IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Machine Learning - Binary Classification

by
QUINTANA Gonzalo
MARRA Tales
WANG Lei
AGUDELO Santiago

Dans le cadre du cours
Nom du cours (Code du cours)

Travail présenté à
Nom du professeur

TAF Mathematical Computing Engenieering
IMT Atlantique

Wednesday 4th December, 2019

Introduction

The following report summarizes our work on Machine Learning course project. This project consisted in binary classification of two datasets. The first dataset we took into account was the “Banknote Authentication Dataset”, in which data was extracted from images that were taken for evaluating the authentication procedure for banknotes. The second dataset we considered was the “Chronic Kidney disease Dataset”, which consisted in 25 features which may allow predicting a patient with chronic kidney disease.

The main goal of the project was to apply on practical datasets some of the different machine learning algorithms that were introduced during course sessions, all this by adopting a teamwork-based collaborative approach. It was also searched to raise consciousness about good programming practices, which are necessary for all developed codes to be organized, understandable and properly documented.

First, the datasets will be described as well as the pre-processing they require. Then, the binary classification problem will be solved for both datasets by using machine learning algorithms such as support vector machines, neural networks, decision trees and clustering techniques. The obtained results will be presented and discussed; this will allow evaluating the pertinence of the different algorithms in terms of their performance and complexity. In the end, a conclusion section will summarize the main points of discussion as well as the principal difficulties we encountered during the project’s realization.

1 Datasets: description and pre-processing

1.1 Banknote Authentication Dataset

These data were extracted from images that were taken from genuine and forged banknote-like specimens. These images have a size of 400x400 pixels with a resolution of about 660 dpi. In addition, wavelet transform was used to extract different features from images.

The variance, skewness and curtosis of the wavelet transformed image, the entropy of the image and its class are included in the dataset. This last information sets whether the image corresponds to a genuine of a forged specimen; it is therefore our objective to develop machine learning algorithms allowing to predict the image’s class from information on the dataset.

1.2 Chronic Kidney Disease Dataset

The data was taken over a 2 months period in India. It includes 25 features that allow predicting the presence of chronic kidney disease. Our goal is therefore to use the information contained in the dataset to determine through machine learning algorithms whether a patient suffers from this disease (we will call this case ckd) or not (we will call this case notckd).

1.3 Preprocessing

The datasets we worked on encountered problems such as missing values, out-of-range values, invalid characters, non-centered data, etc. Also, some of the data corresponded to text that cannot be directly processed; therefore, a conversion to a numerical format is necessary.

For pre-processing data, we developed a module that can be applied to a general dataset.This way, we could use the same procedure to pre-process both of the datasets concerned by this project. In such module, the user must indicate the repertory where data are localized and whether the header arrow and the index column are present or not. The user is also requested if text data should be transformed into numerical data.

The following steps allow obtaining a exploitable dataset:

1. The files (.text or .csv) are read through pandas in Python.

2. Features and targets are separated.

3. For each feature and target, the strings they contain are identified. In this same step, all special characters are left behind.

4. Data are cleansed. If there happens to be a missing value in a numerical column (feature), it is replaced by the average value of the column. For modal columns (i.e, the columns that contain strings), strings are converted into a numerical format. For example, the words yes/no are converted to a binary format 1/0. If these columns happen to have missing values, they are filled with random data in a way the proportion of the values in the column doesn't change.

5. Features are normalized (the mean is subtracted and the result is scaled in order to have unit variance).

In all the algorithms that will be further mentioned, the proportion of the training set and the test set is of 2/3 and 1/3 of the dataset, respectively.

2 PCA

When representing each exemplary as a point in the variables space, principal components are obtained as the succession of orthogonal vectors that best fits the data (this is, the succession of vectors that minimizes the error one commits when projecting the data into these vectors). It is possible to calculate such vectors as the eigenvectors of the correlation matrix of data; each of which is associated to an eigenvalue that indicates the total contribution of each principal component to the variation of the data.

When taking into account the most important components, this is, those corresponding to the highest eigenvalues, it is possible to reduce the problem to a lower dimension. Conversely, it is possible to represent variables in an individual space in order to determine how principal components are correlated to them.

