PREVOTEAU HUGO （雨果）
student id : 6219000083

Learning Note

1 – K-Nearest Neighbours (K-NN)

Discovery of the K Nearest Neighbours Algorithm

The K-NN (K-nearest neighbours) algorithm is a supervised learning method. It can be used for regression as well as for classification. Its operation can be likened to the following analogy "tell me who your neighbours are, I'll tell you who you are".
To make a prediction, the K-NN algorithm will not compute a predictive model from a Training Set as is the case for logistic regression or linear regression. Indeed, K-NN does not need to build a predictive model. Thus, for K-NN there is no actual learning phase. Therefore, it is sometimes categorized in “Lazy Learning”. In order to make a prediction, K-NN relies on the data set to produce a result.

How does K-NN make a prediction?

To make a prediction, the K-NN algorithm will base itself on the entire dataset. Indeed, for an observation, which is not part of the dataset, that we want to predict, the algorithm will look for the K instances of the dataset closest to our observation. Then for these K neighbours, the algorithm will use their output variables y to compute the value of the variable y of the observation we want to predict.

Furthermore:

- If K-NN is used for the regression, the mean (or median) of the variables y of the K closest observations will be used for prediction.
- If K-NN is used for classification, the mode of the variables y of the closest K observations will be used for prediction.

Algorithmic writing

We can schematize the operation of K-NN by writing it in the following pseudo-code:

Start Algorithm

Input data:

- *A set of data D.*
- *A distance definition function d.*
- *An integer number K.*

For a new observation X for which we want to predict its output variable

Do:

1. *Calculate all the distances of this observation X with the other observations in the dataset D.*
2. *Retain the K observations from the dataset D close to X using the distance calculation function d.*
3. *Take the values of y of the K selected observations:*
   - *If a regression is performed, calculate the mean (or median) of the y deductions.*
   - *If a classification is performed, calculate the method of the y deductions.*
4. *Return the value calculated in step 3 as the value that was predicted by K-NN for the observation X.*

End Algorithm

PREVOTEAU HUGO （雨果）
student id : 6219000083

Similarity calculation in the K-NN algorithm

As we have just seen in our algorithm writing, K-NN needs a function to calculate the distance between two observations. The closer two points are to each other, the more similar they are and vice versa.

There are several distance calculations functions, including Euclidean distance, Manhattan distance, Minkowski distance, Jaccard distance, Hamming distance...etc. The distance function is chosen according to the types of data being manipulated. Thus, for quantitative data (example: weight, wages, height, amount of electronic basket etc...) and of the same type, the Euclidean distance is a good candidate. As for the Manhattan distance, it is a good measure to use when the data (input variables) are not of the same type (example: age, sex, length, weight etc...).

There is no need to code these distances yourself, usually Machine Learning libraries such as “Scikit Learn” do these calculations internally. You just must indicate the distance measurement you want to use.

For the curious, here are the mathematical definitions of the distances just mentioned.

The Euclidean distance formula:

$$D_e(x, y) = \sqrt{\sum_{j=1}^{n}(x_j - y_j)^2}$$

the distance that calculates the square root of the sum of the square differences between the coordinates of two points:

Manhattan distance formula:

$$D_m(x, y) = \sum_{i=1}^{k} \mid x_i - y_i \mid$$

the Manhattan distance: calculates the sum of the absolute values of the differences between the coordinates of two points:

Distance Hamming formula:

$$D_h(x, y) = \sum_{i=1}^{k} \mid x_i - y_i \mid$$

the distance between two given points is the maximum difference between their coordinates on one dimension.

With:

- $x = y \implies D = 0$
- $x \neq y \implies D = 1$

PREVOTEAU HUGO （雨果）
student id : 6219000083

How to choose the K value?

The choice of the value K to be used to make a prediction with K-NN, varies according to the data set. As a rule, the fewer (smaller K) neighbors are used, the more underfitting will occur. On the other hand, the more neighbours (a large K number) we use, the more reliable our prediction will be. However, if we use a K number of neighbours with $K = N$ and N being the number of observations, we risk overfitting and consequently having a model that does not generalize well on observations it has not yet seen.

K-NN limitations

K-NN is a simple algorithm. Mainly because it doesn't need a model to be able to make a prediction. The counter cost is that it must keep in memory all the observations to be able to make its prediction. Thus, it is necessary to pay attention to the size of the training game.

