Week09: Supervised Learning Part 3
Multiclass classification, k-NN and SVM

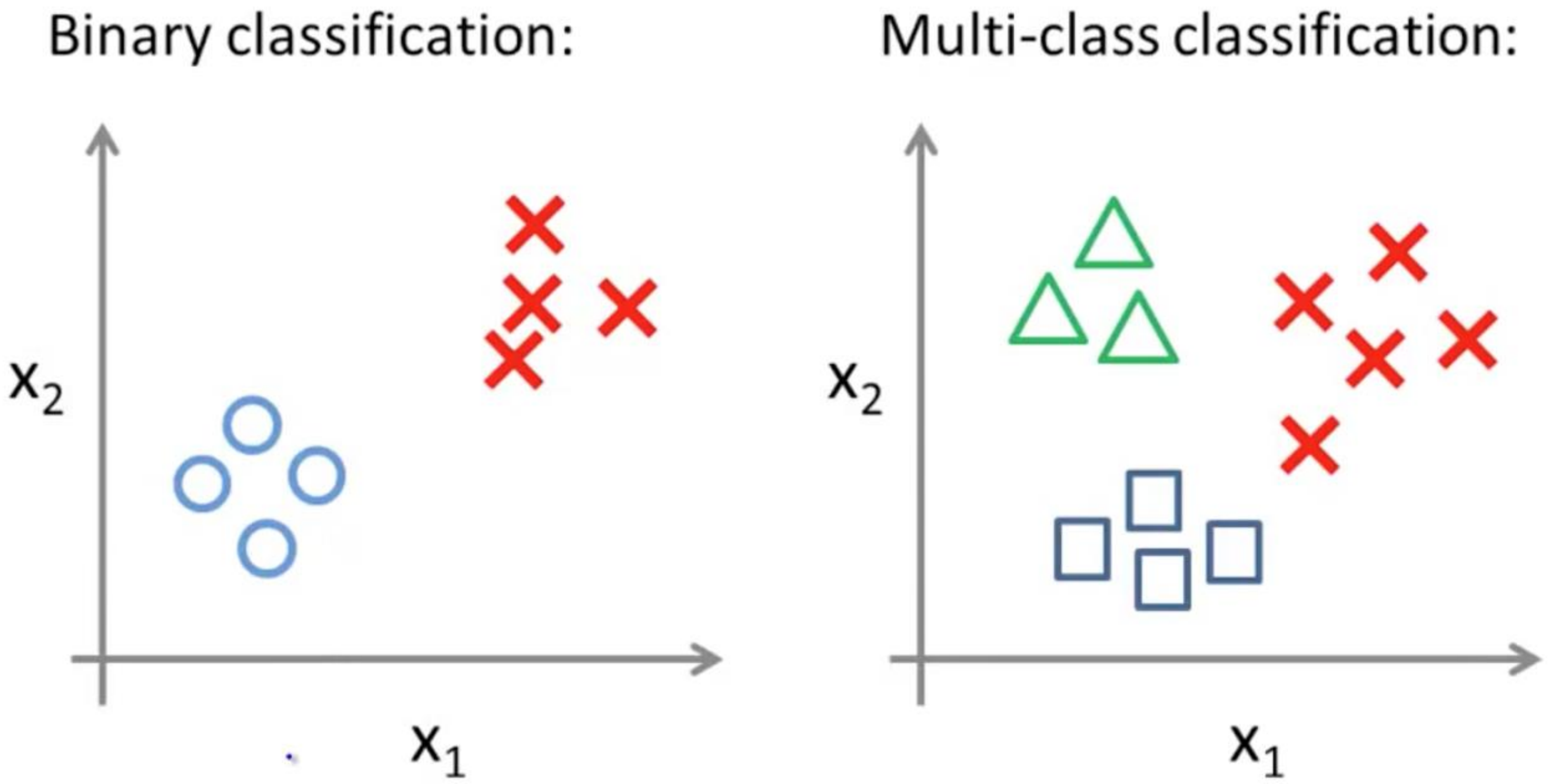
21/03/2024

Dr. Teema Leangarun

Faculty of Engineering, Department of Control Systems and Instrumentation Engineering
King Mongkut's University of Technology Thonburi

Multi-Class Classification

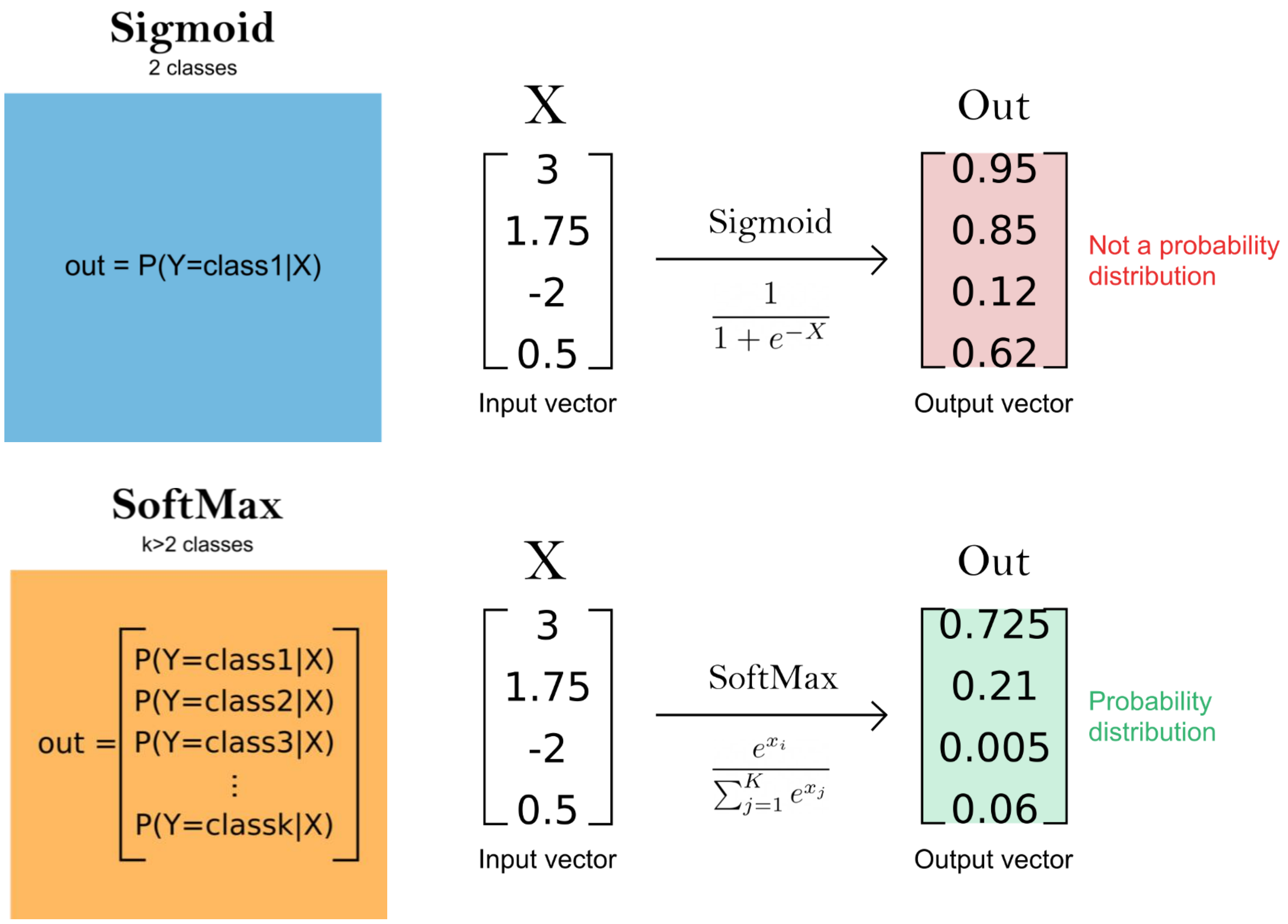
Multi-Class Classification



$y = \{0,1\}$ $y = \{0,1, \ldots, n\}$

https://www.coursera.org/learn/machine-learning

Softmax Regression (Normalized Exponential Function)



The output vector must be a probability distribution over all the predicted classes, i.e. all the entries of the vector must add up to 1.

https://towardsdatascience.com/sigmoid-and-softmax-functions-in-5-minutes-f516c80ea1f9

Softmax function

- Softmax function outputs a vector that represents the probability distributions of a list of potential outcomes.

- *For example,*
  - array([0.09003057, 0.24472847, 0.66524096])
  - Class 1 = 9.00%
  - Class 2 = 24.47%
  - Class 3 = 66.52%
  - Softmax function selects Class 3 (The highest probability).

Example of Multinomial Logistic Regression

https://www.kaggle.com/code/vitorgamalemos/multinomial-logistic-regression-from-scratch

K-Nearest Neighbors (KNN)

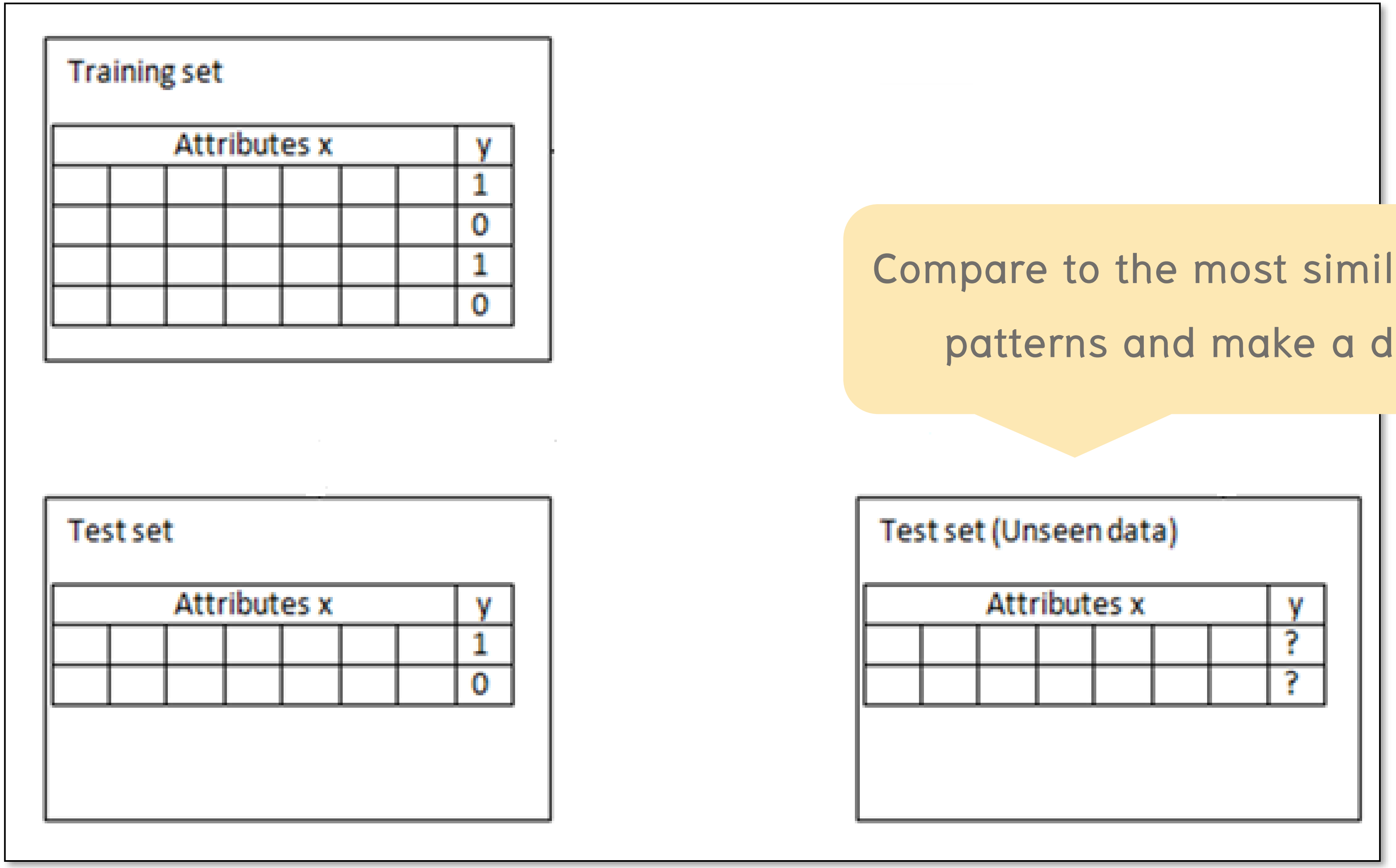
K-Nearest Neighbors (KNN)

- **Simple**, but a very powerful classification algorithm
- Classifies based on a similarity measure
  - Make predictions based on the k most similar training patterns for a new data instance.

- Lazy learning (Instance-based Learning)
  - Learning = storing all training instances
  - Does not "learn" until the test example is given
  - Whenever we have a new data to classify, we find its K-nearest neighbors from the training data.

Lazy learning

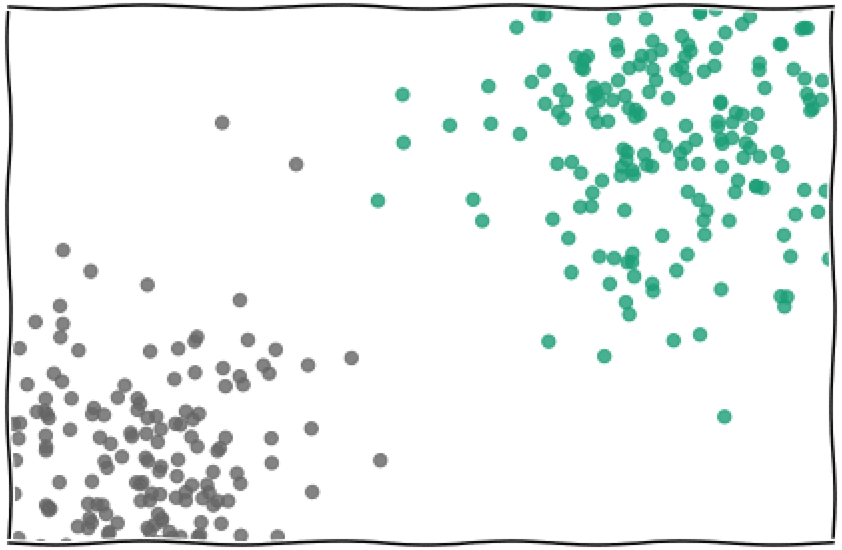
Lazy learning

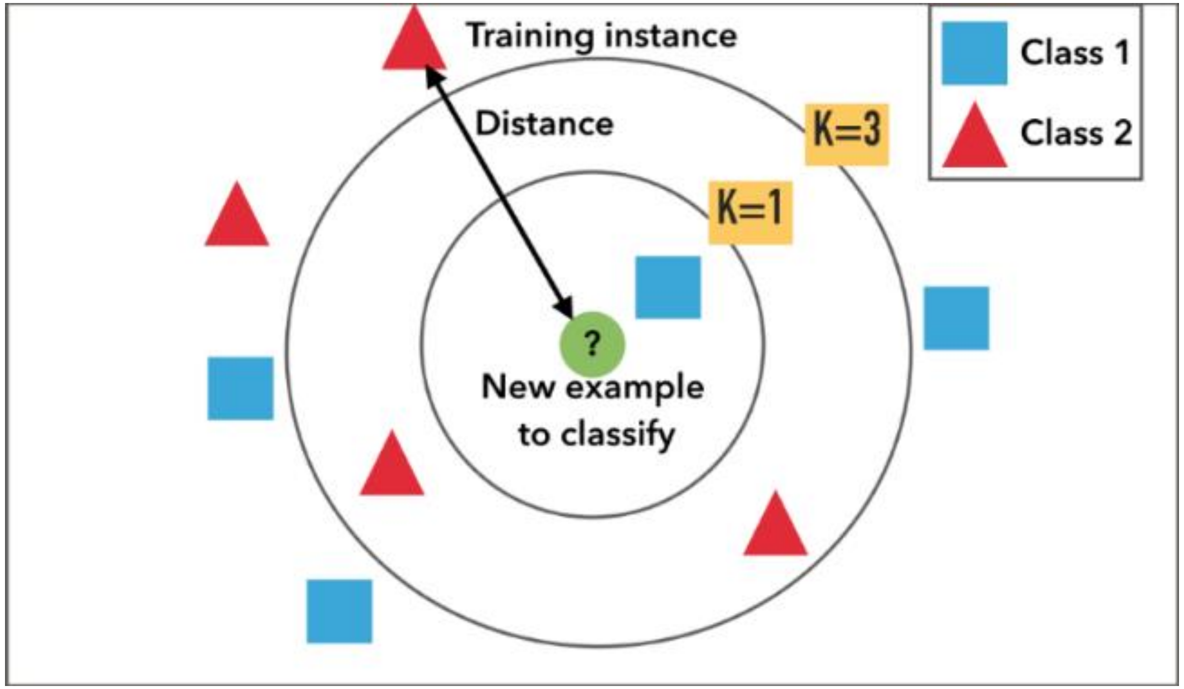


Compare to the most similar training patterns and make a decision

K-Nearest Neighbors (KNN)

- Its purpose is to use a database in which the *data points* are separated into several classes to predict the classification of a new sample point.
- How closely out-of-sample features resemble our training set determines how we classify a given data point.





https://milliams.com/courses/applied_data_analysis/Nearest%20Neighbours.html

What is it used for?

