

Aa

Pau Rullán Ferragut, paurullan@gmail.com  
University of Balearic Islands

Aprendimento automatico I  
Year 2011-12  
Alessio Micheli, Davide Bacciu  
University of Pisa

**Abstract**

Resum.  
La la.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Method</b>	<b>3</b>
2.1	Model verification . . . . .	3
2.2	Normality analysis . . . . .	3
2.3	Correlation analysis . . . . .	4
2.4	Classification techniques . . . . .	4
2.4.1	NBC with Gaussian parameters . . . . .	4
2.4.2	NBC with multivariate Gaussian . . . . .	5
2.4.3	Gaussian mixed model . . . . .	5
2.5	General method . . . . .	5
<b>3</b>	<b>Experimental results for MD5 data set</b>	<b>5</b>
3.1	Normality analysis . . . . .	6
3.2	Correlation analysis . . . . .	6
3.3	NBC with Gaussian parameters . . . . .	6
3.4	NBC with multivariate Gaussian . . . . .	6
3.5	Gaussian mixed model . . . . .	6
3.6	Mixed techniques . . . . .	6
3.7	Summary . . . . .	6
<b>4</b>	<b>Experimental results for MIX5 data set</b>	<b>6</b>
4.1	Normality analysis . . . . .	6
4.2	Correlation analysis . . . . .	6
4.3	NBC with Gaussian parameters . . . . .	6
4.4	NBC with multivariate Gaussian . . . . .	6
4.5	Gaussian mixed model . . . . .	6
4.6	Mixed techniques . . . . .	6
4.7	Summary . . . . .	6
<b>5</b>	<b>Conclusions</b>	<b>6</b>
<b>6</b>	<b>Bibliography</b>	<b>6</b>
<b>A</b>	<b>Experimental replication</b>	<b>7</b>
<b>B</b>	<b>Original code</b>	<b>8</b>
B.1	Main code . . . . .	8
B.2	lib . . . . .	8
B.3	test . . . . .	8
B.4	data . . . . .	8

# 1 Introduction

Descripció de la motivació.

Això és un text sense serifa

Descripció del dataset.

Això és un text monoespai

Definir bé el problema (segons les entrades i sortides)

Definir bé la tasca, model, algorisme d'aprenentatge, hipòtesis

Descriure si s'han tret resultats de literatura existent.

Descriure si s'han generat nous algorismes o resultats.

# 2 Method

In this section I overview the important techniques used along the project:

1. Model verification
2. Normality analysis
3. Correlation analysis
4. Bayesian naive classification with Gaussian parameters
5. Bayesian naive classification with multivariate Gaussian
6. Gaussian mixed model

## 2.1 Model verification

The process of model verification is to evaluate the generalization capabilities of the hypothesis. This issue is very important not only because in many occasions we will be able to quantify the error but most important because we need to avoid to over-fit our model on the training data. This means that without taking into account some basic rules it is very easy to model an hypothesis that seems to work very well only to later find out we were modeling wrong.

Although computationally expensive, one of the preferred methods is to do what is called a k-fold: partition the tagged data set, use only one part for training your solution and test against the other k-1 parts. The goal is to use this scheme on all k-parts and evaluate if the performance is consistent among them. This evaluation is usually done with the help of a *confusion matrix*, an square matrix that summaries the correct and incorrect guesses and allows a quick analysis of the performance of the model and implementation.

## 2.2 Normality analysis

The first thing I did was plot the different attributes for every class and model and look if the histogram felt familiar. When I saw that most of them looked like a Gaussian Bell the most natural thing to do was to pass a normality test.

The normality test is an useful method to analysis a data set and test if it was generated by a normal distribution. Although typically this *p-value* is contrasted against 0.05 of confidence is very important to remember that the method is very fragile to non-large data sets and can easily give false negatives. The interesting part came that the trend showed that many of the class pairs were normal-generated and in many more one of the two classes followed a normal distribution. The final observation is that the Gaussian is good enough for both models and in some special attributes a more complex approach would be used in order to find a better fitting.

## 2.3 Correlation analysis

In a multivariate environment there is the possibility to fuse some dimensions in order to simplify the problem at hand. This approach is called *dimension reduction* and can be applied in many fashions such as creating composed attributes, mixed distributions, kernel mappings or even removing them completely. For this project there is a simple approach: use the Pearson test for correlation between normal distributions and try to join the highly correlated through a Gaussian mixing or using a multivariate for both dimensions at the same time.

## 2.4 Classification techniques

The machine learning field provides to big frameworks tools: regression and classification. In the project the focus is on classification: given a set of tagged items and a collection of attributes for each item, try to infer the tags for a new item of known attributes. For example, in the data set there are eight real variables and a tag that puts the item in *class 0* or *class 1*. This classification can be done by many approaches such as neural networks or statistical methods. During the project we focus in graphical statistical methods.

