Data analysis Project 2:

Correlation and Regression of Movie Ratings Data

Doma Ghale

Introduction

The goal of this project is to find the correlation coefficients of 1097 users who rated 400 different movies and answered 77 behavior questions, and use three types predictive models Ordinary Least Square (OLS) linear regression, Ridge regression and Lasso regression to predict the user's answers to the behavior questions (dependent variables) using their movie ratings (independent variables). For the correlation part, we found the most correlated user of each user, the pair of the most correlated users and their coefficient, and the most correlated user of the first 10 users. For the prediction part, we found error rates of the training and testing datasets for OLS linear regression and found the best alpha for ridge and lasso regression. We also performed some data cleaning and data transformation before producing the results.

Data Methodology and Results

We were provided with a csv file which contained 400 columns of movie ratings and 77 columns of answers to behavioral questions. We used the Python programming language and its built-in libraries such as Pandas and Scikit-learn to analyze and obtain the results.

The first step in data processing was to introduce consistency and find outliers. For the two questions 'Are you an only child? (1: Yes; 0: No; -1: Did not respond)' and 'Movies are best enjoyed alone (1: Yes; 0: No; -1: Did not respond)', -1 has the same meaning as nan, so we replaced -1 with nan. We found that the user 896 didn't have any of the movie ratings and also the only answers to their behavior questions were -1, which means that the user did not respond. Therefore, we removed that user from the dataset. For cases where there are some responses to the questions we fill the missing values with median. We used median instead of mean because ratings are ordinal categorical variables, and also because median is less affected by outliers. We did not normalize the data because the independent variables are on the same scale, and even though the dependent variables are on different scales, we will be building different models independent of other dependent variables. Results that were too long to include in this report can be found in the documentation of the code [1].

Correlation

We calculated the correlation coefficients of the users based on their movie ratings only and did not include the behavior questions because it doesn't make sense to include dependent variables in calculating the correlation between users. Before calculating the correlation matrix of the independent variables, we replaced the missing values with the median value of its features (how other users rated that movie). We decided not to replace the missing value with the median value of that user’s ratings of other movies because it is more reasonable to assume that a movie rating would be closer to the average rating of that movie than to assume a user would rate all movies similarly on average. For example, we found that the user 235 rated 37 out of 400 movies and all of them as “4”. The user might have selectively watched those movies; hence all of them are rated as such. But, expecting the same rating of “4” for all missing movie ratings of that user would have been an overestimation in this case.

After filling in the missing values, we transposed the data frame such that each column represents data for a user. Then, we calculated the correlation coefficients which resulted in a square matrix of size 1096 (minus 1 because of the outlier being removed). The diagonal values are all 1s because the users are perfectly correlated with themselves, so we replaced those values with nan since we need to find the most correlated users of each user. The correlation coefficients are centered at 0, so we also took the absolute value of the matrix. We created a data frame where the first column contained the list of all 1096 users, the second column contained the highest correlation coefficient of that user, and the third column contained the corresponding user. In order to make our results comparable to those who did not remove user 896, we added 1 to all users and their corresponding highest correlated users if they are above 895.The pair of the most correlated users in the data were (239, 831) and their correlation coefficient was 0.9837. The most correlated users for the first 10 users were (0, 118), (1, 831), (2, 831). (3, 704), (4, 784), (5, 990), (6, 1071), (7, 1074), (8, 821) and (9, 1004).

Regression Analysis

We used three different linear regression models to predict the answers for the behavior questions using the movie ratings, and the models were Ordinary Least Squares linear regression, Ridge regression and Lasso regression. At first, we split the data into training and testing sets (0.80 to 0.20 ratio); otherwise our error rates would be an overestimation for the unseen data. We do not think it is a good idea to replace the missing values of dependent variables, especially on the test set, unless we have very limited data. We modeled the data both with and without replacing the missing data of dependent variables and noticed that the model performs better if we replace them [1][Appendix]. We would like to know if that is true in general.

Ordinary Least Squares regression fits a linear model to minimize the residual sum of squares between the actual values of the dependent variables and the predicted values of the dependent variables [2]. Without replacing the missing values of dependent variables, the average mean squared error for 77 models in the training set was 0.59 and in the testing set was 3.36 and with replacing the missing values of dependent variables by median, the average mean squared error in the training set was 0.59 and in the testing set was 3.31.