K nearest neighbors is used for

- Classifying samples according to their numerical features
- Performing a regression of numerical values based on the features of the sample.

Examples

Example 1: Forecast the sales of a mathematics textbook based on features such as price, length, number of university courses in the area, etc.

Example 2: Credit scoring, classify as good or bad risk, or on a scale, based on features such as income, value of assets, etc.

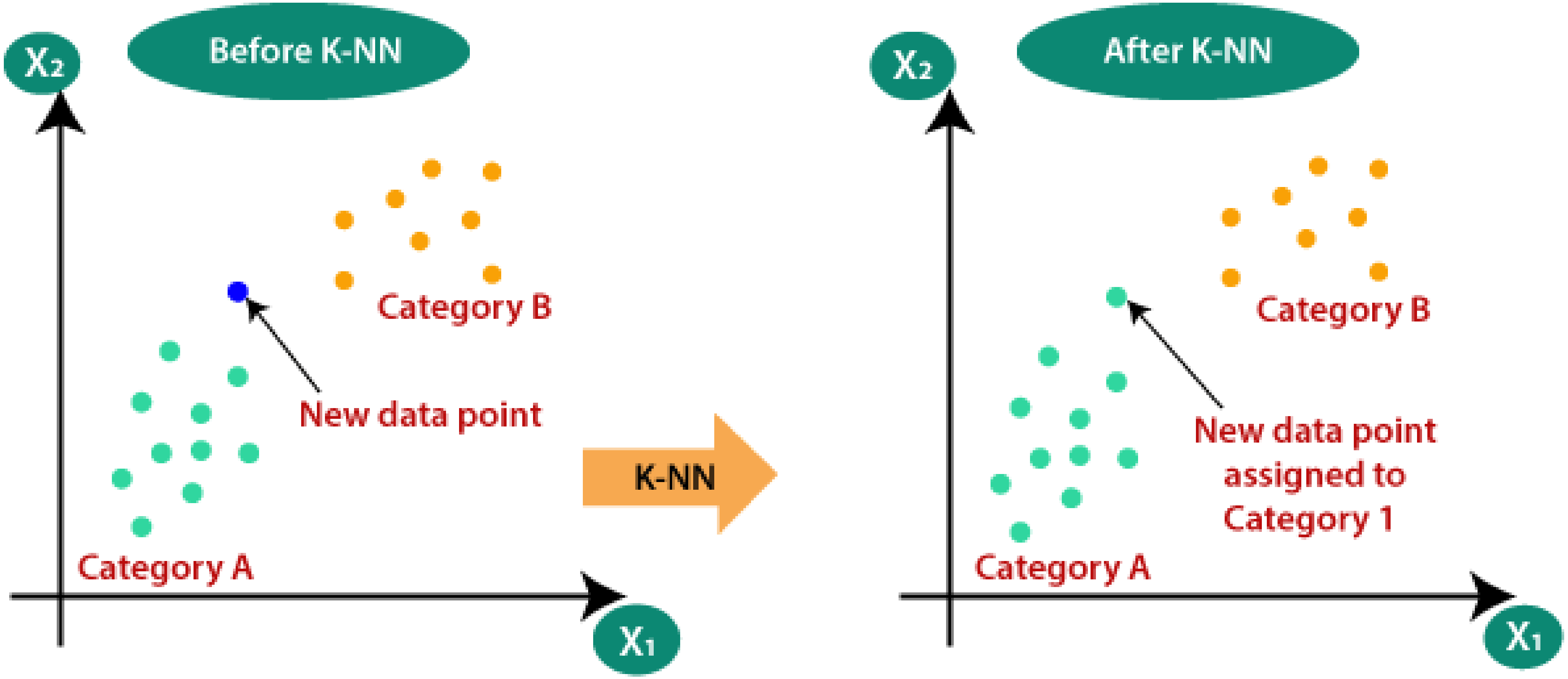
Example 3: Predict movie rating (number of ‘stars’) based on features such as amount of action sequences, quantity of romance, budget, etc.

How it works

- K nearest neighbors (KNN) is perhaps the easiest machine learning technique to grasp conceptually.
- Although really there is no learning at all.

- It can be used for classification, determining into which group an individual belongs.
- Or it can be used for regression, predicting for a new individual of a variable based on the values for similar individuals.

How does K-NN work?



https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

How does K-NN work?

The K-NN working can be explained based on the below algorithm:

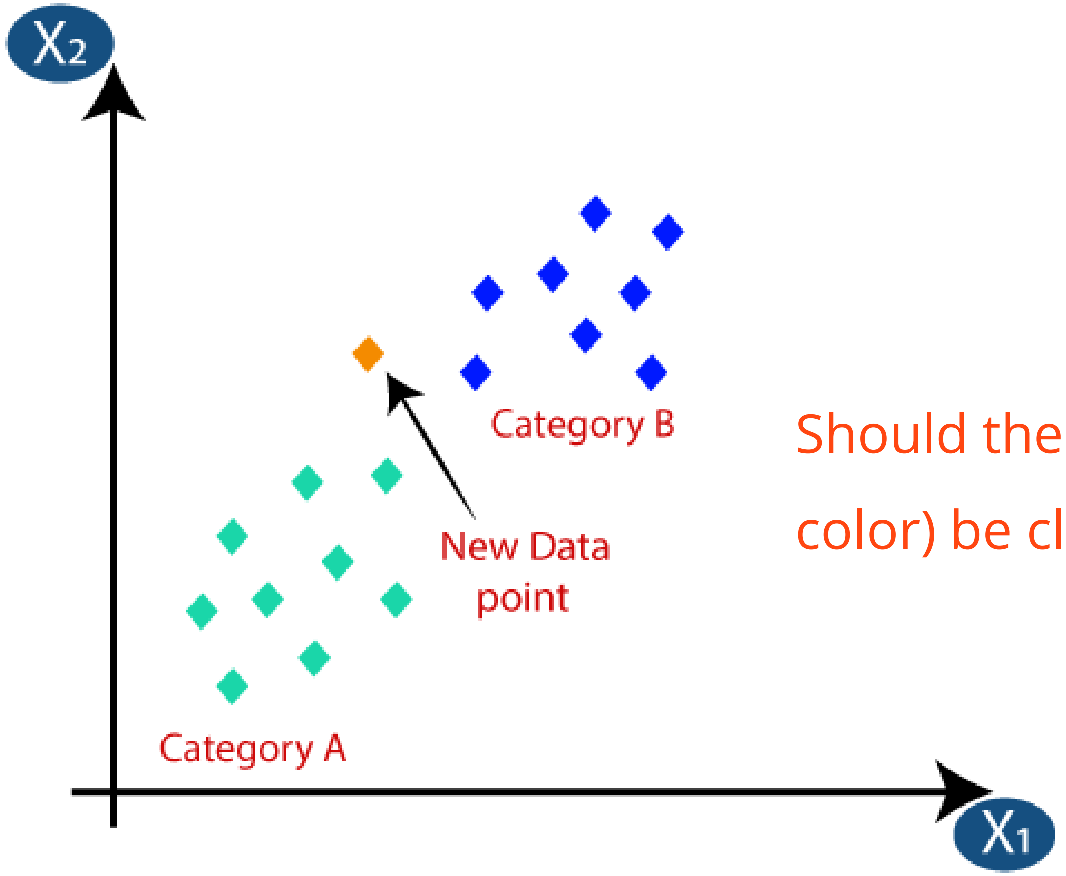
- **Step-1:** Select the number K of the neighbors
- **Step-2:** Calculate the Euclidean distance* of **K number of neighbors**
- **Step-3:** Take the K nearest neighbors as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbors, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbor is maximum.
- **Step-6:** Our model is ready.

*There are several distance metrics used in k-NN.

https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

How does K-NN work?

Suppose we have a new data point and we need to put it in the required category.

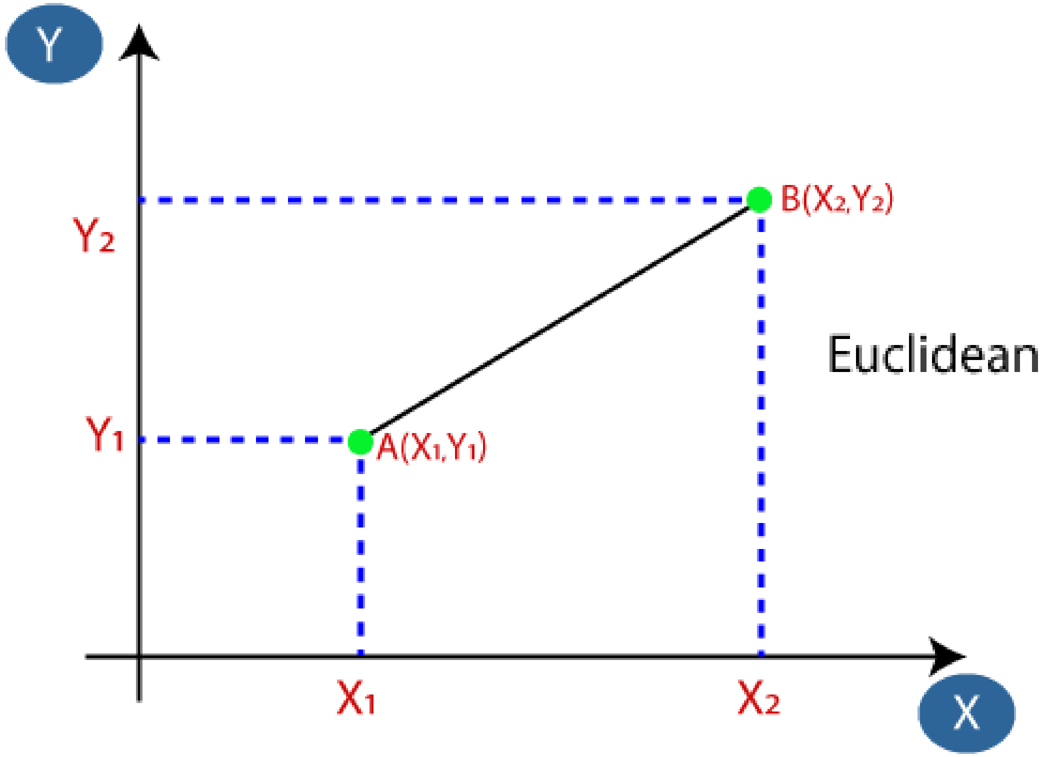


Should the new point (temporarily marked an orange color) be classified as category A or category B?

https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

How does K-NN work?

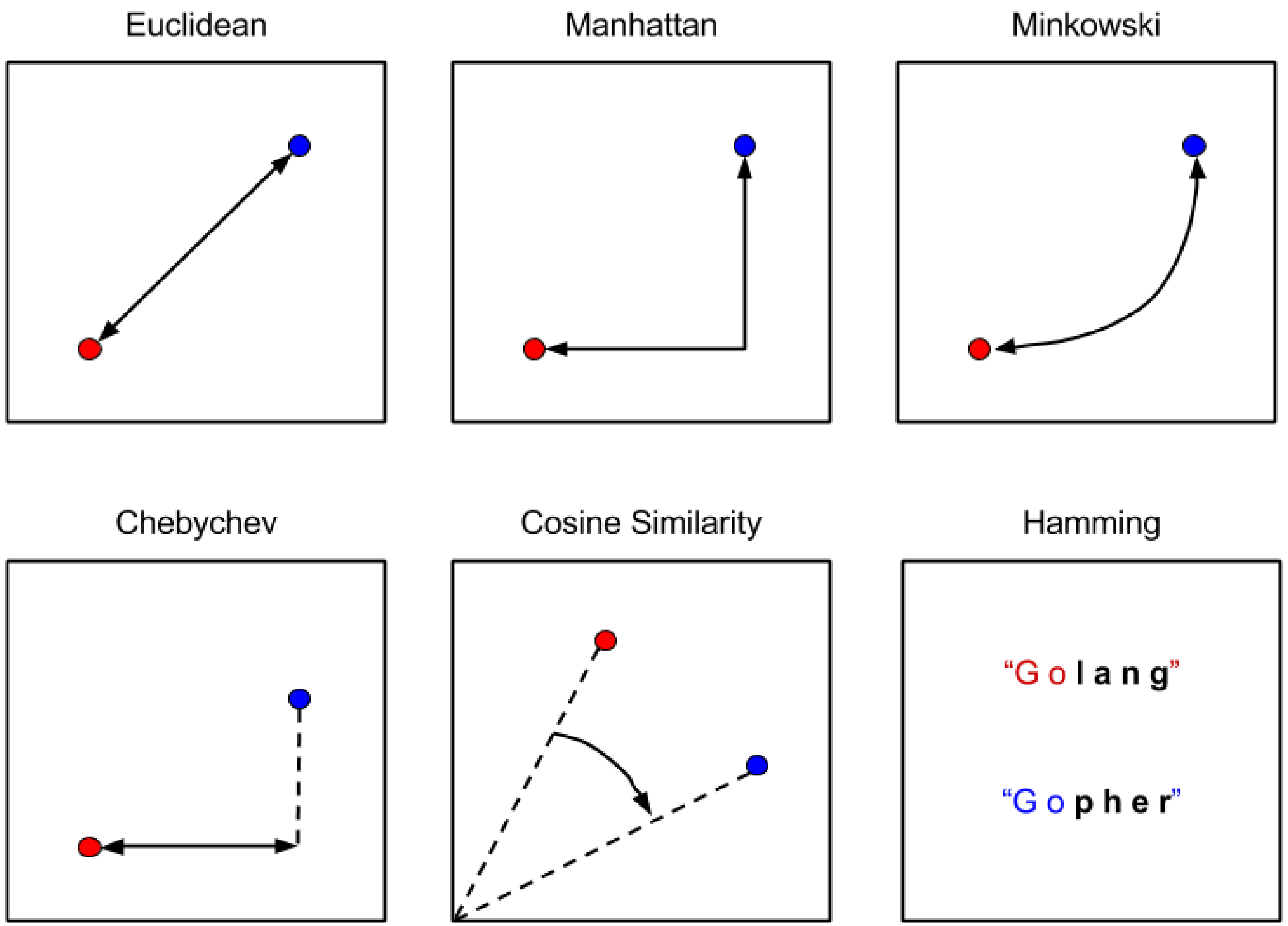
- **Step 1:** Firstly, we will choose the number of neighbors, so we will choose the k=5.
- **Step 2:** Next, we will calculate the Euclidean distance between the data points.
- It calculates the ordinary straight-line distance between two points in a Euclidean space.



