MACHINE LEARNING COURSE

PRESENTED BY ABDEL RAHMAN ALSABBAGH

FINAL LECTURE #6 – SAT – 27.5.2023

بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ

In the name of Allah, the most gracious, the most merciful, we start :)

Today's Quote

"The winds of change may blow away everything in their path, but the lingering effect remains, shaping the landscape of our lives."

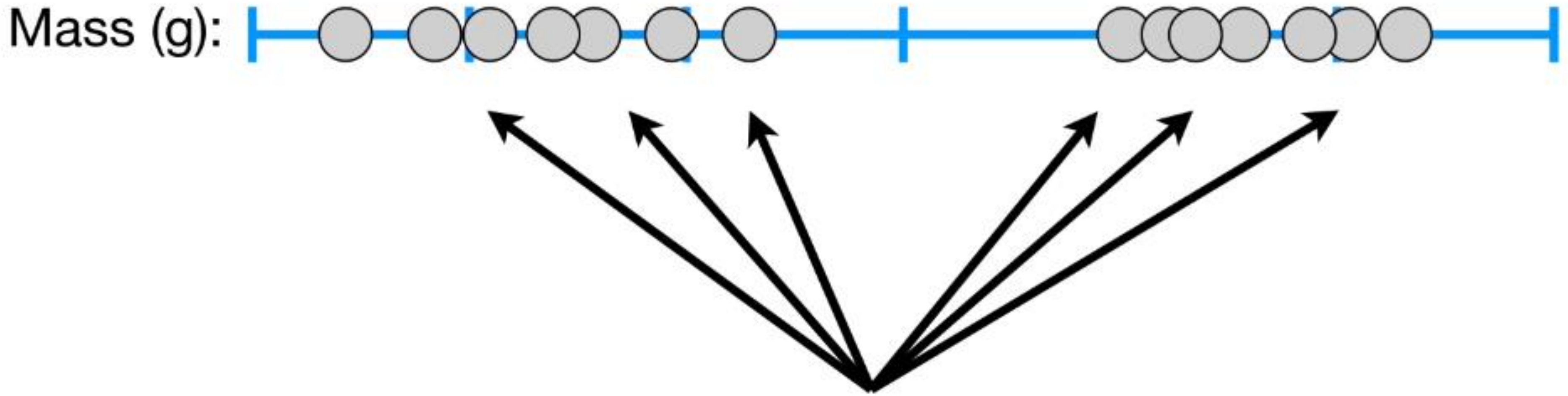
- ChatGPT

Support Vector Machines

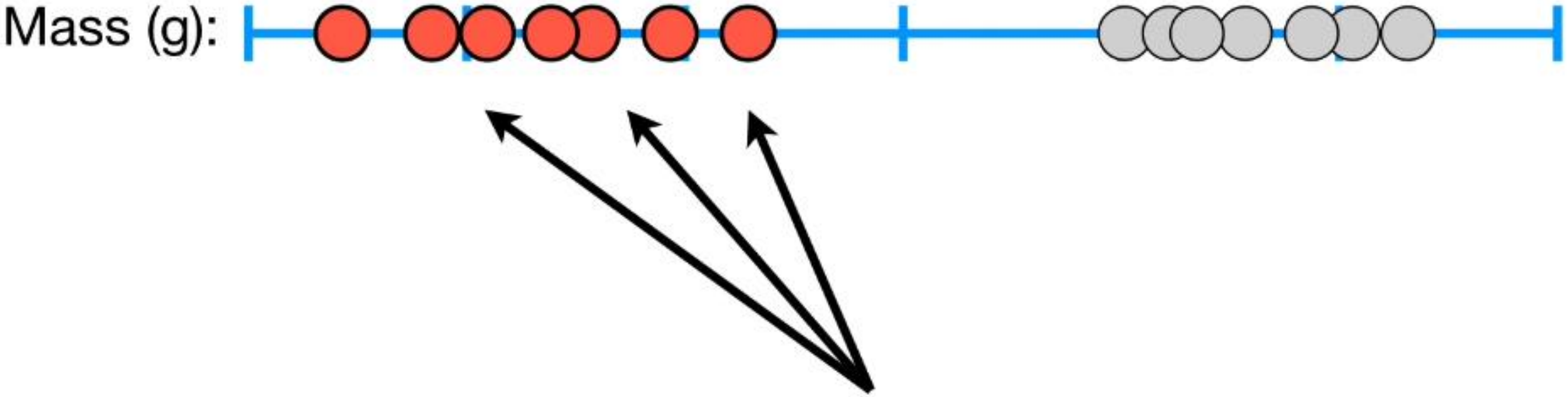
- Support Vector Machines (SVMs).
- Kernel Functions.
- Decision trees.
- Random forests.
- Ensemble learning.
- Evaluation metrics.
- Code practice.

Source: StatQuest by Josh Starmer.

