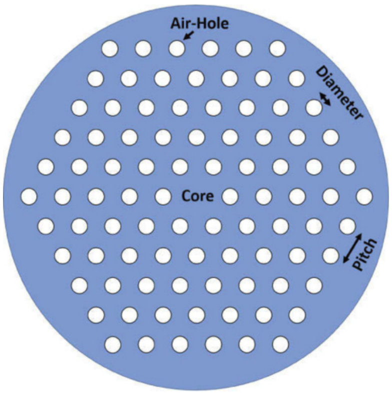
ENEE 691 Homework 2.v3

Spring 2023

Please submit your homework as a zip file which should include a pdf file explaining your solutions and your codes/notebook. Due date: 9:00 AM, March 14, 2023.

Question 1 (60 points)

For this question, you will be working on a photonic crystal fiber dataset, which is derived from the dataset used in https://doi.org/10.1364/OE.27.036414. If you have time, please read the paper.



Features (Independent Variables): Physical properties of photonic crsytal fibers (PCFs)

- Number of the rings (between the inner core and outer core)
- Air hole diameter
- Pitch length (μm)
- Radius of the fiber normalized by number of the rings (μ/m).

Outputs (Dependent Variables): Optical properties of photonic crsytal fibers (PCFs)

- Effective index
- Effective mode area
- Dispersion, and
- Confinement loss

Please download the training and testing datasets from
https://github.com/simsekergun/ENEE691/tree/main/homework2

```
X_Train = fiber_training_data_X.csv
Y_Train = fiber_training_data_Y.csv
X_Test = fiber_testing_data_Y.csv
```

Tasks:
(i) Build a linear regression model to predict effective index (first column of Y_Train) from the features (columns of X_Train). Report your R^2.
(ii) Repeat Task-(i) to predict effective mode area (second column of Y_Train)
(iii) Repeat Task-(i) to predict dispersion (third column of Y_Train)

(iv) Repeat Task-(i) to predict confinement loss (fourth column of Y_Train)

You probably will get high R^2 scores for the first two tasks and low ones for the last two.

(v) Let's repeat the last step (confinement loss prediction) by doing a simple transformation. This time take the $\log_{10}$ of the confinement loss and do the training. What's your new R^2? Please plot the histogram of confinement loss and $\log_{10}$ of the confinement loss, separately, and comment on what might going on.
(vi) Use your regression models and make predictions for all the 4 quantities (neff, Aeff, dispersion, and conf-loss) for the PCFs whose dimensions are provided in X_Test. Please provide your results in a single table. Note that I am not providing you the true values (e.g. Y_Test). The solution manual will include the true values.

Note that if you are going to use the 2nd regression model for the confinement loss prediction, then you will need to take $10^{prediction}$ *because of the transformation you made in Task (v).*

Question 2 (40 points)

For the second question, you will be working on another photodetector dataset. Please download the "MUTC1750designs.csv" either manually (using the address provided below) or using the following lines in a jupyter notebook

```
import pandas as pd
ad="https://raw.githubusercontent.com/simsekergun/photodetectors/main/MUTC1750designs.csv"
df = pd.read_csv(ad)
```

Columns
Column 1: phasenoise : true phasenoise of the photodetector (Unit: dBc/Hz)
Column 2: phasenoise_15mA : hypothetical phasenoise of the photodetector if the output current was equal to 15 mA (Unit: dBc/Hz)
Column 3: current : average current of the photodetector (Unit: mA)
Column 4: IR_max : maximum value of the impulse response of the photodetector
Column 5: decay_time: the time it takes for the impulse response to decay from its maximum value to 1% of that maximum value (Unit: ns)
Columns 6-22: t1 ... t17 : thickness each layer (Unit: nm)
Columns 23-39: d1 ... d17 : doping concentration of each layer (Unit: cm^-3)

Take $\log_{10}$ of the doping concentrations (the last 17 columns of the dataset), so that we deal with numbers in the similar ranges

```
df[df.columns[22:40]] =np.log10(df[df.columns[22:40]])
```

The last 34 columns are our features (independent variables) and the first column is the target (dependent variable)

```
X = df.iloc[:,5:]
y = df.iloc[:,0]
```

Note that Matlab's indexing and Python's indexing are different. In Matlab, df(:,1) corresponds to the first column, in Python, it's df.iloc[:, 0]. So, if you are going to copy the above lines into your MATLAB code, you need to be careful. For example, these two are equivalent:

```
Python: df[df.columns[22:40]] =np.log10(df[df.columns[22:40]])
MATLAB: df(23:39) = log10(df(23:39));
```

Tasks:
(i) Split the data, (%70 for training, %30 for testing), build a linear regression model, make predictions. Report your training and testing R^2 scores. Plot true vs predicted Y values for the testing data.
(ii) Use LASSO regularization and identify 10-15 features that can be removed from the regression model. Note that you need to play with the penalty factor to achieve this task. Since the first 17 features (columns) are layer thicknesses and the last 17 features (columns) are the doping levels, please comment on what features can be removed from the regression model.