$$\text{Euclidean Distance between } A_1 \text{ and } B_2 = \sqrt{(X_2-X_1)^2+(Y_2-Y_1)^2}$$

https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

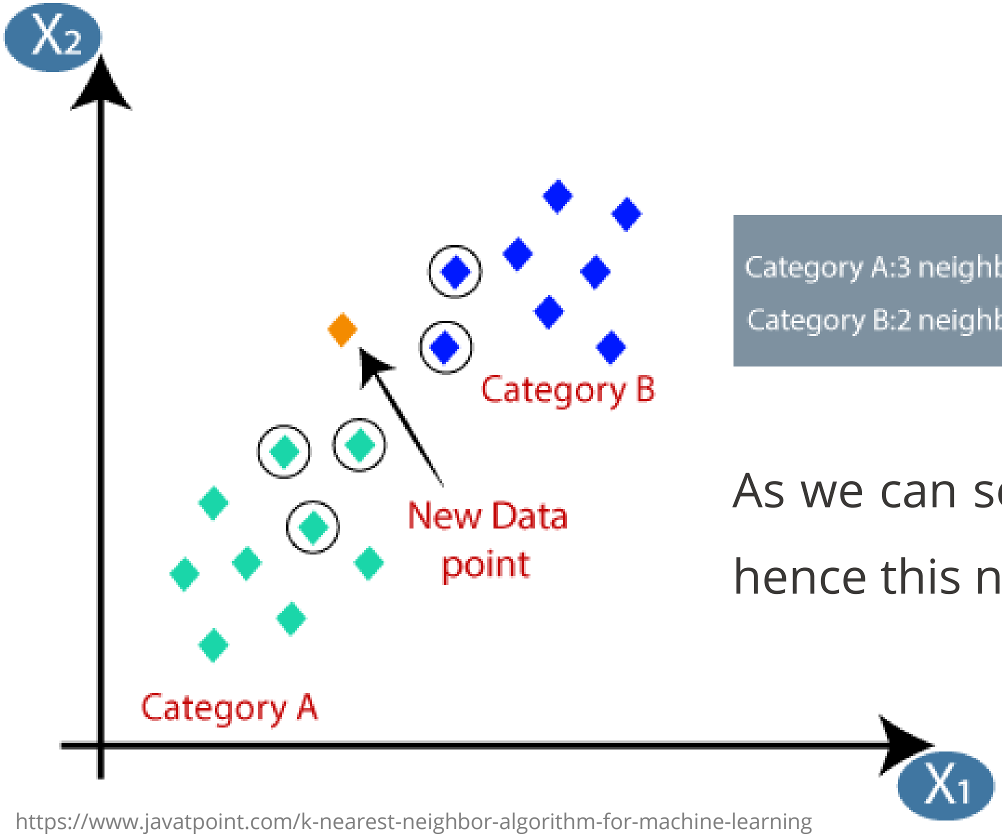
6 Distance measures



https://towardsdatascience.com/getting-acquainted-with-k-nearest-neighbors-ba0a9ecf354f

How does K-NN work?

- **Step 3 & 4:** By calculating the Euclidean distance we got the nearest neighbors,
  - as 3 nearest neighbors in category A and
  - 2 nearest neighbors in category B.



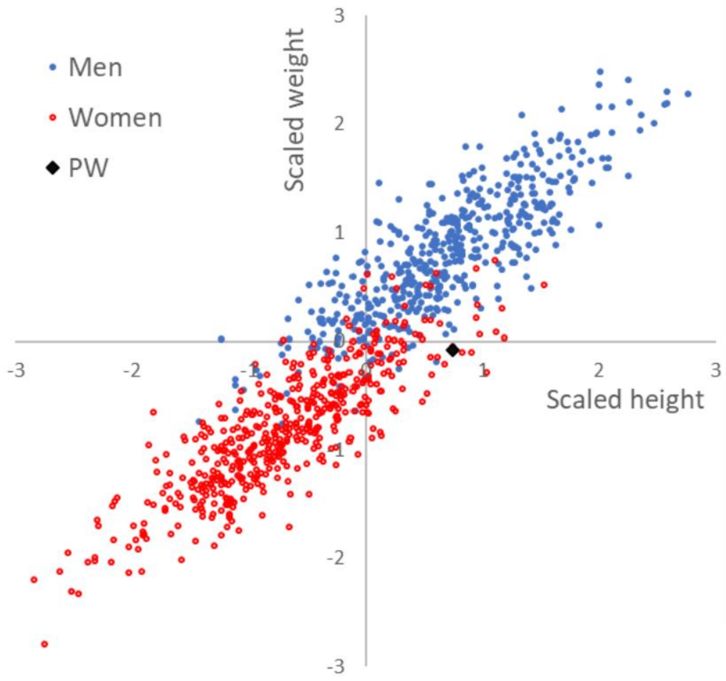
Category A:3 neighbors
Category B:2 neighbors

As we can see the 3 nearest neighbors are from category A, hence this new data point must belong to category A.

https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

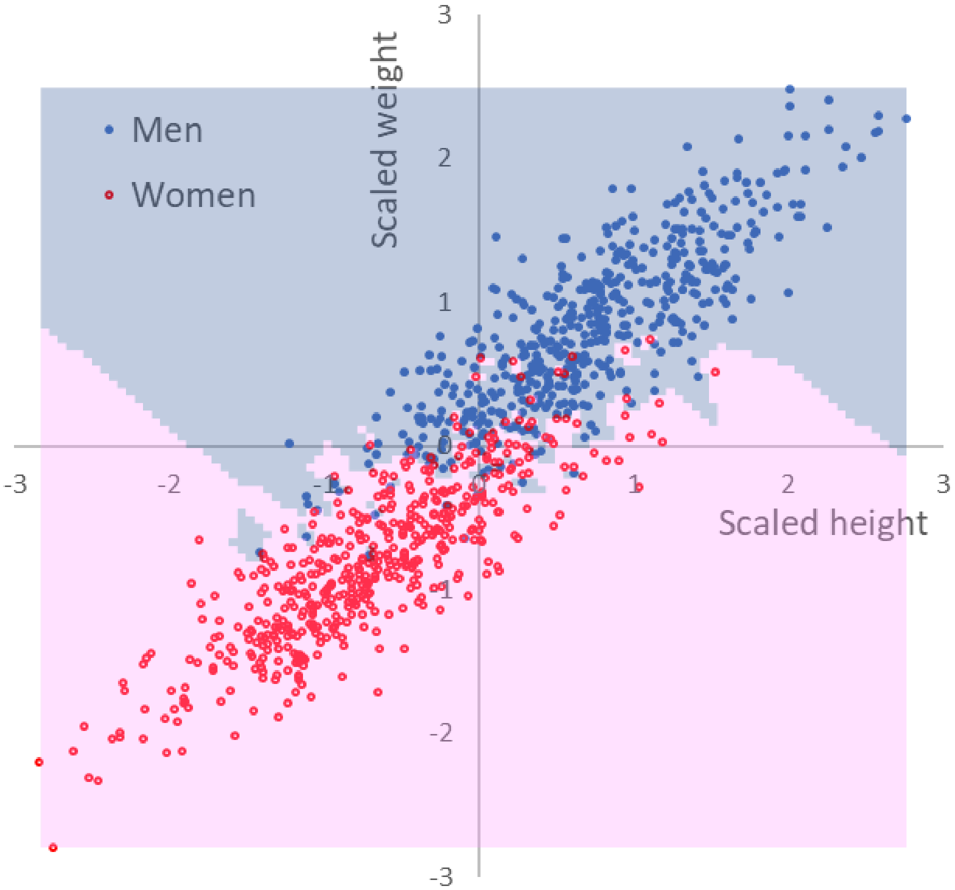
Example: Heights and weights

- Easily find online data for men's and women's heights and weights.
- In the figure, just 500 points for each gender.
- The data has been shifted and scaled for both genders.
- So, have a mean of zero and a standard deviation of one.



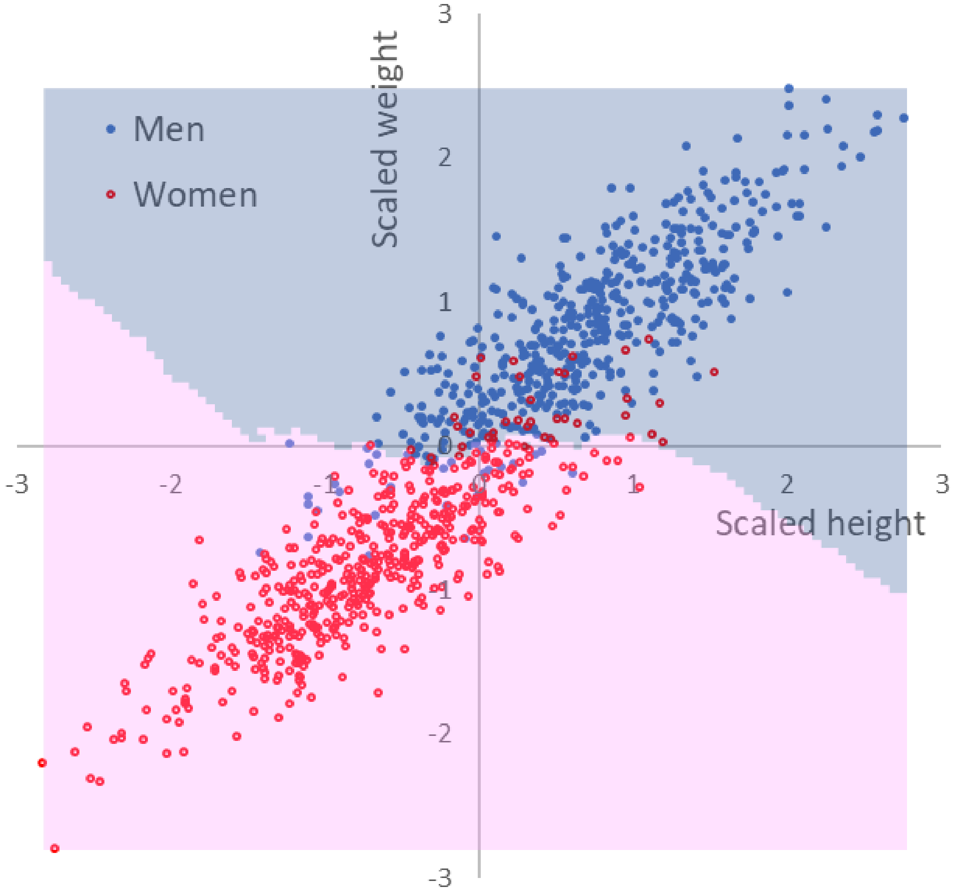
Example: Heights and weights (contd.)

- When K = 1 the male/female regions are as shown below.
- We can see several islands, pockets of males within what looks like, otherwise female zones and vice versa.



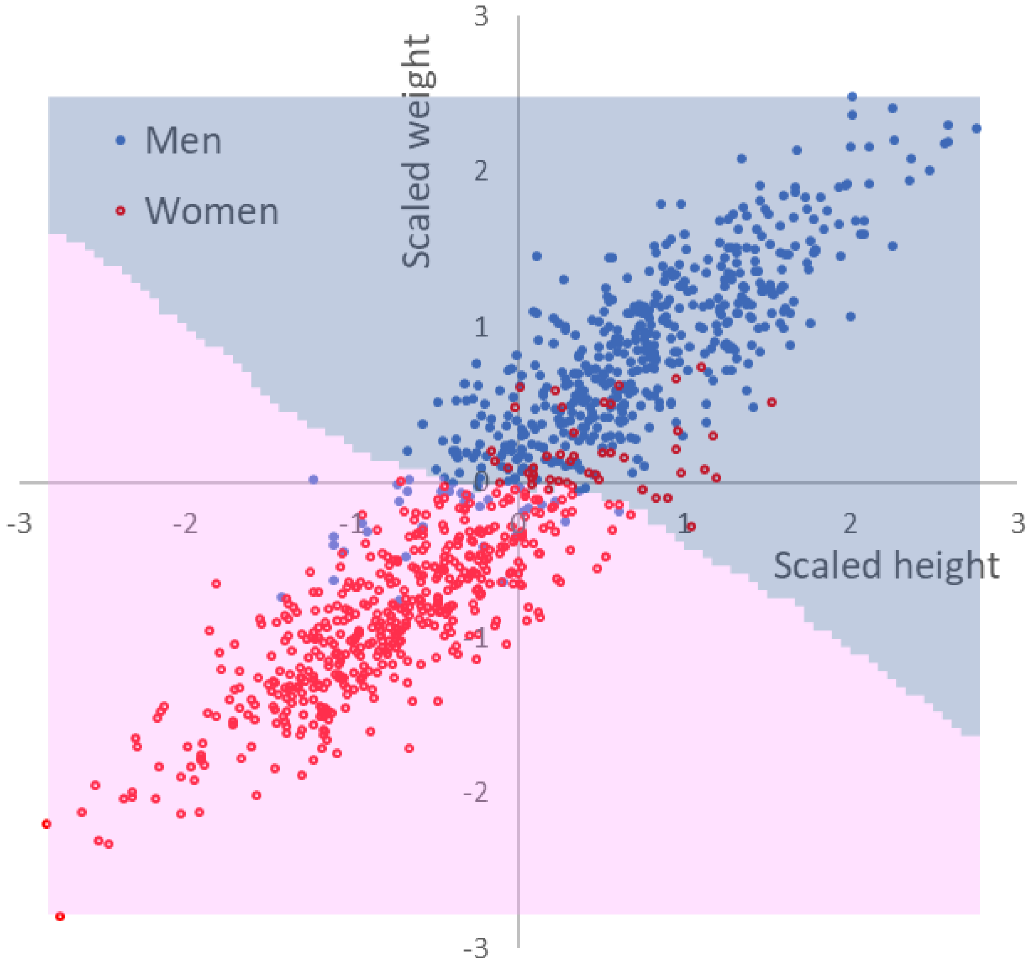
Example: Heights and weights (contd.)

- When K = 21 the boundary has been smoothed out.
- There is just a single solitary island left.



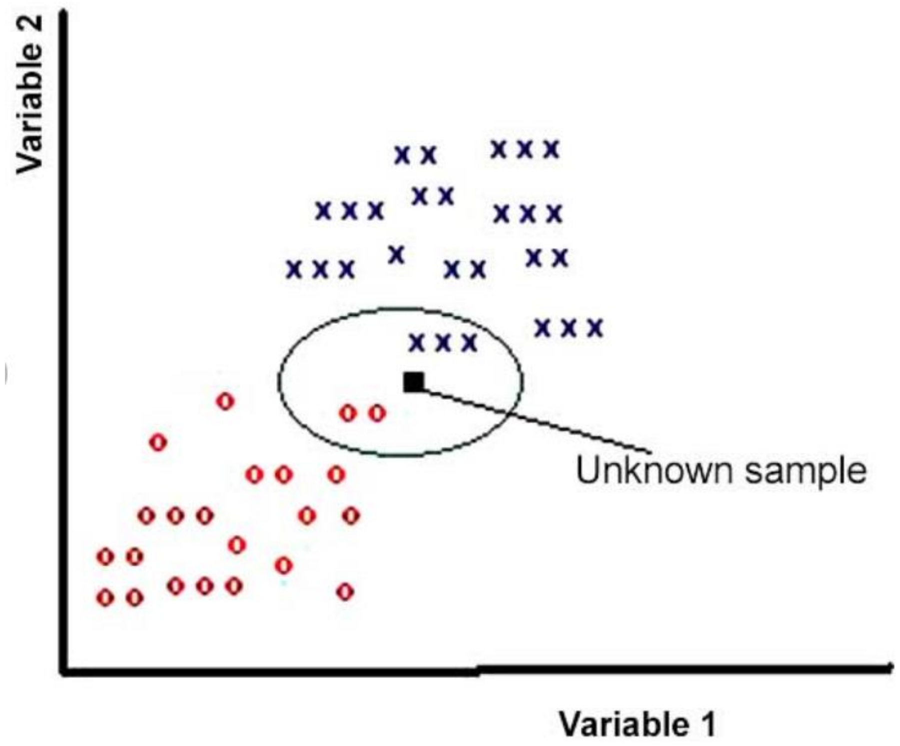
Example: Heights and weights (contd.)

