A Stateless, Query Based API for Model Risk Management (Aletheutes)

Patrick Harned

3/7/2022

Aletheutes ("True" in Greek) is a stateless, streaming, query based API for model risk management. It has the following capabilities.

1. Disparate Impact Ratio evaluations for monitoring deployed ML models for bias (otherwise known as fairness)
2. Generation of local and global surrogate models for explaining predictions of black box models (explainability)
3. Accuracy Drift - Analyze and track the change in probability distribution of model features with respect to model prediction
4. Data Drift - Analyze and track the change in univariate distribution of model features

Aletheutes operates on numerically encoded model transactions stored in a DB2 database. It can be extended to analyze transactions stored in other database by leveraging IBM data fabric offerings like Data Virtualizaton and Big SQL. It leverages IBM DB2s advanced in-memory analytics capabilities to performatively analyze model transactions and offer useful insights on the smallest possible amount of data.

Stateless

Aletheutes is stateless because it requires no user configuration or any external dependencies other than a datbase connection and data to operate on. This means that Aletheutes can provide useful insights into the behavior of a deployed ML model without access to training data or external model APIS.

Query-based

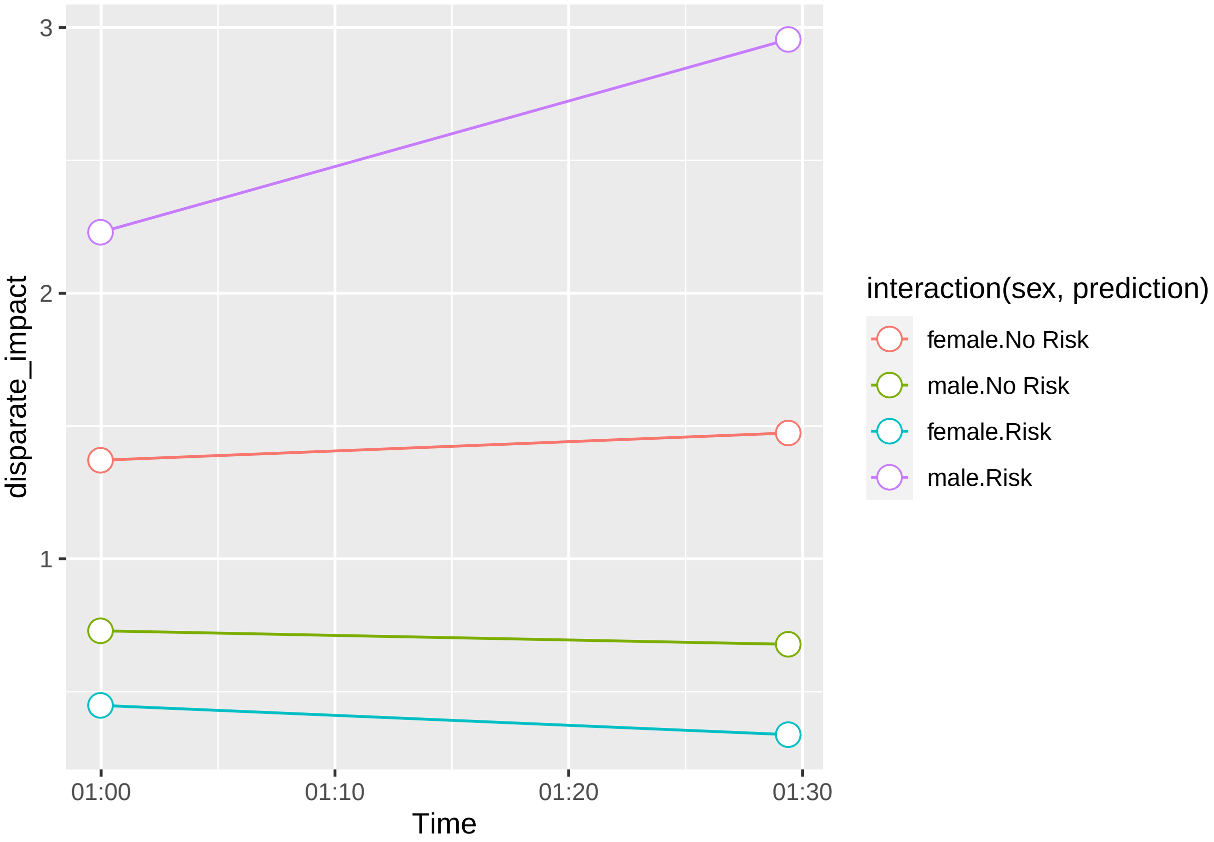
Aletheutes is query-based because it pushes all computations, including ML modeling, into the database. Aletheutes is also a horizontally and vertically scalable solution to MRM, and can operate on massive amounts of data, since IBM DB2 and compatible products are horizontally and vertically scalable through SMP and MMP deployments, and are proven capable of processing extremely high volume workloads.

