Assignment 2: Experiments with Bayesian Linear Regression

Introduction:

Bayesian regressions is the method of finding the best predictive model for an unknown data set. A Bayesian regression uses a Bayesian approach (hence the name) — it treats parameters as random variables with underlying distributions.

This assignment has three experiments:

1) Regularization
2) Learning Curves
3) Model selection
    a) Using Cross Validation
    b) Bayesian Model selection

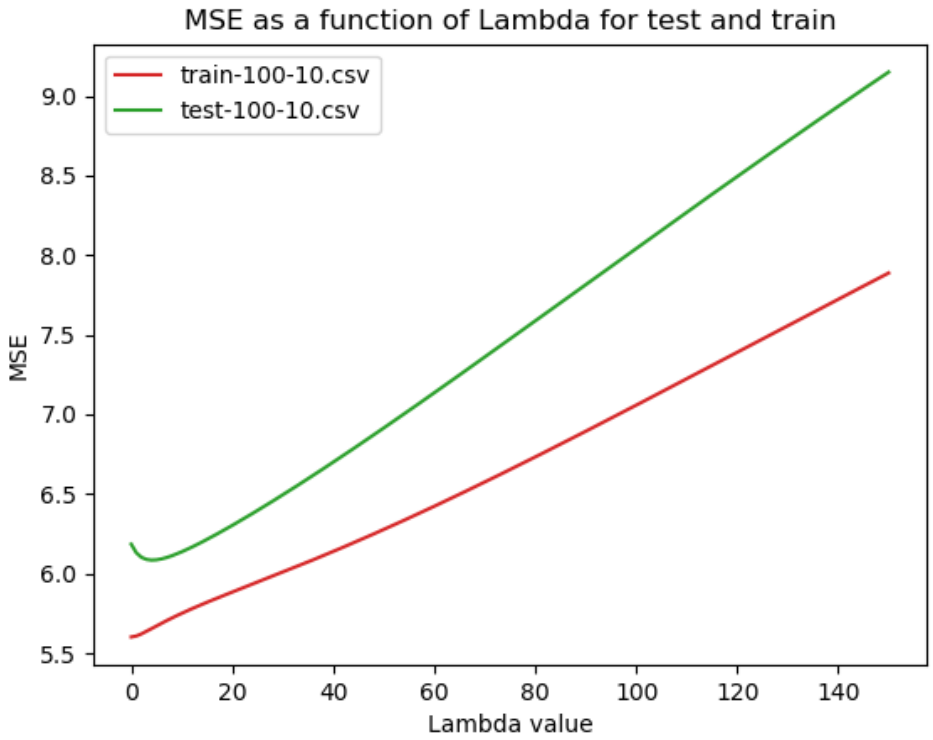
Experiment 1 – Regularization

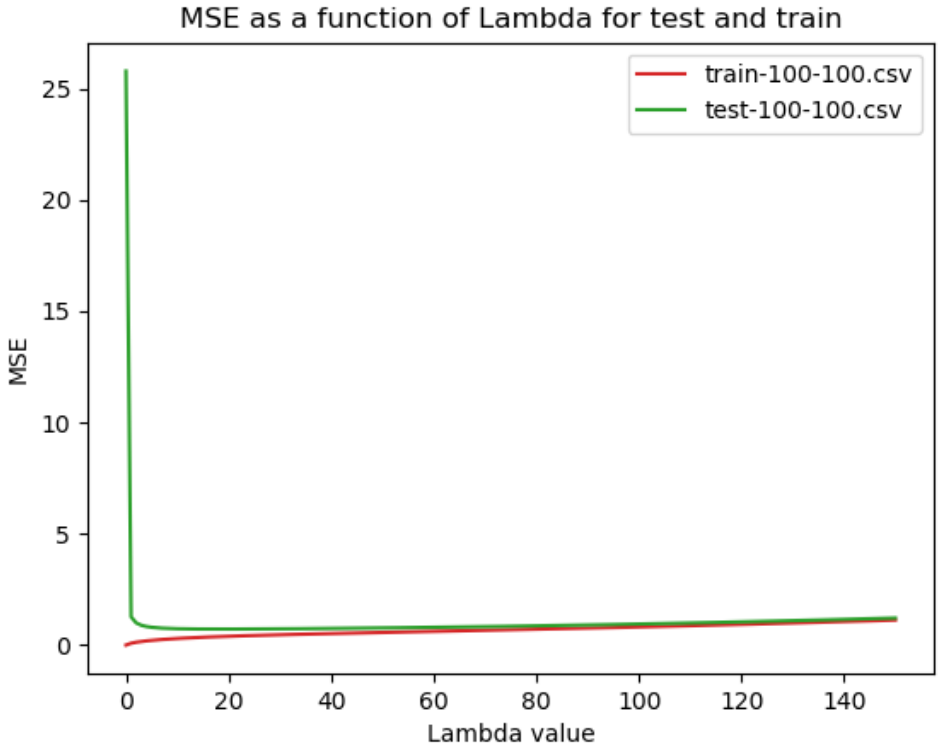
We have been given 5 datasets of train and test which has design and feature matrix pair as shown below for train datasets:

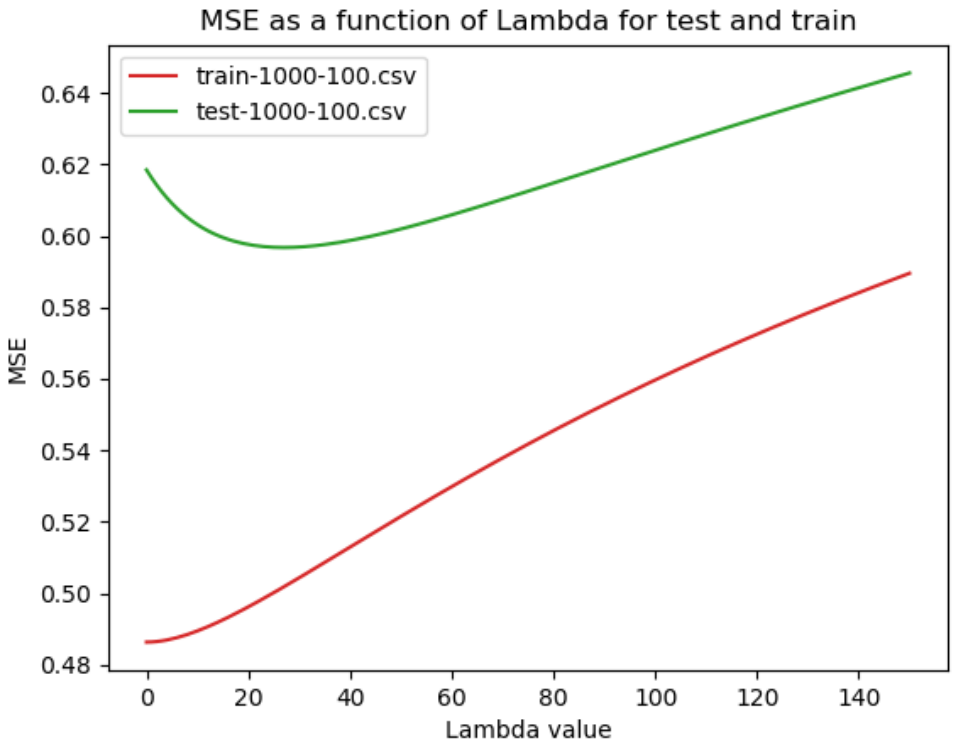
1) Train_100_10	TrainR_100_10
2) Train_100_100	TrainR_100_100
3) Train_1000_100	TrainR_1000_100
4) Train_crime	TrainR_crime
5) Train_wine	TrainR_wine

