1

a.Given $P(x|\mu) = ((1-\mu)/2)^{(1-x)/2}\ ((1+\mu)/2)^{(1+x)/2}$

And $x \varepsilon \{-1,1\}$

We find the probability over the space of x

$P(x|\mu) = \sum_{x=-1}^{1} ((1-\mu)/2)^{(1-x)/2}\ ((1+\mu)/2)^{(1+x)/2}$

$= ((1-\mu)/2)^{(1+1)/2}\ ((1+\mu)/2)^{(1-1)/2} + ((1-\mu)/2)^{(1-1)/2}\ ((1+\mu)/2)^{(1+1)/2}$

$= ((1-\mu)/2)^{(2)/2}\ ((1+\mu)/2)^{(0)/2} + ((1-\mu)/2)^{(0)/2}\ ((1+\mu)/2)^{(2)/2}$

$= ((1-\mu)/2) + ((1+\mu)/2)$

$= (1 - \mu\ 1 + \mu)/2$

$=2/2$

$=1$

So irrespective of μ the given probability mass function sums to 1

b.

$E[x] = \Sigma\, p(x)x$

$= \sum_{x=-1}^{1} ((1-\mu)/2)^{(1-x)/2}\ ((1+\mu)/2)^{(1+x)/2}\, x$

$= ((1-\mu)/2)^{(1+1)/2}\ ((1+\mu)/2)^{(1-1)/2} * -1 + ((1-\mu)/2)^{(1-1)/2}\ ((1+\mu)/2)^{(1+1)/2}\, 1$

$= ((1-\mu)/2)^{(2)/2}\ ((1+\mu)/2)^{(0)/2} * -1 + ((1-\mu)/2)^{(0)/2}\ ((1+\mu)/2)^{(2)/2}\, 1$

$= -((1-\mu)/2) + (1+\mu)/2$

$= (-1 + \mu + 1 + \mu)/2$

$=2\mu/2$

$=\mu$

Mean for the given probability mass function is μ

C. $Var[X] = E[X^2] - \mu^2$

$E[X^2] = \sum_{-1}^{1} x^2 p(x|\mu)$

$=1 * ((1-\mu)/2)^{(2)/2}\ ((1+\mu)/2)^{(0)/2} * + ((1-\mu)/2)^{(0)/2}\ ((1+\mu)/2)^{(2)/2} * 1$

$=(1 - \mu + 1 + \mu)/2$

$=1$

var[X] = E[X^2] - μ^2

var[X] = 1 - μ^2

Therefore variance for the probability mass function is

1 - μ^2

Question 2:

a. Submitted code on github

b.

Lenses Dataset

k	Accuracy
1	66
2	66
3	83
4	50
5	50

Crx Dataset

k	Accuracy
1	81
2	77
3	84
4	82
5	83

C. KNN was simple algorithm to implement.

Lenses Data Set: Form the results we see that for this data set the accuracy is highest when we consider 3 neighbours and then decreases. So for 3 neighbours the decision boundary is kind of defined correctly for the data. Another reason for the low accuracy might be because the number of features we are considering are just 4 and they are not weighted properly in computing the euclidean distance . Another possible reason would be, we are just using 18 training examples to train the model and maybe these examples are not holistic in representing the complete sample of data.

