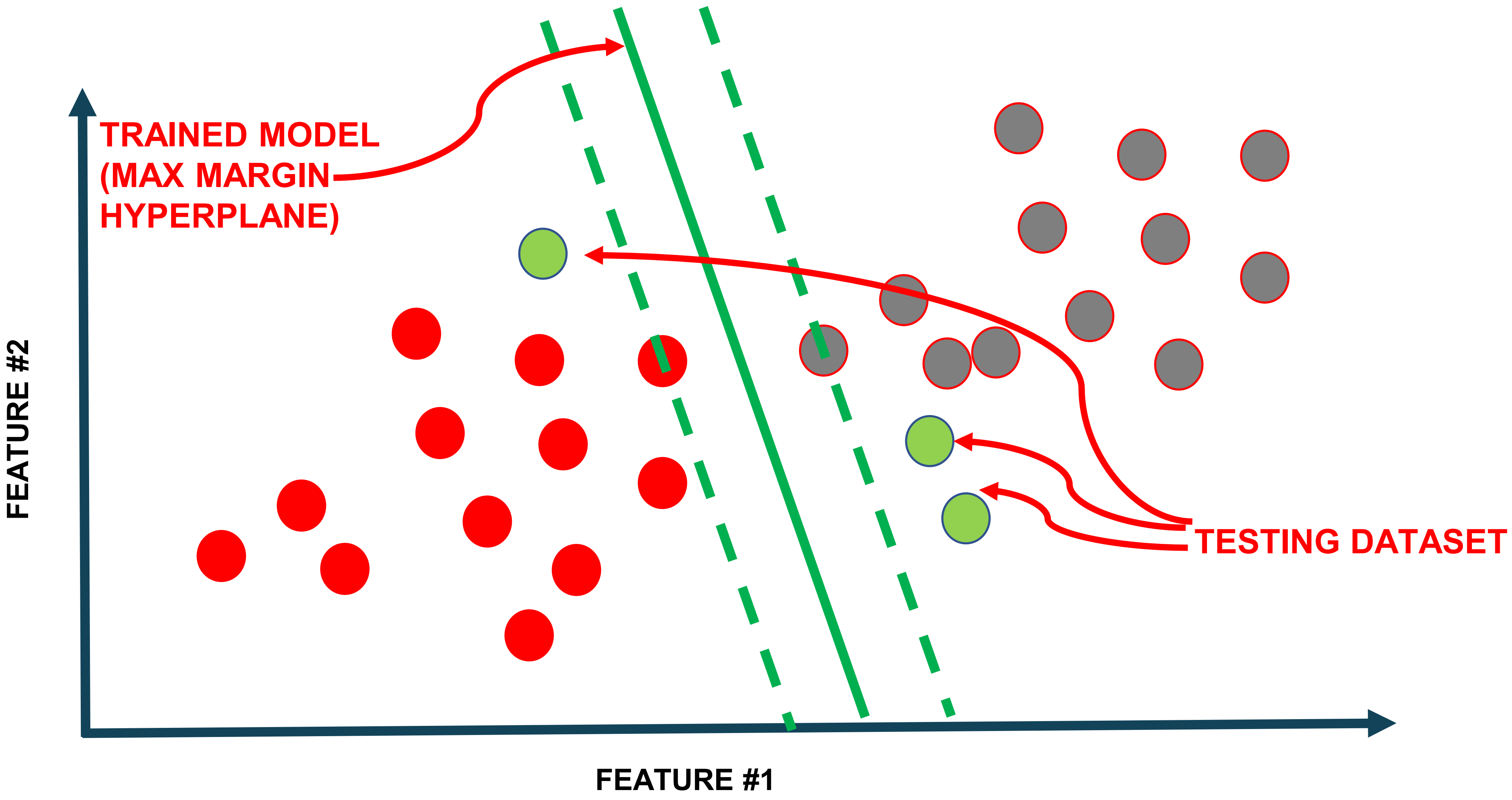
- Assume that you are data scientist working at a major bank in NYC.
- You want to classify a new client as eligible to retire or not, customer features are: Age and Savings.



SUPPORT VECTOR MACHINES: INTUITION

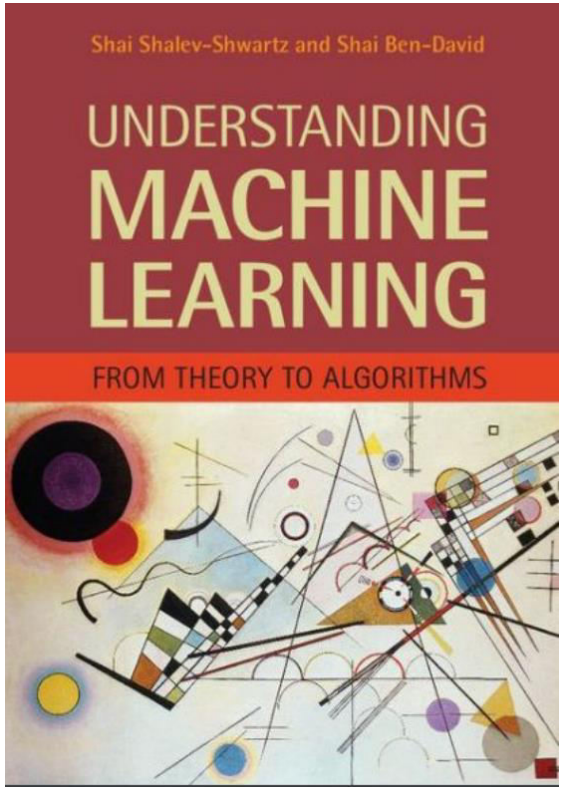


SUPPORT VECTOR MACHINES: MODEL EVALUATION

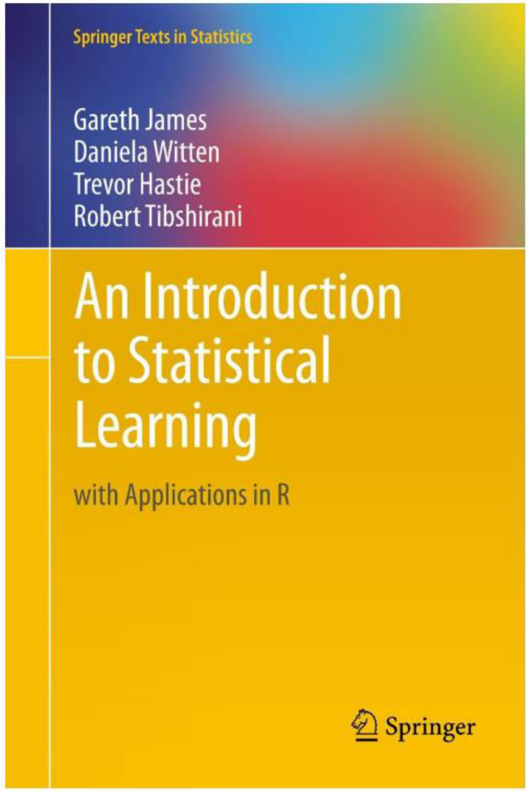


SUPPORT VECTOR MACHINES: ADDITIONAL READING MATERIAL

- Additional Resources, Page #202: http://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/understanding-machine-learning-theory-algorithms.pdf



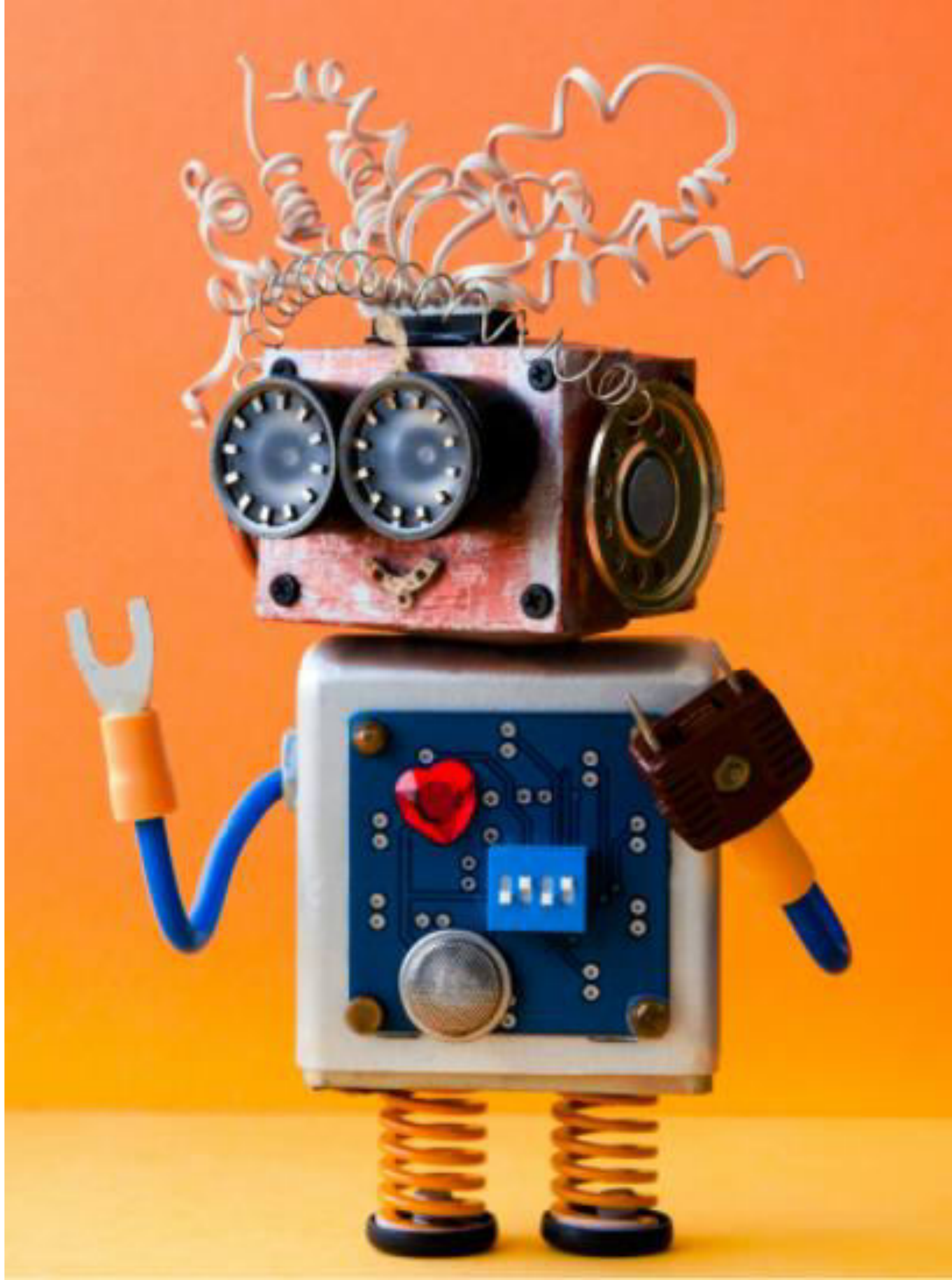
- Additional Resources, Page #337:
- http://www-bcf.usc.edu/~gareth/ISL/ISLR%20Seventh%20Printing.pdf



