SMDA 2018/19 – Exercise 4, Lecture L8 - 05/12/2018

Exercise 4: Analysis of Prostate Cancer dataset – shrinkage methods
Please, execute the following tasks and provide answers to the proposed questions.

1. Open your kernel SMDA_EX2(L5)_ProstateCancer_Surname in Kaggle

2. Generate a copy called SMDA_EX4(L8)_Shrinkage_Surname by the Fork button

3. Import Ridge from sklearn.linear_model

4. Starting from the end of the kernel, generate a ridge regression model with a specific regularization parameter alpha=1.0 (called lambda in the slides)
- Hint: follow the instructions in the Scikit learn documentation (https://scikit-learn.org/stable/modules/linear_model.html)

5. Show model's coefficients, intercept and score (on both the training and the test set)

6. Plot Ridge coefficients as a function of the regularization parameter
- Generate a vector called *alphas* of regularization parameters from 10^(-1) to 10^4 with 200 steps (see functions *r_* or *logspace* from *numpy*)
- Prepare an empty list for the coefficients
- For each value of alphas compute the ridge model and save the list of coefficients
- Plot alphas against related coefficients
- Compare the results with Figure 3.8 of the book (Hastie et al., 2009)

7. Find the best regularization parameter by cross-validation using leave-one-out
- Hint: use function RidgeCV from sklearn with the alphas vector generated in the previous point
- Hint: use only the training set (97 observations)
- Hint: use store_cv_values=True to get the leave-one-out errors
- Hint: the cv_values_ in the result object provided by RidgeCV contains one row for each observation (97 rows) and one column for each alpha in alphas (100 columns). Cell i,j contains the leave-one-out mean squared error on the i-th observation given the j-th alpha. By computing the mean by column of cv_values_ in you obtain the average cross-validation error for each alpha. This is the value that has to be minimized (see next step).

8. Plot the leave-one-out cross-validation curve
- Hint: plot the alphas vector against the average cross-validation error for each alpha computed at the last point

9. Identify the minimum leave-one-out cross-validation error and the related alpha, model coefficients and performance
- Hint: use the output of the previous steps

10. Identify the minimum 10-folds cross-validation error and the related alpha, model coefficients and performance
- Hint: set the cv parameter to 10 in RidgeCV
- Then use the output of the previous steps

11. Identify the minimum 10-folds cross-validation error and the related alpha, model coefficients and performance
- Hint: set the cv parameter to 10 in RidgeCV
- Then use the output of the previous steps

12. Import Lasso from sklearn.linear_model

13. Generate a lasso regression model with a specific regularization parameter alpha=0.1 (called lambda in the slides)

- Hint: follow the instructions in the Scikit learn documentation (https://scikit-learn.org/stable/modules/linear_model.html)

14 Show model's coefficients, intercept and score (on both the training and the test set)

15. Plot Lasso coefficients as a function of the regularization parameter

- Hint: use the lasso_path function

16. Find the best regularization parameter by cross-validation

- Hint: use function LassoCV from sklearn
- Test different number of cv folds

17. Plot the mean square error curve for each fold and for the average of all the folds

18. Show the best alpha, model coefficients and performance on training and test set