3 SVM

The main goal of this algorithm is to separate binary data through an hyperplane. Since data may not be linearly separable, the kernel trick may be used to map data into a higher dimensional space, in which it would become linearly separable. SVM was implemented through scikit-learn library. This function solves the following optimization problem, given training vectors x_i and

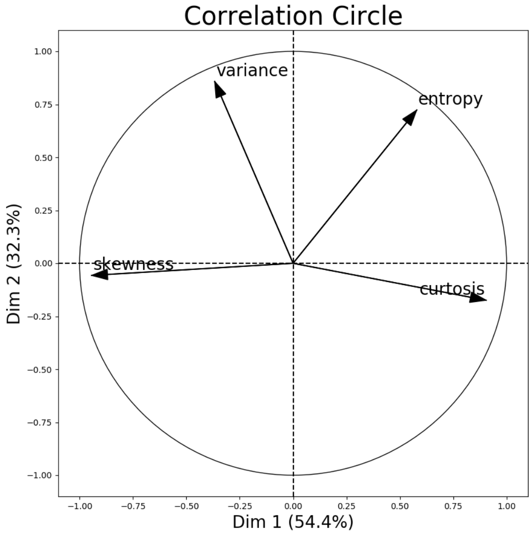


Figure 1: Correlation Circle obtained through PCA

$y_i \in \{0,1\}$:

$$\min_{w,\zeta} \quad \frac{1}{2}w^T w + C\sum_{i=1}^{n}\zeta_i$$

$$st \quad y_i(w^T\varphi(x_i) + b \geq 1 - \zeta_i$$

$$\zeta_i \geq 0$$

Its dual problem:

$$\min_{\alpha} \quad \frac{1}{2}\alpha^T Q\alpha - e^T\alpha$$

$$st \quad y^T\alpha = 0$$

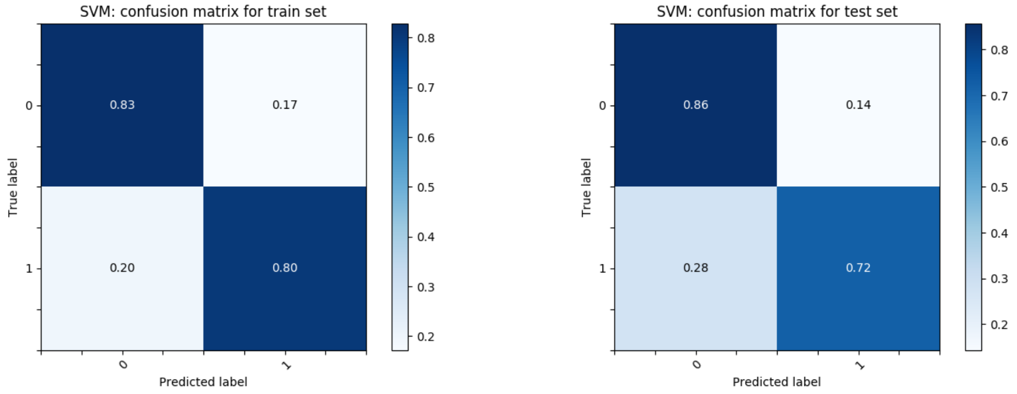
$$0 \leq \alpha_i \leq C$$

e is an only ones vector and the matrix Q is defined as $Q_{ij} = y_i y_j K(x_i,x_j)$, where $K(x_i,x_j) = \varphi(x_i)^T\varphi(x_j)$ is the kernel that implicitly maps our data into a higher dimensional space.

For our implementation, we defined $K(x,x') = exp(-\gamma||x - x'||^2)$ with $\gamma = \frac{1}{N \cdot var(x)}$. The following figures show the confusion matrices for training and test data that were obtained when running the algorithm on both datasets. It can be stated that this algorithm reaches a perfect performance when applied on the banknote dataset : indeed, none of the fake bills in the test set is labeled as true and vice-versa. When looking at the kidney disease dataset, none of the healthy patients from the test data are labeled as sick, but as much as 1% of the sick ones are diagnosed as healthy. This rate should be, however, ideally as close to zero as possible.

