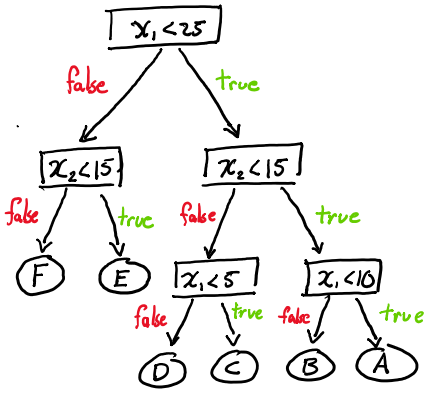
Homework4

Tuesday, November 26, 2024 3:42 PM

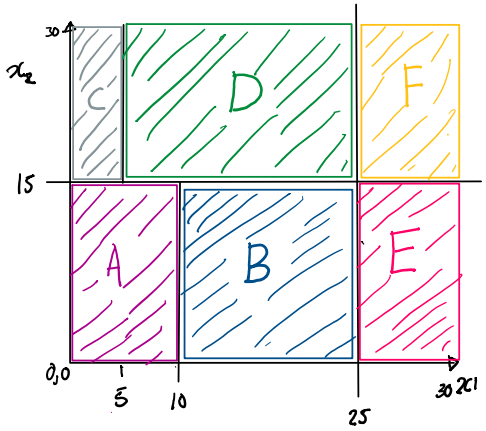
Hai Depweg

# Part 1:

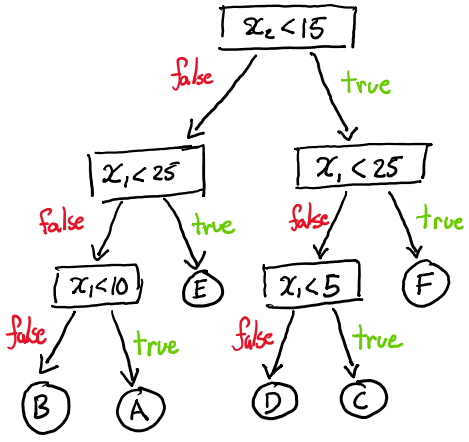
question 1




a) Draw the decision tree defined by this tree over the interval $x_1 \in [0,30]$, $x_2 \in [0,30]$. Each leaf of the tree is labled with a letter.




b) Give another decision tree that is syntactically different but defines the same decision boundaries




C) This demonstrates that the space of decision trees is syntactically redundant. How does this redundancy influence learning - ie., does it make it easier or harder to find accurate tree.

I think with redundancy it can be both easier or harder to find the accurate tree but it leans more to the easier side. It's important to note that with redundancy it adds complexity to the search process and adds the risk of overfitting. It's hard finding the simple model and most of the ti me we are going to be using the greedy

method to make these trees. This challenges though can be fixed though hypermeters like limiting tree depth. Using this we can ensure that the learned tree is generalized well and thus makes it easier to find an accurate tree. So in both ways its easier and harder.

Problem 2)

Consider the following training set and learn a decision tree to predict Y. Use information gain to select attributes to split.

| A | B | C | Y |
|---|---|---|---|
| 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 |

$H(Y) = -\sum_{p(Y)} \log_2(p(Y))$

$Y = \{0, 1\}$

$H(Y) = -\left(\frac{3}{6}\log_2\frac{3}{6} + \frac{3}{6}\log_2\frac{3}{6}\right) = 1$

Split on A

- for $A = 0$ $Y = \{0,1\}$

$H(Y|A=0) = -\left(\frac{2}{3}\log_2\frac{2}{3} + \frac{1}{3}\log_2\frac{1}{3}\right) \approx 0.918$

- for $A = 1$ $Y = \{0,1\}$

$H(Y|A=1) = -\left(\frac{2}{3}\log_2\frac{2}{3} + \frac{1}{3}\log_2\frac{1}{3}\right) \approx 0.918$

Weighted Entropy A

$H(Y|A) = \frac{3}{6}(0.918) + \frac{3}{6}(0.918) = 0.918$

Information gain A

$IG(A) = H(Y) - H(Y|A) = 1 - 0.918 = 0.082$

Split at B

for $B = 0$ $Y = \{0,1\}$

$H(Y|B=0) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$

for $B = 1$ $Y = \{0,1\}$

$H(Y|B=1) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$

Weighted entropy for B

$H(Y|B) = \frac{2}{6} \cdot 1 + \frac{4}{6} \cdot 1 = 1$

Information gain

$IG(B) = H(Y) - H(Y|B) = 1 - 1 = 0$

Information gain

$IG(B) = H(Y) - H(Y|B) = 1 - 1 = 0$

Split at C

for $C=0$

$H(Y|C=0) = -\left(\frac{2}{3}\log_2\frac{2}{3} + \frac{1}{3}\log_2\frac{1}{3}\right) = 0.918$

for $C=1$

$H(Y|C=1) = -\left(\frac{1}{3}\log_2\frac{1}{3} + \frac{2}{3}\log_2\frac{2}{3}\right) = 0.918$

Weighted Entropy for C

$H(Y|C) = \frac{3}{6}\cdot 0.918 + \frac{3}{6}\cdot 0.918 = 0.918$

Information Gain

$IG(C) = H(Y) - H(Y|C) = 1 - 0.918 = 0.082$

Note - both A and a C split give us same Information gain
I split at A but C works too.