Ridge regression imposes a penalty on the size of the coefficients with L2 regularization [3]. For ridge regression we used the hyperparameters: alphas = [0.0, 1e-8, 1e-5, 0.1, 1, 10]. Best choice for alpha was 10, whether we replace the missing values of dependent variables by median or not.

Lasso regression imposes a penalty on the size of the coefficients with L1 regularization [4]. For lasso regression we used the hyperparameters: alphas = [1e-3, 1e-2, 1e-1, 1]. Best choice for alpha is 0.1, whether we replace the missing values of dependent variables by median or not.

References

[1] Documentation of code in 2021_fall_dsga1001_proj02_dg2491.ipynb

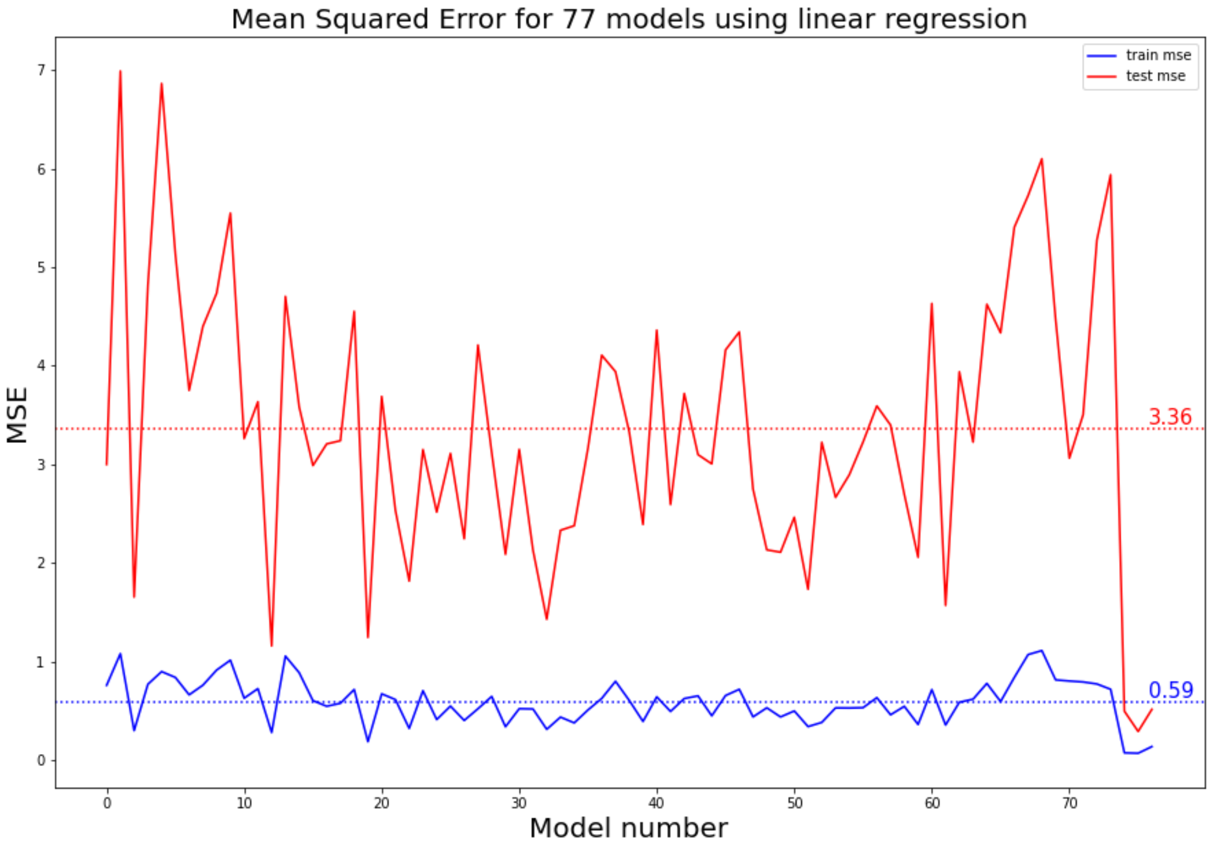
[2]https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

