Part 1.1 - Feature Engineering with Feature Subsets (10 points)

1.1.1 Which model had the best RMSE on the *training data*? (1 point)

Solution Below is the RMSE achieved on trained data
From this

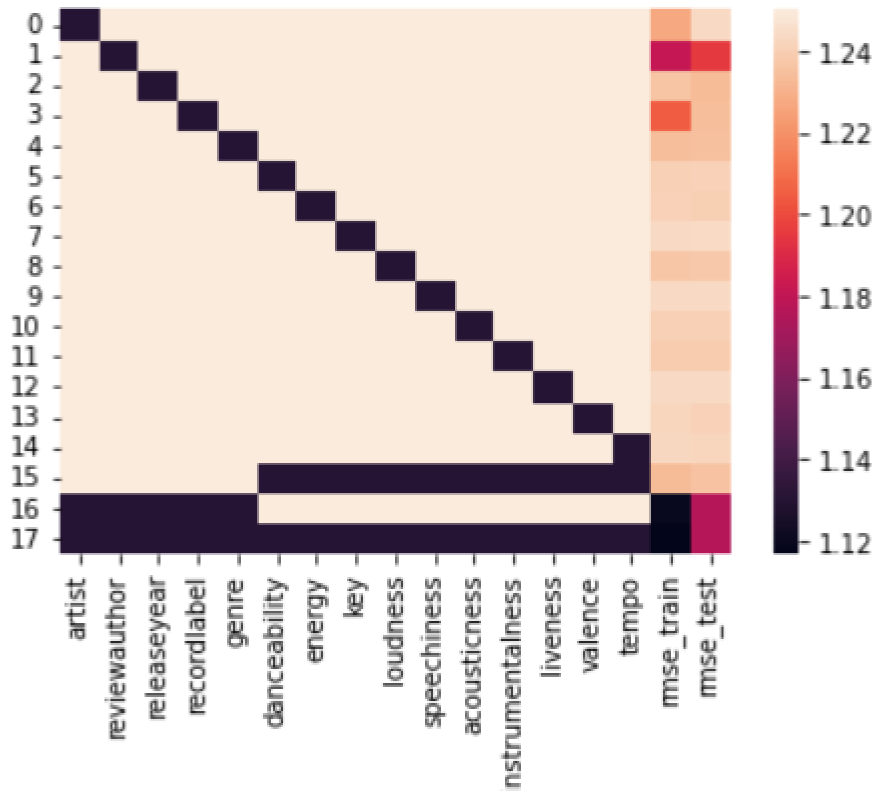
	rmse_train
17	1.116957
16	1.119865
1	1.181273

The model with all features had the best RMSE.
1.1.2 Which model had the best RMSE on the *test data*? (1 point)

	rmse_test
16	1.175888
17	1.176070
1	1.195304

Solution: Looking at this and from our heatmap we can see that model which had the most trainging features had the best rmse on test data
1.1.3 Which feature do you believe was the most important one? Why? *(Note: There is more than one perfectly acceptable way to answer this question)* (2 points)

Solution



Looking at the heatmap ->
The heatmap indicates the presence of features (in black for all the columns except last 2), and test and train rmse in the last 2 columns.
Based on the above Heatmap we can say that presence of '**reviewauthor'** resulted in much lower rmse comparatively. So, '**reviewauthor'** seems to be most import feature

1.1.4 What can we say about the utility of the Spotify features based on these results? (1 point)
Solution
As seen in the heatmap that the addition of Spotify features does not improve results significantly, and thus are not useful for the linear regression model.

Part 1.2 - Feature Engineering with the LASSO (15 points)

1.2.1 - How many new features are introduced by Step 2 above? Provide both the number and an explanation of how you got to this number. (2 points)

Solution We have Initial features = 15

Final features = 691

Features introduced in step 2 = 676

We approached the below mathematics to achieve the result

X_test.shape[1] - len(features)

Here features are the initial features

And X_test is the hstack array is obtained by doing manipulations with the test data and finally doing standard scaling

1.2.2 - What was the best alpha value according to your cross-validation results? (5 points)

The best alpha value is

0.00016857012552111163

1.2.3 - What was the **average RMSE** of the model with this alpha value on the k-fold cross validation on the *training* data? (3 points)

Solution:

1.1229212879739903

1.2.4 What was the **RMSE** of the model with this alpha value on the k-fold cross validation on the *test* data?

1.1649797995719675

Part 1.3 - Interpreting Model Coefficents (15 points)

In this section we will interpret the coefficients from the final model you trained on all of the training data.

