Statistical Machine Learning Homework 1

Keitaro Ogawa

I. Data Cleaning

We are given Communities and Crime dataset and our goal is to the violent crime rate based upon features of a community.

The data (X) came with a mix of numerical and object columns with some columns having Na values. My approaches of data cleaning is as follows: First, I found columns with '?' values and checked their data types. Since they were all 'object' data types and contained over 1000 Na values, except 'OtherPerCap', which only had just one Na value. So, I removed all columns with '?' values, while keeping 'OtherPerCap' column. I imputed the '?' in 'OtherPerCap'with its median value. Next, I checked the data types of the rest of remaining columns in the original data set. In here, 'state', 'fold', and 'communityname' were non-float data types, so I dropped those three columns.

Now, the updated data set (X) had contained 99 columns with zero Na values and all features were float data types, which matched with the data type of y.

In addition, I conducted feature filtering on the updated X data set for Best Subsets and StepWise selection. This is because the updated X still had too many features and it was computationally challenging to run approaches like Best Subset and StepWise selections on the whole data set. My approach was to perform univariate feature filtering. Specifically, I discovered the correlation between all predictors (xi,y) for all i, then filtered through. I set k=0.5 because it shrinks variables to a manageable number, while keeping variables with higher correlations to y. As a result, the new X had 15 features. I used this new X for running Best Subset and StepWise.

II. Question 1

A. (a) (i) Important Features

Listed below are the essential features which were selected by five approaches.

- **Statistical significance via Least Squares:** I selected features with p-value less than 0.05. ['const', 'racepctblack', 'pctUrban', 'pctWWage', 'pctWFarmSelf', 'pctWInvInc', 'pctWRetire', 'whitePerCap', 'HispPerCap', 'PctPopUnderPov', 'PctEmploy', 'PctEmplManu', 'MalePctNevMarr', 'PctKids2Par', 'PctWorkMom', 'PctIlleg', 'PctNotSpeakEnglWell', 'PersPerOccupHous', 'PersPerRentOccHous', 'PctPersDenseHous', 'HousVacant', 'PctVacantBoarded', 'PctVacMore6Mos', 'RentLowQ', 'MedRent', 'MedOwnCostPctIncNoMtg', 'NumInShelters', 'NumStreet']
- **Best Subsets:** ['racePctWhite', 'PctPopUnderPov', 'MalePctDivorce', 'FemalePctDiv', 'TotalPctDiv', 'PctFam2Par', 'PctKids2Par', 'PctIlleg']
- **Step-wise approaches (Forward):** ['racepctblack' 'racePctWhite' 'racePctHisp' 'agePct12t21' 'agePct12t29', 'numbUrban' 'pctUrban' 'pctWWage' 'pctWFarmSelf' 'pctWInvInc', 'pctWSocSec' 'pctWRetire' 'perCapInc' 'whitePerCap' 'blackPerCap' 'indianPerCap' 'AsianPerCap' 'HispPerCap' 'PctPopUnderPov' ''PctLess9thGrade' 'PctBSorMore' 'PctEmploy' 'PctEmplManu' 'PctOccupManu"MalePctDivorce' 'MalePctNevMarr' 'TotalPctDiv' 'PctKids2Par' 'PctWorkMom' 'PctIlleg' 'NumImmig' 'PctSpeakEnglOnly' 'PctNotSpeakEnglWell' 'PctLargHouseFam' 'PersPerOccupHous' 'PersPerOwnOccHous' 'PersPerRentOccHous' 'PctPersOwnOccup' 'PctPersDenseHous' 'PctHousLess3BR' 'HousVacant' 'PctHousOccup' 'PctVacantBoarded' 'PctVacMore6Mos' 'RentLowQ' 'RentHighQ' 'MedRent' 'MedRentPctHousInc' 'MedOwnCostPctInc' 'MedOwnCostPctIncNoMtg"NumInShelters' 'NumStreet' 'PctForeignBorn' 'PctUsePubTrans']
- **Step-wise approaches (Backward):** ['racepctblack' 'racePctHisp' 'agePct12t29' 'pctUrban' 'pctWWage' 'pctWFarmSelf' 'pctWInvInc' 'pctWSocSec' 'pctWRetire' 'medFamInc' 'whitePerCap' 'indianPerCap' 'HispPerCap' 'PctPopUnderPov' 'PctLess9thGrade' 'PctEmploy' 'PctEmplManu' 'PctOccupManu"PctOccupMgmtProf' 'MalePctDivorce' 'MalePctNevMarr' 'TotalPctDiv' 'PctKids2Par' 'PctWorkMom' 'NumIlleg' 'PctIlleg' 'NumImmig' 'PctNotSpeakEnglWell' 'PctLargHouseOccup' 'PersPerOccupHous' 'PersPerRentOccHous' 'PctPersOwnOccup' 'PctPersDenseHous' 'PctHousLess3BR' 'MedNumBR' 'HousVacant' 'PctHousOccup' 'PctHousOwnOcc' 'PctVacantBoarded' 'PctVacMore6Mos' 'OwnOccLowQuart' 'OwnOccMedVal' 'RentLowQ' 'RentHighQ' 'MedRent' 'MedRentPctHousInc' 'MedOwnCostPctInc' 'MedOwnCostPctIncNoMtg' 'NumInShelters' 'NumStreet' 'PctForeignBorn' 'PctUsePubTrans' 'LemasPctOfficDrugUn']
- **Lasso:** ['racepctblack', 'pctUrban', 'pctWPubAsst', 'TotalPctDiv', 'PctIlleg', 'PctPersDenseHous']
- **Elastic Net:** Number of times the tweet has been retweeted year.