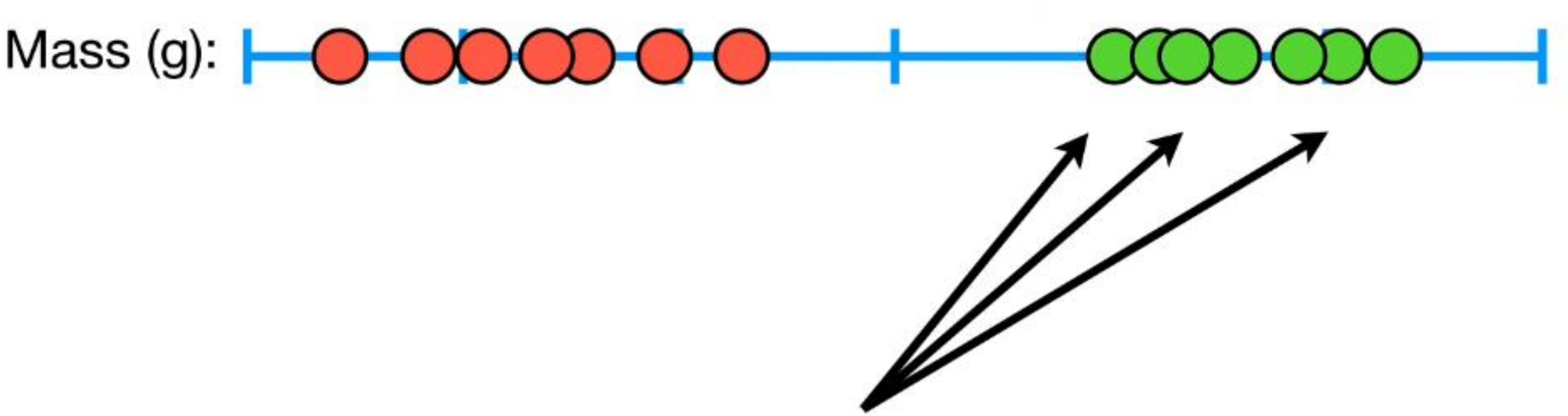
Support Vector Machines



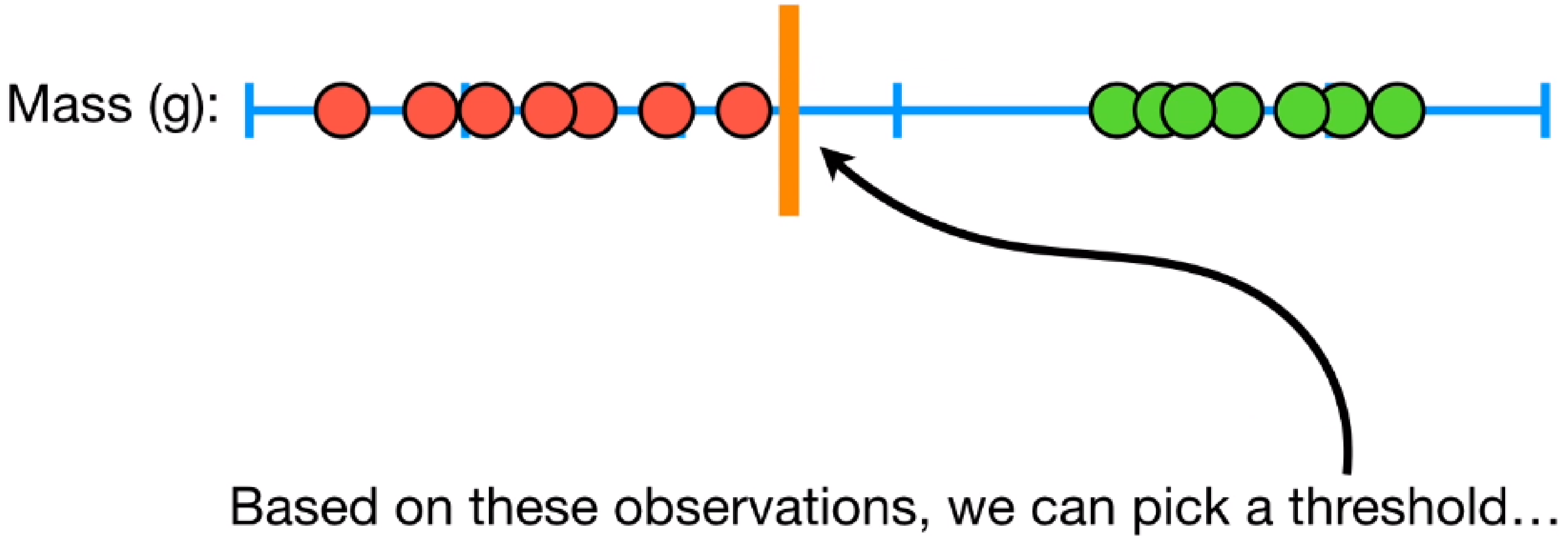
Support Vector Machines



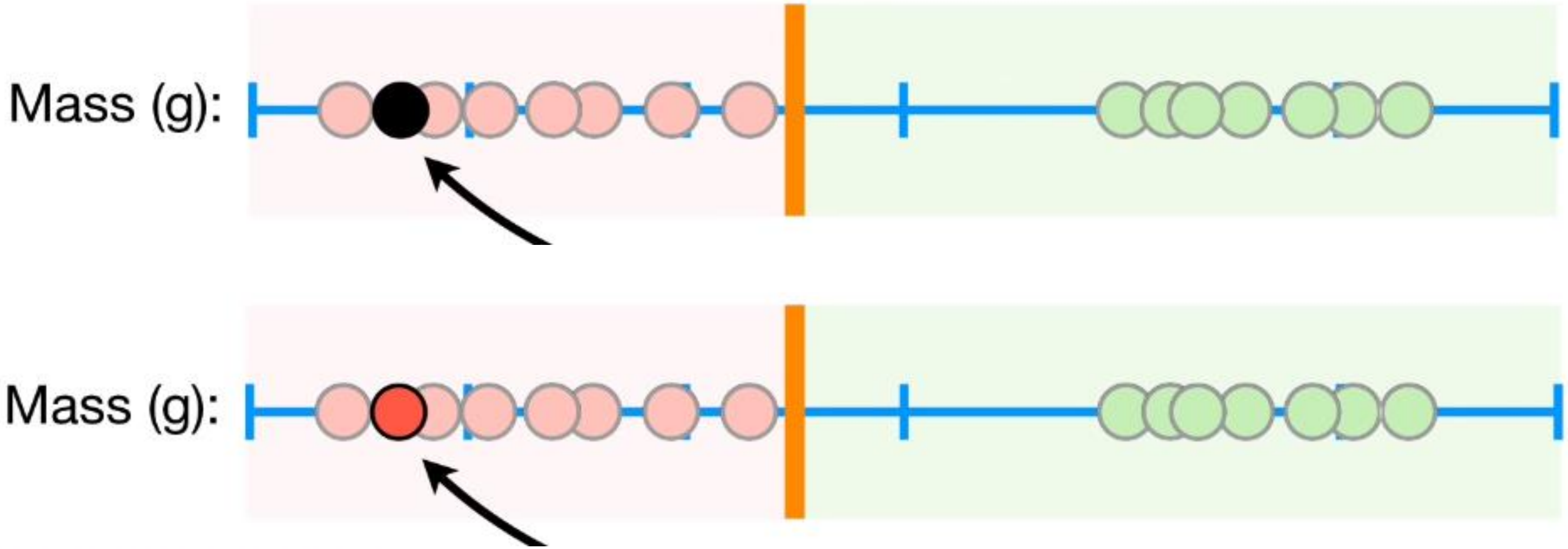
Support Vector Machines



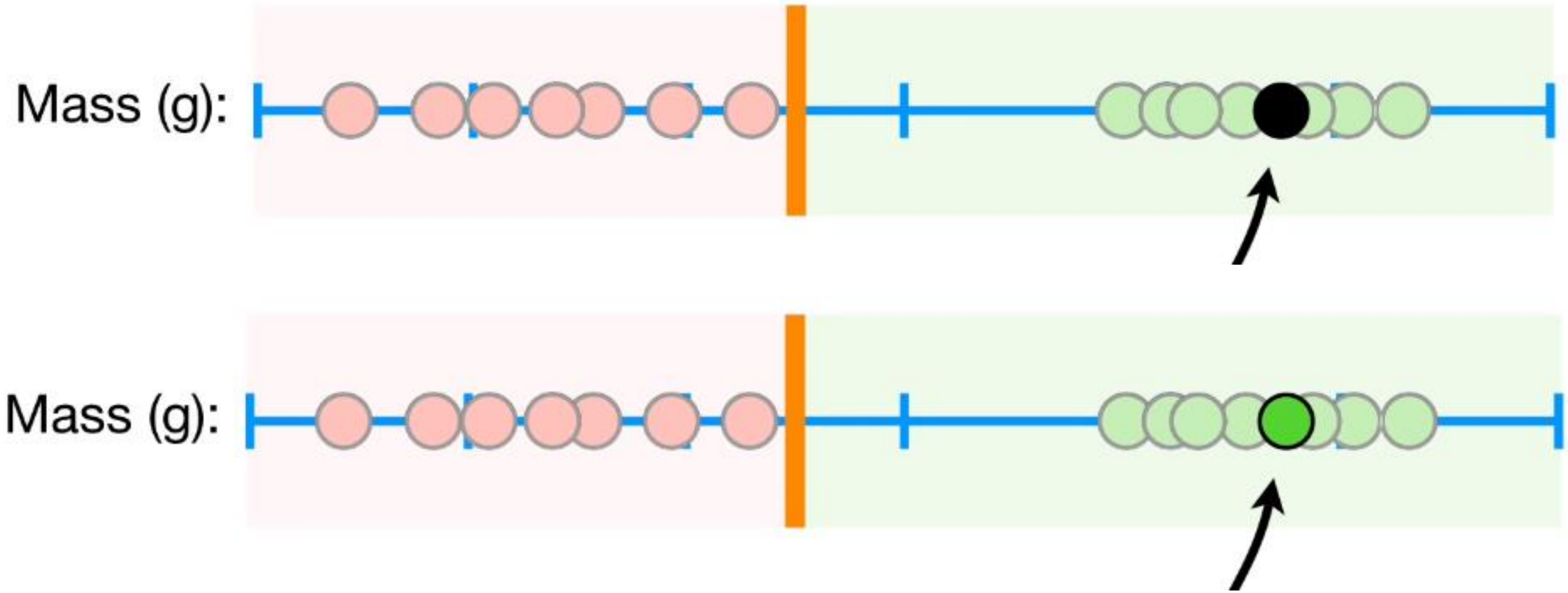
Support Vector Machines



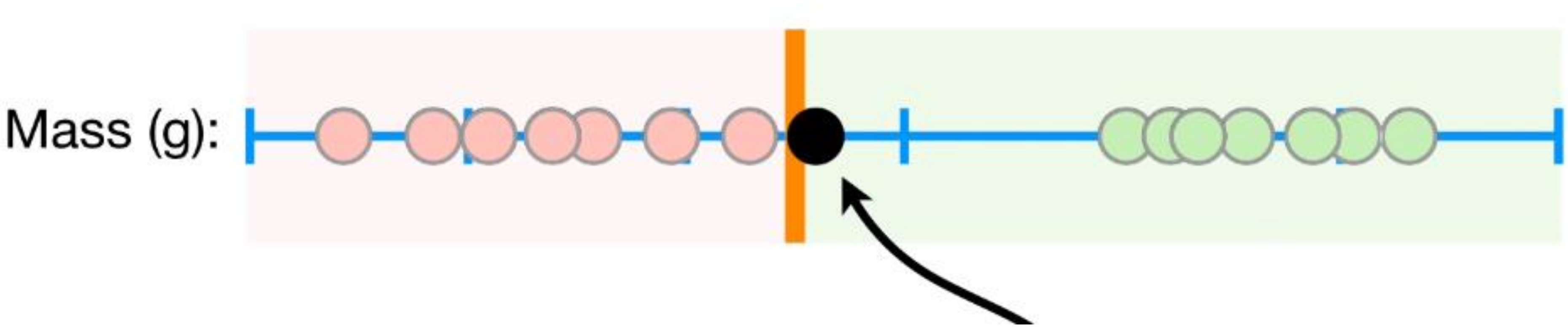
Support Vector Machines