4 KNN

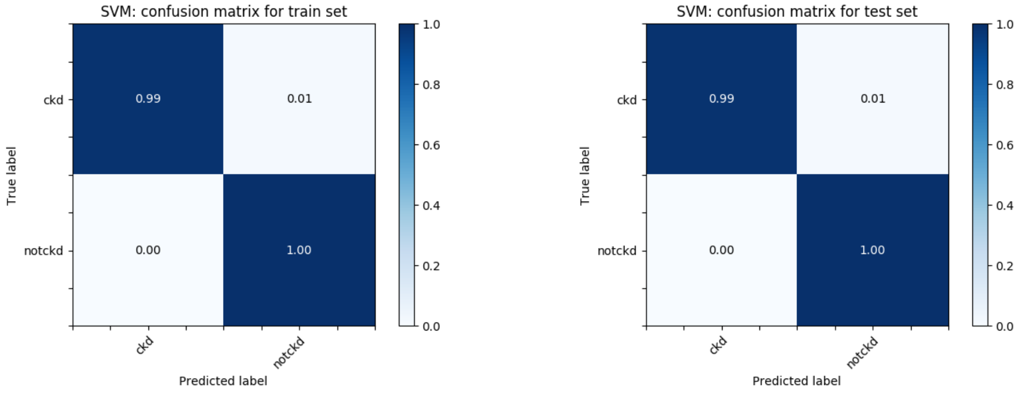
KNN is a non-parametric method used for classification. This algorithm requires placing a certain number of trained examples in the feature space (the class of these examples is therefore



(a) Confusion matrix of training data

(b) Confusion matrix test data

Figure 2: Confusion matrices in the Banknote dataset



(a) Confusion matrix of training data

(b) Confusion matrix test data

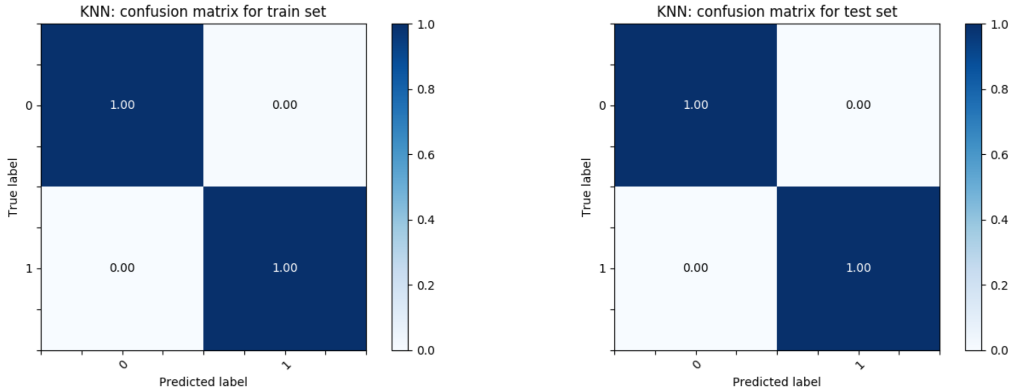
Figure 3: Confusion matrices in the Kidney Disease dataset

known in advance). A new sample is introduced and is classified in function of the class of its K nearest neighbors, where K is a parameter one must previously fix and neighborhood is defined in function a metric that is usually the euclidean distance. The class that will be assigned to the new sample is generally the most frequent one among its K closest neighbors. From a statistical approach, it can be shown that this classification rule is actually the one that maximizes the posterior probability $p(y|x)$, where y stands for the class and x for the sample to be classified.

The following figures show the obtained confusion matrices when applying this algorithm to both datasets. It can be seen that KNN has an outstanding performance on the test data corresponding to the banknote dataset. Indeed, none of the examples are misclassified. When applying the algorithm to the Kidney Dataset, however, it can be seen that even though no healthy patient is diagnosed as sick, as much as 6% of the sick ones are diagnosed as healthy, which is considerably high and results in a bad performance of KNN for this specific application.

5 Neural networks

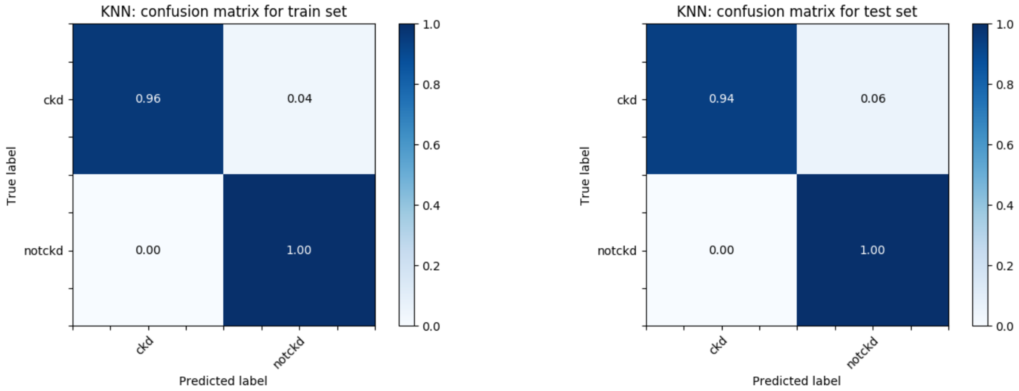
A four layer neural network was implemented by using the library Keras. Each neuron's output is a function of the weighted sum of its inputs. Such function is known as activation function