**A<0**

A=0

| B | C | Y |
|---|---|---|
| 1 | 1 | 0 |
| 0 | 0 | 0 |
| 0 | 1 | 1 |

A=1

| B | C | Y |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |

A=0 side

Split B

$H(Y|B=0) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$

$H(Y|B=1) = -(1\log_2 1) = 0$

weighted

$\frac{1}{3}\cdot 0 + \frac{2}{3}\cdot 1 = \frac{2}{3}$

Gain

$H(Y) - H(Y|B) = 0.918 - \frac{2}{3} \approx 0.251$

Split C

$H(Y|C=0) = -(1\log_2 1) = 0$

$H(Y|C=1) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$

weighted

$\frac{1}{3}\cdot 0 + \frac{2}{3}\cdot 1 = \frac{2}{3}$

Gain

$H(Y) - H(Y|B) = 0.918 - \frac{2}{3} \approx 0.251$

both the same (redundant)

A=1

| B | C | Y |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |

Split B

$H(Y|B=0) = -(1 \log_2 1) = 0$

$H(Y|B=1) = -(\frac{1}{2} \log_2 \frac{1}{2} + \frac{1}{2} \log_2 \frac{1}{2}) = 1$

$0 \cdot \frac{1}{3} + 1 \cdot \frac{2}{3} = \frac{2}{3}$

$0.918 - \frac{2}{3} \approx 0.251$

Split C

$H(Y|C=0) = -(1 \log_2 1) = 0$

$H(Y|C=1) = -(\frac{1}{2} \log_2 \frac{1}{2} + \frac{1}{2} \log_2 \frac{1}{2}) = 1$

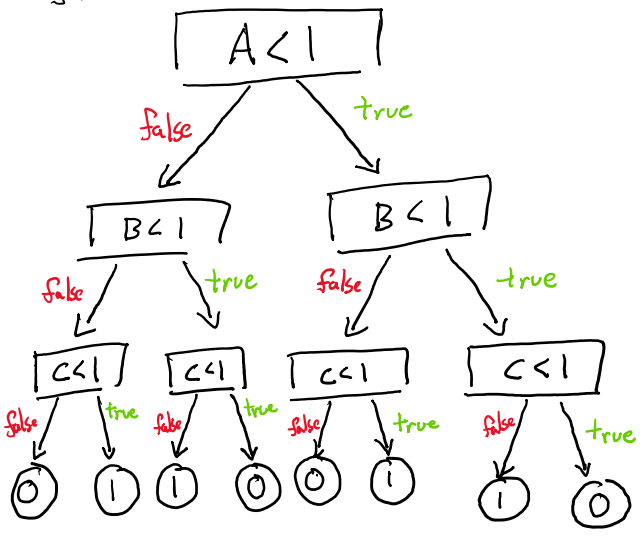
$\frac{1}{3} \cdot 0 + \frac{2}{3} \cdot 1 = \frac{2}{3}$

$0.918 - \frac{2}{3} \approx 0.251$

again redundant no split matters more

knowing that both splits after A split weigh the same we will chose B but you can split on C

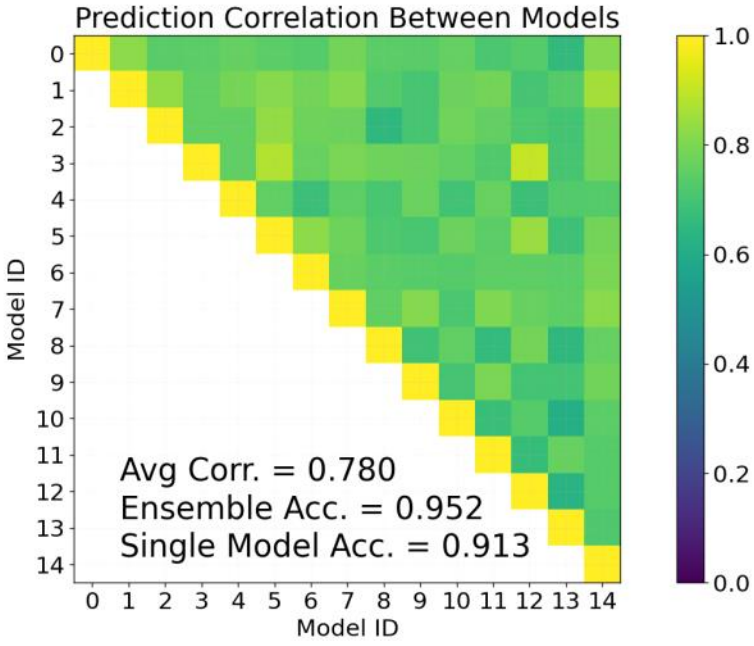
final tree




# Problem 3)

a)

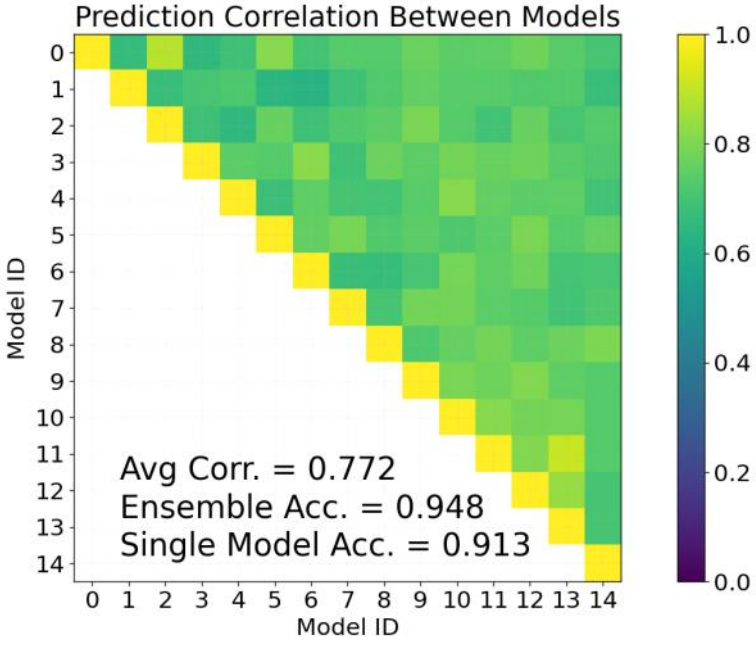
```
sample_indices = np.random.choice(len(X_train), len(X_train), replace=True)
#X_data = X_train
#y_data = y_train
X_data = X_train[sample_indices]
y_data = y_train[sample_indices]
```