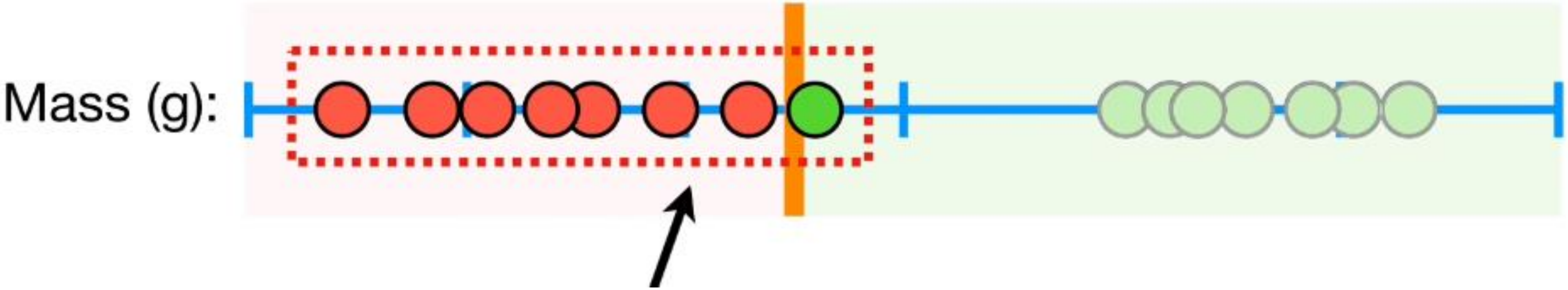
Support Vector Machines



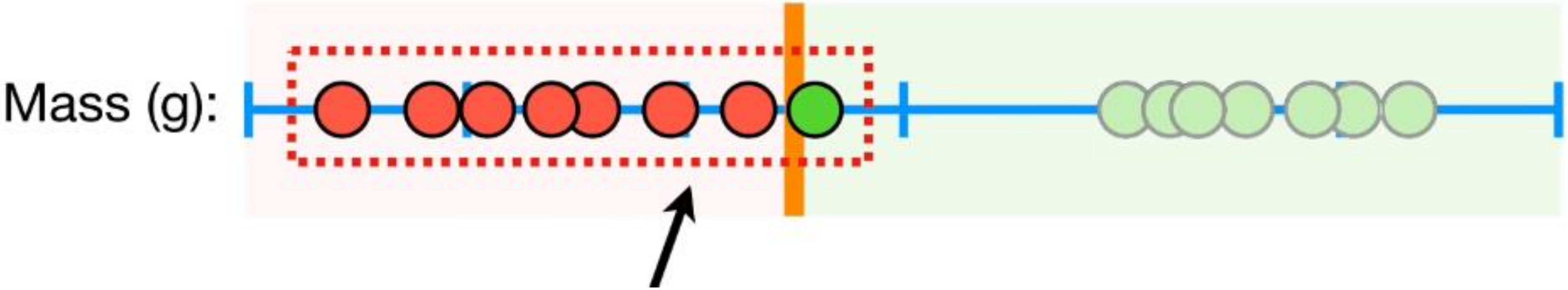
Support Vector Machines



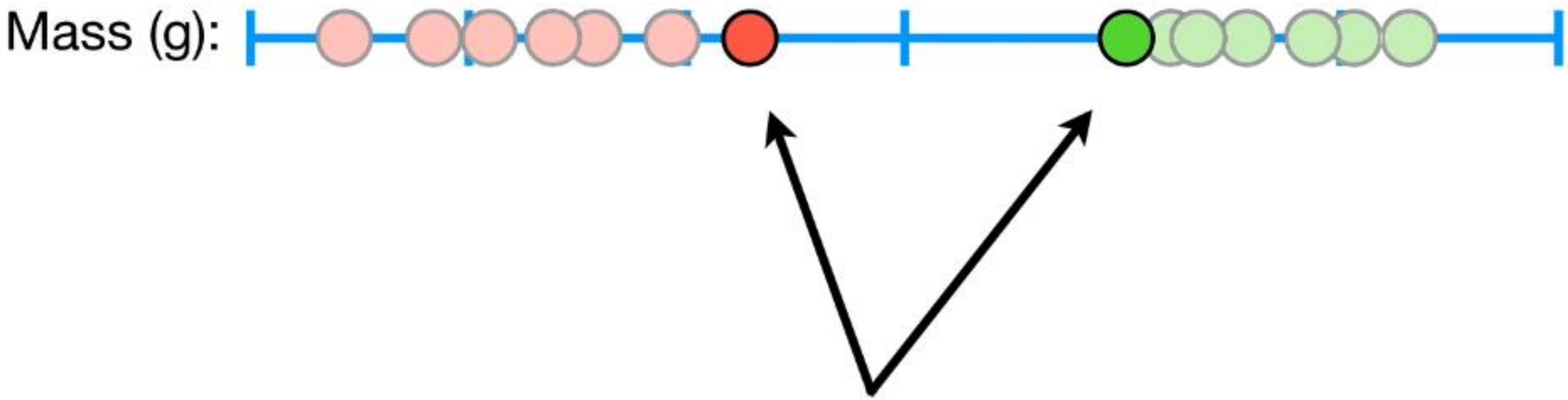
Support Vector Machines



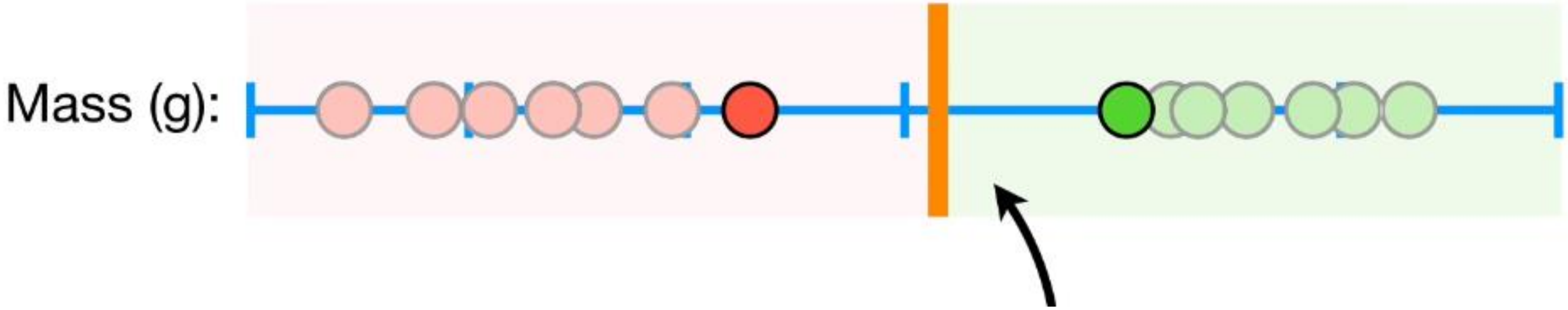
Support Vector Machines



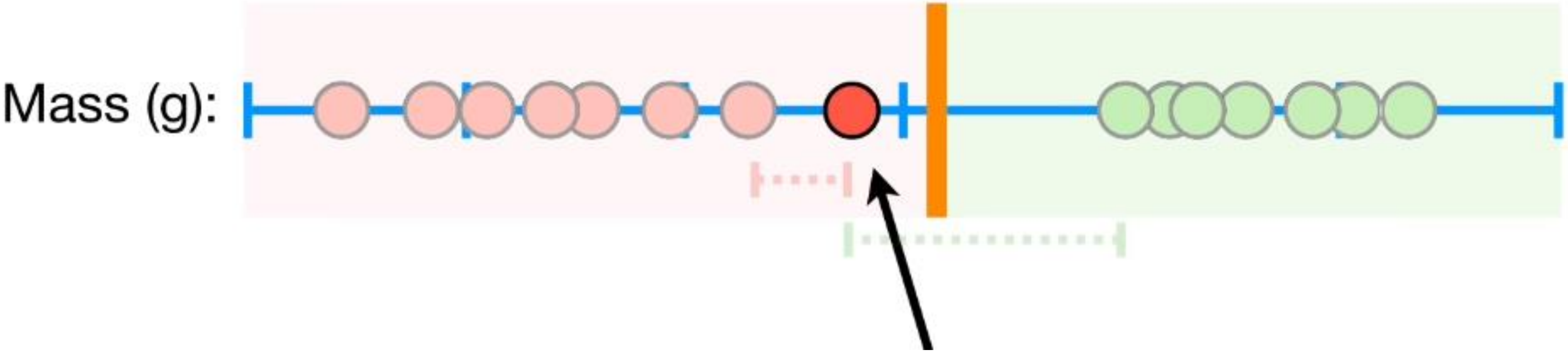
Support Vector Machines