(a) Confusion matrix of training data

(b) Confusion matrix test data

Figure 4: Confusion matrices in the Banknote dataset



(a) Confusion matrix of training data

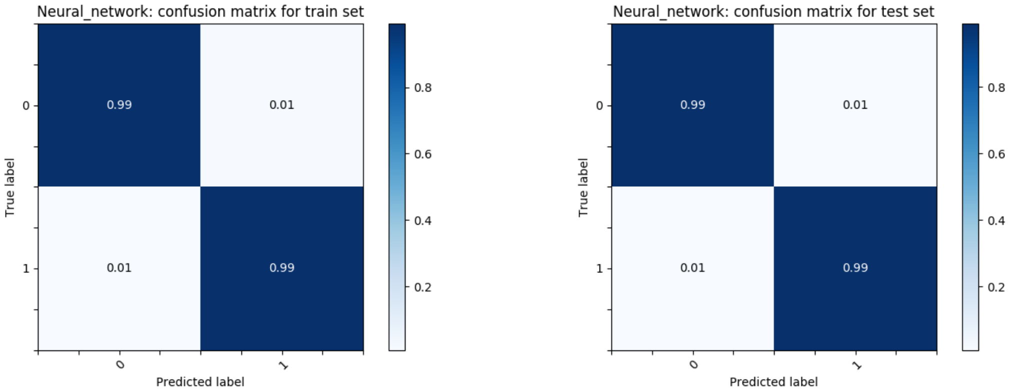
(b) Confusion matrix test data

Figure 5: Confusion matrices in the Kidney Disease dataset

and it determines whether and in what extent the neuron's output is propagated through the network and affect the overall system's output.

The first layer uses five neurons with a ReLU activation function; the second layer consists in three neurons with a tanh activation function; the third layer is composed of two neurons with same activation function as the former, and the final layer consists of a single neuron with a sigmoid activation function.

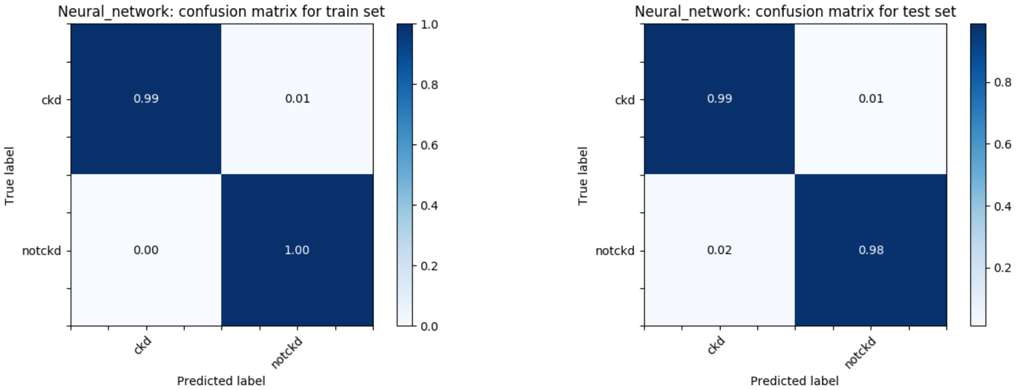
The neural network approach leads to the confusion matrices shown below. For the banknote dataset, 1% of the fake bills are labeled as real and vice-versa. Concerning the kidney disease set, none of the healthy patients as diagnosed as sick, but as much as 1% of the sick ones are diagnosed as healthy. SVM performs better than the used neural network in the first dataset, while it has a similar performance when applied on the second dataset. The result may change in function of the chosen architecture, and it can therefore be possible to build a neural network which performs better than this one or even better than SVM, which is so far the most performing algorithm.



