Lecture 08 – Classification Models

Prof. André Gustavo Hochuli

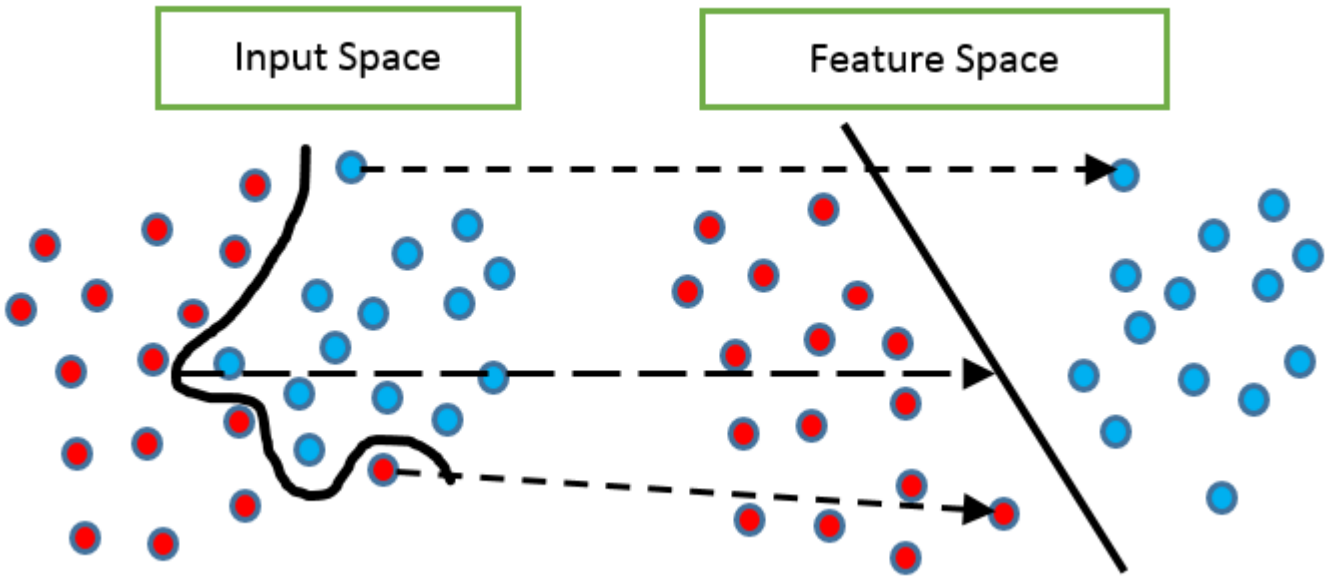
gustavo.hochuli@pucpr.br
aghochuli@ppgia.pucpr.br

Topics

- Discussion of Lecture #07
  - Image Descriptors
- Classification Models
  - K-NN, Logistic Regression, Decision Trees Naïve Bayes, SVM and MLP
- Evaluation Metrics
  - Accuracy, Precision, Recall and F1-Score
- Practice

Problem

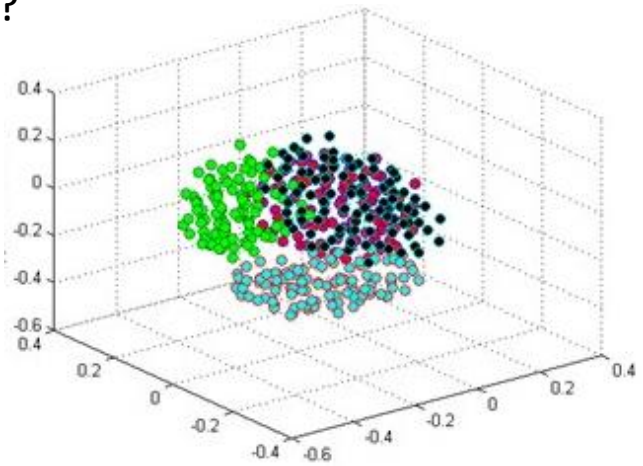
- So far, we extracted features from data to compute the feature space



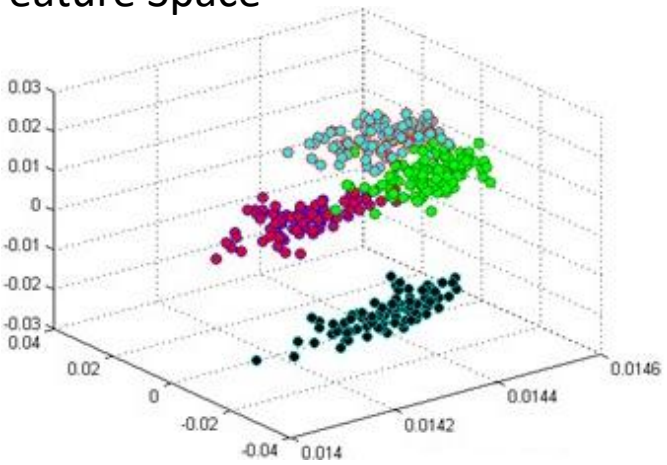
Problem

- How discriminative features are?

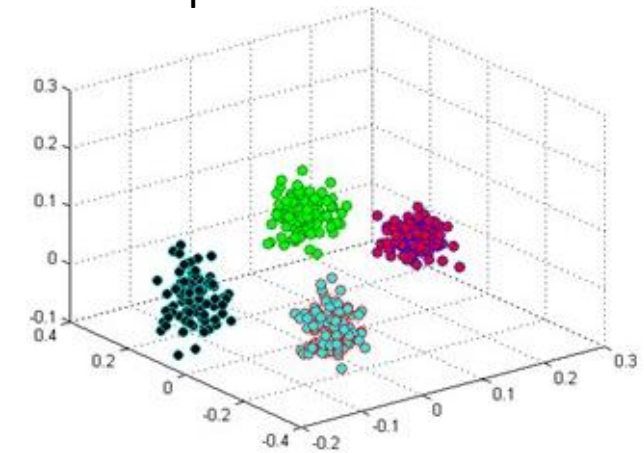
Input Space



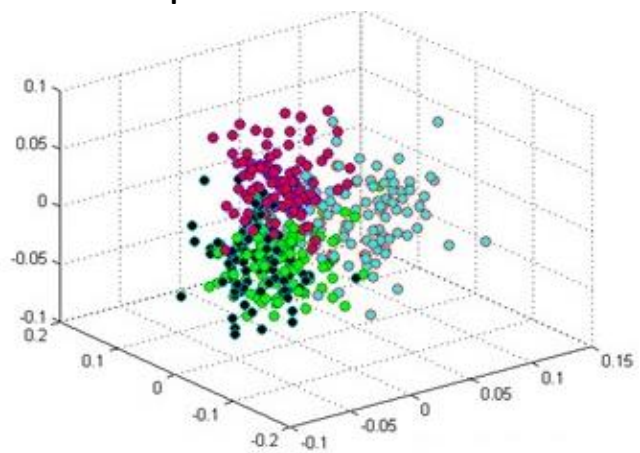
Feature Space'



Feature Space''

