1 Density Estimation

(a) (**10 points**) Suppose we have N i.i.d samples $x_1, x_2, \cdots, x_n$. We will practice the maximum likelihood estimation techniques to estimate the parameters in each of the following cases:

- We assume that all samples can only take value between 0 and 1, and they are generated from the Beta distribution with parameter α unknown and $\beta = 1$. Please show how to derive the maximum likelihood estimator of α.
- We assume that all samples are generated from Normal distribution $\mathcal{N}(\theta, \theta)$. Please show how to derive the maximum likelihood estimator of θ.

(b) (**10 points**) Suppose random variables $X_1, X_2, \cdots, X_n$ are i.i.d sampled according to density function $f(x)$ and the kernel density estimation is in the form of $\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h} K(\frac{x-X_i}{h})$. Show the bias of the kernel density estimation method by the following steps:

- Show $\mathbf{E}_{X_1,\cdots,X_n}[\hat{f}(x)] = \frac{1}{h}\int K(\frac{x-t}{h}) f(t) dt$.
- Use Taylor's theorem around x on the density $f(x - hz)$ with $z = \frac{x-t}{h}$.
- Compute the bias $\mathbf{E}[\hat{f}(x)] - f(x)$.

2 Naive Bayes

(**15 points**) The binary Naive Bayes classifier has interesting connections to the logistic regression algorithm. In this exercise, you will derive the parametric form of the Naive Bayes classifier under certain assumptions and show that the likelihood function implied by the Gaussian Naive Bayes classifier for two classes is identical in form to the likelihood function for logistic regression. In the second part, you will derive the parameter estimation for the Naive Bayes algorithm.

(a) (**7 points**) Suppose $X = \{X_1, \ldots, X_D\}$ is a continuous random vector in $\mathbb{R}^D$ representing the features and Y is a binary random variable with values in $\{0, 1\}$ representing the class labels. Let the following assumptions hold:

- The label variable Y follows a Bernoulli distribution, with parameter $\pi = P(Y = 1)$.
- Each feature X_j, we have $P(X_j|Y = y_k)$ follows a Gaussian distribution of the form $\mathcal{N}(\mu_{jk}, \sigma_j)$.

Using the Naive Bayes assumption that states "*for all $j' \neq j$, X_j and $X_{j'}$ are conditionally independent given Y*", compute $P(Y = 1|X)$ and show that it can be written in the following form:

$$P(Y = 1|X) = \frac{1}{1 + \exp(-w_0 + \mathbf{w}^\top \mathbf{X})}.$$

Specifically, you need to find the explicit form of w_0 and $\mathbf{w}$ in terms of π, μ_{jk}, and σ_j, for $j = 1, \ldots, D$ and $k \in \{0, 1\}$.

(b) (**8 points**) Suppose a training set with N examples $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_N, y_N)$ is given, where $\mathbf{x}_i = (x_{i1}, \cdots, x_{iD})^\top$ is a D-dimensional feature vector, and $y_i \in \{0, 1\}$ is its corresponding label. Using the assumptions in 3.1 (not the result), provide the maximum likelihood estimation for the parameters of the Naive Bayes with Gaussian assumption. In other words, you need to provide the estimates for π, μ_{jk}, and σ_j, for $j = 1, \ldots, D$ and $k \in \{0, 1\}$.

3 Nearest Neighbor

(a) (**5 points**) Suppose we have the locations (coordinates) of 10 USC students during class time, and we know their majors, as follows:
Mathematics: $\{(0, 49), (-7, 32), (-9, 47)\}$
Electrical Engineering: $\{(29, 12), (49, 31), (37, 38)\}$
Computer Science: $\{(8, 9), (13, -1), (-6, -3), (-21, 12)\}$
Economics: $\{(27, -32), (19, -14), (27, -20)\}$