B. (ii) Regularization Paths

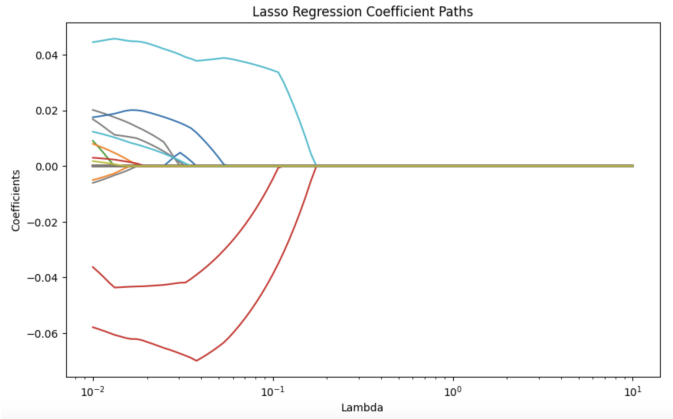

Fig. 1. "Lasso Regularization Path"

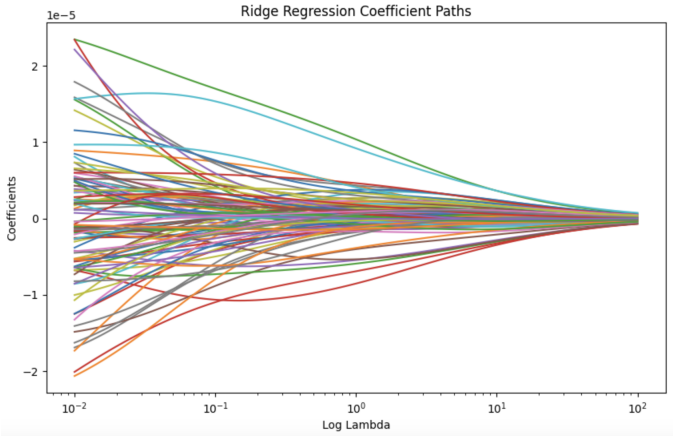

Fig. 2. "Ridge Regularization Path"

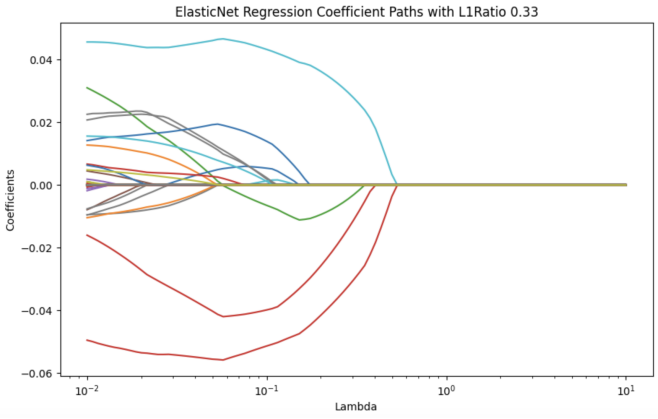

Fig. 3. "Elastic Net at alpha=0.33"