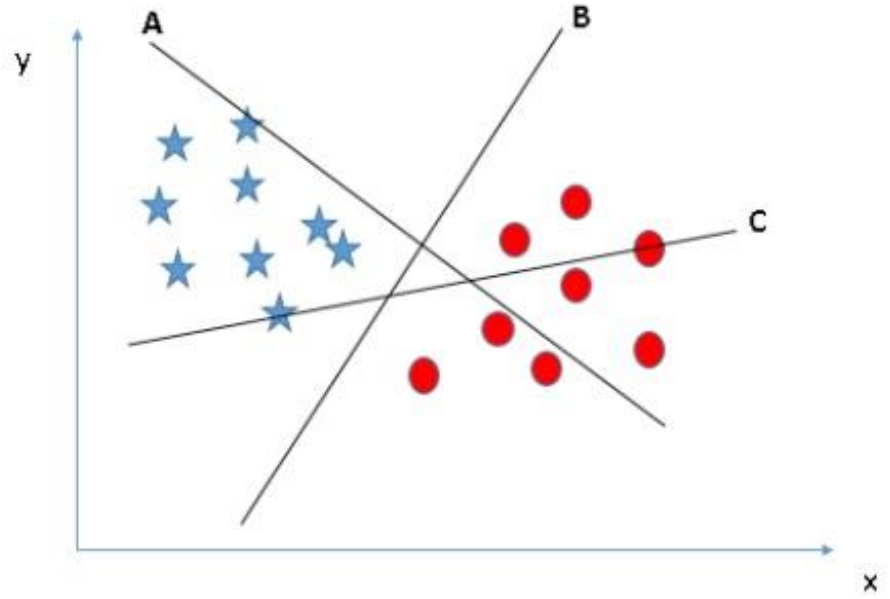


Feature Space '''



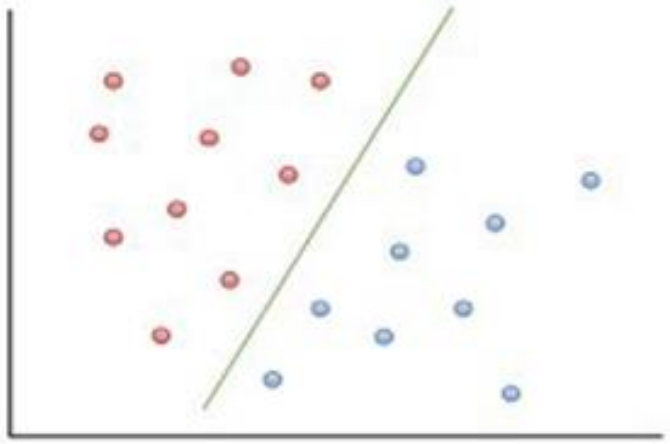
Problem

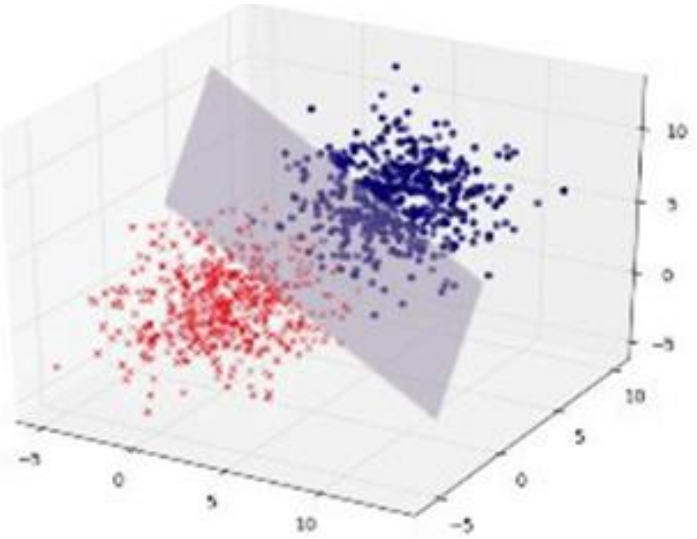
- How to compute the decision boundary?



Problem

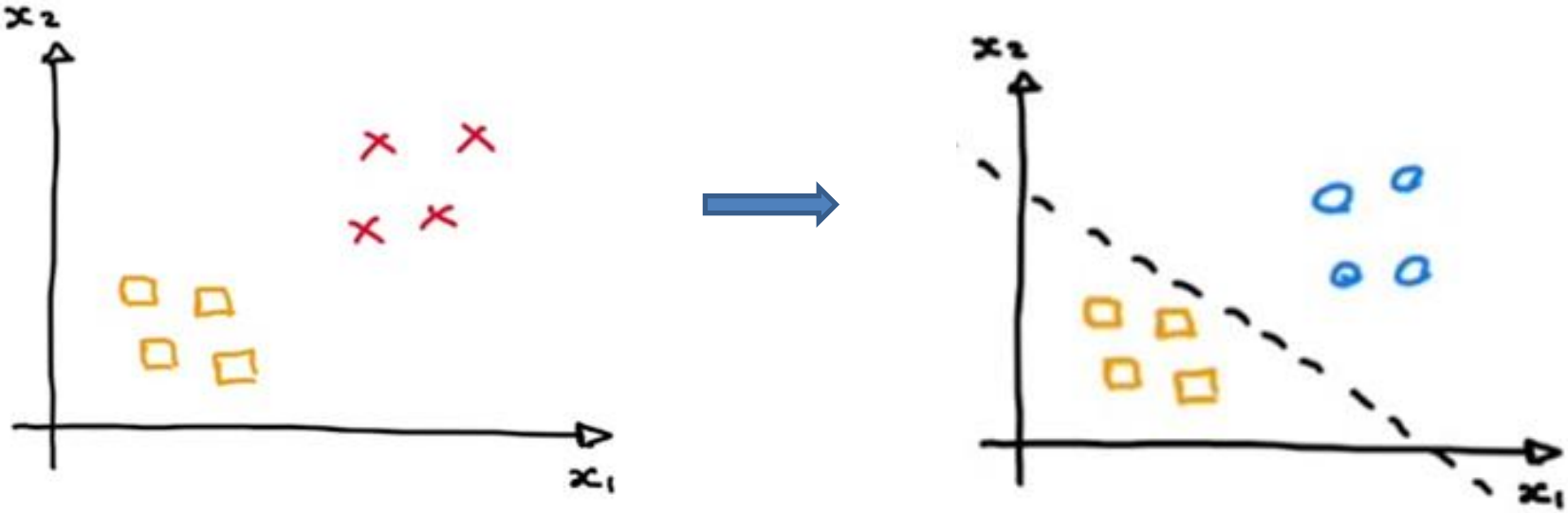
- Hyperplane
  - 2-D, 3-D … N-D (or N-Features)





