Allstate Kaggle Competition

Cristina Andronescu | Oamar Gianan | James Lee | Alex Rohr | Joseph van Bemmelen

Competition Background

How severe is an insurance claim?

> *Allstate is currently developing automated methods of predicting the cost, and hence severity, of claims. In this recruitment challenge, Kagglers are invited to show off their creativity and flex their technical chops by creating an algorithm which accurately predicts claims severity. Aspiring competitors will demonstrate insight into better ways to predict claims severity for the chance to be part of Allstate's efforts to ensure a worry-free customer experience.*

Training data: 188,318 rows and 132 columns of unlabeled data
Test data: 125,546 rows and 131 columns of unlabeled data

Overview

- **Exploring the data**
- Preprocessing
- Supervised methods
  - Linear model
  - Ridge
  - Lasso
  - Random Forest
  - GBM
- Non-supervised methods
  - PCA
- Ensembling

Exploring the data

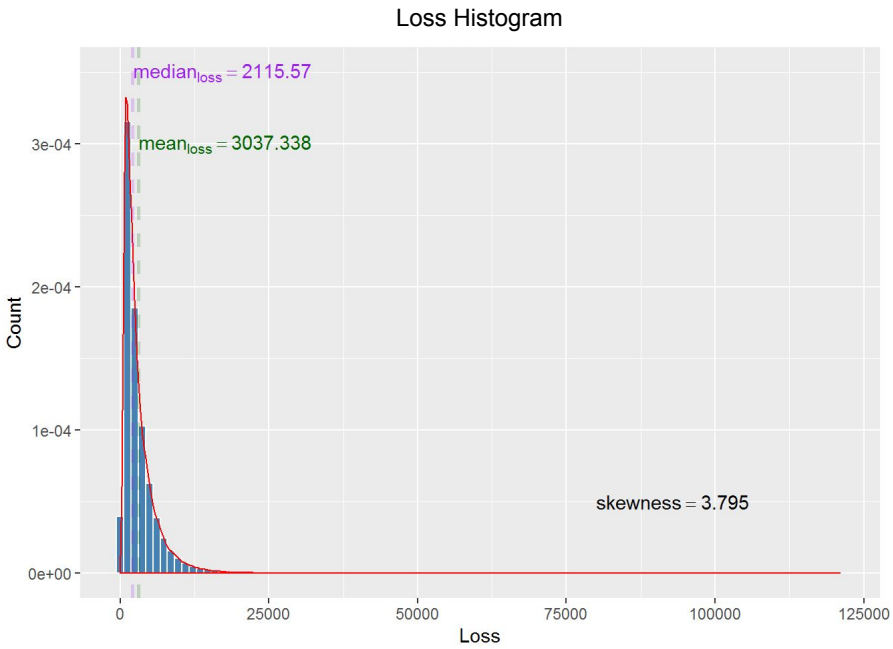
Training data: 188,318 rows and 132 columns of unlabeled data

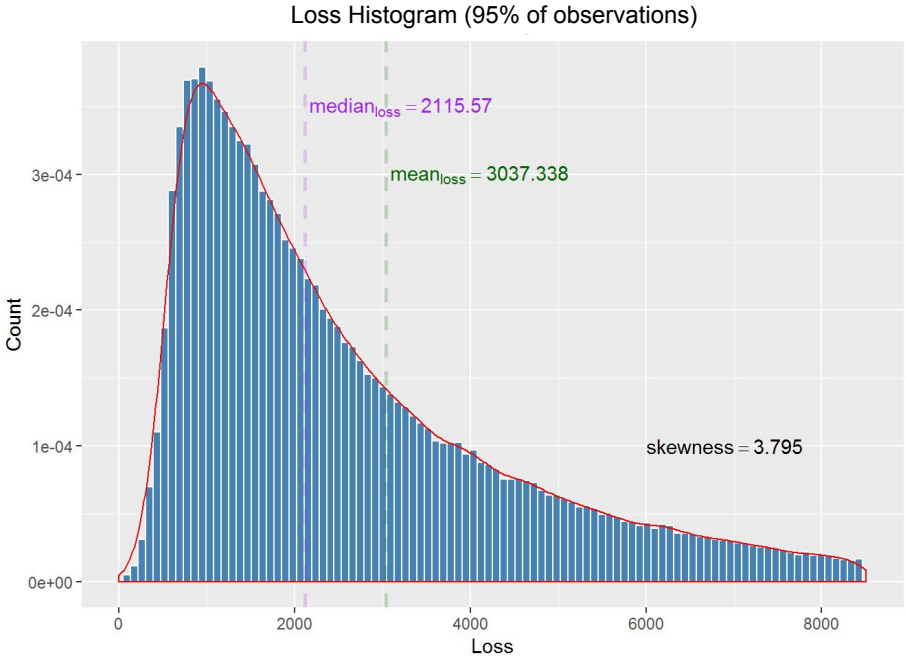
- 72 binary categorical variables (2 levels)
- 43 non-binary categorical variables (3 to 326 levels)
- 14 continuous variables
- Continuous dependent variable “loss”