(a) Confusion matrix of training data

(b) Confusion matrix test data

Figure 6: Confusion matrices in the Banknote dataset



(a) Confusion matrix of training data

(b) Confusion matrix test data

Figure 7: Confusion matrices in the Kidney Disease dataset

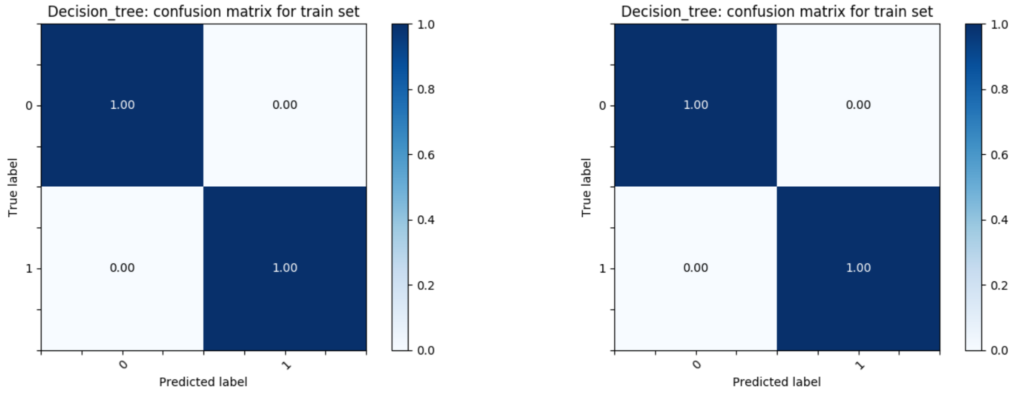
6 Decision trees

Decision Tree is a non-parametric supervised learning method used for both classification and regression tasks. In general, decision trees are constructed via an algorithmic method that identifies ways to split a data set based on different conditions. The decision rules are generally in form of if-then-else statements. The deeper the tree, the more complex the rules and fitter the model.

In our implementation, we used skicit-learn library. The chosen splitting criteria was Gini, which favours pure nodes as it measures how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the leave.

The following confusions matrices were obtained when running this algorithm on both datasets:

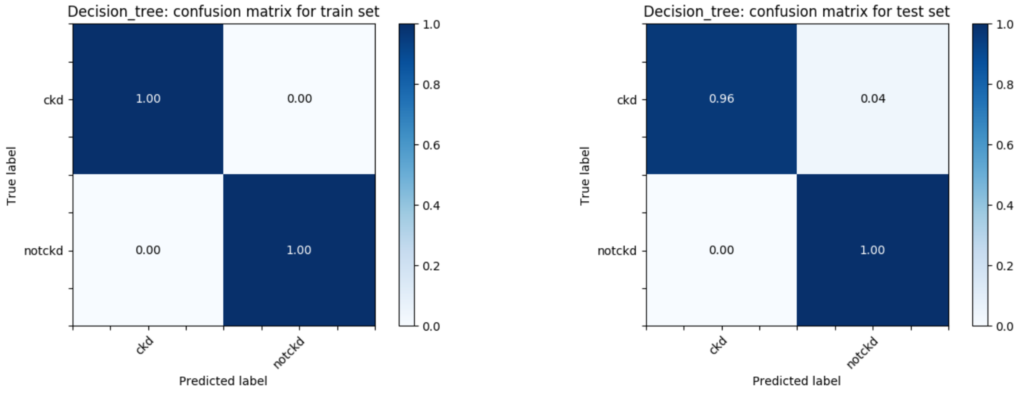
Training data are classified with no error in both datasets. For the banknote dataset, none of the fake bills from the test data is labeled as real, whereas 3% of the true bills are labeled as false. Concerning the kidney disease dataset, as much as 2% of the sick patients are diagnosed as being healthy, whereas none of the healthy patients is diagnosed as being sick. The false negative rate



(a) Confusion matrix of training data

(b) Confusion matrix test data

Figure 8: Confusion matrices in the Banknote dataset



(a) Confusion matrix of training data

(b) Confusion matrix test data

Figure 9: Confusion matrices in the Kidney Disease dataset

doesn't seem to be high, but in health issues it happens to be the most important one and it should be ideally zero. Therefore, some improvements could be made in order to lower this rate, even if it results in a slight augmentation of the false positive rate.