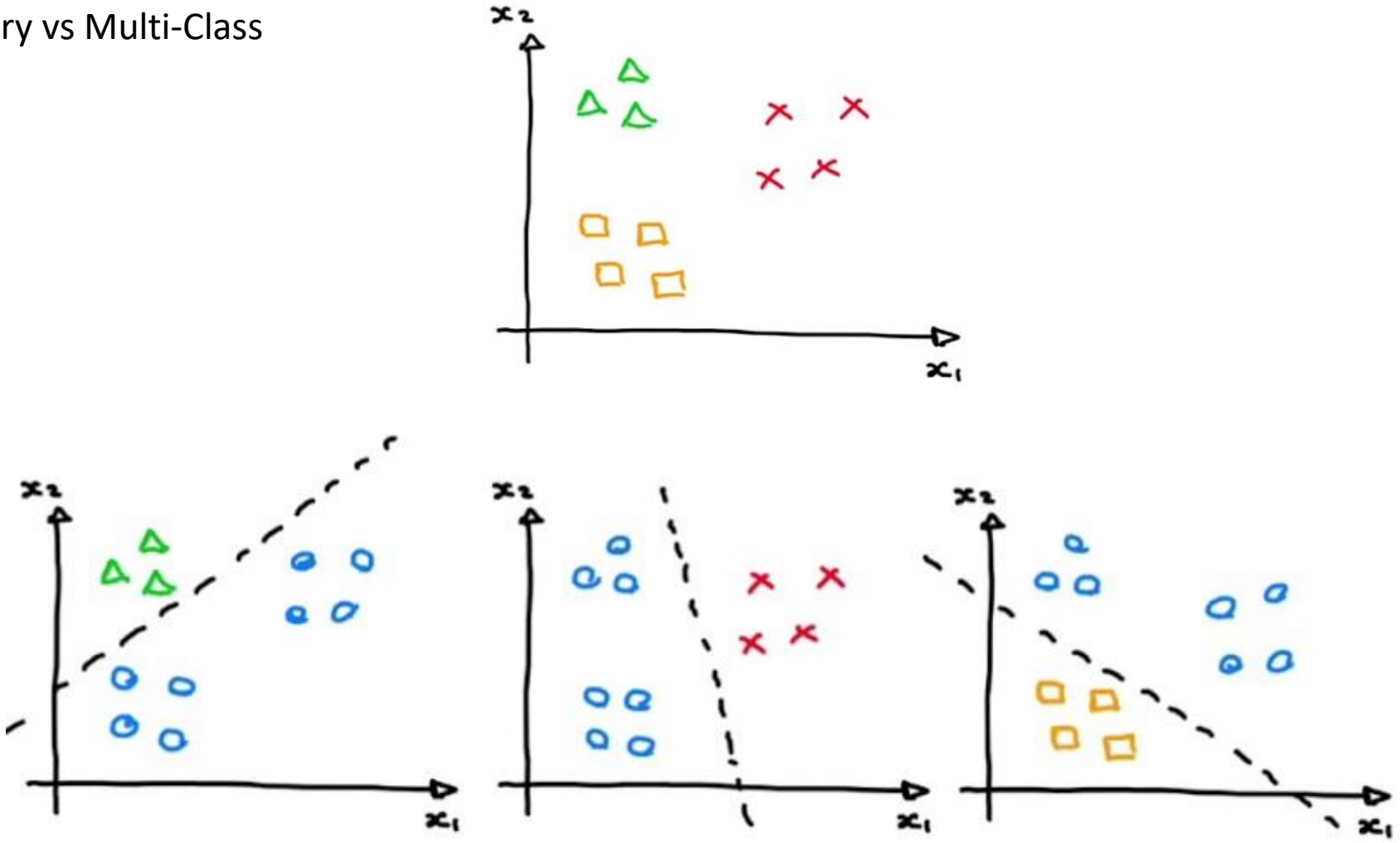
Problem

- Binary Classification vs Multi-Class Classification



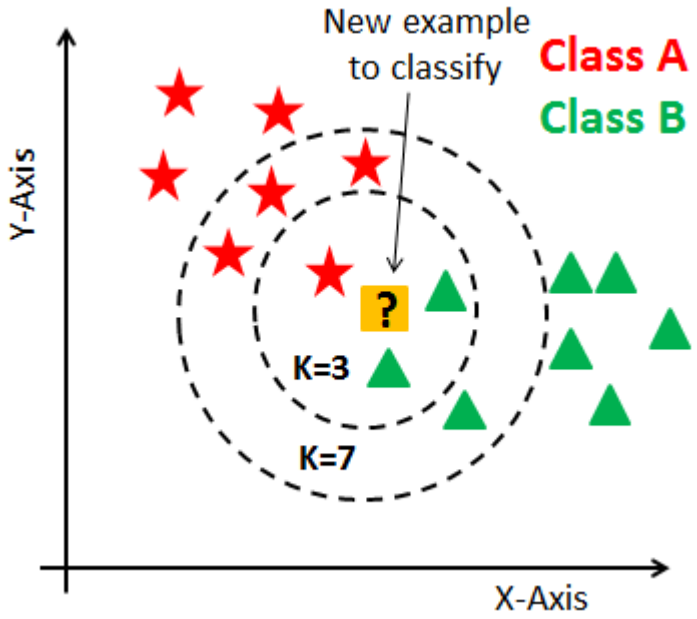
Problem

- Binary vs Multi-Class



Classification Models
KNN

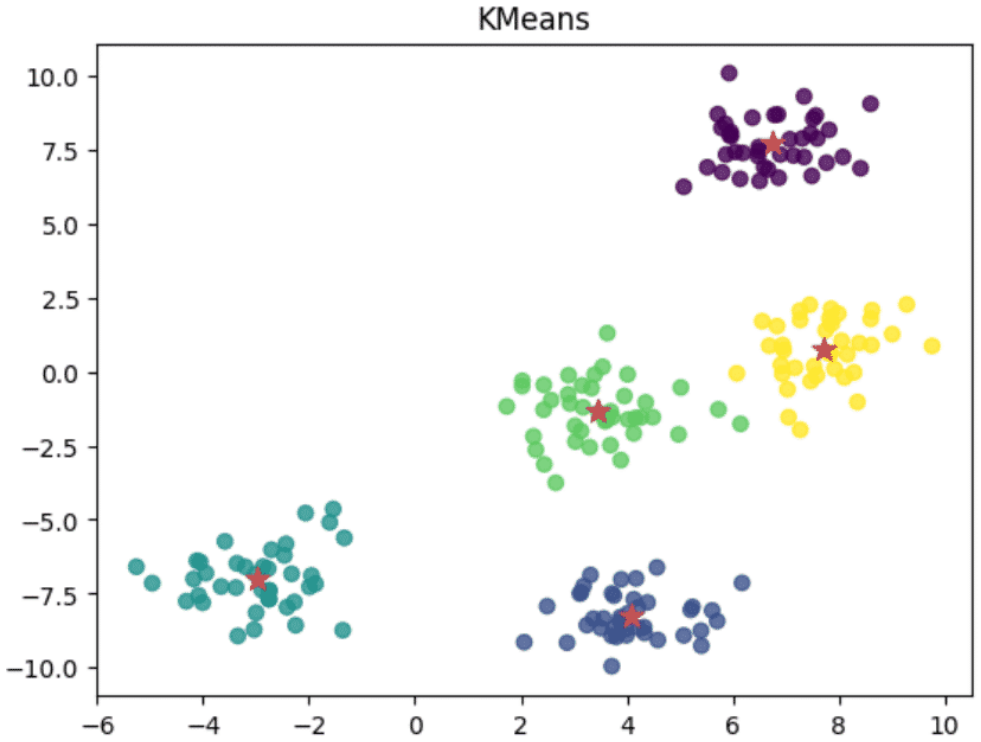
- Computes the similarity in a feature space (Euclidian Distance, Manhattan….)
- The K-Nearest Neighbors determines the class (Majority Vote)
- There is no training step. Compute the distance of the test sample to each training sample



$$d(x,y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2}$$

Classification Models
K-Means

- Computes the distance between k-cluster
- The clusters are defined in training step

