Question 1:

How to get conditional probability

- **Step 1: Arrange the data by Construction type**

House ID	Local Price	Bathrooms	Land Area	Living area	# Garages	# Rooms	# Bedrooms	Age of home	Construction type
1	4.9176	1	3.472	0.998	1	7	4	42	Apartment
4	4.5573	1	4.05	1.232	1	6	3	54	Apartment
5	5.0597	1	4.455	1.121	1	6	3	42	Apartment
10	14.4598	2.5	12.8	3	2	9	5	14	Apartment
15	5.05	1	5	1.02	0	5	2	46	Apartment
17	8.2464	1.5	5.15	1.664	2	8	4	50	Apartment
20	9.0384	1	7.8	1.5	1.5	7	3	23	Apartment

House ID	Local Price	Bathrooms	Land Area	Living area	# Garages	# Rooms	# Bedrooms	Age of home	Construction type
2	5.0208	1	3.531	1.5	2	7	4	62	House
8	5.6039	1	9.52	1.501	0	6	3	32	House
11	5.8282	1	6.435	1.225	2	6	3	32	House
12	5.3003	1	4.9883	1.552	1	6	3	30	House
13	6.2712	1	5.52	0.975	1	5	2	30	House
16	5.6039	1	9.52	1.501	0	6	3	32	House
18	6.6969	1.5	6.902	1.488	1.5	7	3	22	House

House ID	Local Price	Bathrooms	Land Area	Living area	# Garages	# Rooms	# Bedrooms	Age of home	Construction type
3	4.5429	1	2.275	1.175	1	6	3	40	Condo
6	3.891	1	4.455	0.988	1	6	3	56	Condo
7	5.898	1	5.85	1.24	1	7	3	51	Condo
9	16.4202	2.5	9.8	3.42	2	10	5	42	Condo
14	5.9592	1	6.666	1.121	2	6	3	32	Condo
19	7.7841	1.5	7.102	1.376	1	6	3	17	Condo

- **Step 2: construct the look up tables for discrete values**

For example, bathroom, totally 3 values for the bathroom, and 7 samples belong to Apartment, 7 samples belong to House, 6 samples belong to Condo, respectively, the results are as below:

Bathroom	Type=Apartment	Type=House	Type=Condo
1	5/7.0	6/7.0	4/6.0
1.5	1/7.0	1/7.0	1/6.0
2.5	1/7.0	0/7.0	1/6.0

The same method for Garages, Bedrooms

# Garages	Type=Apartment	Type=House	Type=Condo
0	1/7.0	2/7.0	0/6.0
1	3/7.0	2/7.0	4/6.0
1.5	1/7.0	1/7.0	0/6.0
2	2/7.0	2/7.0	2/6.0

# Bedrooms	Type=Apartment	Type=House	Type=Condo
2	1/7.0	1/7.0	0/6.0
3	3/7.0	5/7.0	5/6.0
4	2/7.0	1/7.0	0/6.0
5	1/7.0	0/7.0	1/6.0

- **Step 3:For continuous features like Local Price, LandArea, Living area, Age of home , we calculate the conditional probability modeled with the normal distribution**

$$\hat{P}(X_j \mid C = c_i) = \frac{1}{\sqrt{2\pi}\sigma_{ji}} \exp\left(-\frac{(X_j - \mu_{ji})^2}{2\sigma_{ji}^2} \right)$$

μ_{ji} : mean (avearage) of attribute values X_j of examples for which $C = c_i$

σ_{ji} : standard deviation of attribute values X_j of examples for which $C = c_i$

For example, local price,
# Apartment
local_price = [4.9176, 4.5573, 5.0597, 14.4598, 5.05, 8.2464, 9.0384]
Mean: np.mean(local_price) (numpy), `7.332742857142857`
Standard deviation: np.std(local_price), `3.347762921225858`

The distribution of local price for the apartment type should be: $\frac{1}{\sqrt{2\pi}*7.33} e\left(-\frac{(x-3.35)^2}{2*7.33^2}\right)$