When doing this we see that there is an increase to the Ensemble Accuracy and a decrease to the average correlation. The Single Model Accuracy stays the same value to the base case. This happens because each model is trained on a slightly differing data set introducig variability predictions and this reduces the redundancy among models. This allows for errors to be correct while keepings its accuracy constant.

b)

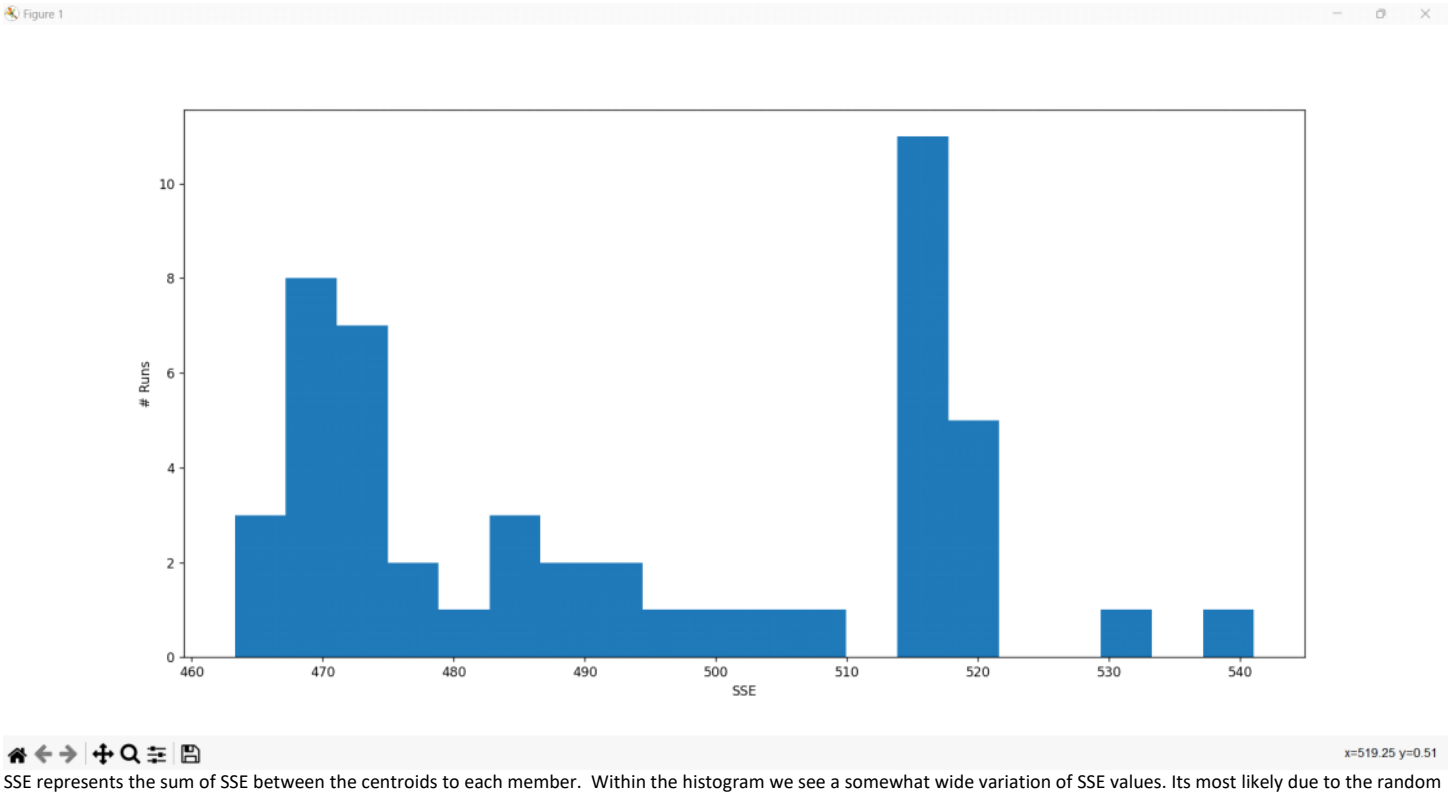
```
clf = tree.DecisionTreeClassifier(criterion="entropy", max_features=5, random_state=m)
```




Once again we see the same here where we have a decreased value in the Avg correlation comparative to the base case and an increased Ensemble accuracy. The single Model Acc. Again stays the same within this change. This happens for a few reasons mostly coming in the form of the model considering differing subsets and features to split.