Also, the choice of the method of calculation of the distance as well as the number of neighbours $\boldsymbol{K}$ may not be obvious. It is necessary to try several combinations and tuning of the algorithm to have a satisfactory result.

Conclusion

- K-NN stores the entire dataset to make a prediction
- K-NN does not compute any predictive models and is within the scope of “Lazy Learning”
- K-NN makes just-in-time (on-the-fly) predictions by calculating the similarity between an input observation and the different observations in the dataset.

PREVOTEAU HUGO （雨果）
student id : 6219000083

2 – K-means

What's K-means

K-means is an unsupervised non-hierarchical clustering algorithm. It allows you to cluster the observations of the data set into K separate clusters. Thus, similar data will be found in the same cluster. Moreover, an observation can only be found in one cluster at a time (exclusivity of membership). The same observation cannot belong to two different clusters.

Notion of similarity

To regroup a dataset into K distinct clusters, the K-Means algorithm needs a way to compare the degree of similarity between the different observations. Thus, two data that are similar will have a smaller distance of dissimilarity, while two different objects will have a larger distance of separation.

Mathematical and statistical literature is full of distance definitions, the best known for clustering cases are:

The Euclidean distance: This is the geometric distance. Either a matrix X with n quantitative variables. In vector space E^n. The Euclidean distance d between two observations x_1 and $x_2$is calculated as follow:

$$d(x_1, x_2) = \sqrt{\sum_{j=1}^{n}(x_{1n} - x_{2n})^2}$$

The Manhattan taxi-distance is the distance between two points travelled by a taxi when it travels in a city where the streets are arranged in a network or grid. A taxiway is the distance a taxi travels when it moves from one node of the network to another using the horizontal and vertical movements of the network. The formula is:

$$D_m(x, y) = \sum_{i=1}^{k} \mid x_i - y_i \mid$$

Choose K: the number of clusters

Choosing a number of clusters K is not necessarily intuitive. Especially when the dataset is large, and you don't have a priori or assumptions about the data. A large number K can lead to an overly fragmented partitioning of the data. This will prevent the discovery of interesting patterns in the data. On the other hand, too small a number of clusters will lead to potentially too generalized clusters containing a lot of data. In this case, there will be no "fine" patterns to discover.

For the same set of data, there is no single clustering possible. The difficulty will therefore lie in choosing a number K of clusters that will allow to highlight interesting patterns between the data. Unfortunately, there is no automated process to find the right number of clusters.

The most common method for choosing the number of clusters is to run K-Means with different values of K and calculate the variance of the different clusters. The variance is the sum of the distances between each centroid of a cluster and the different observations included in the same cluster. Thus, we try to find a number of clusters K so that the selected clusters minimize the distance between their centers (centroids) and the observations in the same cluster. This is called minimization of the intra-class distance.

The variance of the clusters is calculated as follow:

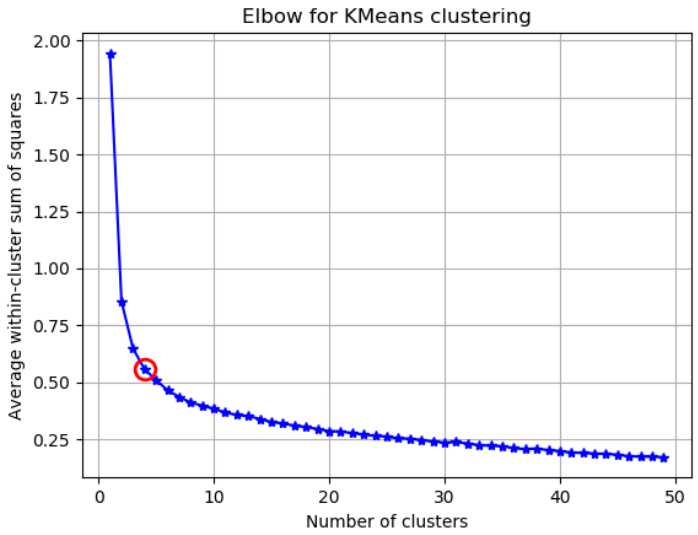
PREVOTEAU HUGO （雨果）
student id : 6219000083

$$V = \sum_j \sum_{x_i \to c_j} D(c_j, x_i)^2$$

With:

- c_j : The center of the cluster (the centroid)
- x_i : the ith observation in the cluster having for centroid c_j
- $\mathrm{D}(c_j, x_i)$: the distance (Euclidean or other) between the center of the cluster and the point x_i

Usually, by putting in a graph the different numbers of clusters K according to the variance, we find a graph similar to this one:



On this graph, we can see the shape of an arm where the highest point represents the shoulder and the point where K equals 9 represents the other end: the hand. The optimal number of clusters is the point representing the elbow. Here the elbow can be represented by K being 2 or 3. This is the optimal number of clusters. Generally, the point of the bend is the point of the number of clusters from which the variance no longer reduces significantly. Indeed, the "fall" of the variance curve (distortion) between 1 and 3 clusters is significantly greater than that between 5 clusters and 9 clusters. The fact of looking for the point representing the elbow, gave name to this method: The Elbow method.

