Kaggle Project

Sam Castillo

Introduction

- Exploratory Analysis
- Modeling strategy
- Model building
  - Linear models
  - Tree-based model
- Model evaluation
- Variable Importance
- Tools & Sources

The two data files

- Train
  - a target variable “Amount”
  - 33 anonymous predictor variables
  - ~200,000 records
- Test
  - Missing “Amount” variable
  - 33 anonymous predictor variables
  - ~50,000 records

Objective

Intro

- Use predictive modeling to estimate an unknown quantity using data provided
- The two most common error metrics for regression are Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE)

$$\mathrm{MAE} = \frac{1}{n}\sum |y_i - \hat{y}_i| \qquad \mathrm{RMSE} = \sqrt{\frac{1}{n}\sum (y_i - \hat{y}_i)^2}$$

- Key difference between using MAE is that no square is taken. In RMSE, larger errors from outliers are heavily penalized
- This project uses MAE

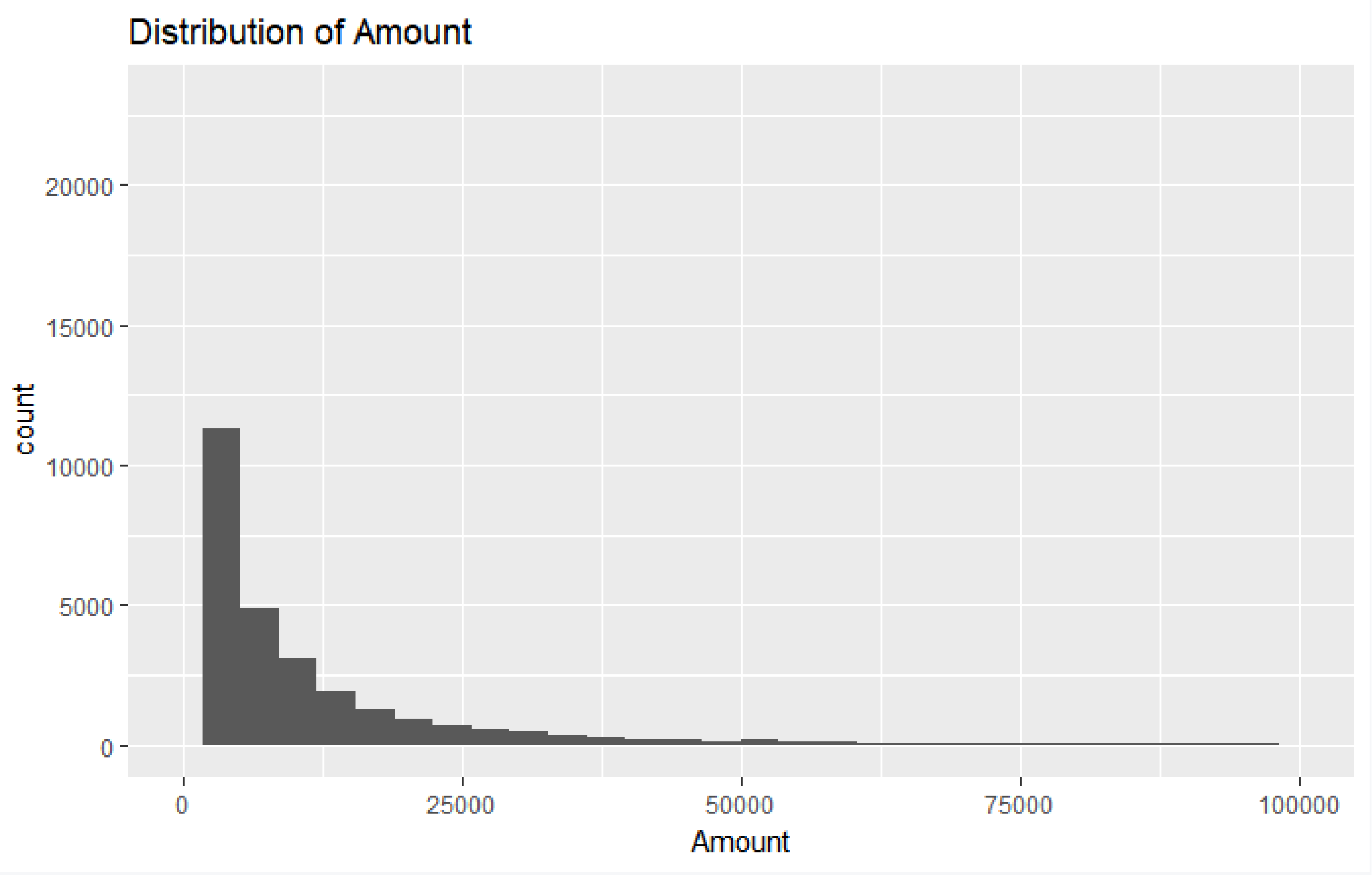
Exploratory Analysis

Visualizations & diagnostic checks

The target distribution

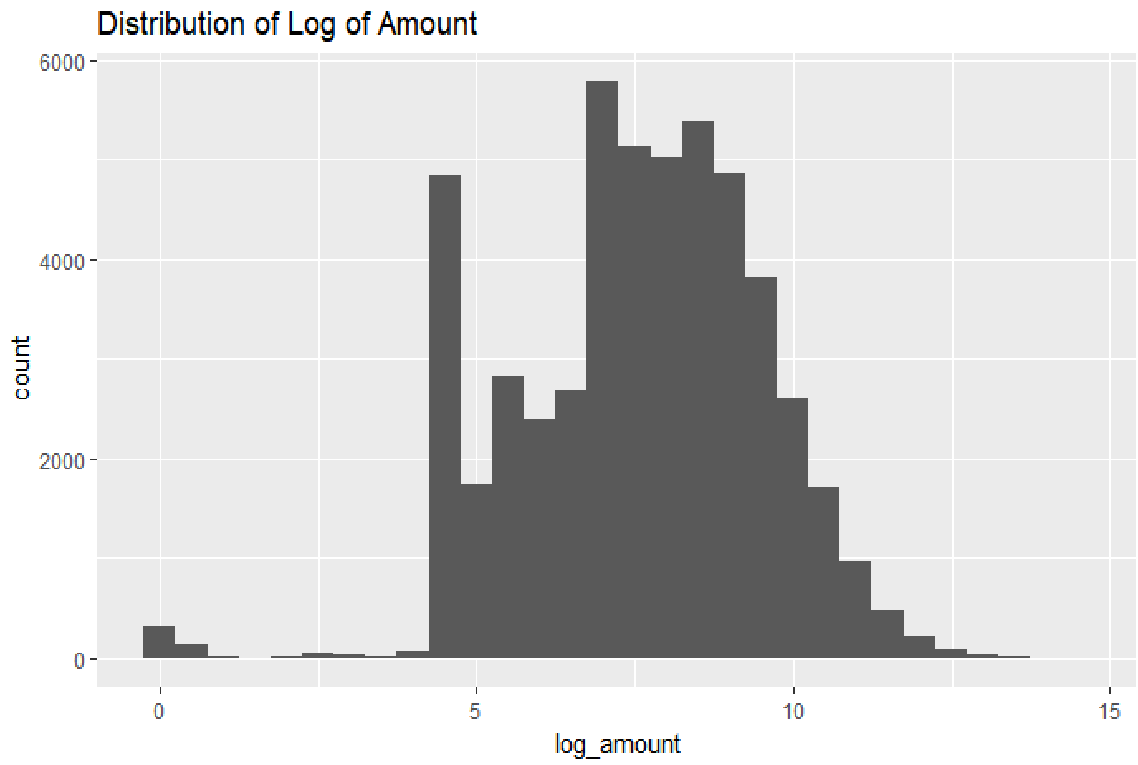
Exploratory Analysis

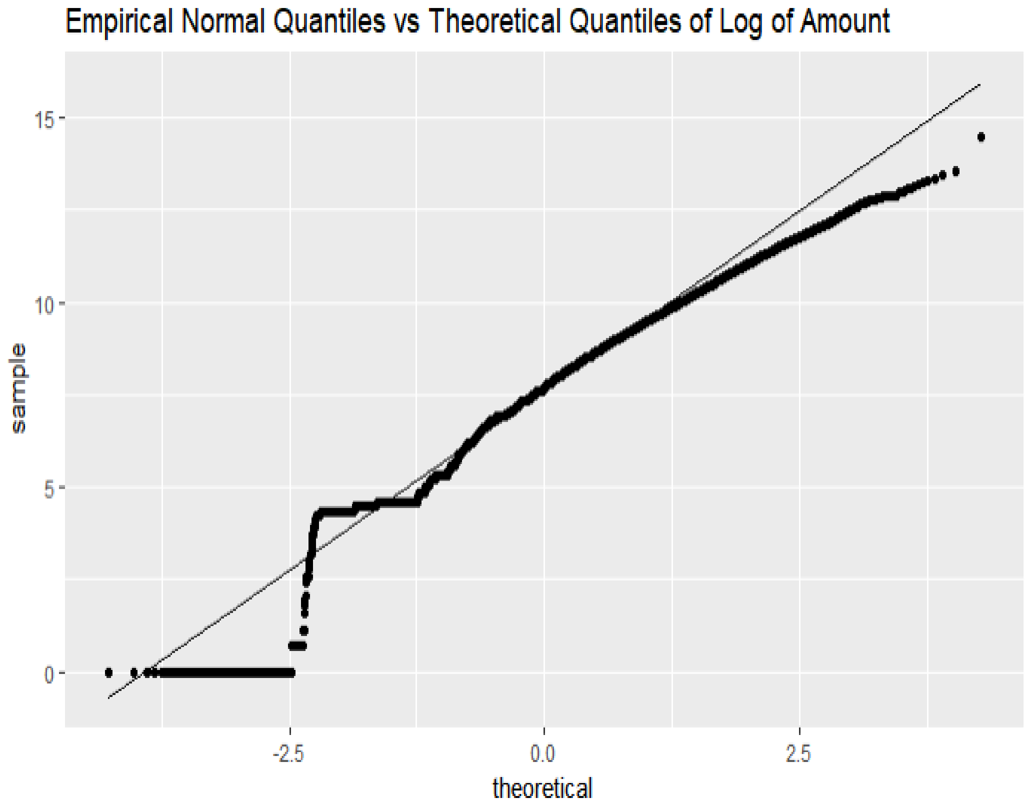
- Long tailed
- Right skewed



The target distribution

Exploratory Analysis


Distribution of Log of Amount


Empirical Normal Quantiles vs Theoretical Quantiles of Log of Amount

- Apply transform Y = log(Amount + 1)
- Point mass spikes at 100.10 and several other values
- Truncated below this value