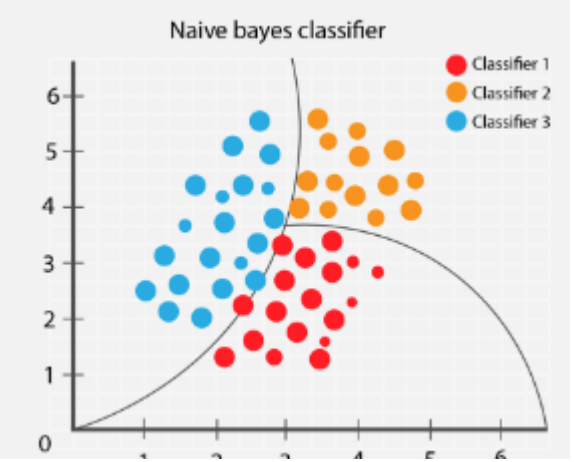


Classification Models
Naïve Bayes

- Bayes Theorem
- *A priori vs Posteriori* Probabilities

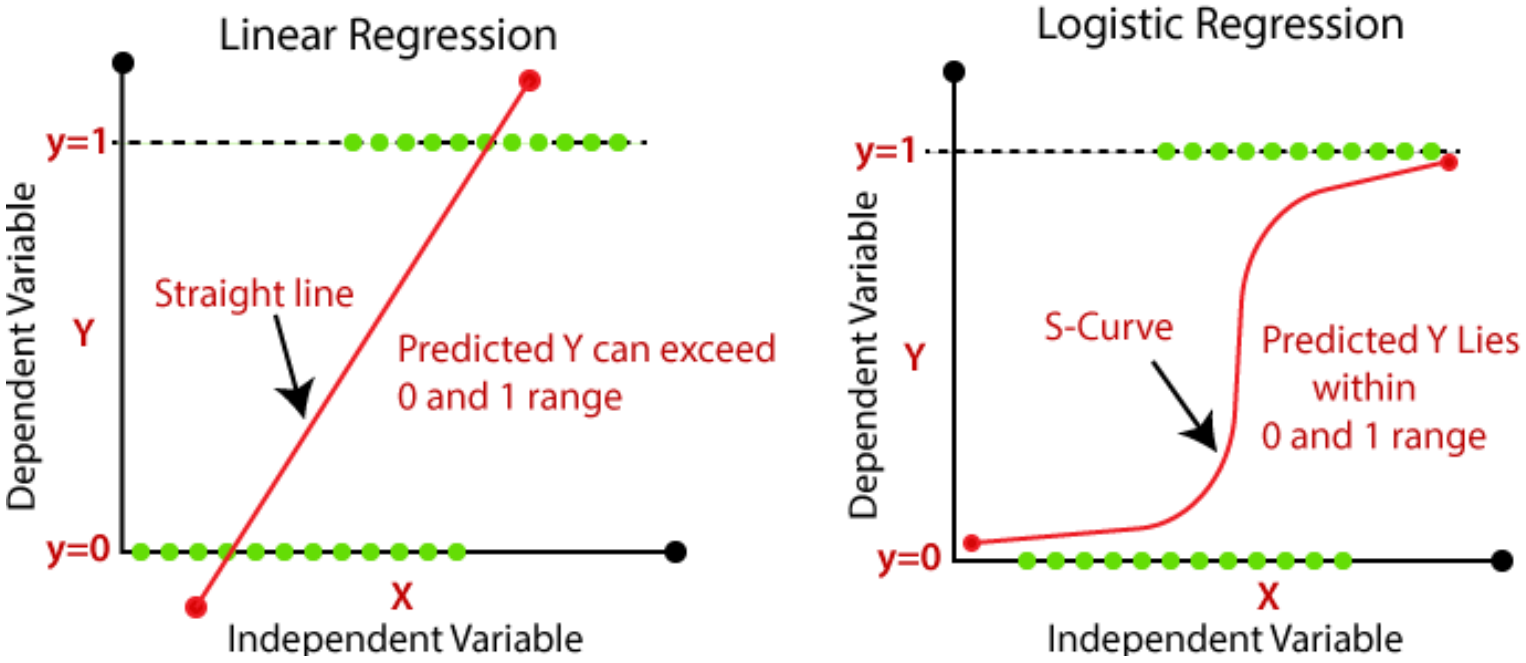
$$P(A|B) = \frac{P(B|A)\ P(A)}{P(B)}$$

$$\text{Posterior} = \frac{\text{prior x likelihood}}{\text{evidence}}$$



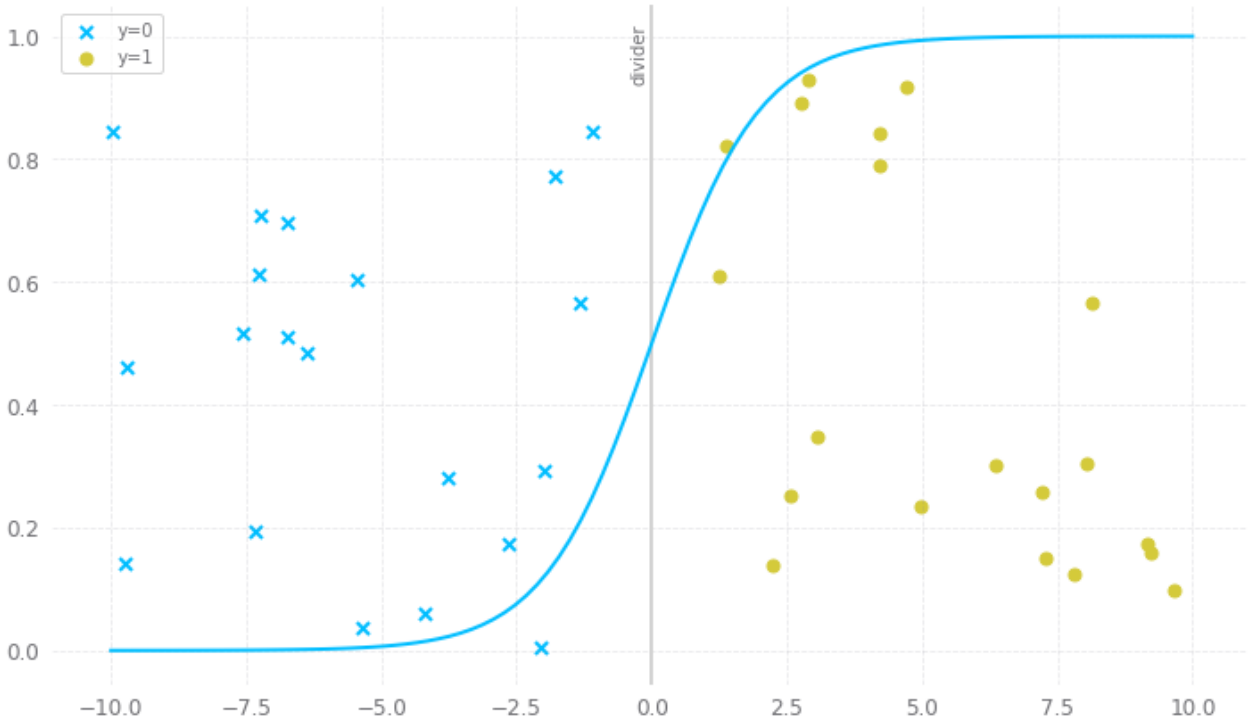
Classification Models
Logistic Regression

