The Chinese University of Hong Kong, Shenzhen

CSC4008

Techniques for Data Mining

Report for Density Based Classifier and Centroid Classification

Author:	*Student Number:*
LIU Yuxuan.	118010200
Wang Sixuan	118010305

March 10th, 2022

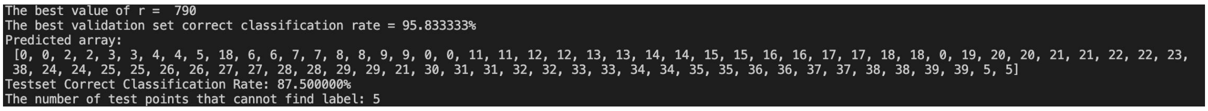
1. Density Based Classifier

1.1 Dataset and algorithm description

We are given with two datasets, training and testing, and their corresponding label set, but a validation set is still needed for determining the best value of r used in this density based algorithm. The method we adopted in this assignment is to randomly select a certain portion of data from the training set as validation set, for example 15%. In this way, we have 3 different sets: 1 training set, 1 validation set, and 1 test set. We first use the training set to predict the validation set with different values of r, and pick the r with the highest rate of correctness. Last, the r we just pick is used for test set prediction and this time, the rest 2 sets are combined as the training set.

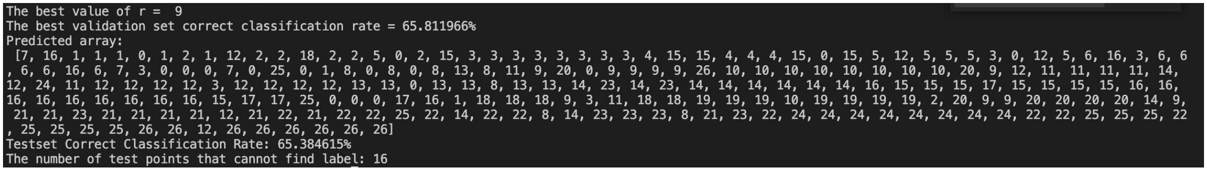
1.2 Results

ATNT Density-based:



```
The best value of r =  790
The best validation set correct classification rate = 95.833333%
Predicted array:
 [0, 0, 2, 2, 3, 3, 4, 4, 5, 18, 6, 6, 7, 7, 8, 8, 9, 9, 0, 0, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 0, 19, 20, 20, 21, 21, 22, 22, 23,
 38, 24, 24, 25, 25, 26, 26, 27, 27, 28, 28, 29, 29, 21, 30, 31, 31, 32, 32, 33, 33, 34, 34, 35, 35, 36, 36, 37, 37, 38, 38, 39, 39, 5, 5]
Testset Correct Classification Rate: 87.500000%
The number of test points that cannot find label: 5
```

BINALFAL Density-based:

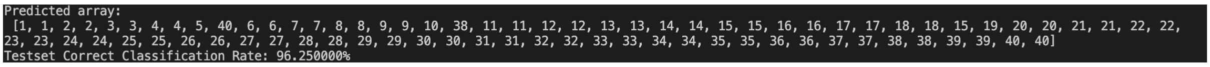


```
The best value of r =  9
The best validation set correct classification rate = 65.811966%
Predicted array:
 [7, 16, 1, 1, 1, 0, 1, 2, 1, 12, 2, 2, 18, 2, 2, 5, 0, 2, 15, 3, 3, 3, 3, 3, 3, 3, 3, 4, 15, 15, 4, 4, 4, 15, 0, 15, 5, 12, 5, 5, 5, 3, 0, 12, 5, 6, 16, 3, 6, 6
, 6, 6, 16, 6, 7, 3, 0, 0, 0, 7, 0, 25, 0, 1, 8, 0, 8, 0, 8, 13, 8, 11, 9, 20, 0, 9, 9, 9, 9, 26, 10, 10, 10, 10, 10, 10, 10, 10, 20, 9, 12, 11, 11, 11, 11, 14,
12, 24, 11, 12, 12, 12, 12, 3, 12, 12, 12, 12, 13, 13, 0, 13, 13, 8, 13, 13, 14, 23, 14, 23, 14, 14, 14, 14, 14, 14, 16, 15, 15, 15, 17, 15, 15, 15, 15, 16, 16,
16, 16, 16, 16, 16, 16, 16, 15, 17, 17, 25, 0, 0, 0, 17, 16, 1, 18, 18, 18, 9, 3, 11, 18, 18, 19, 19, 19, 10, 19, 19, 19, 19, 2, 20, 9, 9, 20, 20, 20, 20, 14, 9,
 21, 21, 23, 21, 21, 21, 21, 12, 21, 22, 21, 22, 22, 25, 22, 14, 22, 22, 8, 14, 23, 23, 23, 8, 21, 23, 22, 24, 24, 24, 24, 24, 24, 24, 24, 22, 22, 25, 25, 25, 22
, 25, 25, 25, 25, 26, 26, 12, 26, 26, 26, 26, 26, 26]
Testset Correct Classification Rate: 65.384615%
The number of test points that cannot find label: 16
```