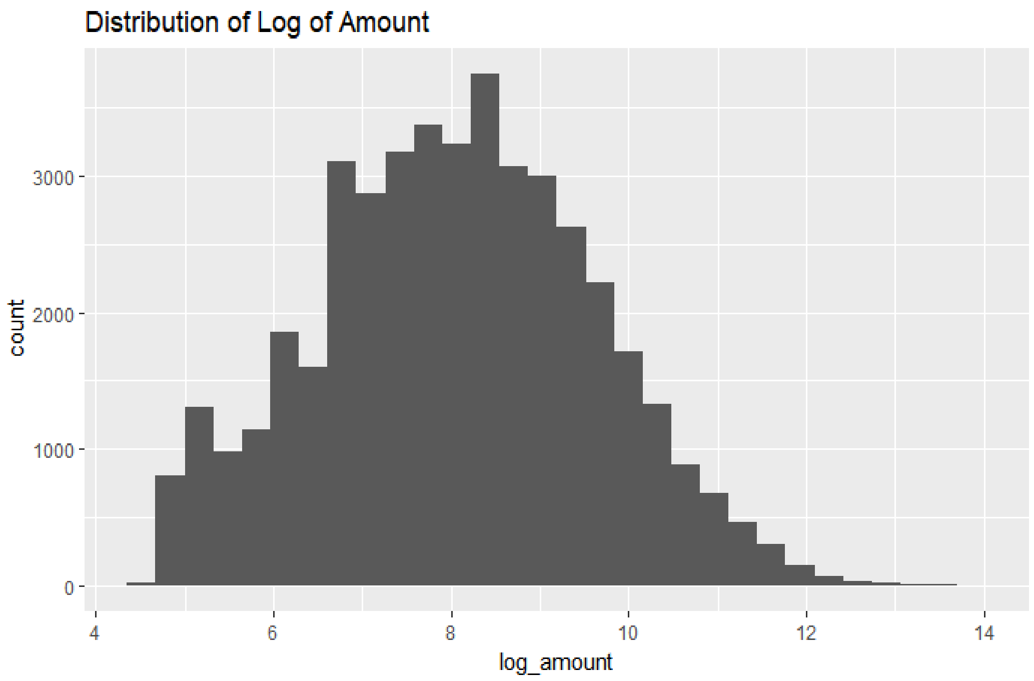
Truncated data “spikes”

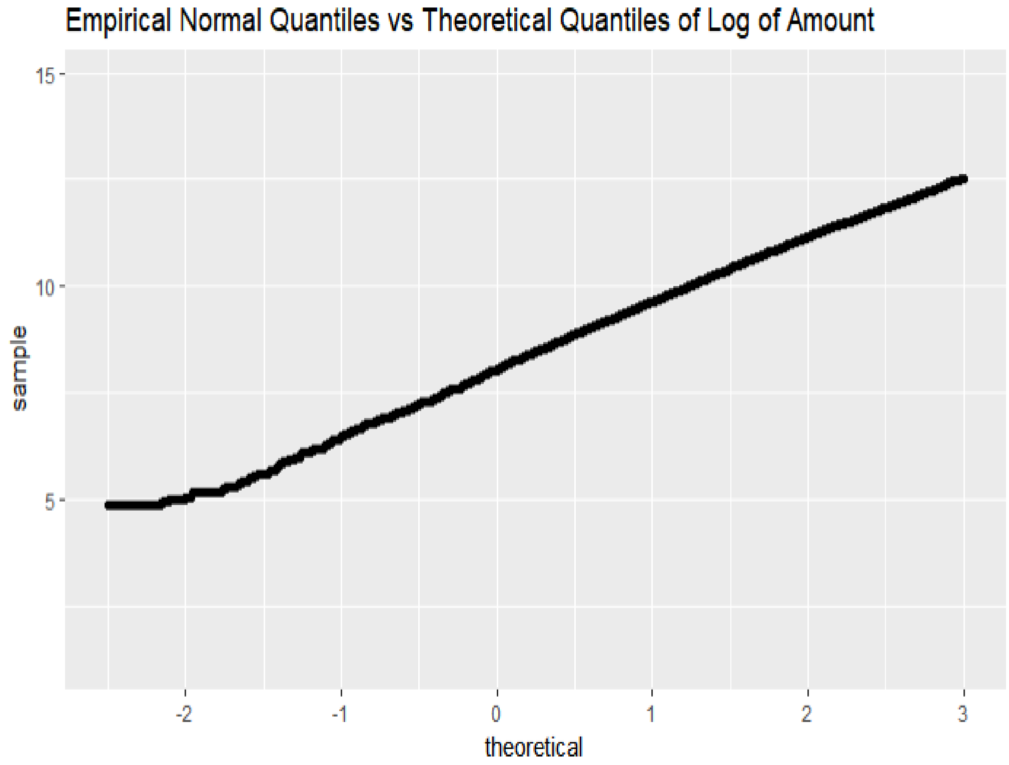
- Several values of “Amount” appear more times than should be expected
- This is very common in insurance data and is known as “left trunctation”

Amount	Number of Observations
100.1	12,302
198.198	5,407
89.089	4,364
999.999	4,272
1501.5	2,912
76.076	2,689
1001	2,655

The target, sans-spikes

Exploratory Analysis





- Histogram looks normal
- Quantiles follow theoretical quantiles

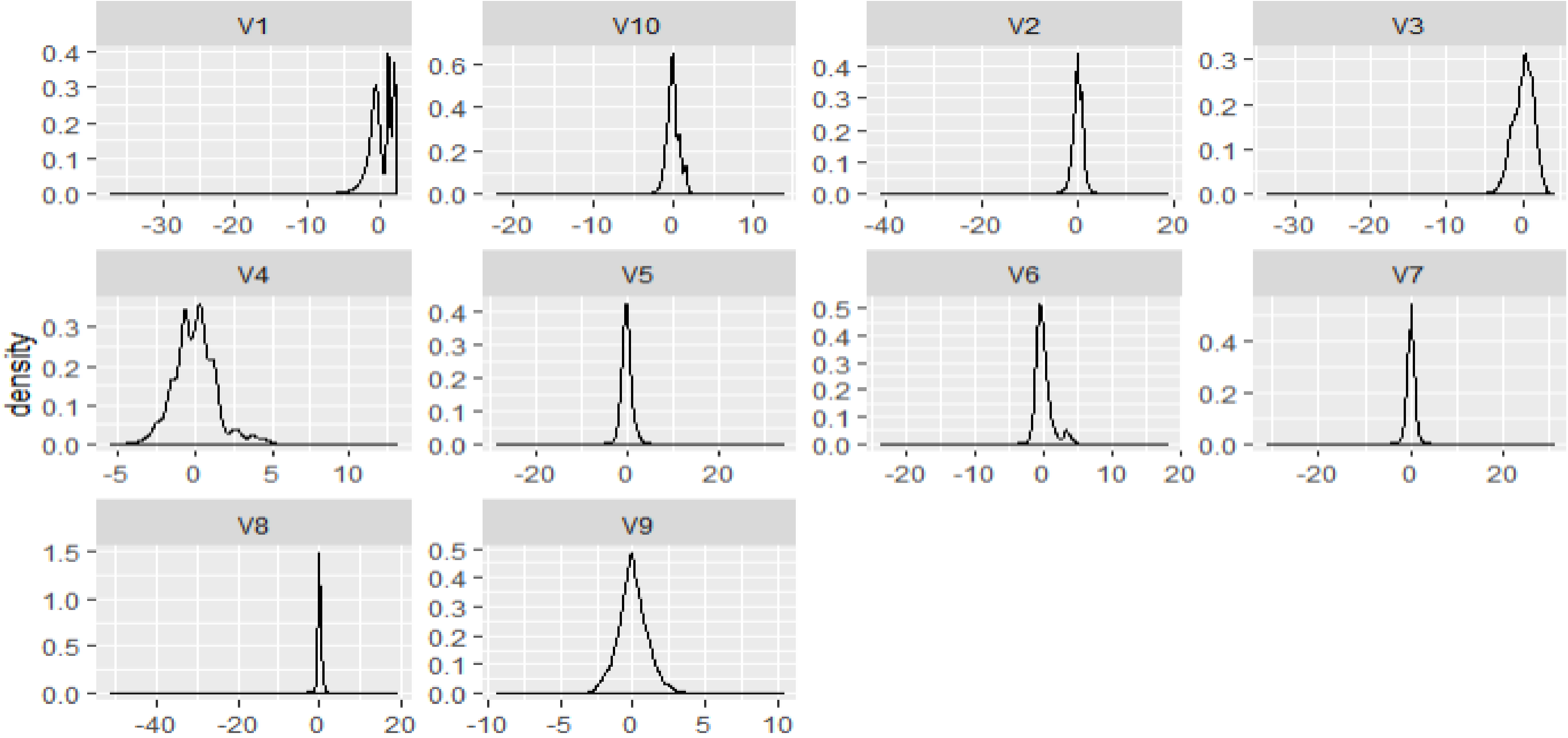
Checks for data consistency

Exploratory Analysis

- Do predictor quantiles of the holdout match that in “train”? When this isn’t the case, the effect is called *covariate shift*
  - I compared the 1st quantile, median, and 3rd quantile between train and holdout
- Are there the same number of outliers in the holdout and training sets?
- These all looked great

