

RPC Currents online monitoring system

Peicho Petkov for CMS RPC

General idea

- RPC Currents monitoring system will consist of 3 modules:
 - RPC current predictor
 - RPC current model trainer
 - RPC current notification manager
- A GLM will be trained for each particular RPC HV supply channel

H2O platform (<https://www.h2o.ai/>)

- H2O is an in-memory platform for distributed, scalable machine learning
- H2O uses familiar interfaces like R, Python, Scala, Java, JSON and the Flow notebook/web interface, and works seamlessly with big data technologies like Hadoop and Spark.
- H2O provides implementations of many popular [algorithms](#) such as Generalized Linear Models (GLM), Gradient Boosting Machines (including XGBoost), Random Forests, Deep Neural Networks, Stacked Ensembles, Naive Bayes, Generalized Additive Models (GAM), Cox Proportional Hazards, K-Means, PCA, Word2Vec, as well as a fully automatic machine learning algorithm ([H2O AutoML](#)).

RPC Current trainer

- Load training data set from a dedicated DB table assembled taking the data prepared by RPC Condition AUTOMATION
- Write MOJO file to Filesystem or DB (to be decided later)
- MOJO file will contain GML object directly callable in any Java application. Almost H2O version independent.

RPC Current predictor

- Will be implemented in the RPC Condition AUTOMATION
- Will call GLM object from the desired MOJO file
- Will write the prediction to DB for further processing

RPC Current notification manager

- Reads the RPC current predictions and assigns notification status to each HV channel
 - OK
 - Warning
 - Error
 - Emergency action

Next steps

- Development of a DB for storing training data
- RPC Current trainer prototype development