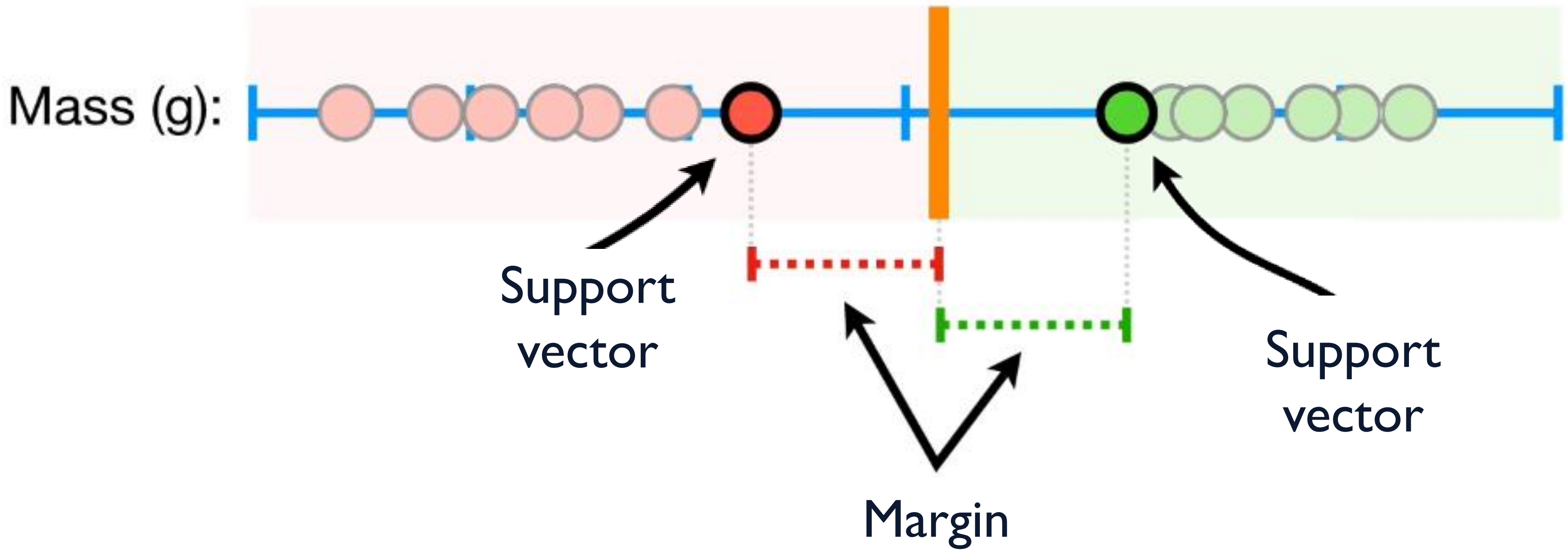
Support Vector Machines



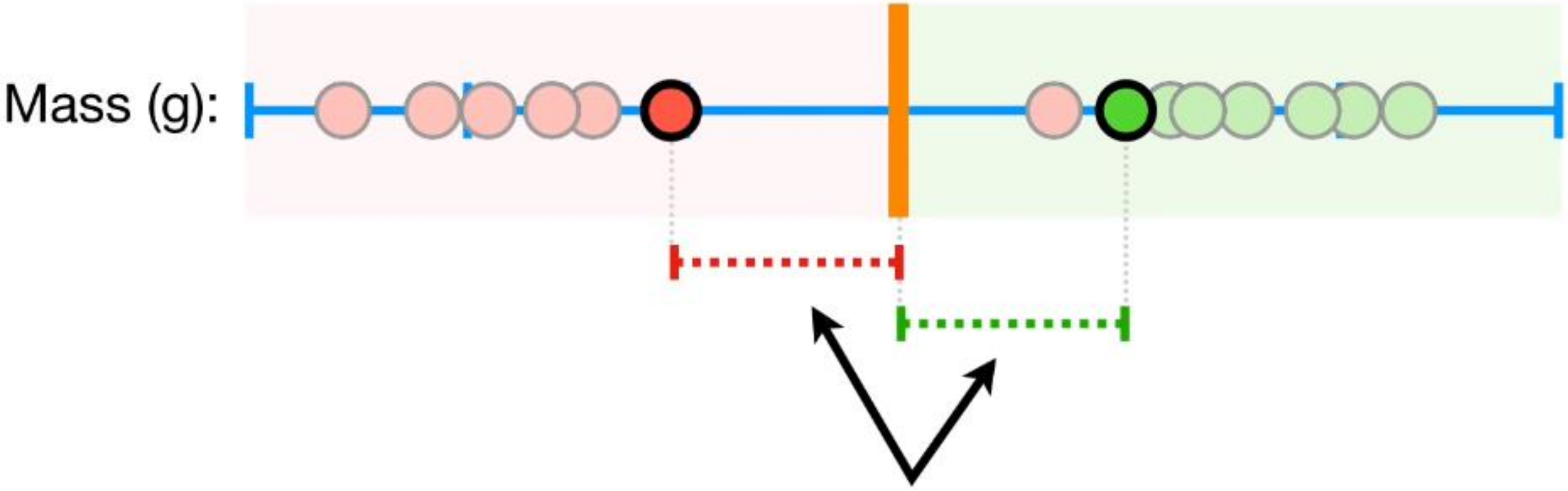
Support Vector Machines



Support Vector Machines

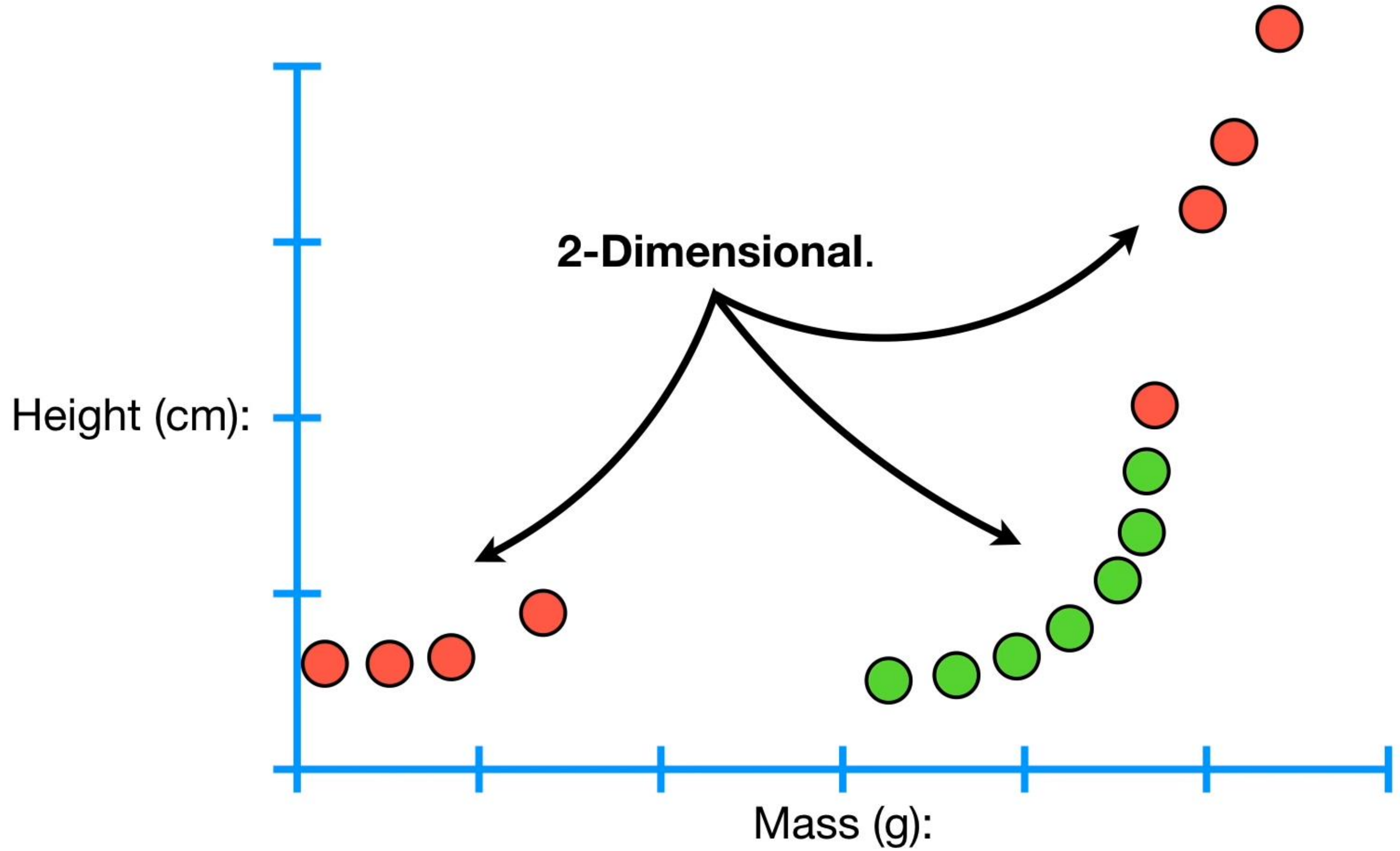


Support Vector Machines

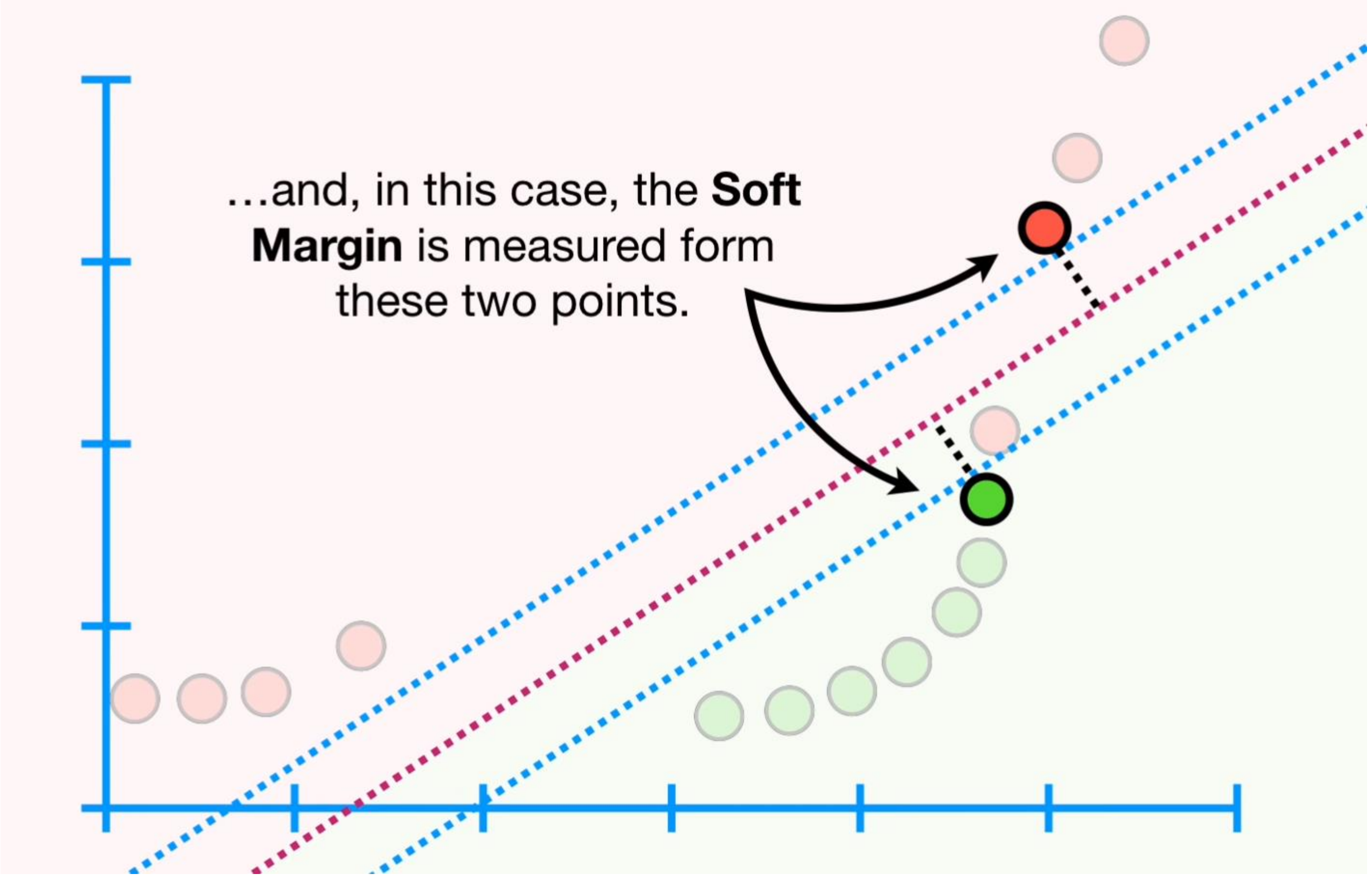


Soft margin

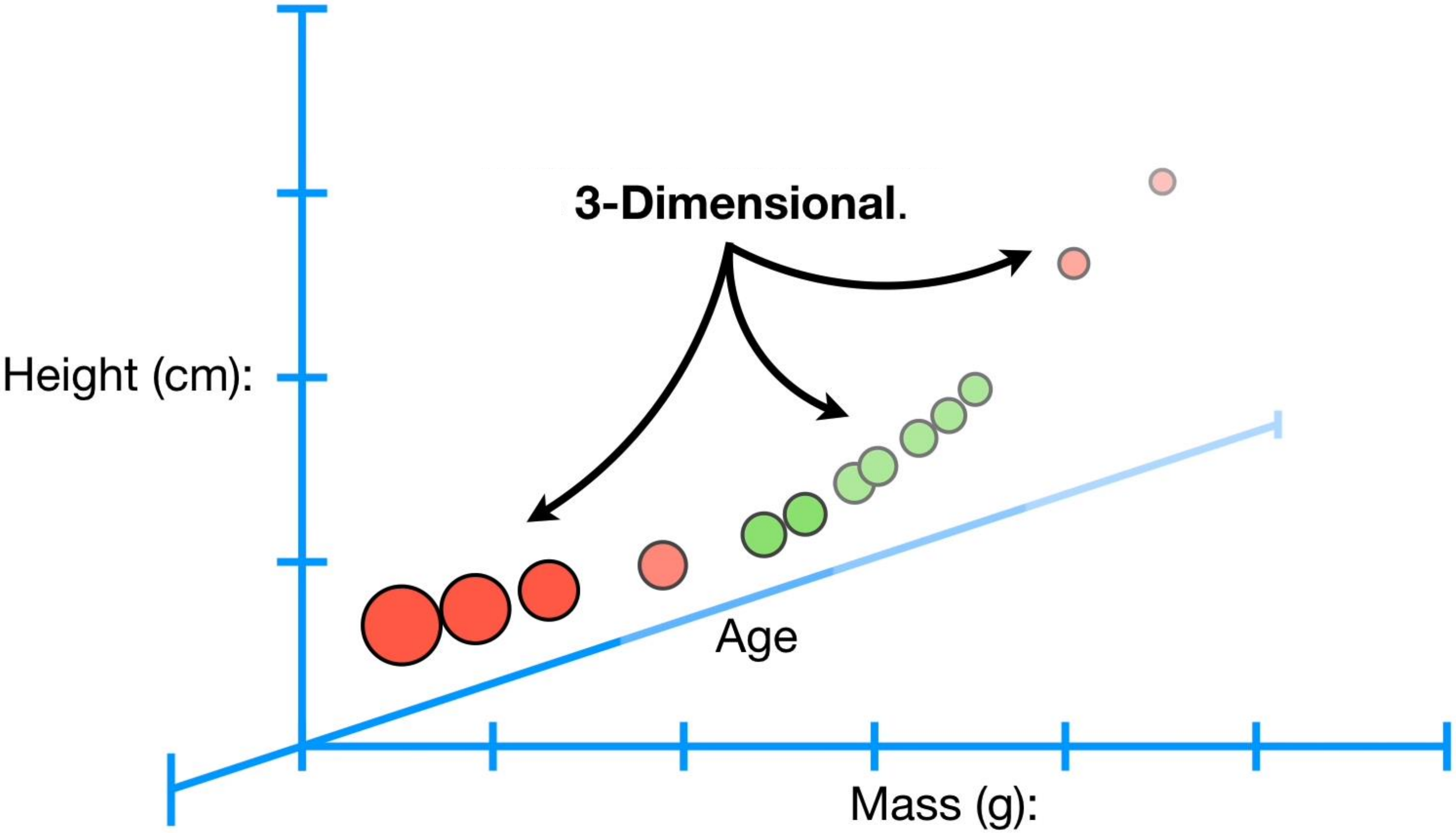