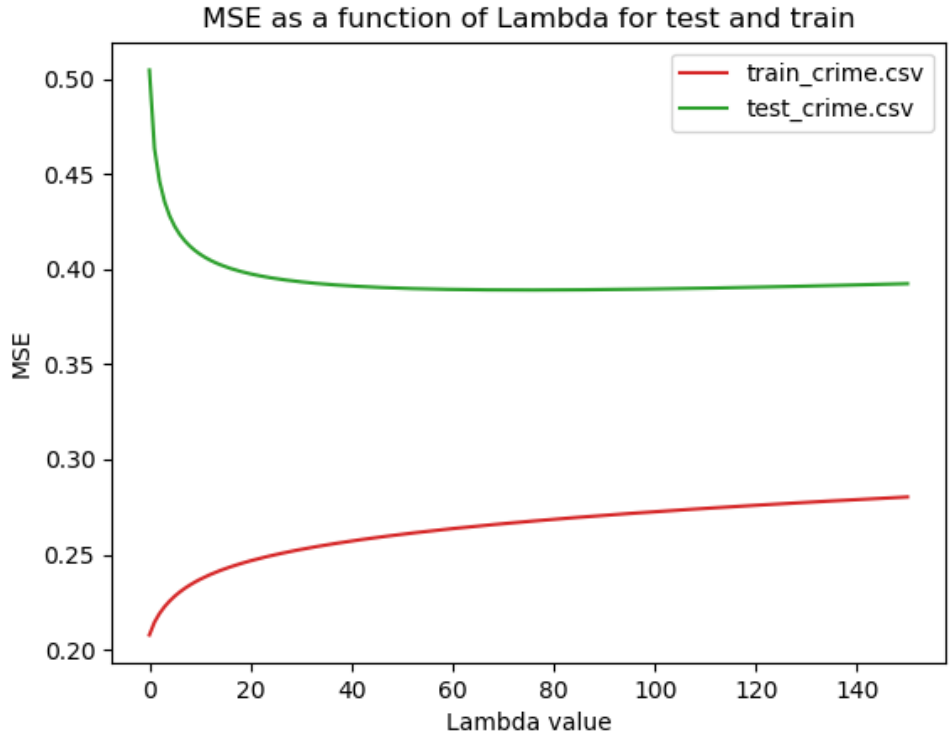
In this experiment, we start by calculating the w parameters which is given by Bishops equation in 3.28 and use this solution vector to calculate the mean square error given by equation 3.26. The solution vector is calculated over range of values of lamda from 0 to 150.

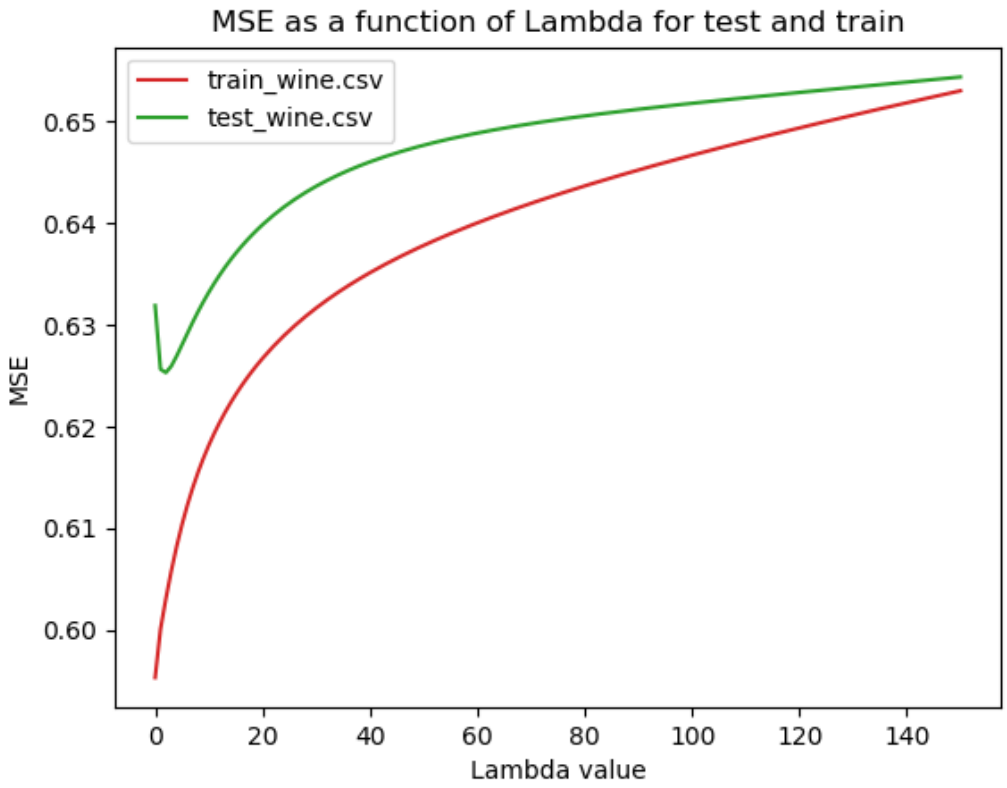
We tested this on all the datasets for lamda wary from 0 to 150 and plotted graphs as a function of MSE.