Test data: 125,546 rows and 131 columns of unlabeled data

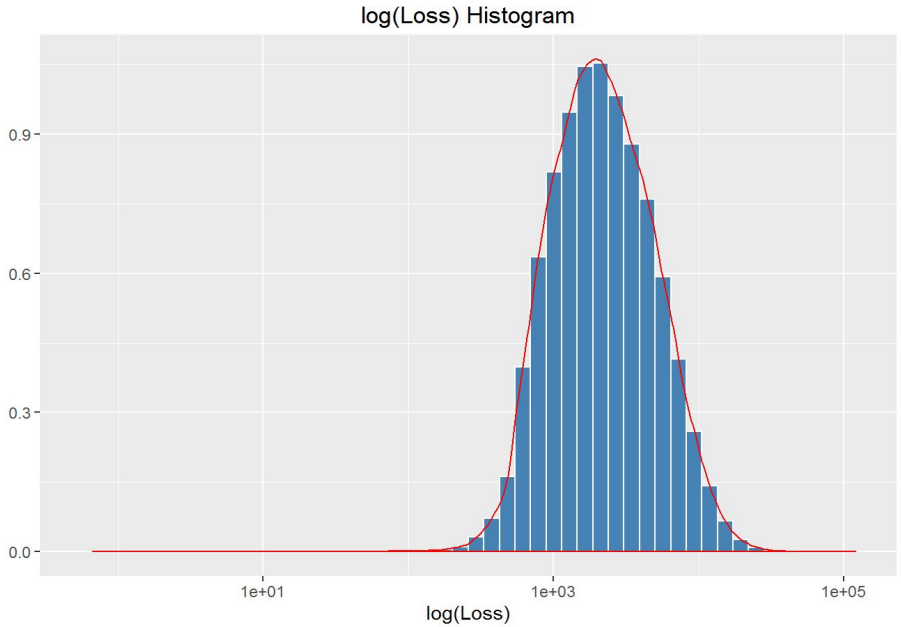
- Some of the variables have additional levels in the test set!