Support Vector Machines



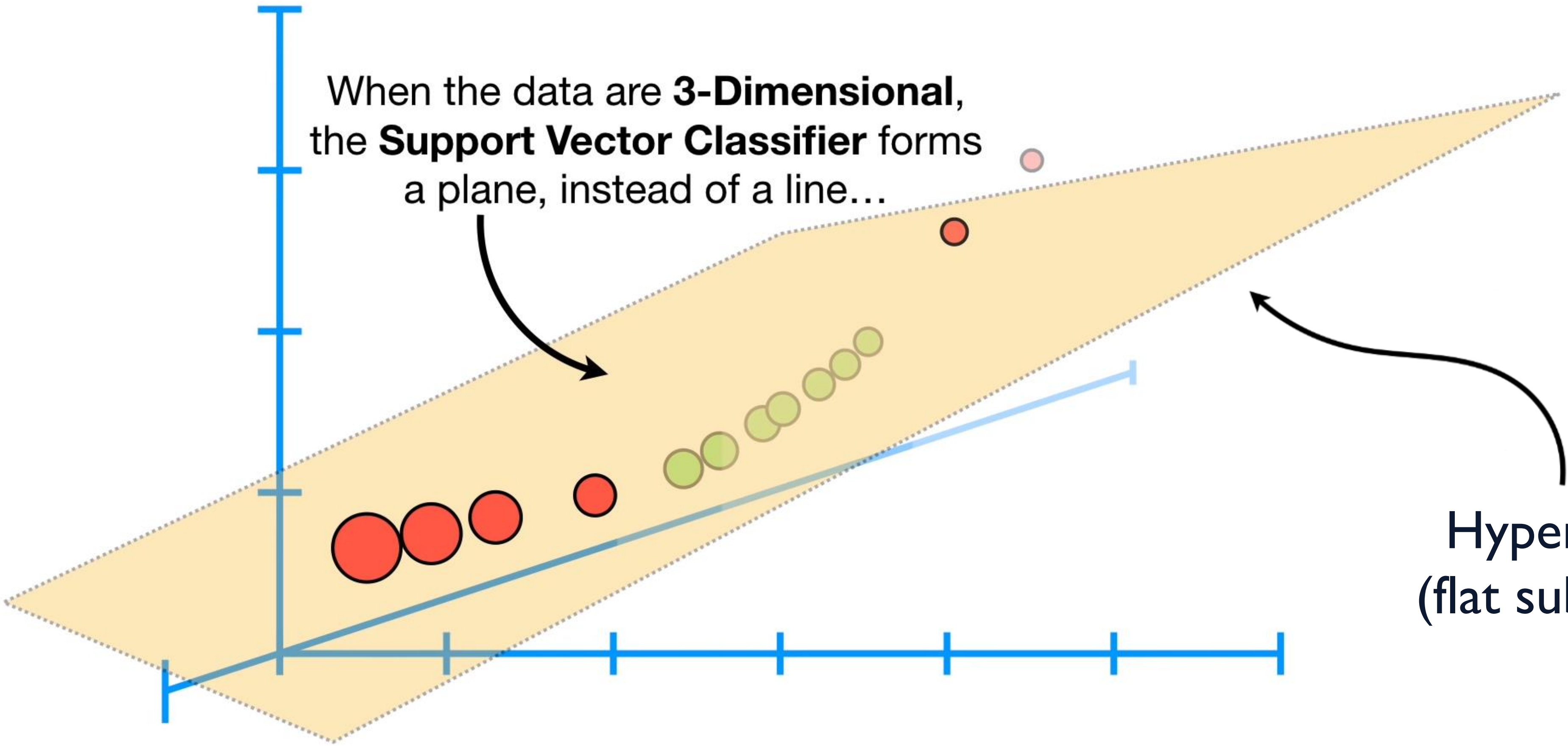
Support Vector Machines



Support Vector Machines

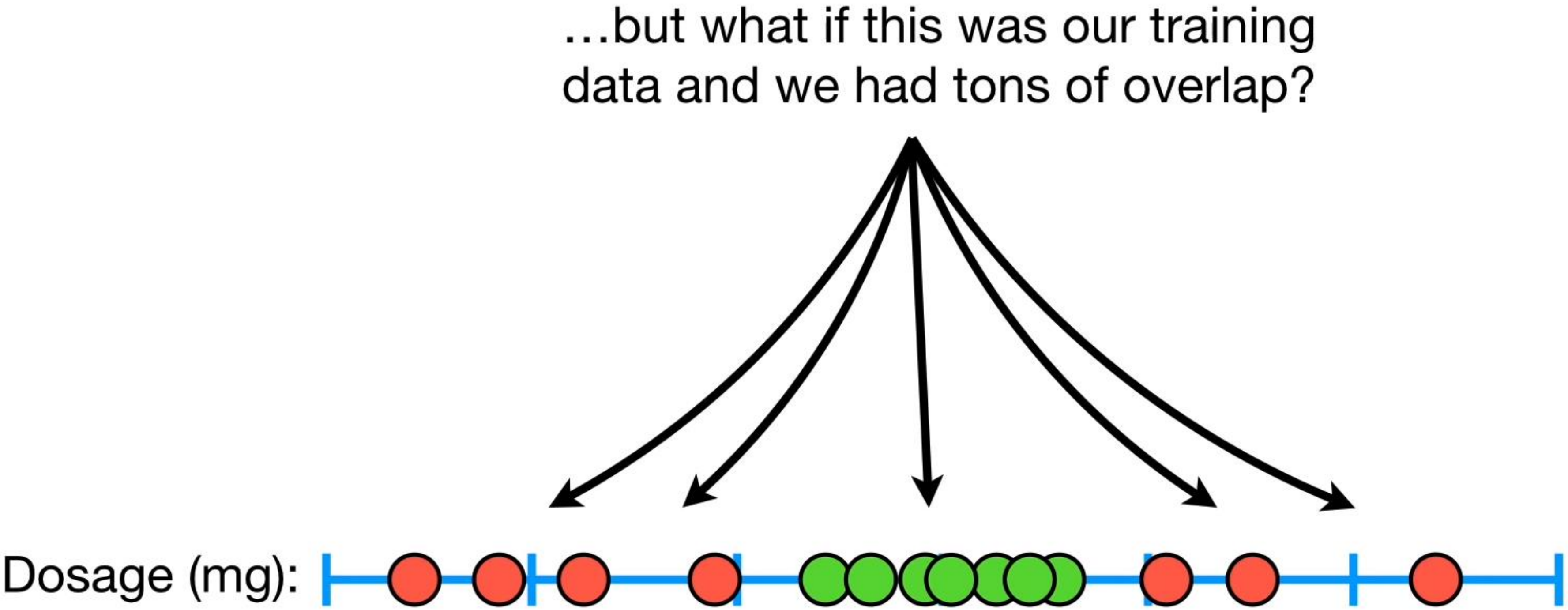


Support Vector Machines

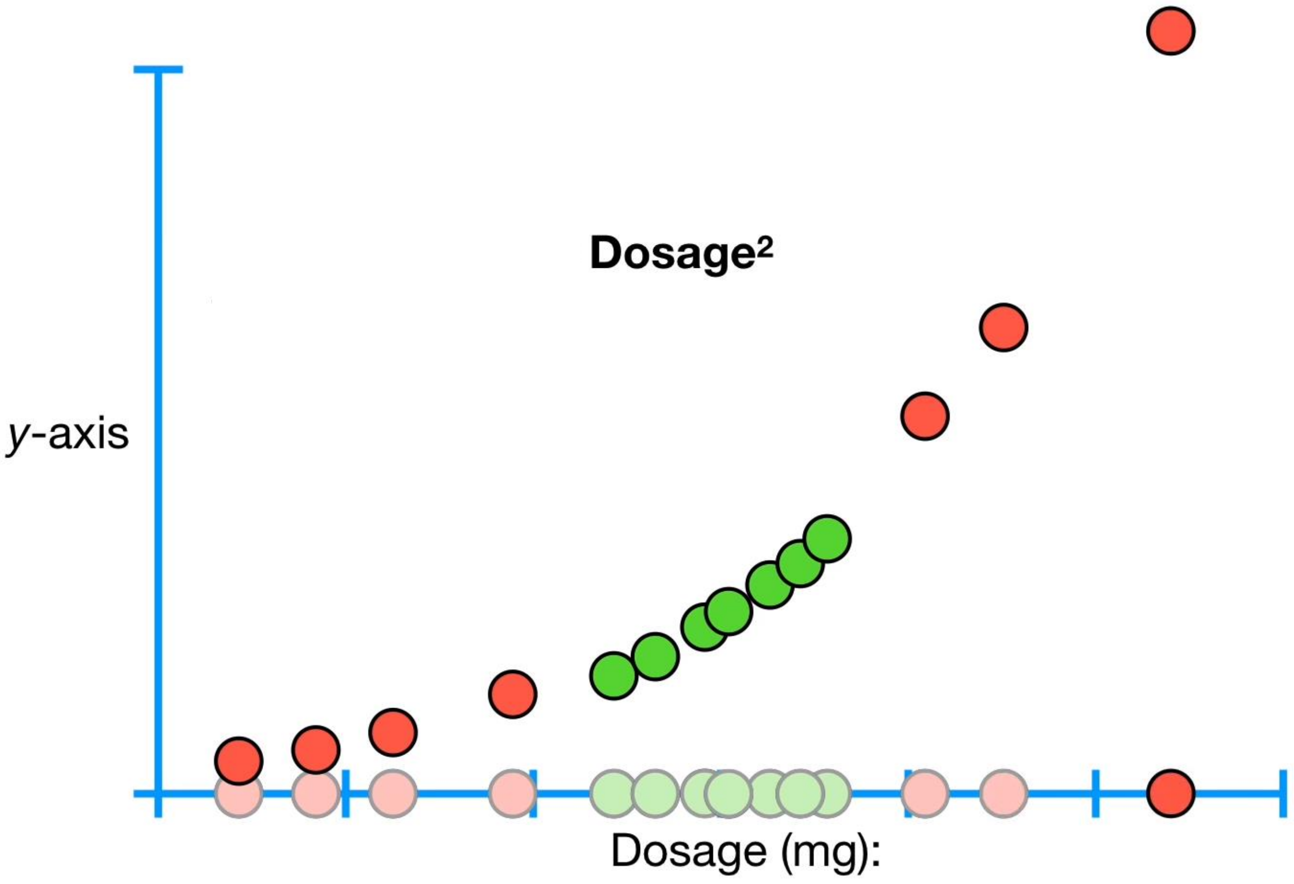


Hyperplane
(flat subspace)

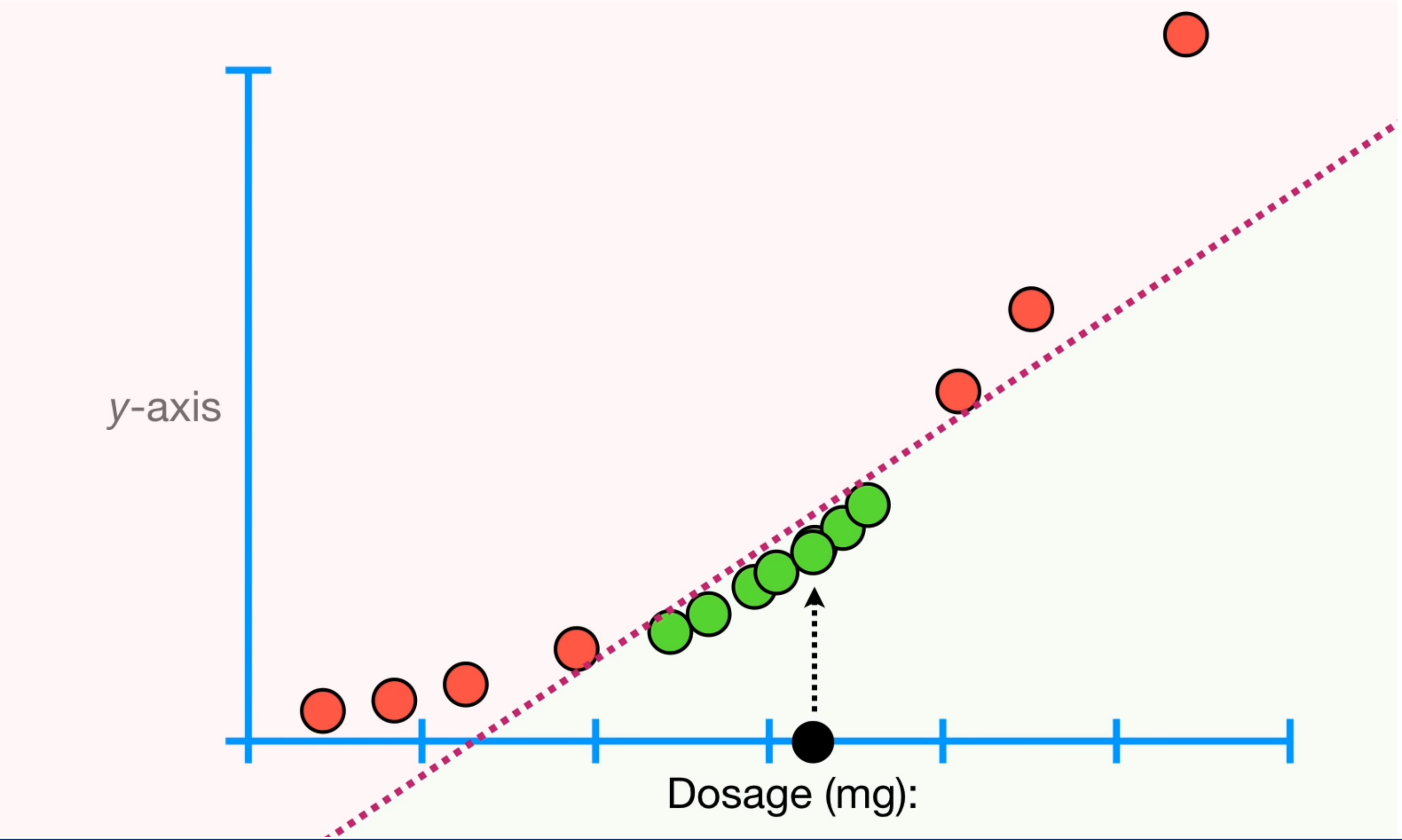
Support Vector Machines



Support Vector Machines