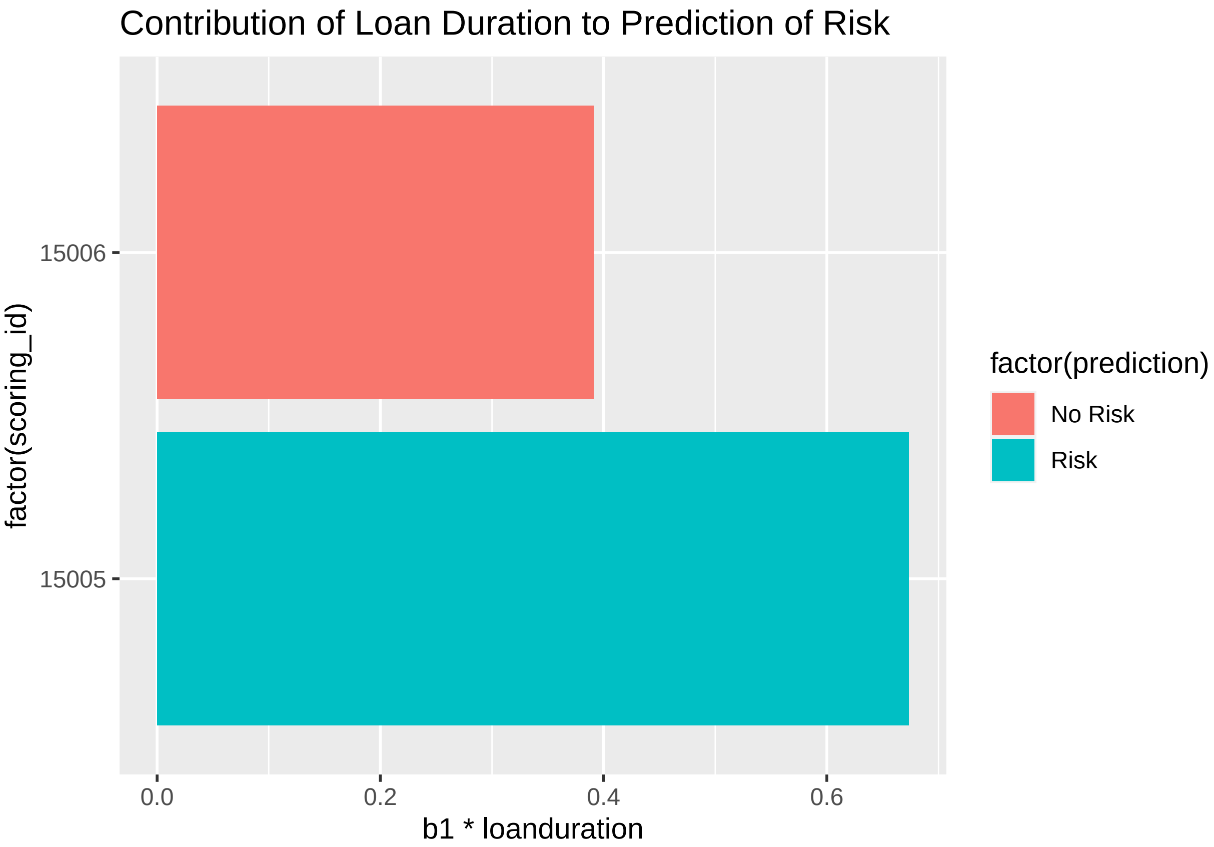
Streaming

Aletheutues is streaming because it requires no blocking ETL in-memory transformations on data in order to provide analytics results. It is written entirely in Scala, and has only two external dependencies: Akka HTTP which provides the web server for the API, and the necessary database drivers from IBM.

```r
url = "http://localhost:8080/api/disparate_impact"
body = '{"prediction":"risk", "table":"test_data2", "protected_column":"sex", "scoring_timestamp":"times
res = POST(url = url, content_type_json(), body = body)
data = content(res)

df = tibble(prediction = lapply(data, `[[`, 1), sex = lapply(data, `[[`, 2 ),group = lapply(data, `[[`,
df = as_tibble(lapply(df, unlist))%>% mutate(time = as.POSIXct(time))
df %>% ggplot(aes(x = time, group = interaction(sex, prediction), colour = interaction(sex,prediction),
```




Contribution of Loan Duration to Prediction of Risk
factor(scoring_id)
15006
15005
0.0
0.2
0.4
0.6
b1 * loanduration
factor(prediction)
No Risk
Risk