ML Assignment1

Arpita Agrawal , Collaborated with: Anjali K Prasad

September 2016

1 Density Estimation

(a) Suppose we have N i.i.d samples x1, x2, $\cdots$, xn. 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.

  Binomial Distribution = $\frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)}$ - (1)

  $$where, B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)} - (2)$$

  and,

  $$\Gamma(\alpha) = (\alpha-1)! - (3) and, \beta = 1(given) - (4)$$

  substituting (2),(3) in equation (1)

  $$\frac{x^{\alpha-1}(1-x)^{\beta-1}}{\frac{(\alpha-1)!(\beta-1)!}{(\alpha+\beta-1)!}} - (5)$$

  substituting (4) in equation (5)

  $$\frac{x^{\alpha-1}(1-x)^{1-1}}{\frac{(\alpha-1)!(1-1)!}{(\alpha+1-1)!}}$$

  $$\frac{x^{\alpha-1}}{\frac{(\alpha-1)!}{(\alpha)!}} = \frac{x^{\alpha-1}}{\frac{(\alpha-1)!}{(\alpha)!}} = x^{\alpha-1}(\alpha)$$

$$\log(p(x)) = \log(x^{\alpha-1}) + \log(\alpha)$$

$$\frac{\partial f}{\partial \alpha} = \frac{\partial(\alpha \log x - \log x + \log \alpha)}{\partial \alpha}$$

$$\log x - 0 + \frac{1}{\alpha} = 0$$

$$\alpha = \frac{-1}{\log x}$$

- We assume that all samples are generated from Normal distribution N(,). Please show how to derive the maximum likelihood estimator of .

$$\mathcal{N}(\theta, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp^{\frac{-(x-\theta)^2}{2\sigma^2}}$$

substituting $\sigma = \theta$

$$\mathcal{N}(\theta, \theta) = \frac{1}{\sqrt{2\pi\theta}} \exp^{\frac{-(x-\theta)^2}{2\theta}}$$

Taking log on both sides - to compute log likelihood

$$\log(\frac{1}{\sqrt{2\pi\theta}} \exp^{\frac{-(x-\theta)^2}{2\theta}})$$

$$\log(\frac{1}{\sqrt{2\pi\theta}}) + \log(\exp^{\frac{-(x-\theta)^2}{2\theta}})$$

$$\log(\frac{1}{\sqrt{2\pi\theta}}) - \frac{(x-\theta)^2}{2\theta}$$

$$\mathcal{N}(\theta, \theta) = p(x) = \frac{-1}{2} \log(2\pi\theta) - \frac{(x-\theta)^2}{2\theta}$$

Differentiating the above equation

$$\frac{\partial p(x)}{\partial \theta} = \frac{\partial(\frac{-1}{2} \log(2\pi\theta) - \frac{(x-\theta)^2}{2\theta})}{\partial \theta} - (1)$$

Simplifying equation (1)

$$\frac{-1}{2\theta} + (\frac{x^2}{\theta^2} - \frac{1}{2})$$

$$\theta^2 + \theta - 2x^2 = 0$$

Solving for θ *using quadratic formula*,

$$\theta = \frac{\pm 1 \pm \sqrt{1 - 4 \times 2x^2}}{2}$$

$$\theta = \frac{\pm 1 \pm \sqrt{1 - 8x^2}}{2}$$

b) Suppose random variables X1, X2, $\cdots$, Xn are i.i.d sampled according to density function f(x) and the kernel density estimation is in the form of $\hat{f}(x) = n$ Show the bias of the kernel density estimation method

E = $[x_1, x_2, \ldots, x_n]^T$

Kernel Density Estimator = $\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h} K(\frac{z - X_i}{h})$

Random variables sampled according to the probability density function = f(x)

$$\mathbf{E}_{X_1, X_2, .. X_n}[f(x)] = \int x f(x) \partial dx$$

$$\mathbf{E}_{X_1, X_2, .. X_n}[\hat{f}(x)] = \int x f(x) \partial x$$

For each i

$$\mathbf{E}_{X_i}[f(x)] = \int \frac{1}{h} K(\frac{z - t}{h}) f(t) \partial t$$

Hence, Expected Value of KDE $\hat{f}(x) =$

$$\mathbf{E}_{X_i}[\hat{f}(x)] = \frac{1}{n} \times n \int \frac{1}{h} K(\frac{z - t}{h}) f(t) \partial t$$

$$\mathbf{E}_{X_i}[\hat{f}(x)] = \int \frac{1}{h} K(\frac{z - t}{h}) f(t) \partial t$$

substituting z = $\frac{x-t}{h}$

substituting z, t = x- zh, $\partial t = \partial z h$

$$\mathbf{E}_{X_i}[\hat{f}(x)] = \int_{-\infty}^{\infty} \frac{1}{h} K(z) f(x - zh) \partial z h$$

$$\mathbf{E}_{X_i}[\hat{f}(x)] = \int_{-\infty}^{\infty} K(z) f(x - zh) \partial z$$