- Linear vs Logistic



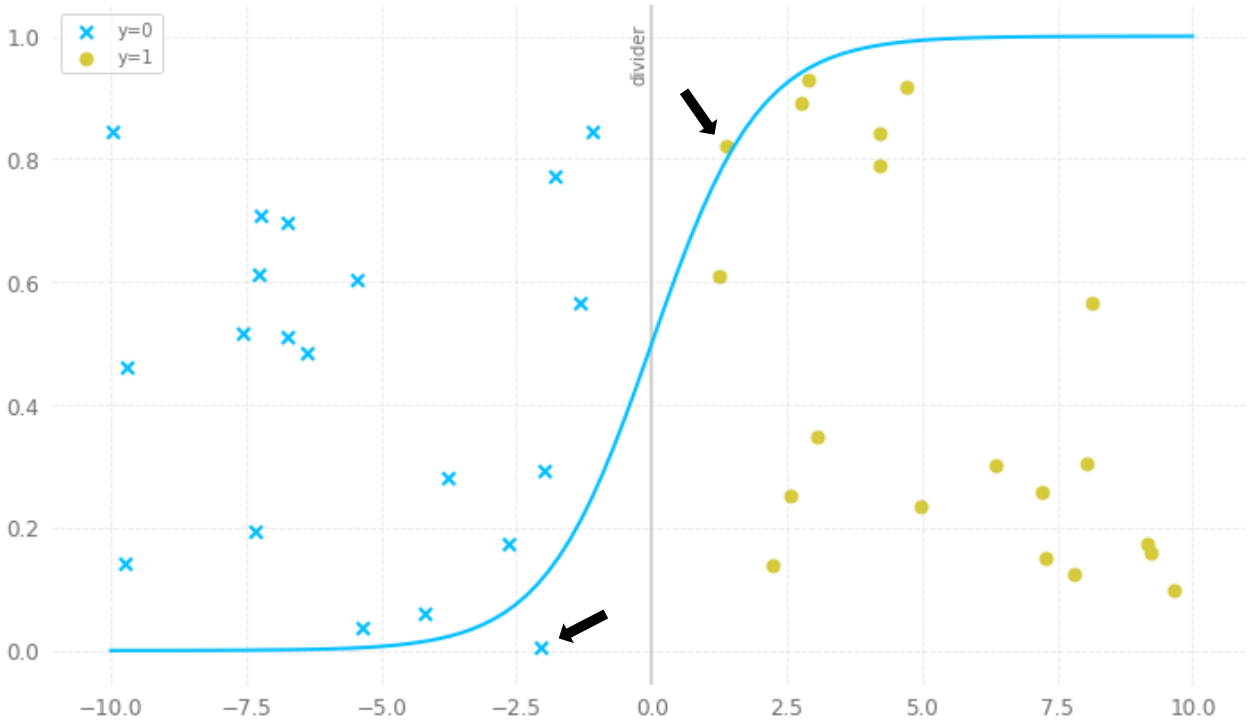
Classification Models
Logistic Regression (LR)

- Logistic Boundary



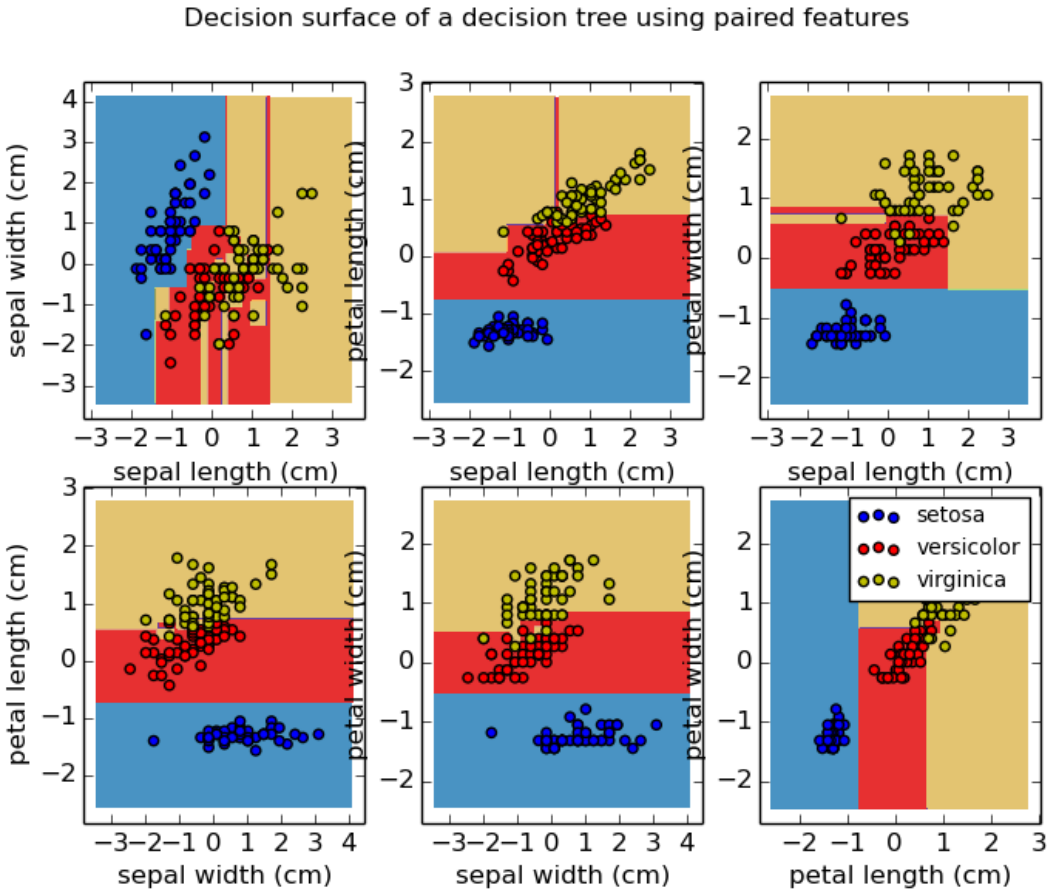
Classification Models
Logistic Regression (LR)

