Machine learning algorithms

1. **K-nearest neighbors:**

* How it works:

It calculates the euclidean distance between the point and every other one and then chooses the class as the k nearest ones and it has no training phase.

* Pros:

No training phase and easy to understand.

* Cons:

Can take a lot of time to calculate if the dataset is big as it needs to calculate the distance between each point.

1. **Decision tree:**

* How it works:

**The training phase:**

The model is trained by recursively splitting the data based on the feature values.

**Prediction phase:**

When a new data is entered the tree is traversed until reaching a leaf node where the class lies.

* Pros:

The tree can be visualized, able to handle both numerical and categorical data.

* Cons:

Can create over-complexed trees that don’t generalize the data (overfitting), sensitive to small variation in data.

1. **Naïve bayes:**

* How it works:

**The training phase:**

The model calculates the probability of a class then calculates the likelihood of each feature given the class using bayes theorem where it assumes the independency of the features.

**Prediction phase:**

It calculates the probability of probability of each class given the features and then chooses the highest probability of them.

* Pros:

Fast, easy to understand and implement

* Cons:

The assumption that features are independent.

**Conclusion:**

There is no single best algorithm they are all good in their part like you should use the knn when dealing with small datasets, decision tree when dealing with mixed data types numerical and categorical, naïve bayes is best used when dealing with features that are independent.