- Suppose we have a student whose coordinate is at $(20, 7)$ with unknown major. Using K-Nearest Neighbor with L_2 distance metric, predict the student's major if we are using $K = 1$ (**1 points**) and if we are using $K = 5$ (**1 points**). Similarly, what are the student's major predictions if we use L_1 distance metric with $K = 1$ (**1 points**) and with $K = 5$ (**1 points**). For $K = 5$, if there is a tie, please choose the label of the data point with closer distance. Please compare the results between these 4 different predictions (**1 points**). Do not forget to normalize/standardize (using $(x - \mu(x))/\sigma(x)$, $\sigma(x)$ is $N-1$ standard deviation) the coordinate of students with known major first. And normalize/standardize the student with unknown major using the mean and standard deviation of the students with known major. Provide intermediate computations of how do you arrive at your predictions.

(b) (**10 points**) Suppose now we want to derive a probabilistic K-Nearest Neighbor for classification of an unlabeled data point $\mathbf{x}$, which is D-dimensional. We have a (multi-dimensional) D-sphere with center at $\mathbf{x}$, allowing its radius to grow until it precisely contains K labeled data points, irrespective of their class. At this size, the volume of the sphere is V. Let there be a total of N labeled data points in the entire space (both inside and outside of the sphere), with N_c data points labeled as class c, such that $\sum_c N_c = N$. Also, a subset of the K data points inside of the sphere belongs to class c, there are K_c of them in total. We model estimated density associated with each class as $p(\mathbf{x} \mid Y = c) = \frac{K_c}{N_c V}$ and the class prior as $p(Y = c) = \frac{N_c}{N}$.

- (**5 points**) Using the fact that $\sum_c K_c = K$, derive the formula for unconditional density $p(\mathbf{x})$.
- (**5 points**) Using Bayes rule, derive the formula for the posterior probability of class membership $p(Y = c \mid \mathbf{x})$.

4 Decision Tree

(a) (**5 points**) Suppose you want to grow a decision tree to predict the *accident rate* based on the following accident data which provides the rate of accidents in 100 observations. Which predictor variable (weather or traffic) will you choose to split in the first step to maximize the information gain?

Weather	Traffic	Accident Rate	Number of observations
Sunny	Heavy	High	23
Sunny	Light	Low	5
Rainy	Heavy	High	50
Rainy	Light	Low	22

(b) (**5 points**) Suppose in another dataset, two students experiment with decision trees. The first student runs the decision tree learning algorithm on the raw data and obtains a tree T_1. The second student, normalizes the data by subtracting the mean and dividing by the variance of the features. Then, he runs the same decision tree algorithm with the same parameters and obtains a tree T_2. How are the trees T_1 and T_2 related?

(c) (**5 points**) In training decision trees, the ultimate goal is to minimize the classification error. However, the classifaction error is not a smooth function; thus, several surrogate loss functions have been proposed. Two of the most common loss functions are the *Gini index* and *Cross-entropy*, see [MLaPP, Section 16.2.2.2] or [ESL, Section 9.2.3] for the definitions. Prove that, for any discrete probability distribution p with K classes, the value of the Gini index is less than or equal to the corresponding value of the cross-entropy. This implies that the Gini index is a better approximation of the misclassification error.

Definitions: For a K-valued discrete random variable with probability mass function $p_i, i = 1, \ldots, K$ the Gini index is defined as: $\sum_{k=1}^{K} p_k(1 - p_k)$ and the cross-entropy is defined as $-\sum_{k=1}^{K} p_k \log p_k$.

5 Programming

(**35 points**) In this assignment, you will experiment with commonly used classification algorithms on a real-world dataset. You are allowed to use MATLAB or Python scripts. For MATLAB, you will find R2013b in http://software.usc.edu/matlab/. Without specific description, you are not allowed using Matlab toolbox functions like *knnclassify*, *knnsearch*. For Python, we only allow Python 2.7 and we strongly recommend to use Anaconda2 (https://www.continuum.io/downloads) for Python 2.7 and you are allowed using other libraries in Anaconda2 except machine learning packages such as *scikit-learn*. Your script should be executable under the Anaconda2 environment and we will grade your code in the same environment. You can build your own functions or modules, however, you should be careful to include them into your submission. If you use other packages not included in Anaconda2 and we fail to run your code, we won't regrade it after installing required packages. You should implement k-nearest neighbor(kNN) and Naive Bayes algorithms by yourself. Below, we describe the steps that you need to take to accomplish this programming assignment.

