Assignment 2

Homework assignments will be done individually: each student must hand in their own answers. Use of partial or entire solutions obtained from others or online is strictly prohibited. Electronic submission on Canvas is mandatory.

1. **Linear Discriminant Analysis** (20 points) Please download the Iris data set from the UCI Machine Learning repository and implement Linear Discriminant Analysis for each pair of the classes and report your results. Note that there are three (3) class labels in this data set. Write down each step of your solution. **Do not use any package/tool**.

Solution:

1) First the data is grouped by categories or class labels.

2) The mean for each class is calculated using the group by aggregation and stored in a dictionary. Secondly the count of samples of each class is calculated and stored in a dictionary. The keys used for both the dictionary is the class label.

3) For each data in data-set the within class scatter(S_{Wi}) is calculated by subtracting the data-point with the mean based on class labels.

4) The total within class scatter(S_w) is obtained by adding all the within class scatter.

5) For each class the between class scatter matrix is calculated by multiplying samples count for each class with the square of difference of mean between the class and mean of all the samples.

6) The total between class scatter(S_B) is calculated by adding all the individual between class scatter.

7) The eigen values and eigen vectors are calculated by multiplying the inverse of S_W with S_B.

2. **Generative methods vs Discriminative methods** (60 points) Please download the breast cancer data set from UCI Machine Learning repository. **Do not use any package/tool**.

    1. Implement a Logistic regression classifier with ML estimator using Stochastic gradient descent and Mini-Batch gradient descent algorithms. Use cross-validation for evaluation and report the recall, precision, and accuracy on malignant class prediction (class label malignant is positive). Write down each step of your solution.
    2. Implement a probabilistic generative model for this problem. Use cross-validation for evaluation and report the recall, precision, and accuracy on malignant class prediction (class label malignant is positive). Write down each step of your solution.

Solution:

2.1:

1) The input data is fetched from .data file and copied to a data-frame. This is being a logistic regression the output or dependent variable(y) is going to be binary. So the input labels are replaced by 1 for Malignant "M" and 0 for benign "B".

2) For Stochastic gradient descent, the gradient descent is calculated for each row by using the predictions, learning rate, error and sigmoid function.

3) The coefficient matrix is calculated for adding the positive gradient.

4) For the test set the, the predictions are calculated using the coefficient matrix and input data-points.

5) The prediction value and actual values are compared as follows:

i) predicted value = 1 and $actualvalue = 1$, true positive count is incremented.

ii) predicted value = 1 and actual value = 0, false positive count is incremented

iii) predicted value = 0 and actual value = 1, false negative count is incremented

iv) predicted value = 0 and actual value = 0, true negative count is incremented

6) Based on the above count the Recall, Precision and Accuracy is calculated using the above counts.

7) This calculations are done for each fold in cross validation and the average of these is calculated as the final values.

8) For Mini Batch Gradient descent, create the batches for gradient evaluation. For each batch in batches calculate the gradient and form the coefficient matrix by subtracting the gradient.

9) For batch creation use size of 32, 64 etc. in the power of 2, as the processor is able to evaluate binary operations. Merge the x-data and y-data, first to form the batches based on index and maintain relation of x-data with y-data for evaluation of error. Form batches of equal sizes.

10) After evaluating the coefficient matrix, use the same for calculating the predictions on test data.

11) Repeat the same evaluation of error as mentioned in step 5 and 6, to calculate Recall, Precision and Accuracy.

2.2:

1) For probabilistic generative model, first fetch the data from .data file and transfer to the data-frame.

2) Find the mean value for the two classes by using the group by clause. Also calculate the count of samples in the group.

3) Evaluate the prior class probabilities $p(C_k)$, by dividing the count of class samples by total samples.

4) We need to evaluate w and ω_0 for two classes.

5) w is evaluated as below,

$w = \sum^{-} 1(\mu_1 - \mu_2)$

6) ω_0 is calculated as below,

$\omega_0 = -\frac{1}{2}\mu_1^T \sum \mu_1 + \frac{1}{2}\mu_2^T \sum \mu_2 + \ln\{\frac{P(C_1)}{P(C_2)}\}$

7) The co-variance matrix is calculated as below,

$\sum = \frac{N_1}{N} S_1 + \frac{N_2}{N} S_2$

where,

$S_1 = \frac{1}{N_1} \sum_{n \in C_1} (x_n - \mu_1)(x_n - \mu_1)^T$

$S_2 = \frac{1}{N_2} \sum_{n \in C_2} (x_n - \mu_2)(x_n - \mu_2)^T$

8) Using these values, class posterior probabilities are calculated using the following equation,

$P(C_k|x) = \sigma(w^T x + \omega_0)$

9) Since we have two classes, the boundary is at probability = 0.5. If the class posterior probability for class 1 is greater than 0.5, the input is classified as class 1 else class 2 otherwise.

10) The prediction value and actual values are compared as follows:

i) predicted value = "M" and actual value = "M", true positive count is incremented.

ii) predicted value = "M" and actual value = "B", false positive count is incremented

iii) predicted value = "B" and actual value = "M", false negative count is incremented

iv) predicted value = "B" and actual value = "B", true negative count is incremented

11) Based on the above count the Recall, Precision and Accuracy is calculated using the above counts.

12) This calculations are done for each fold in cross validation and the average of these is calculated as the final values.