Exploring the data





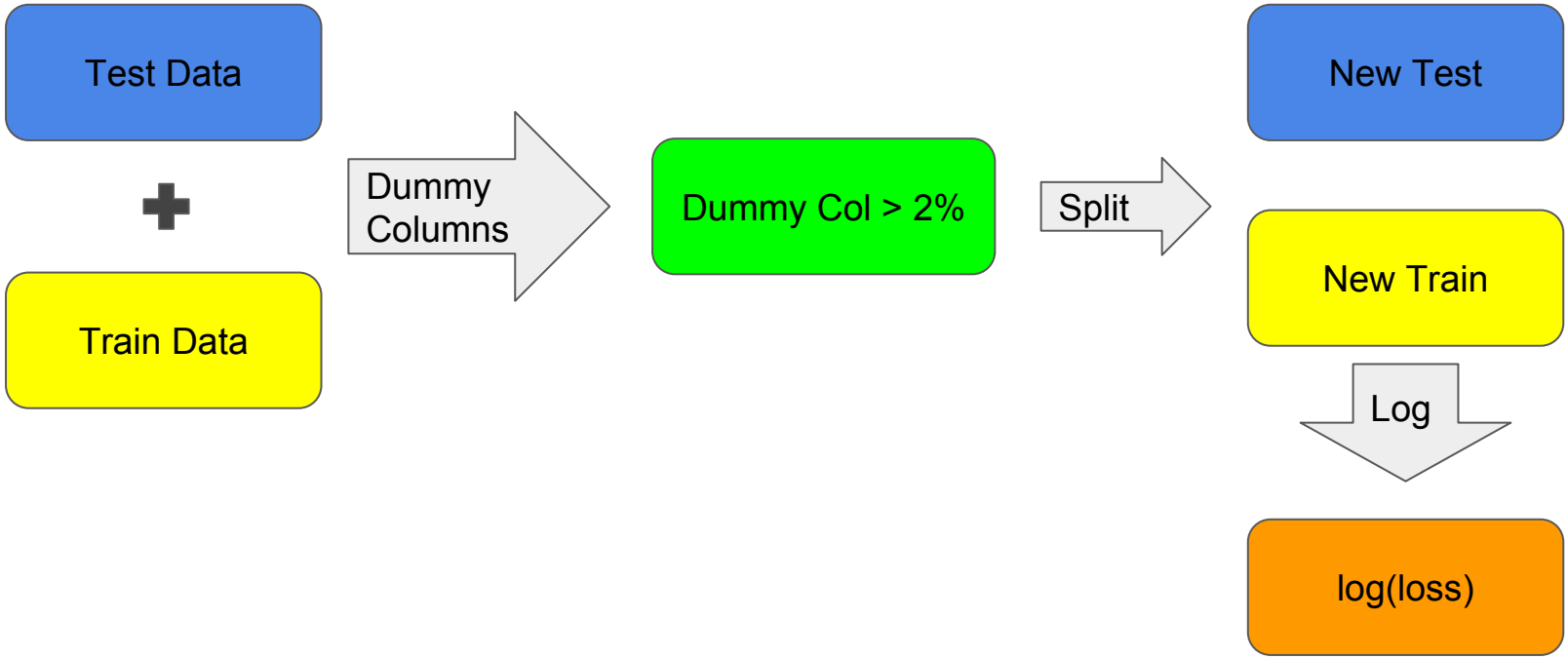
Exploring the data



Overview

- Exploring the data
- **Preprocessing**
- Supervised methods
  - Linear model
  - Ridge
  - Lasso
  - Random Forest
  - GBM
- Non-supervised methods
  - PCA
- Ensembling

Processing the Data



Overview

- Exploring the data
- Preprocessing
- **Supervised methods**
  - Linear model
  - Ridge
  - Lasso
  - Random Forest
  - GBM
- Non-supervised methods
  - PCA
- Ensembling

Linear Model

```
library(caret)
set.seed(0)
inTrain1<- createDataPartition(y=new.all.cd.train$loss, p=0.80, list=FALSE, times=1)
training<-new.all.cd.train[inTrain1,]
testing<-new.all.cd.train[-inTrain1,]

lmFit1 <- train(loss~., data=training, method='lm')

lmFit1adj2 <- train(loss~. - cat114.OTHER -cat111.OTHER -cat103.OTHER -cat101.OTHER
                    -cat102.OTHER -cat90.OTHER -cat89.OTHER, data=training, method='lm')

lmFit1adj3 <- train(loss~. - cat114.OTHER -cat111.OTHER -cat103.OTHER -cat101.OTHER
                    -cat102.OTHER -cat90.OTHER -cat89.OTHER -cat6.B -cat8.B -cat10.B
                    -cat10.B -cat15.B -cat19.B -cat19.B -cat24.B -cat30.B -cat33.B
                    -cat43.B -cat45.B -cat46.B -cat58.B -cat60.B -cat62.B -cat64.B
                    -cat66.B -cat68.B -cat69.B -cat70.B -cat81.OTHER -cat82.B -cat82.B
                    -cat83.B -cat84.OTHER -cat86.D -cat88.D -cat88.OTHER -cat92.OTHER
                    -cat96.OTHER -cat97.C -cat97.E -cat97.OTHER -cat98.C -cat98.D
                    -cat98.OTHER -cat99.R -cat99.T -cat100.I -cat104.F -cat104.G -cat104.H
                    -cat104.K -cat104.OTHER -cat105.E -cat105.F -cat105.H -cat106.F
                    -cat106.G -cat106.J -cat107.H -cat108.F -cat108.G -cat108.G -cat109.BI
                    -cat109.OTHER -cat110.CL -cat110.CO -cat110.EG -cat110.OTHER -cat113.AX
                    -cat113.OTHER -cat115.K -cat115.L -cat115.L -cat115.M -cat115.N -cat115.N
                    -cat115.O -cat115.OTHER -cat115.P -cont3 -cont5 -cont6 -cont13,
                    data=training, method='lm')
```

summary(lmFit1)

Includes all variables

```
Residual standard error: 0.5067 on 150443 degrees of freedom
Multiple R-squared:  0.5215,     Adjusted R-squared:  0.5208
F-statistic: 773.3 on 212 and 150443 DF,  p-value: < 2.2e-16
```

summary(lmFit1adj2)

Excludes all variables with NA coefficients in lmFit1

```
Residual standard error: 0.5067 on 150443 degrees of freedom
Multiple R-squared:  0.5215,     Adjusted R-squared:  0.5208
F-statistic: 773.3 on 212 and 150443 DF,  p-value: < 2.2e-16
```

summary(lmFit1adj3)

Excludes all variables not significant at least at the 90% confidence level in lmFit1

```
Residual standard error: 0.507 on 150513 degrees of freedom
Multiple R-squared:  0.5208,     Adjusted R-squared:  0.5203
F-statistic:  1152 on 142 and 150513 DF,  p-value: < 2.2e-16
```

