Project Part 1: Density Estimation and Classification

Project Report

By Arun Deepak Chandrasekar

1.Naïve Bayes

Naïve Bayes is a generative model that uses the Bayes theorem to find the conditional probability of y given x where y is the label the classification algorithm must predict and x the dataset consisting of few different features.

Bayes theorem states that:

$$P(y|x) = \frac{P(y)P(x|y)}{P(x)}$$

Now given the dataset has two features- mean(f1) and standard deviation (f2) and there are two possible characters 7 and 8 as possible labels, and the fact that Naïve Bayes is naïve as we consider the features to be independent, we can modify the Bayes theorem as so:

$$P(y = 7 \,/\, f_1, f_2) \approx P(y = 7) \cdot P(f_1 \,/\, y) \cdot P(f_2|y)$$

$$P(y = 8/f_1\,, f_2) \approx P(y = 8) \cdot P(f_1/y) \cdot P(f_2|y)$$

P(y=7) and P(y=8) are priors which can simply be calculated by counting the number of labels in training set.

$$P(y=7)= 6265/(6265+5851)= 0.5170848464839881$$

$$P(y=8)= 5851/(6265+5851)= 0.4829151535160119$$

Further, posterior probabilities can be calculated by using normal distribution function as follows:

$$P(f_1|y=7) = \frac{1}{\sqrt{2\pi}\sigma_{f_1}} e \frac{-\left(f1\_test - \mu_{f_1}\right)^2}{2\left(\sigma_{f_1}\right)^2}$$

$$P(f_2|y=7) = \frac{1}{\sqrt{2\pi}\sigma_{f_2}} e \frac{-\left(f2\_test - \mu_{f_2}\right)^2}{2\left(\sigma_{f_2}\right)^2}$$

$$P(f_1|y=8) = \frac{1}{\sqrt{2\pi}\sigma_{f_1}} e \frac{-\left(f1\_test - \mu_{f_1}\right)^2}{2\left(\sigma_{f_1}\right)^2}$$

$$P(f_2|y=8) = \frac{1}{\sqrt{2\pi}\sigma_{f_2}} e \frac{-\left(f2\_test - \mu_{f_2}\right)^2}{2\left(\sigma_{f_2}\right)^2}$$

Here the parameters are calculated as follows:

σ_{f_1}- Standard Deviation of feature f1(mean) in training set (respectively for 7 and 8)

σ_{f_2}- Standard Deviation of feature f2(standard deviation) in training set (respectively for 7 and 8)

μ_{f_1}- Mean of feature f1(mean) in the training set (respectively for 7 and 8)

μ_{f_2}- Mean of feature f2(standard deviation) in the training set (respectively for 7 and 8)

$f1\_test$- Feature f1(mean) for an element in the test set.

$f2\_test$- Feature f2(standard deviation) for an element in the test set.

Table 1- Estimated values of parameters of normal distribution

Parameter	Estimated value observed in program	
	Label 0 (Digit 7)	Label 1 (Digit 8)
σ_{f_1}	0.03063240469648835	0.03863248837395887
σ_{f_2}	0.038201083694320306	0.03996007437065856
μ_{f_1}	0.11452769775108769	0.1501559818936975

μ_{f_2}	0.28755656517748474	0.3204758364888714

For each element in the test set, the respective features are fed into the respective normal distribution equation and probability that it is a 7 and probability that it is 8 is obtained. The priors and posteriors are simply multiplied (since they are independent different posteriors can simply be multiplied is a major assumption in Naïve Bayes).

To classify whether the digit is 7 or 8, the algorithm looks which one has a greater probability. If probability of 7 is higher- then the algorithm classifies it as Label 0 i.e(digit 7) else it classifies it as label 1 i.e. (digit 8). The normal distribution is mainly used to calculate the posterior probabilities. The result is a vector consisting of predicted y labels of all 2002 data elements in the test set.

Classification Accuracy of Naïve Bayes

The predicted values are compared with the real values of y given in test set y. The number of correct predictions is noted and divided by total number of elements to obtain the accuracy of the algorithm.

- The overall accuracy observed for the Naïve Bayes classifier: **69.53046953046953 %**
- Classification accuracy of digit 7 by Naïve Bayes: **75.9727626459144 %**
- Classification accuracy of digit 8 by Naïve Bayes: **62.73100616016427 %**

2) Logistic Regression

Logistic Regression is a discriminative model that calculates P(y|x) directly. The program uses sigmoid function, gradient ascent (which is derived from log likelihood) to obtain the parameters $W=[w_0,w_1,w_2]$ and in turn describe the function boundary for classification.

Note : In the digit recognition program since we are using only 2 features (mean and standard deviation) the number of parameters is restricted to 3.

Important components of Logistic Regression:

a) Sigmoid function- Given training set X and weights W the sigmoid function returns a value between 0 to 1 and if the value is below 0.5 we consider it to be of label 0 and if it is more than 0.5 then label 1.

Sigmoid function:

$\sigma(z) = \frac{1}{1+e^{-z}}$ where z in this case is $w_0+w_1x_1+w_2x_2$ i.e in matrix notation $W^T.X$

b) Gradient Ascent

Major task involves fixing the values of the parameters W.

Firstly, we compute log likelihood and then find the gradient of it. It is observed that differentiating and equating it to 0 does not work in this case, so a step by step weight update operation is incorporated which is basically the essence of gradient ascent.

The following equation of gradient in matrix format was used:

$$\nabla\, log\ likelihood = X^T(Y - LR\ model\ predictions)$$

Here X is the data set, Y is the test labels and Y is subtracted by the label prediction of the Logistic Regression model with the previous weights. Based on this gradient the new weights will be such that it increases the maximum likelihood.

Then this gradient is used to update the weights using the following equation:

$$\text{Weights=Weights+ Learning rate } * \nabla\, log\ likelihood$$

The above steps are repeated up to n iterations till the log likelihood function is maximized. Initially the weights are set to 0, then as the iterations proceed the weights are calibrated such that the likelihood function is maximized. Once the weights for which the function is maximized is obtained, the test set is put into the equation containing those values to predict the digit to be 7 or 8.

Table 2 Estimated values of the parameters of Logistic Regression

Parameters (W)	Estimated value in program
W0	13.98053665
W1	168.21271586
W2	-118.53052209

Classification Accuracy of Logistic Regression

- Learning Rate- the program uses a learning rate of: **7e-4**
- Number of iterations gradient ascent was applied: **50,000**
- Classification accuracy observed on the entire test set: **79.47052947052947 %**
- Classification accuracy of digit 7 by Logistic Regression: **84.53307392996109 %**
- Classification accuracy of digit 8 by Logistic Regression: **74.12731006160165 %**