- As an extreme case take K = 101.



How to select the value of K in the K-NN Algorithm?

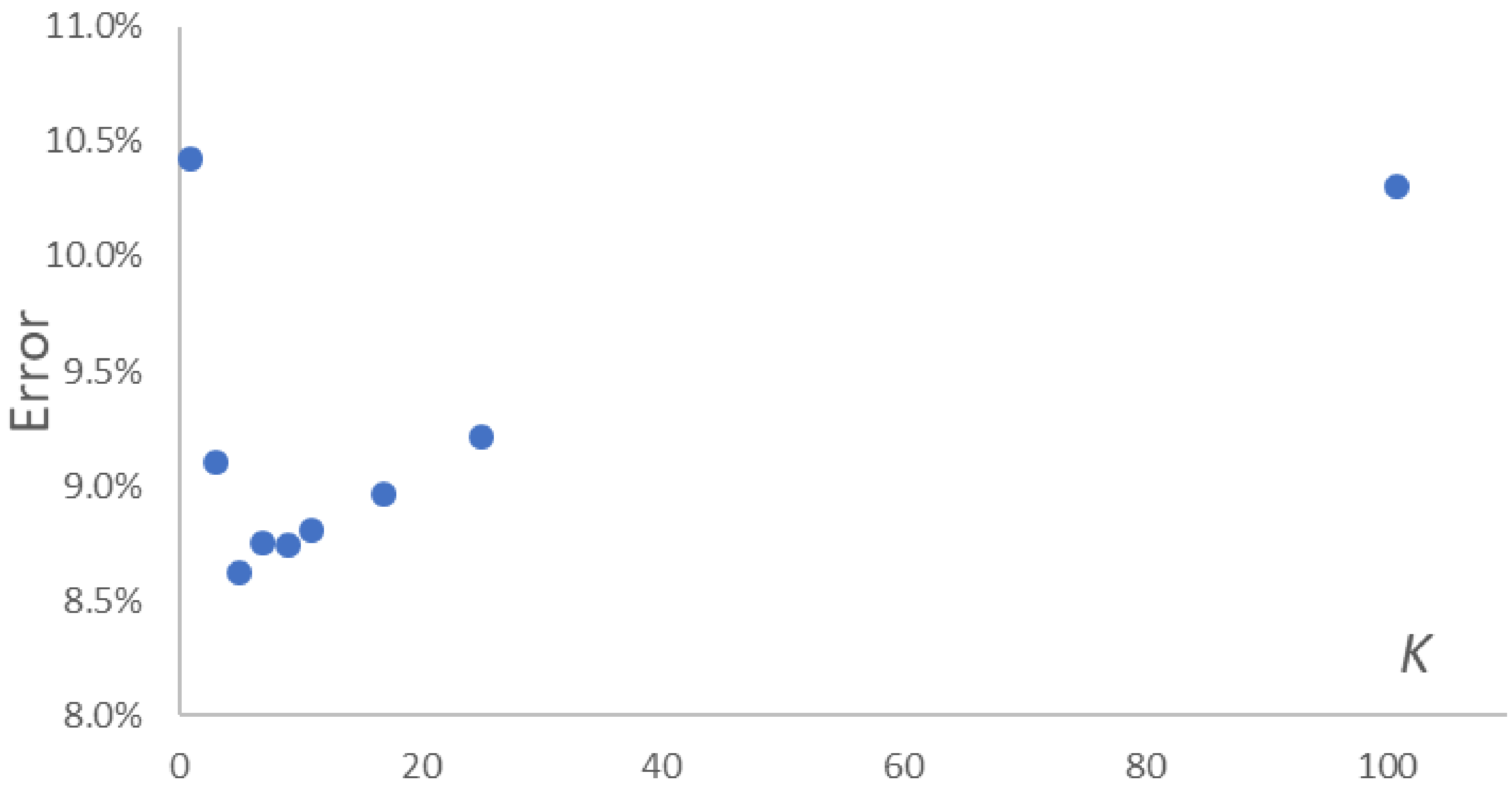
- There is no particular way to determine the best value for "K", so we need to try some values to find the best out of them (lowest error).
- The most preferred value for K is 5 (default value).



https://www.researchgate.net/figure/Fig-4-k-NN-classification-with-k5-since-among-the-5-nearest-neighbors-of-the_fig2_232696878

How to select the value of K in the K-NN Algorithm?

- We can plot misspecification error as a function of the number of neighbors, K.
- It looks like K = 5 is optimal.

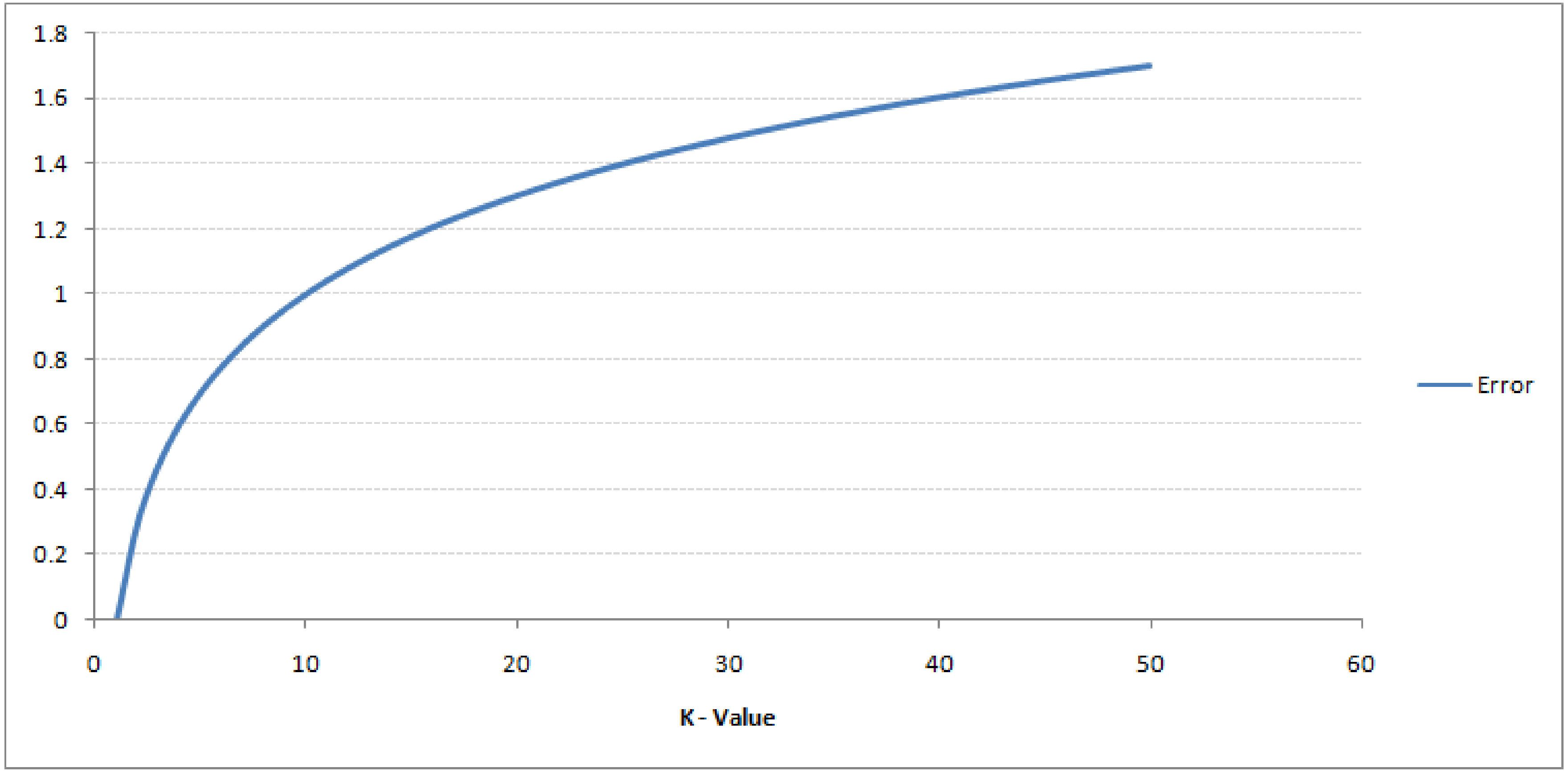


How to select the value of K in the K-NN Algorithm?

- The most preferred value for K is 5 (default value).
- A very low value for K such as K=1 or K=2, can be noisy and lead to the effects of outliers in the model.
- Large values for K are good, but it may find some difficulties.

https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

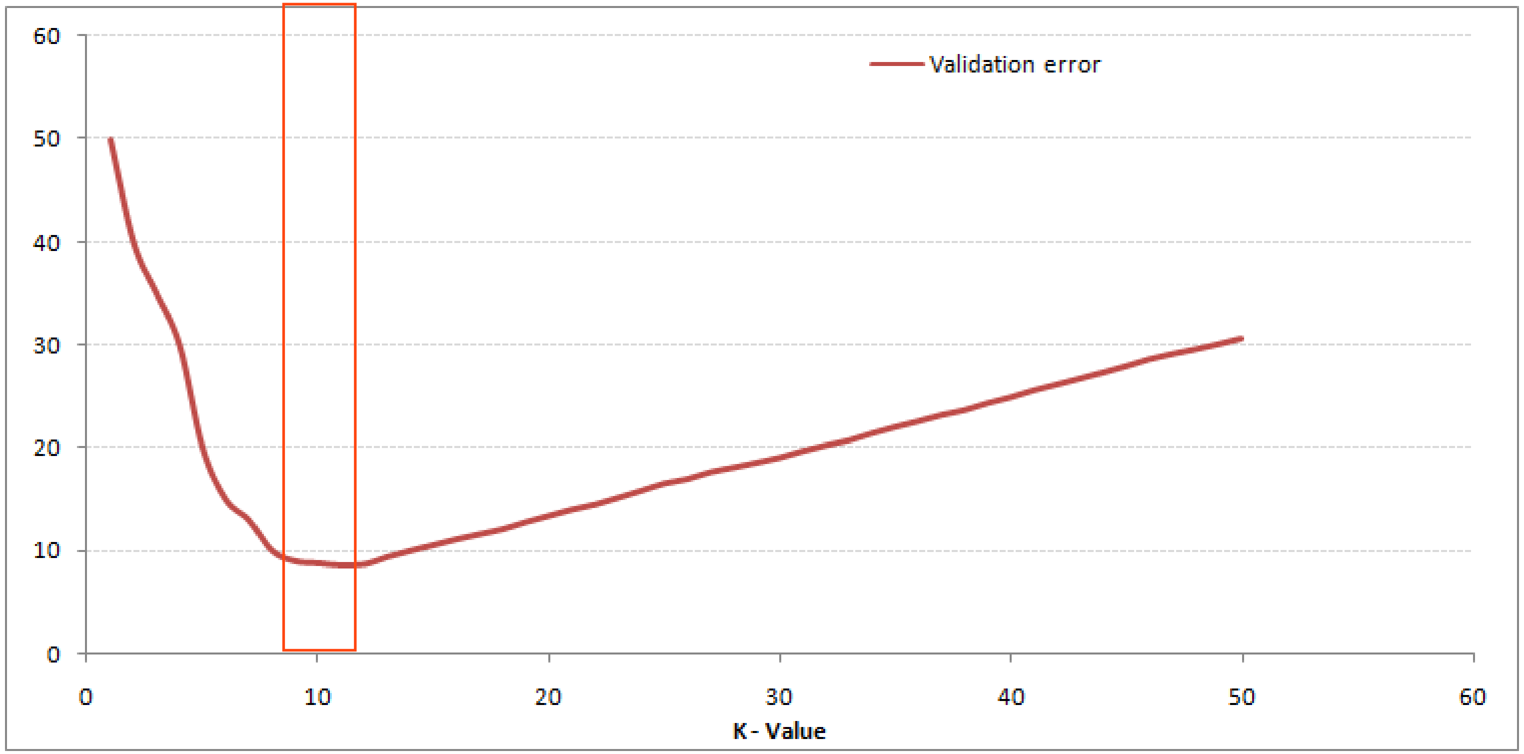
How to select the value of K in the K-NN Algorithm?



- The error rate at K=1 is always zero for the training sample.
- This is because the closest point to any training data point is itself.

https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/

How to select the value of K in the K-NN Algorithm?



- At K=1, we were overfitting the boundaries.
- Thus, error rate initially decreases and reaches a minima.
- After the minima point, it then increase with increasing K.

https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

Limitations of k-NN classifier

- KNN is very sensitive to outliers.
- As dataset grows, the classification becomes slower.
- KNN is not capable of dealing with missing values.
- It is computationally expensive due to high storage requirements.

Summary

Please take away the following important ideas

- *K* nearest neighbors is possibly the easiest machine-learning technique.
- But there isn't any learning.
- It is very easy to understand and can be used for classification or regression.

Support Vector Machine (SVM)

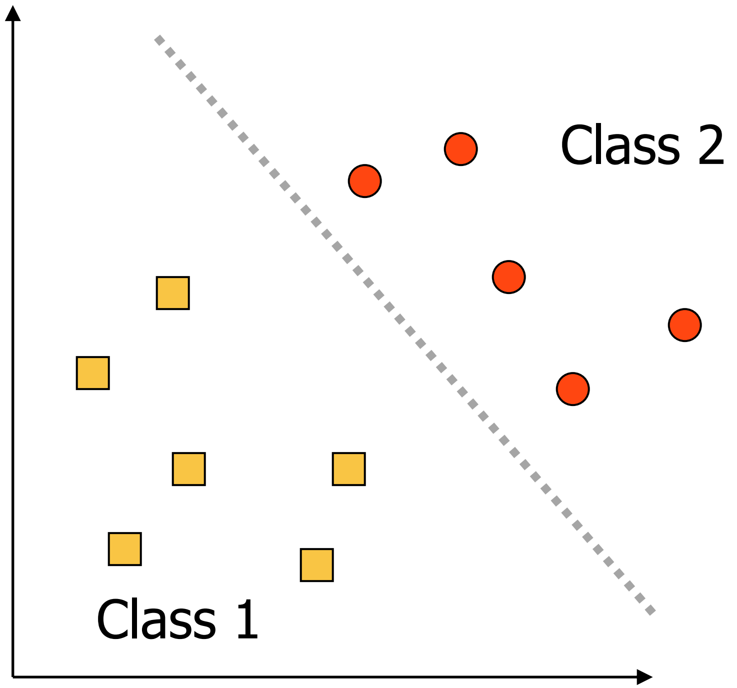
History of SVM

- SVM was first introduced in 1992. [1]
- SVM becomes popular because of its success in handwritten digit recognition
  - 1.1% test error rate for SVM.
- SVM is now regarded as an important example of “kernel methods”, one of the key area in machine learning
  - Note: the meaning of “kernel” is different from the “kernel” function for Parzen windows
  - In SVM, a kernel is a way of computing the dot product of two vectors in a high dimensional feature space.

[1] B.E. Boser *et al*. A Training Algorithm for Optimal Margin Classifiers. Proceedings of the Fifth Annual Workshop on Computational Learning Theory 5 144-152, Pittsburgh, 1992.

What is a good Decision Boundary?