1.3 Homework questions

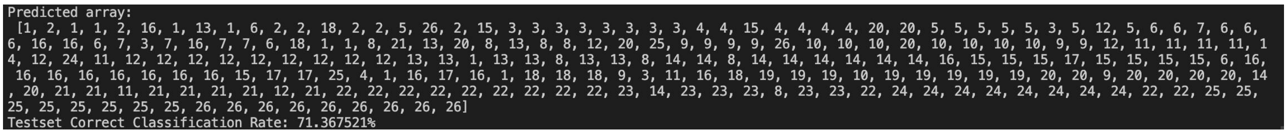
(i). Compare its performance with KNN method

ATNT KNN:



```
Predicted array:
 [1, 1, 2, 2, 3, 3, 4, 4, 5, 40, 6, 6, 7, 7, 8, 8, 9, 9, 10, 38, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 15, 19, 20, 20, 21, 21, 22, 22,
23, 23, 24, 24, 25, 25, 26, 26, 27, 27, 28, 28, 29, 29, 30, 30, 31, 31, 32, 32, 33, 33, 34, 34, 35, 35, 36, 36, 37, 37, 38, 38, 39, 39, 40, 40]
Testset Correct Classification Rate: 96.250000%
```

BINALFAL KNN:



```
Predicted array:
 [1, 2, 1, 1, 2, 16, 1, 13, 1, 6, 2, 2, 18, 2, 2, 5, 26, 2, 15, 3, 3, 3, 3, 3, 3, 3, 3, 4, 4, 15, 4, 4, 4, 4, 20, 20, 5, 5, 5, 5, 5, 3, 5, 12, 5, 6, 6, 7, 6, 6,
6, 16, 16, 6, 7, 3, 7, 16, 7, 7, 6, 18, 1, 1, 8, 21, 13, 20, 8, 13, 8, 8, 12, 20, 25, 9, 9, 9, 9, 26, 10, 10, 10, 20, 10, 10, 10, 10, 9, 9, 12, 11, 11, 11, 11, 1
4, 12, 24, 11, 12, 12, 12, 12, 12, 12, 12, 12, 12, 13, 13, 1, 13, 13, 8, 13, 13, 8, 14, 14, 8, 14, 14, 14, 14, 14, 14, 16, 15, 15, 15, 17, 15, 15, 15, 15, 6, 16,
 16, 16, 16, 16, 16, 16, 16, 15, 17, 17, 25, 4, 1, 16, 17, 16, 1, 18, 18, 18, 9, 3, 11, 16, 18, 19, 19, 19, 10, 19, 19, 19, 19, 19, 20, 20, 9, 20, 20, 20, 20, 14
, 20, 21, 21, 11, 21, 21, 21, 21, 12, 21, 22, 22, 22, 22, 22, 22, 22, 22, 22, 23, 14, 23, 23, 23, 8, 23, 23, 22, 24, 24, 24, 24, 24, 24, 24, 24, 22, 22, 25, 25,
25, 25, 25, 25, 25, 25, 26, 26, 26, 26, 26, 26, 26, 26, 26]
Testset Correct Classification Rate: 71.367521%
```

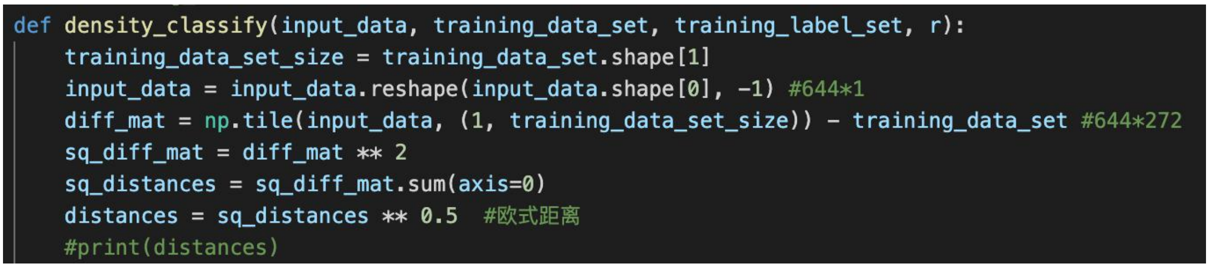
	KNN	Density-based
ATNT	96.25%	87.50%
BINALFAL	71.37%	65.38%

We can conclude from the table that the performance of KNN is better than density-based classifier.