[3]https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html#sklearn.linear_model.Ridge

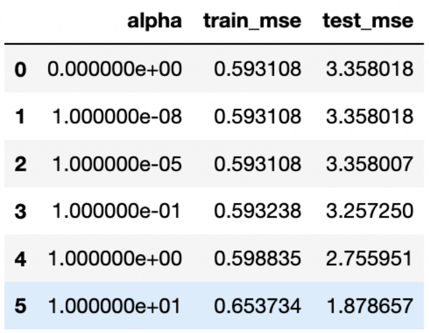
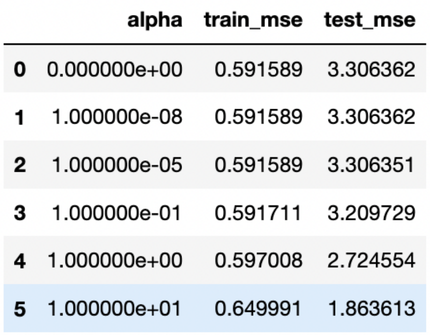
[4]https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html#sklearn.linear_model.Lasso

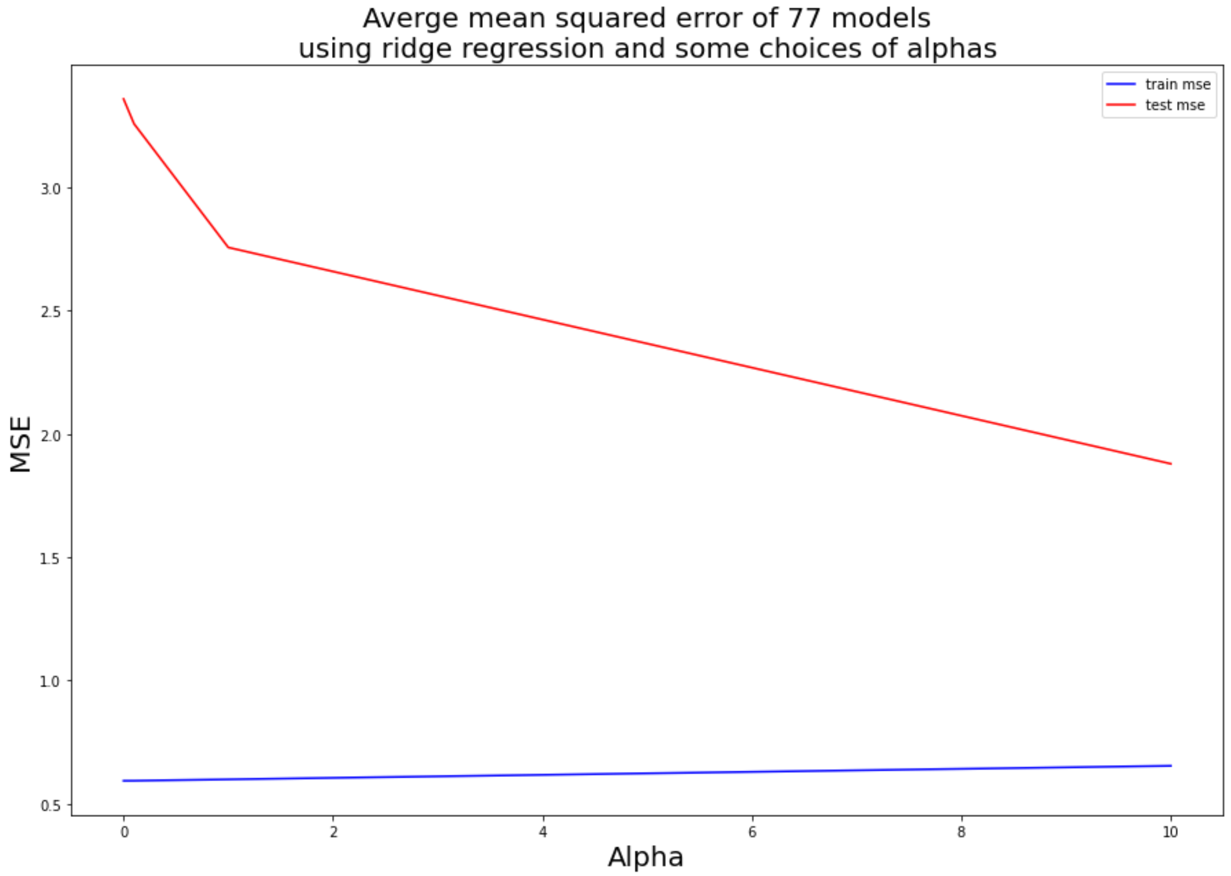
Appendix

1. Ordinary Least Squares linear regression

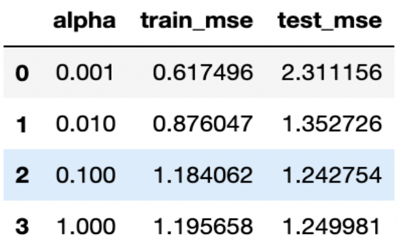
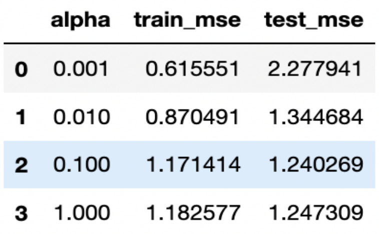