Using Taylor Expansion:

$$f(x-uh) = f(x)+f^{(1)}(x)(-uh)+\frac{1}{2}f^{(2)}(x)(-uh)^2+.....+\frac{1}{n!}f^{(n)}(x)(-uh)^n+O(h^{(n+1)})$$

$$\int_{-\infty}^{\infty} K(z)f(x-zh)\partial z = f(x)+f^{(1)}(x)(-h)\int_{-\infty}^{\infty} K(z)z\partial z+\frac{1}{2}f^{(2)}(x)(-z)^2\int_{-\infty}^{\infty} K(z)z^2\partial z+O(h^3)$$

Using below properties, $f^{(1)}$ term in the above equation cancels out:

$$\int_{-\infty}^{\infty} K(z) z \partial z = 0$$

$$\int_{-\infty}^{\infty} K(z) \partial z = 1$$

$$= f(x) + \frac{1}{2} f^{(2)}(x)(h)^2 \int_{-\infty}^{\infty} K(z) z^2 \partial z + O(h^3) - equation(3)$$

$$Bias(\hat{f}(x)) = \mathbf{E}_{X_i}[\hat{f}(x)] - f(x)$$

Subtracting f(x) from equation 3

$$= \frac{1}{2} f^{(2)}(x)(h)^2 \int_{-\infty}^{\infty} K(z) z^2 \partial z + O(h^3)$$

2 Naive Bayes

a) According to Naive Bayes Theorem,

$$P(Y = y_i|X) = \sum_{i=1}^{N} \frac{\prod_{j=1}^{n} P(x_j^i|y_i = k)P(y_i = k)}{P(X)}$$

$$P(Y=k|X) = \sum_{i=1,k=0/1}^{N} \frac{\prod_{j=1}^{n} P(x_j^i|y_i=k)P(y_i=k)}{P(X)}$$

$$P(Y=k|X) = \sum_{i=1}^{N} \frac{\prod_{j=1}^{n} P(x_j^i|y_i=k)P(y_i=k)}{\prod_{j=1}^{n} P(x_j^i|y_i=0)P(y_i=0) + \prod_{j=1}^{n} P(x_j^i|y_i=1)P(y_i=1)}$$

Dividing both numerator and denominator by numerator

$$P(Y=k|X) = \sum_{i=1}^{N} \frac{1}{1 + \frac{\prod_{j=1}^{n} P(x_j^i|y_i=1)P(y_i=1)}{\prod_{j=1}^{n} P(x_j^i|y_i=0)P(y_i=0)}}$$

Using $\mathrm{N}(\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma_{j,k}^2}} \exp^{\frac{-(x-\mu)^2}{2\sigma^2}}$ for feature distribution in our probability estimation

$$P(Y=1|X) = \sum_{i=1}^{N}\sum_{j=1}^{D} \frac{1}{1 + \frac{\frac{1}{\sqrt{2\pi\sigma_{j,0}^2}} \exp^{\frac{-(x-\mu_{j,0})^2}{2\sigma_{j,0}^2}}}{\frac{1}{\sqrt{2\pi\sigma_{j,1}^2}} \exp^{\frac{-(x-\mu_{j,1})^2}{2\sigma_{j,0}^2}}} \times \frac{1-\pi}{\pi}}$$

$$P(Y=1|X) = \sum_{i=1}^{N}\sum_{j=1}^{D} \frac{1}{1 + \frac{\exp^{\frac{-(x-\mu_{j,0})^2}{2\sigma_{j,0}^2}}}{\exp^{\frac{-(x-\mu_{j,1})^2}{2\sigma_{j,1}^2}}} \times \frac{1-\pi}{\pi}}$$

Taking exp(log) in the above equation, $\exp(\log) = 1\ \sum_{i=1}^{N}$ is required in the below equations to calculate σ, μ for each j, where $y_i = k$ (K = 0/1 in our case.)