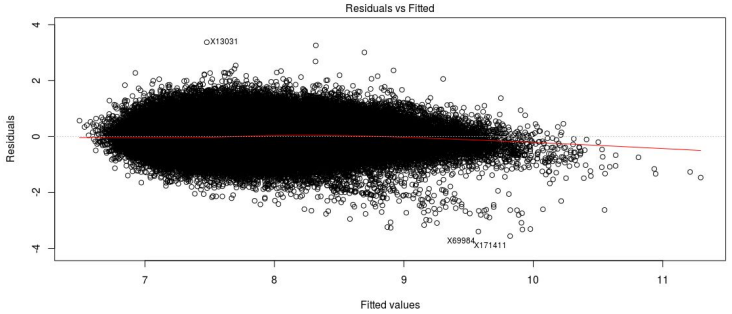
Linear Model

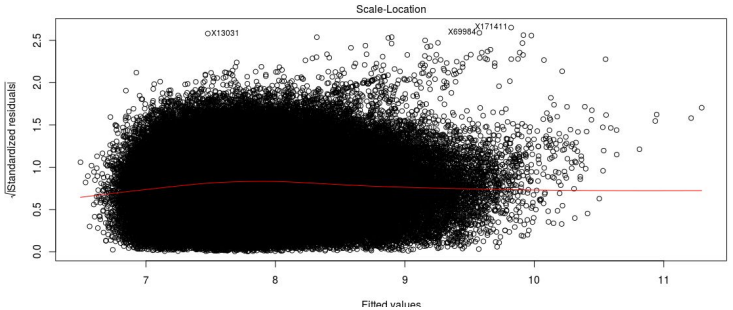
```
varImp(lmFit1adj3, scale = FALSE)

lm variable importance

only 20 most important variables
shown (out of 142)
```

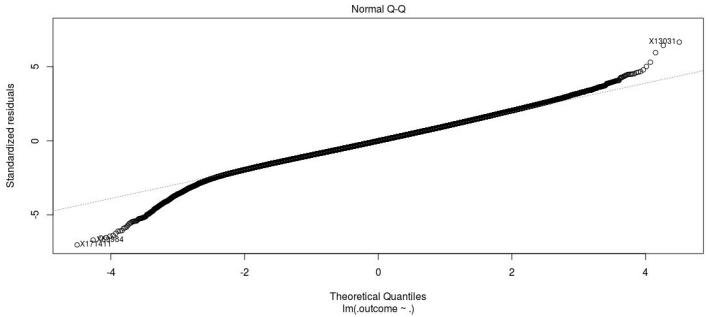
	Overall
cat80.D	71.46
cat53.B	51.28
cat79.D	49.50
cat81.D	40.12
cat100.G	39.95
cat100.L	38.19
cat112.J	37.95
cat2.B	32.56
cat100.H	31.89
cat101.C	29.60
cat112.OTHER	29.50
cat101.D	29.23
cat72.B	28.23
cat26.B	27.96
cat44.B	27.91
cont2	27.45
cat12.B	26.34
cat1.B	25.88
cat100.OTHER	23.22
cont7	22.37