Finally, the choice (in view of this graph), between 2 or 3 clusters, remains a little vague and at your discretion. The choice will depend on your dataset and what you are trying to accomplish. Finally, there is no single solution to a clustering problem.

K-means use case

K-Means in particular and clustering algorithms in general all have a common goal: to group similar elements into clusters. These elements can be anything and everything, as long as they are encoded in a data matrix.

PREVOTEAU HUGO （雨果）
student id : 6219000083

The fields of application of K-Means are numerous, it is notably used in:

- Customer segmentation according to a certain criterion (demographics, buying habits etc...)
- Using clustering in Data Mining during data mining to detect similar individuals. Usually, once these populations are detected, other techniques can be used depending on the need.
- Document Clustering (grouping of documents according to their contents. Think about how Google News groups documents by topic).

How the K-Means algorithm works

K-means is an iterative algorithm that minimizes the sum of the distances between each individual and the centroid. The initial choice of centroids determines the final result.
Admitting a cloud of a set of points, K-Means changes the points of each cluster until the sum can no longer decrease. The result is a set of compact and clearly separated clusters, subject to choosing the right value K for the number of clusters.

Algorithmic principle

K-means algorithm

Inputs:

- K the number of clusters to be formed
- The Training Set (data matrix)

BEGINNING

Randomly select K points (one line of the data matrix). These points are the centers of the clusters (called centroid).

REPEAT

- Assign each point (element of the data matrix) to the group it is closest to at its center.

- Recalculate the center of each cluster and modify the centroid

UNTIL CONVERGENCE

OR (stabilization of the total inertia of the population)

END ALGORITHM

The assignment of a point x to a cluster is done according to the distance of this point from the different K centroids. Moreover, this point x will be assigned to a cluster i if it is closer to its centroid (minimum distance). Finally, the distance between two points in the case of K-Means is calculated by the methods mentioned in the paragraph "notion of similarity".

Remarks on K-Means

Local Optimums

By analysing the way the K-means algorithm works, we notice that for the same dataset, we can have different partitioning. Indeed, the initialization of the very first K centroids is completely random. Therefore, the algorithm will find different clusters according to this first random initialization. Therefore, the configuration of the clusters found by K-Means may not be the most optimal. We are talking about local optimum.

PREVOTEAU HUGO （雨果）
student id : 6219000083

In order to overcome the problems of local optimums, it is sufficient to run K-means several times on the dataset (with the same number K of clusters and different initial initializations of the centroids) and see the composition of the clusters that form. Then keep the one that suits your needs.