(ii). Can this classifier work correctly, if not, list how many test points cannot find label(that is, no training point in its radius) under a specific radius

For ATNT dataset, when r=790, there are 5 test points cannot find their labels.
For BINALFAL dataset, when r=9, there are 16 test points cannot find their labels.

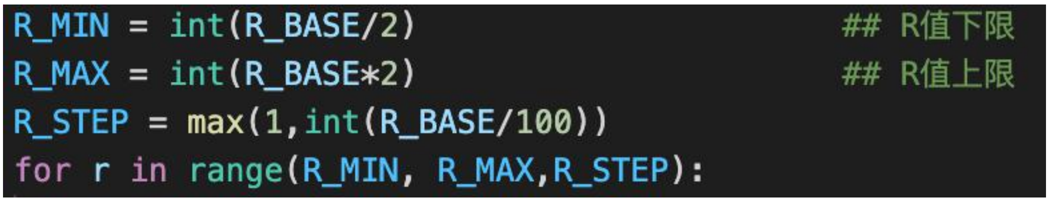
(iii). How to determine the ball radius r



```
def density_classify(input_data, training_data_set, training_label_set, r):
    training_data_set_size = training_data_set.shape[1]
    input_data = input_data.reshape(input_data.shape[0], -1) #644*1
    diff_mat = np.tile(input_data, (1, training_data_set_size)) - training_data_set #644*272
    sq_diff_mat = diff_mat ** 2
    sq_distances = sq_diff_mat.sum(axis=0)
    distances = sq_distances ** 0.5  #欧式距离
    #print(distances)
```

First in the density_classify() function, we have calculated the distances between test point and training points. We can use print(distances) to determine the order of magnitude. For instance, the baseline of r, called **R_BASE**, for ATNT dataset is 1000, while for BINALFAL dataset is 10.

Next, based on this **R_BASE**, we will find the best r value through iterating different r values using validation set. Here is the definition of **R_MIN, R_MAX** and **R_STEP**.



```
R_MIN = int(R_BASE/2)                         ## R值下限
R_MAX = int(R_BASE*2)                         ## R值上限
R_STEP = max(1,int(R_BASE/100))
for r in range(R_MIN, R_MAX,R_STEP):
```

Finally, we will find the best r that generates the best validation accuracy. Using this as the best_r value for the test set.

(iv). What could be the difficulty or drawback of this classifier? For example, can you guess how many training data points in the ball of the test data point?

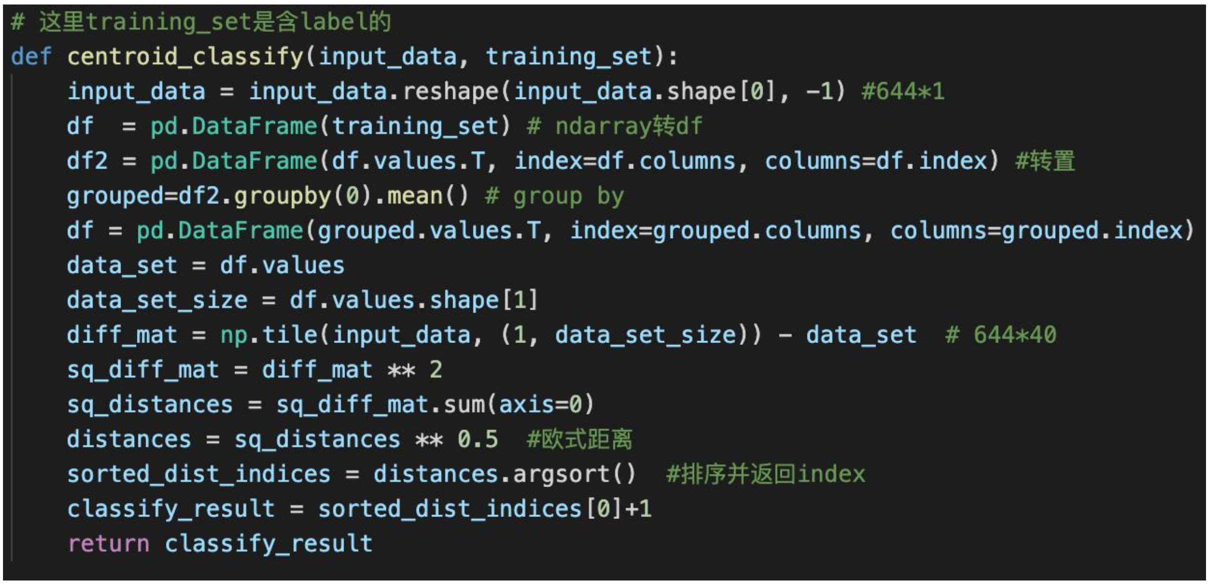
Difficulty: we need to determine the order of magnitude for each dataset.
Drawbacks:

1. Some test points cannot find their label because there is no training points fall in the range determined by a specific radius, so these points will be viewed as noise points.
2. Different from KNN, we can not guess how many training data points fall in the range.

2. Centroid Classification

2.1 Dataset and algorithm description

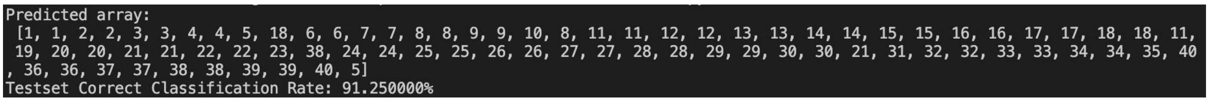
Since there is no hyper-parameter need to tune this case, there is no need for validation set. Use dataframe.groupby().mean() function to get the center for each label. Then calculate and sort the distances between centers and test point, find the closest center and use that label to predict the test point.



```
# 这里training_set是含label的
def centroid_classify(input_data, training_set):
    input_data = input_data.reshape(input_data.shape[0], -1) #644*1
    df  = pd.DataFrame(training_set) # ndarray转df
    df2 = pd.DataFrame(df.values.T, index=df.columns, columns=df.index) #转置
    grouped=df2.groupby(0).mean() # group by
    df = pd.DataFrame(grouped.values.T, index=grouped.columns, columns=grouped.index)
    data_set = df.values
    data_set_size = df.values.shape[1]
    diff_mat = np.tile(input_data, (1, data_set_size)) - data_set  # 644*40
    sq_diff_mat = diff_mat ** 2
    sq_distances = sq_diff_mat.sum(axis=0)
    distances = sq_distances ** 0.5  #欧式距离
    sorted_dist_indices = distances.argsort()  #排序并返回index
    classify_result = sorted_dist_indices[0]+1
    return classify_result
```

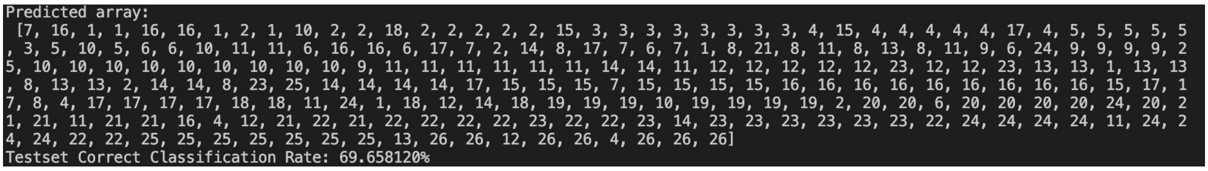
2.2 Results

ATNT centroid:



```
Predicted array:
 [1, 1, 2, 2, 3, 3, 4, 4, 5, 18, 6, 6, 7, 7, 8, 8, 9, 9, 10, 8, 11, 11, 12, 12, 13, 13, 14, 14, 15, 15, 16, 16, 17, 17, 18, 18, 11,
 19, 20, 20, 21, 21, 22, 22, 23, 38, 24, 24, 25, 25, 26, 26, 27, 27, 28, 28, 29, 29, 30, 30, 21, 31, 32, 32, 33, 33, 34, 34, 35, 40
, 36, 36, 37, 37, 38, 38, 39, 39, 40, 5]
Testset Correct Classification Rate: 91.250000%
```