$$P(Y=1|X) = \sum_{i=1}^{N}\sum_{j=1}^{D} \frac{1}{1 + \exp\log \frac{\exp^{\frac{-(x-\mu_{j,0})^2}{2\sigma_{j,0}^2}}}{\exp^{\frac{-(x-\mu_{j,1})^2}{2\sigma_{j,0}^2}}} \times \frac{1-\pi}{\pi}}$$

$$P(Y=1|X) = \sum_{i=1}^{N}\sum_{j=1}^{D} \frac{1}{1 + \exp -\frac{(x-\mu_{j,0})^2}{2\sigma_{j,0}^2} + \frac{(x-\mu_{j,1})^2}{2\sigma_{j,1}^2} + \log\frac{1-\pi}{\pi}}$$

$$P(Y=1|X) = \sum_{i=1}^{N}\sum_{j=1}^{D} \frac{1}{1 + \exp -\frac{(x-\mu_{j,0})^2}{2\sigma_{j,0}^2} + \frac{(x-\mu_{j,1})^2}{2\sigma_{j,1}^2} + \log\frac{1-\pi}{\pi}}$$

Simplifying the above equation:

$$P(Y=1|X) = \frac{1}{1+\exp\sum_{j=1}^{D}\frac{\mu_{j,0}-\mu_{j,1}}{\sigma_j^2}x_i + \frac{\mu_{j,0}^2-\mu_{j,1}^2}{2\sigma_j^2} + \log\frac{1-\pi}{\pi}}$$

The above equation can be written in the below form:

$$P(Y=1|X) = \frac{1}{1+\exp -w_o + w^T X}$$

where $w_o = \log\frac{1-\pi}{\pi} + \frac{\mu_{j,0}^2-\mu_{j,1}^2}{2\sigma_{j,1}^2}$

$$w^T = \frac{\mu_{j,0}-\mu_{j,1}}{\sigma_{j,1}^2}$$

b) Provide the maximum likelihood estimation for the parameters of the Naive Bayes with Gaussian assumption

Accord to Naive Bayes Theorem,

$$P(Y=y_i|X) = \sum_{i=1}^{N}\prod_{j=1}^{n} P(x_j^i|y_i=k)P(y_i=k)$$

Ignoring the P(x) factor in the denominator, as it is common in all the terms, and when you are taking maximum, the denominator will not make a difference.

$$P(Y=k|X) = \sum_{i=1,k=0/1}^{N}\prod_{j=1}^{n} P(x_j^i|y_i=k)P(y_i=k)$$

Using $\mathrm{N}(\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma_{j,k}^2}}\exp^{\frac{-(x-\mu)^2}{2\sigma^2}}$ for feature distribution in our probability estimation

Taking log on both sides - to compute log likelihood of Naive Bayes Estimation

$$LogLikelihood = \log\sum_{i=1,k=0/1}^{N}\prod_{j=1}^{n} P(x_j^i|y_i=k)P(y_i=k)$$

Applying logarithmic properties, to the above equation

$$LL = \sum_{i=1,k=0/1}^{N}\sum_{j=1}^{D} P(x_j^i|y_i=k) + \sum_{i=1,k=0/1}^{N} P(y_i=k)$$

Since, we are assuming our data follows Gaussian estimations, Substituting the Gaussian probability distribution parameters in the above equation :

$\mathrm{N}(\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma_{j,k}^2}} \exp^{\frac{-(x-\mu)^2}{2\sigma^2}}$

$$LL = \sum_{i=1,k=0/1}^{N} \sum_{j=1}^{D} \frac{1}{\sqrt{2\pi\sigma^2}} \exp^{\frac{-(x-\mu_{j,k})^2}{2\sigma^2}} + \sum_{k=0,1} (\pi^{N_k}(1-\pi)^{N-N_k}) - equation(1)$$

