Practical work 03 – 02.10.2018
Classification with Bayes - System Evaluation

Summary for the organisation :

- Submit the solutions of the practical work before Monday 12h00 next week in Moodle.
- Preferred modality : archive with iPython notebook(s).
- Alternative modality : pdf report with annotated code and outputs.
- The file name must contain the number of the practical work, followed by the 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 (or report).
- Only one submission per team.

Exercice 1 Classification system using Bayes

In a similar way as for the practical work of last week, 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 [1].

You have historical data from previous applicants that you can use as a training set. For each training example n, you have the applicant's 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 applicant's 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.

a) 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.

b) Compute the priors of both classes $P(C_0)$ and $P(C_1)$.

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.

[1] Data source : Andrew Ng - Machine Learning class Stanford

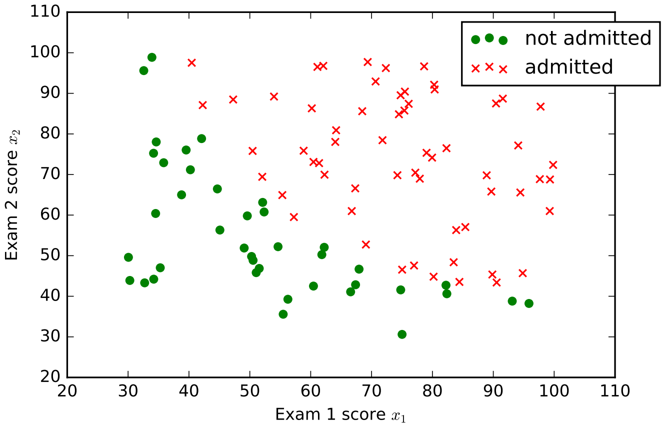

FIGURE 1 – Training data

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 `likelihoodHist(x,histValues,edgeValues)` 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 μ and variance σ^2 of the distribution. To model the likelihood of both features, you may also do the naive Bayes hypothesis of feature independence, i.e. $p(X|C_k) = p(x_1|C_k) \cdot p(x_2|C_k)$.

Exercice 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.

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

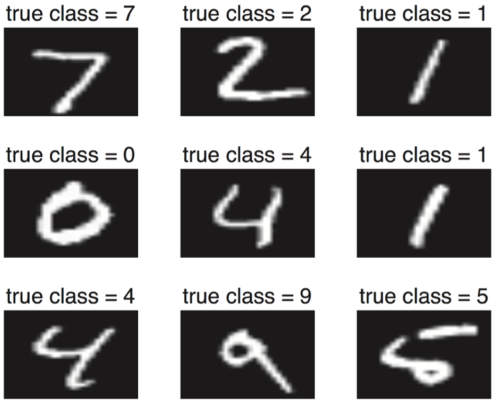

FIGURE 2 – Digit classification system

probabilities $P(C_k|\mathbf{x})$ for $k = 0, 1, 2 \ldots, 9$.

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.

a) Write a function to take classification decisions on such outputs according to Bayes'rule.

b) What is the overall error rate of the system ?

c) Compute and report the confusion matrix of the system.

d) What are the worst and best classes in terms of precision and recall ?

e) 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.

Exercice 3 System evaluation

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.

a) 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 estimator could we use in this case ?

b) **Optional** : implement it and report performance !