7 Bayesian approach

Given a class variable y and a dependent feature vector x, the Bayes rules states:

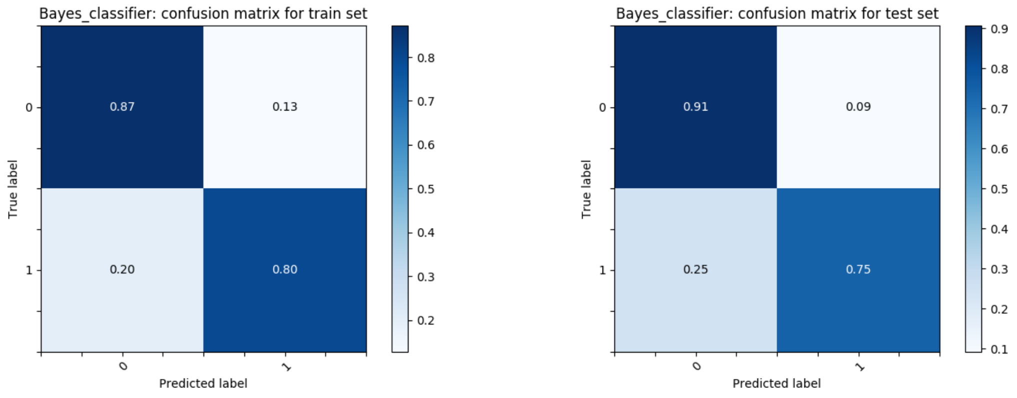
$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

Since the denominator is constant for a given input, maximizing the posterior probability $p(y|x)$ is equivalent to maximizing the numerator of the expression above. Therefore, given a feature vector x, its class can be estimated as it follows:

$$\hat{y} = \arg\max_{y} p(x|y)p(y)$$

$p(y)$ is estimated as the frequency of class y in the dataset. The performance of the classification relies on how accurate our previous knowledge of the distribution $p(x|y)$ is. A naive yet good first approach is to model this distribution as Gaussian.

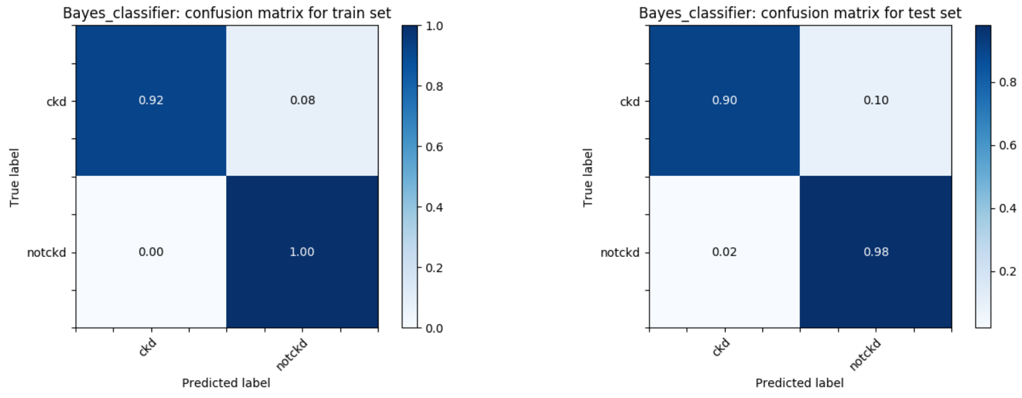
Implementing the previous decision rule by using skicit-learn gives the confusion matrices for the considered datasets. It can be stated that the overall performance of this technique is way lower than that of the other algorithms. For example, when taking into account the Kidney Disease dataset, the test data false negative rate is as high as 7%. This means that if we were talking about actual patients, 7% of the sick ones would have been diagnosed as healthy. This rate is relatively high when considering disease diagnose. When looking at the banknote dataset, as much as 9% of counterfeit bills are classified as real and as much as 25% of the real ones are classified as fake ; both results show the bayesian classifier is not well suited for this dataset.



(a) Confusion matrix of training data

(b) Confusion matrix test data

Figure 10: Confusion matrices in the Banknote dataset



(a) Confusion matrix of training data

(b) Confusion matrix test data

Figure 11: Confusion matrices in the Kidney Disease dataset

This performance can be explained by errors in our modeling hypotheses. Indeed, we supposed our training data had Gaussian distribution when fixing the class y, which might actually be a coarse approximation. Previous statistic analysis of the data can be performed in order to determine a better estimate of $p(x|y)$; this way, our modeling hypothesis would have fitted our training data and it would probably have led to a more acceptable performance.

8 Discussion on good programming practices

Conclusion

Through this document, we exposed the implementation of different machine learning algorithms on two different datasets. The analysis of their performance relies on the metric we use, and the choice of the metric isn't evident since it depends on the nature of the problem itself. For example, when studying the Kidney Disease dataset some of the algorithms threw a false negative rate of about 2%. However, if the test is applied on a population that happens to have a million sick people, this apparently small percentage would mean two thousand people who do have the disease would be incorrectly diagnosed, which wouldn't be acceptable.