

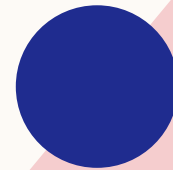
# **SOFT COMPUTING**

Aya Mohamed Saad

Eng/Noha Hamdy

# CONTENTS

- Naïve Bayes Classifier
- KNN Classifier
- SVM Classifier



- Naive Bayes, K-Nearest Neighbors (KNN), and Support Vector Machines (SVM) are all popular machine learning algorithms used for classification tasks. Here are some points of comparison between these algorithms:

### 1. Algorithm Type:

- Naive Bayes is a probabilistic algorithm based on Bayes' theorem.
- KNN is a lazy learning algorithm that stores all the training data points in memory and classifies new points based on their proximity to these stored points.
- SVM is a discriminative algorithm that finds the optimal hyperplane in a high-dimensional space that separates the different classes.

## 2. Training Time:

- Naive Bayes is a simple algorithm that can be trained very quickly on large datasets.
- KNN requires no training phase, but can be slow at classification time due to its computational complexity.
- SVM can take longer to train on large datasets, but its training time is still relatively fast compared to other algorithms.

## 3. Performance on Small and Large Datasets:

- Naive Bayes performs well on small datasets but may not be as effective on larger datasets due to the assumption of independence between features.
- KNN can perform well on small datasets but may become less effective on larger datasets due to the curse of dimensionality.
- SVM can perform well on both small and large datasets.

#### 4. Handling of Non-Linear Data:

- Naive Bayes assumes that the features are independent and linearly related, and may not perform well on non-linear data.
- KNN can handle non-linear data but may suffer from the curse of dimensionality in high-dimensional spaces.
- SVM can handle non-linear data by using kernel functions to transform the data into a higher-dimensional space where it can be separated linearly.

#### 5. Interpretability:

- Naive Bayes is a simple algorithm that provides good interpretability of the results.
- KNN is also relatively interpretable, as it simply classifies points based on their proximity to other points.
- SVM can be more difficult to interpret due to the complexity of the hyperplane it generates.

Example: Suppose we have a dataset of emails that are either spam or not spam, and we want to classify new emails as either spam or not spam. Here are some ways that the three algorithms could be used:

- Naive Bayes: We can use the occurrence of certain words in the email as features, and use Naive Bayes to compute the probability that the email is spam or not spam based on these features. For example, if an email contains the words "free" and "prize", it may have a higher probability of being spam.
- KNN: We can use the occurrence of certain words in the email as features, and classify new emails based on their similarity to the training data. For example, if an email has many words in common with previously identified spam emails, it may be classified as spam.
- SVM: We can use the occurrence of certain words in the email as features, and use SVM to find the optimal hyperplane that separates spam emails from non-spam emails. For example, if an email contains many words associated with spam, it may be classified as spam.



**THANK YOU**