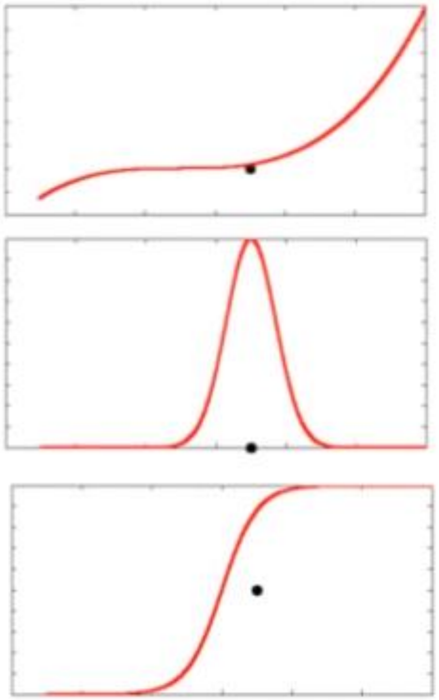


Support Vector Machines



Kernal Functions in SVMs

Some commonly used kernel functions & their shape:



Polynomial $K(a,b) = (1 + \sum_j a_j b_j)^d$

Radial Basis Functions

$$K(a,b) = \exp(-(a-b)^2/2\sigma^2)$$

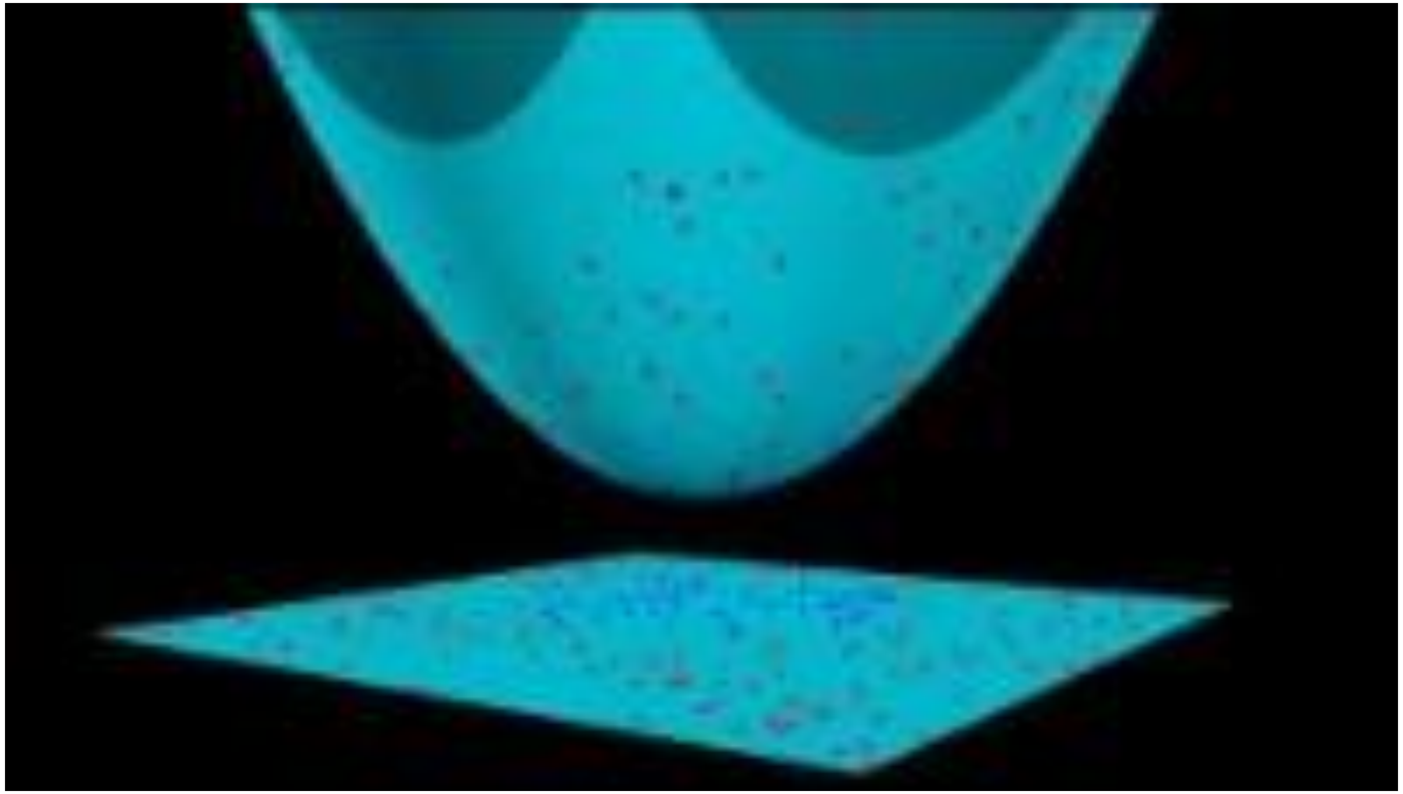
Saturating, sigmoid-like:

