

Boosting Actuarial Regression Models in \mathbb{R}

Carryl Oberson

Faculty of Business and Economics
University of Basel

R in Insurance 2015

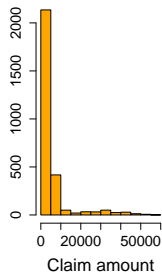
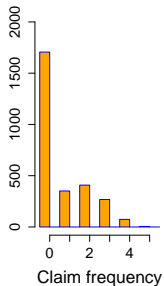
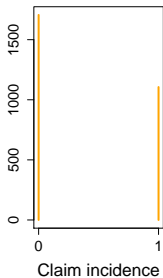
- Build regression models (GLMs) for car insurance data.
- 3 types of response variables:
 - claim incidence: $y_i = 0, 1$
 - claim count: $y_i = 0, 1, 2, \dots$
 - claim amount: $y_i \in \mathbb{R}_{>0}$
- Fit each model using the gradient boosting algorithm as implemented in the R package `mboost`.
- Assessment of the out-of-sample predictive power using 5-fold cross-validation.
- Does boosting increase the predictive accuracy of the models?

Car insurance data set

- The dataset is retrieved from the SAS Enterprise Miner database.
- Only a subset of the raw dataset is used (similarly as in Yip and Yau, 2004).
- We have $N = 2'812$ observations on 29 variables.
- Information on claim profiles for each policyholder
- 22 Potential risk factors affecting the response variables:
 - Policy details (e.g. policy date, usage of the car, etc.)
 - Driving records (e.g. whether driving licence has been revoked)
 - Personal information (gender, age, job category, etc.)

Car insurance data set

```
> library("cplm")  
> data(AutoClaim)  
> data <- subset(AutoClaim, IN_YY == 1)
```



The component-wise gradient boosting algorithm

- ... is a machine learning method for optimizing prediction accuracy.
- ... carries out variable selection.
- ... results in prediction rules that have the same interpretation as common statistical model fits

The optimal prediction function f^* to estimate is defined by

$$f^* := \operatorname{argmin}_f \mathbb{E}_{Y, \mathbf{X}}[\rho(y, f(\mathbf{x}^\top))] \quad ,$$

where ρ is a loss function assumed to be differentiable wrt f . In practice, the observed mean $R := \sum_{i=1}^n \rho(y_i, f(\mathbf{x}_i^\top))$ is minimized.

The algorithm minimizes R over f :

- 1 Initialize the function estimate $\hat{f}^{[0]}$ with offset values.
 $\hat{f}^{[m]}$ denotes the vector of function estimates at iteration m .
- 2 Specify a set of P base-learners
- 3 Increase m by one
- 4
 - Compute the negative gradient $-\frac{\partial \rho}{\partial f}$ and evaluate it at $\hat{f}^{[m-1]}(\mathbf{x}_i^\top)$, $i = 1, \dots, n$. This yields $u^{[m]} = (u_i^{[m]})_{i=1, \dots, n}$
 - Fit each of the P base-learners to $u^{[m]}$.
 - Set $\hat{u}^{[m]}$ equal to the fitted values of the best fitting base-learner according to the RSS criterion.
 - Update the estimate: $\hat{f}^{[m]} = \hat{f}^{[m-1]} + \nu \hat{u}^{[m]}$, $0 < \nu < 1$.
- 5 Iterate 3 and 4 until stopping iteration m_{stop} is reached.

Illustration of boosting in R: claim frequency

```
> library("mboost")
> NB_boost <- glmboost(formula, data = da_NA_omit, center=TRUE,
+                       family=NBinomial(nuirange = c(0, 100)))
> coef(NegBin_boost, off2int=T)
  (Intercept)      BLUEBOOK      MVR_PTS      AREAUrban
-1.048850e+00 -1.207679e-06  1.650647e-01  6.700787e-01
> plot(NegBin_boost, main="")
```

Estimate the optimal number of boosting iterations:

```
> set.seed(1234)
> m_stop_NB <- cvrisk(NegBin_boost)
> mstop(m_stop_NB)
[1] 100
> plot(m_stop_NB)
```

Illustration of boosting in R : claim frequency

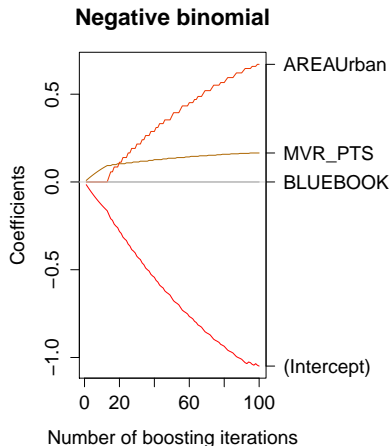
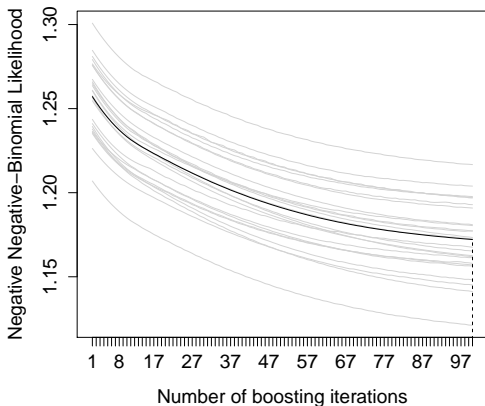


Illustration of boosting in R : claim frequency

25-fold bootstrap



k -Fold Cross-Validation is used to assess the predictive power of the models.

- 1 randomly divide the data set into k groups, or *folds*.
- 2 first fold is treated as the validation (or test) set
- 3 the method is fitted on the remaining $k - 1$ folds.
- 4 MSE_1 is computed on the observations in the held-out fold.
- 5 Compute similarly MSE_i for $i = 2, \dots, k$.
- 6 The test error rate is then simply estimated by

$$CV_{(k)} = \frac{1}{k} \sum_{i=1}^k MSE_i$$

Claim incidence: logit regression

Criterion	complex model		small model	
	glm.boost	glm	glm.boost	glm
logLik	-1191.8	-1174.5	-1207.8	-1206.1
AIC	2413.7	2422.9	2427.6	2424.1
$CV_{(5)}$	0.30046	0.030770	0.29715	0.30165

Claim frequency: negative binomial regression

Criterion	complex model		small model	
	glm.boost	glm	glm.boost	glm
logLik	-2732.6	-2673.9	-2732.6	-3434.4
AIC	5475.2	5423.7	5475.2	6878.797
$CV_{(5)}$	1.14188	1.26534	1.14139	1.10398

Claim amount: log-normal regression

Criterion	complex model		small model	
	glm.boost	glm	glm.boost	glm
logLik	-1074.5	-1055.9	-1073.0	-1072.8
AIC	2158.2	2187.8	2154.1	2155.7
$CV_{(5)}$	86406113	95110077	82423508	85144234

- Gradient boosting improves forecasting accuracy of statistical models
- Performs variable selection: useful in a context of high dimensional data
- Further issues to explore:
 - use of more flexible regression models: GAM, GAMLSS
 - claim frequency: Hurdle model
 - other?

For Further Reading I



References:

- 1 B. Hofner, A. Mayr, D. Windover, N. Robinzonov, M. Schmid.
Model-based Boosting in R; A Hands-on Tutorial Using the R Package mboost
Computational Statistics, 29:3-35., February 2012
- 2 T. Hastie, R. Tibshirani, J. Friedman.
The Elements of Statistical Learning
Springer Series in Statistics, Second Edition, 2008.