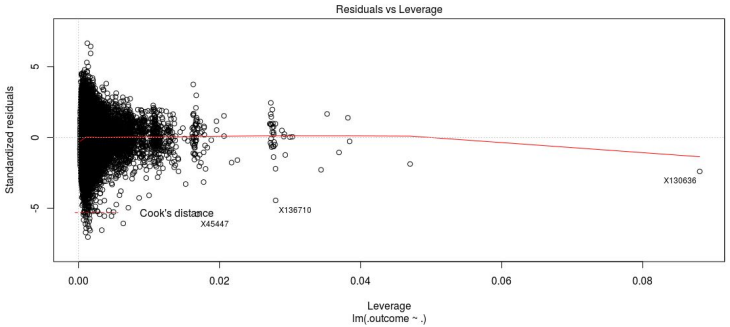
RMSE:
0.5054915

MAE score:
1249.45









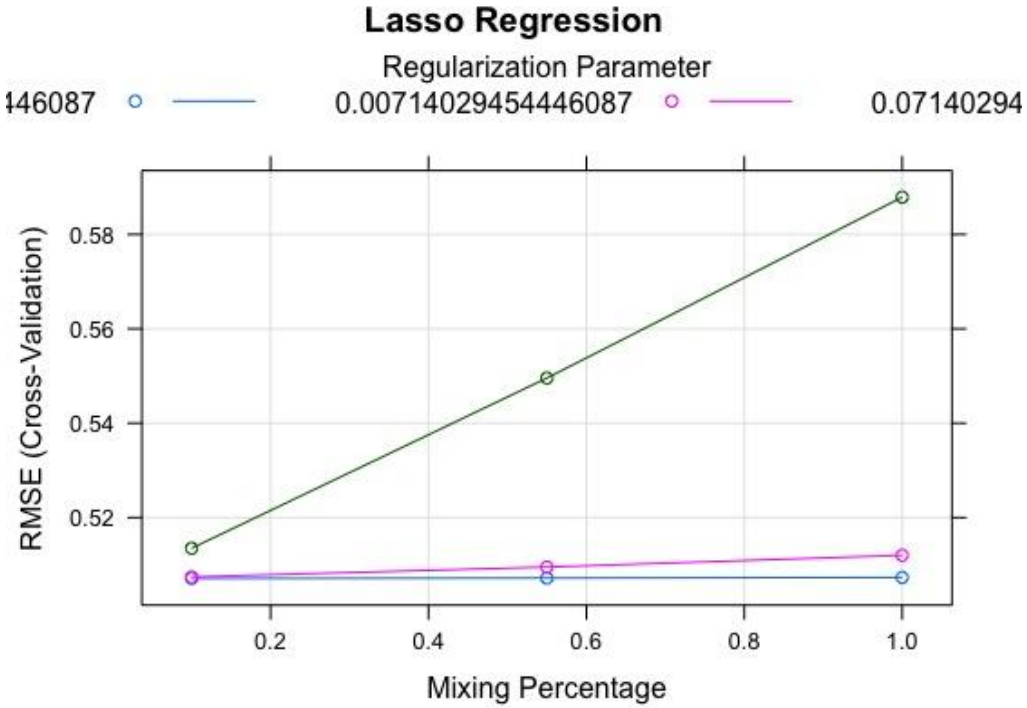
Lasso model

150656 samples
218 predictor

Pre-processing: scaled (218), centered (218)
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 135590, 135591, 135592, 135591, 135589, 135589, ...
Resampling results across tuning parameters:

alpha	lambda	RMSE	Rsquared
0.10	0.0007140295	0.5070992	0.5200337
0.10	0.0071402945	0.5074262	0.5194636
0.10	0.0714029454	0.5135051	0.5111662
0.55	0.0007140295	0.5071834	0.5198790
0.55	0.0071402945	0.5095135	0.5160203
0.55	0.0714029454	0.5495715	0.4566534
1.00	0.0007140295	0.5073161	0.5196412
1.00	0.0071402945	0.5120191	0.5121090
1.00	0.0714029454	0.5878813	0.3832404

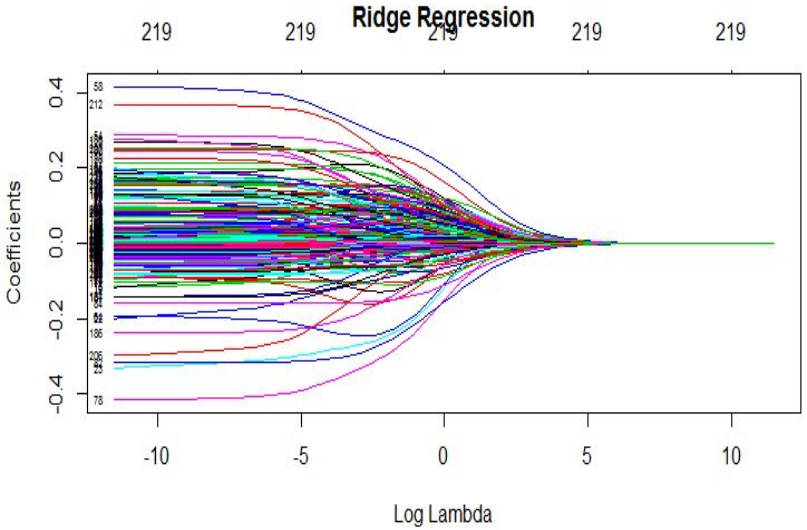
RMSE was used to select the optimal model using the smallest value.
The final values used for the model were alpha = 0.1 and lambda
= 0.0007140295.

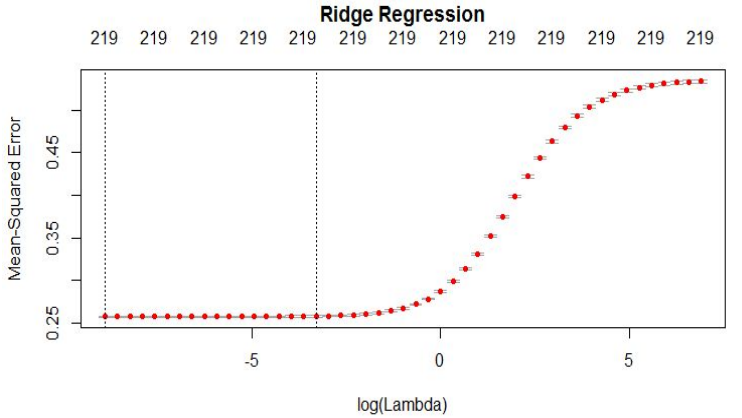


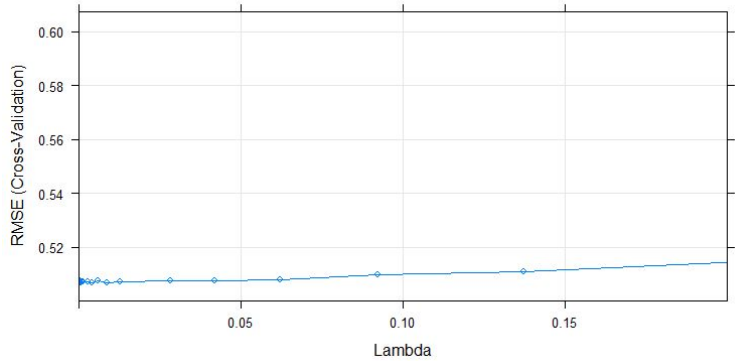
RMSE for the model: 0.5070992

MAE score: 1248.36

Ridge Model


Ridge Regression
219 219 219 219 219
Coefficients
0.4
0.2
0.0
-0.2
-0.4
-10
-5
0
5
10
Log Lambda


