**CMPE541 - PROJECT #3**

**Publication Date:** 07/05/2020

**Due Date:** 29/05/2020 (Project spans a 3-week period. Ideally you should implement a task per week)

The objective of this project is:

* Have Experience in Using Regression Techniques.
* Use at least one Off-The-Shelf machine Learning Toolkit.
* Set up experimental tasks for a machine learning project.
* Do analysis on your results.

In the last part of the project, you will be preparing several regression tasks for the diabetes dataset that we have used in the previous projects, prepare several test settings, evaluate the result performance of the classifiers. In this regard, you will need two preliminaries:

**PRELIMINARIES:**

First, hopefully you have prepared the output of Project #2, because for this stage of the project, you will be using the dataset that you have produced in Project #2. If you did not, or you are not confident at the dataset you have prepared in your second project, you are allowed to ask one of your colleagues for their dataset and use it (of course with proper citations in your last report).

Second, you will need a machine learning toolkit in order to test classification tasks. For this purpose you are allowed to use any of the off-the-shelf machine learning toolkits. Some very commonly used and widely known machine learning toolkits are:

* Weka: https://www.cs.waikato.ac.nz/ml/weka/
* PhyTorch: https://pytorch.org/
* Sci-kit: https://scikit-learn.org/stable/user\_guide.html
* Orange: https://orange.biolab.si/

You can select any of the above toolkits and use it in this part of the project. Some, such as Weka has Java plug-in, terminal console application as well as a user interface while, some are python libraries. Orange is a small machine learning toolkit which uses a drag and drop system in order to build tests and training data. Weka and Sci-kit would probably be my personal choices. Alternatively, you can select some other toolkit, but it is essential that the toolkit you select supports at least Naive Bayes, SVM and at least one decision tree-based classification algorithm. All of the above toolkits also have excellent documentation and examples distributed over many easily accessible Web resources.

**CLASSIFICATION TASK:**

Next, we define our problem for the blood sugar measurement regression task. In the blood sugar measurement problem, each individual blood sugar measurement is an **instance**, The attributes we have extracted are the **features**, and the blood sugar level is a numeric **objective**. Since we would like to predict a numeric value for each instance instead of a class, we model the problem as a regression problem (Regression is a numeric generalization over the classical classification problem. Here, you will select regression algorithms, or regression versions of the algorithms from the toolkits that you will be using).

**TESTING:**

We use the time-ordered version of our dataset in order to aid us dividing the dataset into training and test splits. Hence, we define the partitioning of our dataset next. As you would remember from the second project, some of the features are dependent on previous blood sugar measurements and hence, they will not be available in the first few measurements (at least they would not be stable or reliable). We should remove these attribute values during training of the models. As a consequence, instead of dividing the dataset into two, we will divide the dataset into three:

We assume the dataset is time ordered. Then,

* **Warmup Data:** First 20% of the blood sugar measurements will be used as warmup, letting all engineered features to stabilize.
* **Training Data:** 60% of the next measurements will be used in order to train the classification models.
* **Testing Data:** remaining 20% will be used as testing data.

We will also use three different settings:

* **Global prediction:** The training model uses the blood sugar measurements of all patients. Similarly, the performance is evaluated for all patients.
* **Personalized prediction:** For each patient, we train a patient-specific classification model, tested over that particular patient's testing data. Performance is evaluated via averaging over all patients.
* **Similarity-based prediction:** Each patient is represented by a centroid calculated over all their features averaged over all observed (i.e., training samples only!) blood sugar measurements. Next, K (= 5) most similar patients is selected for each patient. Model training is performed using these patients' data (So the third setting is actually in the middle of global and personalized prediction), while the testing is performed personally for each patient. Performance is evaluated via averaging over all patients.

In order to create regression models, you are to use four algorithms: Naive Bayes, SVM Regression, C4.5, and a decision tree-based model (random forest and boosted decision trees generally perform well on real life problems. You can select either, or you can select some other algorithm)

Finally, you are required to prepare a 1-3 page report along with you codes. In the final report, we would like to see performance comparisons, reporting accuracy as MSE, RMSE. In addition you should generate a final accuracy result showing how well you did while predicting the tendency: Here, accuracy is calculated as what percent of your predictions fall into 2%, 5%, 10%, and 20% proximity of the actual value.

**Note on Feature Selection:** Instead of experimenting with different feature selection rates, try your feature selection algorithm rates against the global prediction problem, for all settings use the same feature selection rate.