$$K(a,b) = \tanh(ca^T b + h)$$

Many for special data types:

- String similarity for text, genetics

Polynomial Kernel Function



The Kernel Trick

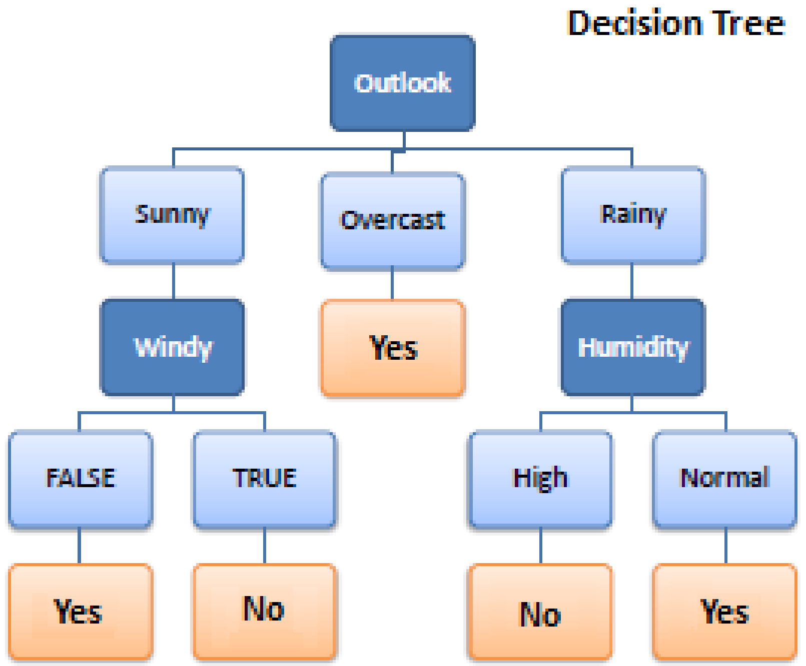
This **trick**, calculating the high-dimensional relationships without actually transforming the data to the higher dimension, is called **The Kernel Trick**.

The Kernel Trick reduces the amount of computation required for **Support Vector Machines** by avoiding the math that transforms the data from low to high dimensions…

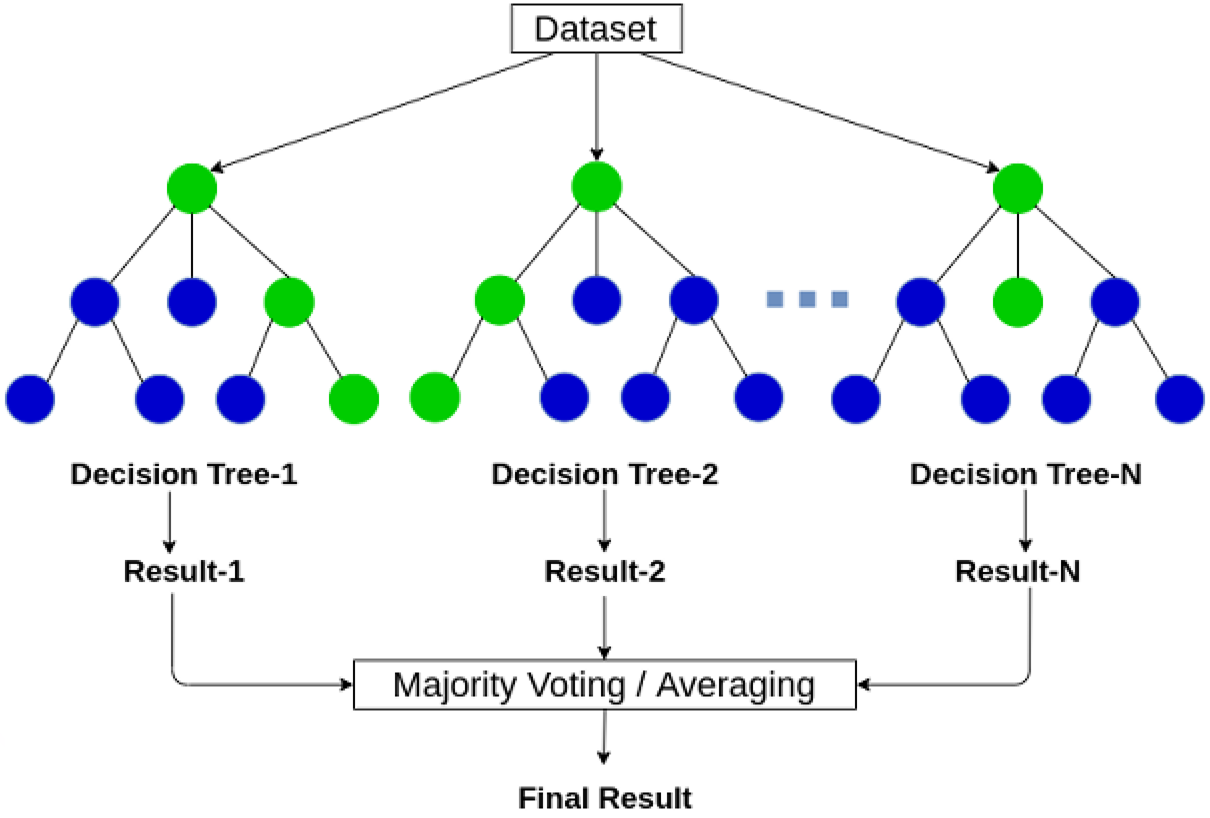
Decision Trees

Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

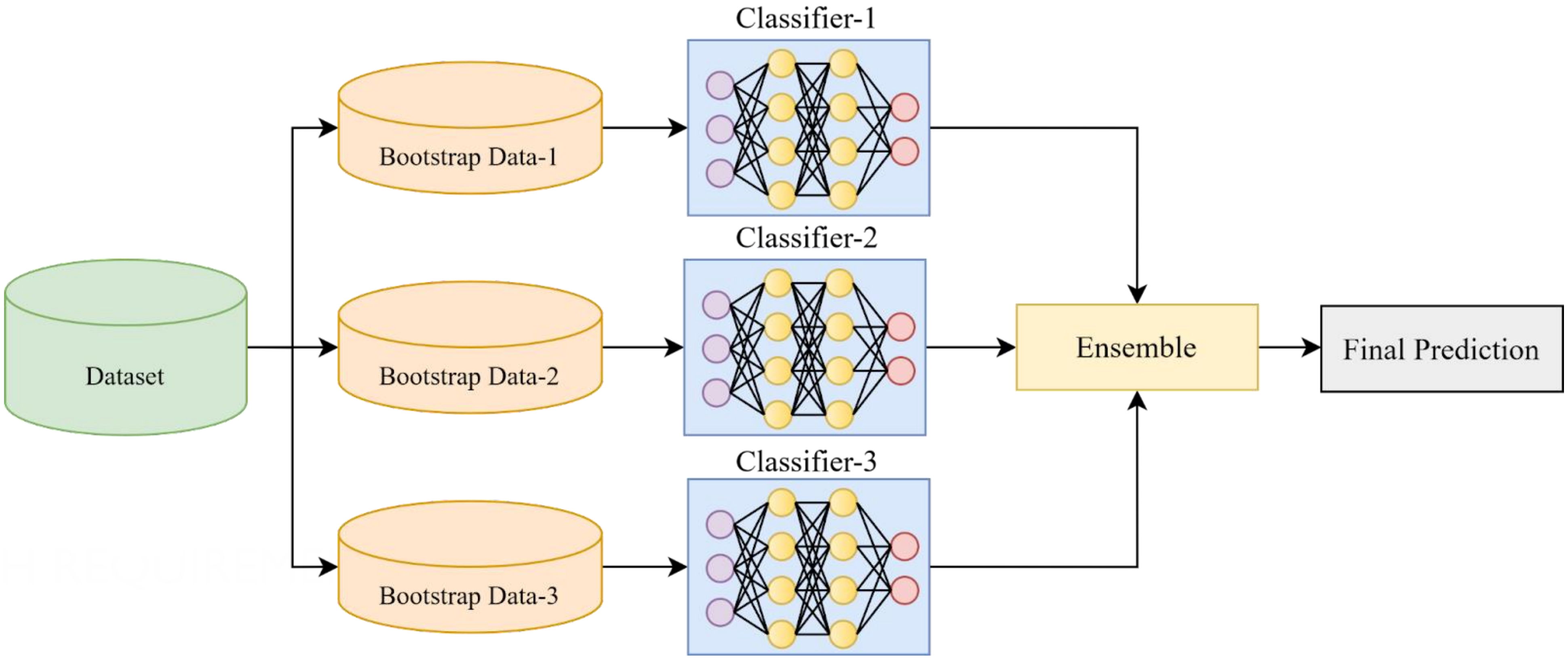
Predictors: Outlook, Temp., Humidity, Windy. Target: Play Golf.