1.3.1- How many non-zero coefficients are in this final model? (5 points)
Solution: 514

1.3.2-What percentage of the coefficients are non-zero in this final model? (1 point)
Solution
0.7004341534008683 that's approx. 70 %

1.3.3 - Who were the three most critical review authors, as estimated by the model? How do you know? (3 points)
Solution

	Col_name	Parent_column	Coeff	abs_Coeffs
123	Alison Fields	reviewauthor	-1.528591	1.528591
360	kris ex	reviewauthor	1.136912	1.136912
188	Evan McGarvey	reviewauthor	-1.123692	1.123692

From the above Alison Fields ,kris ex and Evan McGarvey are most critical.
We sort the results by using using abs_coeff of the parent column that's reviewauthor and sorted the result in ascending order .Hence doing this we were able to gain insights of the most critical review of authors

1.3.4- Who were the three artists that reviewers tended to like the most? How do you know? (3 points)
Solution

artist	score
The Beatles	9.075000
Miles Davis	8.984615
Prince	8.300000

We did the group by using artist of the training data and then calculated the score.mean () of that and finally sorted the results to to find the artists

1.3.5 - What genre did Pitchfork reviewers tend to like the most? Which genre did they like the least? (3 points)

Solution

Reviewers Liked most genre ->Jazz 7.601172

Reviewers Liked least genre - >Rap 6.949527

Part 1.4 - "Manual" Cross-Validation + Holdout for Model Selection and Evaluation (25 points)

Write out the result to a file called part_1.4_results.csv and submit this along with your assignment. (10 points) (You do not need to submit anything for your report for this part.)
Solution:
Below is the result analysis .We will use this to answer the following questions

model_name	hyperparameter_setting	mean_training_rmse	sd_training_rmse	test_rmse
DTR	_criterion_squared_error_max_depth_5_	1.45899345371421	0.060824573181645	1.20210762252857
KNN	_n_neighbors_15_	1.45558396230281	0.049360883546208	1.21235031615682
Ridge	_alpha_10.0_	1.34666670384435	0.062091601767111	1.16178567849332

s

1.4.1 Report, for each model, the hyper parameter setting that resulted in the best performance (3 points)
Solution
Below is the hyper paramter setting that resulted in the best performance

model_name	hyperparameter_setting
DTR	_criterion_squared_error_max_depth_5_
KNN	_n_neighbors_15_
Ridge	_alpha_10.0_

1.4.2 Which model performed the best overall on the cross-validation? (3 points)
Solution: Ridge Regression performed the best among the three.

1.4.3 Which model performed the best overall on the final test set? (3 points)
Solution: Ridge Regression performed the best with accuracy

1.4.4 With respect to your answer for 1.4.3, why do you think that might be? (*Note: there is more than one correct way to answer this question*) (1 point)
Solution: Because Ridge is relatively simpler, and it is also regularized.

1.4.5 Which model/hyperparameter setting had the highest standard deviation across the different folds of the cross validation? (3 points)

Solution:

sd_training_rmse
0.060825
0.049361
0.062092

Ridge had the highest standard deviation with 0.062092.

1.4.6 With respect to your answer for 1.4.5, why do you think that might be? (*Note: there is more than one correct way to answer this question*) (2 points)
Solution: The reason for higher standard deviation is that on different folds of cross validation gave sometimes high or low RMSE value. Leading to higher variation.

Part 2.1 - Logistic Regression with Gradient Descent (25 points)

2.1.1 - How did you go about selecting a good step size, i.e. one that was not too big or too small? (*Note: There is more than one correct answer to this*) (2 points)

We selected the step size value for final result as small. Below is our analysis
->
We started with the higher value of step size, but we found that algorithm was not converging rather oscillating hence finally playing with the values we choose the step size small which results in a step towards the minimum, hence resulting in convergence of the algorithm

2.1.2 - What is the condition under which we assume that the gradient descent algorithm has converged in the code here? (2 points)
The condition we are using right now for convergence is that if the change in our coefficients values is not more than 0.00001. Then we assume that our algorithm has converged.

2.1.3 - What is a different convergence metric we could have used? (*Note: There is more than one correct answer to this*) (1 points)

We could have used a few other convergence metrics as well. The following are a few of them:

If we set the maximum number of iterations, the system will not go into an infinite loop if the algorithm is not converging or is converging excessively slowly.
We can establish an acceptable mean squared error and calculate it in each iteration. When a weight vector produces an acceptable MSE, we can stop iterating and presume convergence.

Part 2.2 - 574 Only - Logistic Regression with Newton- Raphson (10 points)

For Part 2.2, correct results for both parts will be worth 5 points each.