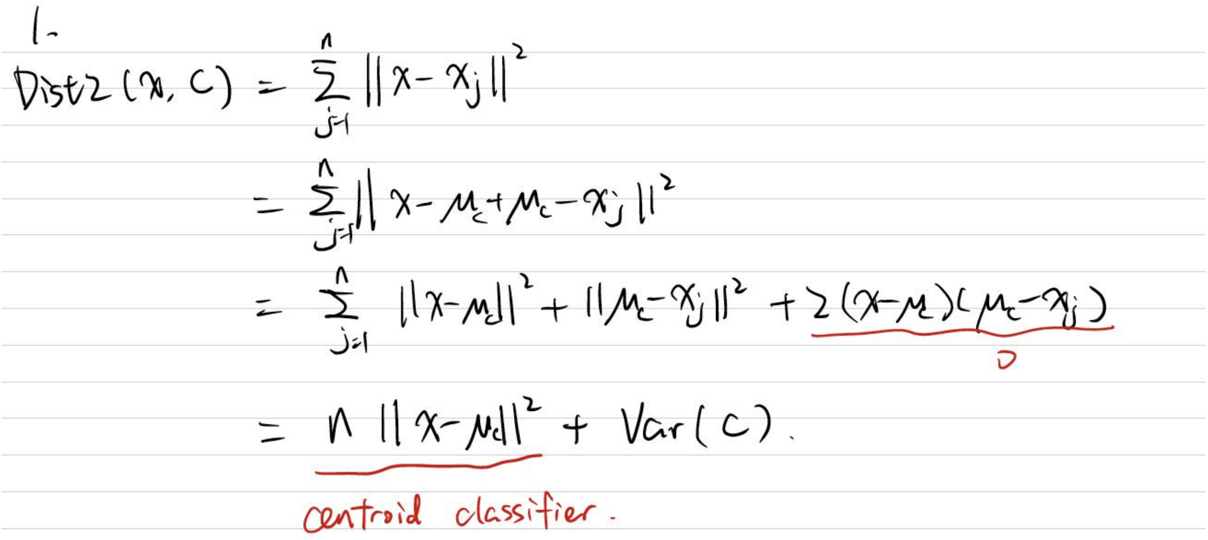
BINALFAL centroid:



```
Predicted array:
 [7, 16, 1, 1, 16, 16, 1, 2, 1, 10, 2, 2, 18, 2, 2, 2, 2, 2, 15, 3, 3, 3, 3, 3, 3, 3, 3, 4, 15, 4, 4, 4, 4, 4, 17, 4, 5, 5, 5, 5, 5
, 3, 5, 10, 5, 6, 6, 10, 11, 11, 6, 16, 16, 6, 17, 7, 2, 14, 8, 17, 7, 6, 7, 1, 8, 21, 8, 11, 8, 13, 8, 11, 9, 6, 24, 9, 9, 9, 9, 2
5, 10, 10, 10, 10, 10, 10, 10, 10, 10, 9, 11, 11, 11, 11, 11, 11, 14, 14, 11, 12, 12, 12, 12, 12, 23, 12, 12, 23, 13, 13, 1, 13, 13
, 8, 13, 13, 2, 14, 14, 8, 23, 25, 14, 14, 14, 14, 17, 15, 15, 15, 7, 15, 15, 15, 15, 16, 16, 16, 16, 16, 16, 16, 16, 16, 15, 17, 1
7, 8, 4, 17, 17, 17, 17, 18, 18, 11, 24, 1, 18, 12, 14, 18, 19, 19, 19, 10, 19, 19, 19, 19, 2, 20, 20, 6, 20, 20, 20, 20, 24, 20, 2
1, 21, 11, 21, 21, 16, 4, 12, 21, 22, 21, 22, 22, 22, 22, 23, 22, 22, 23, 14, 23, 23, 23, 23, 23, 23, 22, 24, 24, 24, 24, 11, 24, 2
4, 24, 22, 22, 25, 25, 25, 25, 25, 25, 25, 13, 26, 26, 12, 26, 26, 4, 26, 26, 26]
Testset Correct Classification Rate: 69.658120%
```

3. Math Proof

- Average squared distance classifier is identical to Centroid classifier with another parameter (details on slides page 49).



1.