3. **Naive Bayes** (10 points) From Project Gutenberg, we downloaded two files: The Adventures of Sherlock Holmes by Arthur Conan Doyle (pg1661.txt) and The Complete Works of Jane Austen(pg31100.txt). Please develop a multinomial Naive Bayes Classifier that will learn to classify the authors from a snippet of text into: Conan Doyle or Jane Austen. Report the recall, precision, and accuracy on your testing data. Write down each step of your solution. **Do not use any package/tool**.

Solution:

Not Attempted.

4. **Linear classification** (10 points) Please prove that 1) the multinomial naive Bayes classifier essentially translates to a linear classifier. 2) Logistic regression is a linear classifier.

Solution:

1) The multinomial naive Bayes classifier essentially translates to a linear classifier.

Naive Bayes classifiers are linear classifiers that are simple yet very efficient. The probabilistic model of naive Bayes classifiers is based on Bayes theorem. The prefix naive is for the assumption that the features in a data-set are mutually independent; no two feature influence other's probabilities. The probability model for Naive Bayes is as follows,

posterior probability = $\frac{conditionalprobability \times priorprobability}{evidence}$

The general notation of the posterior probability can be written as,

$P(c_j|x_i) = \frac{P(x_i|c_j) \times P(c_j)}{P(x_i)}$

where,

x_i is the feature vector

$c_j$is the class label of j classes

The posterior probability determines the probability of the object belonging to a particular class by its given features.

Prior probabilities are also called class priors, it describes the general probability of encountering a particular class in the whole data-set.

Class conditional probability or likelihood determines the likelihood of features belonging to a particular class.

The evidence is the probability of features in the whole data-set.

The objective in the naive Bayes probability is to maximize the posterior probability.

The naive Bayes classifier is not linear, but if the likelihood factors $P(x_i|c)$ are from exponential families, the Naive Bayes classifier corresponds to a linear classifier in a particular feature space.

For two classes,

$P(c = 1|x) = \frac{P(x|c=1) \times P(c=1)}{P(x|c=1) \times P(c=1) + P(x|c=0) \times P(c=0)}$

Dividing the numerator and denominator by $P(x|c = 1) \times P(c = 1)$, we get

$P(c = 1|x) = \frac{1}{1 + \frac{P(x|c=0) \times P(c=0)}{P(x|c=1) \times P(c=1)}}$

$= \frac{1}{1 + \exp - \ln \frac{P(x|c=0) \times P(c=0)}{P(x|c=1) \times P(c=1)}}$

$= \frac{1}{1 + \exp - \{\ln \frac{P(x|c=0)}{P(x|c=1)} + \ln \frac{P(c=0)}{P(c=1)}\}}$

$P(c = 1|x) = \sigma\{\sum_i \ln \frac{P(x_i|c=0)}{P(x_i|c=1)} + \ln \frac{P(c=0)}{P(c=1)}\}$

Which represents a logistic sigmoid function.

If $P(x_i|c)$ is a Multivariate Gaussian function,

$P(x_i|c) = \frac{1}{(\sum_k^{\frac{1}{2}} \sqrt{2\pi}))} exp^{-\frac{1}{2} \times (x-\mu_k)^T \sum^{-1} (x-\mu_k)}$

We assume that the shape of Gaussian is same for all classes,

$\sum_k = \sum \forall k$

$P(c = 1|x) = \sigma\{-\frac{1}{2}(\mu_1 + \mu_2)^T \sum^{-1} (\mu_1 - \mu_2) + x^T \sum^{-1} (\mu_1 - \mu 2) \ + \ln \frac{P(c=0)}{P(c=1)}\}$

This is linear in x^T, due to the assumption made.

2) Logistic regression is a linear classifier.

A linear classifier develops a hyper-plane as the boundary of separation between classes. It is formed by taking a linear combination of the features, such that one 'side' of the hyper-plane predicts one class and the other 'side' predicts the other.

Logistic regression is an approach toward learning functions of the form P(Y —X), in the case where Y is discrete-valued, and X denotes a vector containing discrete or continuous values. Logistic Regression assumes a parametric form for the distribution P(Y —X), then directly estimates its parameters from the training data. The parametric model in the case where Y is Boolean is:

$P(Y=0|X) = \frac{1}{1+\exp\{w_0+\sum_i w_i X_i\}}$

$P(Y=1|X) = 1 - P(Y=0|X) = \frac{\exp\{w_0+\sum_i w_i X_i\}}{1+\exp\{w_0+\sum_i w_i X_i\}}$

We predict positive if $P(Y=1|X) > P(Y=0|X)$. The log odds functions for logistic regression is given as follows,

$\ln(\frac{P(Y=1|X)}{1-P(Y=0|X)}) = \ln(P(Y=1|X)) - \ln(1 - P(Y=0|X)))$

$= \ln\{\frac{\exp\{w_0+\sum_i w_i X_i\}}{1+\exp\{w_0+\sum_i w_i X_i\}}\} - \ln\{\frac{1}{1+\exp\{w_0+\sum_i w_i X_i\}}\}$

$= \ln\{\exp\{w_0 + \sum_i w_i X_i\}\} - \ln\{1 + \exp\{w_0 + \sum_i w_i X_i\}\} + \ln\{1 + \exp\{w_0 + \sum_i w_i X_i\}\} - \ln 1$

$= \{w_0 + \sum_i w_i X_i\} + 0$

$0 = w_0 + \sum_i w_i X_i$

The decision boundary is given by this equation, as $\frac{P(Y=1|X)}{P(Y=0|X)} = 1$ at the boundary separating the two classes.