RANDOM FOREST CLASSIFIER MODELS

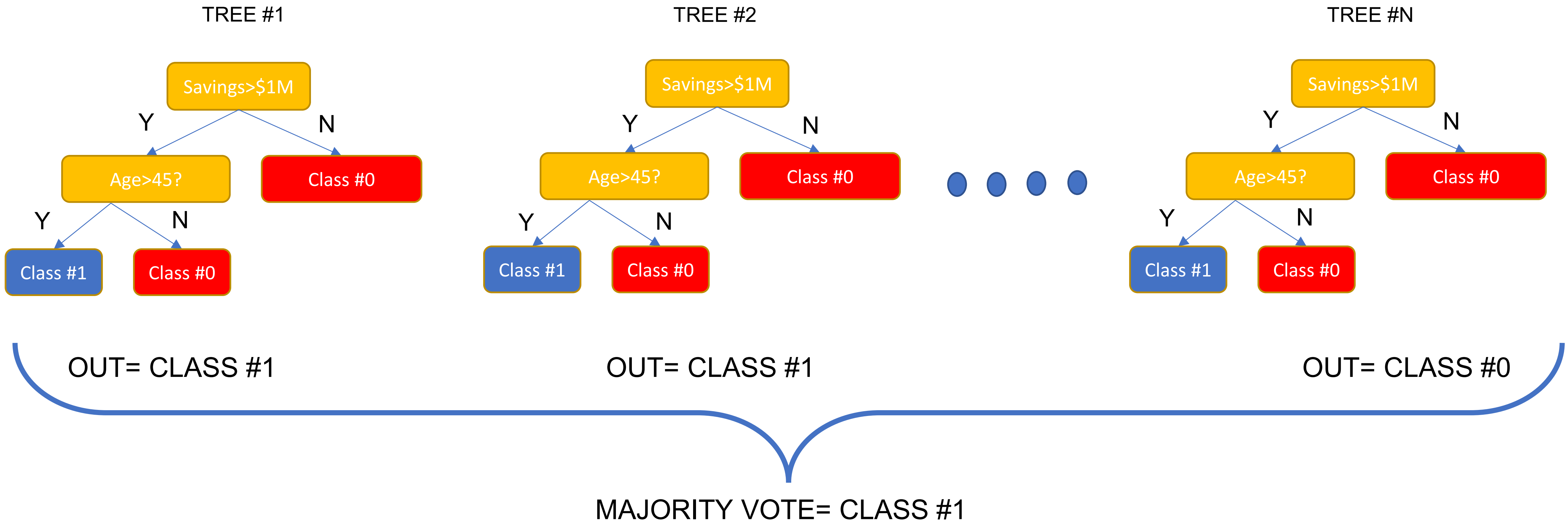
EASY

ADVANCED



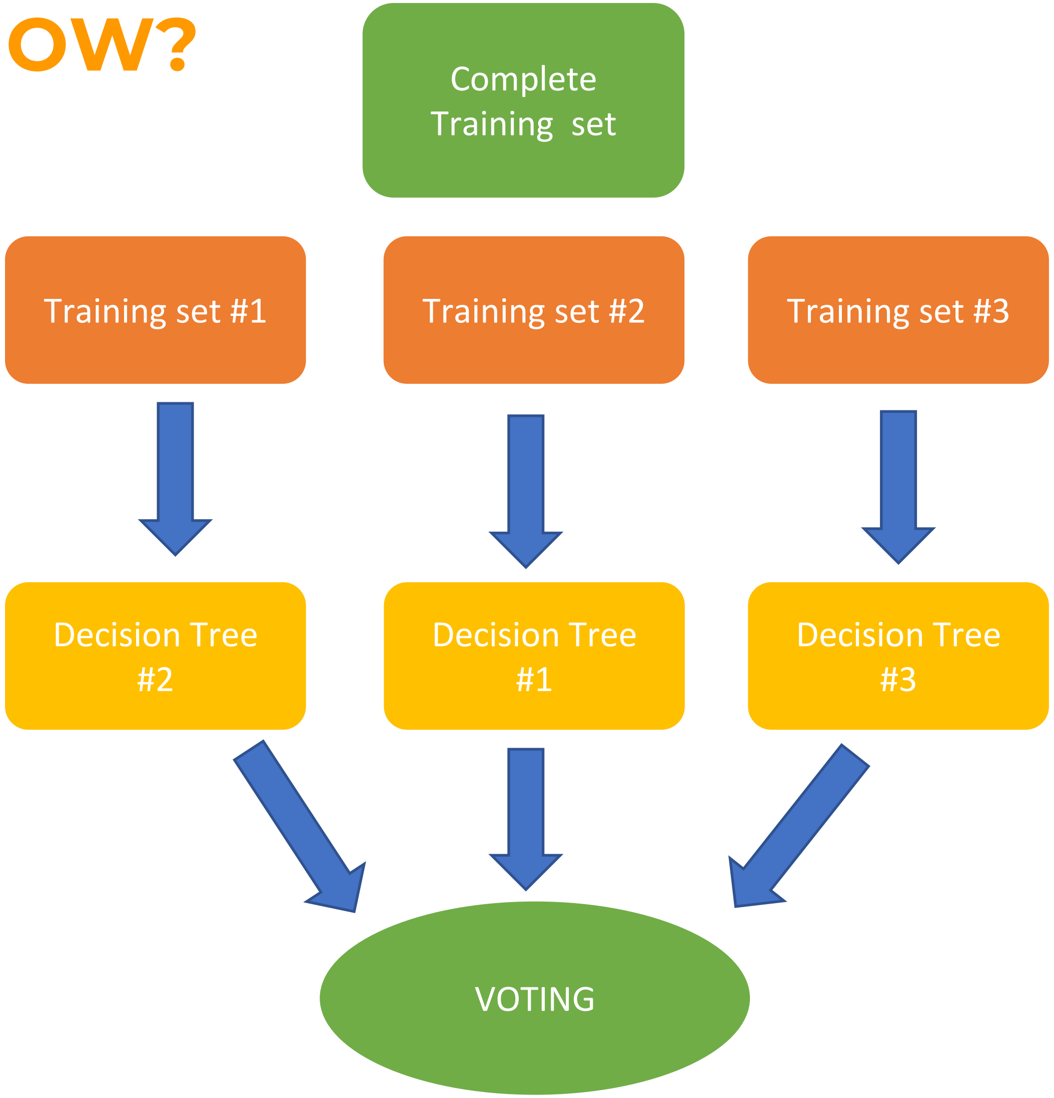
RANDOM FOREST CLASSIFIER: INTUITION

- Random Forest classifier is a type of ensemble algorithm
- It creates a set of decision trees from randomly selected subset of training set
- It then combines votes from different decision trees to decide the final class of the test object.



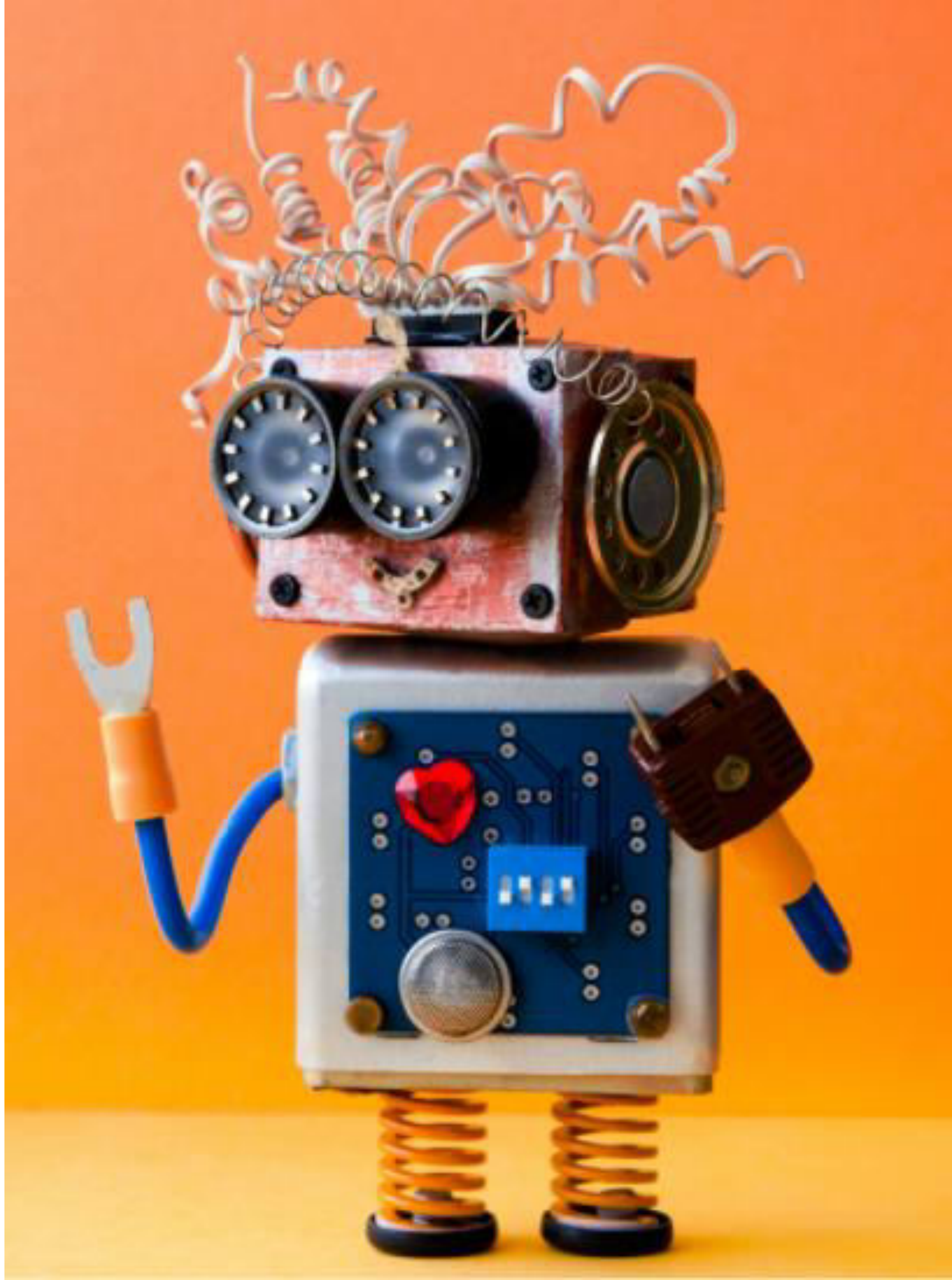
RANDOM FOREST: WHY AND HOW?