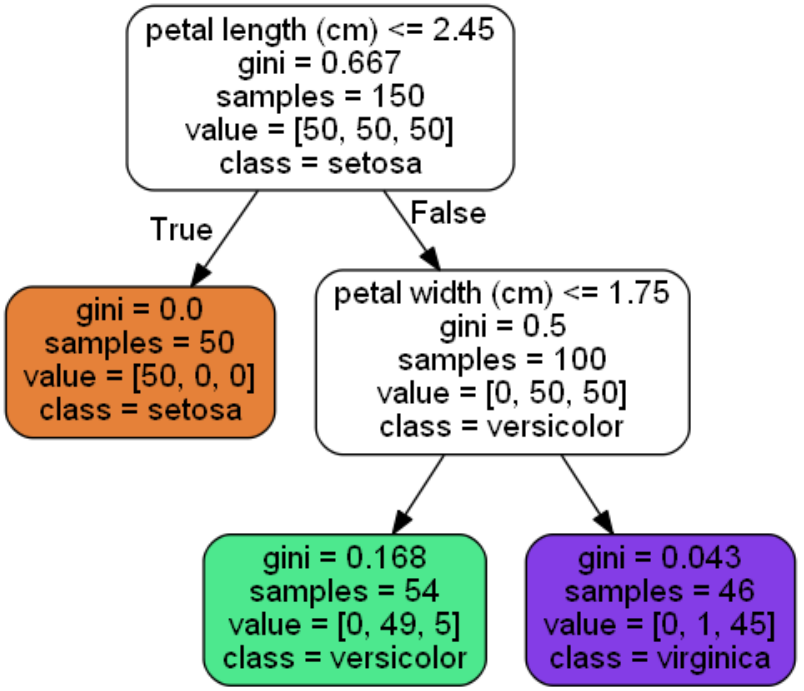
- Logistic Boundary



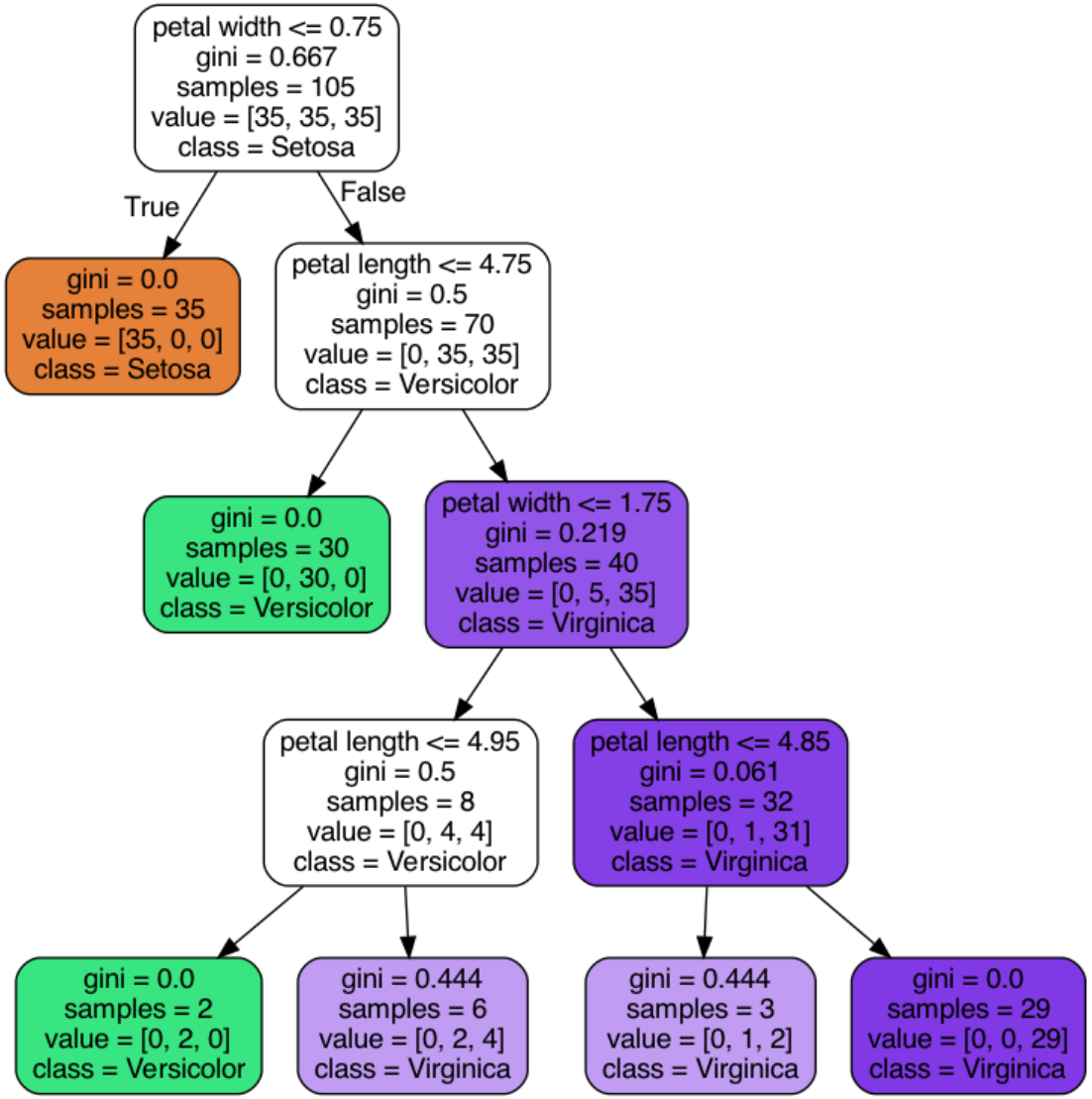
Classification Models
Decision Tree

- Creates decision rules direct from the data features



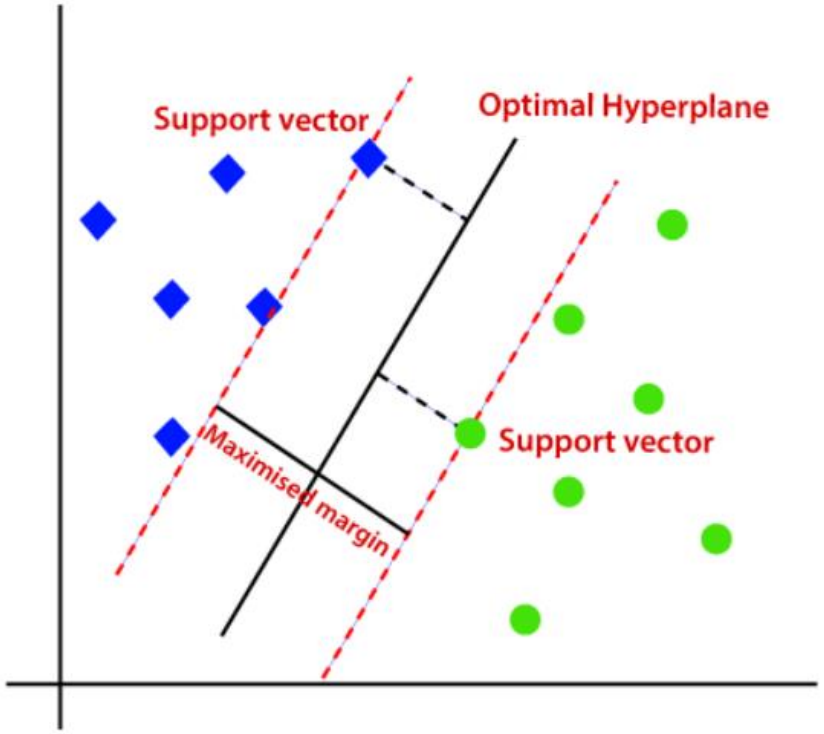


Classification Models
Decision Tree



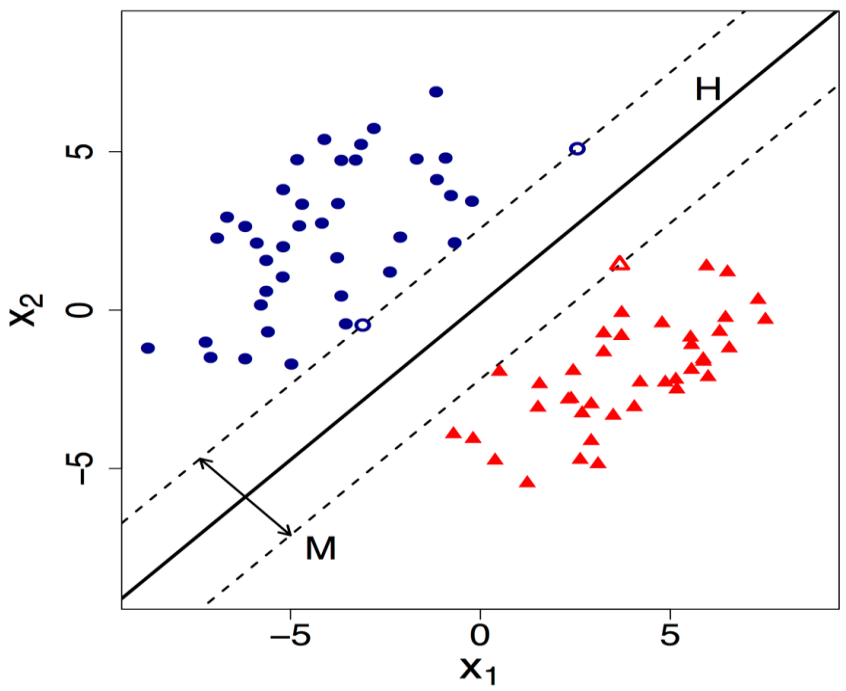
Classification Models
Support Vector Machine (SVM)

- Compute Kernel: Linear, RBF, Poly or Sigmoid
- The clusters are defined in training step