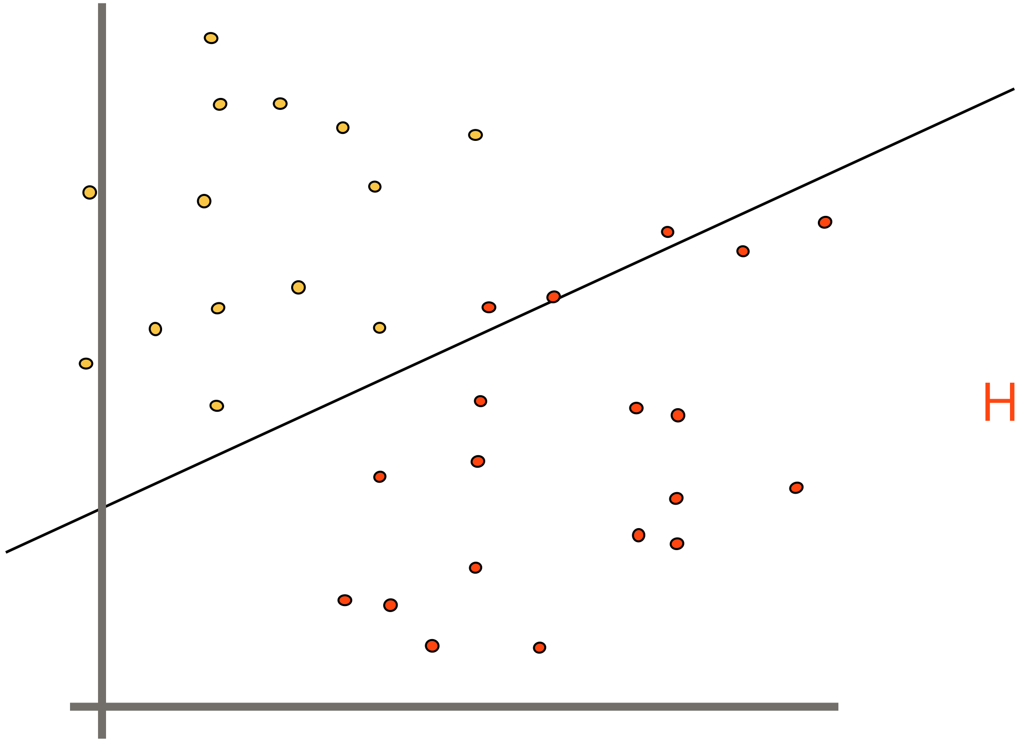
- Consider a two-class, linearly separable classification problem
- Many decision boundaries!
  - Different algorithms have been proposed
- Are all decision boundaries equally good?



What is a good Decision Boundary?

Linear Classifiers

- denotes +1
- denotes -1

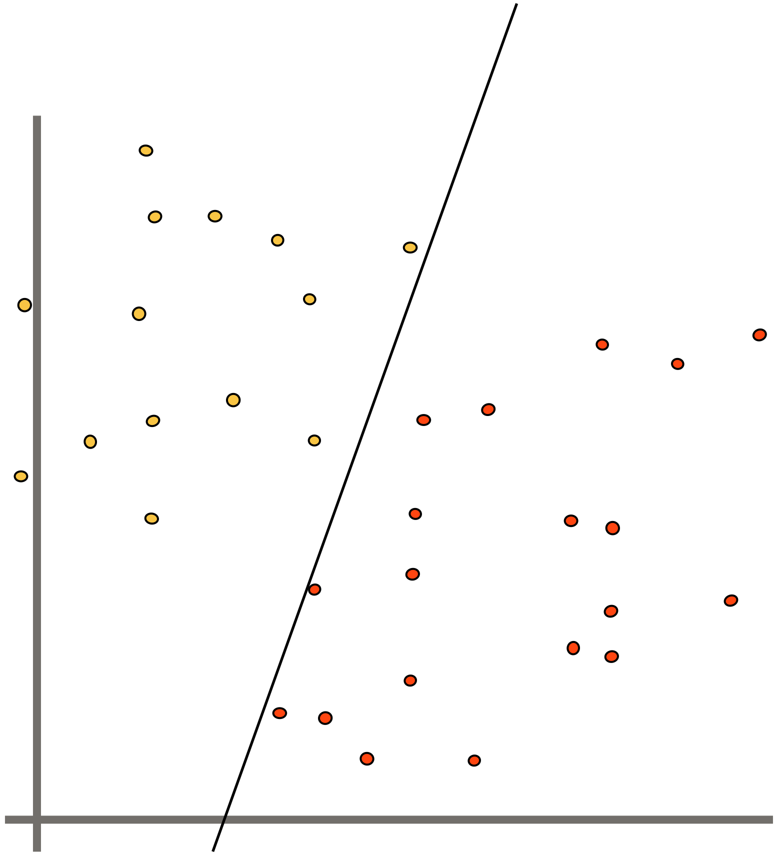


How would you classify this data?

What is a good Decision Boundary?

Linear Classifiers

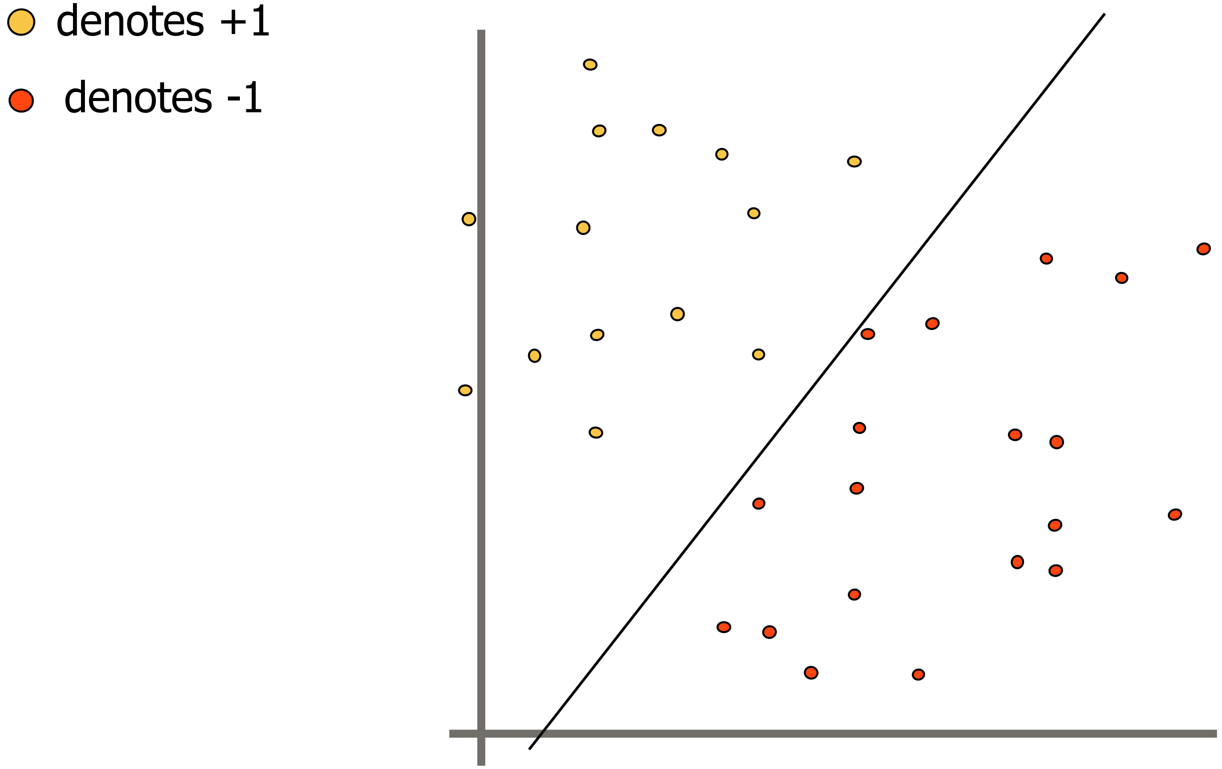
- denotes +1
- denotes -1



How would you classify this data?

What is a good Decision Boundary?

Linear Classifiers

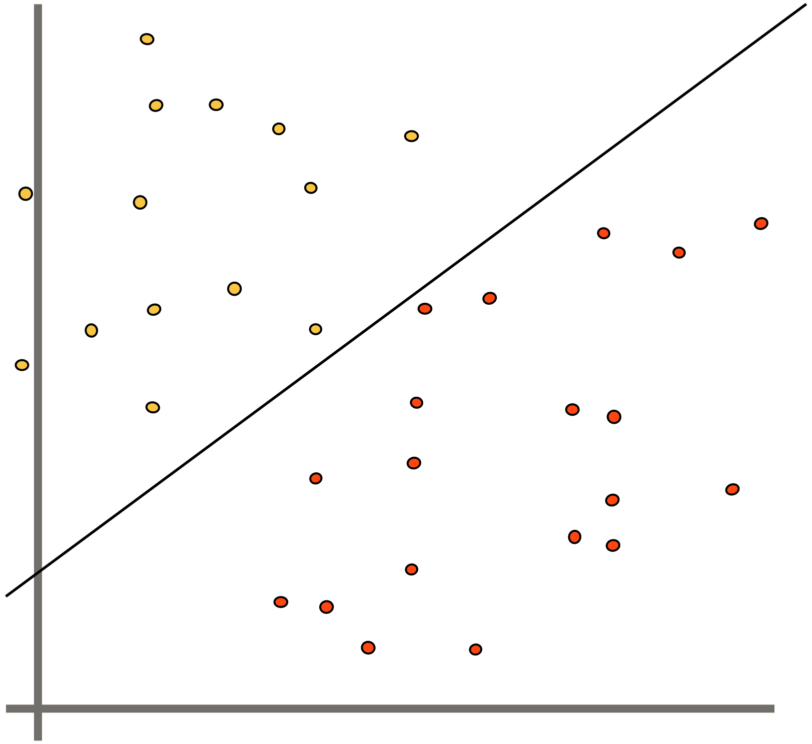


How would you classify this data?

What is a good Decision Boundary?

Linear Classifiers

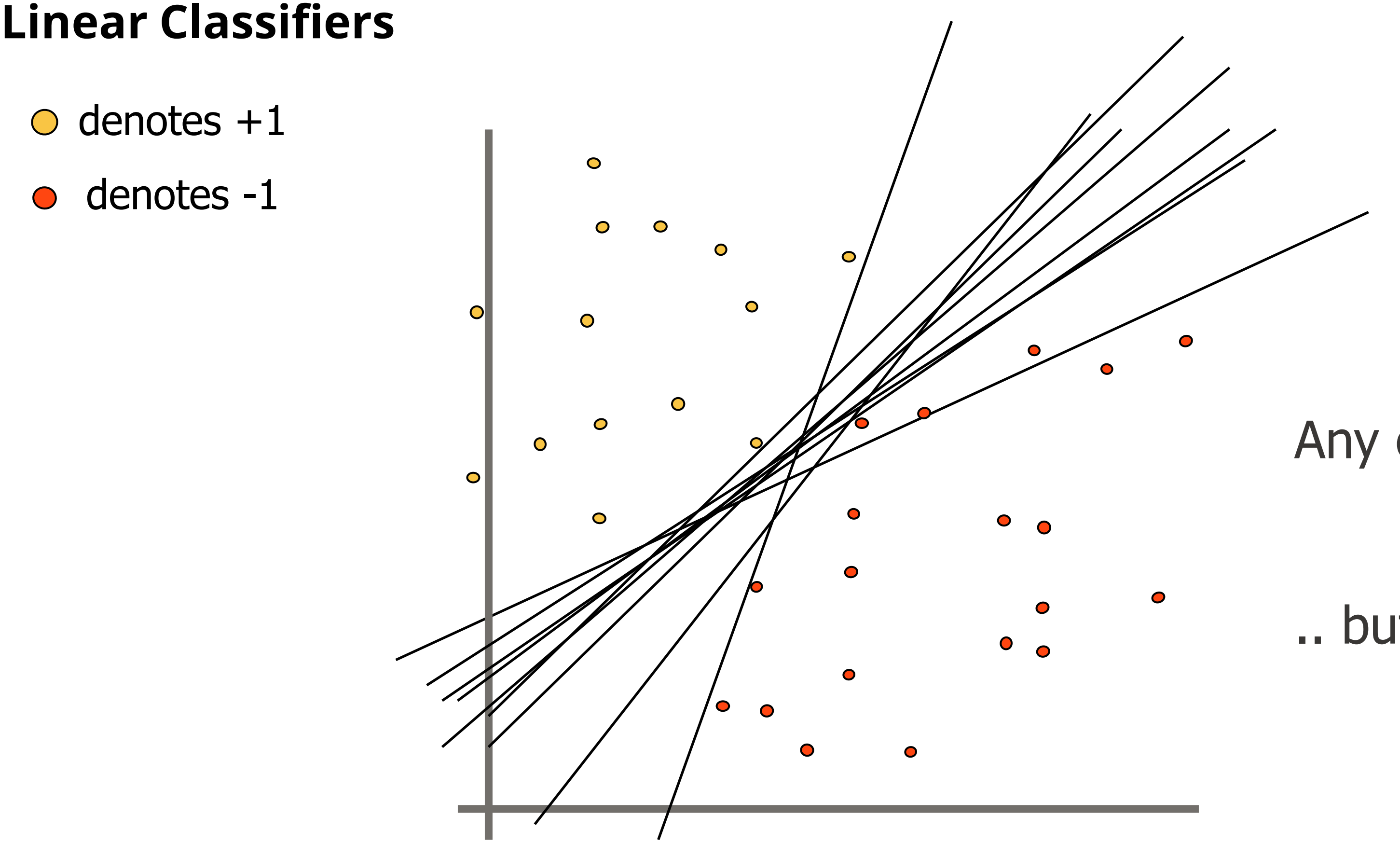
- denotes +1
- denotes -1



How would you classify this data?

What is a good Decision Boundary?

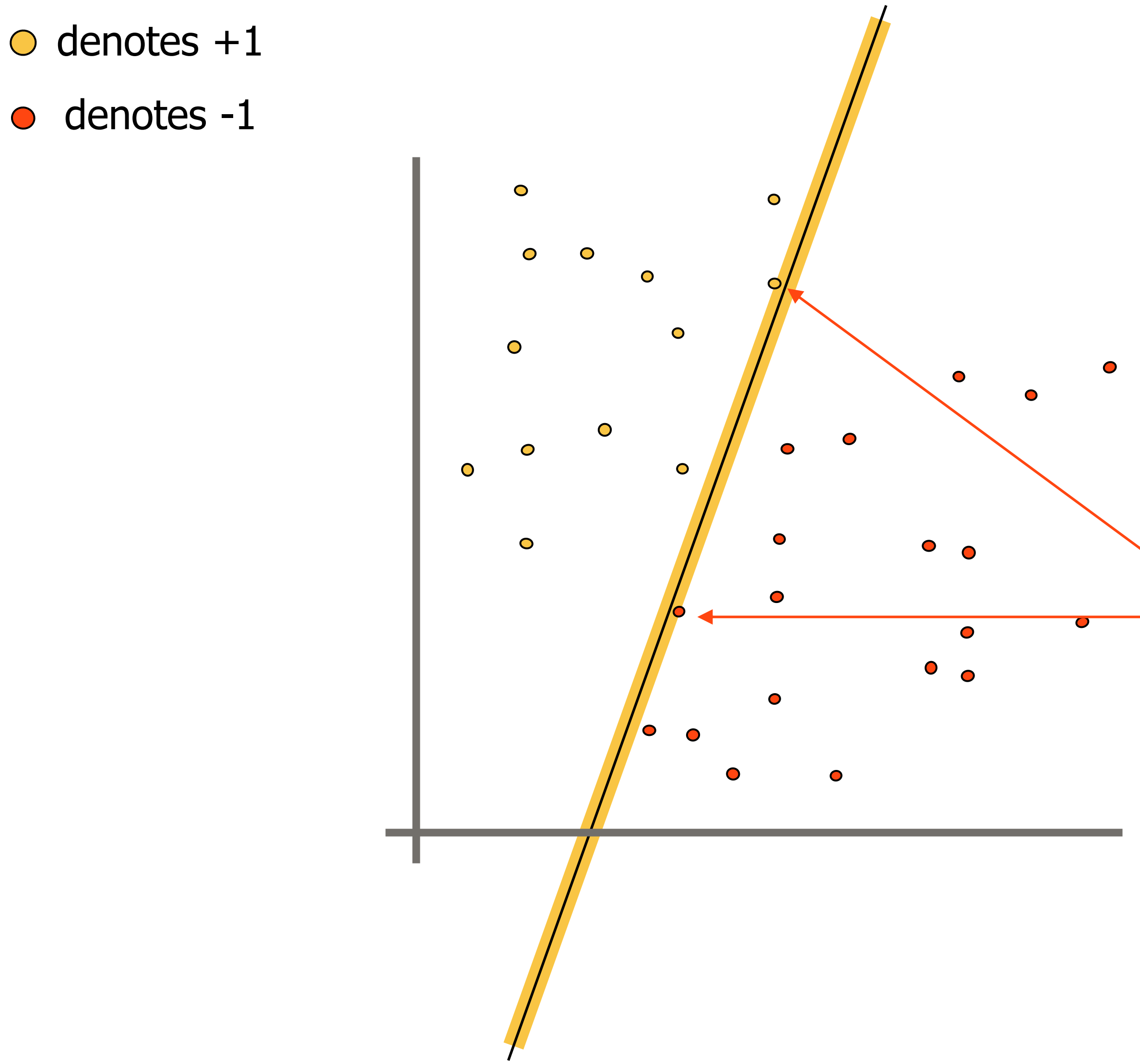
Linear Classifiers



Any of these would be fine..