Dataset: We have preprocessed the *Glass Identification Data Set* from UCI's machine learning data repository. The training/test sets are provided in Blackboard as **train.txt** and **test.txt**. For data description, please see https://archive.ics.uci.edu/ml/datasets/Glass+Identification.

Please follow the steps below:

Data Inspection (**5 points**) The first step in every data analysis experiment is to inspect the datasets and make sure that the data has the appropriate format. You will find that the features in the provided dataset are continuous. Please answer following questions:

- How many attributes are?
- Do you think that all attributes are meaningful for the classification? If not, explain why.
- How many classes are? Class is a type of a glass.
- Please explain the class distribution. Which class is majority? Do you think that it can be considered as a uniform distribution?

Implement Naive Bayes (**10 points**) You will implement *Naive Bayes* algorithm. The inputs of your script are training data and unseen data (testing data). The script needs to output the accuracy on both training and testing data. As you see, all feature values are continuous real values. Thus, we are going to use a Gaussian distribution assumption. For continuous probability distribution functions, we can't exactly compute a probability value on a certain point because the probability is only defined in a certain range. However, it is fine to compare the probability density values to see which one is more probable. Note that a value greater than 1 is fine here because it is a probability density rather than a probability. Please provide training accuracy and testing accuracy.

Implement *k*NN (**10 points**) You will implement ***k****NN* algorithm. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of that single nearest neighbor. The inputs of your script are training data and unseen data (testing data). The script needs to output the accuracy

on both training and testing data. You should provide accuracy results when $k = 1, 3, 5, 7$ and $(L1, L2)$ distances, respectively. Note that you should check if the nearest neighbor of a testing sample is itself. If so, it will give you 100% accuracy when $k = 1$. When computing the training accuracy of kNN, we use leave-one-out strategy, i.e. classifying each training point using the remaining training points.

Performance Comparison (**10 points**) Compare the two algorithms (kNN, Naive Bayes) on the provided dataset.

***k*NN:** Consider $k = 1, 3, 5, 7$. For each k, report the training and test accuracy. When computing the training accuracy of kNN, we use leave-one-out strategy, i.e. classifying each training point using the remaining training points. Note that we use this strategy only for kNN in this assignment.

Naive Bayes: Report the training and test accuracy.

Discussion: The results from the different classifiers are similar? If so, explain why. If not, which one is better? And please explain why.

Submission Instruction: You need to provide the followings:

- Provide your answers to problems 1-4 and 5 [Data Inspection] and [Performance Comparison] in hardcopy. The papers need to be stapled and submitted into the locker#19 on the first floor of PHE building.
- Provide your answers to problems `*.pdf` file, named as `CSCI567_hw1_fall16.pdf`. You need to submit the homework in both hard copy (at the collection locker #19 at the PHE building 1st floor by 23:59pm of the deadline date) and electronic version as `*.pdf` file on Blackboard. If you choose handwriting instead of typing all the answers, you will get 40% points deducted.
- Submit ALL the code and report via Blackboard. The only acceptable language is MATLAB or Python2.7. For your program, you MUST include the main script called `CSCI567_hw1_fall16.m` or `CSCI567_hw1_fall16.py` in the root of your folder. After running this main file, your program should be able to generate all of the results needed for this programming assignment, either as plots or console outputs. You can have multiple files (i.e your sub-functions), however, the only requirement is that once we unzip your folder and execute your main file, your program should execute correctly. Please double-check your program before submitting. You should only submit one `*.zip` file. No other formats are allowed except `*.zip` file. Also, please name it as `[lastname]_[firstname]_hw1_fall16.zip`.

Collaboration You may collaborate. However, collaboration has to be limited to discussion only and you need to write your own solution and submit separately. You also need to list with whom you have discussed.