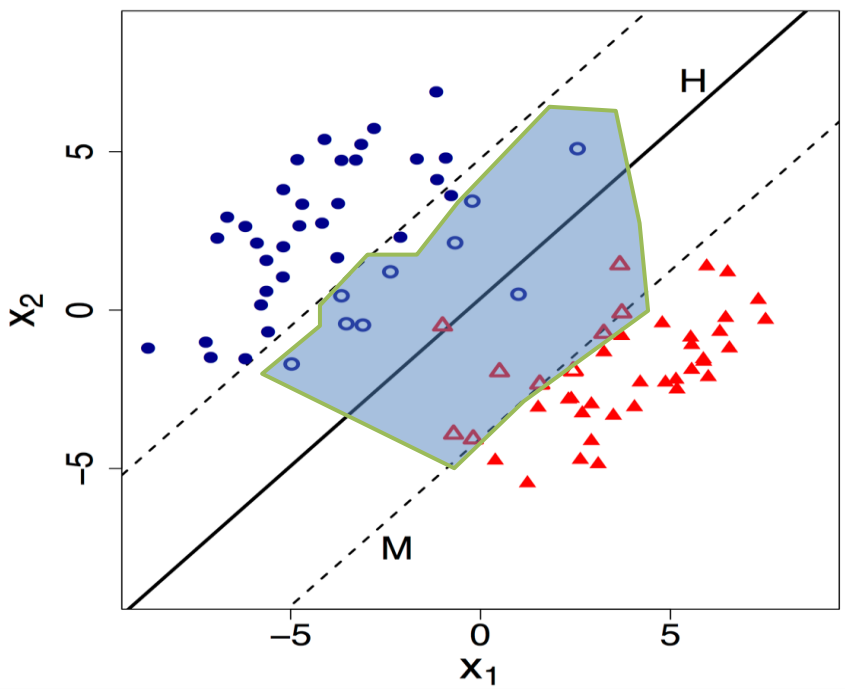


Classification Models
Support Vector Machine (SVM)

Hard Margin

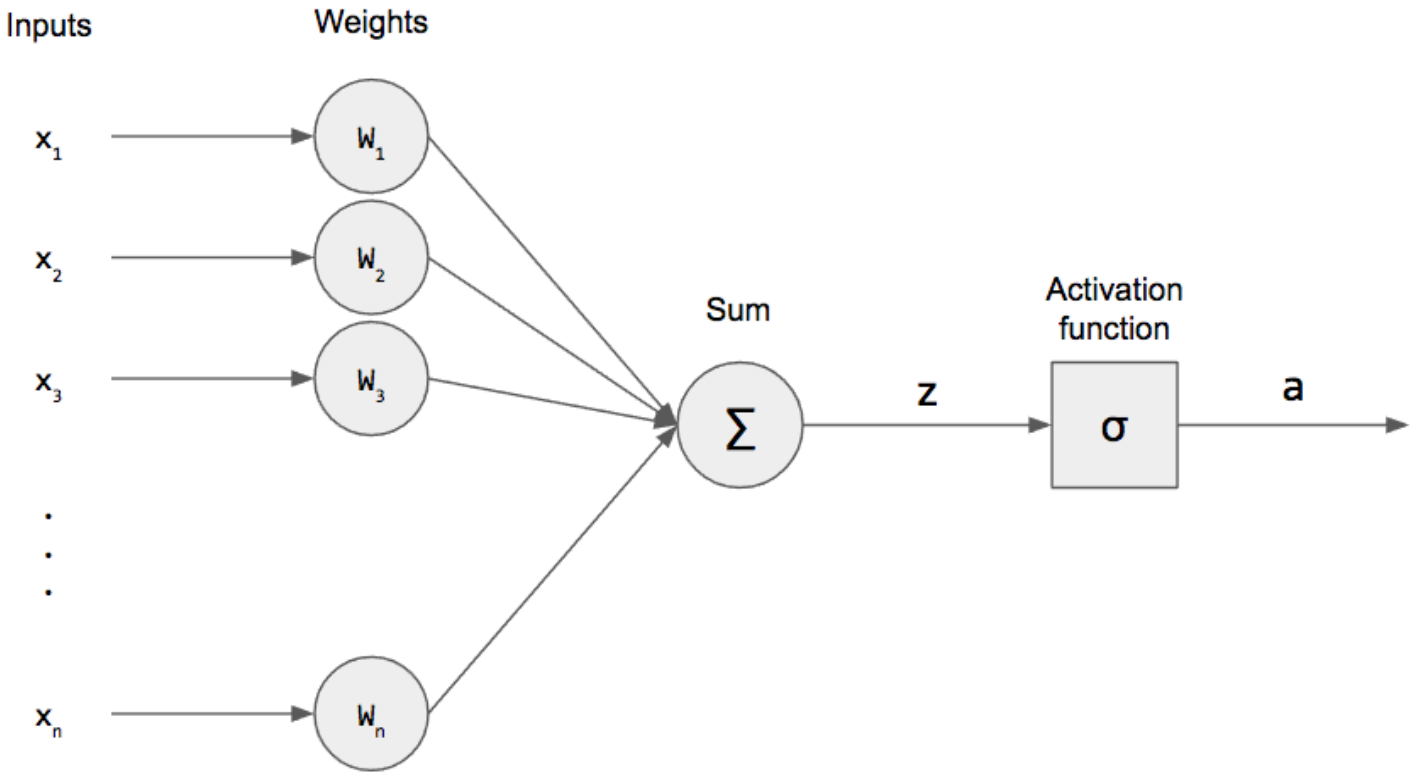


Soft Margin



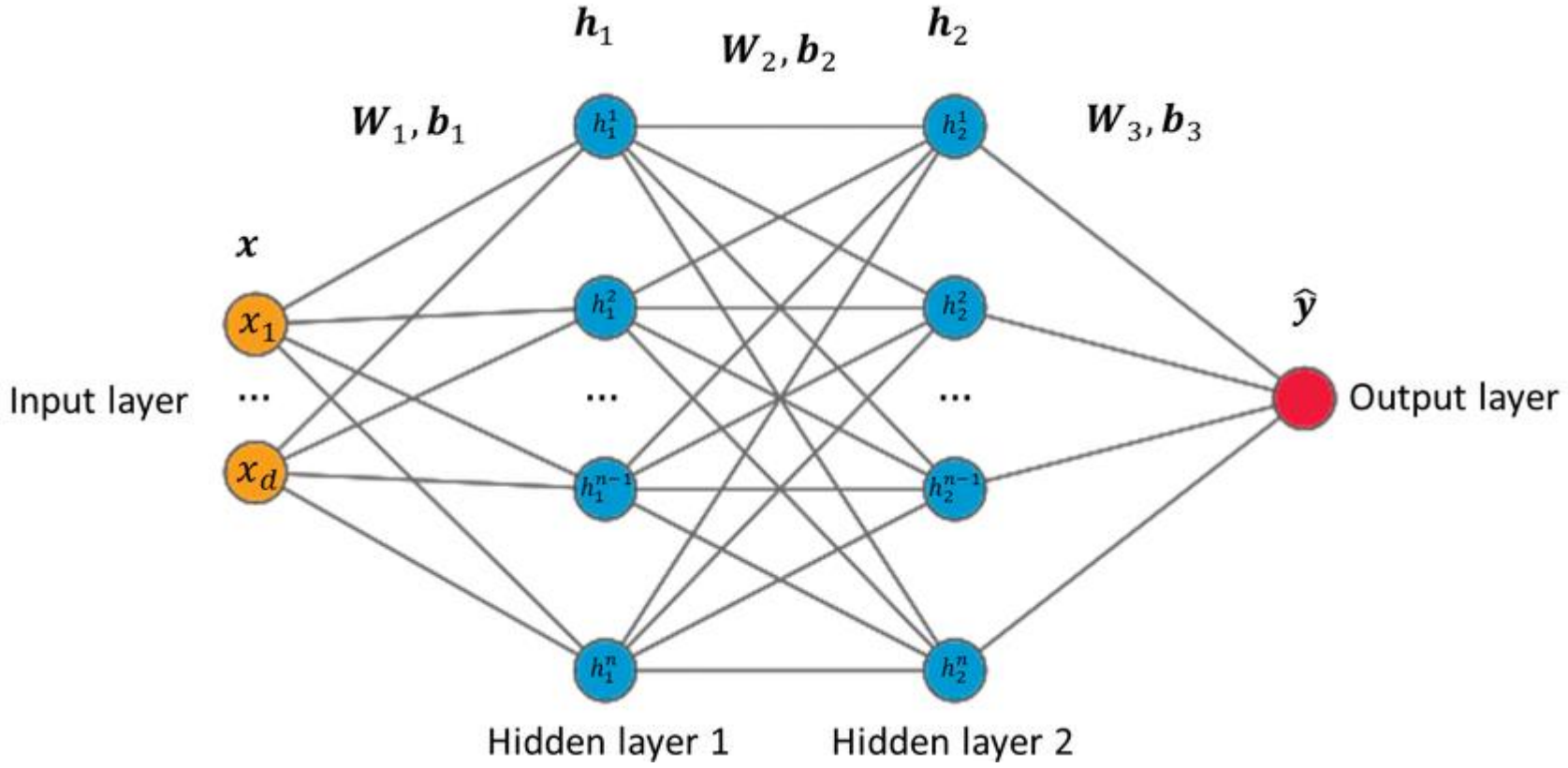
Classification Models
Multi-Layer Perceptron