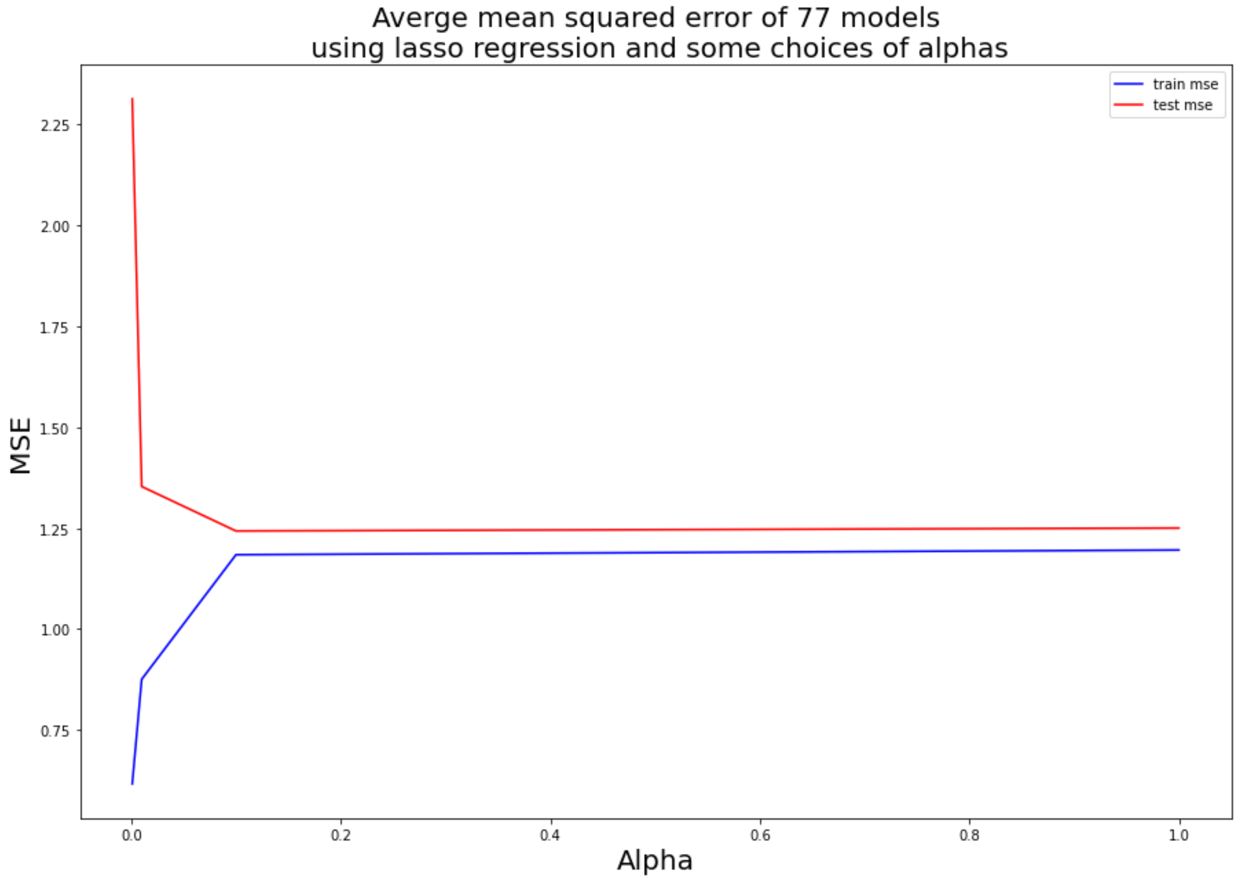
2. Ridge regression

<table>
<tr><th>missing values not replaced</th><th>missing values replaced</th></tr>
<tr><td>Best choice for alpha is 10.</td><td>Best choice for alpha is 10.</td></tr>
<tr><td>
<table>
<tr><th></th><th>alpha</th><th>train_mse</th><th>test_mse</th></tr>
<tr><td>0</td><td>0.000000e+00</td><td>0.593108</td><td>3.358018</td></tr>
<tr><td>1</td><td>1.000000e-08</td><td>0.593108</td><td>3.358018</td></tr>
<tr><td>2</td><td>1.000000e-05</td><td>0.593108</td><td>3.358007</td></tr>
<tr><td>3</td><td>1.000000e-01</td><td>0.593238</td><td>3.257250</td></tr>
<tr><td>4</td><td>1.000000e+00</td><td>0.598835</td><td>2.755951</td></tr>
<tr><td>5</td><td>1.000000e+01</td><td>0.653734</td><td>1.878657</td></tr>
</table></td><td>
<table>
<tr><th></th><th>alpha</th><th>train_mse</th><th>test_mse</th></tr>
<tr><td>0</td><td>0.000000e+00</td><td>0.591589</td><td>3.306362</td></tr>
<tr><td>1</td><td>1.000000e-08</td><td>0.591589</td><td>3.306362</td></tr>
<tr><td>2</td><td>1.000000e-05</td><td>0.591589</td><td>3.306351</td></tr>
<tr><td>3</td><td>1.000000e-01</td><td>0.591711</td><td>3.209729</td></tr>
<tr><td>4</td><td>1.000000e+00</td><td>0.597008</td><td>2.724554</td></tr>