The predictor distributions

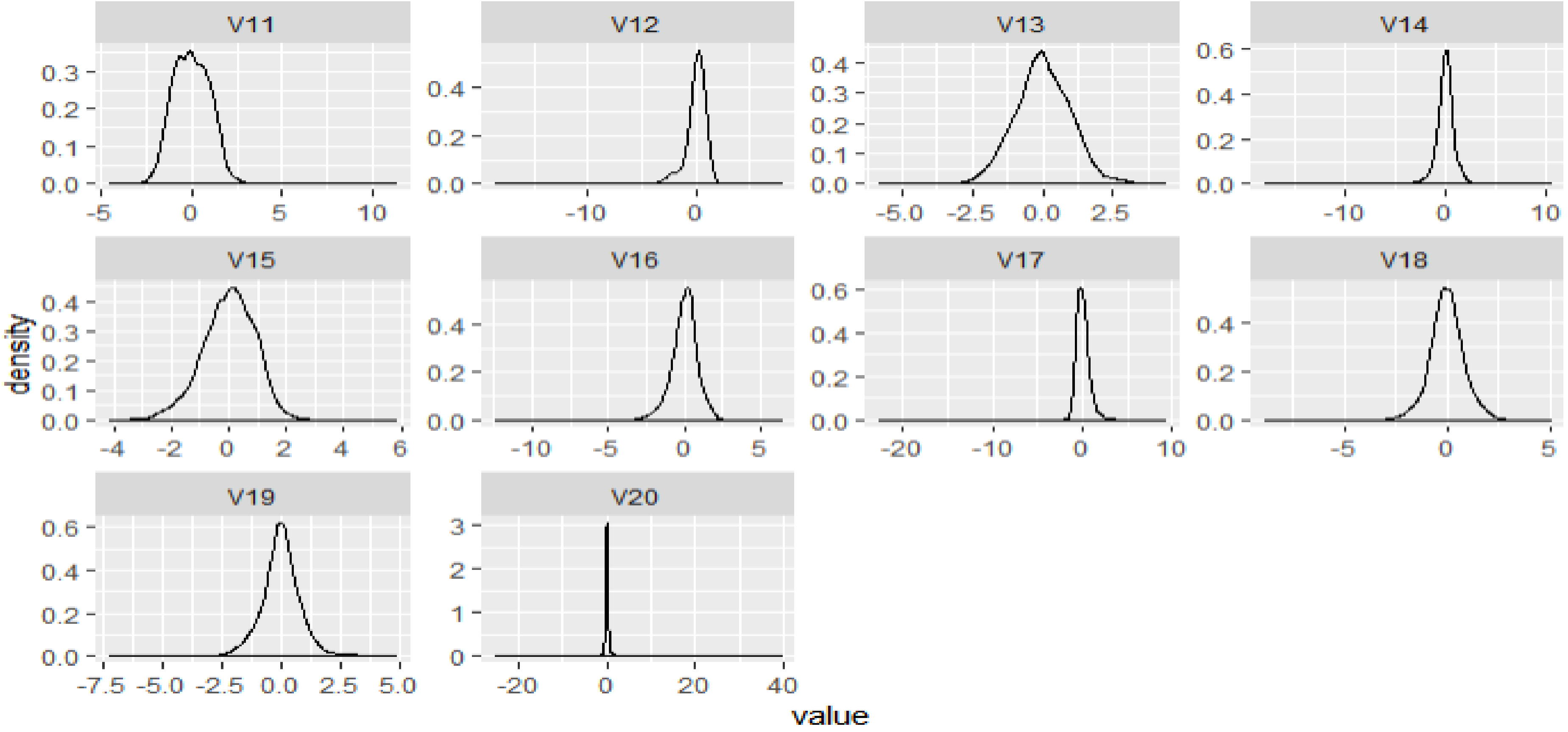
Exploratory Analysis



- All distributions are centered at zero and symmetric for the most part
- Coincidently, they were already arrange in order of variance so that V1 has the highest variance