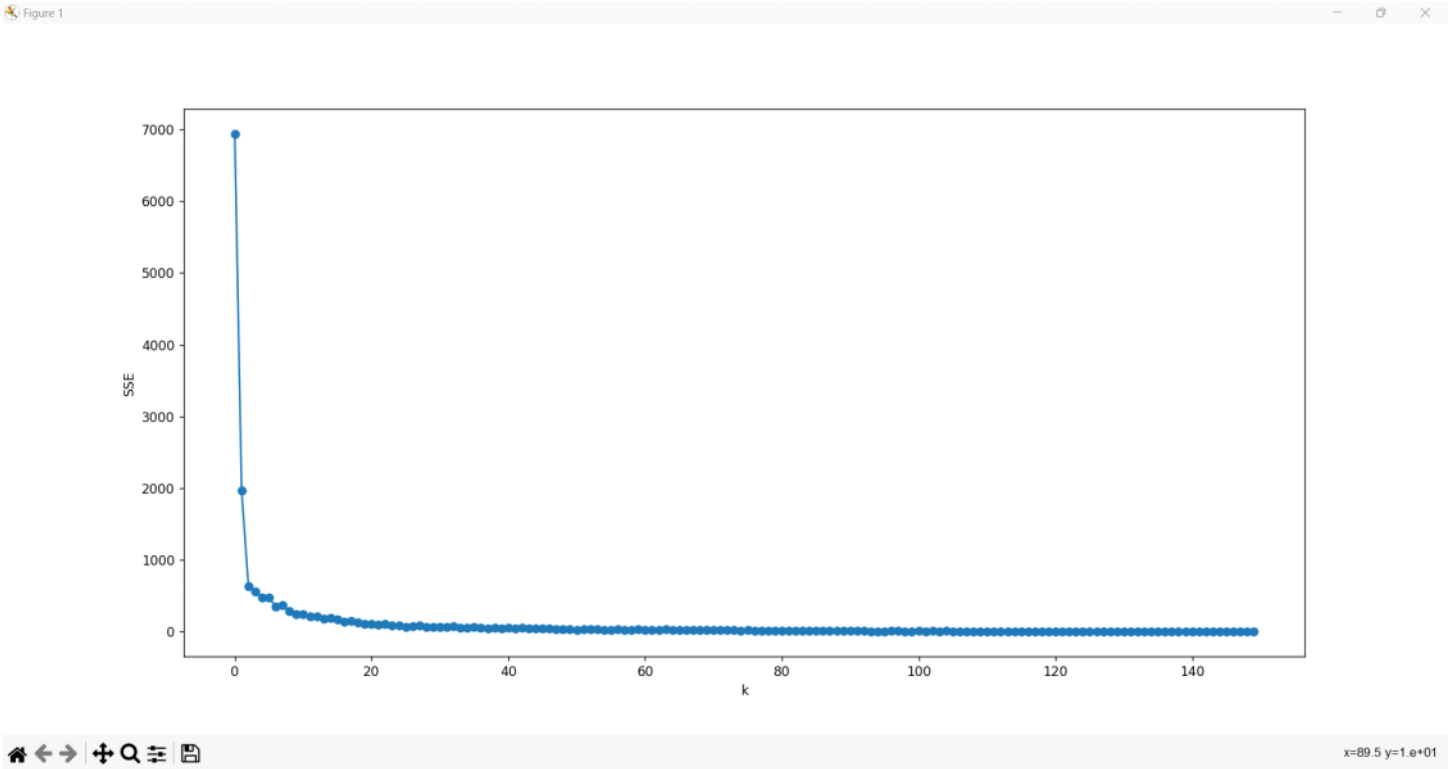
Comparing the two we observe something interesting when bagging we see an better increase to the ensemble accuracy compared to part b. This being said the avg correlation on the part b was lower than that of the bagging version. This shows the interplay between diversity and individual model quality. Both changed the avg correlation in a good way but both had slightly different effects when applying it to an ensemble.

## Question 5)




SSE represents the sum of SSE between the centroids to each member. Within the histogram we see a somewhat wide variation of SSE values. Its most likely due to the random initializations resulting in better cluster results than others. This means that a single run of k-means would not yield the best result often and the program should be ran more than one time and for the results to be compared. This is even more true for complex datasets where this problem can persist even fruther.

## Question 6)




In the plot we can see the SSE decrease as the number of clusters k increases, this is expected due because if we have a higher number of k but with the same number of datapoints each cluster contains less and less points. At a certain point if k = n then we would get an SSE of 0 but this is not great because of a few reasons. As we increase k beyond a certain point the SSE decrease comes with the tradeoff of overfitting the model and the development of fragmented clusters. This means that if our k number is too high we could lead to a place where we no longer capture the structure of the data but factures of the structure of data or in worst case each point as a cluster. Because of this we usually stop at a bracket looking drop to start off when looking at these graphs to chouse our k value.

## Question 7)

a. There are a few images clusters that displayed the main grouping seemed to be; skyscrapers, parks/trees, and highways/roads. It seems that the k=10 may seem to be too high but not by a significant amount. We can say this by looking at the images and the SSE graphs and outputs where we see that there is a significant drop between 1 and 3 and then

2 bracket shaped drops at somewhere around 7 and 10 meaning that it's a close value but we are still not overfitting too much.

b. After playing around with the hyper parameters I chose 7 due to it have somewhat the best results when looking at the images. I had tried numbers like 3 but this lead to the outputted images having an increased miss assignment.