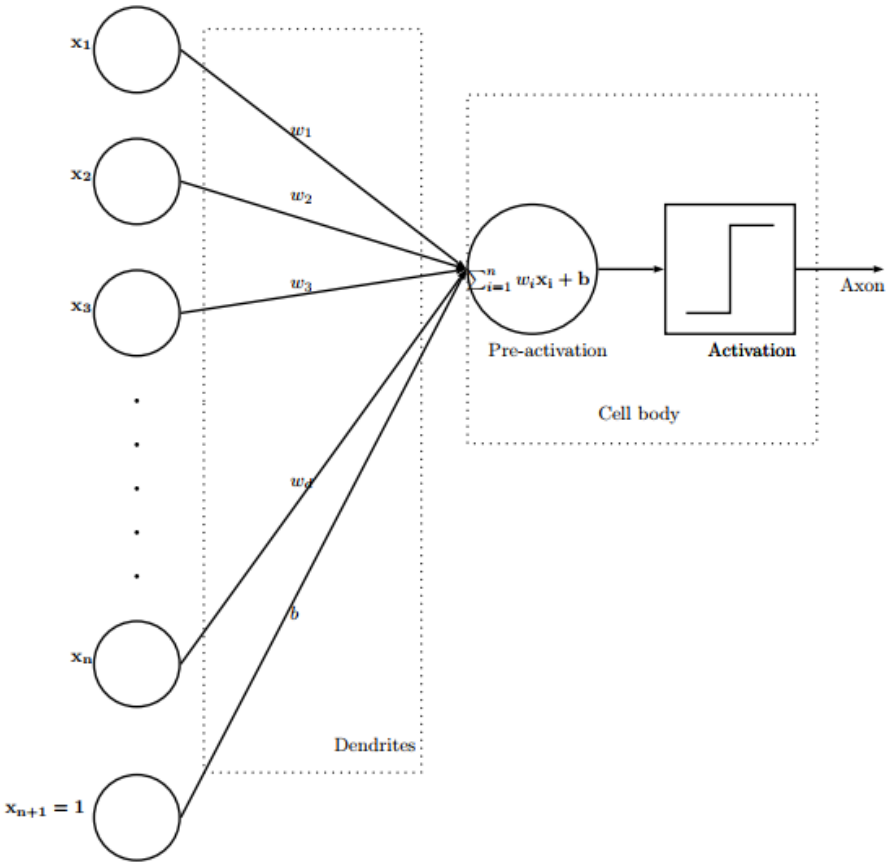
PREVOTEAU HUGO （雨果）
student id : 6219000083

3 – Neural Networks

A machine language neuron?

The concept of the artificial neuron is part of the research and action fields of Deep Learning. The objective of Deep Learning is to predict an output Y (a characteristic) through a set of input Xi data, called observations. So, we can imagine different shapes and predict if they correspond to a class. One of the ways to achieve this, highlighted by the research, was to simulate the response of a so-called "artificial" neuron to these observations and to develop an algorithm to process and weight the observations to predict a characteristic.

The graph below is the representation of an artificial neuron, the dendrites receive signals, the frequency of each signal depends on the W_i weight. These signals are then transmitted to the body of the cell. The latter is characterized by two important elements, the activation function and the activation condition. It is by checking the activation condition that the axon is activated to send signals back to the neurons that follow through their dendrites and so on.

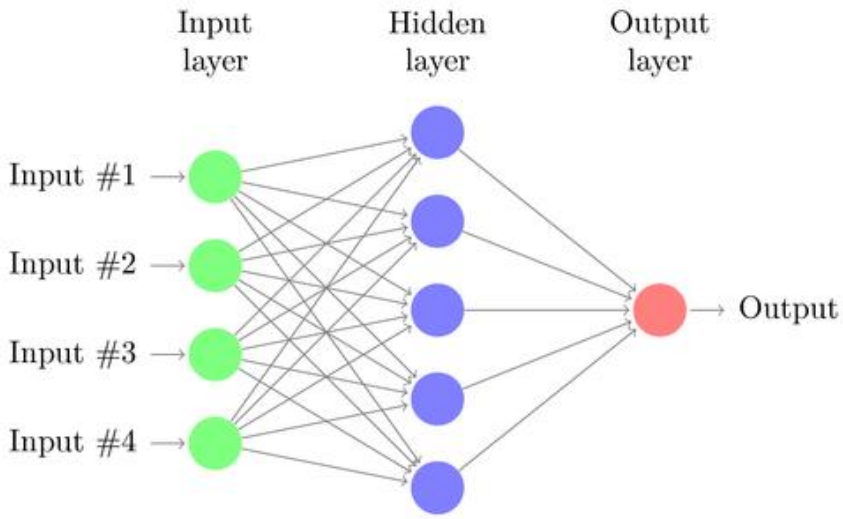


A deep neural network consists of thousands or even millions of neurons organized in layers interconnected by weighted links (W_i). Each node is characterized by the set:

$$\mathrm{Fi(activation\_function)} + \mathrm{Ci(activation\_condition)}$$

PREVOTEAU HUGO （雨果）
student id : 6219000083

The work of learning is therefore to find the optimal combination of $W_i/F_i/C_i$ to generate the best results in terms of prediction and estimation.



(An algorithm such as a neural network is too long to be shown here)

Backpropagation: How is this learning done in practice?

Learning phase

In this step, we will first try to build a model that will serve us in a second step to make predictions. The idea is to determine the right combination of $W_i/F_i/C_i$. To do this, we define an error function that calculates the difference between the actual output of the network and its expected output after a case has circulated through the network.

Let Y and Y' be two vectors respectively the actual output and the expected output of the network.