Ridge Regression
219 219 219 219 219 219 219 219 219 219 219 219 219
Mean-Squared Error
0.45
0.35
0.25
-5
0
5
log(Lambda)


RMSE (Cross-Validation)
0.60
0.58
0.56
0.54
0.52
0.05
0.10
0.15
Lambda

Ridge Model

lambda	RMSE	Rsquared
1.000000e-05	0.5071109	0.5200358
2.212216e-05	0.5075730	0.5192098
3.290345e-05	0.5069499	0.5199159
4.893901e-05	0.5072538	0.5197626
7.278954e-05	0.5072210	0.5196924
1.082637e-04	0.5071089	0.5200397
2.395027e-04	0.5067146	0.5208755
3.562248e-04	0.5075697	0.5192165
5.298317e-04	0.5069426	0.5199301
7.880463e-04	0.5072481	0.5197738
1.172102e-03	0.5071043	0.5200493
2.592944e-03	0.5070029	0.5206113
3.856620e-03	0.5067233	0.5208621
5.736153e-03	0.5075894	0.5191892
8.531679e-03	0.5069802	0.5198728
1.268961e-02	0.5071969	0.5199062
2.807216e-02	0.5075871	0.5191559
4.175319e-02	0.5076327	0.5197923
6.210169e-02	0.5078219	0.5195180
9.236709e-02	0.5096351	0.5170850
1.373824e-01	0.5108307	0.5168964
3.039195e-01	0.5205381	0.5131789
4.520354e-01	0.5334754	0.5099944
6.723358e-01	0.5577743	0.5070403
1.000000e+00	0.6006881	0.5015040

RMSE was used to select the optimal model using the smallest value.

The final value used for the model was lambda = 0.0002395027.
Kaggle score of 1232
RMSE: 0.5067146

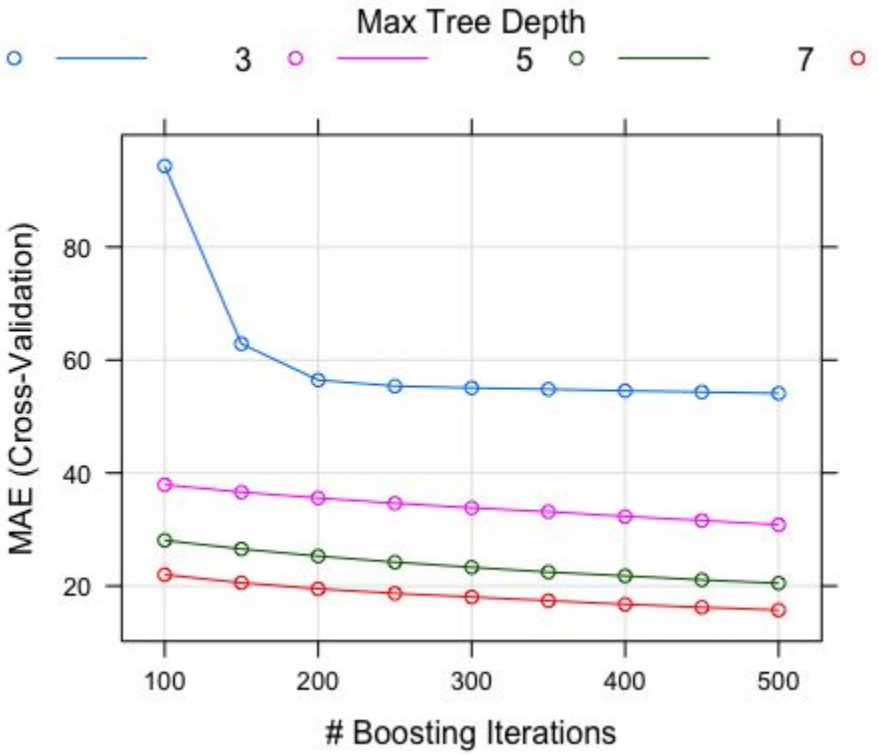
loess r-squared variable importance

only 20 most important variables shown (out of 218)

	Overall
cat80.D	100.00
cat79.D	76.08
cat101.OTHER	49.97
cat12.B	48.69
cat10.B	38.45
cat81.D	35.45
cat1.B	30.04
cat2.B	29.47
cat87.D	29.47
cat9.B	26.72
cat72.B	24.78
cat11.B	24.57
cat13.B	23.60
cat57.B	22.21
cat100.I	21.54
cat7.B	19.15
cat89.OTHER	19.15
cat3.B	16.66
cat87.OTHER	16.30
cat16.B	16.15

