uc3m | Universidad Carlos III de Madrid

OPENCOURSEWARE
ADVANCED PROGRAMMING
STATISTICS FOR DATA SCIENCE

Ricardo Aler

Third Assignment
2.5 points

Wind energy production

Wind energy production

- Wind blows, wind generators rotate, then electricity is generated
- Energy goes to the electricity network, and it is used by customers (heating, light, fridges, …)

Wind energy is non-operable

- The main issue with wind energy (and also photovoltaic solar) is that it is not under the control of the operator.
- It depends on the weather.
- At this point in time, wind energy cannot be stored.
- This is a problem because of the way the electricity market works

The electricity market

- Every day at, let's say 12:00, energy providers must give a forecast about how much energy they are going to provide for the next day, for every hour (i.e. 0:00h, 1:00h, …, 23:00h).
- This is not a problem for traditional energy sources (gas, oil, …). For instance, if at time 3:00h the provider forecasted it is going to produce x energy units, all it needs to do is to burn the appropriate amount of gas.
- But this cannot be done for wind, because it depends on the weather at 3:00h.
- All the wind energy provider can do is to forecast the weather for the next day at 3:00h.

Weather forecasts

- Nowadays, there are advanced mathematical / physical / computational models (called Numerical Weather Prediction models), which are able to forecast the weather several days in advance.
- The *Global Forecast System* (GFS, USA) and the *European Centre for Medium-Range Weather Forecasts* (ECMWF) are two of the most important NWPs.
  - http://www.ecmwf.int/

ECMWF Meteorological variables

- Some of the variables predicted by ECMWF:
  - 2 metre temperature
  - 10 metre U wind component; 10 metre V wind component
  - **100 metre U wind component; 100 metre V wind component**
  - Convective available potential energy
  - Forecast logarithm of surface roughness for heat
  - Forecast surface roughness
  - Instantaneous eastward turbulent surface stress
  - Instantaneous northward turbulent surface
  - Leaf area index, high vegetation
  - Leaf area index, low vegetation
  - …
- However, the relation between those variables and the electricity actually produced is not straightforward. Machine Learning models can be used for this task

From meteo to energy

- We intend to train a machine learning model *f*, so that:
  - Given the 00:00am ECMWF forecast for variables $A_{6:00}$, $B_{6:00}$, $C_{6:00}$, … at 6:00 am (i.e. six hours in advance)
  - $f(A_{6:00}, B_{6:00}, C_{6:00}, \ldots)$ = electricity generated at **Sotavento** at 6:00

Sotavento (http://www.sotaventogalicia.com/en)

- Sotavento is a wind farm at Galicia (North West Spain)

- We will use two sources of data:
  - Meteorological variables come from ECMWF
  - Electricity production data comes from Sotavento

The data

- It is common practice to use meteorological variables in a grid around the desired location
- In this case, we will use a 5x5 grid (Sotavento is actually located at the center = 13)
- Therefore, there are 22 ECMWF variables, forecasted at 5*5=25 locations = 550 variables (input attributes). Quite a lot.
- The first column in the dataset contains the outcome to be predicted (energy generated).

WHAT TO DO

1. Read the wind dataset (in pickle format) into a Pandas dataframe using this sentence:
2. *data = pd.read_pickle('wind_pickle')*
3. Historical data is available both from ECMWF (for the meteorological variables) and Sotavento (for energy production) from 2005 to 2010. The dataset has many columns. **Energy** is the response variable and must be kept. **Steps, month, day, hour should be removed** using Pandas.
4. The remaining variables are the input attributes. They are defined for the 25 locations in the grid. But we are going to use only those variables for location 13 (Sotavento). Therefore, use Pandas for selecting the variables that end in **.13**.
5. We are going to use Holdout (train/test) for model evaluation. Using Pandas, divide the *data* into a training set (for years 2005-08) and a test set (for years 2009-2010).
6. Remove now attribute **year**. You should have now a Pandas dataframe with column **energy**, and the input attributes for location 13 (Sotavento).
7. Convert the training and test sets from Pandas dataframes to Numpy matrices, so that they can be used by scikit-learn.
8. Do hyper-parameter tuning with **Random Search** for:
    1. Decision trees
    2. Support Vector Machines
9. Choose the best model, and evaluate it on the test partition.