1.a
N_components,alpha,eta,perplexity = [(5, 1, 0.2, 494.84255091355107)]

```
no_top_words = 10
display_topics(lda, tf_feature_names, no_top_words)
```

```
Topic 0:
thi place like sandwich tri price better make good onli
Topic 1:
order food wait time servic came ask thi restaur tabl
Topic 2:
drink bar thi friend wine seat night tabl beer peopl
Topic 3:
dish chicken flavor good tast fri order sauc like veri
Topic 4:
great food good place veri servic love thi delici friendli
```

2.a
X3 = X1*X2 if X1*X2 > 0, then the target will be the yellow point, if X1*X2 < 0, the target will be the purple point

X3 = abs(X1) + abs(X2) - 1, if abs(X1) + abs(X2) - 1 < 0, target will be the blue point that stays in the small diamond, if abs(X1) + abs(X2) - 1 > 0, target will be the orange point that stays outside the small diamond.

2.b
Weather-whether the weather is good or bad could be impactful to passenger load since more people might walk or ride a bike if weather is good.

Whether the train goes through CBD or tourist attractions-trains that go through CBD or attractions might have significantly higher passenger load.

Number of neighborhoods that trains go through-trains that go through more neighborhoods might have more passengers on board.

Public holiday- trains might have significantly different passengers during normals days nad public holidays

3.a
The model is not doing a good job. Since there is only 1.4% of fraud in all transactions. If in the extreme case all transactions are predicted to be normal, we will have a model with 98.6% accuracy, which is larger than the tested model with an accuracy of 98%.

3.b
Our model is doing a good job. As the median of the single family home is around 770000, our model is only off by around 30000, which has an approximate MAPE of around 3.9%. Such MAPE is a lot smaller than 5.9% of Zillow, the tech company that specialized in home price predictions. Therefore, our model is doing a great job.

4.a
For forward execution:
Size 1: Considered: X1,X2,X3,X4 selected:X2
Size 2: Considered: X1,X3,X4 selected:X2,X3
Size 3: Considered: X1,X4 selected:X2,X3,X4

4.b
For backward execution:
Size 3 Considered removing X1,X2,X3,X4 selected:X3,X1,X4
Size 2 Considered removing X1,X3,X4 selected:X1,X4
Size 3 Considered removing X1,X4 selected:X1

4.c
Yes, this will always happen for all datasets with four predictors, because when selecting a subset with only 1 predictors, there are only 4 possible models, and forward selection tested all of them, and the one selected will be the best.

4.d
No, that is not always the case, when considering subsets of only two predictors, there are 6 possible models, but backward selection only considers 3 of them, so it is possible that the other 3 combinations of predictors might yield a lower MSE compared to the one selected by backward selection.
5.a

```
data = [(2, 5),(5, 3),(4, -5),(3, -3),(1, 5)]
data = pd.DataFrame(data,columns = ['x','y'])
data['predict'] = 10 - 3.5*data['x']
MSE = ((data['y'] - data['predict'])**2).mean()
print(f'The MSE is {MSE}')
```

```
The MSE is 24.75
```

5.b

```
data = [(2, 5),(5, 3),(4, -5),(3, -3),(1, 5)]
data = pd.DataFrame(data,columns = ['x','y'])
data['predict'] = 9 - 2.5*data['x']
MSE = ((data['y'] - data['predict'])**2).mean()
print(f'The MSE is {MSE}')
```

```
The MSE is 16.35
```

5.c
For lambda = 0.5
Model1 = Original MSE + Lambda*(B1)**2 = 24.75 + 0.5*((-3.5)**2) = 30.875
Model2 = Original MSE + Lambda*(B1)**2 = 16.35 + 0.5*((-2.5)**2) = 19.475
The second one is a better fit to training data with lower objective function

5.d
For lambda = 0.5
Model1 = Original MSE + Lambda*(B1)**2 = 24.75 + 0.5*abs(-3.5) = 26.5
Model2 = Original MSE + Lambda*(B1)**2 = 16.35 + 0.5*abs(-2.5) = 17.6
The second one is a better fit to training data with lower objective function