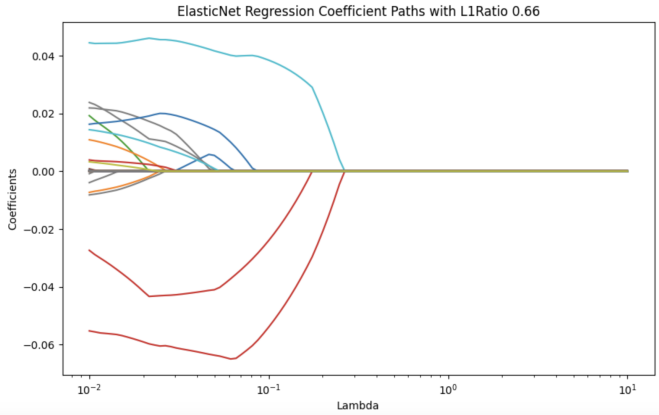

Fig. 4. "Elastic Net at alpha=0.66"

C. (iii) Results Reflection

We see a wide range of features selected for each method. Ordinary Least Squares regression includes a broader set of variables, as it does not perform any selection and instead attempts to fit all available predictors. Best subset selection, in contrast, picks a smaller set of features that provide the best fit according to a specific criterion, which explains why it includes only a few key predictors such as 'racePctWhite,' 'PctPopUnderPov,' and 'TotalPctDiv.

Stepwise methods—both forward and backward selection—tend to include a larger number of variables compared to best subset selection, but they also differ from each other due to the iterative nature of the selection process, which results in a broad set of predictors but with some differences in inclusion, such as 'PctOccupMgmtProf' and 'MedNumBR.'

Regularization methods like Lasso and Elastic Net introduce a penalty term, which helps in selecting only the most relevant variables by shrinking less important ones to zero. Lasso has a highly selective nature and picks only six features, including 'racepctblack,' 'pctUrban,' and 'PctIlleg.' Elastic Net, which blends Lasso and Ridge regression, selects a slightly different set, balancing between strict selection and keeping correlated variables.

Tuning parameters play a crucial role in determining feature selection, particularly for Lasso and Elastic Net. The choice of the regularization parameter (lambda) affects the number of selected features: a higher lambda leads to stronger penalization and fewer selected features, while a lower lambda allows more variables to remain in the model. Different values of lambda can yield different important features, especially for Lasso, as it tends to arbitrarily select one variable among highly correlated ones. Elastic Net mitigates this by allowing some grouped features to be retained together, thus leading to slightly different top predictors.

Despite these variations, certain features appear consistently across multiple methods. 'racepctblack,' 'PctIlleg,' 'PctPopUnderPov,' and 'pctUrban' are among the most frequently selected variables. This indicates the frequent inclusion of demographic and socioeconomic indicators indicates their strong association with the response. To summarize, Determining the most important features depends on the method used. For OLS and best subset selection, importance can be inferred from significance tests and adjusted R squared. For stepwise methods, the fact that certain variables survive

multiple iterations implies their predictive power. Regularization methods highlight features by shrinking coefficients, with nonzero coefficients in Lasso and Elastic Net indicating key predictors.

III. (B) MSE Predictions

A. (i) Average MSE for Six Methods

- **Least Squares:** 0.0192
- **Ridge:** 0.0198
- **Best Subset:** 0.0216
- **StepWise:** 0.0403
- **Lasso** 0.0189
- **Elastic Net:** 0.0185

B. (ii) Average MSE Performance Visualization

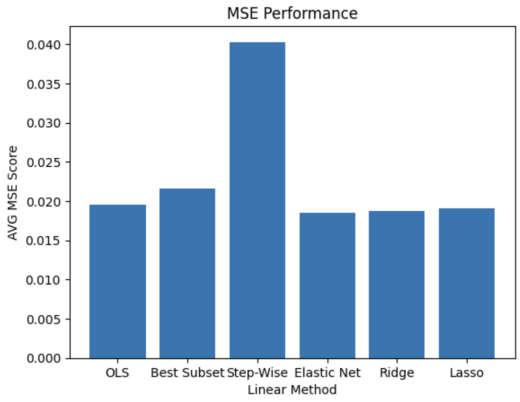

Fig. 5. "AVG MSE for Six Methods"

C. (iii) Model Prediction Reflections

Among the methods used, Elastic Net achieves the lowest average mean squared error at 0.0185, followed closely by Lasso (0.0189) and OLS (0.0192). Ridge regression has a slightly higher MSE (0.0198), while Best Subset Selection (0.0216) and Stepwise Selection (0.0403) perform worse in terms of prediction error. These results suggest that regularization methods such as Elastic Net and Lasso provide the best predictive performance, likely because they effectively balance bias and variance by preventing overfitting to noise in the training data.