- It overcomes the issues with single decision trees by reducing the effect of noise
- Overcomes overfitting problem by taking average of all the predictions, canceling out biases
- Suppose training set: [X1, X2, X3, X4] with labels: [L1, L2, L3, L4]
- Random Forest creates three decision trees taking inputs as follows: [X1, X2, X3], [X1, X2, X4], [X2, X3, X4]
- Example: Combining votes from a pool of expert, each will bring their own experience and background to solve the problem resulting in a better outcome.
- Runs effectively on large database
- For large data, it produces highly accurate predictions



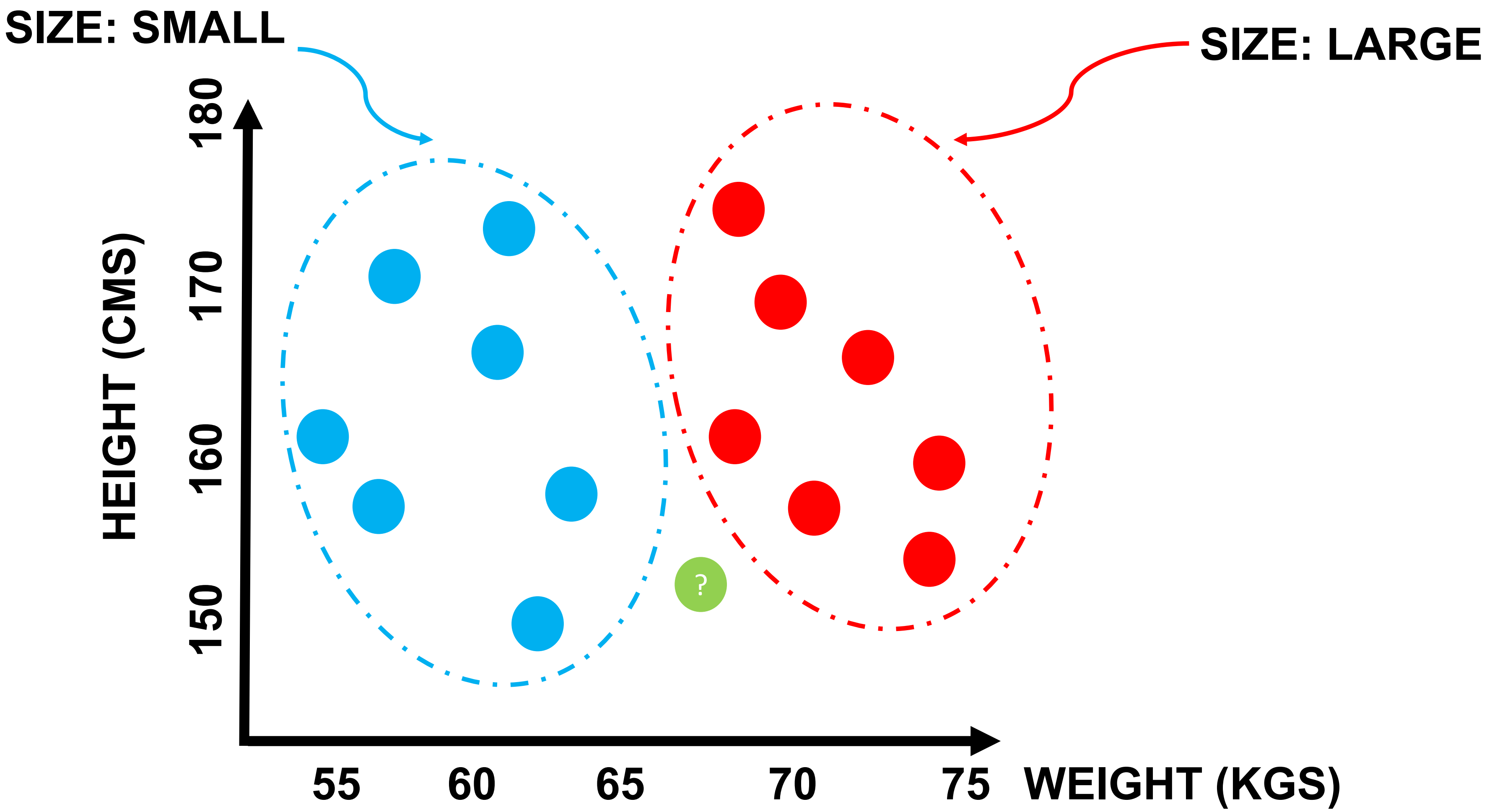
K-NEAREST NEIGHBOUR (KNN)

EASY ADVANCED



K NEAREST NEIGHBORS (KNN): INTUITION

- K-Nearest Neighbors (KNN) algorithm is a classification algorithm
- KNN works by finding the most similar data points in the training data, and attempt to make an educated guess based on their classifications



K NEARSET NEIGHBORS (KNN): ALGORITHM STEP

Select a value for k (e.g.: 1, 2, 3, …..)

Calculate the Euclidian distance between the point to be classified and every other point in the training data set

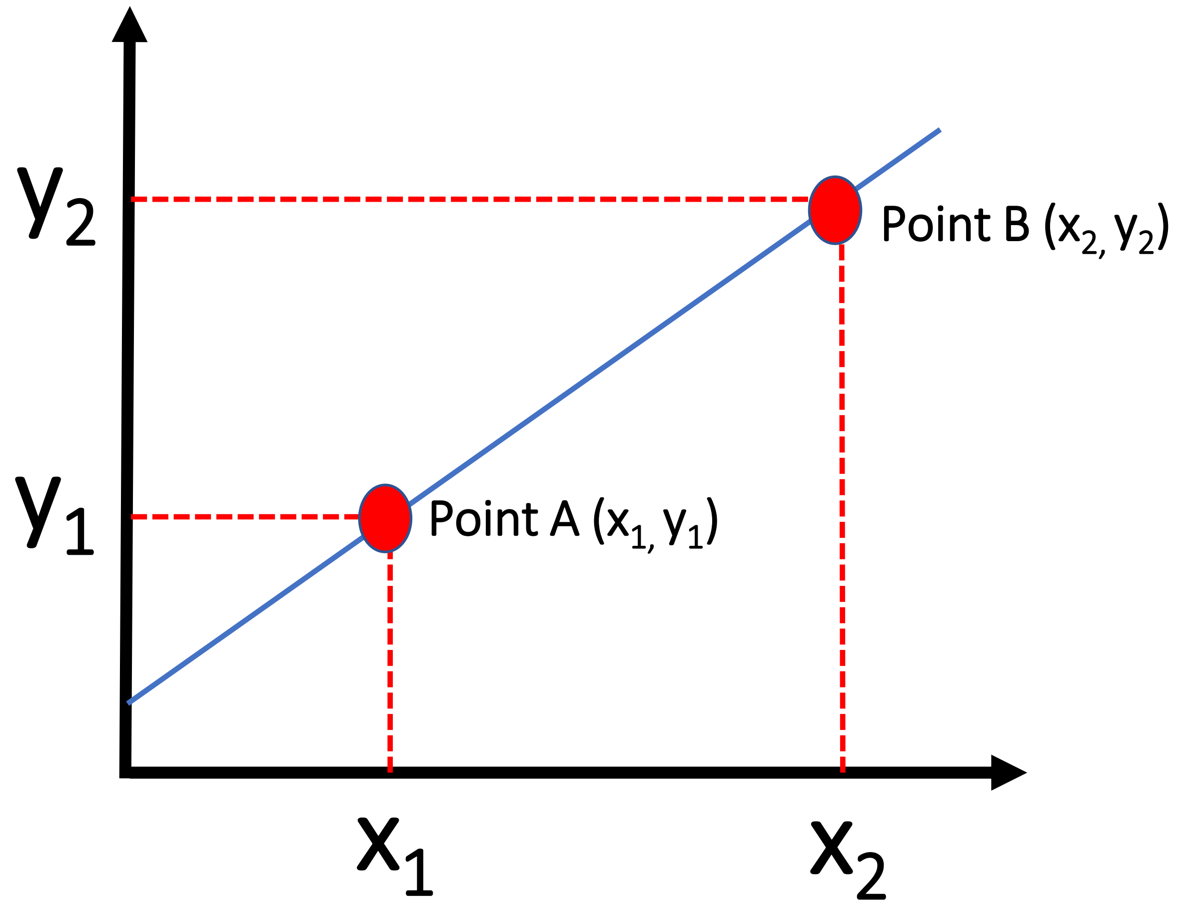
Pick the k closest data points (points with the k smallest distances)

Run a majority vote among selected data points, the dominating classification is the winner. Point is classified based on the dominant class

Repeat

EUCILIDEAN DISTANCE: INTUITION

- Euclidean Distance= $\sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}$



K NEARSET NEIGHBORS (KNN): EXAMPLE

- KNN will look for the 5 data points that are closest to the new customer data point
- The algorithm will determine which category (class) are these 5 points in
- Since 4 points had class "SMALL" and 1 had "LARGE", then new customer shall be assigned small size

Height	Weight	T-Shirt Size	Euclidian Dist	Vote
158	58	S	4.242640687	
158	59	S	3.605551275	
158	63	S	3.605551275	
160	59	S	2.236067977	3
160	60	S	1.414213562	1
163	60	S	2.236067977	3
163	61	S	2	2
160	64	L	3.16227766	5
163	64	L	4	
165	61	L	4.123105626	
165	62	L	5.656858249	

New Customer Information:

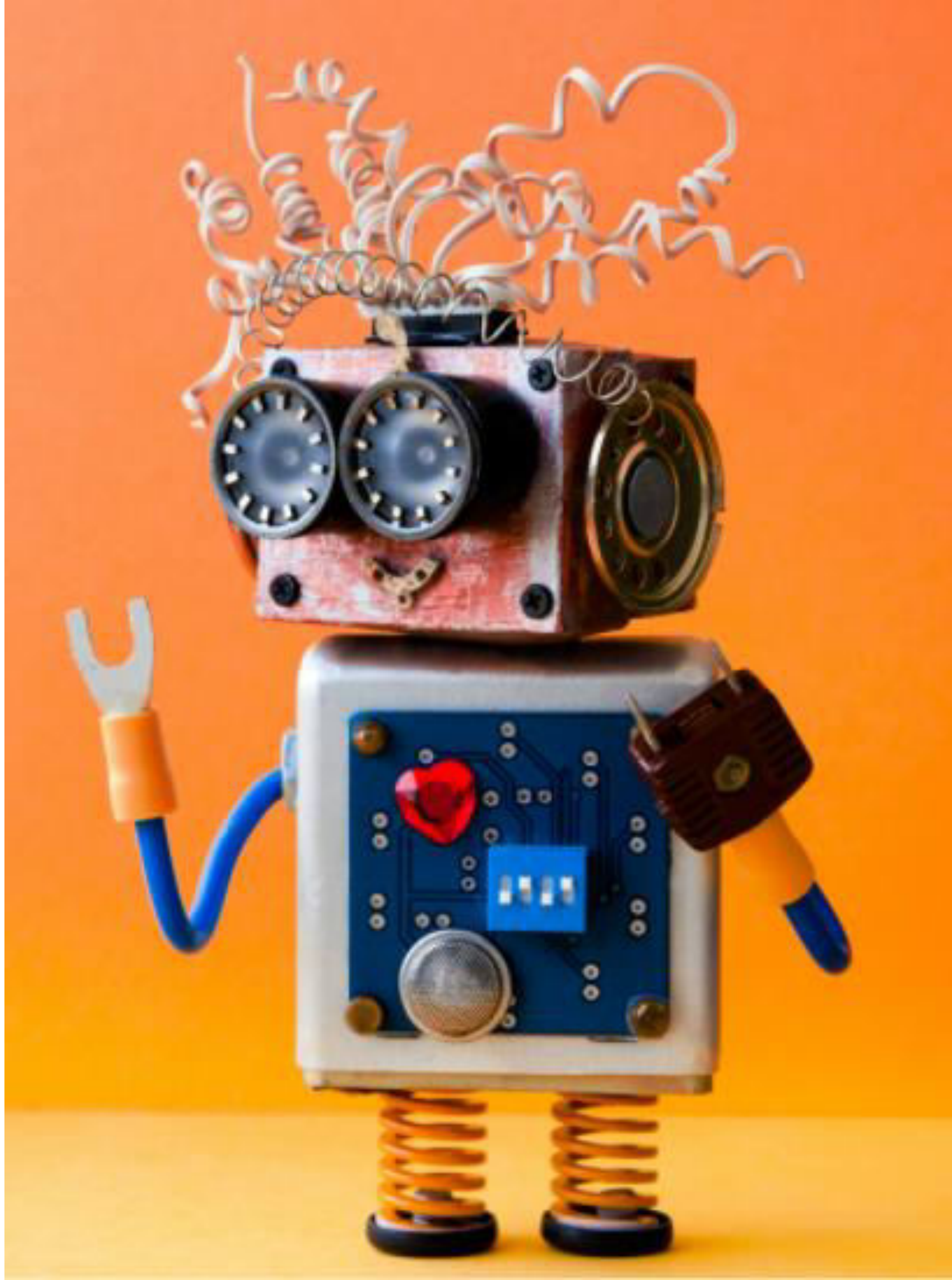
Height: 161
Weight: 61

Assume, k= 5

Example Source: https://www.listendata.com/2017/12/k-nearest-neighbor-step-by-step-tutorial.html