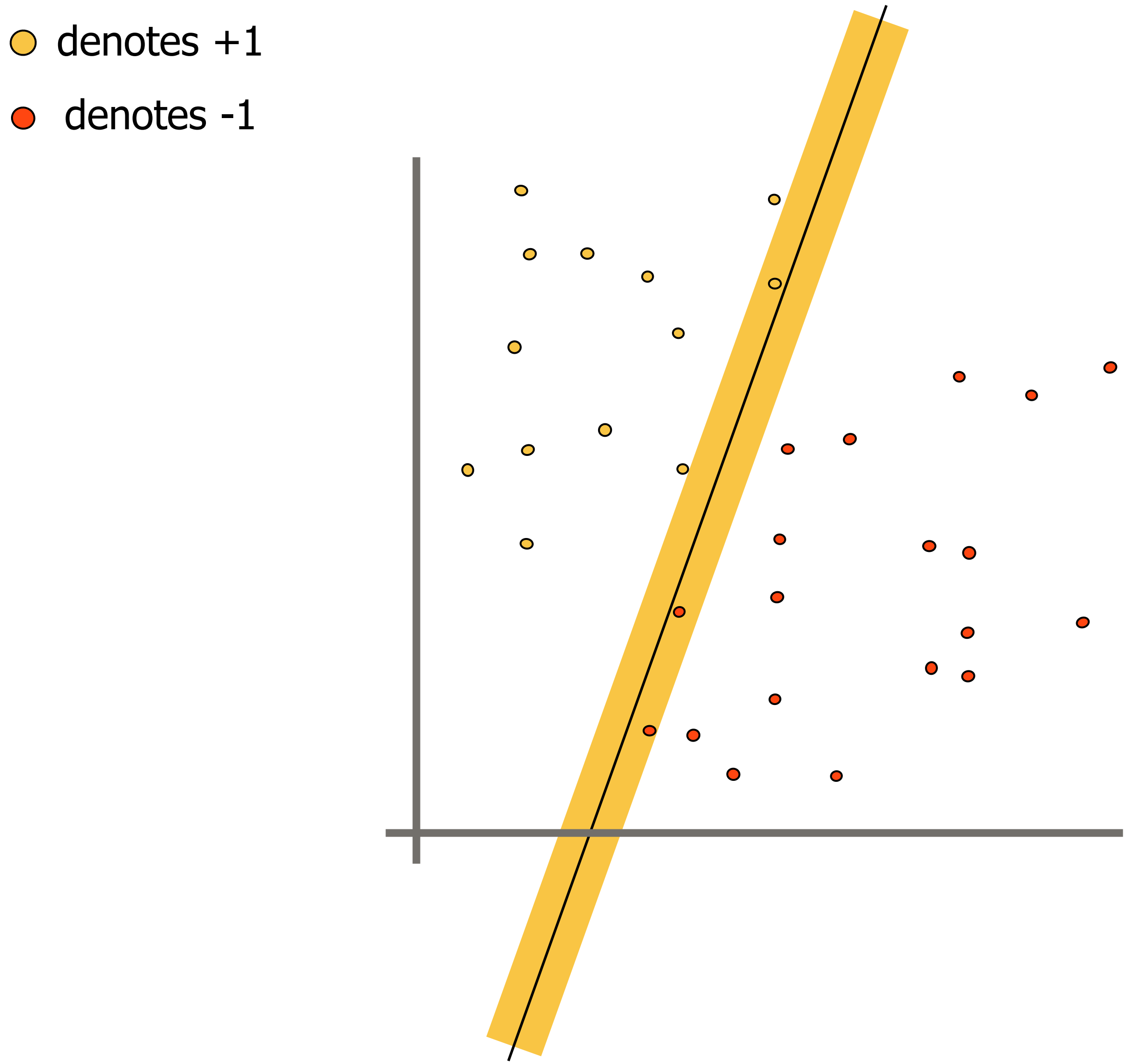
.. but which is best?

Classifier Margin



Define the margin of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.

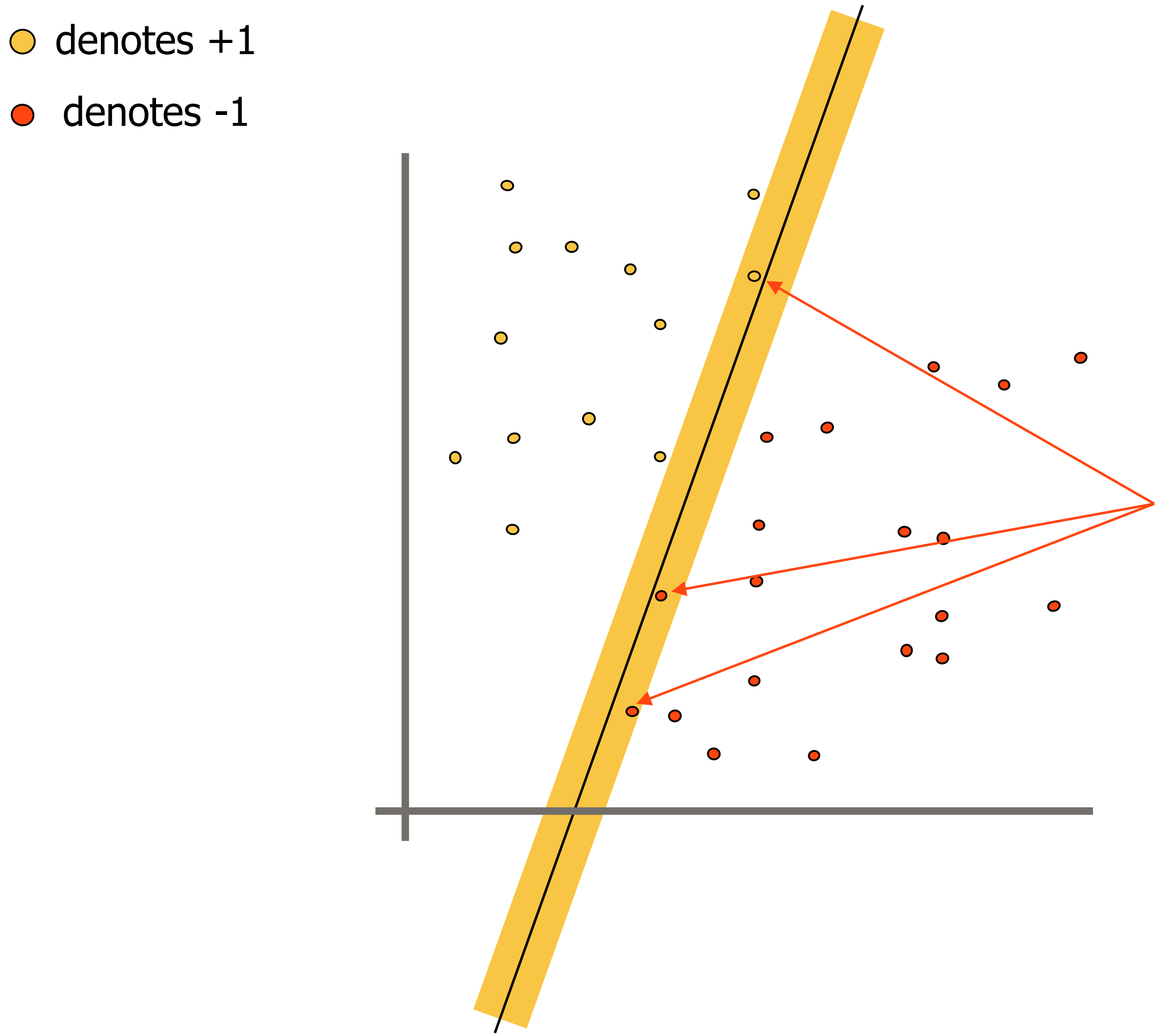
Maximum Margin



The maximum margin linear classifier is the linear classifier with the, maximum margin.

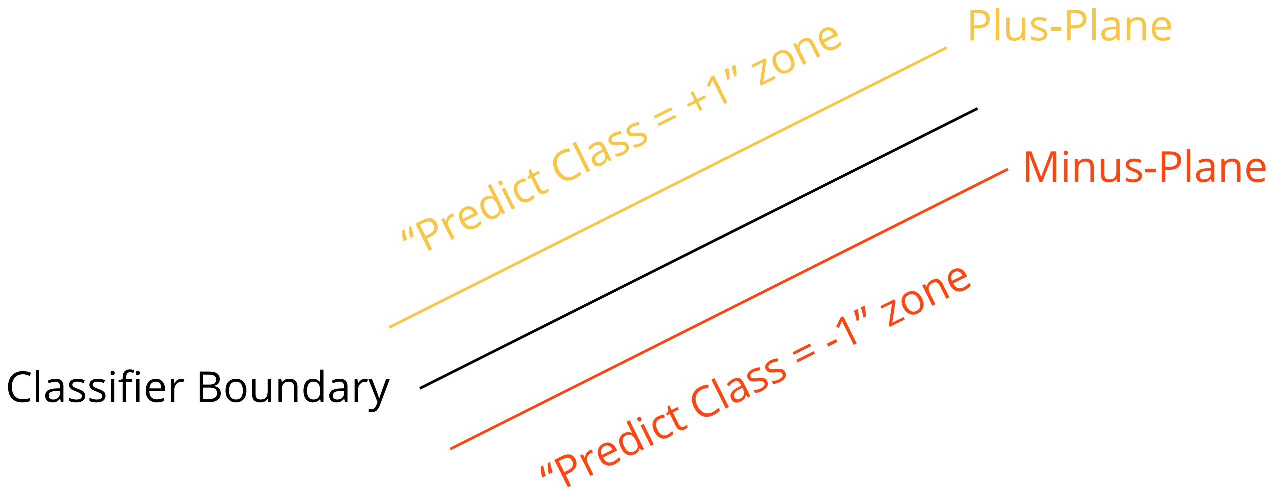
This is the simplest kind of SVM (Called a Linear SVM or LSVM)

Why Maximum Margin?



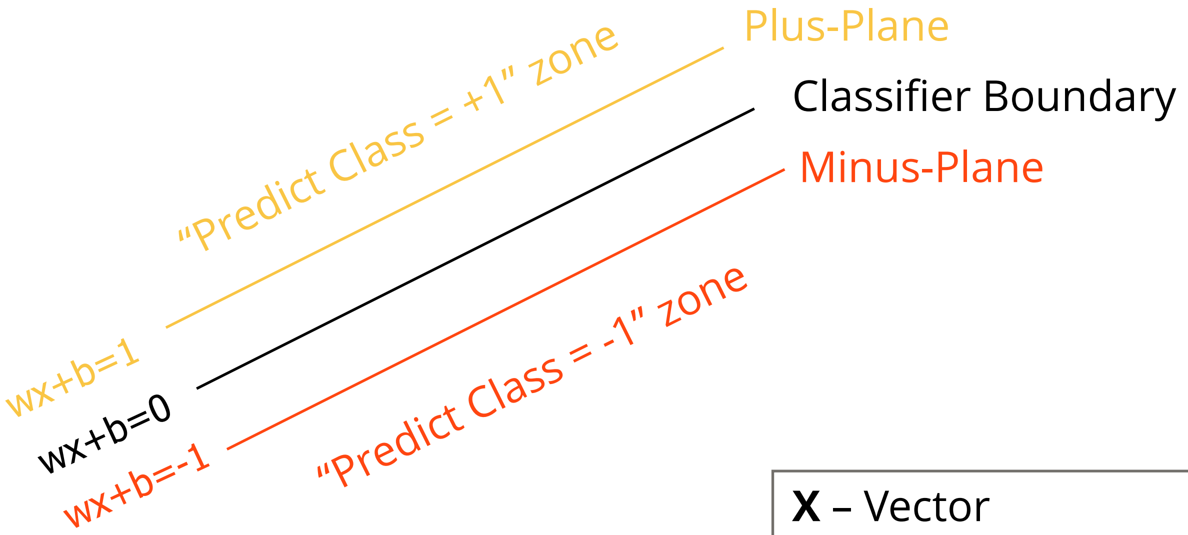
Support Vectors are those datapoints that the margin pushes up against.

Specifying a line and margin



- How do we represent this mathematically?
- ...in *m* input dimensions?

Specifying a line and margin

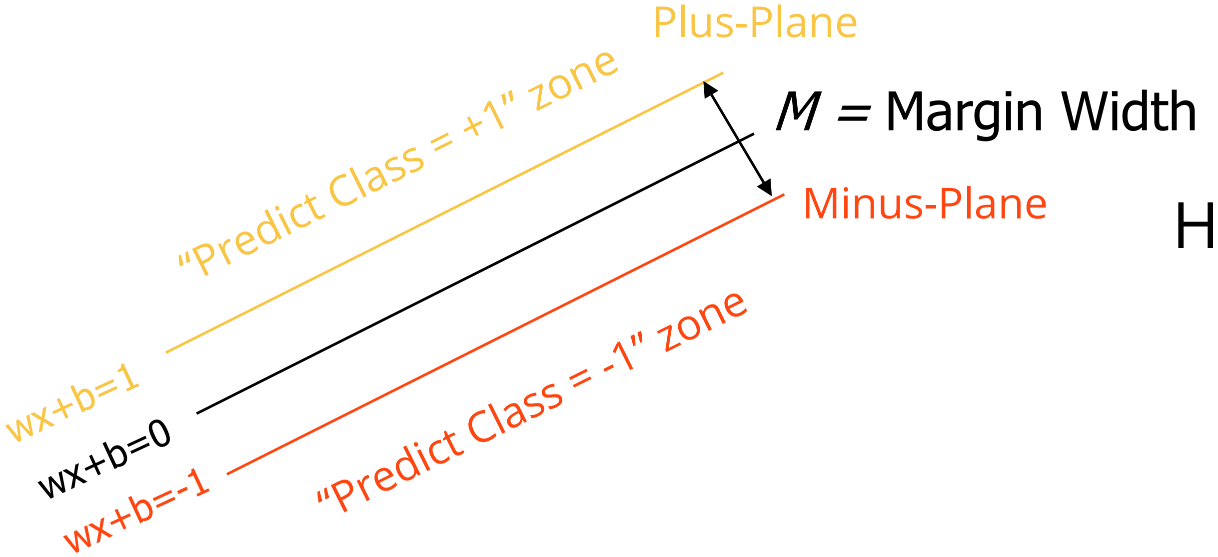


- Plus-plane = { $\mathbf{x}$: $\mathbf{w}$. $\mathbf{x}$ + b = +1 }
- Minus-plane = { $\mathbf{x}$: $\mathbf{w}$. $\mathbf{x}$ + b = -1 }

X – Vector
W – Normal Vector
b – Scale Value

Classify as..	+1	if	$\boldsymbol{w} \, . \, \boldsymbol{x} + b >= 1$
	-1	if	$\boldsymbol{w} \, . \, \boldsymbol{x} + b <= -1$
	Universe explodes	if	$-1 < \boldsymbol{w} \, . \, \boldsymbol{x} + b < 1$

Specifying a line and margin

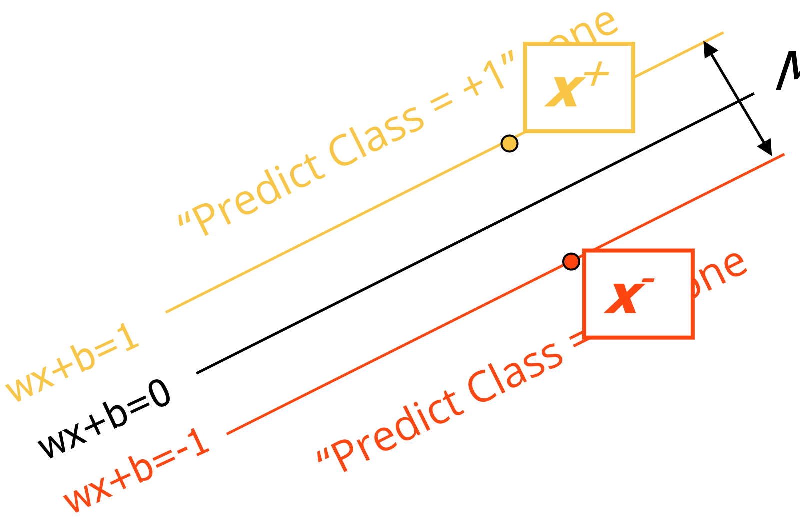


How do we compute M in terms of $\boldsymbol{w}$ and b?

