Homework 5 - Regularization

Wage_regularization data set (100 points)

Due Date: Monday March 1 at 11:59 pm

Instruction:

- This is a group-work assignment!
- You are expected to submit the **.ipynb** file and the exported **.html**.
- Only one member in each group needs to submit the assignment. It will be automatically submitted for the rest of group members.
- This is a long assignment, start early!
- You will be qualified to get full mark if you beat the following performance metrics:
  - Question 1:
    - Ridge regression: RMSE_test = 0.9
    - Ridge regression: RMSE_test = 0.9
    - Lasso regression: RMSE_test = 0.9
    - Elastic net regression: RMSE_test = 0.9

Question 1 Regularization (115 points)

In this exercise I want you to apply penalized regression models to the wage data set which is available on the GitHub folder for HW5. Import the wage_regularization.csv as a data frame and call it df. I specifically want you to do the followings:

1. Standardize all the variables using StandarScaler() class from sklearn package. With the standardized variables, make a new data frame and call it df_sc.

   (**5 points**)
2. Define your feature space and target variables and then split the data into test (20%) and train set (80%) **(5 points)**
3. As a benchmark, use sm.OLS() function from statsmodel.api package to run the linear regression model on the train set. **(10 points)**
   1. Report the summary output. (5 points)
   2. From the summary report, What is the R^2 of the model in train set?(3 points)
   3. Are any of the features statistically significant at 5% level? (2 points)
4. From sklearn.linear_model import the relevant functions for Linear Regression, Ridge, Lasso and ElasticNet regression functions. Do the followings: **(25 points)**
   1. Train all the 4 models with the default features. (5 points)
   2. Save the predicted values for the test set in y_hat_linear, y_hat_ridge, y_hat_lasso and y_hat_net. (5 points)
   3. Construct a data frame named df_predictions with 5 columns. y_test, and the four y_hats from previous part (5 points)
   4. Estimate the coefficients from each model and stack them all along with the feature names in a new data frame named coefficients. (5 points)
   5. Why do you think all the coefficients of Lasso and ElasticNet models are zero? (5 points)
5. Use cross validation to find the optimal hyper parameters (alphas) for the penalized regression models. Save these optimal alphas in a new object. you need to use them in next part. **(15 points)**
6. Now go back to part 4, copy codes from cells in part 4.1, 4.2 and 4.3. You need to refit the models using the optimal hyper parameters (alphas) that you obtained from cross validation in part 5. However, name your final predictions data frame as df_predictions_optimal **(10 points)**
7. Use the variables in df_predictions_optimal to report the RMSE_test (RMSE in the test set) for all the four models. Rank the models based on their performance in the test set. Were you able to beat the simple linear model? What does this mean? **(15 points)**

8. Plot the coefficients vs alphas for each of the penalized regression models. How do you interpret each of them? You can use this range for all the models:
   (15 points)
   *alphas = 10**np.linspace(-4,2,100)*
9. From the 3 plots you generated in part 8, answer the following questions:
   (15 points)
   1 Ridge regression plot: Which coefficients drop most significantly when alpha increases from 0.0001 to 10. (name the top two) (5 points)
   2 Lasso regression plot: What are the top 2 variables that survive when alpha=0.1? (5 points)
   3 From the Lasso plot and ElasticNet plot, why the magnitude of Lasso coefficients are larger than the ElasticNet coefficients for alpha=0.0001? (5 points)

Good luck and enjoy machine learning!