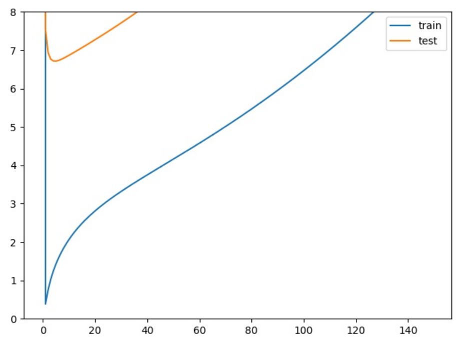
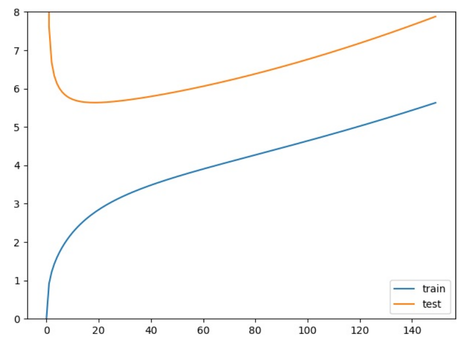
Alexa Bosworth
COMP136
PP2

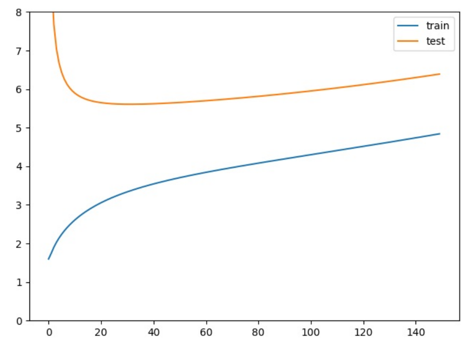
Task 1:
Plots



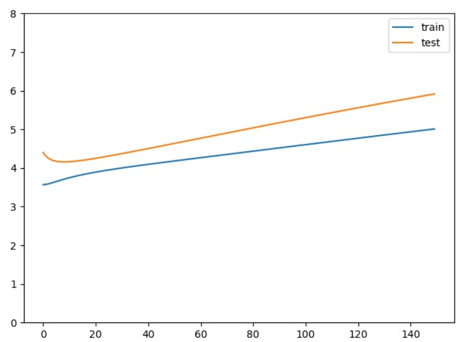
50(1000)-100



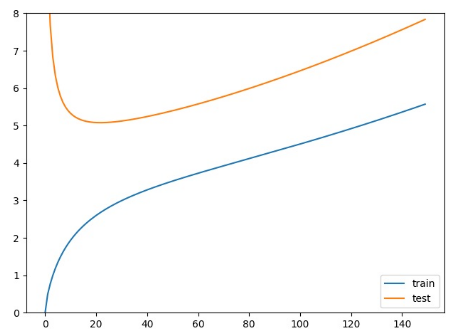
100(1000)-100



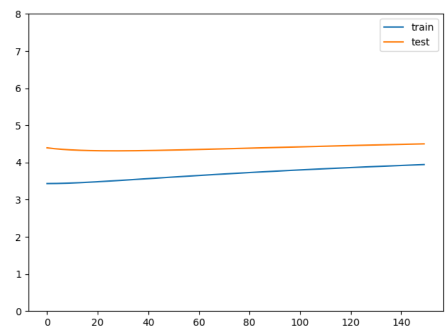
150(1000)-100



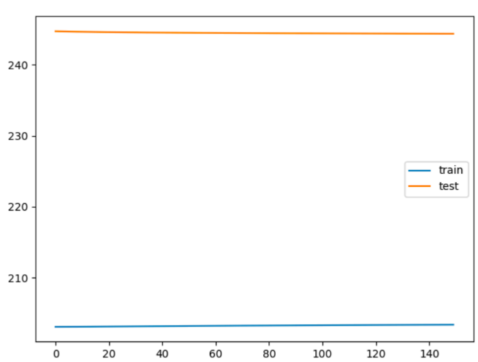
100-10



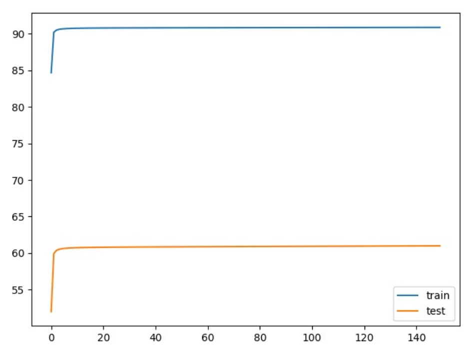
100-100



1000-100



Forest Fire



Real Estate

Data set	Lambda
50(1000)-100 train	1
50(1000)-100 test	5
100(1000)-100 train	0
100(1000)-100 test	18
150(1000)-100 train	31
150(1000)-100 test	0
100-10 train	8
100-10 test	0
100-100 train	22
100-100 test	0
1000-100 train	27
1000-100 test	0
Forest fire train	149
Forest fire test	0
Real Estate train	0
Real Estate test	0

Why can't the training set be used to select λ?
Except for the real estate dataset, there were no instances where the optimal λ found on the training set was the same as the optimal λ found for the test set. Almost every value of lambda selected for training is 0, which makes sense, in that the model has been trained on that data and any deviation from the model would increase error.

How does the choice of the optimal λ vary with the number of features and number of examples?
By observation of the datasets of sizes 50,100, and 150, we find that as the number of examples increases, so does the value of lambda. On the artificial sets, we find that where there are fewer features, the value of lambda increases

Consider both the cases where the number of features is fixed and where the number of examples is fixed. How do you explain these variations?
I would posit that as the model becomes more complex, the model has a strong tendency to overfit. Larger values of lambda , aka stronger regularization, is required for the model to be able to generalize over new data, for which it has not been trained.

Task 2:

Dataset	Lambda	MSE on test
50(1000)-100	25	7.490645485473915
100(1000)-100	26	5.664928126081487
150(1000)-100	43	5.630422965794571
100-10	17	4.220060059315382
100-100	23	5.07962657282722
1000-100	30	4.316182020126277
Forest Fire	149	244.36569812049174
Real Estate	0	52.00475424984673

How do the results compare to the best test set results from Task 1 both in terms of the choice of λ and test set MSE?
Relative to the training sets, the choice of lambda are much higher in task 2 than in task 1. However, the MSE are relatively similar.

