TP 2 : Linear regression

For this lab, you have to upload a **single** `ipynb` file. Please use the following script to format your filename (bad name will lead to a 1 point penalty):

```
# Change here using YOUR own first and last names
fn1 = "john"
ln1 = "smith"
filename = "_".join(map(lambda s: s.strip().lower(),
                        ["SD-TSIA204_lab2", ln1, fn1])) + ".ipynb"
```

You have to upload it on eCampus before Wednesday 03/02/2023, 23h59 in the folder corresponding to your lab group. Out of 20 points, 5 are specifically dedicated to:

- Presentation quality: writing, clarity, no typos, visual efforts for graphs, titles, legend, colorblindness, etc. (2 points).
- Coding quality: indentation, PEP8 Style, readability, adapted comments, brevity (2 points)
- No bug on the grader's machine (1 point)

Note: you can use https://github.com/agramfort/check_notebook to check your notebook is fine, and also use https://github.com/kenkoooo/jupyter-autopep8 to enforce `pep8` style.

Beware: labs submitted late, by email or uploaded in a wrong group folder will be graded 0/20.

We are given the dataset `meatspec`, see https://rdrr.io/cran/faraway/man/meatspec.html for details; we are trying to predict the value of variable fat with some covariates X (as columns) for which there are n i.i.d. measurements (as the rows of the data provided) over p covariates. However, it is not clear whether all the covariates are relevant for the prediction of Y. In this TP, we are going to consider several variants of the OLS to make a regressor under this setting and identify relevant variables.

1) Preprocess the data:
   (a) Set the random seed to 0.
   (b) Load the data. Print the mean, and standard deviation of every covariate. Is the data centered? Normalized? Standardized?
   (c) Separate the data in train and test sets: save one fourth of the data as testing (you can use `train_test_split from sklearn.model_selection)` and standardize both the training and testing sets using the `fit_transform` and transform functions in `sklearn.preprocessing.StandardScaler`.
   (d) Fit a regular OLS, do we need to fit the intercept?
   (e) Create a dataFrame `df_coef` and store the R2 coefficients of the estimated model. This data frame will be used along the TP to store and compare R2 coefficients of other variants of the OLS problem.

Variable selection

2) Program the method of the forward variable selection. You can use the test statistics of the test for nullity (as seen during the course). Do not define the stop criterion for the method, i.e. add a variables at each time until all the variables are used. Store the order of the variable selection and the associated p-value for each of them.

3) Run OLS on the variables with a p-value smaller than 0.05.

   (a) Apply the `OLS` of the `sklearn` library.

   (b) Store the R2 coefficient in `df_coef`.

4) Using `SequentialFeatureSelector` on a linear regression estimator select (with forward selection), select the same number of variables as in the previous question.

   (a) Elaborate on why the 2 algorithms do not return the same variables and store the R2 onto the corresponding dataFrame.

Ridge

5) Code your own ridge estimator using expression derived in class. Test it for a penalty parameter α spaced evenly on a log scale 10e-9 to 10e2.

   (a) Plot how the values of the coefficients change with α.

   (b) Plot how MSE of both the train and test sets change with α. Signal the minimum with a point.

   (c) For the best performing value of α (the one with smallest training error) store the R2 results.

Crossvalidation, Lasso and elastic net

6) Use the sklearn version of the Lasso. Test it for a penalty parameter α spaced evenly on a log scale 10e-5 to 10e-2.

   (a) To avoid having warnings and error you want to decrease the parameter `tol` or increase `max_iter`. Elaborate on why these warning arise and on the solution.

   (b) Plot the number of coefficients that are different from 0 for each value of α.

   (c) Plot how MSE of both the train and test sets change with α. Signal the minimum with a point.

   (d) For the best performing value of α on the test set store the R2 results.

7) Code your own version of the crossvalidation. Preferable, in the same way as sklearn's version, the length of every pair of folds should differ at most by one. Use the `sklearn` version of the Elastic net. Validate with a cross-validation that you implement. Test it for a penalty parameter α-ridge spaced evenly on a log scale 10e-10 to 10e3 and α-lasso in [0, 0.1, 0.5, 0.7, 0.9, 0.95, 0.99].

Bootstrap

8) For this question, we are going to use only variable 40 of the dataset original (non-centered) X. Plot the dataset and the regression line fitted with the whole sample. Generate 50 bootstrap samples, for each of the samples fit a regression model and plot the 50 estimated regression lines in the same plot (by setting alpha=.4 in the plotting function you can make the lines more transparent for the sake of readability of the plot). Finally, in the same plot, plot the prediction intervals (see exercise 12 in the lecture notes for the expression of the confidence intervals for the one dimensional case).

PCA

9) Compute the covariance matrix. Compute the singular value decomposition of the covariance matrix. For consistency in the notation use $U, s, V = SVD(X^\top X)$.

   (a) Plot a heatmap of the covariance matrix.

   (b) In PCA we transform the data to a new coordinate system such that the greatest variance by some scalar projection of the data lies on the first coordinate (called the first principal component, PC1), the second greatest variance in the second PC and so on. The PCs are computed given the above SVD, as XU. Instead of using the whole transformation, XU

   (c) Plot the amount of variance explained by the first k components for $k \in 2..p$.

   (d) We will use (as an approximation) the first 2 PCs. Plot the projected data using as color the value of y and interpret the plot.

(e) Run OLS on the projected data using k components for k evenly spaced in $2..p$. Store the best score in the dataFrame.

Comparison of the models

10) Summarize the results of the models and elaborate in their main characteristics.