

# Practical work 03 – 03.10.2023

## Classification with Bayes - System Evaluation

---

### Summary for the organisation :

- Submit the solutions of the practical work before the date specified in Moodle.
- **Rule 1.** Submit an archive (\*.zip!) with your Python notebooks (one per exercise), including datasets and all necessary files.
- **Rule 2.** The archive file name must contain the number of the practical work, followed by the family names of the team members by alphabetical order, for example `02_dupont_muller_smith.zip`. Put also the name of the team members in the body of the notebook (in first cell). Only one submission per team.
- **Rule 3.** We give a **fail** for submissions that do not compile (missing files are a common source of errors...). So, make sure that your whole notebooks give the expected solutions by clearing all cells and running them all before submitting.

### Exercise 1 Classification system using Bayes

The objective of this exercise is to build a bayesian classification systems to predict whether a student gets admitted into a university or not based on their results on two exams<sup>1</sup>.

You have historical data from previous applicants that you can use as a training set. For each training example  $n$ , you have the applicants scores on two exams  $(x_{n,1}, x_{n,2})$  and the admissions decision  $y_n$ . Your task is to build a classification model that estimates an applicants probability of admission based on the scores from those two exams.

#### a. Bayes - Histograms

Implement a classifier based on Bayes using histograms to estimate the likelihoods.

- Read the training data from file `ex1-data-train.csv`. The first two columns are  $x_1$  and  $x_2$ . The last column holds the class label  $y$ .
- Compute the priors of both classes  $P(C_0)$  and  $P(C_1)$ .

---

1. Data source : Andrew Ng - Machine Learning class Stanford

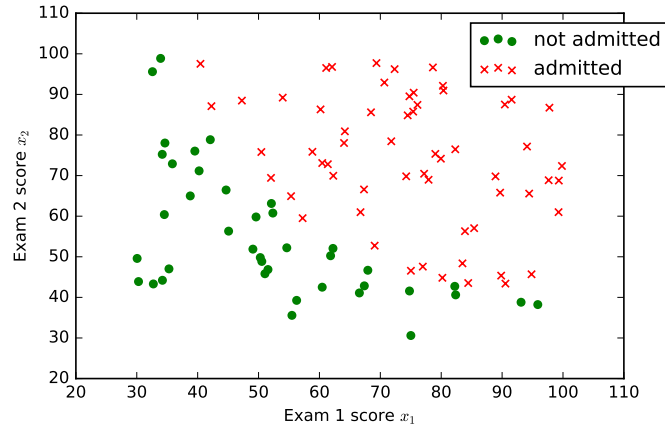


FIGURE 1 – Training data

- c) Compute histograms of  $x_1$  and  $x_2$  for each class (total of 4 histograms). Plot these histograms. Advice : use the numpy `histogram(a,bins='auto')` function.
- d) Use the histograms to compute the likelihoods  $p(x_1|C_0)$ ,  $p(x_1|C_1)$ ,  $p(x_2|C_0)$  and  $p(x_2|C_1)$ . For this define a function `likelihood_hist(x,hist_values,edge_values)` that returns the likelihood of  $x$  for a given histogram (defined by its values and bin edges as returned by the numpy `histogram()` function).
- e) Implement the classification decision according to Bayes rule and compute the overall accuracy of the system on the test set `ex1-data-test.csv` :
  - using only feature  $x_1$
  - using only feature  $x_2$
  - using  $x_1$  and  $x_2$  making the naive Bayes hypothesis of feature independence, i.e.  $p(X|C_k) = p(x_1|C_k) \cdot p(x_2|C_k)$

Which system is the best ?

