INFO-H515 - Big Data Scalable Analytics
Project Assignment - Phase II - 2nd session

Yann-Aël Le Borgne, Jacopo De Stefani and Gianluca Bontempi

2017-2018

1 Objective

The goal of the assignment for the second session is to design a scalable distributed online forecasting system that extends the KakfaSparkStreamingRLS notebooks.

2 Phase II requirements

The KakfaSparkStreamingRLS notebooks provide a basis for a distributed online prediction system that relies on the recursive least square (RLS) algorithm. One notebook (KafkaSendRLS) allows to send a stream of data where one output variable y is a noisy linear combination of a set of inputs x ($y = x^T\beta + w$), and one notebook (KafkaReceiveRLS) allows to receive the data stream, and run two RLS models with different forgetting factors in parallel.

You should extend these notebooks in such a way that:

- On the data generation side:
  - KafkaSendRLS should send n outputs y_j, with $y_j = x^T\beta_j + w$. Coefficients $\beta_j \in \mathbb{R}^{10}$, $1 \leq j \leq n$ should be drawn from a multivariate normal distribution before the start of the simulation.
- On the data receving side:
  - KafkaReceiveRLS should compute k RLS models with different forgetting factors, for each of the n output variables y_j, in parallel. The forgetting factors should be equally spaced in the interval $[0.5, 1]$. For example, $k = 3$ should run three models for each output y_j with $mu_1 = 0.5$, $mu_2 = 0.75$ and $mu_3 = 1$.

The prediction system should be scalable in the number of models k, and output variable n.

3 Deliverables

You need to deliver your implementation of the online distributed prediction system (only the code, no report required), and prepare a presentation of your solution, that describes the system, explains how your approach is scalable, and provides an experimental assessment of the model accuracies in terms of MSE).

Implementation environment: You are free to implement your system either on the ULB hosted cluster (which we also used during the lab sessions), install all required software in a locally-hosted environment, or use the Docker container.

Presentation: You will have 15 minutes to present this project (10 minutes presentation, 5 minutes questions). The presentation should address the following points:

- Description of the overall architecture, and why it is scalable
- Experimental results in terms of scalability
- Experimental results in terms of predictions accuracy

4 Modalities

1. The assignment should be solved **individually** (not in groups of two as in the first session).
2. The assignment will be graded on (1) the implementation itself, and (2) the presentation.
3. As for the first session, you will have to create a GIT repository, in the **INFO-H-515/2017-2018-2** repository group at `http://wit-projects.ulb.ac.be/rhodecode/` to submit your code (using the convention `project-<student>` where `student` corresponds to your student number). This repository must be made private. It is recommended that you create this repository as soon as possible to avoid last minute technical difficulties, and that you use it throughout the project to synchronize your changes.
4. Your solution should be pushed to the repository no later than the **15th of August 2018**. You get a penalty of -1 points for each day that your solution is delayed. Only the latest commit will be considered as the solution.
5. Sharing of code is not allowed (you may, however, verbally discuss ideas on how to tackle the project).
6. This project counts for 25% of your grade (5 points). This project **shall be completed individually and it shall represent your sole efforts**. The result or the copy of another student's efforts (current, or past, semester(s)), is considered academic dishonesty. Plagiarism, in the sense of copy-pasting from existing reports or code is a serious issue. To avoid plagiarism, be sure to always quote your sources and indicate clearly if something has been copied verbatim.