MSE as a function of Lambda for test and train
train-100-10.csv
test-100-10.csv
MSE
Lambda value
9.0
8.5
8.0
7.5
7.0
6.5
6.0
5.5
0
20
40
60
80
100
120
140


MSE as a function of Lambda for test and train
train-100-100.csv
test-100-100.csv
MSE
Lambda value
25
20
15
10
5
0
0
20
40
60
80
100
120
140


MSE as a function of Lambda for test and train
train-1000-100.csv
test-1000-100.csv
MSE
Lambda value
0.64
0.62
0.60
0.58
0.56
0.54
0.52
0.50
0.48
0
20
40
60
80
100
120
140


MSE as a function of Lambda for test and train
train_crime.csv
test_crime.csv
MSE
Lambda value
0.50
0.45
0.40
0.35
0.30
0.25
0.20
0
20
40
60
80
100
120
140



Attached it the output for best test results for part 1. We will use this to compare result sets in experiments 3.1 and 3.2

```
Task 1 in progress

Dataset -100-10:
Minimum MSE found at lamda  4  =  6.084749362987847

Dataset -100-100:
Minimum MSE found at lamda  18  =  0.7202788056527184

Dataset -1000-100:
Minimum MSE found at lamda  27  =  0.5967438457326986

Dataset - Crime:
Minimum MSE found at lamda  75  =  0.38902338771344375

Dataset - Wine:
Minimum MSE found at lamda  2  =  0.6253088423046731
Task 1 completed and graphs have been saved
```

Question:

1) Why can't the training set MSE be used to select lambda?

From the above experiment, we can clearly see that the MSE increases with the regularization parameter i.e. lambda(λ). We get the best performance ie lowest MSE when λ=0. At λ=0, there is no regularization. We can see from the above graphs, that for TEST dataset we have highest MSE at λ=0., which is because of overfitting the model with the train set. Hence, we can't depend simply on training set performance to choose λ as it would perform badly for test dataset. Thus, regularization address this problem of overfitting on training set.

2) How does λ affect error on the test set?

As we can clearly see from the above graphs, the MSE decreases initially with increase in λ, and then tends to increase after a certain value. This signifies that there exist a correct regularization parameter λ for every dataset on which we have performed the experiment. Small λ values result in the problem of overfitting whereas large value of λ overshadows the patterns in the dataset. But, there exists a regularization parameter for which the performance of the algorithm is maximum ie it has minimum MSE.

3) Does this differ for different datasets? How do you explain these variations?