<tr><td>5</td><td>1.000000e+01</td><td>0.649991</td><td>1.863613</td></tr>
</table></td></tr>
</table>



2. Lasso regression

<table>
<tr><th>missing values not replaced</th><th>missing values replaced</th></tr>
<tr><td>Best choice for alpha is 0.1.</td><td>Best choice for alpha is 0.1.</td></tr>
<tr><td>
<table>
<tr><th></th><th>alpha</th><th>train_mse</th><th>test_mse</th></tr>
<tr><td>0</td><td>0.001</td><td>0.617496</td><td>2.311156</td></tr>
<tr><td>1</td><td>0.010</td><td>0.876047</td><td>1.352726</td></tr>
<tr><td>2</td><td>0.100</td><td>1.184062</td><td>1.242754</td></tr>
<tr><td>3</td><td>1.000</td><td>1.195658</td><td>1.249981</td></tr>
</table></td><td>
<table>
<tr><th></th><th>alpha</th><th>train_mse</th><th>test_mse</th></tr>
<tr><td>0</td><td>0.001</td><td>0.615551</td><td>2.277941</td></tr>
<tr><td>1</td><td>0.010</td><td>0.870491</td><td>1.344684</td></tr>
<tr><td>2</td><td>0.100</td><td>1.171414</td><td>1.240269</td></tr>
<tr><td>3</td><td>1.000</td><td>1.182577</td><td>1.247309</td></tr>
</table></td></tr>
</table>


Averge mean squared error of 77 models
using lasso regression and some choices of alphas
train mse
test mse
MSE
Alpha