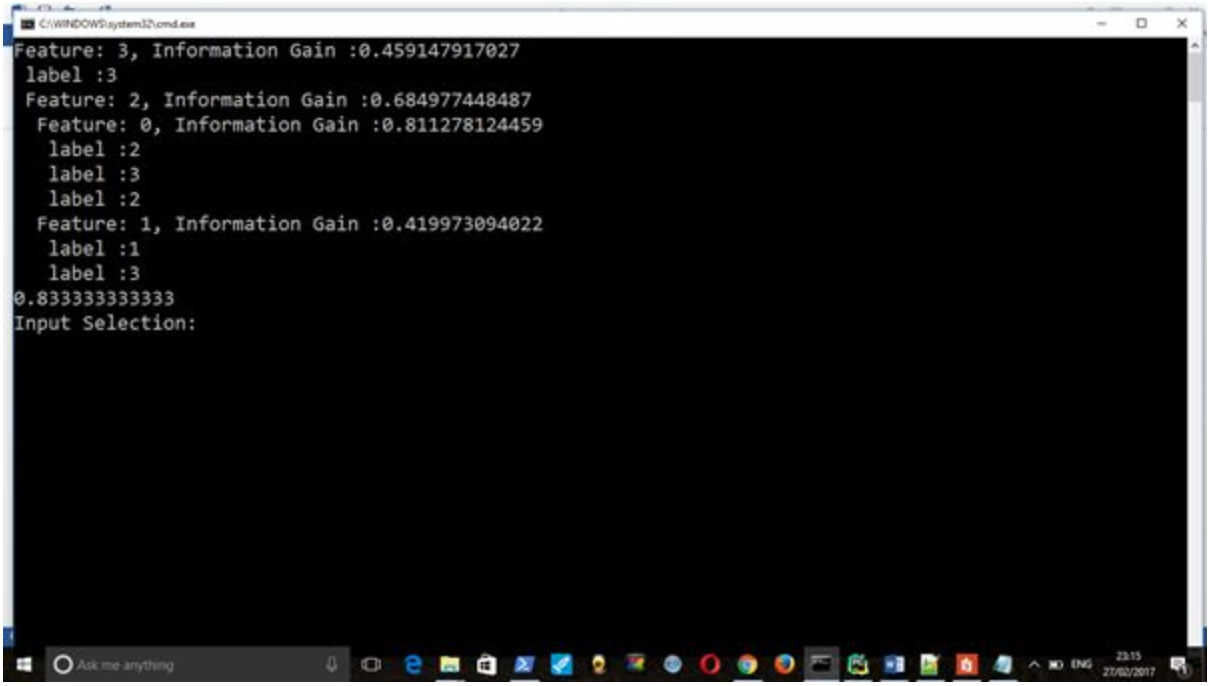
Crx Data Set: Compared to the lenses data set the accuracy for this data set is much better. Primary reasons for this could be one the decision boundary is consistent. Decision boundary is consistent because the number of features we consider are quite high compared to lenses. Also we have used almost 530 training examples, with so many examples the model learnt the decision boundary more precisely. Also the training examples seem to be holistic of the sample data.

For Crx since data was a mix of categorical values and real values, the categorical values had to be transformed into some sort of numbers to compute the euclidean distance. One decision i made was to one hot encode the data, doing this has greatly reduced the complexity when I compute the euclidean distance.

Question 3:

a. Submitted the code on github

b. For a tree of infinite height, the accuracy I get on lense data set is 83.33%



c.

Max Depth	Format	type	Accuracy
3	Binary	Standard	68.3333
5	Binary	Standard	68.333
7	Binary	Standard	73.33
infinity	Binary	Standard	66.6667
3	Multi	Standard	50.

5	Multi	Standard	53.33
7	Multi	Standard	53.33
Infinity	Multi	Standard	53.33

3	Binary	Extra	60.0
5	Binary	Extra	45
7	Binary	Extra	46.67
infinity	Binary	Extra	46.67
3	Multi	Extra	46.667
5	Multi	Extra	51.667
7	Multi	Extra	51.667
Infinity	Multi	Extra	51.667

d.

Training Dataset Size	Accuracy
40	55
80	55
160	55
234	66.67

E.

Form the results is pretty evident that the more features we have the closer we get to pick the correct hypothesis.

For multivariate the accuracy is never going beyond 53.3% and this might be possibly due to overfitting of data. For binary the accuracy goes upto to 73.33%. Just by changing the representation of data there is variability in the accuracy of prediction, this shows that trees are not that robust. Change in representation of data has an impact on the tree learning.

We also see that as we increase the depth of the tree, Decision tree does not generalize well. For the given data only at a particular depth of the tree the generalization is good after that depth the accuracy falls.

For the binary and extra data set features that I considered are

1. First name first character is vowel
2. Second Name first character is vowel
3. name has character Y
4. Name has character X or XU
5. Name has character w
6. Name has character o

7. Name has character q
8. Length of first name = length of second name
9. Last character in first name is vowel
10. Last character in second name is vowel

For this feature dataset the accuracy is very bad. These attributes are not representative of the data set.

For Multi and extra data set features that I considered are:

1. Length of first name + second name
2. Length of first name
3. Length of second name
4. Total Number of vowels
5. Total number of consonants
6. Even number of vowels
7. Even number of consonants
8. Total length is even
9. Total number of unique characters

This set of features also gave bad accuracy. These features were not representative of the data set

There has to be balance in the data and features for the decision tree to make predictions accurately.