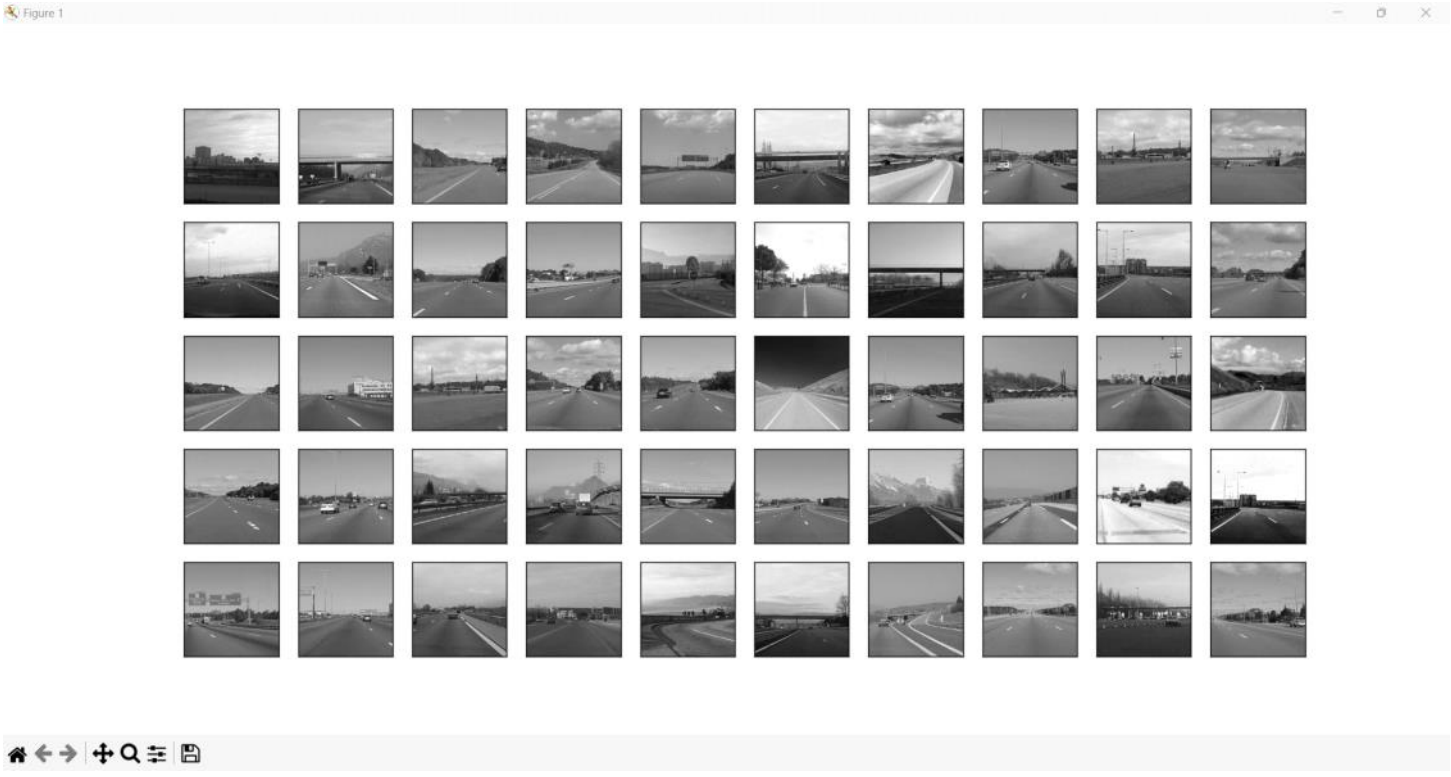




Figure 1

