COMP7019 Applied Machine Learning Report

(*Due Date: 13-Jun-2022 (***Week 15***), 23:59pm*)
Total marks = 40

Contents

1 Report Overview

Given the "`Life_Expectancy_Data.csv`" dataset, build a *model* to *predict* a country's "`Life expectancy`" using some of the following features from the "`Life_Expectancy_Data.csv`" dataset[1]:

- `Year`;
- `Status`;
- `Adult Mortality`;
- `infant deaths`;
- `Alcohol`;
- `percentage expenditure`;
- `Hepatitis B`;
- `Measles`;
- `BMI`;
- `under-five deaths`;
- `Polio`;
- `Total expenditure`;

[1] Source: `https://www.kaggle.com/kumarajarshi/life-expectancy-who`

- `Diphtheria`;
- `HIV/AIDS`;
- `GDP`;
- `Population`;
- `thinness 1-19 years`;
- `thinness 5-9 years`;
- `Income composition of resources`;
- `Schooling`.

2 Report's Contents

Ideally, your report should contain the following contents corresponding to the *machine learning project* checklist we discussed during `Week_2`'s lecture.

~~2.1 Frame the Problem~~

~~At this initial step, you may first consider what type of machine learning solution would the problem take, e.g.:~~

- ~~supervised or unsupervised learning;~~
- ~~batch or mini-batch/online learning;~~
- ~~instance-based or model based,~~

etc.

2.2 Get the Data

~~Preferably, the data can be loaded *automatically* from a fixed folder within your local machine [2], e.g., see the download script from *Slide No. 134* of `Week_2` lecture. It is also a good idea to convert the dataset into a `panda` frame format.~~

~~Examine the general dataset structure and perhaps consider missing (*null*) values within the columns (attributes) of some instances (you may also profile you data set using the `info()` method of `panda` data frame objects). Recall also that it is at this step where you should *create your test set*.~~

2.3 Explore the Data to Gain Insights

~~Visualise the data to look for possible *correlations* [3]. You may also want to experiment with different attribute combinations.~~

[2] It is enough to assume that some other outside automated process downloads the data into that local folder in your machine.

[3] For this reason, it is a good idea to convert the dataset into a `pandas` data frame to readily use the data frame object methods.

2.4 Prepare the Data for Machine Learning Algorithms

At this step, you may consider:

Data cleansing: *null/missing values cannot be handled by some machine learning algorithms.*

Handling non-numerical data: *convert text/categorical data into numerical*.

Custom transformers: *creating your own custom transformers, e.g., see the code in Slide No. 322 in* `Week_2`*'s lecture that introduces* combined attributes *as new features*.

Feature scaling: *some machine learning algorithms (e.g., SVMs) are sensitive to unscaled features, perhaps you may consider scaling the features for these algorithms.*

Transformation pipelines: *ideally, automate the whole data transformation and training processes, e.g., see Slide No. 348 of* `Week_4`*'s lecture.*

2.5 Select and Train a Model

2.5.1 Consider several models and evaluate using *cross-validation*

For this step, you may further consider training several models, e.g.:

- *Linear/logistic/softmax regression*;
- *Polynomial regression*;
- *SVM regression*;
- *Decision trees/random forests*;
- *Ensemble learning*;
- *Artificial neural networks*,

etc. Each model can be further evaluated using *cross validation*, e.g., see Slides No. 378-415. Preferably, you should also discuss why you have not considered some of the models above in your machine learning solution. Also, consider the computation cost of training and generating the predictions from your models.

2.5.2 Fine-tuning the model

You may further consider fine-tuning your model using:

- *Grid/Randomized search*;
- *Performance measures*, e.g.: *accuracy*, *precision*, *f1_scores*, *mean square error*, etc;
- *Ensemble methods*;
- *Evaluating on the test set*.

3 Machine Learning Solution Format

Your *machine learning solution* should be coded under Python and where the machine learning algorithm classes are from the `scikit-learn` library. You should submit a zipped folder containing both your Report document and your Python codes. Your report should contain enough empirical evaluations and arguments to show that your machine learning model is indeed fit-enough.