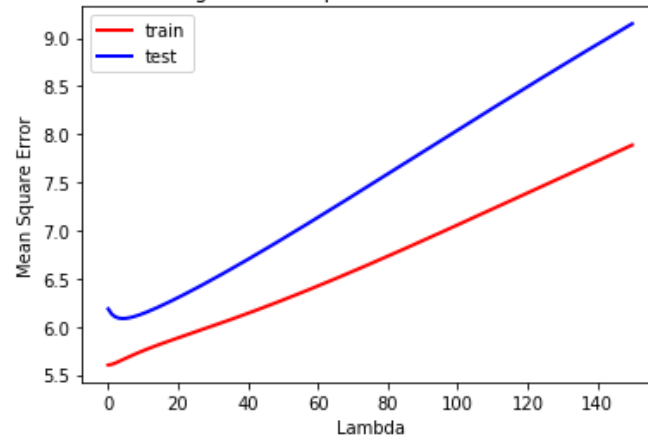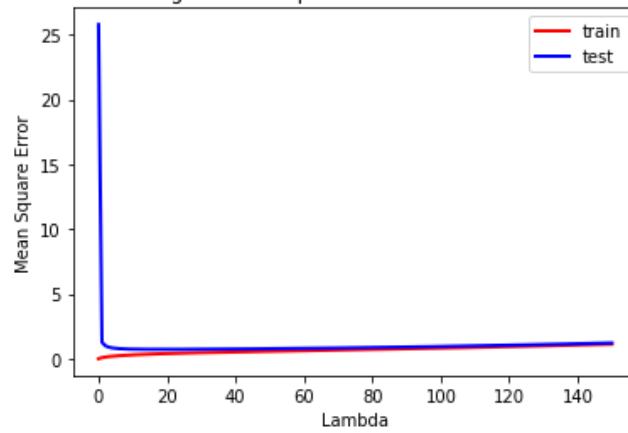# Task 1: Regularization

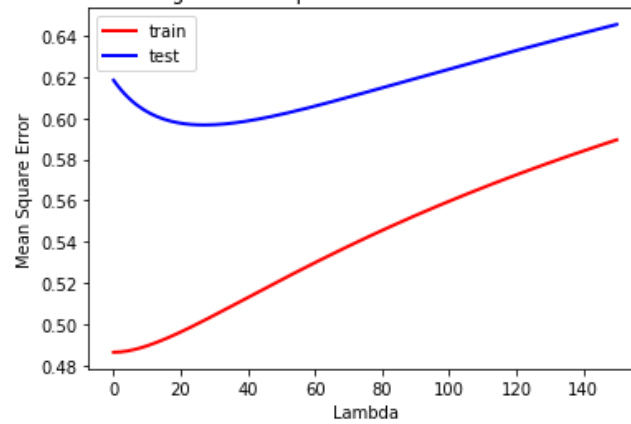Train and Test MSE as a function of the regularization parameter lambda for file train-100-10.csv and test-100-10.csv
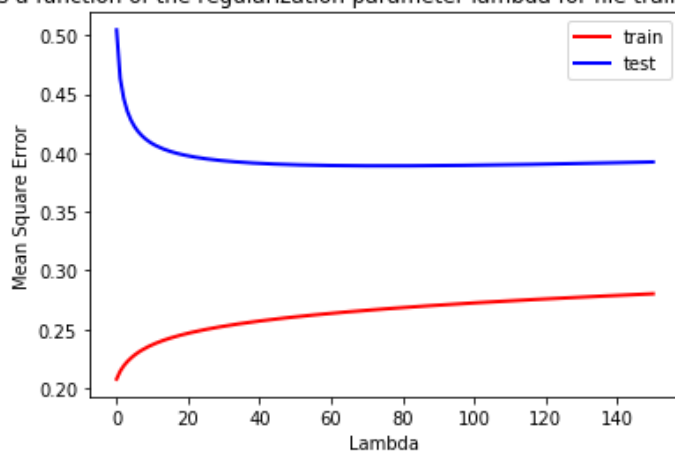

Train and Test MSE as a function of the regularization parameter lambda for file train-100-100.csv and test-100-100.csv
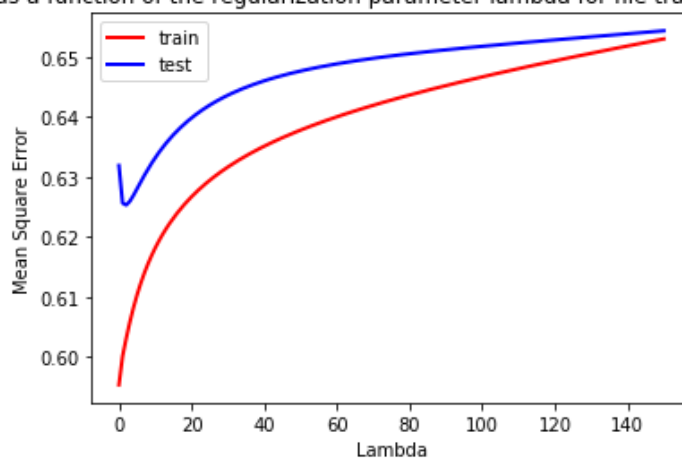