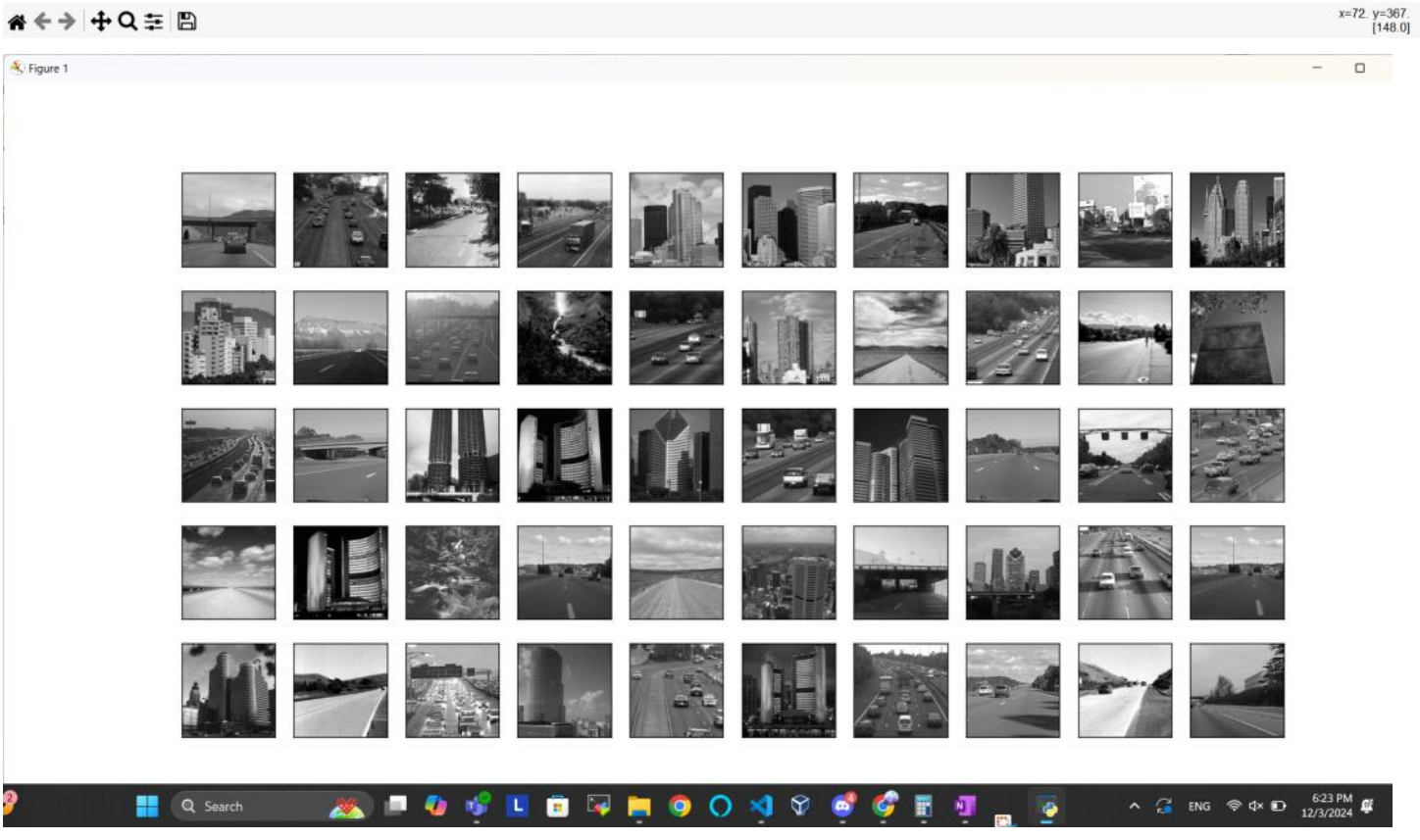
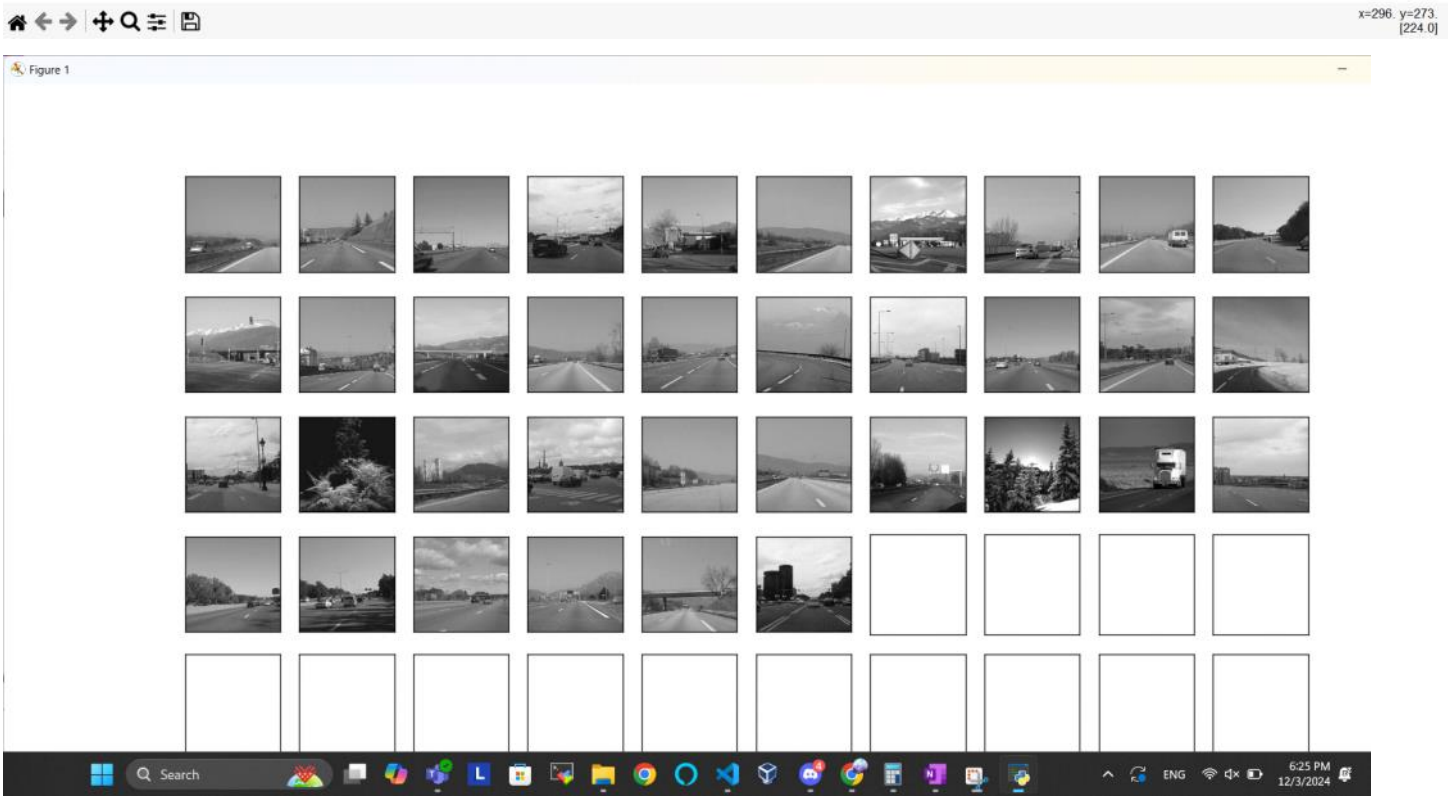

