IFT3395/6390 Devoirs 1

Professeur : Aaron Courville

28 septembre 2017

— This homework is to be done in team of 2. Do not forget to put the names of all the members in the headings of the files and in the comments when you handout the file.
— We ask the format of the submitted homework to be in .pdf and all codes should be in python using numpy and matplotlib.
— We strongly recommend your python code to be done in the format of ipython notebook like in the demonstration. To produce mathematical formulas, you can use LATEX ; LYX ; Word ; If you use ipython notebook you can directly put equations in it.
— When you sumbit your homework, it will only be accepted via StudiUM and only one file per team. If you have mutliple files, an archive (zip, tar..) is accepted.

1 Probabilities

A study done on doctors in the United-States some years ago regarding their "probability intuition" had the following question :

The probability of breast cancer in women aged in their 40s participating in a mammograph is 1%. If a woman has breast cancer, there is 80% chance that the test be positive. If a women does not have this cancer, the probability of the test being positive is 9.6%.

If a women in her 40s is being tested and the test is positive, what is the probability that she is indeed suffering from breast cancer ?

1. more than 90%.
2. between 70% and 90%.
3. between 50% and 70%.

4. between 30% and 50%.
5. between 10% and 30%.
6. less than 10%.

95% of doctors in the study answered 2. What do you think ?
Formalise this question and compute the exact probabilties.
Hint : Bayes' rule.

2 Density estimation : parametrized Gaussian v.s. Parzen windows

In this question we use a dataset $D = \{x(1), ..., x(n)\}$ with $x \in R^d$.

1. Say that we have trained the parameters of a isotropic Gaussian density on D (to maximize the likelihood) so we can estimate the probability density.
   (a) Name the parameters and state their dimensions.
   (b) If the parameters are learned using maximum likelihood, express in function of points in D the equation that would produce the optimal parameters values (only the equation that computes the result, you do not need to rederive it).
   (c) What is the algorithmic complexity of this training, this computation of the parameters ?
   (d) For a test point x, write the function that will give the predicted probability density at point x : $\hat{p}_{\text{gauss}-\text{isotrop}}(x) = ?$
   (e) What is the algorithmic complexity of this prediction for each x ?
2. Consider that Parzen windows are used for the estimation of the probability density. These windows have an isotropic Gaussian kernel with width σ, and were trained on D.
   (a) Say that σ is fixed. What does the training phase consist of for these Parzen windows ?
   (b) For a test point x, write one detailed equation (with exponentials) the function that computes the probability density at point x : $\hat{p}_{mathrmParzen}(x) = ?$
   (c) What is the algorithmic complexity of this prediction for each x ?

3. Capacity
   (a) Which between these 2 (parametrized Gaussian v.s. Parzen with Gaussian kernel) has more capacity ? Explain.
   (b) With which of these techniques, and in what precise case, do you have chances to be overfitting ?
   (c) The σ in Parzen windows is also considered a hyper-parameter whereas in a parametrized Gaussian it is considered a parameter. Why ?
4. Consider the same problem but now with a diagonal Gaussian density.
   (a) Write down the formula for a diagonal Gaussian density in $\mathbb{R}^d$. Name the parameters and state their dimensions.
   (b) Demonstrate that the componants of a random vector following a diagonal Gaussian distribution are independant random variables.
   (c) Using $-\log p(x)$ as cost, write down the equation of the minimisation of empirical risk on the training set D (to learn the parameters).
   (d) Solve (with math, not code) this equation in order to find the optimal parameters.
5. Consider a classification problem with $D = \{(x(1), y(1)), ..., (x(n), y(n))\}$, we can build a Bayes classifier by using any of these 3 density estimations.
   (a) Explain in your words how you would train a Bayes classifier (training phase).
   (b) For a test point x, write the function that computes the predicted probability vector for each class for $x : g(x) = (..., ..., ..., ...)$

3 Practice : Density estimation

1. Implement a Gaussian diagonal parametric density estimator. It will take an input of arbitrary dimension d. As seen during the labs, it will have a method **train** used to learn parameters and a method **compute_predictions** that will compute the log density.
2. Implement a Parzen density estimator with a istropic Gaussian kernel. It will take an input of arbitrary dimension d. Similar to 1, it will

have a method train used to learn parameters and a method **compute_predictions** that will compute the log density.

3. 1D density : Choose a subset of the Iris dataset that corresponds to a single class (of your choice), and a feature, so that the data is in dimension $d = 1$, and plot (using the function plot) the following on a single chart :
   (a) the points of the subset on the x-axis
   (b) a plot of the density function estimated by your diagonal Gaussian parametric density estimator
   (c) a plot of the density function estimated by your Parzen estimator using a hyperparameter σ (standard deviation) that is too small
   (d) a plot of the density function estimated by your Parzen estimator using a hyperparameter σ that is too big
   (e) a plot of the density function estimated by your Parzen estimator using a hyperparameter σ that you think best fits the data.

   Use a different color for each plot, and add a clear legend.
4. 2D density : Add a second feature so that now the data is in dimension $d = 2$ and plot 4 charts, each showing the points of the subset (with the function plot), and the contour of the following density estimator (with the function contour) :
   (a) your diagonal Gaussian parametric density estimator
   (b) your Parzen estimator using a hyperparameter σ (standard deviation) that is too small
   (c) your Parzen estimator using a hyperparameter σ that is too big
   (d) your Parzen estimator using a hyperparameter σ that you think best fits the data.

4 Practice : Bayes classifier

1. Shuffle the Iris dataset (use numpy.random.shuffle after initiliazing the random number generator using numpy.random.seed(123)). Divide the dataset into 2 parts : a training set and a validation set. Prepare two versions of each of these datasets : one with all $d = 4$ features, the other one with only the $d = 2$ first features that we will only use to visualize (plot) the results of our classifier.

2. Bayes classifier using diagonal Gaussian parametric density functions
   (a) Implement the Bayes classifier using diagonal Gaussian parametric density functions
   (b) 2D visualization. Using only the 2 first features, train your Bayes classifier on the train set ; plot the decision boundary as well as the training examples and validation examples.
   (c) Error rate in dimension $d = 2$: Compute and print the error rate of your classifier (trained on $d = 2$ features), on the train set and the validation set.
   (d) Error rate in dimension $d = 4$: Train your classifier using all features. Compute and print the error rate on the train set and the validation set.
3. Bayes classifier using Parzen density functions with isotropic Gaussian kernels
   (a) Implement the Bayes classifier using Parzen density functions with isotropic Gaussian kernels
   (b) 2D visualization. Using only the 2 first features, train your Bayes classifier on the train set ; plot the decision boundary as well as the training examples and validation examples. Provide 3 such charts of the decision boundary : one for σ too small, too big and appropriate.
   (c) Training curves for $d = 2$. Compute the training error and validation error both on the training and validation sets, as a function of the hyperparameter σ (use 100 different values for σ so that you can plot a curve. Indicate the best value for σ.
   (d) Training curves for $d = 4$. We will now use all features. Compute the training error and validation error both on the training and validation sets, as a function of the hyperparameter σ (use 100 different values for σ so that you can plot a curve. Indicate the best value for σ.
4. Using these experiments, for the Iris classification task (and for this particular split train/validation), indicate the best algorithm between Bayes classifier with diagonal Gaussian density functions and Parzen windows and the best correponding hyperparameters : dimension (2 ou 4) and σ (if relevant). Specify the train error and validation error.