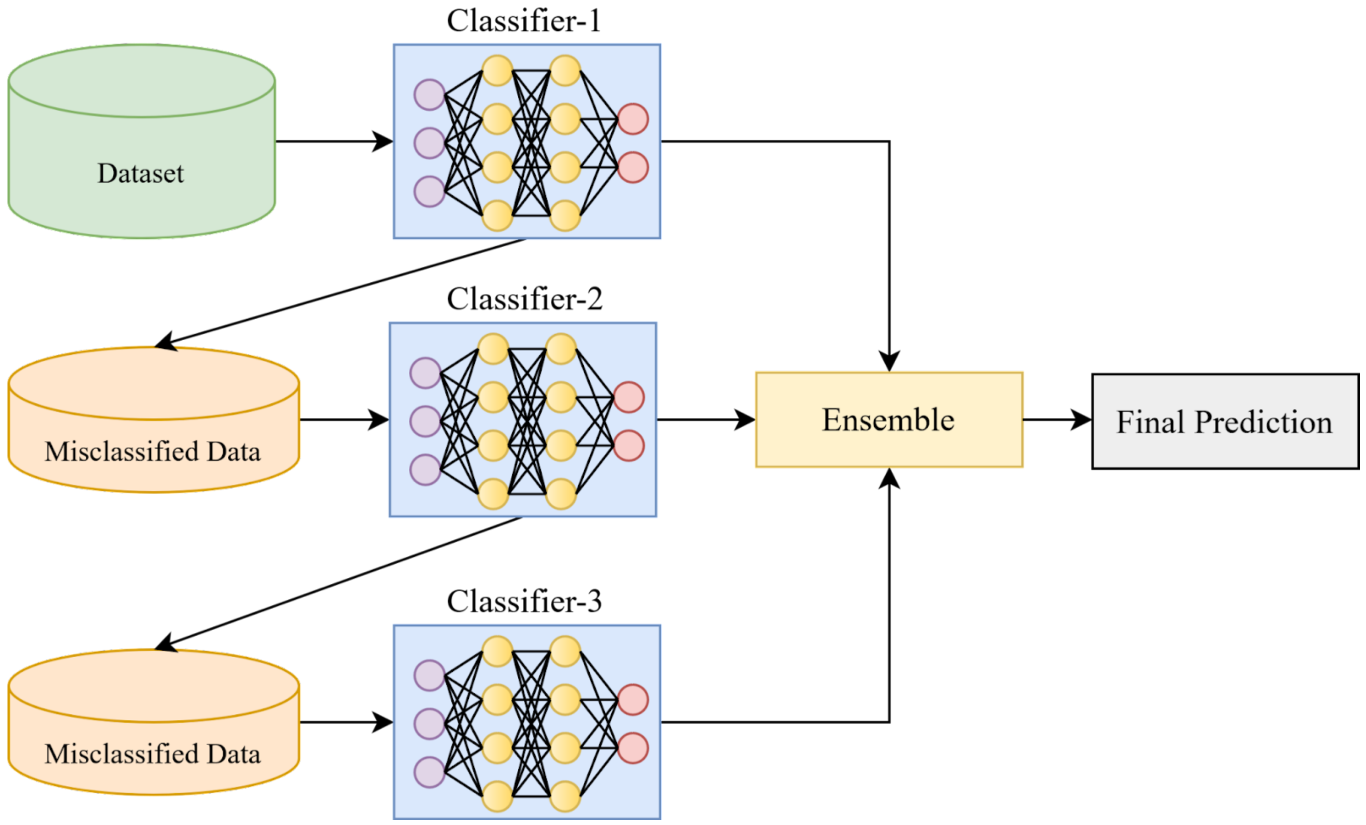
Random Forest



Ensemble Learning



Ensemble Learning



Evaluation Metrics for Classification Tasks

Confusion Matrix

PREDICTED VALUE

		Positive	Negative
ACTUAL VALUE	Positive	**TP**	**FN**
	Negative	**FP**	**TN**

- True Positive (TP) : Observation is positive, and is predicted to be positive.
- False Negative (FN) : Observation is positive, but is predicted negative.
- True Negative (TN) : Observation is negative, and is predicted to be negative.
- False Positive (FP) : Observation is negative, but is predicted positive.

Evaluation Metrics for Classification Tasks

$$Accuracy = \frac{TrueNegatives + TruePositive}{TruePositive + FalsePositive + TrueNegative + FalseNegative}$$

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

The proportion of True predictions on the total of predictions made.

Evaluation Metrics for Classification Tasks

$$\text{Precision} = \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Positive}}$$

$$= \frac{\textit{True Positive}}{\textit{Total Predicted Positive}}$$

The proportion of True Positives among the total of Positive Predictions.

Evaluation Metrics for Classification Tasks

$$\text{Recall} = \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Negative}}$$

$$= \frac{\textit{True Positive}}{\textit{Total Actual Positive}}$$

The proportion of True Positive among the total number of actual Positives.

Evaluation Metrics for Classification Tasks

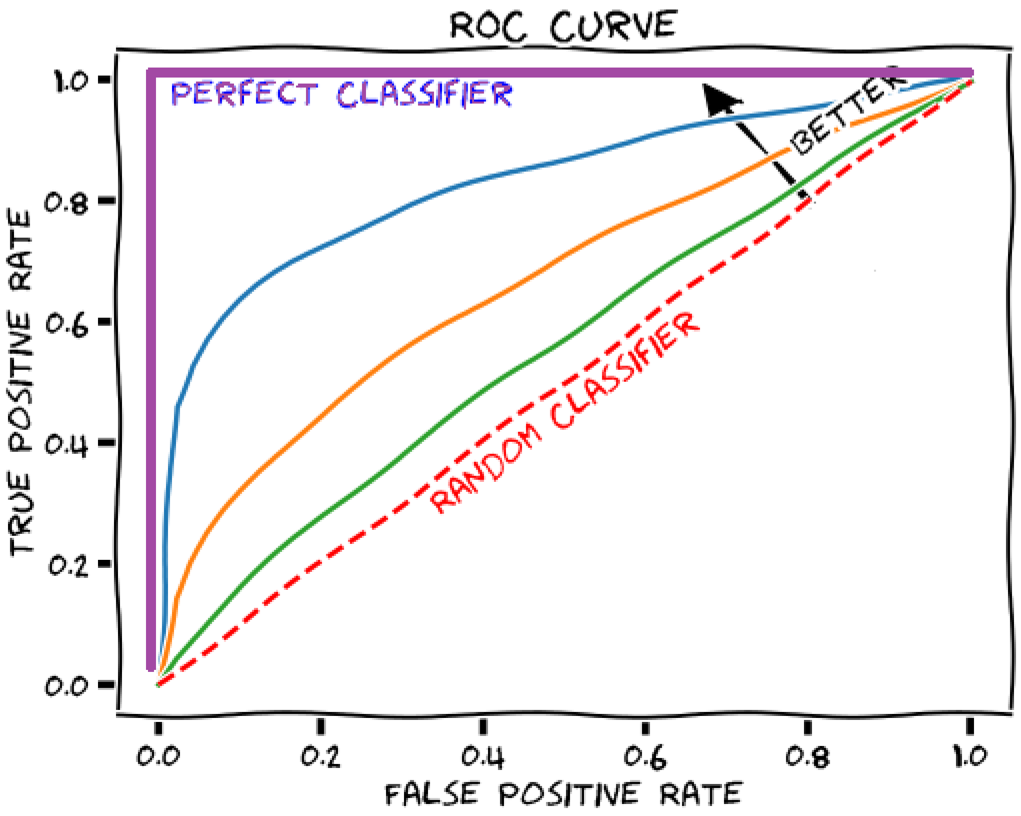
$$\text{F1 Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

The harmonic mean of Precision and Recall.

Evaluation Metrics for Classification Tasks

Metric Name	Metric Forumla	Code	When to use
Accuracy	**Accuracy** $= \frac{tp + tn}{tp + tn + fp + fn}$	`tf.keras.metrics.Accuracy()` or `sklearn.metrics.accuracy_score()`	Default metric for classification problems. Not the best for imbalanced classes.
Precision	**Precision** $= \frac{tp}{tp + fp}$	`tf.keras.metrics.Precision()` or `sklearn.metrics.precision_score()`	Higher precision leads to less false positives.
Recall	**Recall** $= \frac{tp}{tp + fn}$	`tf.keras.metrics.Recall()` or `sklearn.metrics.recall_score()`	Higher recall leads to less false negatives.
F1-score	**F1-score** $= 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$	`sklearn.metrics.f1_score()`	Combination of precision and recall, usually a good overall metric for a classification model.
Confusion matrix	NA	`Custom function` or `sklearn.metrics.confusion_matrix()`	When comparing predictions to truth labels to see where model gets confused. Can be hard to use with large numbers of classes.

Evaluation Metrics for Classification Tasks



We're Almost Finished

Let's code!

TECH REQUIREMENTS

Final Project

- Document each of the following steps in your notebook.
- Choose a dataset of your choice from Kaggle.com.
- Preprocess the data acquired (clean it, normalize the data if needed, do feature engineering, etc…) and prepare it for the ML model. Explain every single step.
- Make a comparison study between 3 types of models on the dataset used. Use different hyperparameters and features and try to obtain the highest results possible.
- Get the results and analyze why certain models performed better that others.
- Submit a .pynb file with data used.

Deadline: 1/6, 6 PM.

Good luck!

Final Project

Use our colleague “Basel”’s tool to help you in your project:
https://clickml.streamlit.app/

TECH REQUIREMENTS

Final Project

I would also love to have your feedback of the course for the purpose of future improvements!

THANK YOU

WISHING YOU ALL THE BEST IN YOUR UPCOMING ENDEAVORS!

SABBAGH@IEEE.ORG

CONNECT ON LINKEDIN