Assignment #5

This assignment can be completed by team.
Total score: 100
Due date: 12/6

Objective: Application of machine learning

In this assignment, you will develop a **predictive model** for chemical engineers using any AI or mathematical approach such as machine learning, evolutionary algorithm, or regression analysis.

Problem description

A chemical engineer measured various properties of a gas and created a **training data set**. The measured properties include temperature, pressure, thermal conductivity, and sound velocity. The **schema** of the data set is shown below:
Dataset(Temperature, Pressure, ThermalConductivity, SoundVelocity, O1, O2) where all of the attributes are type double. The data set will be available on the course page.

The chemical engineer wonders if there are any functional relationships (or patterns) between the measured physical properties (temperature, pressure, thermal conductivity, sound velocity), O1, and O2 where O1 and O2 are outputs for given physical properties. Assume that there is no relationship between O1 and O2.

This problem can also be restated as a **regression problem** as follows:
A regression model approximates dependent variables Y to a function of independent variables X and β, that is Y ≈ f(X, β). In order to find functional relational relationships between X and Y, we rely on a training data set that consists of examples. The general multiple regression equation for a dependent variable, y_i is $y_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_m x_{mi} + \varepsilon_i$ for $i = 1..n$ where x_{mi} represents independent variables; β represents coefficients that are unknown but need to be found by your algorithm; ε represents the error; n represents the number of dependent values (or number of examples in a training data set); and m represents the number of independent variables. In this problem, O1 and O2 are dependent variables, and the physical properties of a gas (the rest of the attributes in the schema) are independent variables. **Our goal** is to find or learn a vector $\tilde{\beta}$ representing coefficients for the model that minimizes the square sum of the error, that is $\tilde{\beta} = argmin_\beta \sum_{i=1}^{n} (y_i - \beta^T x_i)^2$.

Required activities

Develop **<u>two predictive models</u>**, one for between the physical properties and O1 and the other for between the physical properties and O2. Then, write a **brief report** in Word format that includes the following:

(a) Your team name, member name(s), email addresses, and also the percentage contribution to this assignment if the assignment was completed by a team. (If a team cannot reach a consensus on the individual contribution, include the individual's claimed percent contribution with a brief description on the specific tasks performed.)

(b) A **brief description** about the software or tool used including the **name** (e.g., scikit-learn, Tensorflow, MATLAB, Weka, HeuristicLab, etc.) and its **source** (e.g., URL, company, etc.).

(c) A **brief description** about the **specific algorithm or method** used (e.g., genetic algorithm/programming, artificial neural network, multiple regression, etc.)
(d) A brief description regarding **your two models with their parameter settings** and **errors. Two types of errors** should be calculated: **average error** and **upper-bound error**. For example, suppose that a data set consists of n examples $s_1, s_2, \ldots, s_n$ with errors $e_1, e_2, \ldots e_n$ where e_i is the error of an example s_i. The average error is computed by $\sum \boldsymbol{e_i/n}$, and the upper-bound error is computed by $\boldsymbol{max(e_1, e_2, \ldots, e_n)}$. An error e_i for an example s_i is calculated between the desired output y_i in the training data set and the predicted output $\tilde{y}_i$ from your model, e.g., $e_i = (y_i - \tilde{y}_i)^2$.

Warning: Although code reuse from source codes available on the Internet is allowed, copying code from another student or team in this class is strictly prohibited. Any student or team violating this policy will receive a **ZERO** score for this assignment.

Note: The **best performing model** will receive bonus points of **20%**. The **second best performing model** will receive **10% bonus points**. The performance of a model will be measured by least average error and least upper-bound error.

What and How to submit this assignment
Turn in **your report**. Optionally, you may upload the source code ONLY when you actually implemented the algorithm used to build your model. If you have more than one file, include all the files, and zip/compress them into one file by **your (or team's) name**. Then submit the **zipped** or **compressed file** to **Titanium**. For example, if your team name is "ABC", then the zip file name should be **ABC.zip**. If the assignment was completed by team, only **ONE** of **your team members** needs to submit your team's work.

Grading policy
Your work will be graded based on the quality of your (or the team's) work as well as the completion of the requirements, the level of understanding on the problem and results, and the written report.