The predictor distributions

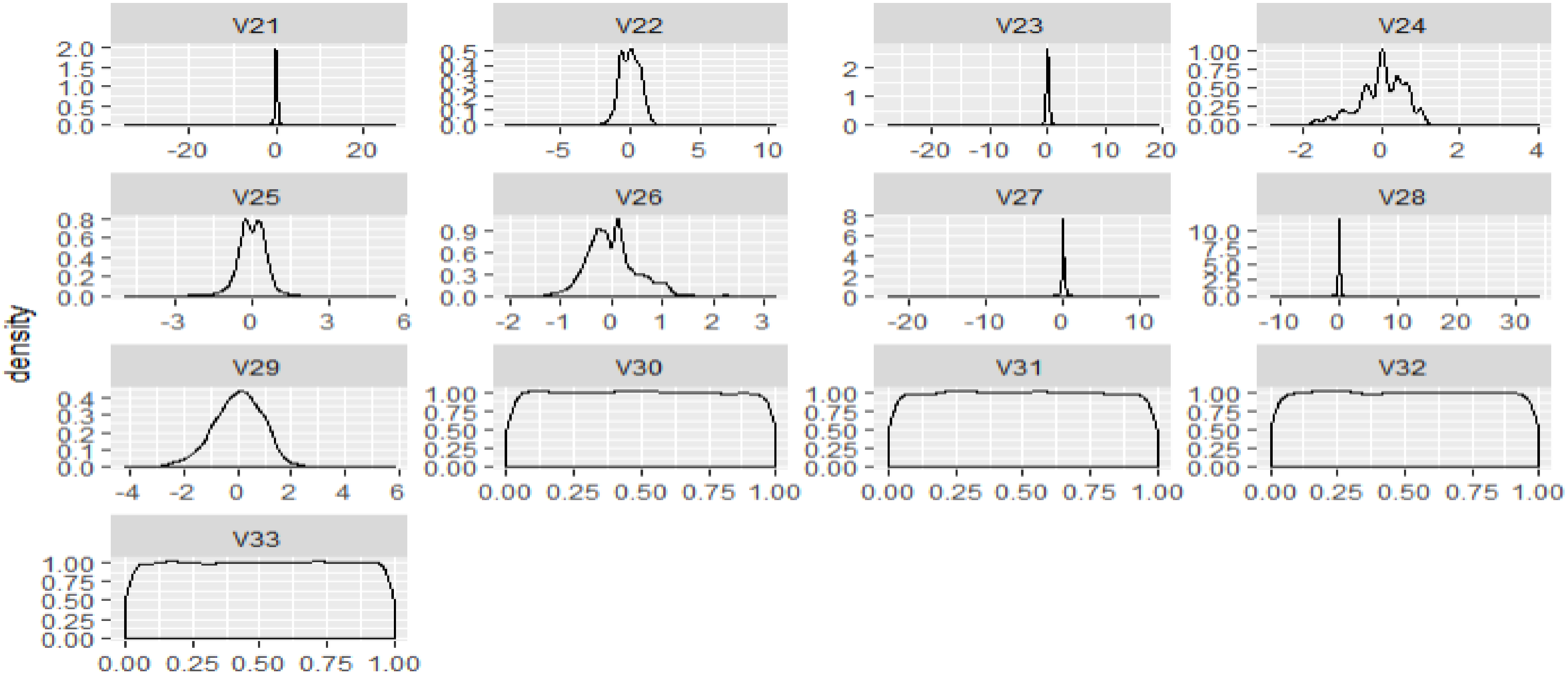
Exploratory Analysis



- Many have very long tails due to extremely high or low values, such as V20

The predictor distributions

Exploratory Analysis



- V30 – V33 are uniformly distributed

Univariate Outlier Analysis

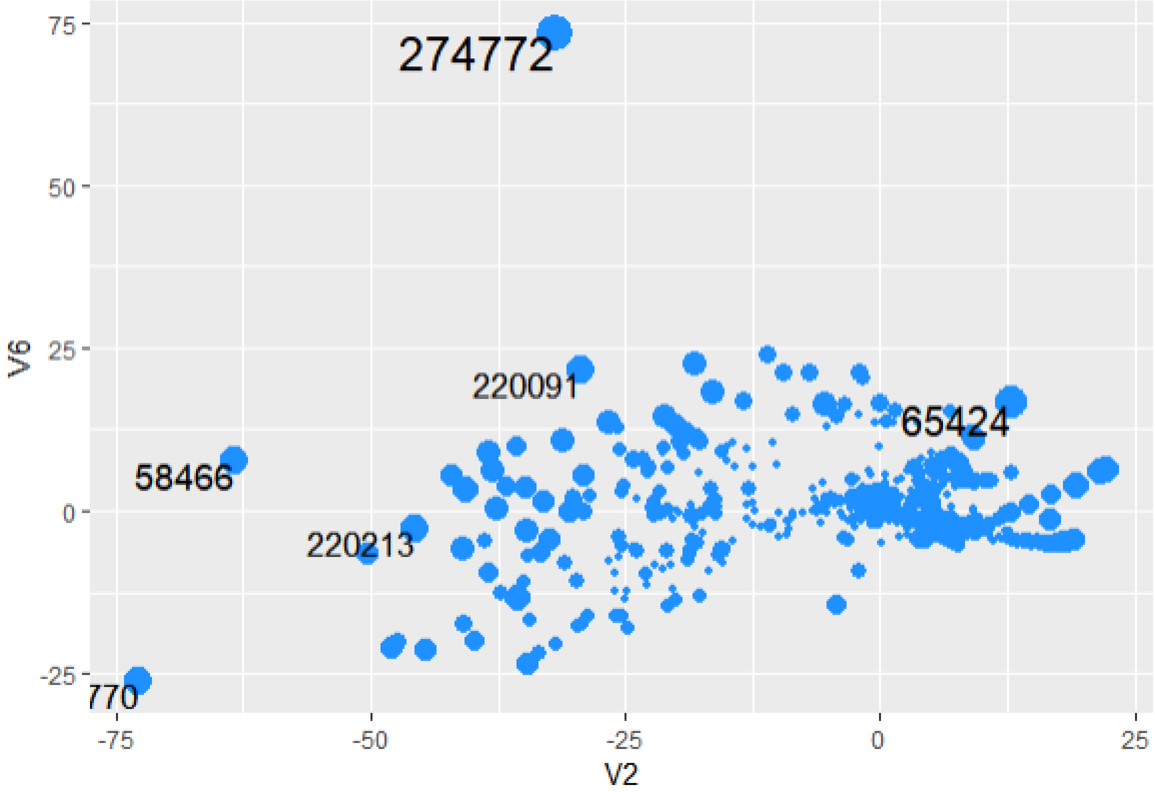
Exploratory Analysis

- Due to time constraints, only looked at univariate outlier definitions
- These were defined as being outside the inter quantile range, where IQR Range = below 0.001st quantile or above 99.99th
- All features were had a consistent distribution of outliers (0.2%) except for Amount, which was < 0.01%

* Note: Quantile defined by R, which uses 9 different methods for estimating empirical quantiles

Tracking Specific outliers

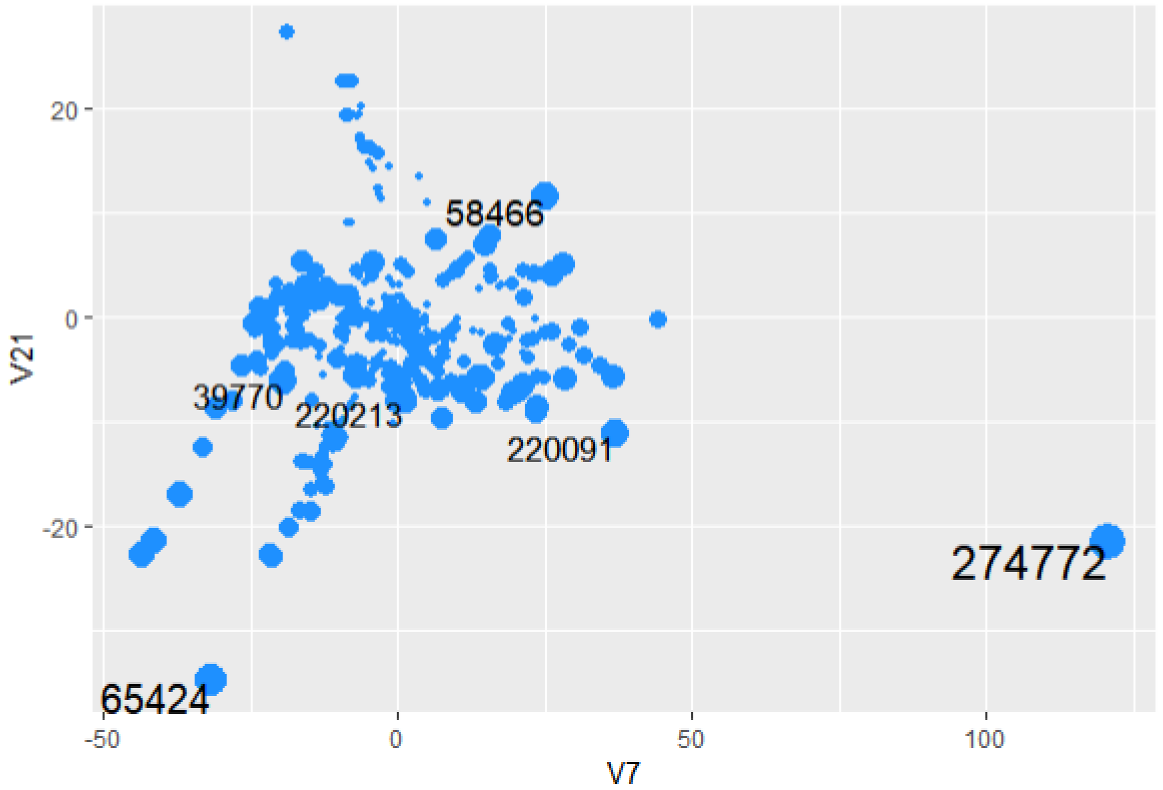
Exploratory Analysis



- Size of ● = number of dimensions where point is outlier.
- Observation 2774772 is an outside the quantile range in 21 dimensions
  - The second closest was at 15, then 12, 11, etc

Tracking Specific outliers

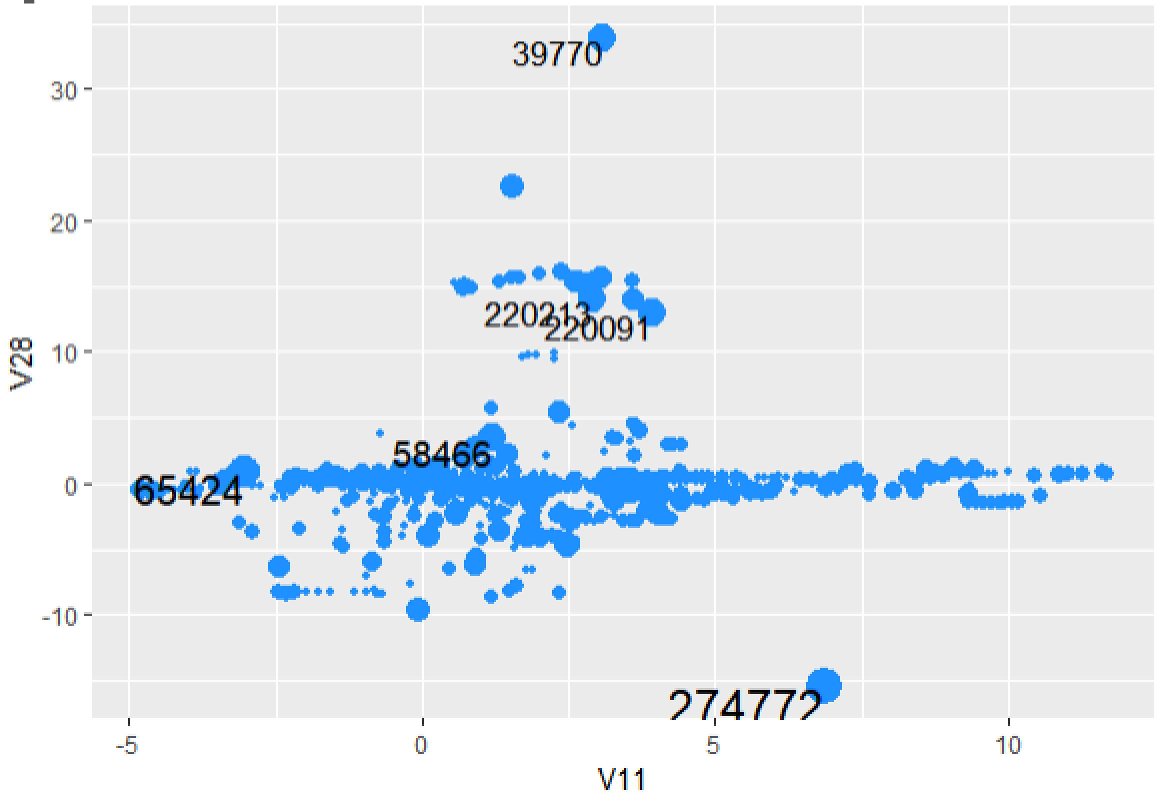
Exploratory Analysis



- Size of ● = number of dimensions where point is outlier.
- Observation 2774772 is an outside the quantile range in 20 dimensions

Tracking Specific outliers

Exploratory Analysis



- Size of ● = number of dimensions where point is outlier.
- Observation 2774772 is an outside the quantile range in 20 dimensions

Correlations

- V15 = V29
- After removing V29, other correlations between predictors were weak (less than 0.001)
- V2, V5, V6, V7, V20, V21 were correlation with target

