Homework 3: black carbon proxy sensor in air quality sensor monitoring networks.

Jose M. Barcelo Ordinas

Universidad Politècnica de Catalunya (UPC-BarcelonaTECH),
Computer Architecture Dept.
jose.maria.barcelo@upc.edu

May 3, 2024

Black carbon is a major component of fine particulate matter, a potent warming agent in the atmosphere which contributes to regional environmental disruption and accelerates glacier melting. BC appears from incomplete combustion and comes mainly from road traffic, it is present in urban aerosols and is linked to cardiovascular and respiratory diseases. The monitoring of BC is not easy and is not regulated by the European Union (EU) Air Quality Directives. In contrast to regulated pollutants, there is not much affordable equipment to monitor BC, which makes the availability of BC measurements operationally expensive and difficult. The new proposal for a Directive of the EU Parliament on ambient air quality and cleaner air for Europe, published in October 2022, states that introducing additional sampling points for unregulated air pollutants of emerging concern, such as ultrafine particles, black carbon, ammonia or the oxidative potential of particulate matter (PM), will support scientific understanding of their effects on health and the environment. This supports the need to determine BC concentrations, either by direct or indirect methods, in European urban areas. In recent years there has been a great interest in using low-cost sensors (LCSs) to measure regulated pollutants such as CO, NO_2, NO, SO_2, O_3, and PM_{10}, $PM_{2.5}$ particles. One way to increase the availability of BC measurements without the need to deploy expensive equipment is the use of virtual sensors. A virtual sensor is defined as a mathematical model that estimates the target phenomenon at a specific location where no physical sensor is available. A proxy is a specific type of virtual sensor that estimates a target pollutant from indirect sensor measurements.

1 BC proxy using machine learning

The objective of homework 3 is to build a BC proxy using machine learning tools. The estimation model is non-linear, so, you can use any non-linear model. For this homework, we propose you to compare the proxy results with several well-known machine learning regression tools, support-vector regression (SVR, with your favorite kernel or with several of them) , random forest (RF), and a Feed-Forward Neural Network (FNNN).

Optionally, if you have time and want to test more models, you can add a Gaussian Process, a Kernel Ridge Regression or a KNN (you can choose one or several of them). It is mandatory that at least you compare SVR, RF and FFNN.

The data consists on a CSV file: ”BC-Data-Set.csv”, where it can be seen that the first row is a header that describes the data:

- **date:** Timestamp (UTC) for each measurement,
- **BC:** true values of BC concentrations, in μgr/m^3,
- **N_CPC:** ultrafine particle number concentration,
- **PM-10:** sensor measurements, in μgr/m^3,
- **PM-2.5:** sensor measurements, in μgr/m^3,
- **PM-1:** sensor measurements, in μgr/m^3,
- **NO:** sensor measurements, in μgr/m^3,
- **O_3:** sensor measurements, in μgr/m^3,
- **SO_2:** sensor measurements, in μgr/m^3,
- **CO:** sensor measurements, in μgr/m^3,
- **NO:** sensor measurements, in μgr/m^3,
- **NOx:** sensor measurements, in μgr/m^3,
- **TEMP:** temperature sensor, in °C,
- **HUM:** relative humidity sensor, in %.

Below are some steps that you can follow.

The first step consists on understanding the data. For that purpose, the best approach is to obtain some statistics (means, correlations, etc) and to plot several curves to see dependencies of the data (scatter-plots and temporal trends).

Now, the second step is to produce the proxy using the machine learning models and compare results in terms of R^2 and RMSE.

However, you have too many input parameters and it is possible that some of the features are not useful at all. A good approach is to regularize the model or use a forward subset selection mechanism (FSS). Since it is very time consuming to do this FSS for every machine learning model (it would be a good practice to see which combination produces better selection of features for each one of the machine learning model), I propose you that you do it for one of them (choose your favorite one), and then when you get your best selection, fix it to compare the same feature selection with the rest of machine learning models.

Again, plot results or/and use tables to show your results. Remember to optimize the hyperparameters of the models and it would be nice if you put some results on how evolve a model depending on the hyperparameters.

In this homework, it is not so important the temporality of your data, so you can shuffle the data if you want to improve the results. Try both cases, shuffling and not shuffling to see how it works. In either case, recover the temporality if you plot the results.

As a summary, plot results, show results, show intermediate values (e.g. results with several features using the subset selection), hyperparametrs, etc. Get your conclusions.