Non life pricing: empirical comparison of classical GLM with tree based Gradient Boosted Models

Innovative approach to pure premium estimation

Leonardo Petrini

Paris, 8th June 2017

Outline

Motivation: Why should we bother ?

- A correct and accurate pricing
- A better understanding of the risk components
- Number of Claims (NB) and Claim Severity (CS)

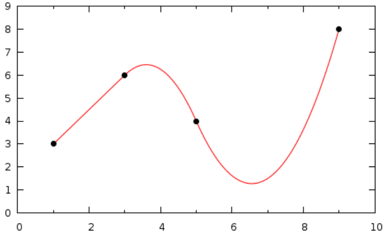
Key Quantity

$$BurningCost = NB * CS$$

Beyond GLM: Generalised Additive Models

$$g(E[y|x]) = \beta_0 + f_1(x_1) + \ldots + f_p(x_p)$$

- Advantages
  - Effective in treating non-linearity
  - Can adapt to a large variety of scenarios
- Disadvantages
  - Can easily lead to overfitting
  - Computationally intensive
- The 'mgcv' package:
  - Define a formula
  - Create a parallel cluster
  - Run the 'mgcv::bam(...)' function



Beyond GLM: Sample R code

```
library(mgcv)
library(parallel)

ctrl <- list(nthreads = ...)
cl <- makeCluster(...)

gamNB <- bam(formula = ... , data = ..., family = ...,
             cluster = ...)

stopCluster(cl)
```

Gradient Boosted Models: Understanding the Hype

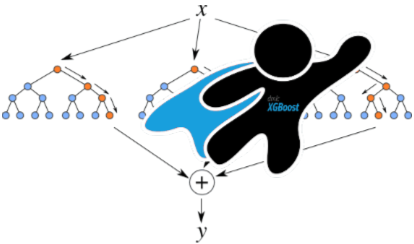
- Decision Tree based models
- Proven to work in Insurance
- XGBoost: The Kaggle "to-go" model
- Actively used by companies as ...

Allstate
You're in good hands.

Prudential

eXtreme Gradient Boosting: The State of Art

- Ensemble of Decision Trees
- Boosting Algorithm
- Active community
- Computationally attractive
- 10x Faster than GBM



XGBoost: Sample R code

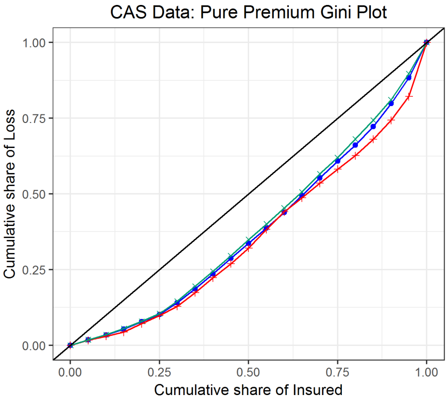
```
library(xgboost)

train <- xgb.DMatrix(data = ..., label = ...)
test <- xgb.DMatrix(data = ..., label = ...)

watchlist <- list(train = ..., test = ...)

model <- xgb.train(params = list(...) ,
                   data = ..., nround = rounds_eta,
                   objective = ... ,  eval_metric = ... )
```

Adopted Datasets

- CAS Dataset: 'freMTPL'
- Private Dataset: 'Actuarial Pricing Game'
- Pre-Processing
- Cross-Validation
- Metrics Used:
    - Number of Claims: Poisson Log-Loss
    - Claim Severity: Root Mean Square Error
    - Burning Cost: Normalised Gini Index

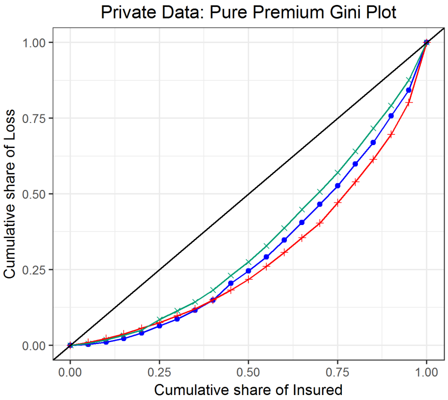
CAS Dataset: GAM vs XGBoost

	LogLoss	RMSE	Gini
GAM	0.16	3852	0.24
XGB	0.17	1980	0.30
XGB Tweedie	-	-	0.25
Gain			25%



Private Dataset: GAM vs XGBoost

	LogLoss	RMSE	Gini
GAM	0.38	2530	0.35
XGB	0.39	990	0.44
XGB Tweedie	-	-	0.38
Gain			26%



Thank You!

leopetrini@mail.com | GitHub: XGBoost-in-Insurance-2017