Independence of predictors

Exploratory Analysis

- Column 29 removed
- Principal component analysis on the predictors shows very weak linear association

Principal Component Summary

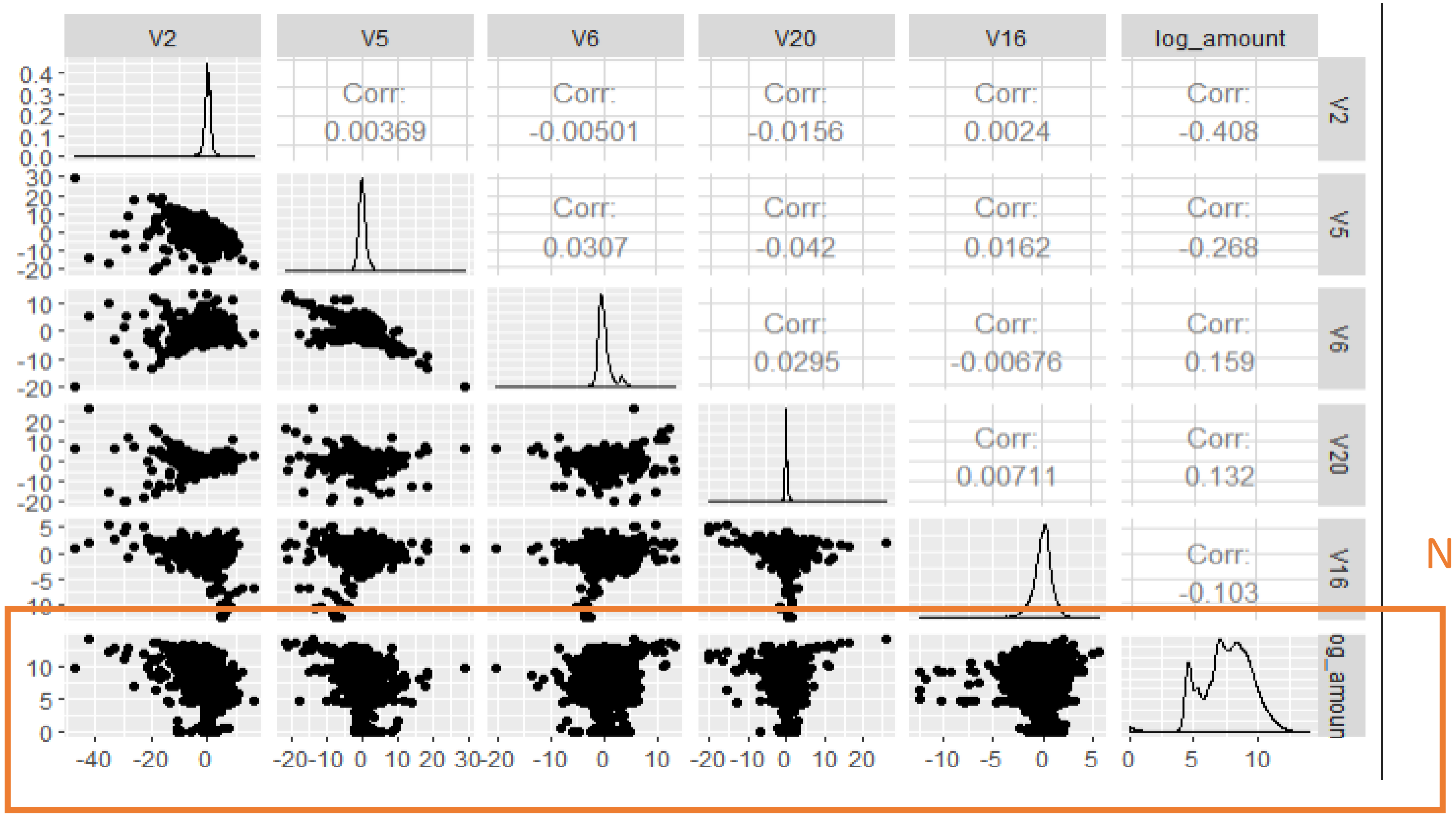
	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11	PC12	PC13	PC14	PC15	PC16	PC17
Standard Deviation	1.4	1.3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Incremental Variance Explained	6%	5%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%

	PC18	PC19	PC20	PC21	PC22	PC23	PC24	PC25	PC26	PC27	PC28	PC29	PC30	PC31	PC32	PC33	PC34
Standard Deviation	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.6	0.0
Incremental Variance Explained	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	3%	1%	0%

Predictors correlation with target

Exploratory Analysis

- Top 5 highest correlations with Amount
- Relationships with target do not appear linear



No linear relationships

Modeling Strategy

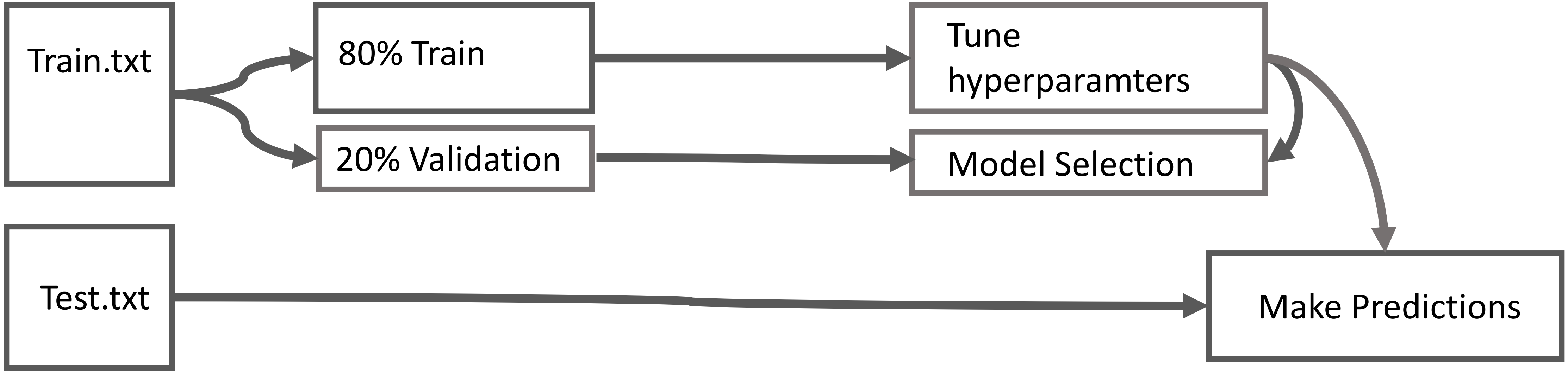
Training and tuning procedures

Modeling Strategy

1. Set up train cross validation pipeline
2. Start with a very basic linear model
    1. Add additional predictors
    2. Train on test set
    3. Make final model selection based on holdout set
3. Non-linear model
    1. Trian on test set
    2. Make final model selection based on holdout set

Train-Test Split

Modeling Strategy



Train.txt

80% Train

20% Validation

Tune hyperparamters

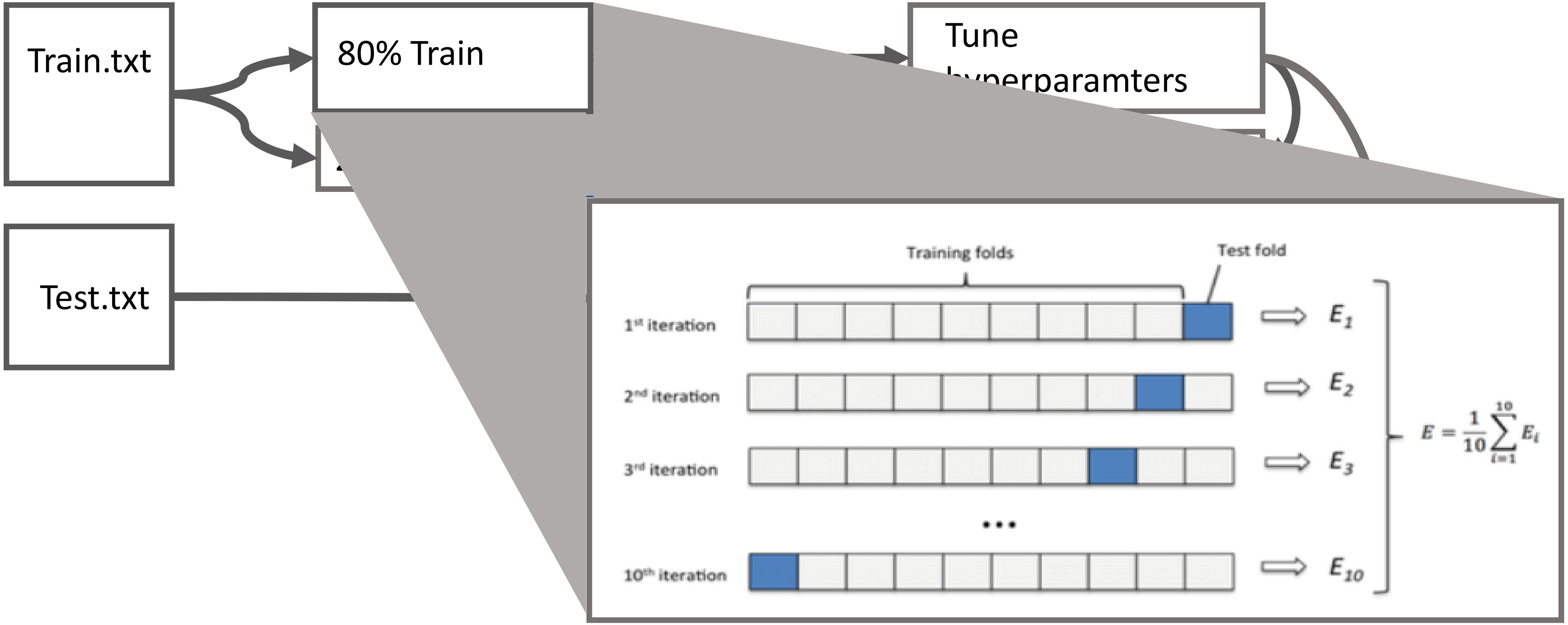
Model Selection

Test.txt

Make Predictions

Train-Test Split

Modeling Strategy



Train.txt

80% Train

Test.txt

Tune hyperparamters

Linear Models

GLM-based regression methods

Linear Model Assumptions

- Y is a member of the exponential family of distributions
- X’s are independent
- Error is normally distributed
- Y is linearly related to X (or through link function for GLM)
- There are no patterns in the residuals

