PROGRAMMING R WORKGROUP PROJECT

Table of Contents

1) Introduction

Welcome to the workgroup project of "PROGRAMMING - R MBD-EN-2020A-1" course. In this project you will be working with a real-world dataset from the Kaagle platform corresponding to the "AMS 2013-2014 Solar Energy Prediction Contest".

2) Objectives Summary

Your goals in this project will be three:

1. Perform some Exploratory Data Analysis, EDA, on this dataset (see Section 6.1) – **5 points**.
2. Create a R Markdown report with your findings after doing EDA (see Section 6.2) – **3 points**.
3. Train a Machine Learning model to try to predict solar energy production of 98 stations using the given dataset (see Section 6.3) – **2 points**.

You do not need to achieve the three goals as they will be evaluated independently.

3) Submission

You must submit the following items

1. All the R code used for EDA.
2. Your R Markdown code and output file.
3. The R code used to get your solar energy predictions and a .csv file of these predictions in the format indicated in the Evaluation section of the Kaggle competition.

The submission deadline for this project will be on **December 25th, 2020 (included)**.

4) About Kaggle

Kaggle, https://www.kaggle.com/, is an online community of data scientists and machine learners, owned by Google, Inc., which allows users to find and publish data sets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers, and enter competitions to solve data science challenges.

In particular, we will be working on the "AMS 2013-2014 Solar Energy Prediction Contest". You can check the description, evaluation metric used, leaderboard, how to make a submission and the discussion forum corresponding this competition accessing this url: https://www.kaggle.com/c/ams-2014-solar-energy-prediction-contest/overview/evaluation. It is highly recommended to carefully read this information.

5) Dataset

In this project you will work with a partly preprocessed dataset created from the original data given in the Kaggle dataset. You can find this dataset in the R object *solar_dataset.RData*, which contains a data.table with the following properties:

- A total dimension of 6909 rows and 456 columns.
- Each row corresponds to information of a particular day, ranging from 1994-01-01 to 2012-11-30. The first column, 'Date', informs you of which day corresponds to each row.
- The next 98 columns (from 2nd to 99th position) gives the real values of solar production recorded in 98 different weather stations. These columns are only informed until 2007-12-31 (row 5113); after this date these 98 columns contain NA or missing values. These missing values that you must predict to achieve the fourth goal of the project (see Section 6.4).
- The remaining columns are variables created from different weather predictors given in the Kaggle competition. They are the result of performing Principal Component Analysis, PCA, over the original data.

You have also available two other files:

1. **station_info.csv:** File with name, latitude, longitude, and elevation of each of the 98 stations.

2. **additional_variables.rds:** 100 new variables to optionally add to the ones in solar_contest.rds. All these variables correspond to real Numerical Weather Prediction, NWP, values. As in solar_contest.rds, each row corresponds to a particular day.

It is not mandatory to use these files, use them only if you think they can be helpful in your analysis.

6) Objectives Description

6.1) EDA

The goal here is to perform Exploratory Data Analysis on the given dataset to extract information, carry on data cleaning and pre-processing, and/or visualize the dataset.

Some **ideas** of what can be done here are:

- Compute statistics of each column.
- Compute correlations.
- Outlier detection/removal/correction.
- Data scaling e.g. subtract mean and divide by standard deviation.
- Dimensionality reduction.
- Visualization of column values and distributions.
- Visualization of correlations.
- Visualization on a map using leaflet or similar.
- Anything that comes to your mind and makes sense. Creativity will be rewarded.

Evaluation criteria:

1. Quality of code.
2. Correctness in a statistical/mathematical sense.
3. Use of new libraries specific for the task carried on, e.g. *outliers* package.
4. Variety in the analysis.
5. Usefulness of preprocessed steps regarding the other objectives of the project.
6. Use of ggplot for visualizations.
7. Creativity.

6.2) R Markdown

For this objective you must create a R Markdown report which includes part of the operations you carried out in 6.1. You can of course perform new operations and steps to include in this R Markdown, but it seems logical to mainly reuse the code you would create for your EDA and employ R Markdown to make your results more visually pleasing.

Some **ideas** of what can be included in this R Markdown report are:

- Use of output file configuration in the header, e.g., create a floating table of contents.
- Use of headers and other graphical customizations available through Markdown syntax.
- Use of include, echo and eval parameters for your R code blocks.
- Statistical information of each column shown as a table. You can use kable here.
- Your EDA visualizations.

Evaluation criteria:

1. Quality of code.
2. Modification of default R Markdown header.
3. Use of Markdown functionality.
4. Correct use of include, echo and eval parameters.
5. Use of new libraries specific for better visualization on R Markdown, e.g. kable package.
6. Variety in the report, it should include text blocks and code blocks with both tables/information and plots.
7. Reuse/adaptation of EDA code for your R Markdown will be rewarded.
8. Use of ggplot for visualizations.
9. Creativity.

6.3) Machine Learning Model

Your fourth and final goal is to train a machine learning model to predict the solar production in the 98 stations from dates ranging from 2008-01-01 to 2012-11-30 (both included). These predictions would be uploaded to Kaggle after the submission to check your score and compare it to the other class groups results.

Some **ideas** of what can be done are:

- Use the final data after the pre-processing steps of 6.1.
- Split the dataset in train (model training), validation (model hyperparameters tuning) and test (prediction). https://en.wikipedia.org/wiki/Training,_validation,_and_test_sets.
- Code the evaluation metric used in Kaggle for evaluation/validation purposes.
- Try different combination of hyperparameters for your models.
- Use of foreach to parallelize the training of your models.
- Build first some basic model using glm or similar and upload them to Kaggle to get an initial benchmark.
- Then try mode advanced models: random forests, neural networks, SVMs, xgboost, deep learning...Submit these predictions also to Kaggle to see if your score is improving.
- For the final submission, select the model and predictions that gave you the best score in Kaggle.

Evaluation criteria: 1 point will be given considering only your score on Kaggle after uploading your predictions. This will be rewarded following this criterion:

- **Position 1:** 1 point.
- **Position 2:** 0.8 points.
- **Position 3:** 0.65 points.
- **Position 4:** 0.5 points.
- **Position 5:** 0.4 points.
- **Position 6:** 0.3 points.
- **Position 7:** 0.2 points
- **Position 8:** 0.1 points

Only groups that submit to campus online a valid prediction file and that properly prove how they got these predictions submitting the corresponding R code used to obtain them will be rewarded with points. Models that “cheat”, i.e. use any information outside the given dataset,

or predictions created “manually” (not the output of a machine learning model) will get a zero score.

The other point will be awarded considering

1. Quality of code.
2. Code optimization e.g. use of foreach and/or vectorization.
3. Impact of pre-processing steps in the final score.
4. Correct splitting of data in train/validation/test and proper use of these three datasets.
5. Good hyperparameter tuning via train and validation datasets.
6. Test at least one advanced model (you have some examples in the **ideas** section), even if the final prediction does not come from this model because other gave better results.