x=72. y=367.
[148.0]
Figure 1
Search
ENG
6:23 PM
12/3/2024


Figure 1


x=296. y=273.
[224.0]
Figure 1
Search
ENG
6:25 PM
12/3/2024

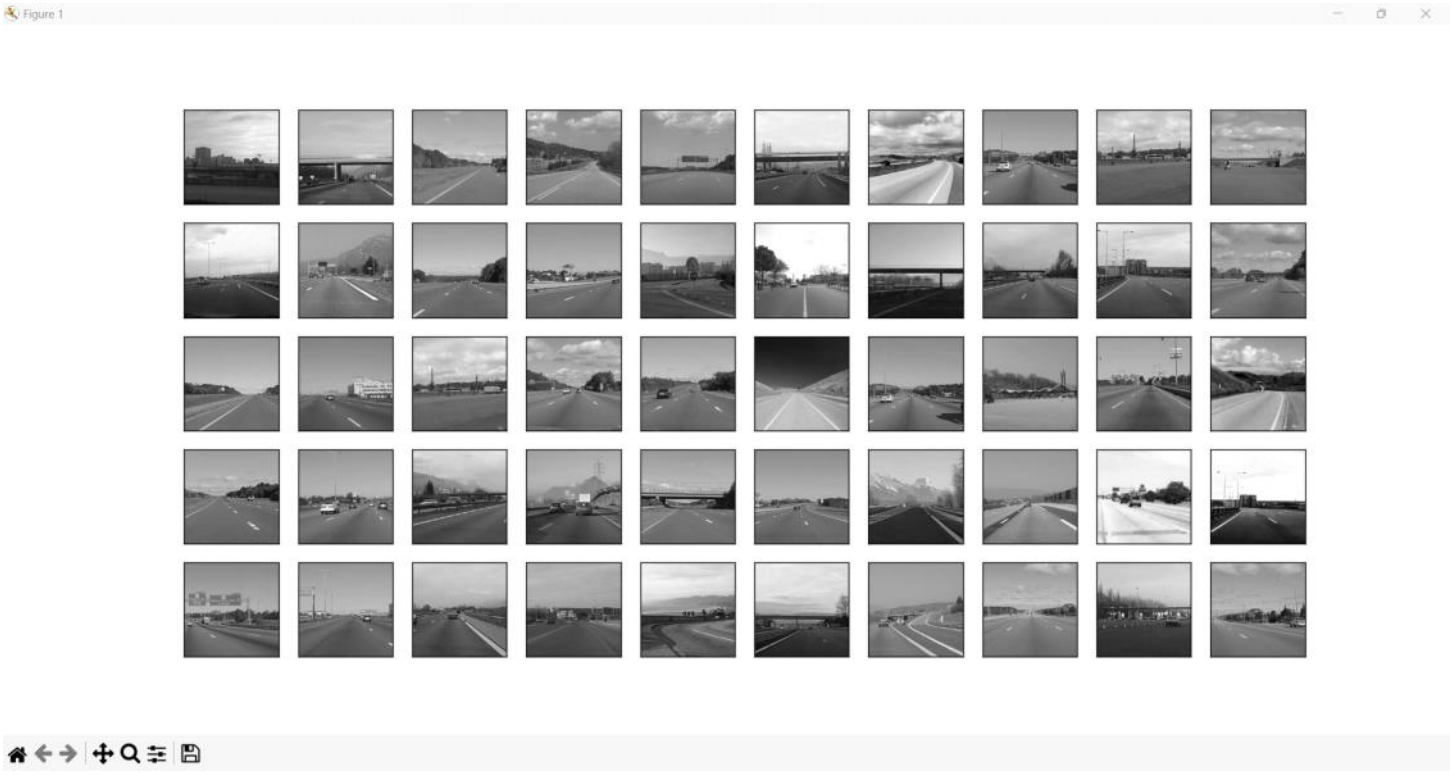

c)
It doesn’t seem that SSE is a great indicator of the clustering quality. This is because it can ignore the separation between clusters meaning that in our case k = 10 we get a better a sse value but it doesn't mean that our real cluster is better than if we set k = 7. This can be better represented when k = n where we are severely overfitting but we are not getting cluster.