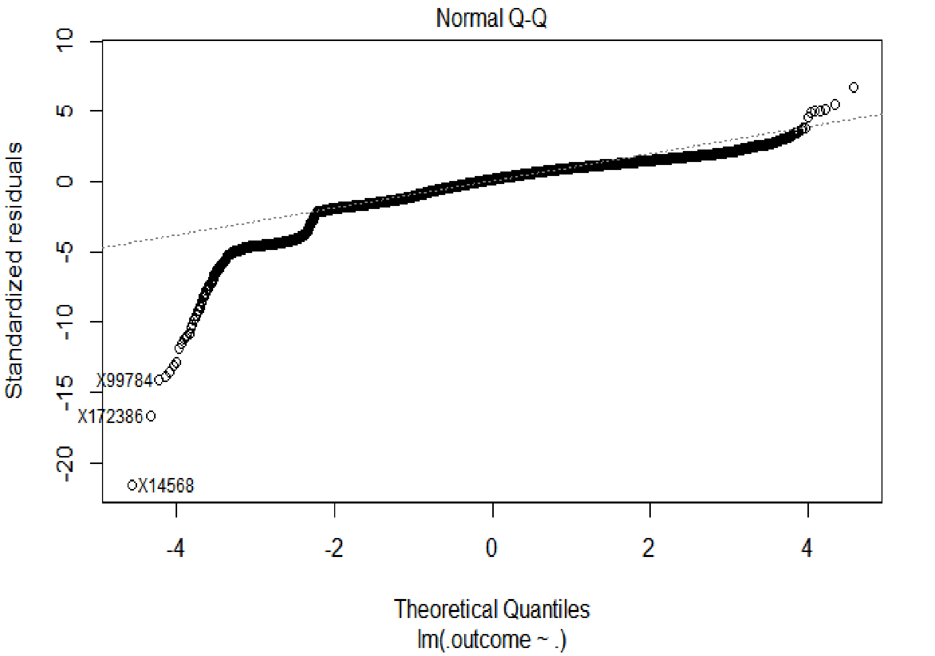
Baseline Linear Model

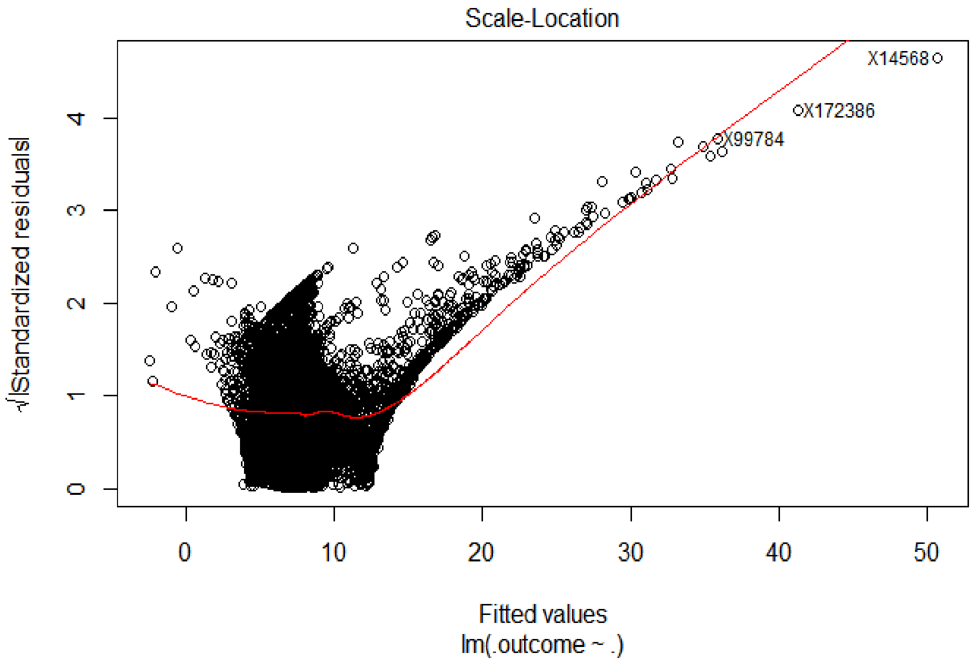
Model Building

- This provides a benchmark
- This tells me if something is seriously wrong with the data
- Can identify outliers
    - Log(Amount + 1) ~ V2 + V7 + V6
    - These variables have the highest correlation with target

Baseline OLS (MAE = 1.30908)

Model Building





- Residuals are approximately normal for amounts > 100.100
- Residuals are NOT independent of Y

More predictors (MAE = 1.212)

Model Building

- Log(Amount + 1) ~ V1 + … + V33

Predictor	Coefficient	Std. Error	P-value
V30	0.00	0.01	0.92
V32	0.01	0.01	0.56
V33	-0.01	0.01	0.42
V31	0.01	0.01	0.37
V13	0.00	0.00	0.18
V28	0.02	0.01	0.05
V19	-0.02	0.00	0.00
V10	-0.02	0.00	0.00
V25	-0.04	0.01	0.00
V24	-0.05	0.01	0.00
V8	-0.03	0.00	0.00
V12	-0.03	0.00	0.00

Dropping predictors (MAE = 1.208)