- Plus-plane = $\{ \boldsymbol{x} : \boldsymbol{w} \cdot \boldsymbol{x} + b = +1 \}$
- Minus-plane = $\{ \boldsymbol{x} : \boldsymbol{w} \cdot \boldsymbol{x} + b = -1 \}$

Claim: The vector $\boldsymbol{w}$ is perpendicular (ตั้งฉาก) to the Plus Plane.

Specifying a line and margin

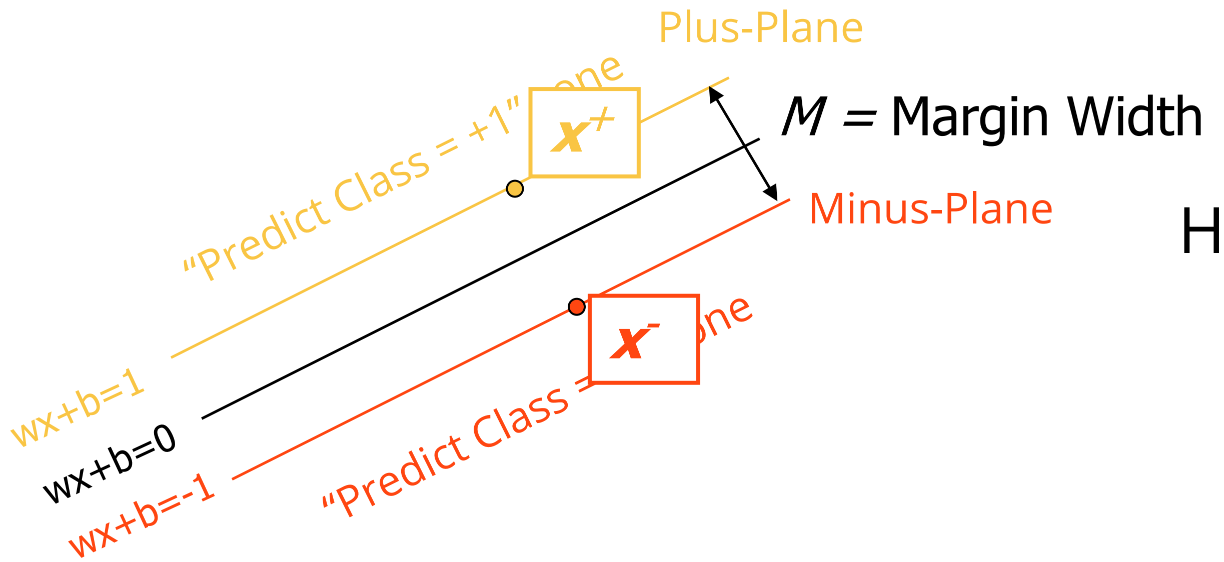


How do we compute M in terms of $\boldsymbol{w}$ and b?

- Plus-plane = $\{ \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + b = +1 \}$
- Minus-plane = $\{ \boldsymbol{x} : \boldsymbol{w} \,.\, \boldsymbol{x} + b = -1 \}$

Claim: The vector $\boldsymbol{w}$ is perpendicular (ตั้งฉาก) to the Plus Plane.

Specifying a line and margin



How do we compute M in terms of ***w*** and b?

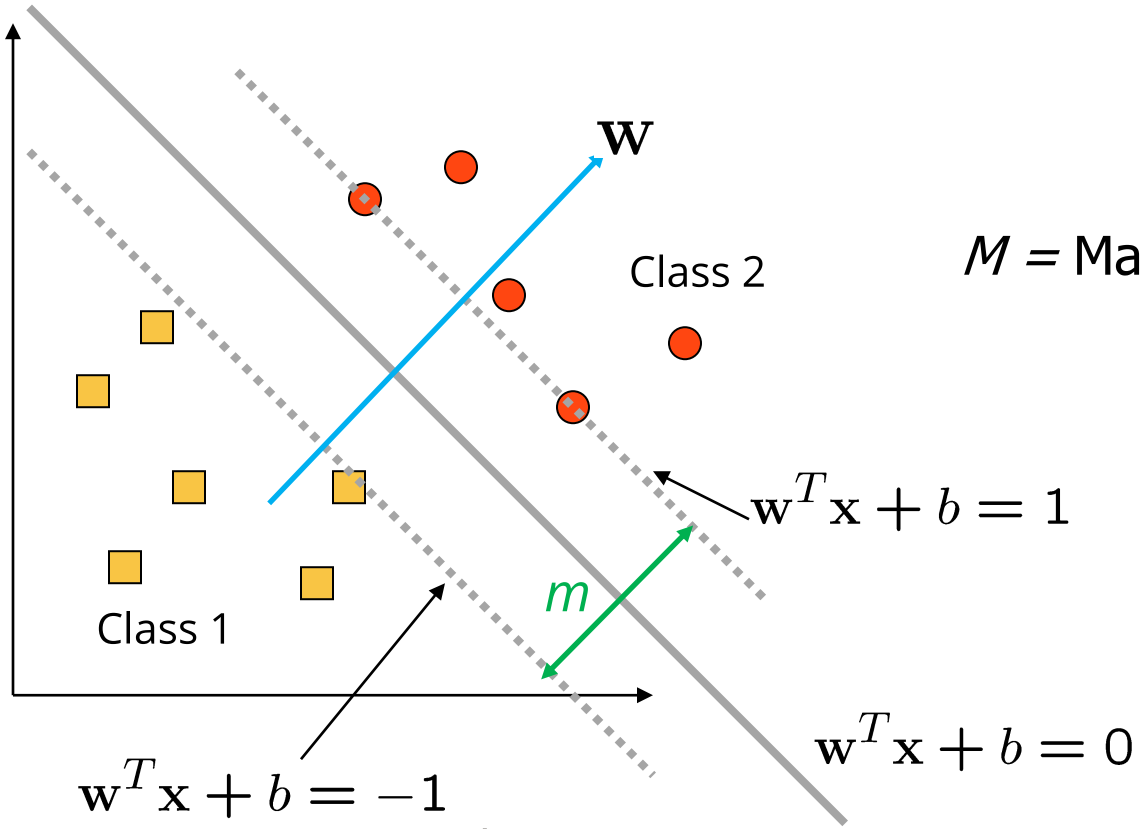
- What is the distance expression for a point **x** to a line **wx**+b= 0?

$$d(\mathbf{x}) = \frac{\left|\mathbf{x}\cdot\mathbf{w}+b\right|}{\sqrt{\left\|\mathbf{w}\right\|_2^2}} = \frac{\left|\mathbf{x}\cdot\mathbf{w}+b\right|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

X – Vector	
W – Normal Vector	
b – Scale Value	

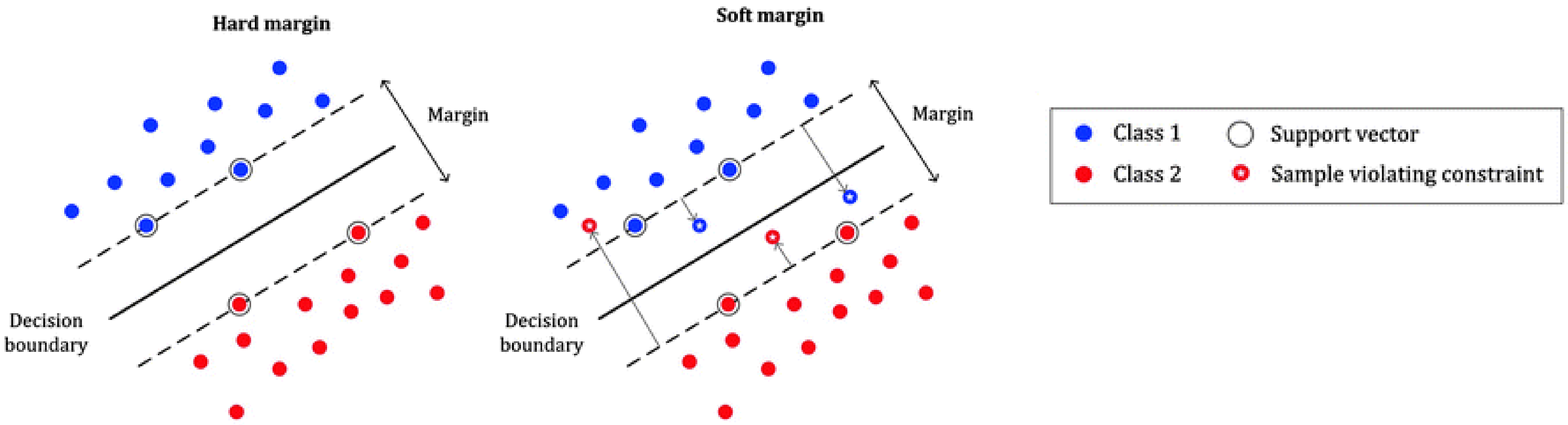
Large-margin Decision Boundary

- The decision boundary should be as far away from the data of both classes as possible
  - We should maximize the margin, *m*
  - Distance between the origin and the line $\mathbf{w}^t\mathbf{x}$



$$M = \text{Margin Width} = \frac{2}{\sqrt{\mathbf{w}.\mathbf{w}}}$$

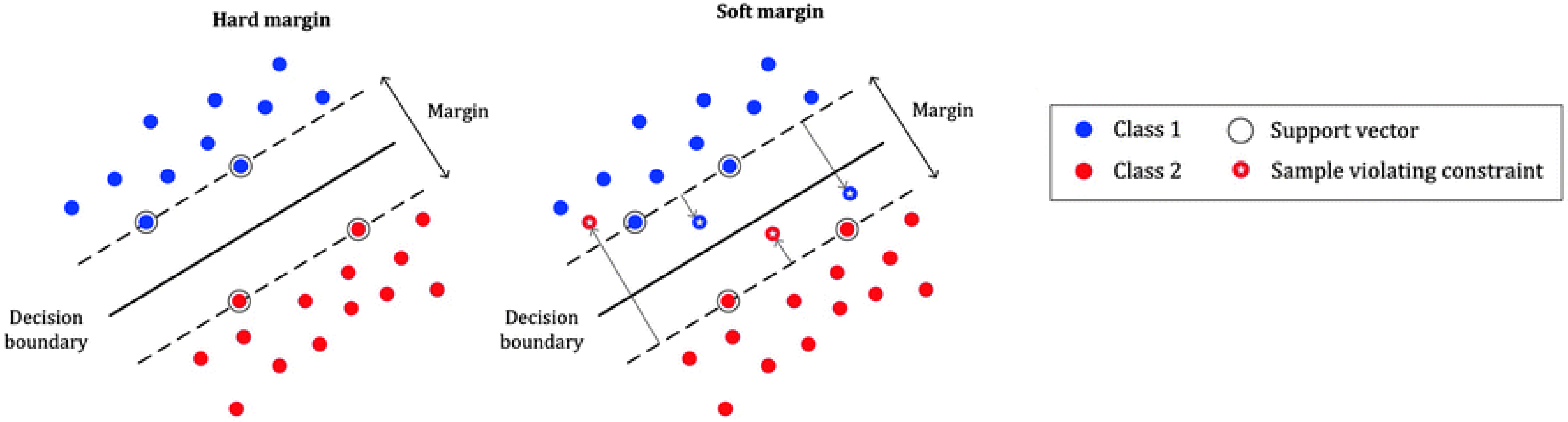
Robustness of soft margin and hard margin



- Hard margin does not allow any misclassification to happen.
- In case our data is non-separable/ nonlinear then the Hard margin SVM will not return any hyperplane.
- Soft margin allows some misclassification to happen by relaxing the hard constraints of SVM.
- Soft margin SVM is implemented with the help of the **Regularization parameter (C).**
- **Trade-off: width of the margin vs. number of training errors committed by the decision boundary.**

https://ankitnitjsr13.medium.com/math-behind-svm-support-vector-machine-864e58977fdb

Robustness of soft margin and hard margin (contd.)

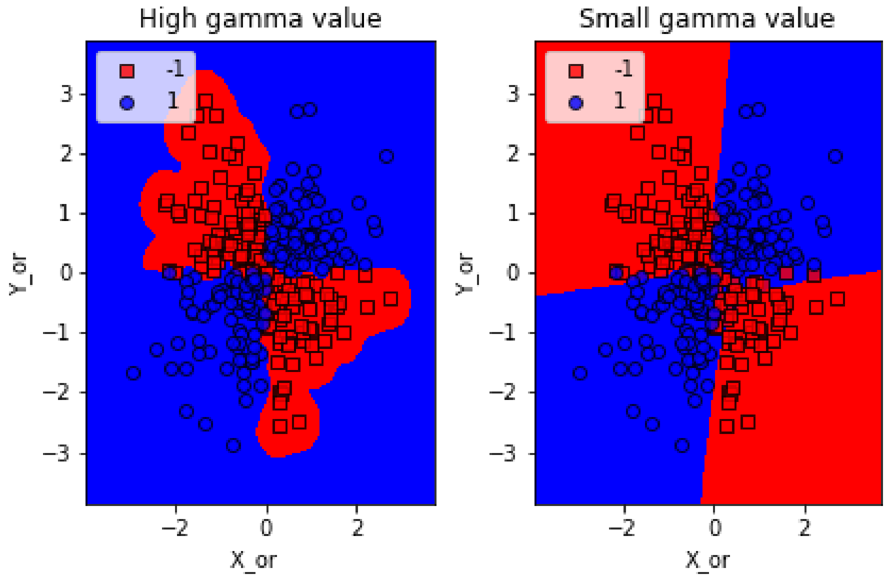


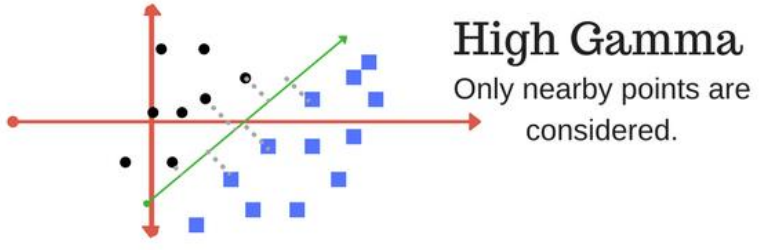
- As the value of C increases the margin decreases thus Hard SVM.
- If the values of C are very small the margin increases thus Soft SVM.
- Large value of C can cause overfitting therefore we need to select the correct value using Hyperparameter Tuning.

https://ankitnitjsr13.medium.com/math-behind-svm-support-vector-machine-864e58977fdb

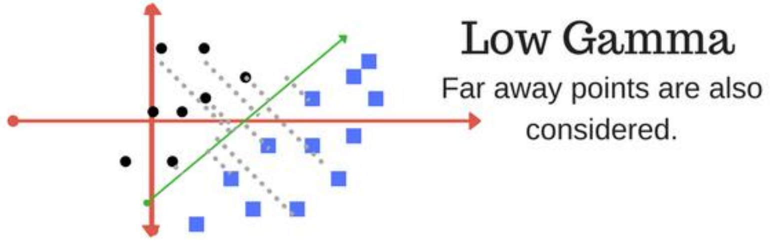
Other Parameters of SVM (Gamma values)

- It tells us how much will be the influence of the individual data points on the decision boundary.
    - **Large Gamma:** Fewer data points will influence the decision boundary.
    - Therefore, decision boundary becomes non-linear leading to overfitting
    - **Small Gamma:** More data points will influence the decision boundary.
    - Therefore, the decision boundary is more generic.