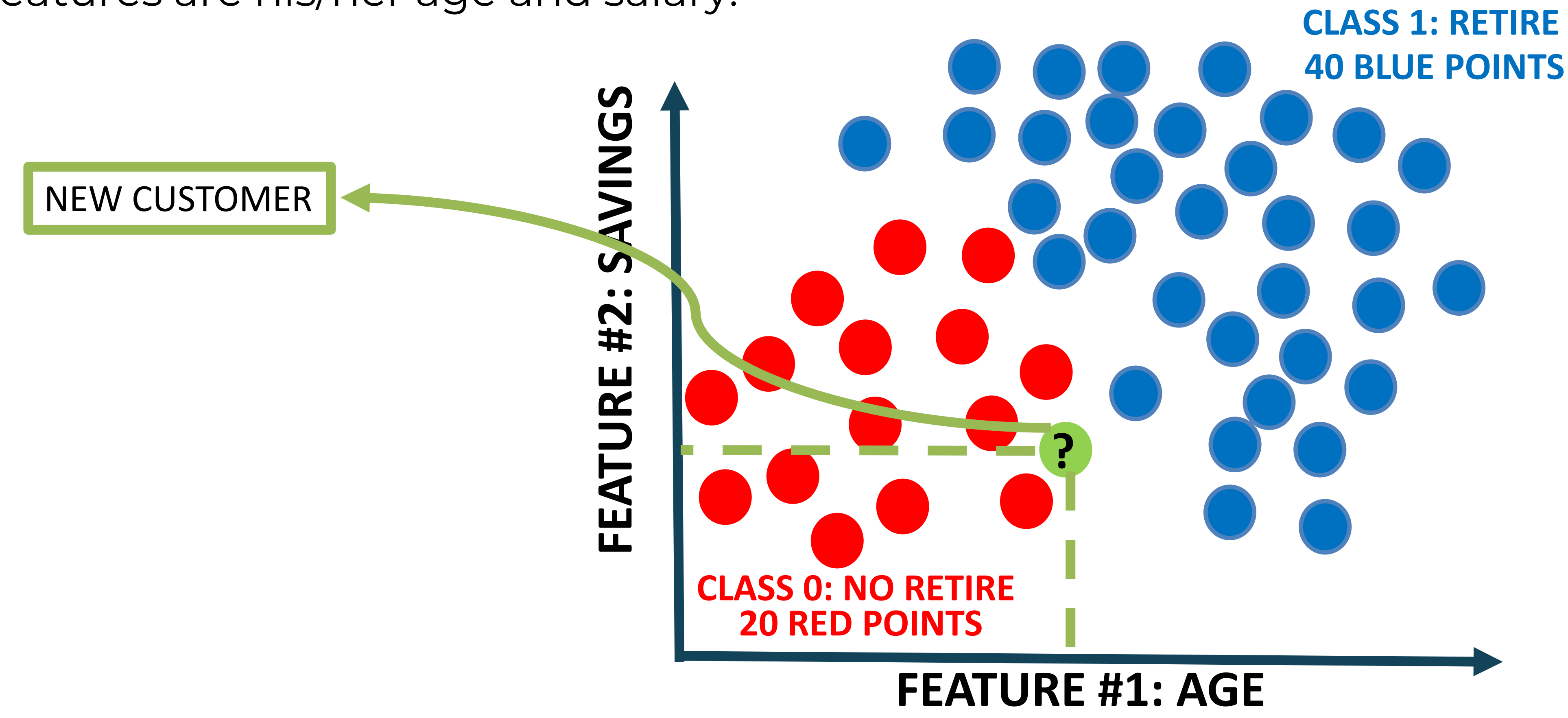
NAÏVE BAYES CLASSIFIER MODEL

EASY ADVANCED



NAÏVE BAYES: INTUITION

- Naïve Bayes is a classification technique based on Bayes' Theorem.
- Let's assume that you are data scientist working major bank in NYC and you want to classify a new client as eligible to retire or not.
- Customer features are his/her age and salary.

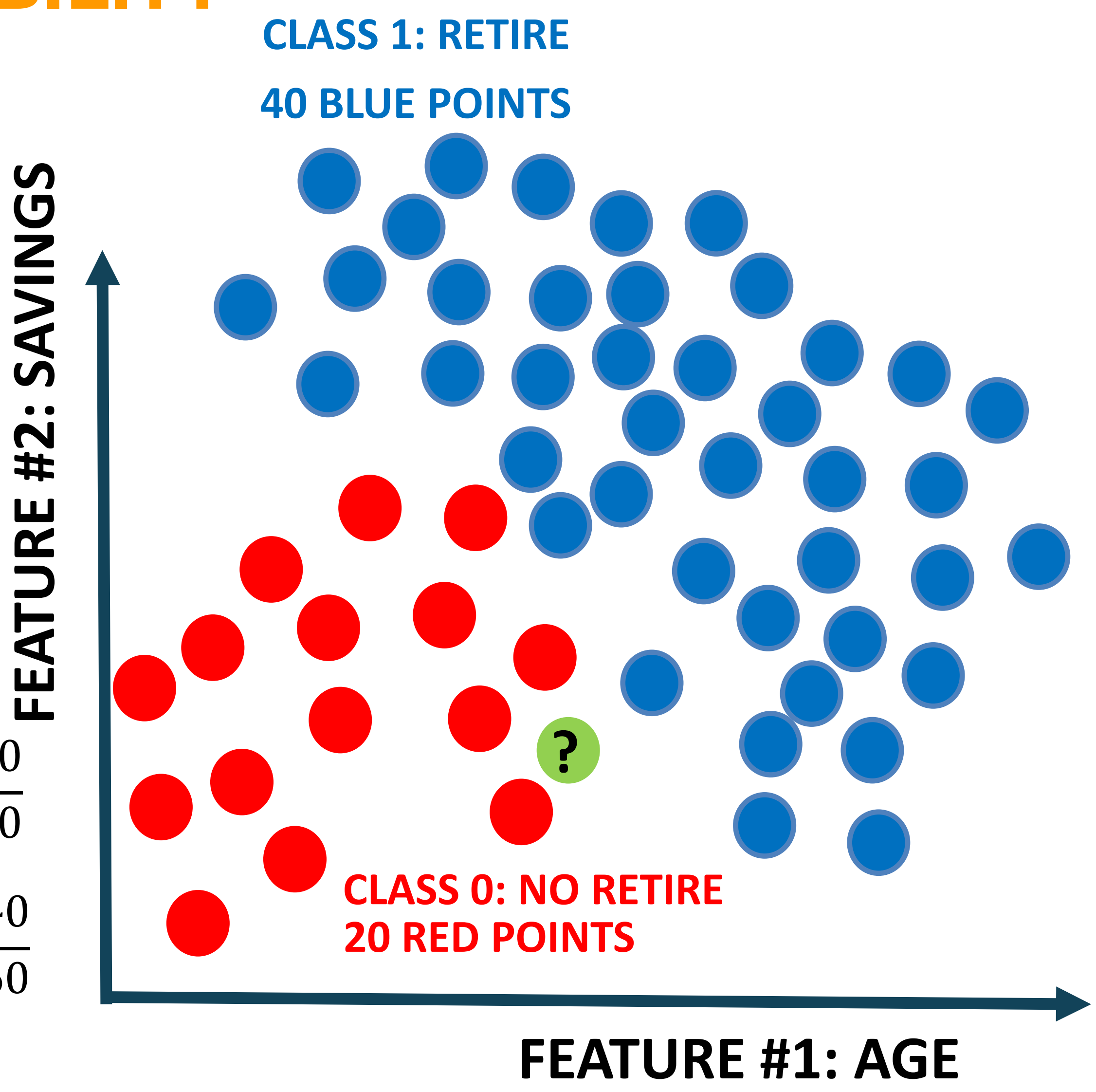


NAÏVE BAYES: 1. PRIOR PROBABILITY

- Points can be classified as RED or BLUE.
- Our task is to classify a new point to RED or BLUE.
- **Prior Probability:** Since we have more BLUE compared to RED, we can assume that our new point is twice as likely to be BLUE than RED.

$$Prior\ Probability\ for\ RED = \frac{Number\ of\ RED\ Points}{Total\ Number\ of\ Points} = \frac{20}{60}$$

$$Prior\ Probability\ for\ BLUE = \frac{Number\ of\ BLUE\ Points}{Total\ Number\ of\ Points} = \frac{40}{60}$$

