k-NN is one of the simplest supervised learning algorithms and methods in machine learning. It stores all of the available examples and then classifies the new ones based on similarities in distance metrics.

k-NN is a famous classification algorithm and a lazy learner. What does a lazy learner mean?

K-nearest neighbor mainly stores the training data. It doesn’t do anything else during the training process. That’ s why it is lazy.

k-NN just stores the labeled training data. When new unlabeled data arrives, kNN works in 2 main steps:

Looks at the k closest training data points (the k-nearest neighbors).

Then, as it uses the k-nearest neighbors, k-NN decides how the new data should be classified.

How does k-NN know what’s closer?

It uses density-based anomaly detection methods. For continuous data (see continuous vs discrete data), the most common distance measure is the Euclidean distance. For discrete data, Hamming distance is a popular metric for the “closeness” of 2 text strings.

The pick of distance metric depends on the data.

The k-NN algorithm works very well for dynamic environments where frequent updates are needed. In addition, density-based distance measures are good solutions for identifying unusual conditions and gradual trends. This makes k-NN useful for outlier detection and defining suspicious events.

k-NN also is very good techniques for creating models that involve non-standard data types like text.

k-NN is one of the proven anomaly detection algorithms that increase the fraud detection rate. It is also one of the most known text mining algorithms out there.

It has many applications in business and finance field. For example, k-NN helps for detecting and preventing credit card fraudulent transactions.

<https://journals.sagepub.com/doi/full/10.1177/17483026221078111>

1. K-Nearest Neighbor

K-nearest neighbor (k-NN) is a modest and conventional nonparametric technique for classifying samples. It calculates the approximate distances between points on the input vectors and then assigns the unlabeled point to its K-nearest neighbors’ class.

Pros

It is very easy to understand when there are few predictor variables.

It is useful for building models that involve non-standard data types, such as text.

Cons

Have large storage requirements.

It is sensitive to the choice of the similarity function that is used to compare instances.

It lacks a principled way to choose k, except through cross-validation or similar.

Computationally-expensive technique.

<https://www.fromthegenesis.com/pros-and-cons-of-k-nearest-neighbors/>