Yes, this differs for the datasets. From the graph of 1000-100 and 100-100 we can see that for the same number of features, if we increase the number of training data we get a better performance of the perfect lamda. In 100-10 and 100-100, for same training size, we can see that adding more features doesn't necessarily contribute to a perfect model. We can also see that from 100-100, at λ=0 the model performs the worst without regularization.

Experiment 2 – Learning Curves

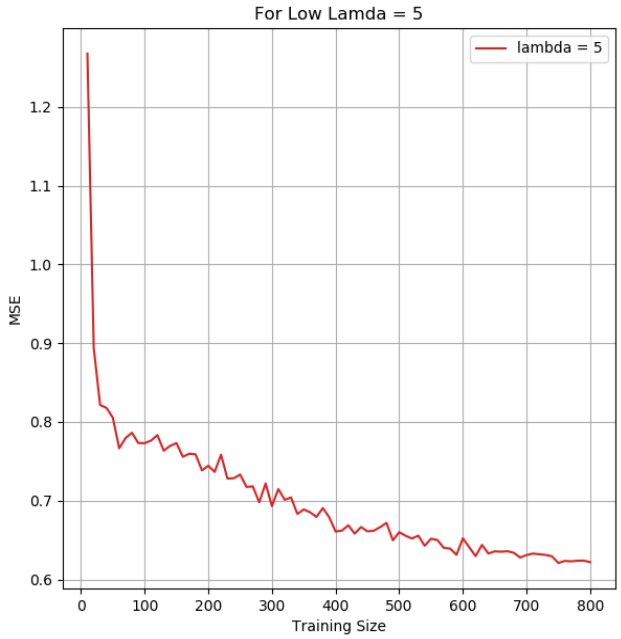
In this experiment, we pick three values of lamda as small, perfect and big and plot the learning curves as a function of MSE across various training sizes from 10-1000.

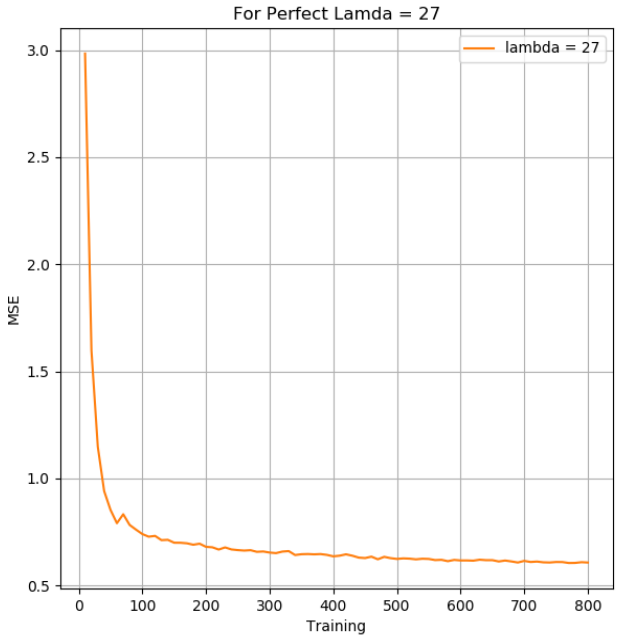
In the graph plotting, I have plotted four sublots:

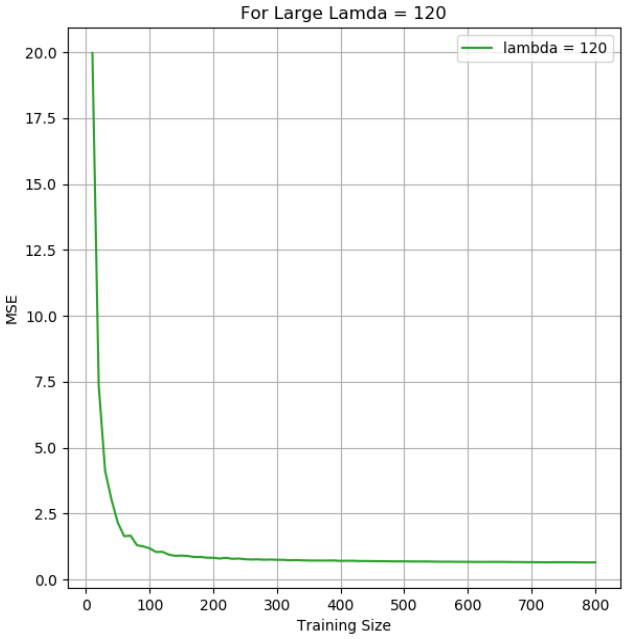
1) Graph for Lambda small = 5
2) Graph for Lambda perfect = 27

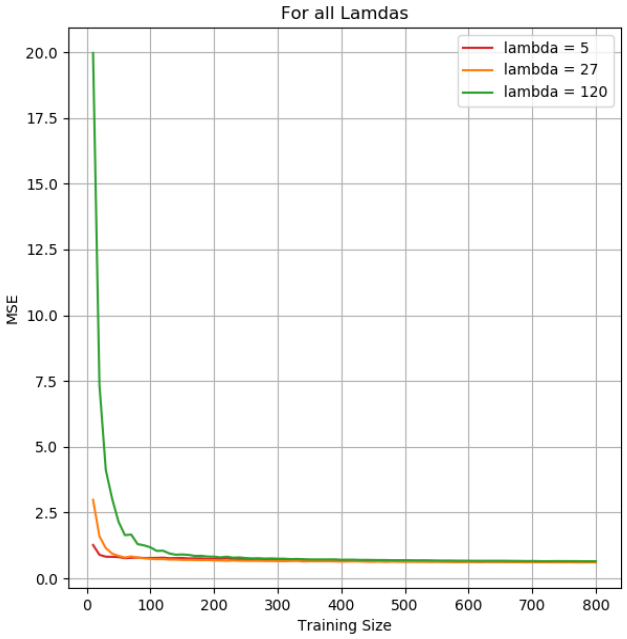
3) Graph for Lambda Large = 120
4) A common graph showing all three values for comparison

Task 2 : Learning curves









Question : What can you observe from the plots regarding the dependence of the error on λ and on the number of samples? Consider both the case of small training set sizes and large training set sizes. How do you explain these variations?

From the graph4, which has all the plotting of lambda, shows that as the training size increases the regularization parameter λ gives the same performance on the test data. The reason being as we provide more training data, the model learns to generalize better and thus overfitting reduces and MSE decreases.

The huge fluctuation is seen in the training size range of 5 to 120 which is very low training size. When the training size is less, too small and too big lambda result in large value of MSE. The perfect lambda gives the better performance i.e. low MSE independent of training size.

Experiment 3.1 Model Selection using Cross Validation

In this part we use 10 fold cross validation on the training set to pick the value of λ in the same range as 0 to 150, then retrain on the entire train set and evaluate on the test set. We follow below steps:

1. We divide the Dataset in 10 Disjoint sets.
2. And then for each value of alpha,
    a. train on all portions of i except i and and test on recording the performance on i.
    b. We then calculate the mean on the 10 folds for a lambda value.
3. We pick lambda with the best average performance ie minimum MSE

Below table depicts the value of Lambda, MSE and run time for each dataset.

	100_10	100_100	1000_100	Crime	Wine
Lambda	15	18	23	150	2
MSE	6.214	0.720	0.597	0.392	0.625
Run Time	0.362	4.00	15.10	9.34	0.861

Observations:

1. Comparing above results with best set from part 1, we can see that the MSE matches very closely except for dataset 100 10 and Crime, where

there is a slight difference in the MSE computed. This could be because of low dataset as cross validation doesn't work well in case of less data.