### 2.4.1 NBC with Gaussian parameters

A naive Bayes classification (NBC) is a supervised learning approach for classification using statistical techniques and the Bayes theorem with the special feature that we assume complete independence between the different attributes. This requirement makes the NBC much less powerful than a neural network but very interesting to use if the problem allows it because the modeling and optimization is clear and simple.

The usual approach is to model the different attributes of the data set in independent normal variables and calculate the combined probability using a logarithmic form of the Bayes theorem. This makes the training as easy as collecting the data and the prediction

simply a calculation of the density distribution function for the trained set.

#### **2.4.2 NBC with multivariate Gaussian**

There is a direct extension of the NBC that still considers the attributes independent but are modeled under the same multivariate Gaussian distribution. The implementation details involve just changing the probability distribution function for a version that expects a multivariate distribution. This change may or may not improve the performance of the classifier.

#### **2.4.3 Gaussian mixed model**

In many cases we would like to cluster and study a data set in order to find some common characteristics. While this is mostly used in unsupervised classification the goal in our case is to get a more fitted data distribution. The usual approach is to combine many Gaussian distributions using an expectation maximization and later produce a probability distribution function as a weighted sum of the many components.

### **2.5 General method**

As a first step I visualized the data in order to try to find some kind of pattern or known distribution. The intuition was to plot an histogram for every attribute with both classes on the same graphic. (fig 1) Since most of the attributes look like normal Gaussian distributions I passed a normality test on them. Even that many single class-attribute data did not pass the test the global trend was positive so I decided to use a naive Bayes classification for the base case and later expand the investigation with some multivariate Gaussian distribution and a Gaussian mixed model.

In top of that I did a covariance analysis that showed how some attributes were correlated and could suffer a dimensional reduction.

All this procedure was done on both data sets and studied through a simple 4-fold model selection.

## **3 Experimental results for MD5 data set**

In this section I explain the experimental results with the procedure described in the general methodology.

### **3.1 Normality analysis**

### **3.2 Correlation analysis**

### **3.3 NBC with Gaussian parameters**

### **3.4 NBC with multivariate Gaussian**

### **3.5 Gaussian mixed model**

### **3.6 Mixed techniques**

On combining the different techniques to improve the performance.

### **3.7 Summary**

## **4 Experimental results for MIX5 data set**

In this section we change data sets and focus on the MIX5, redoing all the analysis and studies described on the methodology.

### **4.1 Normality analysis**

### **4.2 Correlation analysis**

### **4.3 NBC with Gaussian parameters**

### **4.4 NBC with multivariate Gaussian**

### **4.5 Gaussian mixed model**

### **4.6 Mixed techniques**

### **4.7 Summary**

## **5 Conclusions**

Què hem fet.

Results of the classification, explaining the best model we could find.

Resultats dels blindtests. Indicar el nom del grup (nickname, pex batman).

Link to bitbucked repo.

Autorització de la publicació.

## **6 Bibliography**

Bibliografia

## A Experimental replication

The whole project can be found in a git repository at the URL `X`. If you do not want to get the data using git there is a package in `X`.

The project itself is organized in four sections:

1. The study of the data set
2. A simple NBC
3. The multivariate and mixture models
4. The final model for each data set

All the code has been written in the Python programming language so it does not need compiling but to install the interpreter and the scientific modules. On a Linux system this is should be easy and on a Windows or Mac OS X not much more difficult than downloading the packages from the website.

The requirements are:

- Python2.7
- numpy and scipy
- python matplotlib
- python scikit-learn

The code has been designed to make easy the replication process:

- `python workon.py X` (where `X` is a data set)  
(or `make work_md work_mix` or `make work` for both)  
Creates all the plotting for the initial study, computes covariance, normality and the Pearson correlation.
- `python classification.py X` (where `X` is a data set)  
(or `make class_md class_mix` or `make class` for both)  
Makes the general NBC, a single multivariate and a mixed model  $prior = 6$ .
- `python mix.py MIX5-TR.csv`  
(or `make mix`)  
Do a 4-fold on the data set with the final model.
- `python mix.py MIX5-TR.csv MIX5-TS.csv X.csv`  
(or `make mix_final`)  
Uses the final model with the full training set, test is against the blind test and save the result on the `X` file name. The default is the write it on `batman_MIX5-TS.csv`.
- All the last three points have their equivalent for the MD5 data set.

## **B Original code**

The published code is organized in user programs in the main directory, libraries, test and data sets.

```
cup/  
  Makefile  
  workon.py  
  classification.py  
  mix.py  
  md.py  
  lib/  
  test/  
  data/  
  doc/
```

### **B.1 Main code**

### **B.2 lib**

### **B.3 test**

### **B.4 data**