What is the run time cost of this scheme?
Let's say that linear regression is O(n). We complete cross validation 150 (a constant) number of times for values of lambda. We perform cross validation on 10 folds as specified by the spec, calling linear regression 10 times (operations of partitioning the data set are constant). We return the best value of lambda for each file, of which there are 8. It is roughly polynomial time.

How does the quality depend on the number of examples and features?
Not being certain by what you mean by quality, I would say that if a model has many parameters, then it is necessary to have sufficient data to train those parameters, which can be observed in dataset 1000-100.

Task 3:

Dataset	MSE on test	A	B
50(1000)-100	4.275914274314603e-09	3.6460716146766967	-1590781206.2119315
100(1000)-100	0.43485652771941735	1.250096049573983	-2.6736813022621932
150(1000)-100	1.6153807057977343	3.4654536254015245	-1.7135405578845069
100-10	3.569867357838193	2.8777419629158763	0.20610780854496444
100-100	0.2030063930691643	1.616623217050976	-5.777637247081544

Dataset	MSE on test	A	B
1000-100	3.435004974157081	8.18916260172197	0.2069148709455429
Forest Fire	203.09289875298194	6.694056869973871	0.004532573227568274
Real Estate	84.65135367717956	0.0003121050504978216	0.011246694195357235

How do the results compare to the best test set results from Task 1 both in terms of the choice of λ and test set MSE?
On the artificial training sets and Forest Fire, Empirical Bayes parameter selection yielded significantly lower MSE's. However, it underperformed on the Real Estate data set. Perhaps it is not normally distributed.

What is the run time cost of this scheme?
I would guess that this method would take O(n) or O(nlogn).

Task 4:
How do the two model selection methods compare in terms of test set MSE and in terms of run time?
Cross validation, being a brute force solution, has a worse complexity than empirical bayes. Generally though, empirical bayes yielded smaller MSE's.

What are the important factors affecting performance for each method?
Empirical Bayes can get stuck in local minima so its important to choose enough initial values for a and b. It's also important that the data is normally distributed. Cross-validation uses the training set partitioned into validation to search for the optimal lambda. Because it doesn't use test data, it may underestimate the value of lambda needed to mitigate high model variance.