Partial differentiating the below equation (2) above with respect to $\mu_{j,k}$

$$\sum_{i=1,k=0/1}^{N} \sum_{j=1}^{D} -\log\sqrt{2\pi}\sigma_{j,k} + \frac{-(x-\mu_{j,k})^2}{2\sigma^2} + \sum_{k=0,1} (\pi^{N_k}(1-\pi)^{N-N_k}) - (2)$$

$$\frac{\partial(LL)}{\partial\mu_{j,k}} = \sum_{i=1,k=0/1}^{N} \sum_{j=1}^{D} -2\frac{(x-\mu_{j,k})}{2\sigma^2} = 0$$

$$\sum_{i=1,K=k}^{N} \mu_{j,k} = \sum_{i=1,k=0/1}^{N} \sum_{j=1}^{D} x_j^{(i)}$$

Summation of $\mu_{j,k} -> N_k$ *times(Take into account all the* x_i *where* $y_i = k$)

$$\hat{\mu}_{j,k} = \frac{1}{N_k} \sum_{i=1}^{N} x_j^i$$

Partial differentiating the above equation (2) above with respect to $\sigma_{j,k}$

$$\frac{\partial p(x)}{\partial\sigma_{j,k}} = \sum_{i=1,k=0/1}^{N} \sum_{j=1}^{D} \frac{-1}{\sigma_{j,k}} + 2\frac{(x-\mu_{j,k})^2}{2\sigma_{j,k}^3} = 0$$

$$\sum_{i=1,k=0/1}^{N} \sum_{j=1}^{D} -1 + 2\frac{(x-\mu_{j,k})^2}{2\sigma_{j,k}^2} = 0$$

$$-N + 2\frac{(x-\mu_{j,k})^2}{2\sigma_{j,k}^2} = 0$$

$$\sigma_{j,k}^2 = \frac{(x-\mu_{j,k})^2}{N_k}$$

Partial differentiating the above equation (2) above with respect to π

$$\frac{\partial p}{\partial \pi} = \frac{\partial log \pi^{N_k}(1-\pi)^{N-N_k}}{\partial \pi}$$

$$\frac{\partial p}{\partial \pi} = \frac{\partial N_k \pi + (N - N_k)(1-\pi)}{\partial \pi}$$

$$\frac{N_k}{\pi} - \frac{(N-N_k)}{(1-\pi)} = 0$$

$$N_k(1-\pi) - (N-N_k)\pi = 0$$

$$N_k - N_k\pi - N\pi - N_k\pi = 0$$

$$N_k - N\pi = 0$$

$$\pi = \frac{N_k}{N}$$

where, N_k = Number of training examples with y_i = k

3 Nearest Neighbor

Predict the Major of student located at location (20,7)

```
mean : mean [  12.76923077  12.30769231 ]

standard_deviation ( Feature wise ) :std [ 20.71695701  25.93062737]

Normalized Training Set:

  [[  1.0  -6.16366137e-01   1.41501812e+00   1.0]
 [  2.0  -9.54253598e-01   7.59422725e-01   1.0]
 [  3.0  -1.05079287e+00   1.33788925e+00   1.0]
 [  4.0   7.83453343e-01  -1.18659801e-02   2.0]
 [  5.0   1.74884609e+00   7.20858289e-01   2.0]
 [  6.0   1.16961044e+00   9.90809336e-01   2.0]
 [  7.0  -2.30209039e-01  -1.27559286e-01   3.0]
 [  8.0   1.11391471e-02  -5.13203638e-01   3.0]
 [  9.0  -9.05983961e-01  -5.90332509e-01   3.0]
 [  10.0  -1.63002852e+00  -1.18659801e-02   3.0]
 [  11.0   6.86914069e-01  -1.70870113e+00   4.0]
 [  12.0   3.00756971e-01  -1.01454130e+00   4.0]
 [  13.0   2.70000000e+01  -2.00000000e+01   4.0]]
```

Eculidean Distance =

$$\sqrt{\sum_{i=1}^{N}(p_i - q_i)^2}$$

```
Euclidean Distance Array(Distance of (20,7) from
   each normalized point in training set):

[[1.0, 1.8855852105564139],
[2.0, 1.6211258700274549],
[3.0, 2.0830361597698159],
[4.0, 0.47529672841019782],
[5.0, 1.6781331298803805],
[6.0, 1.4500248557016155],
[7.0, 0.58434818182463799],
[8.0, 0.45754752624908102],
[9.0, 1.3129253956141309],
[10.0, 1.9884264106288678],
[11.0, 1.5415002313248662],
[12.0, 0.81129037113461344],
[13.0, 33.198324562192816]]
```