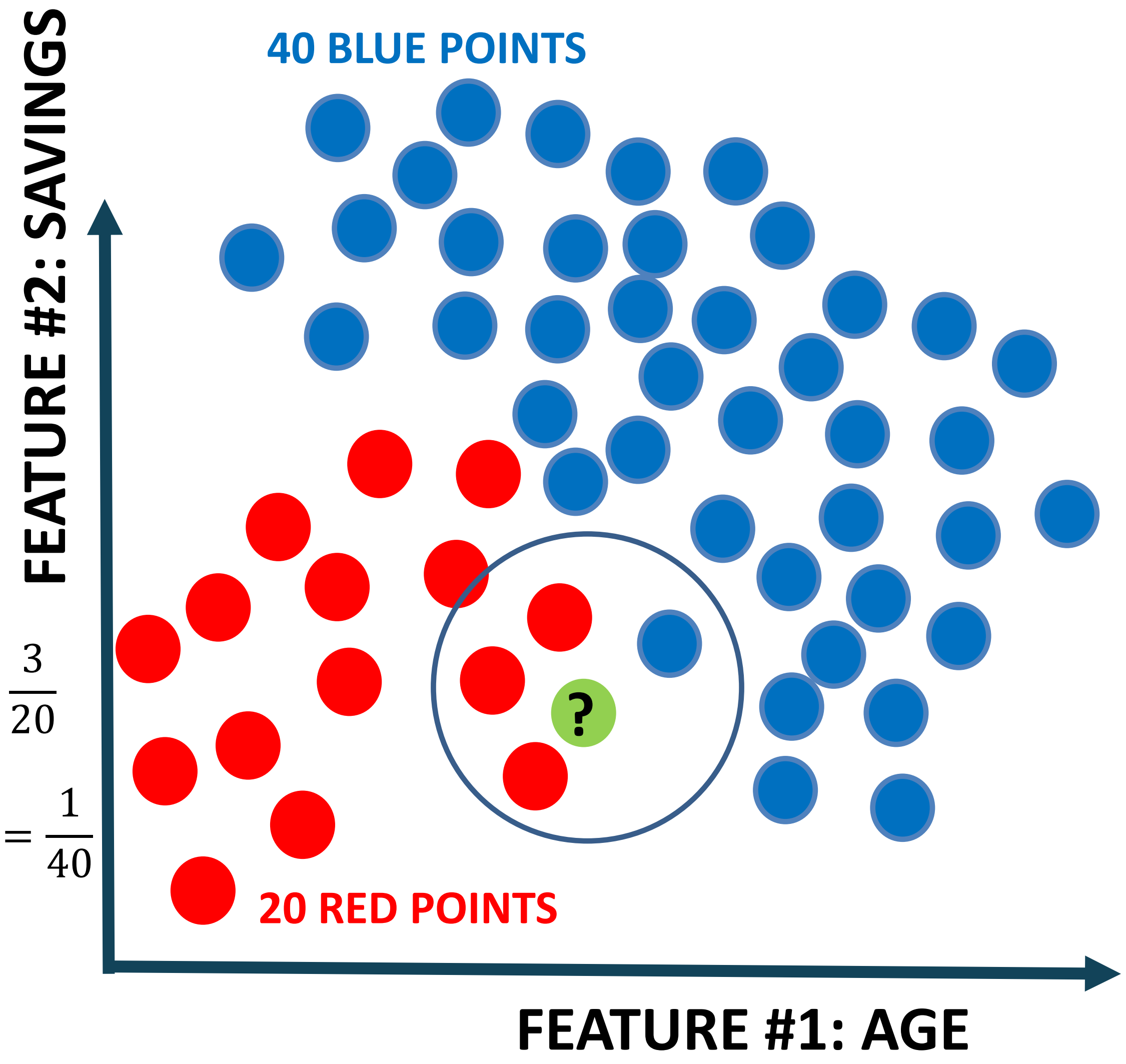


NAÏVE BAYES: 2. LIKELIHOOD

- For the new point, if there are more **BLUE** points in its vicinity, it is more likely that the new point will be classified as **BLUE**.
- So we draw a circle around the point
- Then we calculate the number of points in the circle belonging to each class label.

$$Likelihood\ of\ X\ being\ RED = \frac{Number\ of\ RED\ Points\ in\ vicinity}{Total\ Number\ of\ RED\ Points} = \frac{3}{20}$$

$$Likelihood\ of\ X\ being\ BLUE = \frac{Number\ of\ BLUE\ Points\ in\ vicinity}{Total\ Number\ of\ BLUE\ Points} = \frac{1}{40}$$

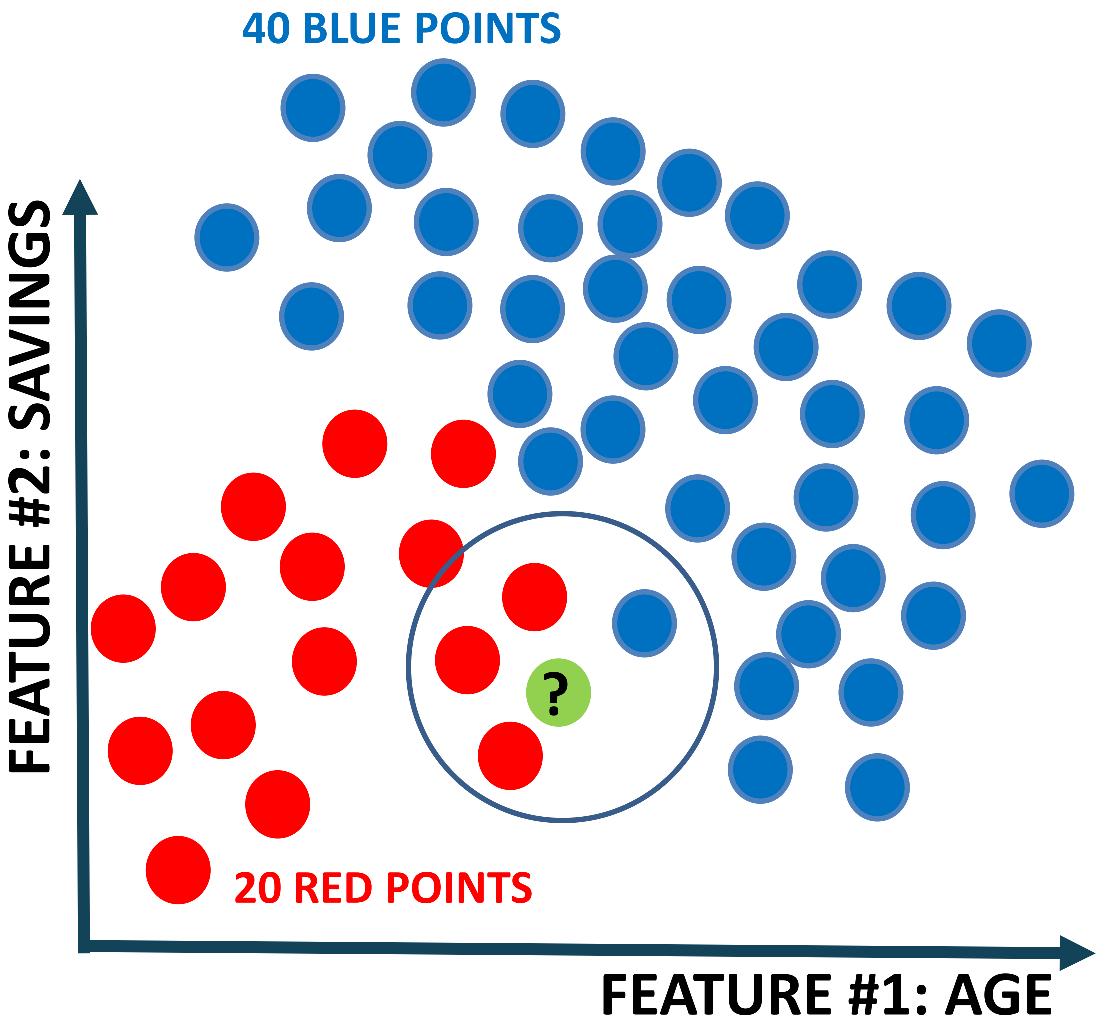


NAÏVE BAYES: 3. POSTERIOR PROBABILITY

- Let's combine prior probability and likelihood to create a posterior probability.
- **Prior probabilities:** suggests that X may be classified as BLUE Because there are twice as much blue points.
- **Likelihood:** suggests that X is RED because there are more RED points in the vicinity of X.
- Bayes' Rule combines both to form a posterior probability.

$$\begin{aligned} &\textit{Posterior Probability of X being RED} \\ &\quad = \textit{Prior Probability of RED} \\ &\quad * \textit{Likelihood of X being RED} = \frac{20}{60} * \frac{3}{20} = \frac{1}{20} \end{aligned}$$

$$\begin{aligned} &\textit{Posterior Probability of X being BLUE} \\ &\quad = \textit{Prior Probability of BLUE} \\ &\quad * \textit{Likelihood of X being BLUE} = \frac{40}{60} * \frac{1}{40} = \frac{1}{60} \end{aligned}$$



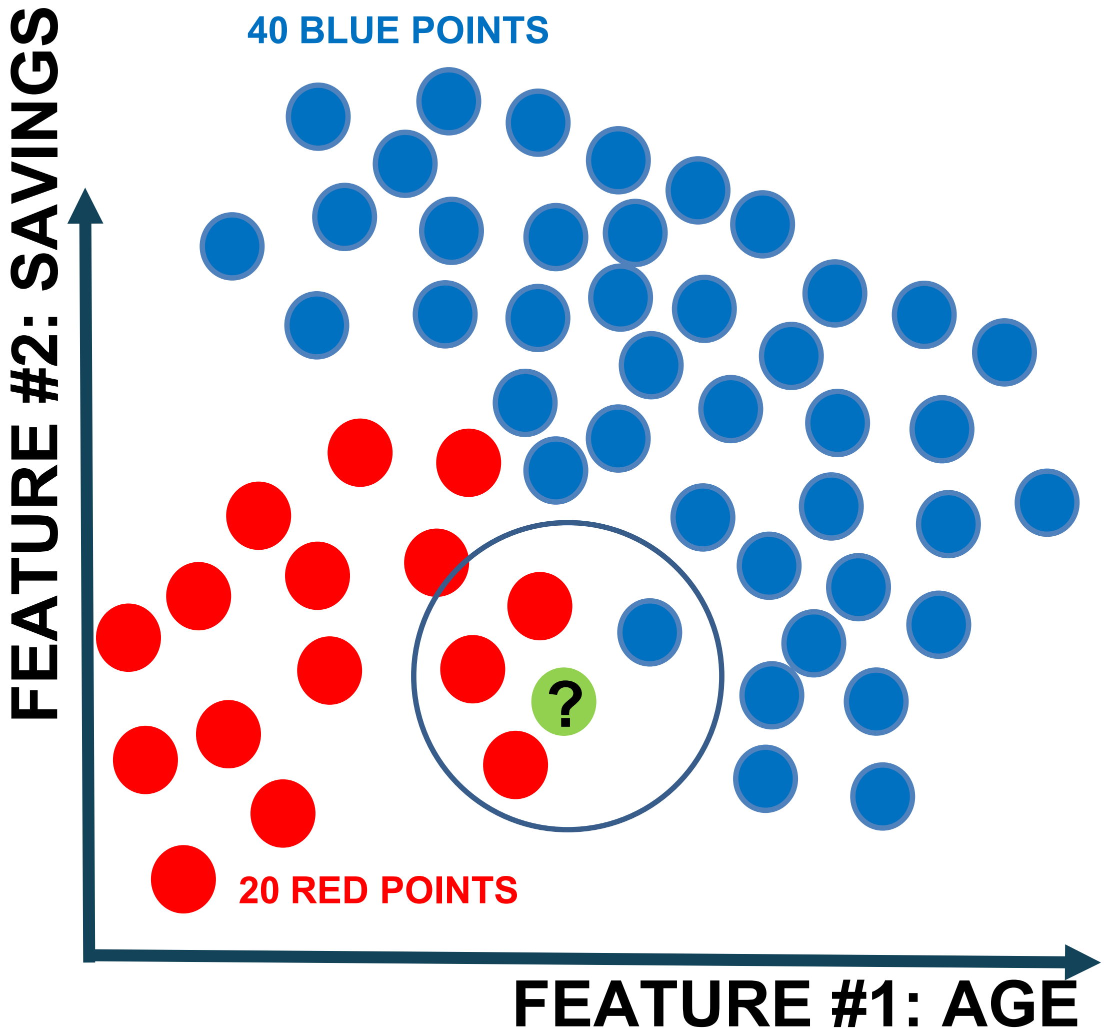
X CLASSIFIED AS RED (NON RETIRING) SINCE IT HAS LARGER POSTERIOR PROBABILITY

NAÏVE BAYES: REVIEW

- Let's combine prior probability and likelihood to create a posterior probability.
- **Prior probabilities:** suggests that X may be classified as BLUE Because there are twice as much blue points.
- **Likelihood:** suggests that X is RED because there are more RED points in the vicinity of X.
- Bayes' Rule combines both to form a posterior probability.

$$Posterior\ Probability\ of\ X\ being\ RED = Prior\ Probability\ of\ RED * Likelihood\ of\ X\ being\ RED = \frac{20}{60} * \frac{3}{20} = \frac{1}{20}$$

$$Posterior\ Probability\ of\ X\ being\ BLUE = Prior\ Probability\ of\ BLUE * Likelihood\ of\ X\ being\ BLUE = \frac{40}{60} * \frac{1}{40} = \frac{1}{60}$$



X CLASSIFIED AS RED (NON RETIRING) SINCE IT HAS LARGER POSTERIOR PROBABILITY

NAÏVE BAYES: SOME MATH!