## Problem 8)

Label: Sky Scraper


Purity 49/ 50

Label: roads

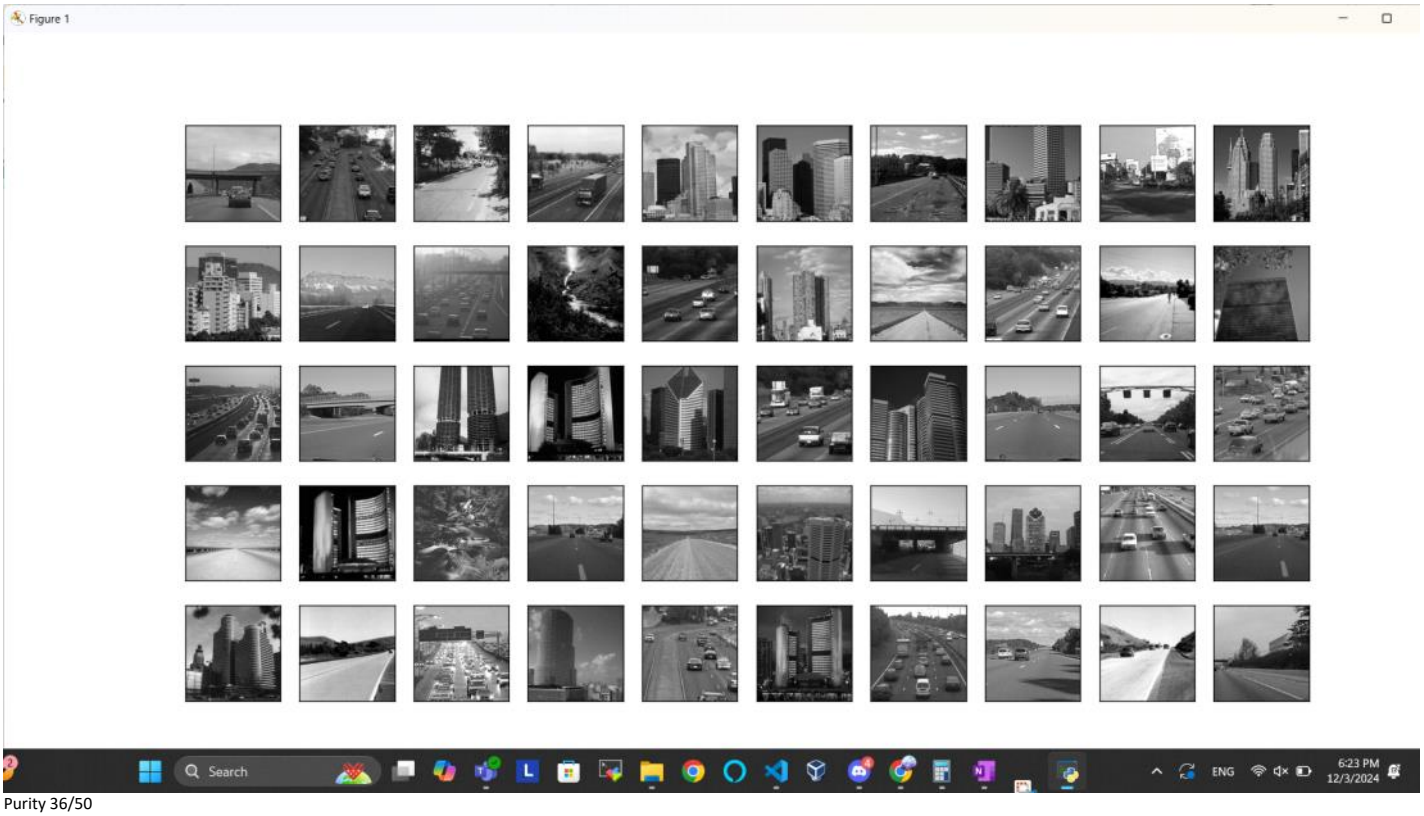



Purity: 49/50

Label: Trees


Purity 49/50

Label Roads?

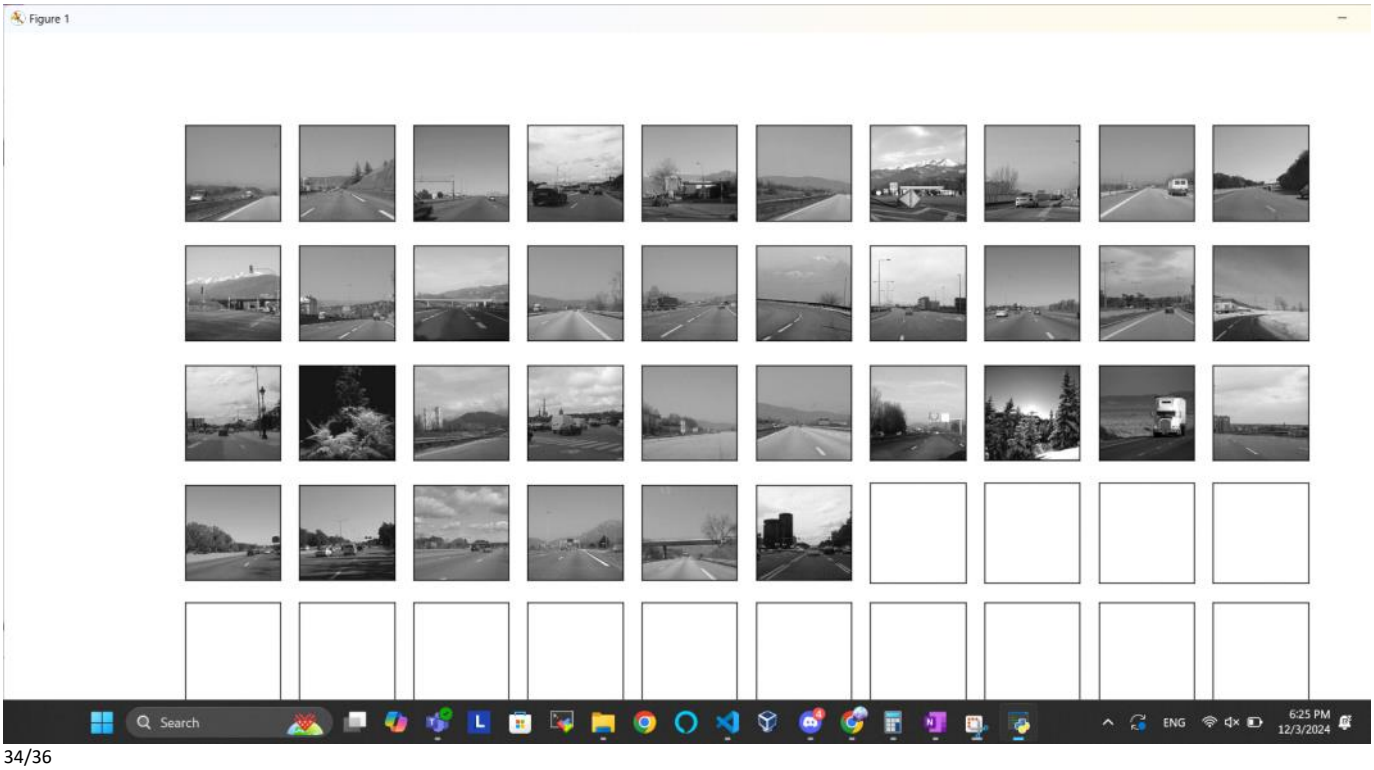



Purity 36/50

Label Sky Scrapers


Purity 50/50

Label: Roads




34/36

Label: Skycrapers


48/50

## Debriefing

1) Approximate how many hours
   10 hours
2) Would you rate it as easy moderate difficult
   I think I understand most of it
3) Did you work on it mostly alone or did you discuss the problem with others
   Alone
4) How deeply do you feel you understand the material it covers
   80%
5) Any other comments
   no