2. Also, we can see that there is a difference in the lambda value at which the MSE is minimum in dataset 100 10 and crime.

```
Task 3.1 10 fold cross validation for model selection started:
---------------------------------
Results for 100 10 dataset:
Lamda :  15
MSE :  6.214438800288887
Run Time : 0.3626681000000005
---------------------------------
Results for 100 100 dataset:
Lamda :  18
MSE :  0.7202788056527184
Run Time : 4.005828000000001
---------------------------------
Results for 1000 100 dataset:
Lamda :  23
MSE :  0.59700238030345
Run Time : 15.100938
---------------------------------
Results for Crime dataset:
Lamda :  150
MSE :  0.3923389920343814
Run Time : 9.3463292
---------------------------------
Results for Wine dataset:
Lamda :  2
MSE :  0.6253088423046731
Run Time : 0.8617615000000001

Part 3.1 completed
```

Experiment 3.2 Bayesian Model Selection

In this experiment, we yield an iterative algorithm for selecting α and β using the training set. We then calculate the MSE on the test set using the MAP (mn) for prediction. We start by randomly initializing alpha and beta between 1 and 10 values. Below table shows the comparison between different datasets:

Table for Comparison

	100_10	100_100	1000_100	Crime	Wine
Alpha	1.359	1.068	9.761	204.102	5.368
Beta	0.159	74.967	1.870	3.491	1.614
Lambda	8.516	0.014	5.219	58.446	3.325
MSE	6.117	5.722	0.608	0.389	0.626
Run Time	0.027	0.188	0.406	0.211	0.083

Observations:

1. Prior probability distributions are used to describe the uncertainty surrounding all unknowns. After observing the data, the posterior distribution provides a coherent post data summary of the remaining uncertainty which is relevant for model selection.
2. We can see that even though the Lambda values vary from the best test set results, the MSE is quite comparable to the best true set.

```
Task 3.2 in progress
---------------------------------
Results for 100 10 dataset:
Alpha :  1.3590563769860617
Beta :  0.1595806073429387
Lamda :  8.516425646040123
MSE :  6.117323753790675
Run Time : 0.02878269999999361
---------------------------------
Results for 100 100 dataset:
Alpha :  1.0691297577871695
Beta :  74.86317075153364
Lamda :  0.014281117765310086
MSE :  5.718312402649745
Run Time : 0.20609569999999167
---------------------------------
Results for 1000 100 dataset:
Alpha :  9.76110021129644
Beta :  1.8700741495263598
Lamda :  5.219632715509526
MSE :  0.6087527263666067
Run Time : 0.4898407999999961
---------------------------------
Results for Crime dataset:
Alpha :  204.1024937292641
Beta :  3.492106516763505
Lamda :  58.4468121890013
MSE :  0.3893753362716179
Run Time : 0.2889454000000029
---------------------------------
Results for Wine dataset:
Alpha :  5.368890456621008
Beta :  1.6145947115815773
Lamda :  3.325224849375301
MSE :  0.6262351469247622
Run Time : 0.10499109999999234
---------------------------------
Task 3.2 completed
```

Experiment 3.3 Comparison

A Good Model is not the one that gives accurate predictions on the known data or training data but the one which gives good predictions on the new data and avoids overfitting and underfitting. The challenge with overfitting is that we can't well know how well our model will perform on new data until we test it. Cross validation works well in this model selection.

Bayesian model selection is selecting the model which assigns the highest probability to the data after all parameters have been integrated out. Bayesian model selection also maximizes generalizability by trading off goodness-of-fit and model complexity.

The practical implementation of Bayesian model selection approach often requires carefully tailored priors and novel posterior calculation methods.

In terms of execution time, Bayesian model performs fast and it is cheap compared to Cross validation model. Cross validation creates fold and runs the model on same data multiple times.

In terms of accuracy, Cross validation selection works well compared to Bayesian model selection. As seen from above experiment in 3.1, we can see that the model work with great accuracy even while simply training our model on the train data and it helps in solving the issue of overfitting in less training size data.