- Naïve Bayes is a classification technique based on Bayes' Theorem.

LIKELIHOOD

PRIOR PROBABILITY OF RETIRING

$$P(Retire|X) = \frac{P(X|Retire) * P(Retire)}{P(X)}$$

MARGINAL LIKELIHOOD

- X: New Customer's features; age and savings
- $P(Retire|X)$: probability of customer retiring given his/her features, such as age and savings
- $P(Retire)$: Prior probability of retiring, without any prior knowledge
- $P(X|Retire)$: likelihood
- $P(X)$: Marginal likelihood, the probability of any point added lies into the circle

NAÏVE BAYES: SOME MATH!

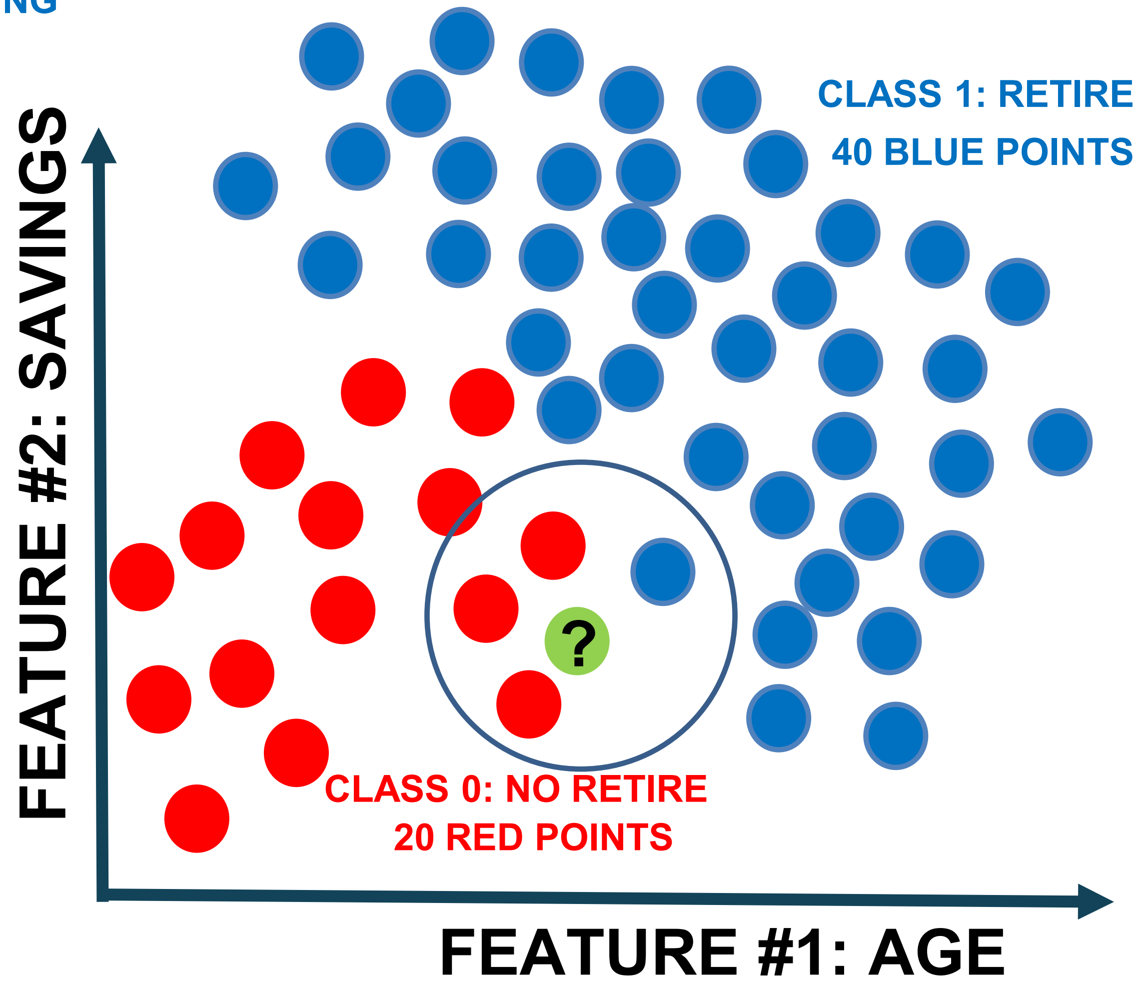
LIKELIHOOD

PRIOR PROBABILITY
OF RETIRING

$$P(Retire|X) = \frac{P(X|Retire) * P(Retire)}{P(X)}$$

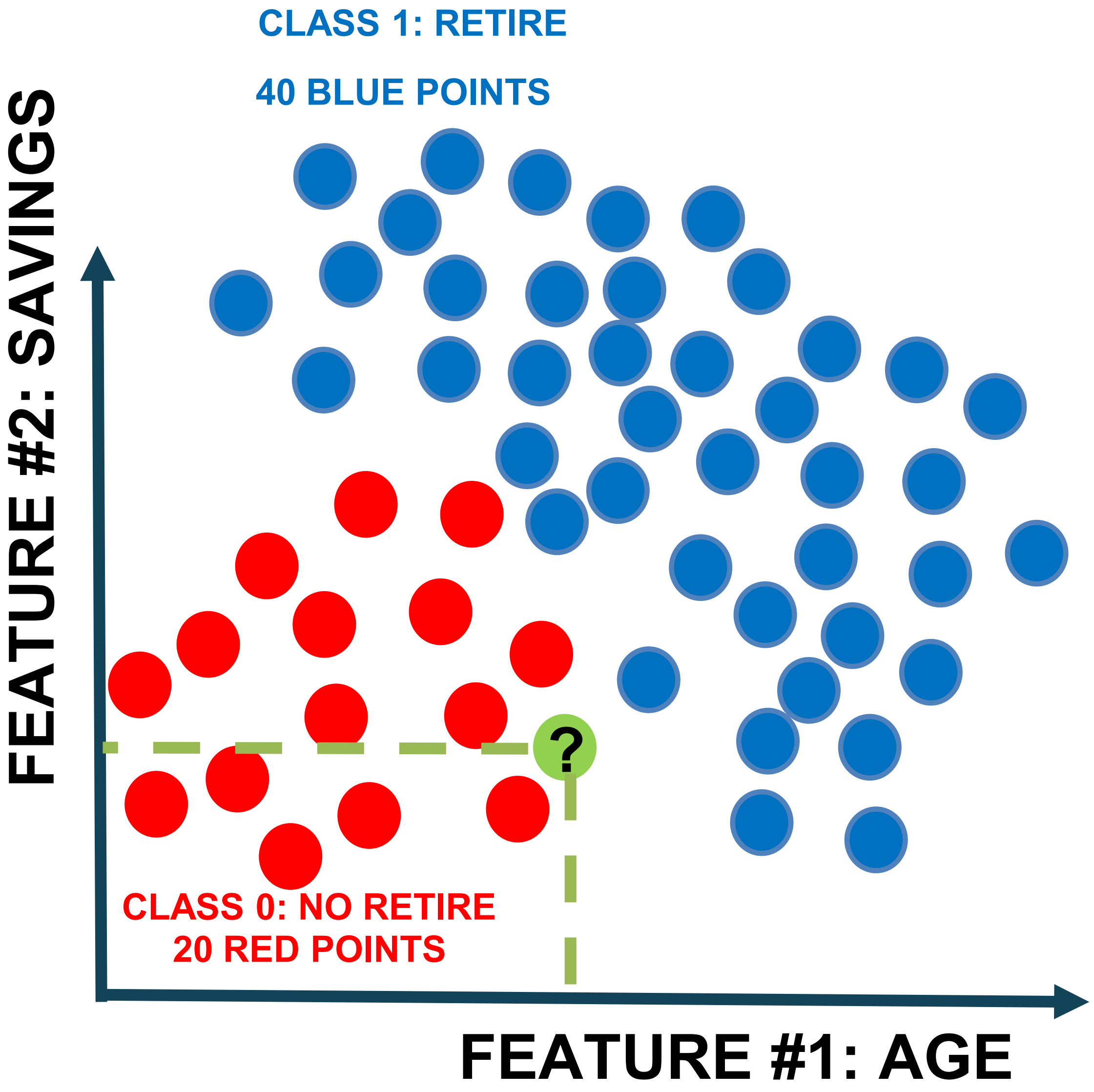
MARGINAL LIKELIHOOD

- $P(Retire) = \frac{\text{\# of Retiring}}{\text{Total points}} = 40/60$
- $P(X|Retire) = \frac{\text{\# of smilar observations for retiring}}{Total \text{ \# } retiring} = 1/40$
- $P(X) = \frac{\text{\# of Similar observations}}{Total \text{ \# } Points} = 4/60$
- $P(Retire|X) = \frac{\frac{40}{60} * \frac{1}{40}}{\frac{4}{60}} = \frac{1/60}{4/60} = 0.25$



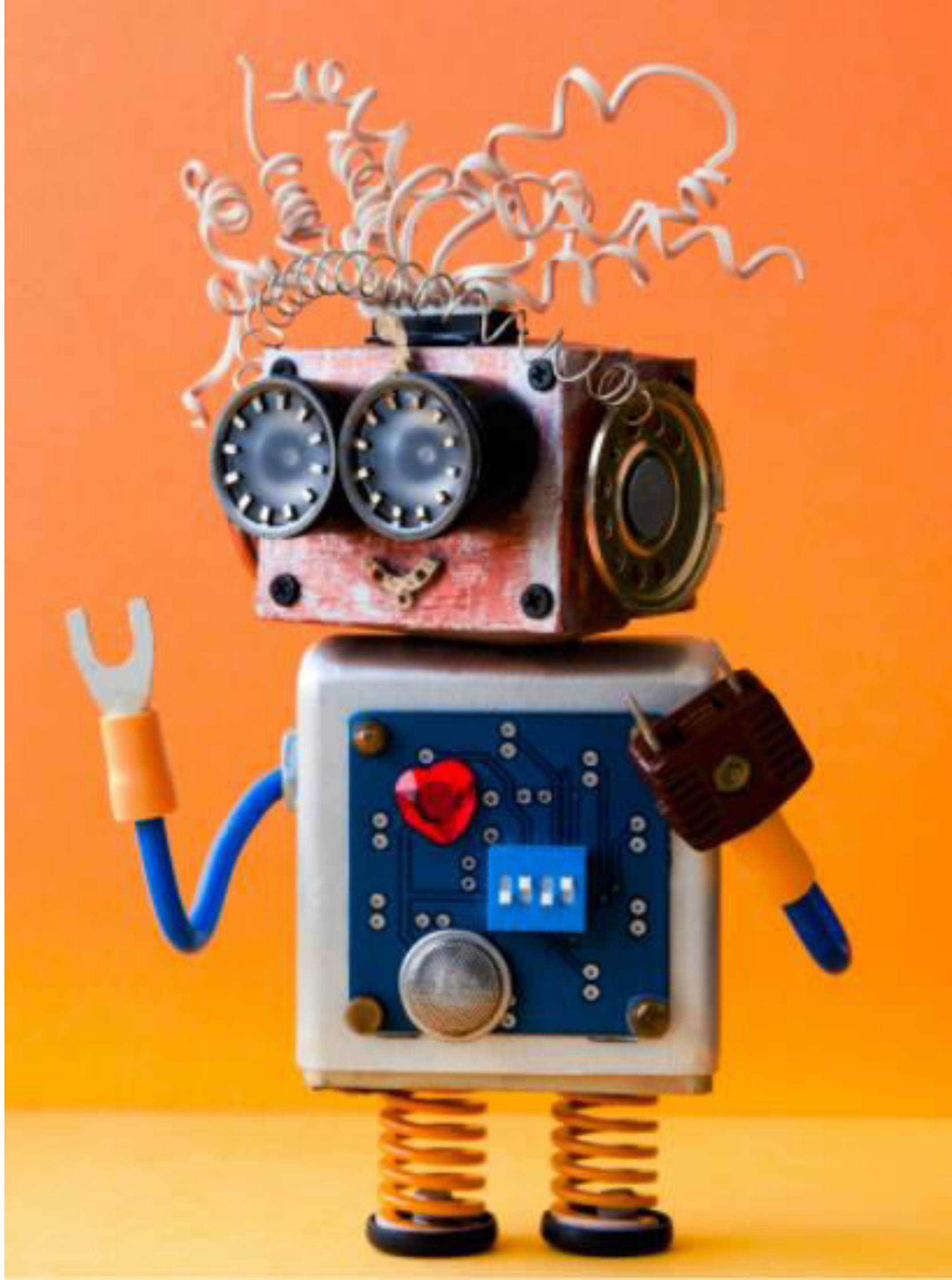
NAÏVE BAYES: WHY NAÏVE?