High Gamma



Low Gamma

https://www.analyticsvidhya.com/blog/2021/04/insight-into-svm-support-vector-machine-along-with-code/

Types of SVMs

There are 2 different types of SVMs, each used for different things:

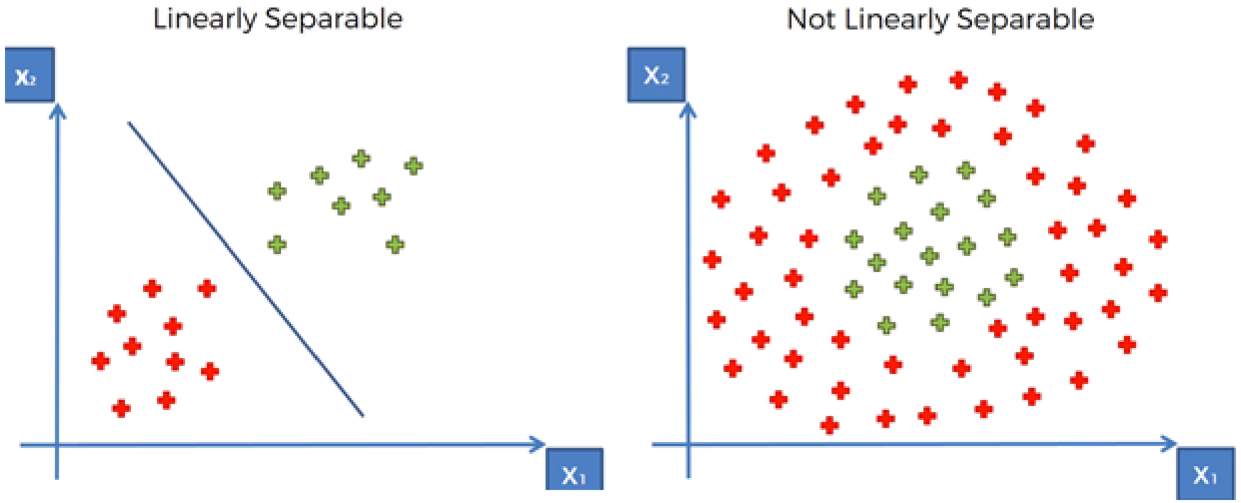
• **Simple SVM:** Typically used for linear regression and classification problems.

• **Kernel SVM:** Has more flexibility for non-linear data because you can add more features to fit a hyperplane instead of a two-dimensional space.

Types of SVMs

There are 2 different types of SVMs, each used for different things:

- **Simple SVM:** Typically used for linear regression and classification problems.
- **Kernel SVM:** Has more flexibility for non-linear data because you can add more features to fit a hyperplane instead of a two-dimensional space.

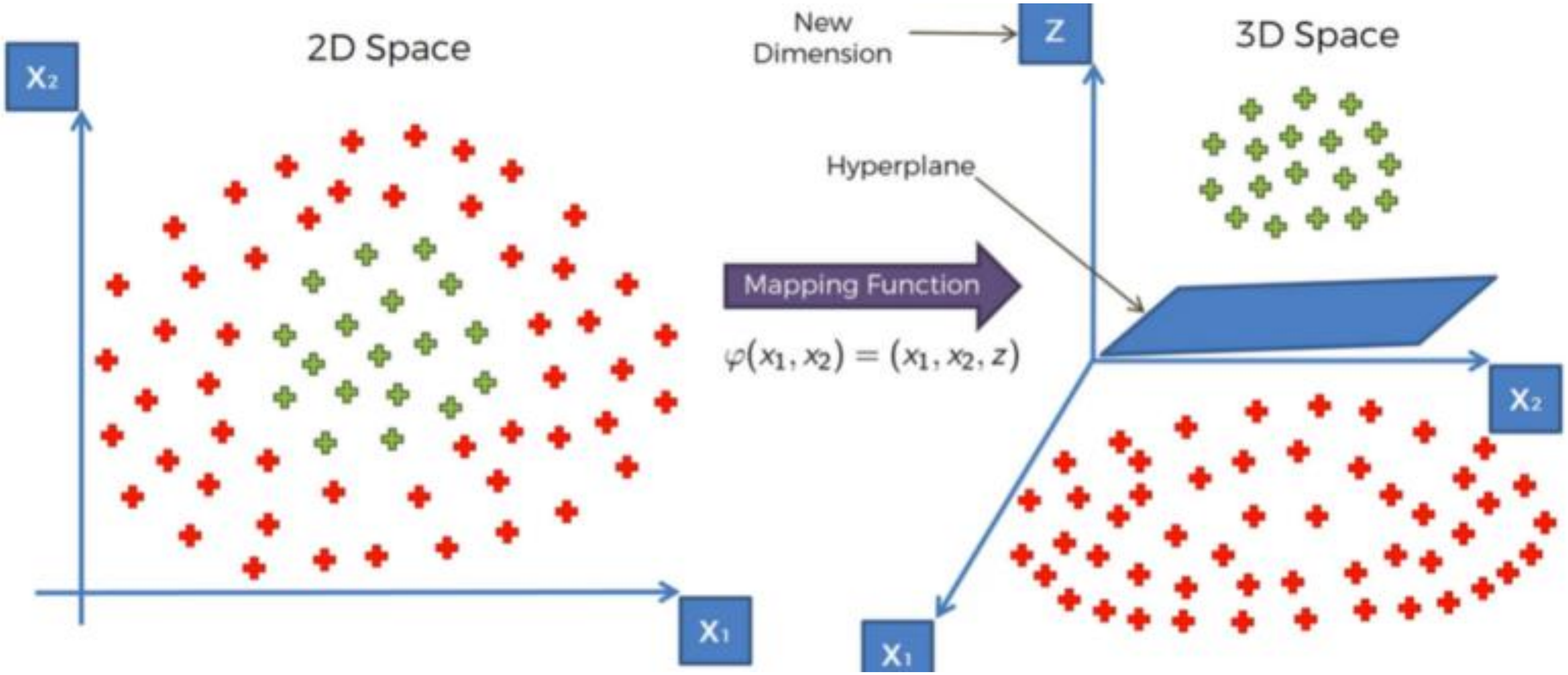


We have to separate green points from the red points by drawing a circle around them and to do so we use the **Gaussian RBF Kernel Function.**

https://medium.com/pursuitnotes/day-12-kernel-svm-non-linear-svm-5fdefe77836c

Gaussian RBF Kernel Function

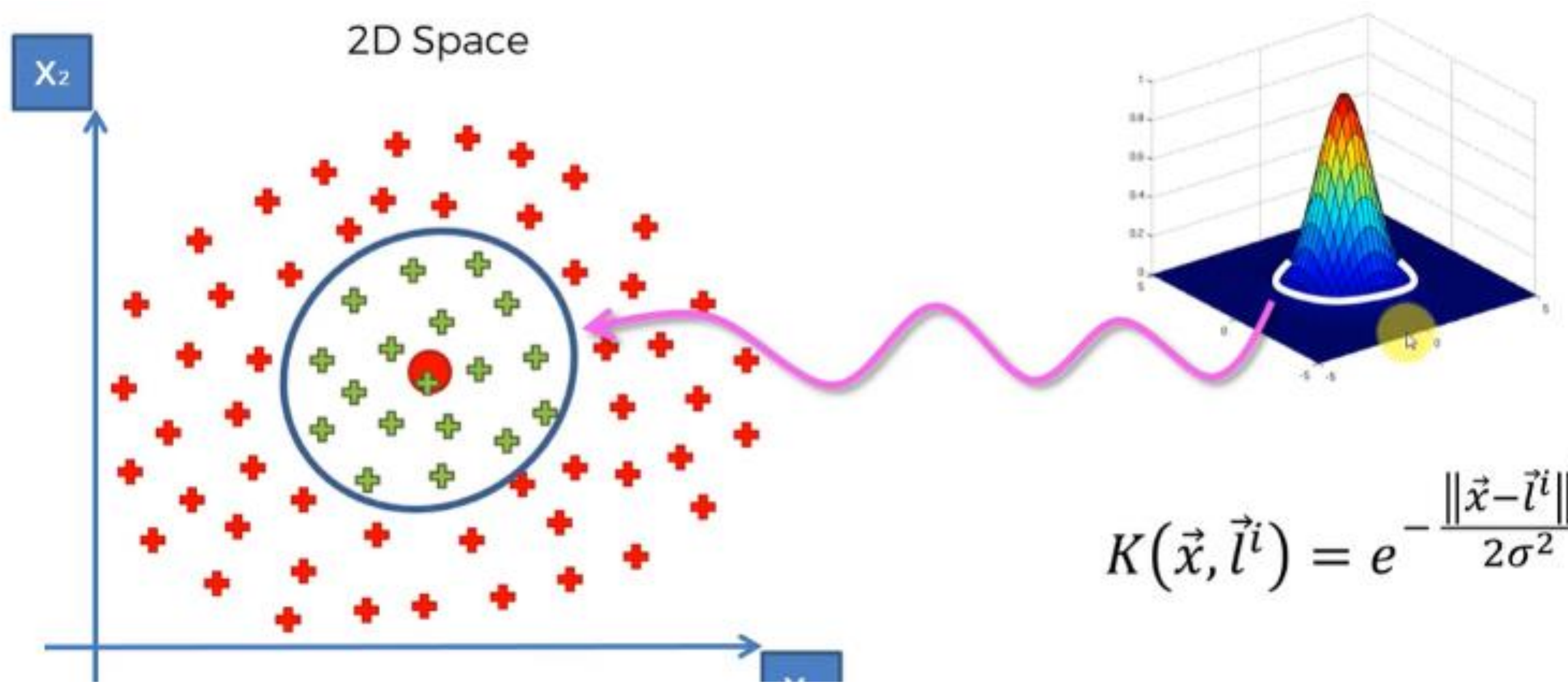
We can *shift* the points from a 2D plane to a 3D plane by just shifting all the green points above the red ones by using a mapping function like gaussian RBF.



https://medium.com/pursuitnotes/day-12-kernel-svm-non-linear-svm-5fdefe77836c

Gaussian RBF Kernel Function (contd.)

We can *shift* the points from a 2D plane to a 3D plane by just shifting all the green points above the red ones by using a mapping function like gaussian RBF.

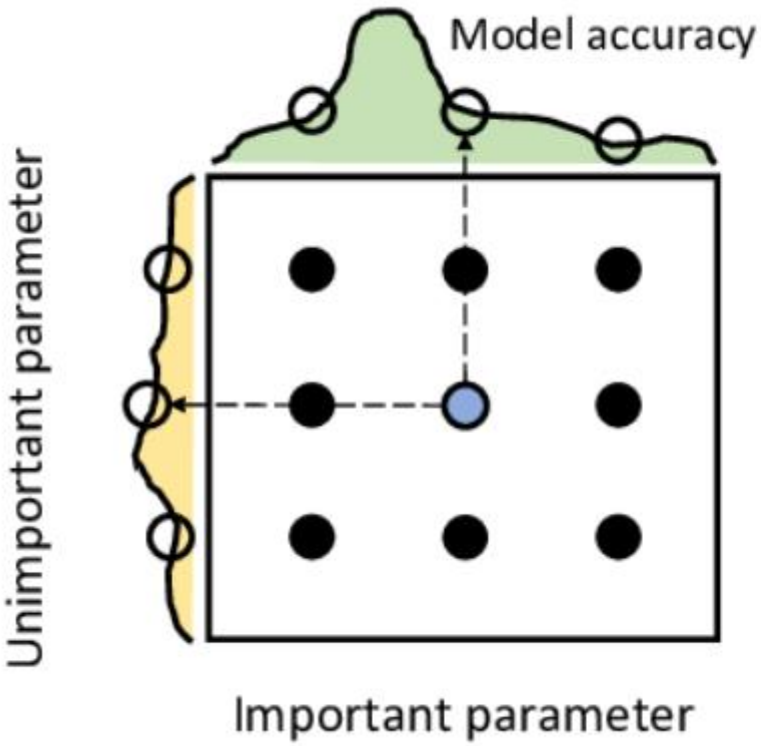


$$K(\vec{x}, \vec{l^i}) = e^{-\frac{\left\|\vec{x}-\vec{l^i}\right\|^2}{2\sigma^2}}$$

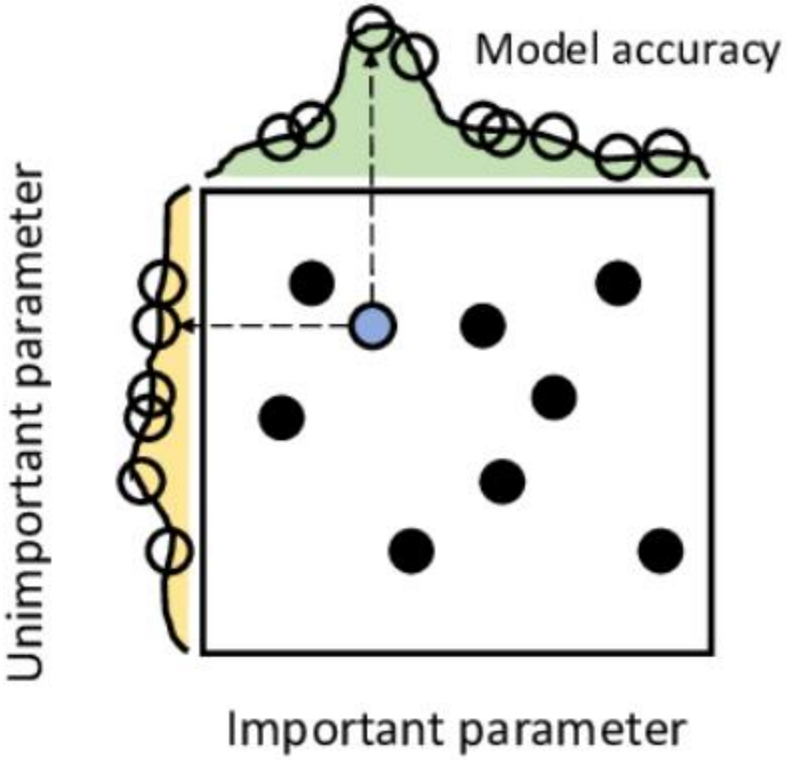
https://medium.com/pursuitnotes/day-12-kernel-svm-non-linear-svm-5fdefe77836c

Grid search

Grid-search is used to find the optimal hyperparameters of a model which results in the most 'accurate' predictions.



Grid search



Random search

https://medium.com/pursuitnotes/day-12-kernel-svm-non-linear-svm-5fdefe77836c