GBM

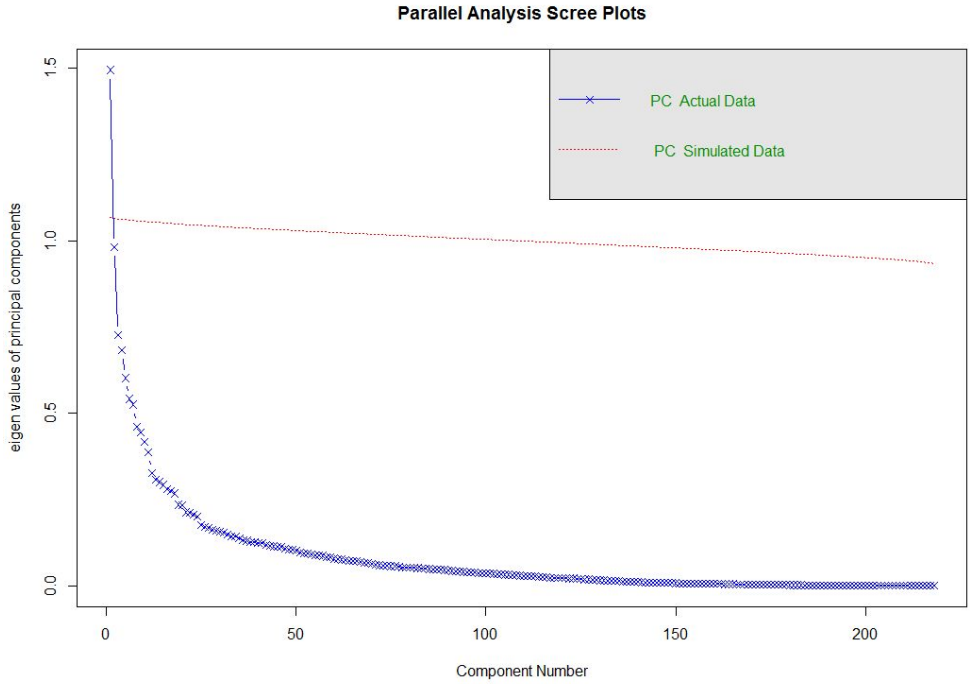
- GBM was done on the pre-processed dataset.
- The following parameters were used:
  - N.trees - 500
  - Interaction depth - 1,3,5,7
  - Shrinkage - 0.1
- An MAE of 1161.419 was achieved on a subset of the train dataset.



Overview

- Exploring the data
- Preprocessing
- Supervised methods
  - Linear model
  - Ridge
  - Lasso
  - Random Forest
  - GBM
- **Non-supervised methods**
  - PCA
- Ensembling

PCA


Parallel Analysis Scree Plots
PC Actual Data
PC Simulated Data
eigen values of principal components
Component Number
0.0
0.5
1.0
1.5
0
50
100
150
200

