Assignment 2, CSE 474/574

Speaking potatoes

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

• (Nothing for the report, listing here to provide point value) Write out the result to a file called part_1.1_results.csv and submit this along with your assignment. (5 points)
• **1.1.1** Which model had the best RMSE on the *training data*? (1 point) **['artist', 'reviewauthor', 'releaseyear', 'recordlabel', 'genre', 'danceability', 'energy', 'key', 'loudness', 'speechiness', 'acousticness', 'instrumentalness', 'liveness', 'valence', 'tempo']**
• **1.1.2** Which model had the best RMSE on the *test data*? (1 point) **same as 1.1.1**
• **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) **release year, because it has the lowest rmse both in training and test among single features**
• **1.1.4** What can we say about the utility of the Spotify features based on these results? (1 point) **the more features, the more predictable. Spotify features could help predict the scores correctly**

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. (2points) **16 features are introduced by Step2. How we got this number was first the number of features became double throgh drop='if_binary', and secondly the number of features became half because of sparse=False. As a result, the number of features didn't change from before.**
• **1.2.2** - What was the best alpha value according to your cross-validation results? (5 points) **The best alpha value according to our cross-validation results was 0.004392**
• **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) **The average RMSE of the model 0.004392 on the k-fold cross validation on the training data is 1.2256783882283901.**
• **1.2.4** - What was the **RMSE** of the model with this alpha value on the k-fold cross validation on the *test* data? (5 points) **1.2242650520928589**

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) **13**
• **1.3.2** - What percentage of the coefficients are non-zero in this final model? (1 point) **86.66666666666667**
• **1.3.3** - Who were the three most critical review authors, as estimated by the model? How do you know? (3 points) **Allison Hussey Evan Minsker Alphonse Pierre, We wrote the code for this**
• **1.3.4** - Who were the three artists that reviewers tended to like the most? How do you know? (3 points) **Brian Eno OTHER, We wrote the code for this**
• **1.3.5** - What genre did Pitchfork reviewers tend to like the most? Which genre did they like the least? (3 points) **Rap, the most. Jazz, the least**

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.)
• **1.4.1** Report, for each model, the hyper parameter setting that resulted in the best performance (3 points) **DTR-squared_error_5,Ridge-10000.0,KNN-15**

• **1.4.2** Which model performed the best overall on the cross-validation? (3points) **KNN**
• **1.4.3** Which model performed the best overall on the final test set? (3points) **DTR**
• **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*) (1point) **If test_rmse is the lowest, that means the model has the best performance**
• **1.4.5** Which model/hyperparameter setting had the highest standard deviation across the different folds of the cross validation? (3 points) **Ridge/100000.0**
• **1.4.6** With respect to your answer for 1.4.6, why do you think that might be? (*Note: there is more than one correct way to answer this question*) (2points) **If hyperparameter value is too high, it has the danger of underfitting. The hyperparmeter of Ridge for 1.4.5 was too high. So it yields highest standard deviation**

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

We will test each of the three functions you implemented; logistic_objective, logistic_gradient, and run_gradient_descent. Correct results for these will receive 5 points, 5 points, and 10 points, respectively. Partial credit will not be awarded at the time of grading (but the tests we run will be very similar to the one in the project, and it is very hard to get correct results without implementing these correctly!), but Kenny reserves the right to change this to allow for partial credit after initial grading. You do not need to submit anything for your report for this part.

• **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 can build learning curve.**
• **2.1.2** - What is the condition under which we assume that the gradient descent algorithm has converged in the code here? (2 points)
• **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)