Train and Test MSE as a function of the regularization parameter lambda for file train-1000-100.csv and test-1000-100.csv

Train and Test MSE as a function of the regularization parameter lambda for file train-crime.csv and test-crime.csv

Train and Test MSE as a function of the regularization parameter lambda for file train-wine.csv and test-wine.csv
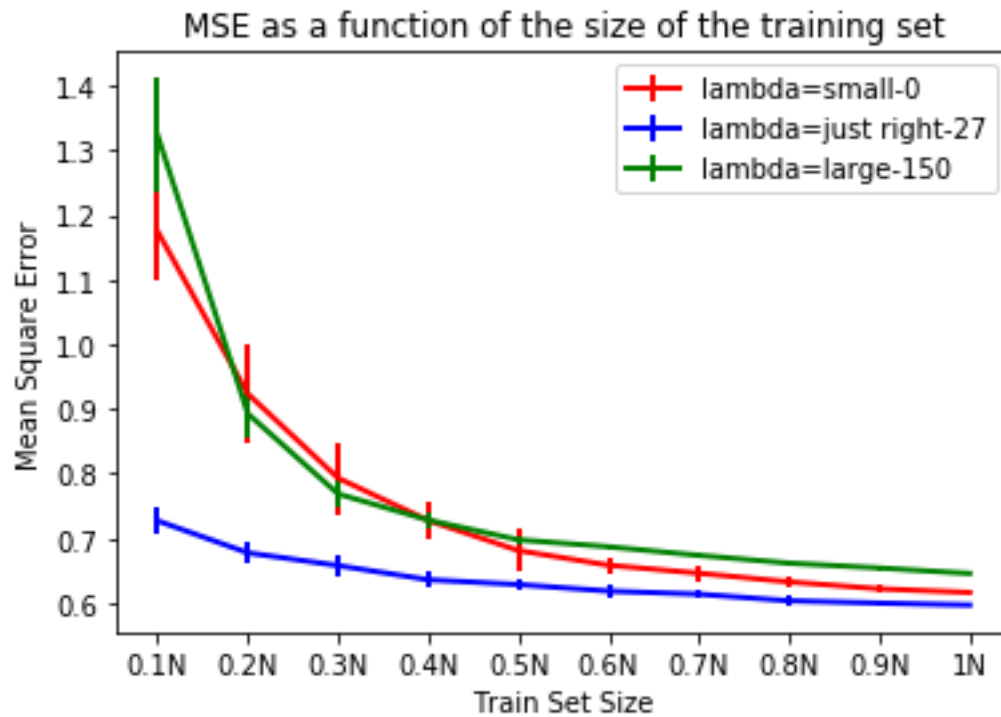

Observations:

1. MSE of the true hidden function closely matches with MSE calculated in task 1 for 3 artificial datasets.
2. Training set MSE cannot be used to select lambda since they don't represent the lambda well and It doesn't make a sense to calculate hidden parameter w with train data and calculating MSE on same train data. It won't be a good representation of lambda.
3. In most of the plots for error on test data, you will find the curves which has a global minima representing lambda which gives best accuracy/ minimum MSE out of all the lambdas. And yes, it differs for different datasets.
4. These variations in lambda can be based on number of examples, number of features present in the dataset.

Task 2: Learning Curves

If I use 10% of sample, 20% of sample….100% of random sample subsets as train set size, my graph with std as error bars looks like:

MSE as a function of the size of the training set

If I use 5% of sample, 10% of sample….100% of random sample subsets as train set size, my graph with std as error bars looks like:

MSE as a function of the size of the training set
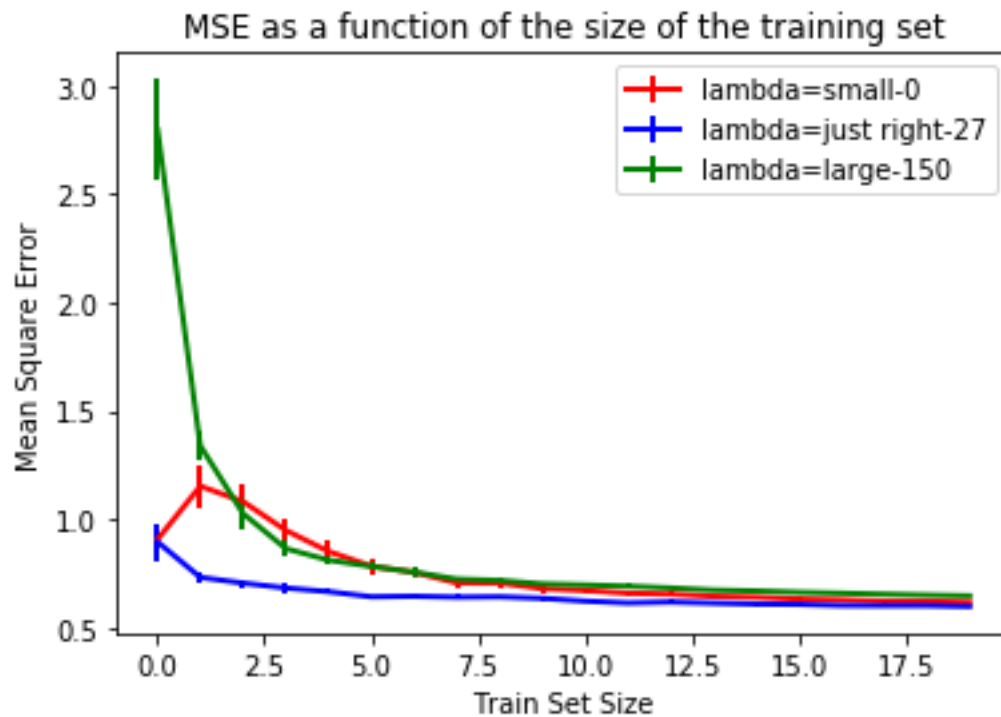
Observations:

1. For 1000-100 dataset, my "too-small" value is 1, "Just-right" is 27 and "too-large" is 150
2. From graphs, it can be said that as the training set sizes increases, accuracy is more for all the lambda.
3. "Just-right" lambda value has better accuracy overall even if train set sizes are less and it also has lowest standard deviation as error bar seem almost negligible.
4. Also, small and large lambda values have more MSE than just-right lambda value.

Task 3.1: Model Selection using Cross Validation

Dataset	Part 3.1 Lambda	Part 3.1 corresponding MSE	Part 3.1 Run-Time (In Secs)	Part 1 choice of lambda	Part 1 MSE
100-10	15	**6.214**	0.46	4	**6.0847**
100-100	18	**0.7202**	1.30	18	**0.7202**
1000-100	23	**0.597**	4.47	27	**0.5967**
Crime	149	**0.3922**	1.82	75	**0.3890**
Wine	2	**0.6253**	2.16	2	**0.6253**

Observations:

1. For some of the dataset, Lambda values for task 3.1 and task 1 are very close
2. Also, MSE values are almost the same for both the tasks
3. Advantage of cross validation is we are not seeing any test data, we are training on some part of train set and testing on one part of train set which is independent of the actual test data, also considering different test sets with each fold iteration.

Task 3.2 Bayesian Model Selection

Dataset	Alpha	Beta	Lambda	Associated MSE	Run time (In Secs)	Part 1 choice of lambda	Part 1 MSE
100-10	1.538	0.15	9.756	**6.133**	0.0285	4	**6.0847**
100-100	6.125	2.30	2.655	**0.9186**	0.7566	18	**0.7202**
1000-100	10.75	1.85	5.806	**0.6079**	0.0275	27	**0.5967**
Crime	545.38	3.16	172.55	**0.3939**	0.122	75	**0.3890**
Wine	8.570	1.597	5.364	**0.6284**	0.128	2	**0.6253**

Observations:

1. Even the lambda differs for task 1 and task 3.2, it has very close MSE.
2. The Bayesian model approach chooses the lambda after iterating multiple times using convergence conditions instead of checking all the lambdas from 1-150 like done in task 1. Hence, this is better approach than task 1.
3. The reason for so much variation in lambda for both tasks is we are calculating alpha and beta hyperparameters only using train data and test data is unknown to us.

Task 3.3: Comparison

From tables of task 3.1 and 3.2, we can conclude that time taken to run the approach is lesser in Bayesian approach than cross validation. Since, task 3.1 considers all folds which takes it longer to compute MSE and lambda. Also, Bayesian runs within a second even for large dataset.

But in case of choice of lambda and MSE (by looking at the datasets we have), model selection using cross validation (task 3.1) is better as compared to Bayesian approach.