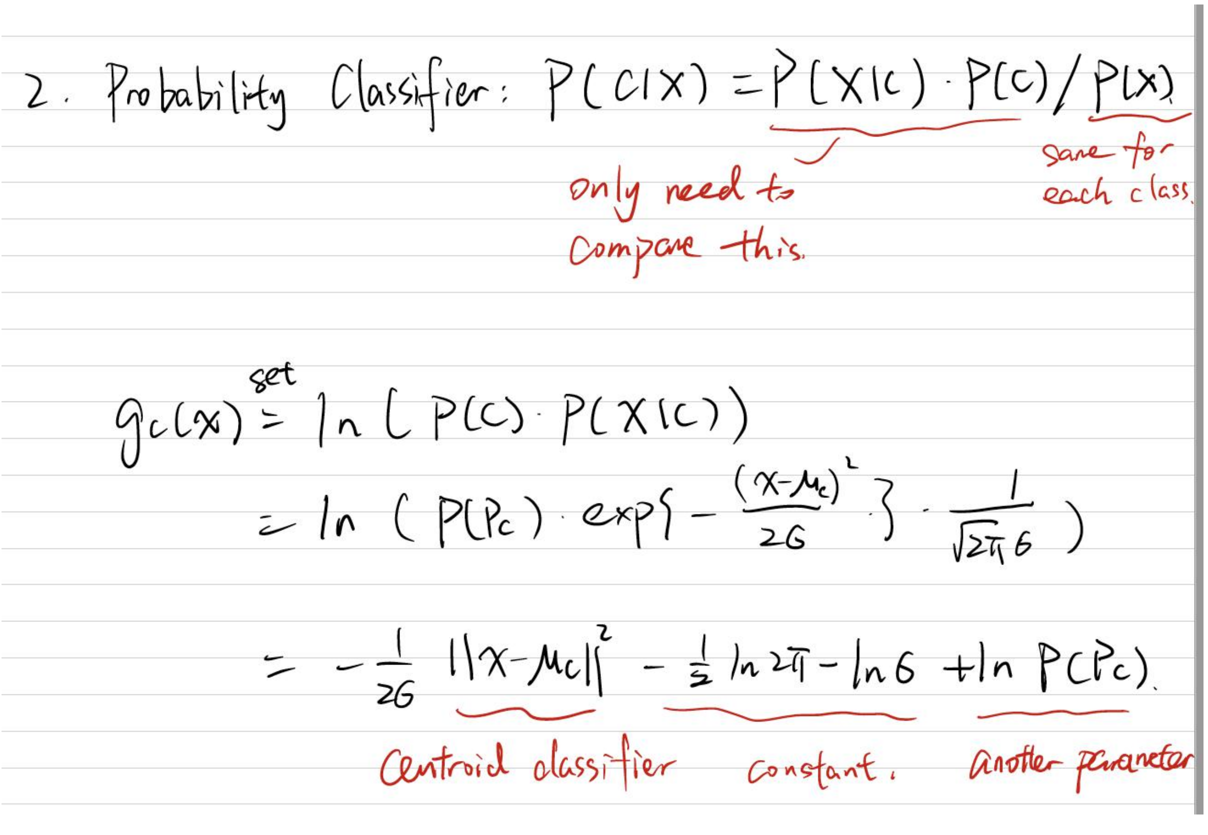
$$\mathrm{Dist2}(x, C) = \sum_{j=1}^{n} \|x - x_j\|^2$$

$$= \sum_{j=1}^{n} \|x-\mu_c+\mu_c-x_j\|^2$$

$$= \sum_{j=1}^{n} \|x-\mu_c\|^2 + \|\mu_c-x_j\|^2 + \underbrace{2(x-\mu_c)(\mu_c-x_j)}_{0}$$

$$= \underbrace{n\|x-\mu_c\|^2}_{\text{centroid classifier.}} + \mathrm{Var}(C).$$

- The probability classifier is identical to Centroid classifier (details on slides page 50 and 51).



2. Probability Classifier: $P(C|X) = \underbrace{P(X|C)\cdot P(C)}_{\text{only need to compare this.}} / \underbrace{P(X)}_{\text{same for each class.}}$

$$g_c(x) \overset{\text{set}}{=} \ln(P(C)\cdot P(X|C))$$

$$= \ln\left(P(P_c)\cdot \exp\left\{-\frac{(x-\mu_c)^2}{2\sigma}\right\}\cdot\frac{1}{\sqrt{2\pi}\sigma}\right)$$

$$= -\frac{1}{2\sigma}\underbrace{\|x-\mu_c\|^2}_{\text{Centroid classifier}} \underbrace{-\frac{1}{2}\ln 2\pi - \ln\sigma}_{\text{constant.}} + \underbrace{\ln P(P_c)}_{\text{another parameter}}.$$