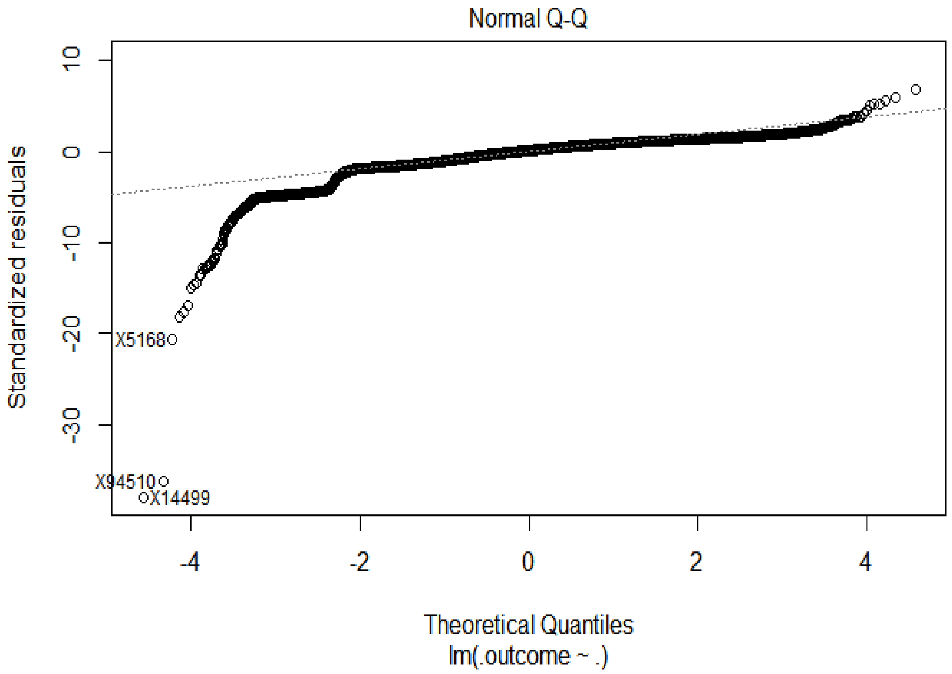
Model Building

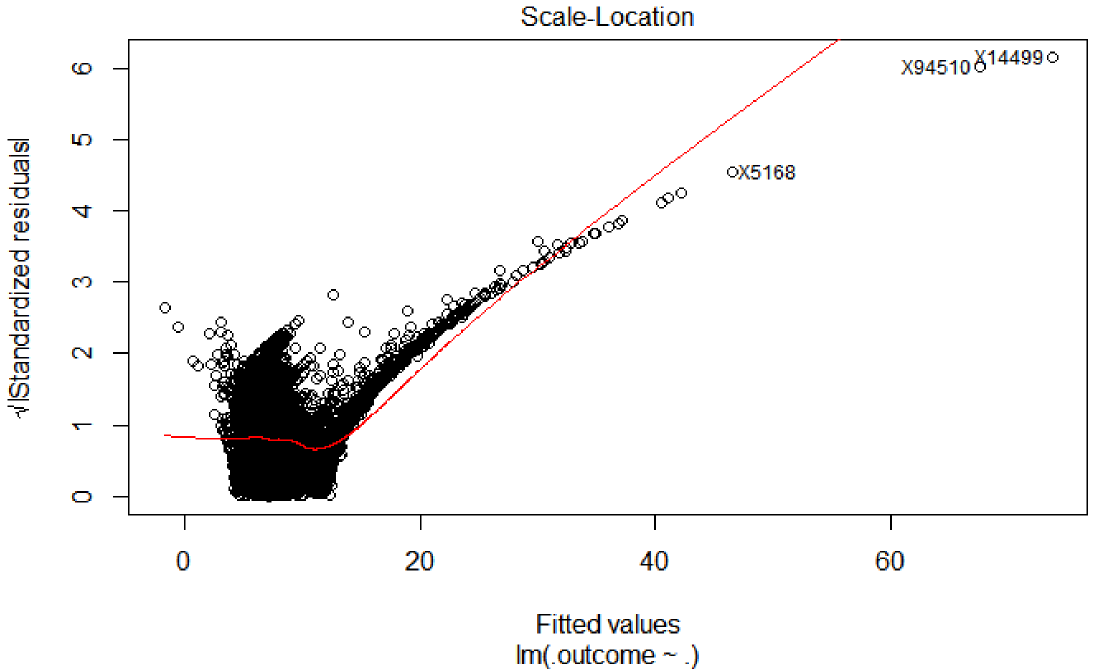
- Log(Amount + 1) ~ V1 + … + V28

Predictor	Coefficient	Std. Error	P-value
V19	-0.02	0.00	0.00
V10	-0.02	0.00	0.00
V25	-0.03	0.01	0.00
V24	-0.05	0.01	0.00
V12	-0.03	0.00	0.00
V8	-0.03	0.00	0.00
V23	-0.07	0.01	0.00

Dropping predictors (MAE = 1.208)

Model Building





- Model is still performing poorly at predicting large values
- Accuracy decreases as target increases

Why these models fail

Model Building

- Y is a member of the exponential family of distributions
  - Not the case due to left-truncation
- Error is normally distributed
  - Not the case due to left-truncation
- Y is linearly related to X (or through link function for GLM)
  - Many non-linear relationships can be seen
- There are no patterns in the residuals
  - Correlated with X and Y
  - Not at centered zero

Gradient-boosted trees

Non-parametric, non-linear models

The gradient boosted tree

Model Building

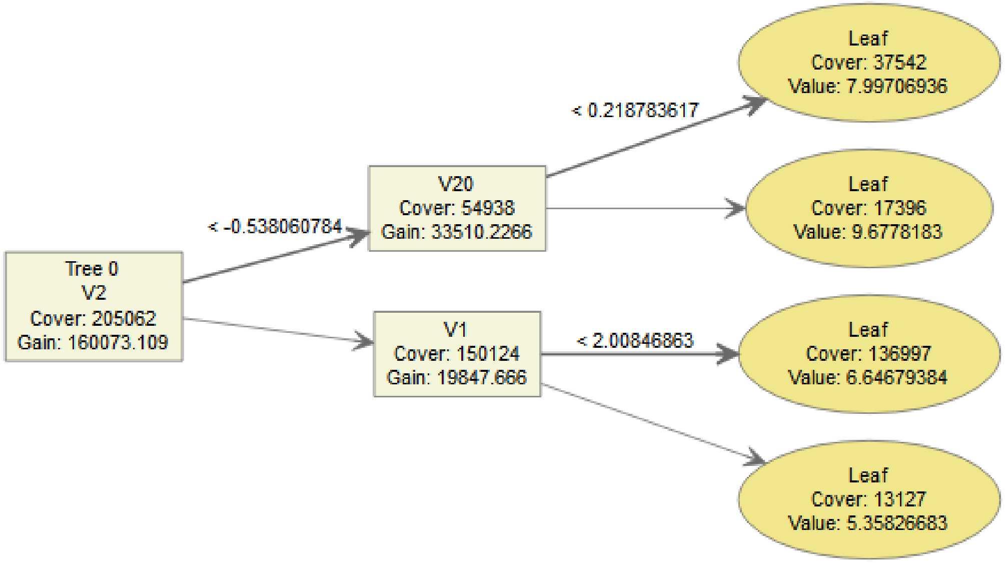
- Advantages
  - Robust to outliers
  - Handles interaction effects
  - Learns non-linear relationships between X and Y
  - Can optimize for MAE directly, rather than going through maximum likelihood as in the GLM case
- Disadvantages
  - Takes much more compute power to train
  - Requires special attention to train in order to avoid overfitting

What is a GBM?

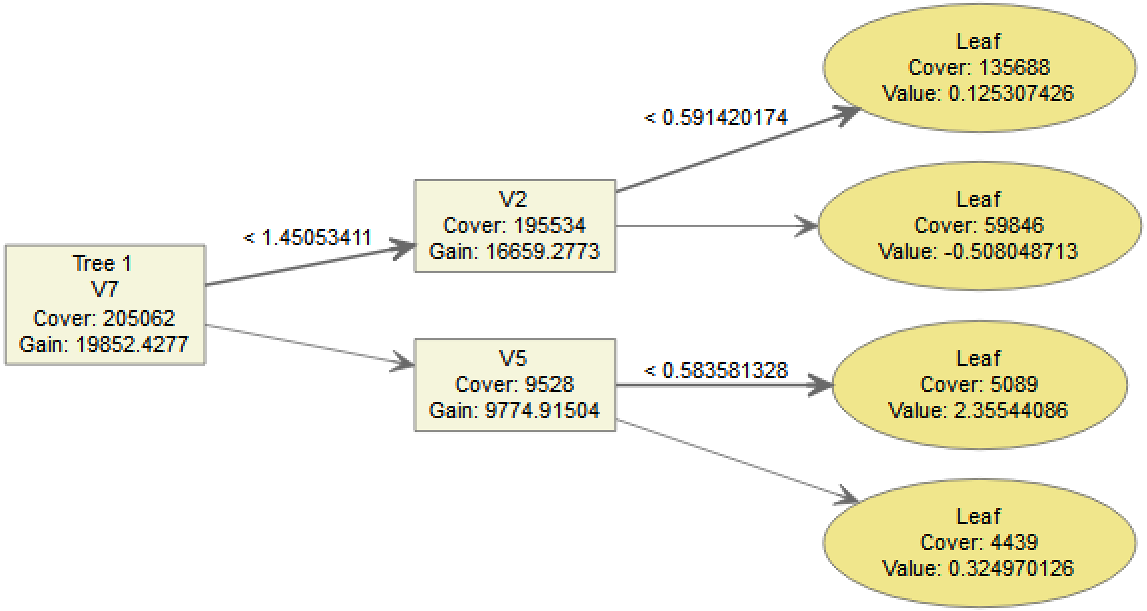
- Two concepts:
  - Regression trees
  - Boosting

Two real regression trees

Tree # 1



Tree # 2



Boosting

Model Building

Let Y = Amount, X = Matrix of Predictors, F(X) = Prediction from model F

Step 1:

Fit an initial model F_0. This will have a residual = $(Y – F_0)$

Step 2:

Fit a new model to the residuals from step 1 called h_1

Step 3:

Create a “boosted” model $F_1 = F_0 + h_1$

This will be slightly more accurate than F_0 by itself

…

Step m:

Continue “boosting” the previous models until cross-validation says to

$F_m = F_{m-1} + h_m$

GBM Parameters

Boosting Parameters

- How should the all of the trees be combined?

Tree Parameters

- How should each individual tree be fit?

GBM Parameters

Model Building

Boosting Parameters

- Learning rate: controls how quickly each tree's contributions impact outcome
- Number of trees: the number of boosting iterations
- Subsample: Fraction of observations to use in each tree

Tree Parameters

- Min node samples: the minimum number of observations required to split an internal node
- Min leaf samples: the minimum number of observations required in a terminal node for a split to be valid
- Max depth: the max "height" of each tree
- Max terminal nodes: the max number of leaf nodes
- Max features: Max number of features to consider at each split