- It is called naive because it assumes that the presence of a certain feature in a class is independent of the presence of other features.
- EXAMPLE #1: Age/savings, the assumption is not necessarily true since age and savings might be dependant on each others
- EXAMPLE #2: fruit can be classified as watermelon if its color is green, tastes sweet, and round.
- These features might be dependant on each others, however, we assume they are all independent and that's why its 'Naive'!



PRACTICE OPPORTUNITY 5

EASY ADVANCED



NAÏVE BAYES: QUIZ/CALCULATE THE PROBABILTY OT NON-RETIRING (RED CLASS)

$$P(No\ Retire|X) =?$$

NAÏVE BAYES: QUIZ/CALCULATE THE PROBABILTY OT NON-RETIRING (RED CLASS)

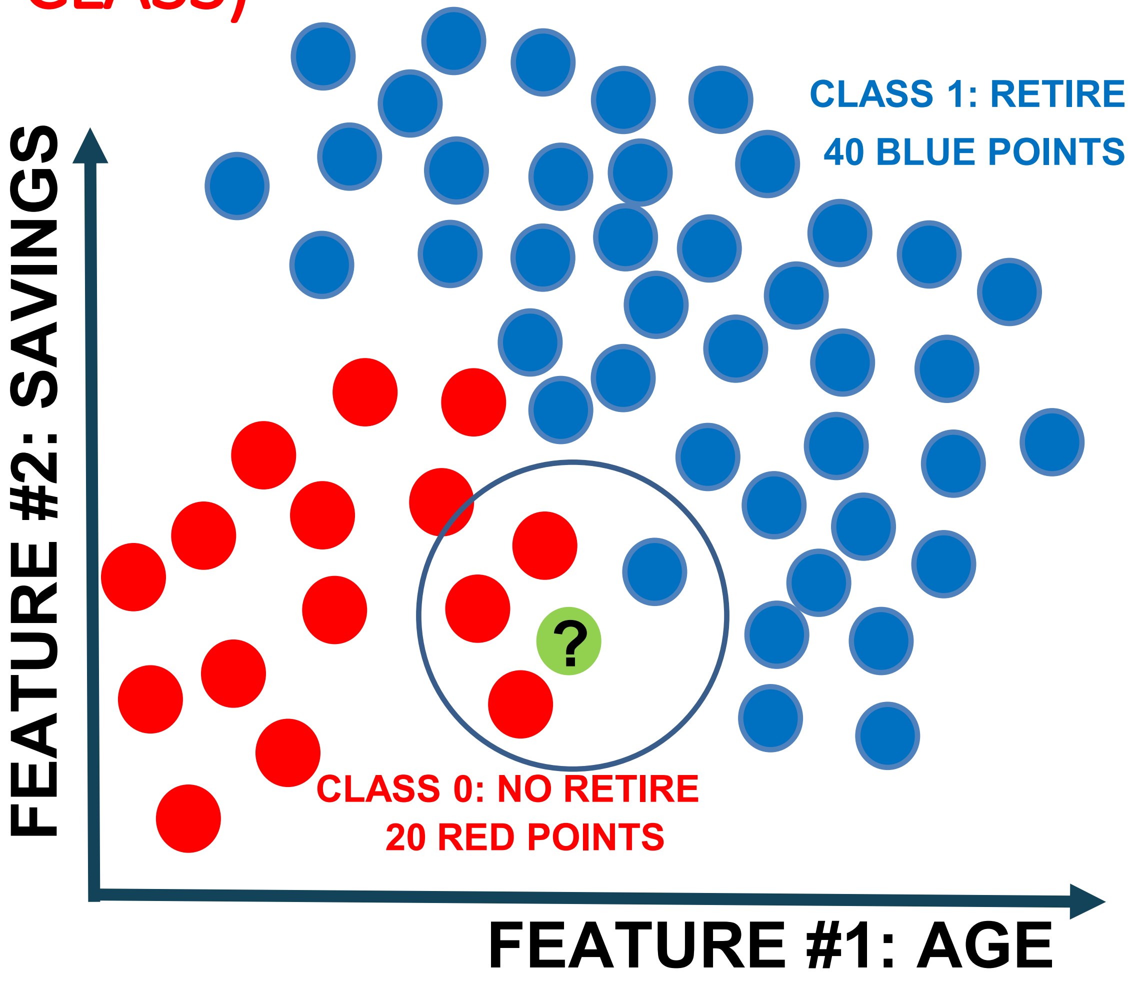
LIKELIHOOD

PRIOR PROBABILITY OF NO RETIRING

$$P(No\ Retire|X) = \frac{P(X|No\ Retire) * P(No\ Retire)}{P(X)}$$

MARGINAL LIKELIHOOD

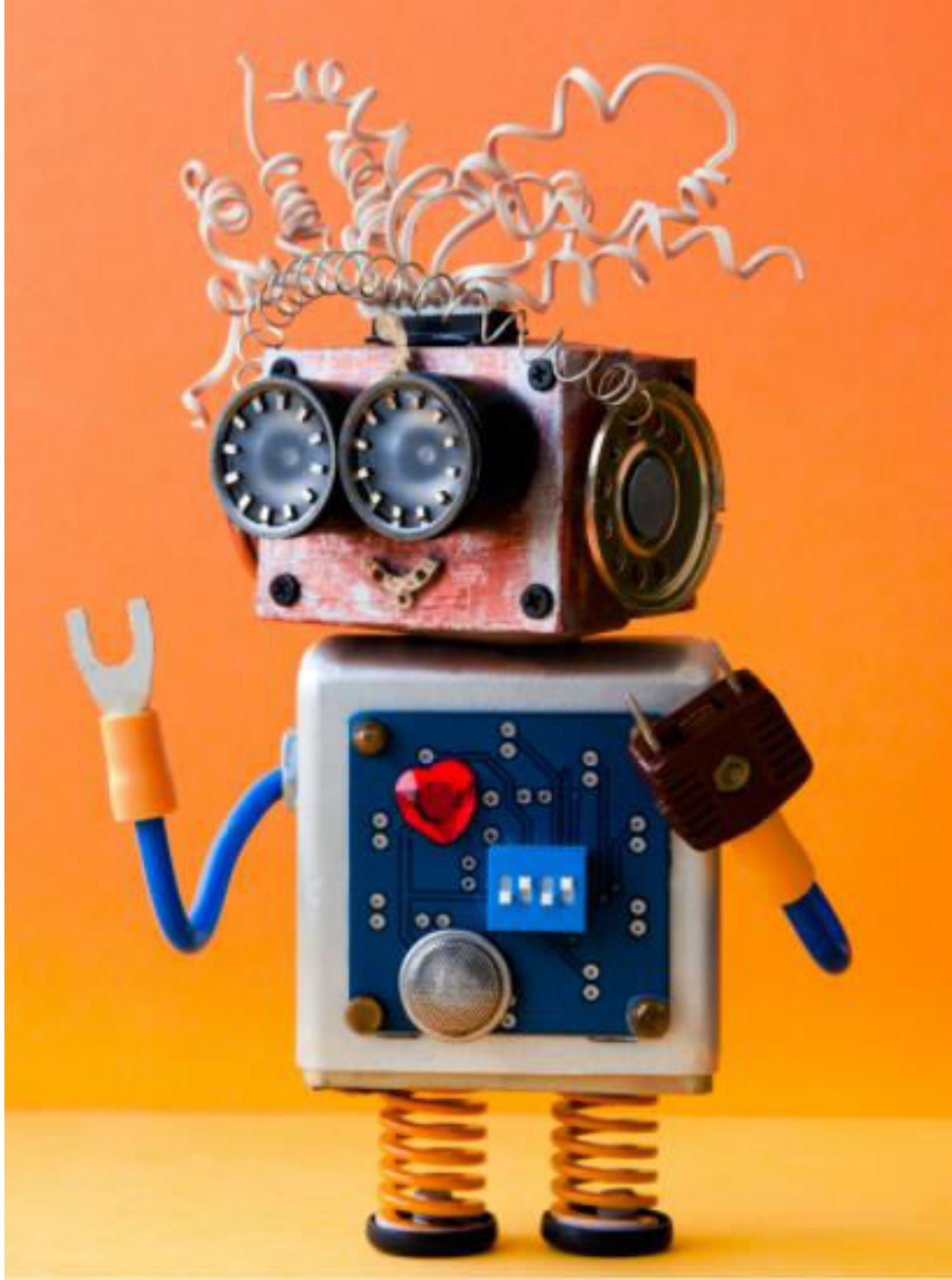
- $P(No\ Retire) = \frac{\text{\# of No Retiring}}{\text{Total points}} = 20/60$
- $P(X|No\ Retire) = \frac{\text{\# of smilar observations for No retiring}}{Total\ \#\ no\ retiring} = 3/20$
- $P(X) = \frac{\text{\# of Similar observations}}{Total\ \#\ Points} = 4/60$
- $P(No\ Retire|X) = \frac{\frac{20}{60} * \frac{3}{20}}{\frac{4}{60}} = \frac{3/60}{4/60} = 0.75$