## b. Bayes - Univariate Gaussian distribution

Do the same as in a. but this time using univariate Gaussian distribution to model the likelihoods  $p(x_1|C_0)$ ,  $p(x_1|C_1)$ ,  $p(x_2|C_0)$  and  $p(x_2|C_1)$ . You may use the numpy functions `mean()` and `var()` to compute the mean  $\mu$  and variance  $\sigma^2$  of the distribution. To model the likelihood of both features, do the naive Bayes hypothesis of feature independence, i.e.  $p(X|C_k) = p(x_1|C_k) \cdot p(x_2|C_k)$ .

## Exercise 2 System evaluation

Let's assume we have trained a digit classification system able to categorise images of digits from 0 to 9, as illustrated on Figure 2.

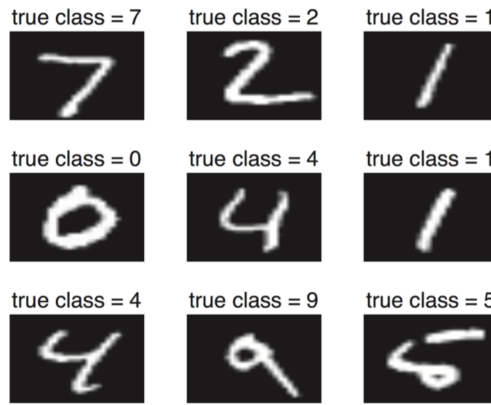


FIGURE 2 – Digit classification system

After training, the system has been run against a test set (independent of the training set) including  $N_t = 10'000$  samples. The system is able to compute estimations of a posteriori probabilities  $P(C_k|\mathbf{x})$  for  $k = 0, 1, 2 \dots, 9$ .

In file `ex1-system-a.csv`, you find the output of a first system A with the a posteriori probabilities  $P(C_k|\mathbf{x})$  in the first 10 columns and with the ground truth  $y$  in the last column.

- Write a function to take classification decisions on such outputs according to Bayes' rule.
- What is the overall error rate of the system?
- Compute and report the confusion matrix of the system.
- What are the worst and best classes in terms of precision and recall?
- In file `ex1-system-b.csv` you find the output of a second system B. What is the best system between (a) and (b) in terms of error rate and F1.

### Exercise 3 Review questions

- a) Assuming an univariate input  $\mathbf{x}$ , what is the complexity at inference time of a Bayesian classifier based on histogram computation of the likelihood?
- b) Bayesian models are said to be *generative* as they can be used to generate new samples. Taking the implementation of the exercise 1.a, explain the steps to generate new samples using the system you have put into place. **Optional.** Provide an implementation in a function `generateSample(priors, histValues, edgeValues, n)` that returns  $n$  new samples  $[(x_1^{(1)}, x_2^{(1)}), (x_1^{(2)}, x_2^{(2)}), \dots, (x_1^{(n)}, x_2^{(n)})]$  in an array.
- c) What is the minimum overall accuracy of a 2-class system that is built on a training set that includes 5 times more samples in class A than in class B?
- d) Let's look back at the PW02 exercise 3 of last week. We have built a KNN classification systems for images of digits on the MNIST database.

How would you build a Bayesian classification for the same task? Comment on the prior probabilities and on the likelihood estimators. More specifically, what kind of likelihood estimator could we use in this case? **Optional** : implement it and report performance!

- e) Read <https://theintercept.com/2019/07/26/europe-border-control-ai-lie-detector/>. The described system is "*a virtual policeman designed to strengthen Europeans borders*". It can be seen as a 2-class problem, either you are a suspicious traveler or you are not. If you are declared as suspicious by the system, you are routed to a human border agent who analyses your case in a more careful way.
  - i) What kind of errors can the system make? Explain them in your own words.
  - ii) Is one error more critical than the other? Explain why.
  - iii) According to the previous points, which metric would you recommend to tune your ML system?
- f) When a deep learning architecture is trained using an unbalanced training set, we usually observe a problem of bias, i.e. the system favors one class over another one. Using the Bayes equation, explain what is the origin of the problem.

Comments:

Ex1: Well done  
Ex2: Well done  
Ex3: Well done

Grade: Pass