The backpropagation error is $E(Y, Y') = (1/2) * ||Y - Y'||^2$

Through this loss function, we try to optimize the model (i.e. $W_i/F_i/C_i$) in such a way as to minimize $E(\mathrm{Y}, \mathrm{Y}')$

Convolution

The first step in this architecture is that of the convolution of the image that one wishes to classify. The convolution is the highlighting of a few well-chosen characteristics in the source image in order to have the same image but with a kind of filter, as you can see in the image of step 2, the emphasis is rather on the contour of the face. This is done by applying a predefined matrix to the source matrix of the initial image. It is important to know that there are several convolution matrices and each one of them targets an aspect of the image. During this first step, several convolution filters are applied to the initial image, leading to the generation of several distinct images in terms of shape.

PREVOTEAU HUGO （雨果）
student id : 6219000083

Pooling

The second step, known as pooling, consists of reducing the dimensions of the images (matrices). The goal of this step is to keep as much relevant information as possible even while reducing the dimensions. The most commonly used pooling functions are (max/avg pooling).

The Convolution and Pooling steps are then iterated as many times as necessary until all the characteristics of the image are classified (example here: face contour, eyes, nose, mouth...).

Fully connected

The fully connected layers step is simply the application of a classical neural network to all the features classified by the previous steps. The output of this network is finally reduced to a probability that makes it possible to specify the degree to which the image in question is indeed a face.

PREVOTEAU HUGO （雨果）
student id : 6219000083

A machine learning platform: Dataiku

The Dataiku DSS data science platform enables teams of data professionals to collaborate within the same environment. It enables not only the analytical processing of data, but also the development of new solutions.

Who is Dataiku developed for?

The Dataiku DSS platform is designed for Data Analysts, Data Scientists and Data Ops. For analysts, it is an interactive visual interface in which it is possible to point, click, and develop using languages such as SQL. It is then possible to compare data, model, re-launch workflows, visualize results, and get insights on demand. These features allow Data Analysts to increase their efficiency.

For Data Scientists and developers, Dataiku DSS allows data to be prepared and modeled in seconds. It also helps to enhance ML libraries such as scikitlearn, R, MLlib, or H2O. The automation of tasks, through a fully customizable interface, offers a range of opportunities.

For Data Ops, Dataiku DSS removes the worry of using multi-technology platforms. It enables coordinated development and operations through workflow automation, creation of predictive web services, and monitoring of data and model status on a daily basis.

Why choosing Dataiku

The Dataiku DSS platform connects to an existing infrastructure and detects schedules and formats. Moreover, the data remains where it is: processing is disconnected from storage. In total, it has extended compatibility to 25 different storage systems. These systems include file upload platforms such as Filesystem, FTP, HTTP, SSH and SFTP. There are also businessQL's such as Oracle, MS SQL Server, but also PostgreSQL and MySQL. Analytical SQL such as Vertica, Greenplum, Redshift, Teradata and Exadata are also supported, as are other SQL databases. The platform supports NoSQL databases such as MongoDB, Cassandra and Elasticsearch. Hadoop's HDFS, Cloud S3 and many others complete the picture.

Finally, there is no need to transfer the data for processing. DSS takes care of processing within existing infrastructures such as SQL, Hadoop or Spark.

Machine Learning on Dataiku

With the DSS platform, which offers step-by-step guided Machine Learning, to clean up data or add new features and build a model in a single environment.

Thanks to visual feedback on the tasks performed, it is easier to evaluate the model and thus refine it. With visual reports and automated statistics, it is for example easier to interpret clusters created by an unsupervised algorithm such as KMEANS for example.

Dataiku in companies

For the data governance part, Dataiku allows to organize all tasks in a clear way, in a centralized catalog that gathers data, comments as well as elements and especially models.

PREVOTEAU HUGO （雨果）
student id : 6219000083

In order to guarantee the security of the platform, several solutions are proposed such as data governance with a system of permissions. In addition, all activities are listed on a specific dashboard so that they can be tracked.

The progress of projects can be permanently visualized by the team leaders, since the size and location of the datasets as well as the current DB tasks can be tracked. All this can be viewed via a dedicated dashboard.

A project with Dataiku

This is an example of a project in Dataiku, we can see the clarity of the project as a whole, with the data gathering, data cleaning, filters or algorithms applied to the data and at the end the different outputs of the model.

We can imagine that the improvement in terms of productivity increase even more when the processes are repeated on a daily basis.