Manhattan Distance =

$$\sum_{i=1}^{N}(p_i - q_i)$$

```
Manhattan Distance Array(Distance of (20,7) from
   each normalized point in training set):

[[1.0, 2.58509902532324],
[2.0, 2.267391087095559],
[3.0, 2.9423968902140283],
[4.0, 0.6272489115669935],
[5.0, 2.3253659263827022],
[6.0, 2.0160813258993024],
[7.0, 0.6563645176385212],
[8.0, 0.6464029427587663],
[9.0, 1.6406549212403907],
[10.0, 2.171877303992608],
[11.0, 1.8419004351819606],
[12.0, 0.858122777291916],
[13.0, 46.446285235825165]]

After, sorting the distance arrays(L1, L2), Eg, For k = 1, the nearest neighbor
is training data point 4(for L1) with class 2 (CS) :
```

	k = 1 (Neighbours)	k = 5 (Neighbours)
L1 (Manhattan Distance)	Electrical Engineering	Computer Science(win) tied with Economics
L2 (Euclidean Distance)	Computer Science	Computer Science(wins) tied with Economics

b) Using the fact that c Kc = K, derive the formula for unconditional density p (x).

Given:

$$P(X|Y = c) = \frac{K_c}{N_c V}$$

$$P(Y = c) = \frac{N_c}{N}$$

$$\sum_c N_c = N$$

$$\sum_c K_c = K$$

Summation of all the data points on all classes from sphere V = S

i) Using the fact that c Kc = K, derive the formula for unconditional density p (x).

$$P(x) = P(x|y = c_1)P(y = c_1)+P(x|y = c_2)P(y = c_2)+P(x|y = c_3)P(y = c_3)+...+P(x|y = c_n)P(y = c_n)$$

Summation of all marginal probabilities

$$= \sum_c \frac{K_c}{N_c V} \times \frac{N_c}{N}$$

$$= \sum_c \frac{K_c}{VN}$$

$$= \frac{1}{VN} \sum_c K_c$$

$$= \frac{K}{VN}$$

ii) Using Bayes rule, derive the formula for the posterior probability of class membership p (Y = c — x).

$$P(Y = c|x) = \frac{P(X|y = c).P(y = c)}{P(x)}$$

using the results retrieved from the above derivations,

$$P(Y = C|X) = \frac{\frac{K_c}{N_c V} \times \frac{N_c}{N}}{\frac{K}{VN}}$$

$$= \frac{K_c}{VN} \times \frac{VN}{K}$$

$$= \frac{K_c}{K}$$

4 Decision Tree

a) Which predictor variable (weather or traffic) will you choose to split in the first step to maximize the information gain?

Splitting on Traffic would be the better choice, because, Information gained when we split on traffic first is more than the information gained when you split on Weather. Calculation as per below:

Splitting on Traffic first:

Entropy(at Initial point)

$$= \sum_{k=1}^{N} -p_k \log p_k$$

At the intial point, there are 73 High Accident rates and 27 Low Accident rates.

$$H(x) = -p_+ \log p_+ - p_- \log p_-$$

$$H(x) = -\frac{73}{100} \log \frac{73}{100} - \frac{27}{100} \log \frac{27}{100}$$

$$H(x) = 0.84146$$

After splitting on Traffic Attribute, at Heavy Traffic Node -

$$H(HeavyTraffic) = -\frac{73}{73} \log \frac{73}{73} - \frac{0}{0} \log \frac{0}{0}$$

$$H(HeavyTraffic) = 0$$

$$H(LightTraffic) = -\frac{73}{73} \log \frac{27}{27} - \frac{0}{0} \log \frac{0}{0}$$

$$H(LightTraffic) = 0$$

H(Traffic) = H(Heavy Traffic) + H(Light Traffic) Information Gain = H(Y—x) -H(X) = 0.84146 - 0 IG1 = 0.84146

After splitting on weather Attribute,

$$H(InitNode) = -\frac{73}{100}\log\frac{73}{100} - \frac{27}{100}\log\frac{27}{100}$$

$$H(InitNode) = 0.84146$$

$$H(Sunny) = -\frac{23}{28}\log\frac{23}{28} - \frac{5}{28}\log\frac{5}{28}$$

$$H(Sunny) = 0.67694$$

$$H(Rainy) = -\frac{50}{77}\log\frac{50}{77} - \frac{22}{77}\log\frac{22}{77}$$

$$H(Rainy) = 0.88798$$

$$InformationGain = H(InitNode) - \frac{28}{100} \times H(Sunny) - \frac{72}{100} \times H(Rainy)$$

$$IG2 = 0.84146 - \frac{28}{100} \times 0.67694 - \frac{72}{100} \times 0.88798 = 0.012571$$

IG1 is greater than IG2 , Therefore, Splitting on Traffic first in contrast to Weather is a better option.

b) 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 T1. 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 T2.Even if the parameters are normalized feature wise, the split of the data will remain unchanged, we might be able to visualise the data better, the features will be more comparable to each other, But, the split of the data will remain the same. So, the tree T_1 *and* T_2 *will be the same.*

c) 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.