Elastic Net and Lasso perform well because they impose a penalty on large coefficients, which helps prevent models from being overly complex. Elastic Net, in particular, combines Lasso's sparsity with Ridge regression's ability to handle correlated variables, making it particularly robust. Lasso alone tends to arbitrarily drop some correlated predictors, which might lead to suboptimal selection in some cases. This explains why Elastic Net has the lowest MSE—it benefits from Lasso's feature selection while mitigating its tendency to exclude useful correlated variables.

In contrast, Stepwise Selection has the highest MSE (0.0403), suggesting that it is prone to overfitting. Stepwise methods rely on iterative variable selection based on significance testing, which can lead to an unstable model, especially when dealing with correlated predictors. These methods do not incorporate any penalty for model complexity, which can lead to overfitting the training set, thereby reducing generalizability to new data.

OLS performs reasonably well (0.0192 MSE) because it uses all available features without penalization, ensuring that it captures most of the variance in the data. However, its slightly higher error compared to Lasso and Elastic Net suggests that including all variables introduces some unnecessary noise, which may reduce predictive accuracy. Ridge regression (0.0198 MSE) performs similarly to OLS but slightly worse than Lasso and Elastic Net. This is expected since Ridge does not eliminate irrelevant variables, but rather shrinks their coefficients.

Despite the differences in MSE, not all methods that produce similar prediction errors select the same subset of features. OLS, by definition, includes all variables, whereas Lasso and Elastic Net perform feature selection by shrinking some coefficients to zero. Ridge keeps all predictors, but penalizes their magnitudes. Best Subset Selection and Stepwise Selection choose different subsets of predictors based on statistical significance rather than a penalty framework, leading to different models. This means that even though Elastic Net, Lasso, and OLS have similar MSEs, the actual variables that contribute to these predictions may differ significantly.

IV. 2. Empirical/Mathematical Demonstration of Regression Properties

A. (i) Show that fitting linear regression with an intercept term is equivalent to fitting linear regression when centering Y and centering the columns of X

To empirically demonstrate this, I created synthetic data with two predictors and a true model with noise. I then fitted linear regression model; one with an intercept term and the other with centering X and Y. In the end, I compared the coefficients of both models to validate the statement. The process and result are in the appendix.

B. (ii) Show that fitting linear regression with an intercept term is equivalent to fitting linear regression when adding a column of ones to X

To empirically demonstrate this, I similarly generated synthetic data with two predictors and a true model with some noise. Then I compared the coefficients of running a linear regression model with an intercept term and the case of adding a column of ones to X. The coding process and results are in the Appendix.

C. Show that the least squares solution has zero training error when $p > n$

When p>n, the design matrix X has more predictors than observations. This means that The system Y=X*Beta is underdetermined, meaning there are infinitely many solutions. Also, The least squares solution will find a beta that perfectly interpolates the training data, resulting in zero training error. To approach this, I generate synthetic data where p>n and

fitted least squares model and the training error is calculated as 0. The coding process and results are in the Appendix.

D. Show properties of methods with correlated features

(i) To demonstrate that a least squares estimate has high variance when features are correlated, I used the Variance Inflation Factor to analyze how it increases significantly when features are highly correlated. The coding process and results are in appendix.

(ii) To demonstrate that ridge regression groups highly correlated features for sufficiently large lambda, I fitted Ridge models with various lambda values and visualized how the coefficient path behaves when the lambda increases. The coding process and results are in appendix.

(iii) To demonstrate lasso regression selects only one feature from a group of highly correlated features, I generated a simulated dataset with a set of highly correlated features, applied lasso regression, and observed that the model typically assigns a non-zero coefficient to only one feature from each correlated group, while effectively selecting it while setting the others to zero. The coding process and results are in the Appendix.

V. Appendix