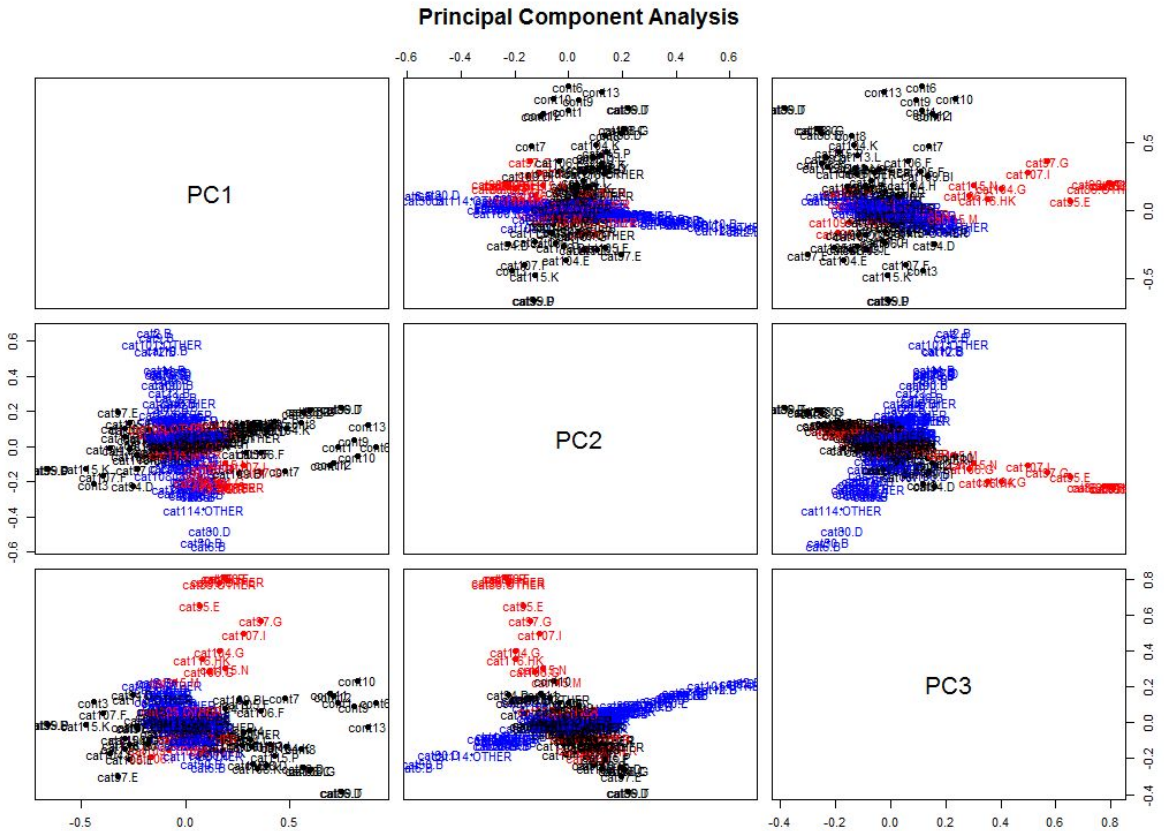
PCA

	PC1	PC2	PC3
SS loadings	12.37	7.03	6.38
Proportion Var	0.06	0.03	0.03
Cumulative Var	0.06	0.09	0.12
Proportion Explained	0.48	0.27	0.25
Cumulative Proportion	0.48	0.75	1.00

Mean item complexity = 1.6
Test of the hypothesis that 3 components are sufficient.

The root mean square of the residuals (RMSR) is 0.06

Fit based upon off diagonal values = 0.53



Overview

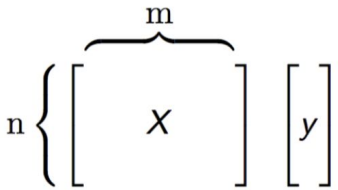
- Exploring the data
- Preprocessing
- Supervised methods
  - Linear model
  - Ridge
  - Lasso
  - Random Forest
  - GBM
- Non-supervised methods
  - PCA
- **Ensembling**

Ensembling

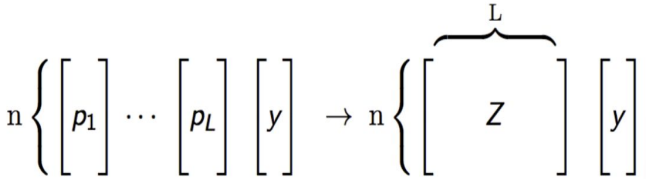
- None of the models did well on its own.
- But choosing from the models and parameters we tried, we assembled a group of learners.
- H2O and H2O ensemble was used.

Ensembling

- Start with L base learners (each with its own model parameters)
  - Base learners will be trained on the “Level-zero data” to produce L number of predictions, p.
- Column bind all predictions, p.
  - These will be the new predictors for response, y.
- Specify a metalearner.
  - Metalearner will be used on the “Level-one data”



“Level-zero” data



“Level-one” data

Ensembling

- Creating learners with parameters:
  - h2o.glm.3 <- function(..., alpha = 1.0) h2o.glm.wrapper(..., alpha = alpha)
  - h2o.randomForest.1 <- function(..., ntrees = 300)
  - h2o.gbm.3 <- function(..., ntrees = 500, max_depth = 7, seed = 1)
  - h2o.deeplearning.1 <- function(..., hidden = c(500,500), epochs = 50, seed = 1)
- Setup base learners and metalearner to be used on Level-zero data:
  - learner <- c("h2o.glm.wrapper", "h2o.randomForest.1", "h2o.gbm.3", "h2o.deeplearning.wrapper", “h2o.deeplearning.1”))
  - metalearner <- "h2o.gbm.1"
- Train & test:
  - fit <- h2o.ensemble(x = x, y = y, data = train, family = family, learner = learner, metalearner = metalearner)
  - pred <- predict(fit = fit, newdata = test)

Ensembling

- Base learners:
  - Glm: MAE: 1266.261
    - Lambda = 1e-5
  - RandomForest: MAE: 1210.532
    - N.trees = 300
    - Max_depth = 20
  - Gbm.1: MAE: 1163.804
    - N.trees = 500
    - Max_depth = 5
  - Gbm.2: MAE: 1167.37
    - N.trees = 300
    - Max_depth = 5
  - Gbm.3: MAE: 1190.555
    - N.trees = 300
    - Max_depth = 3
  - Deeplearning MAE: 1173.592
    - Hidden = c(20,20)
    - Epochs = 10
- Metalearner:
  - Gbm.1
- Kaggle score of 1125.39604

Thank you!