- Perceptron



Classification Models
Multi-Layer Perceptron

- Multi-Layer Perceptron (MLP)

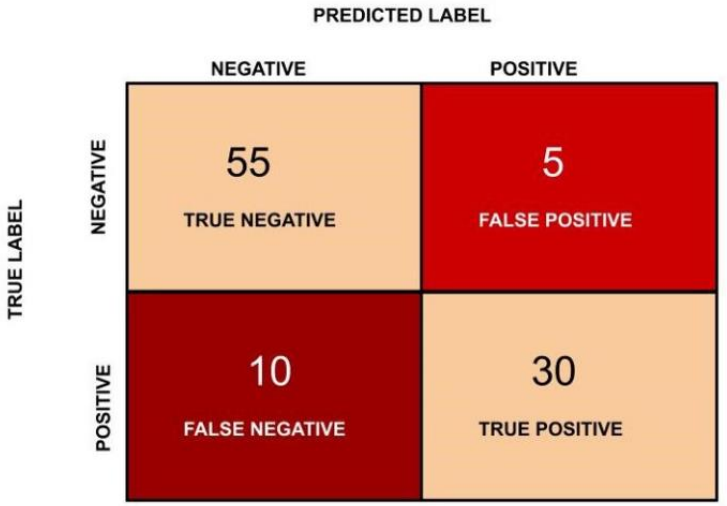


Evaluation Metrics

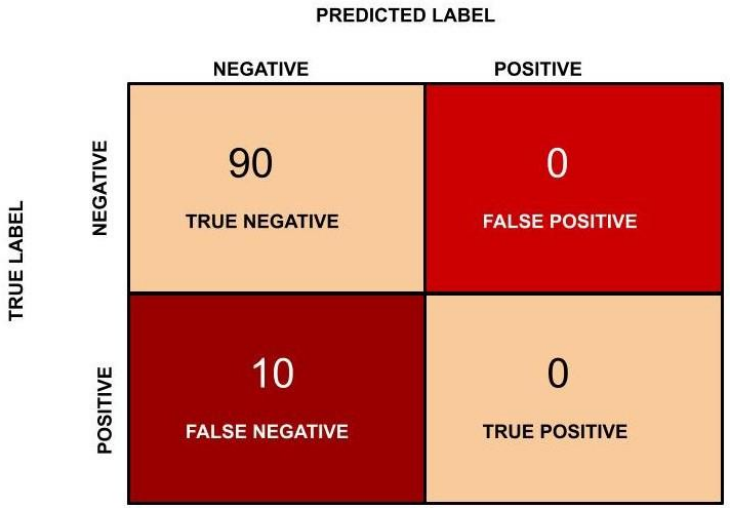
- Accuracy:
  - Correctly classified instances over **total** instances

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

  - (55 + 30)/(55 + 5 + 30 + 10) = 0.85



- What is the problem with accuracy?
  - Imbalanced Data
    - Acc: 90% (90/100)
    - Error TP: 100% (10/10)

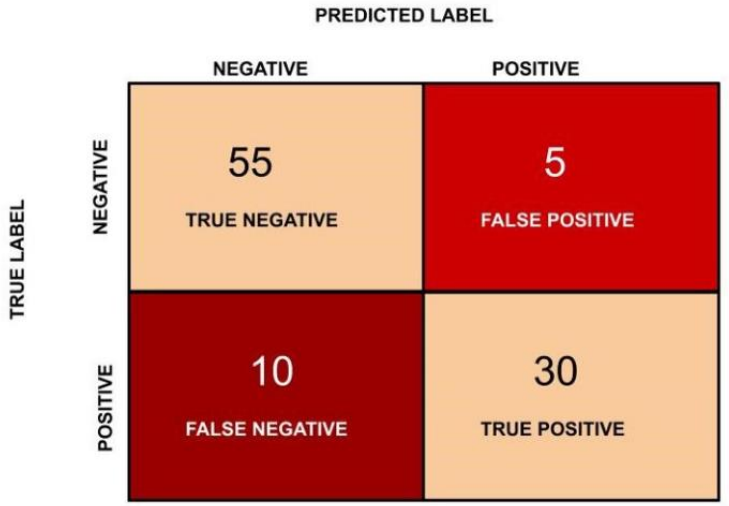


Evaluation Metrics

- Precision:
  - Correctly **positive classified instances** over **positive predictions**

$$Precision = \frac{TP}{TP + FP}$$

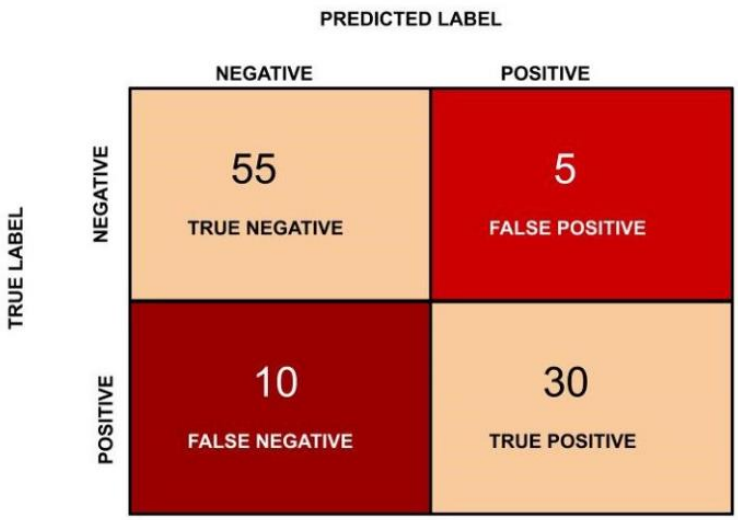
  - 30/(30+ 5) = 0.857



- Recall
  - Correctly **positive classified instances** over **positive instances** (A.K.A Sensitivity or TP Rate)

$$Recall = \frac{TP}{TP + FN}$$

  - 30/(30+ 10) = 0.75



Let’s Code!

- Lecture 08 - Image Classification.ipynb [LINK]