Based on the above method, we can calculate other continuous valued input attributes

```
#########################    local_price   ############
Apartment
mean of arr :  7.332742857142857
std of arr :  3.347762921225858
House
mean of arr :  5.760742857142858
std of arr :  0.527829731358629
Condo
mean of arr :  7.415900000000001
std of arr :  4.209474116798915
#########################    Land Area   ############
Apartment
mean of arr :  6.103857142857143
std of arr :  3.0167935877385466
House
mean of arr :  6.6309
std of arr :  2.0821446093597133
Condo
mean of arr :  6.0246666666666675
std of arr :  2.323053282978149
#########################    Living Area   ############
Apartment
mean of arr :  1.5050000000000001
std of arr :  0.6518753167373563
House
```

```
mean of arr :  1.3917142857142857
std of arr :  0.19712919207298277
Condo
mean of arr :  1.5533333333333335
std of arr :  0.8429827334464739
########################    Rooms   ############
Apartment
mean of arr :  6.857142857142857
std of arr :  1.2453996981544782
House
mean of arr :  6.142857142857143
std of arr :  0.6388765649999399
Condo
mean of arr :  6.833333333333333
std of arr :  1.462494064565354
########################    Age of home   ############
Apartment
mean of arr :  38.714285714285715
std of arr :  13.593215594703176
House
mean of arr :  34.285714285714285
std of arr :  11.78030178747903
Condo
mean of arr :  39.666666666666664
std of arr :  12.736648783028533
```

Question 2:

1.Accuracy:
Training set: 0.25
Test set: 0.6

2. What is the effect of restricting the maximum depth of the tree? Try different depths and find the best value.

You can build a complex tree when using a large depth value, and it will capture more features. But it may overfit in the decision tree as you fit for the training data.

- Larger the depth of the tree more are the chances of variance(overfitting).
- Whereas smaller the depth of the tree more are the chances of bias tree(underfitting).

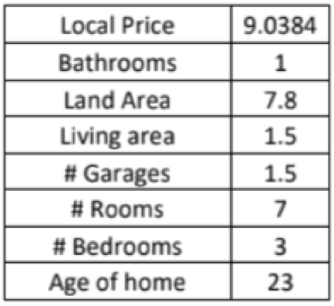
Depth	Training accuracy	Test accuracy	Comment

5	0.75	0.4	Overfit
4	0.75	0.4	
3	0.5	0.6	Good
2	0.25	0.4	Underfit

3.Why does restricting the depth have such a strong effect on the classifier performance?

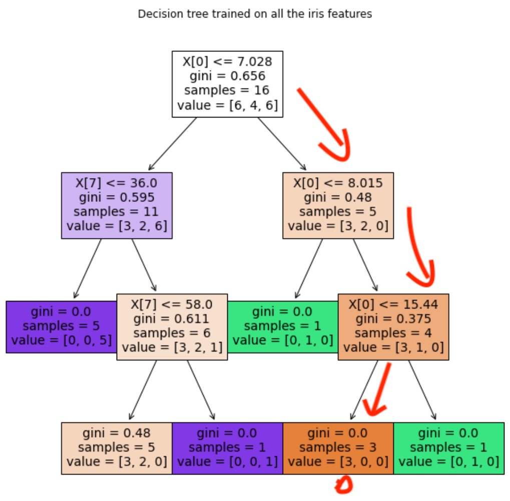
- The deeper the tree you build, it is more complex, and it will capture more features. But it may overfit in the decision tree as you fit for the training data. Increasing tree depth should increase performance on the training set, but it may lead overfitting
- Smaller depth of the tree may lead to underfitting, as it can not learn enough features from training set

4.Visualize the resulting tree. Perform the inference on this tree manually (i.e. show/trace the path taken towards classification) and provide a classification for the following example:



Local Price	9.0384
Bathrooms	1
Land Area	7.8
Living area	1.5
# Garages	1.5
# Rooms	7
# Bedrooms	3
Age of home	23

It is an apartment regarding the decision tree .


Decision tree trained on all the iris features
X[0] <= 7.028
gini = 0.656
samples = 16
value = [6, 4, 6]
X[7] <= 36.0
gini = 0.595
samples = 11
value = [3, 2, 6]
X[0] <= 8.015
gini = 0.48
samples = 5
value = [3, 2, 0]
gini = 0.0
samples = 5
value = [0, 0, 5]
X[7] <= 58.0
gini = 0.611
samples = 6
value = [3, 2, 1]
gini = 0.0
samples = 1
value = [0, 1, 0]
X[0] <= 15.44
gini = 0.375
samples = 4
value = [3, 1, 0]
gini = 0.48
samples = 5
value = [3, 2, 0]
gini = 0.0
samples = 1
value = [0, 0, 1]
gini = 0.0
samples = 3
value = [3, 0, 0]
gini = 0.0
samples = 1
value = [0, 1, 0]
0