To prove Gini Index is a better a better approximation of the misclassification error.Prove **GI - CE** ≤ 0

$$\sum_{k=1}^{K} P_k(1-P_k) - \sum_{k=1}^{K} -P_k log P_k \leq 0$$

$$P_k(1-P_k) + P_k log P_k \leq 0$$

$$P_k(1-P_k+logP_k) \leq 0 - eq(1)$$

$$\leq (1-P_k+logP_k) - (as\ 0 \leq P_k \leq 1)$$

`if p_k > 0, then, the other term has to be < 0, to make the whole product  < 0`

$$\leq logP_k - (as\ 0 \leq P_k \leq 1)$$

`Addition of positive value make it even bigger`

$$\log P_k \leq 0$$

`because 0 < P_k < 1, so log(P_k) is going to be < 0 which is true, so our assumption is also true`

5 Programming

```
Data Inspection

How many attributes are?
10 + 1 Attributes (10 Feature vectors, and 1 output vector)

Do you think that all attributes are meaningful for the classification?
If not, explain why.

No, the first attribute, serial number is not meaningful for classification.
This column is a sequential number and is not random at all,
with respect to our data. All the other attributes seem
to be useful for our estimations


How many classes are? Class is a type of a glass.
There are 6 classes in the data (1,2,3,5,6,7)
```

Please explain the class distribution. Which class is majority? Do you think that it can be considered as a uniform distribution?

Class distributions are as follows numerically,
Class 1.0: 0.34183673469387754
Class 2.0: 0.37244897959183676
Class 3.0: 0.07142857142857142
Class 5.0: 0.05102040816326531
Class 6.0: 0.030612244897959183
Class 7.0: 0.1326530612244898

The above distribution seems to be following Bernoulli distribution,
with its peaks around class 1,2 and the tail at class(3,4,6,7)

Class 2 is in majority

No, I don't think we can consider this as a uniform distribution.

6 Performance Comparison

Estimating class based on K Nearest Neighbor Algorithm ::

k = 1
L2 (Eucledian):
Accuracy(Testing Data), :: 11 0.611111111111
Accuracy(Training Data) :: 140 0.714285714286
L1(Manhattan)
Accuracy(Testing Data), :: 12 0.666666666667
Accuracy(Training Data) :: 147 0.75

k = 3
L2 (Eucledian):
Accuracy(Testing Data), :: 11 0.611111111111
Accuracy(Training Data) :: 140 0.714285714286
L1(Manhattan)
Accuracy(Testing Data), :: 11 0.611111111111
Accuracy(Training Data) :: 144 0.734693877551

k = 5
L2 (Eucledian):

```
Accuracy(Testing Data), :: 10 0.555555555556
Accuracy(Training Data) :: 133 0.678571428571
L1( Manhattan )
Accuracy(Testing Data), :: 10 0.555555555556
Accuracy(Training Data) :: 133 0.678571428571


k =  7
L2 ( Eucledian ):
Accuracy(Testing Data), :: 10 0.555555555556
Accuracy(Training Data) :: 131 0.668367346939
L1( Manhattan )
Accuracy(Testing Data), :: 9 0.5
Accuracy(Training Data) :: 135 0.688775510204


Estimating class based on Naive Bayes Algorithm ::



Testing Data Accuracy on Naive Bayes Algorithm ::
0.333333333333

Training Data Accuracy on Naive Bayes Algorithm ::
0.551020408163


Comparision:

1) The results from KNN and Naive Bayes Algorithm are not similar, KNN has
a better accuracy than Naive Bayes. The reason for the bad performance
for Naive Bayes might be because, we have assumed Gaussian distribution
for the features given in the data for the data points. But, if we look
at the feature vectors and try and plot them on the graph, some of these
features are not following Gaussian distribution per say, and that might
be leading to the bad estimations of conditional probabilities
eventually leading to wrong estimations.

2) For KNN estimations, after normalising the data, the feature
vectors become more comparable and eventually help us obtain better
estimations for our testing data set.
```