$\boldsymbol{NOTE: P(Non\ Retire|X) = 1 - 0.25 = 0.75}$

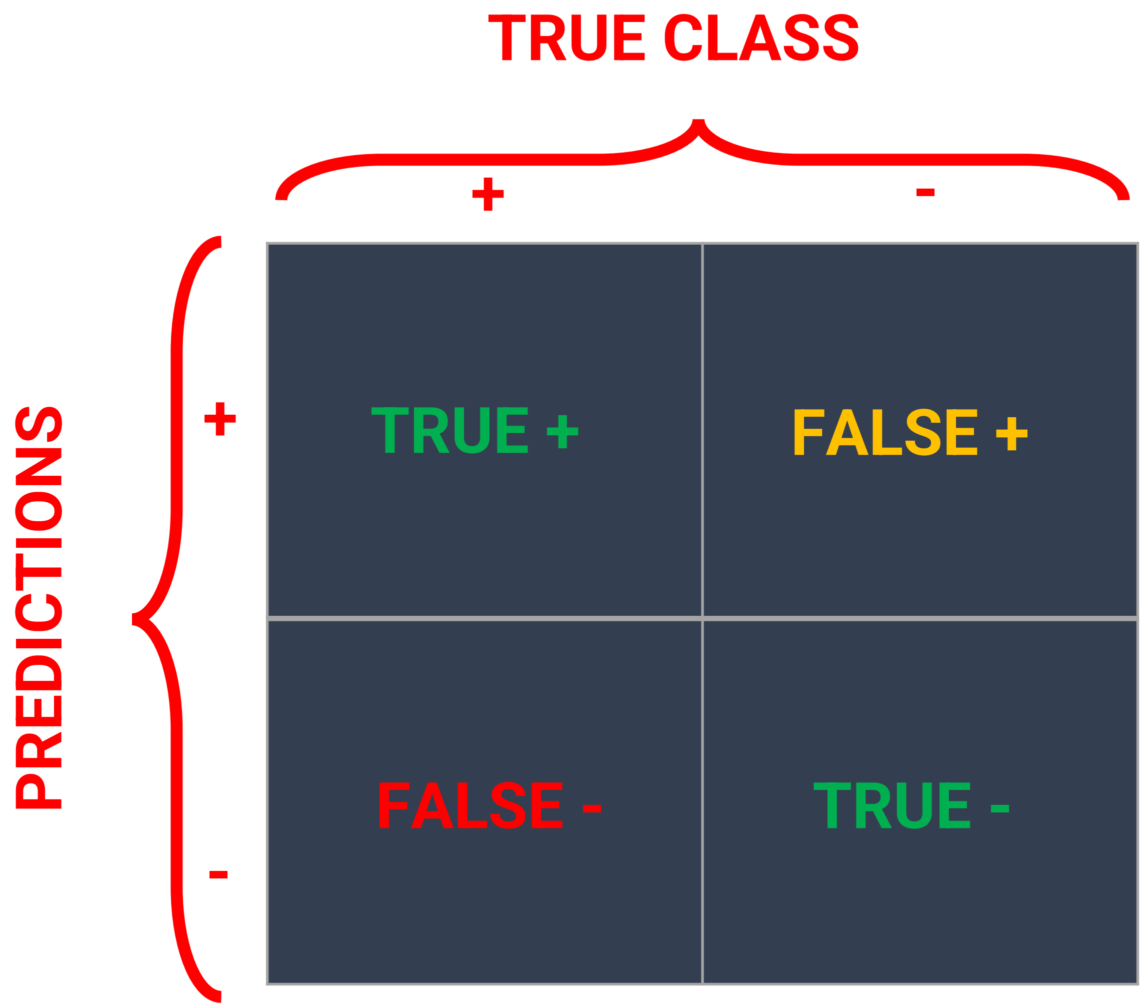
CLASSIFICATION MODELS KPIs RECAP [SKIP IF FAMILIAR]

EASY ADVANCED



CLASSIFICATION MODEL KPIs

- Classification Accuracy = (TP+TN) / (TP + TN + FP + FN)
- Misclassification rate (Error Rate) = (FP + FN) / (TP + TN + FP + FN)
- Precision = TP/Total TRUE Predictions = TP/ (TP+FP) (When model predicted TRUE class, how often was it right?)
- Recall = TP/ Actual TRUE = TP/ (TP+FN) (when the class was actually TRUE, how often did the classifier get it right?)



PRECISION Vs. RECALL EXAMPLE

FACTS:
100 PATIENTS TOTAL
91 PATIENTS ARE HEALTHY
9 PATIENTS HAVE CANCER

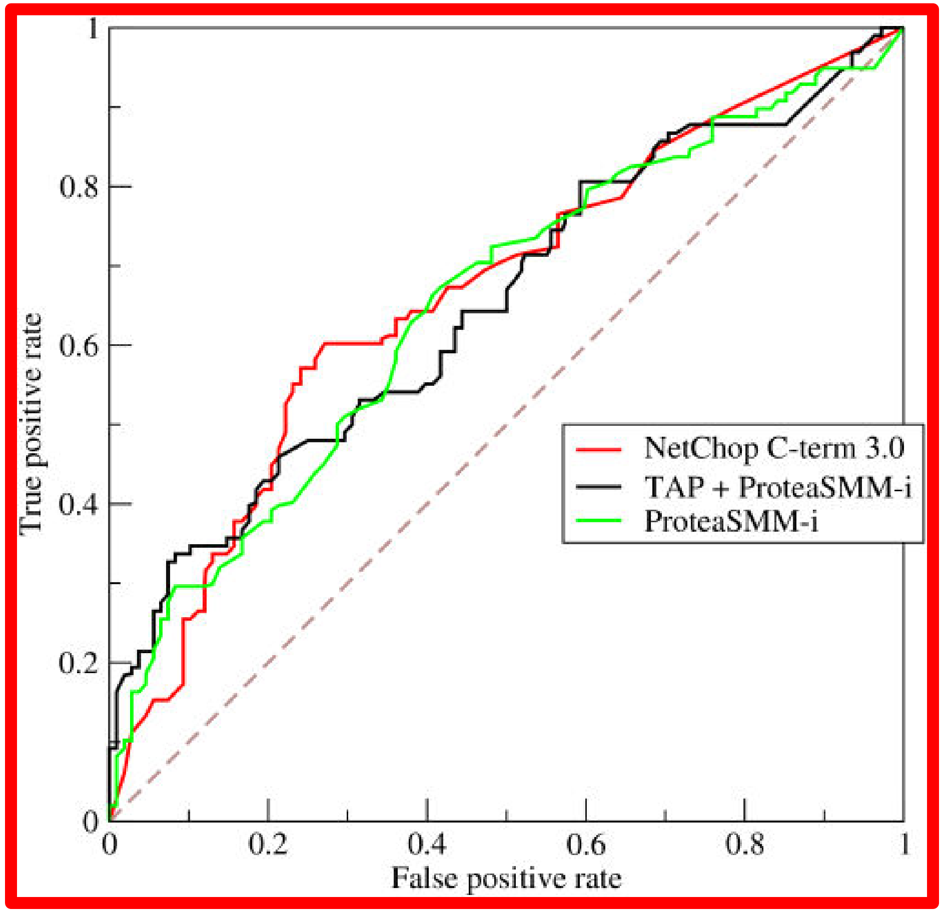
TRUE CLASS

PREDICTIONS	+	-
+	TP = 1	FP = 1
-	FN = 8	TN = 90

- Accuracy is generally misleading and is not enough to assess the performance of a classifier.
- Recall is an important KPI in situations where:
  - Dataset is highly unbalanced; cases when you have small cancer patients compared to healthy ones.

- Classification Accuracy = (TP+TN) / (TP + TN + FP + FN) = 91%
- Precision = TP/Total TRUE Predictions = TP/ (TP+FP) = ½=50%
- Recall = TP/ Actual TRUE = TP/ (TP+FN) = 1/9 = 11%

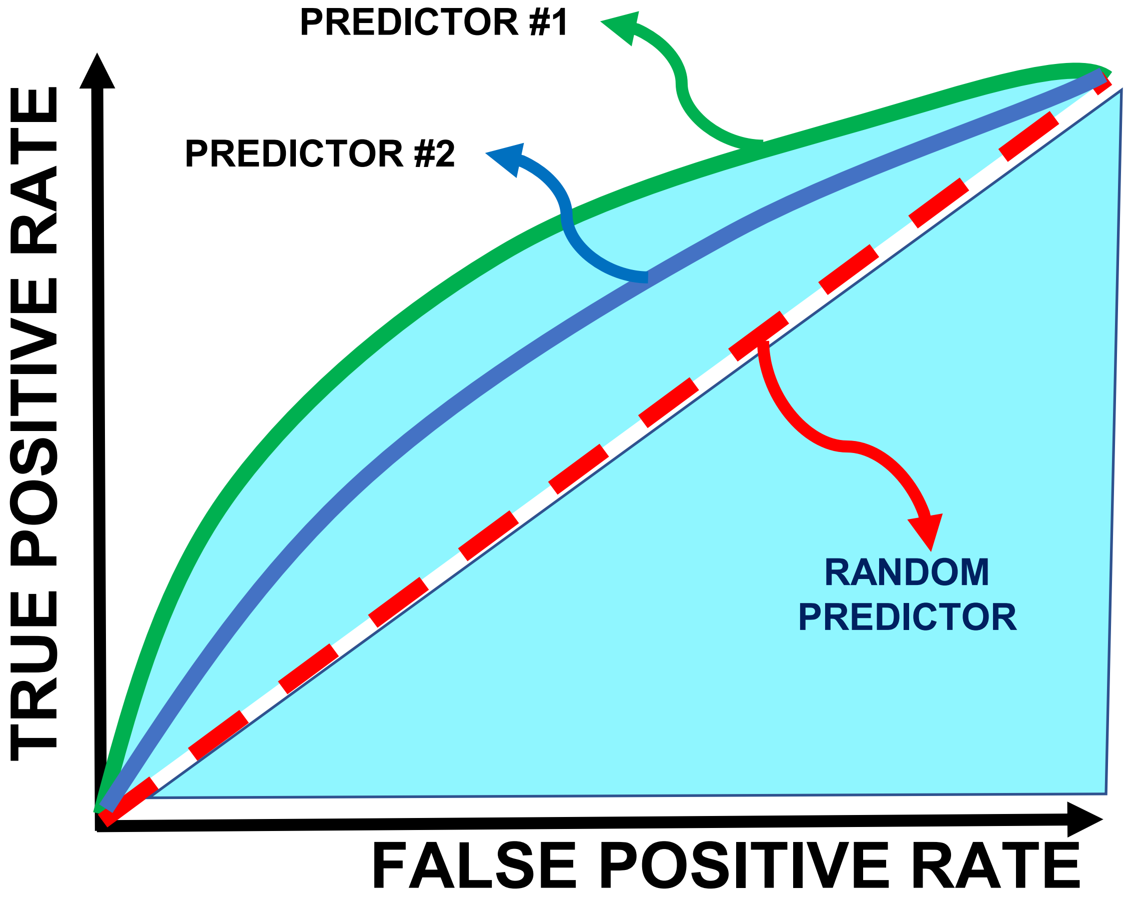
ROC (RECEIVER OPERATING CHARACTERISTIC CURVE)



- ROC Curve is a metric that assesses the model ability to distinguish between binary (0 or 1) classes.
- The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.
- The true-positive rate is also known as sensitivity, recall or probability of detection in machine learning.
- The false-positive rate is also known as the probability of false alarm and can be calculated as (1 − specificity).
- Points above the diagonal line represent good classification (better than random)
- The model performance improves if it becomes skewed towards the upper left corner.

Photo Credit: https://commons.wikimedia.org/wiki/File:Roccurves.png

AUC (AREA UNDER CURVE)



- The light blue area represents the area Under the Curve of the Receiver Operating Characteristic (AUROC).
- The diagonal dashed red line represents the ROC curve of a random predictor with AUROC of 0.5.
- If ROC AUC = 1, perfect classifier
- Predictor #1 is better than predictor #2
- Higher the AUC, the better the model is at predicting 0s as 0s and 1s as 1s.