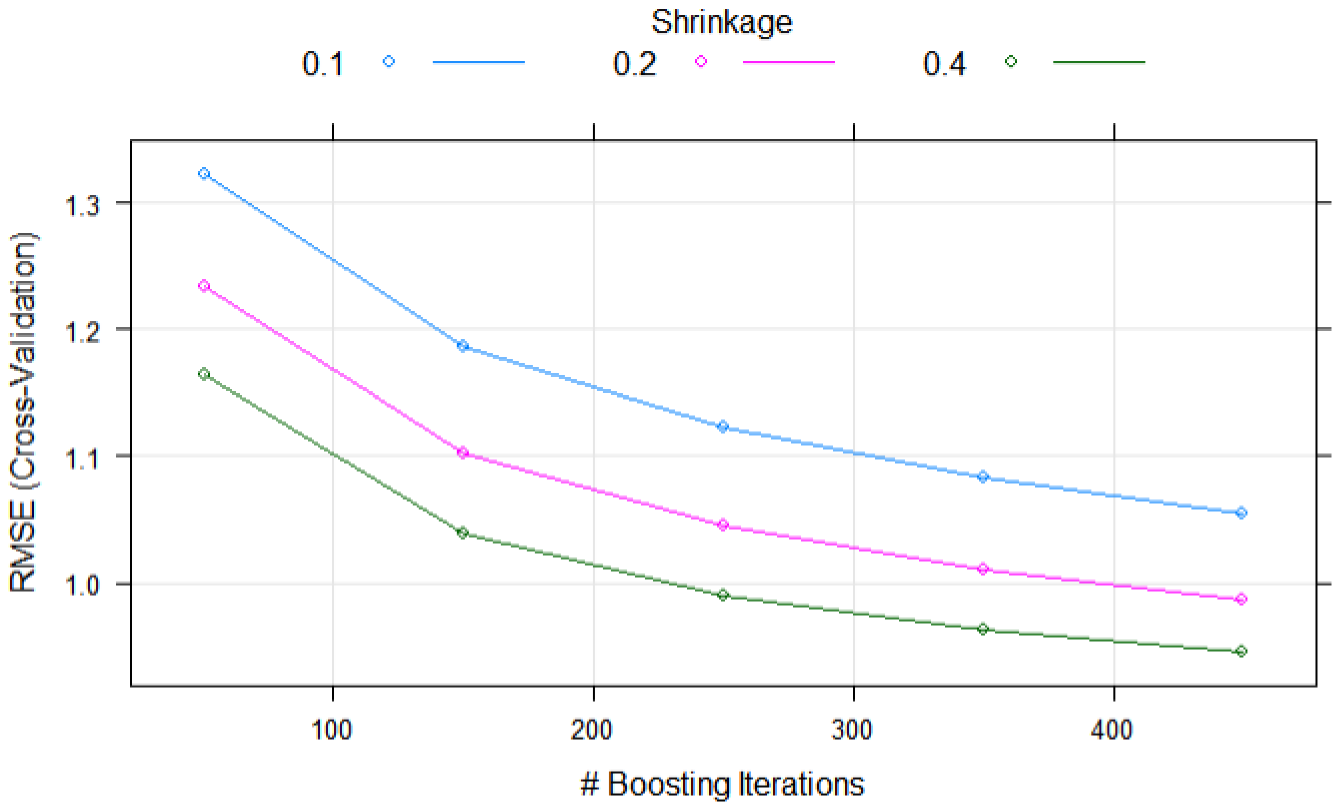
Step 1: Baseline (MAE = 0.77329)

Model Building

- Starting Parameters
  - nrounds = 100,
  - max_depth = 3,
  - eta = 0.3,
  - gamma = 0,
  - colsample_bytree = 1,
  - min_child_weight = 1,
  - subsample = 1

Step 2: (MAE = 0.77329)

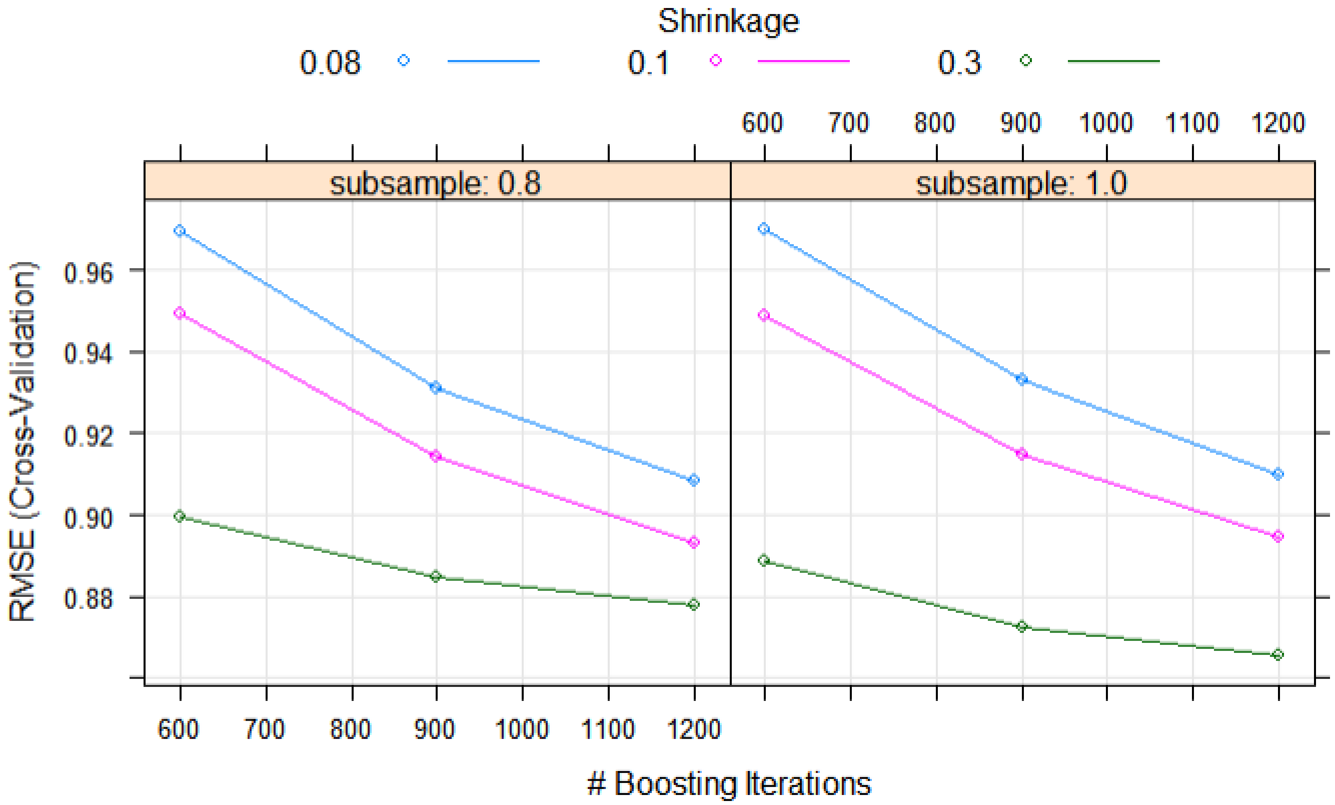
Model Building



- Find a good combination of number of trees and a learning rate

Step 3: (MAE = 0.6305632)

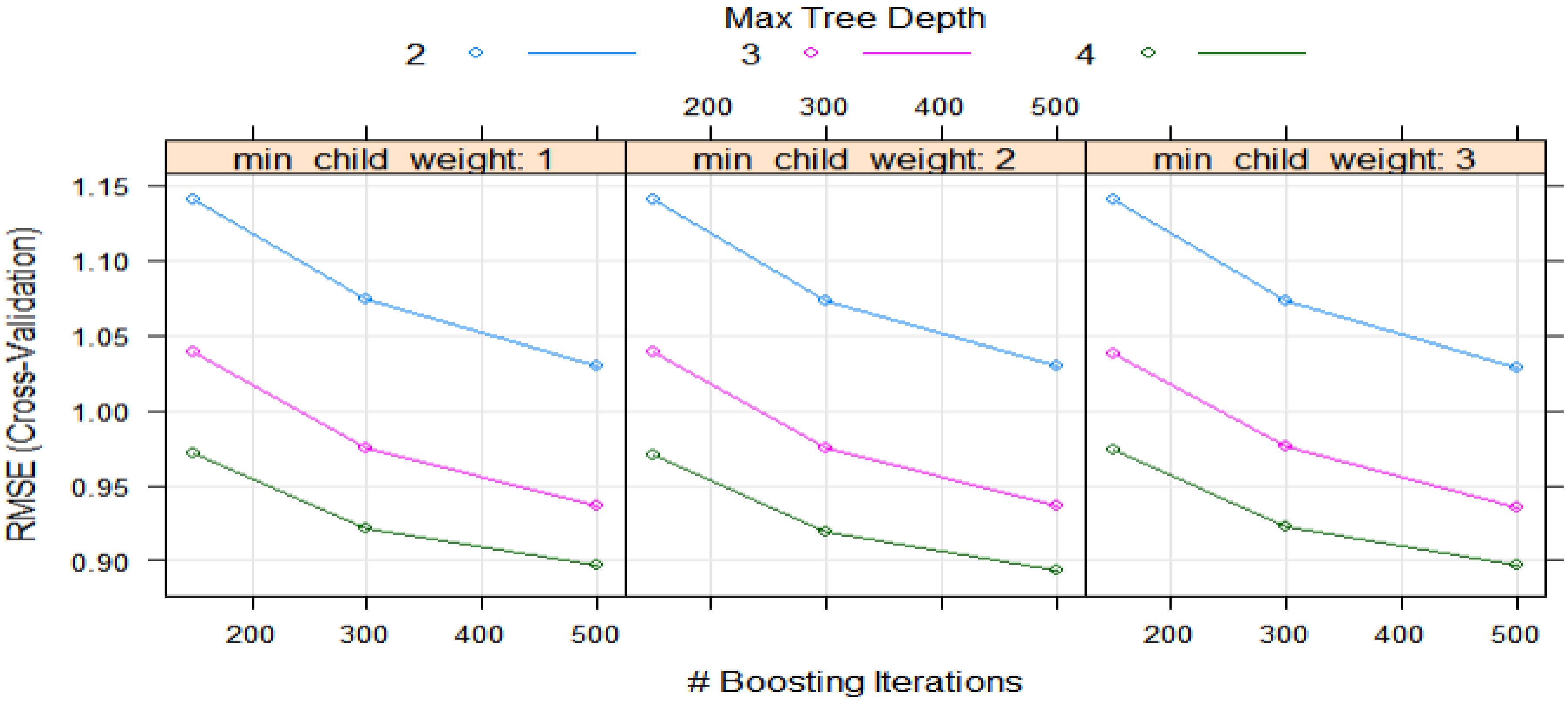
Model Building



- Find a good combination the percentage of features to select at each node, “subsample”

Step 4: (MAE = 0.50624)

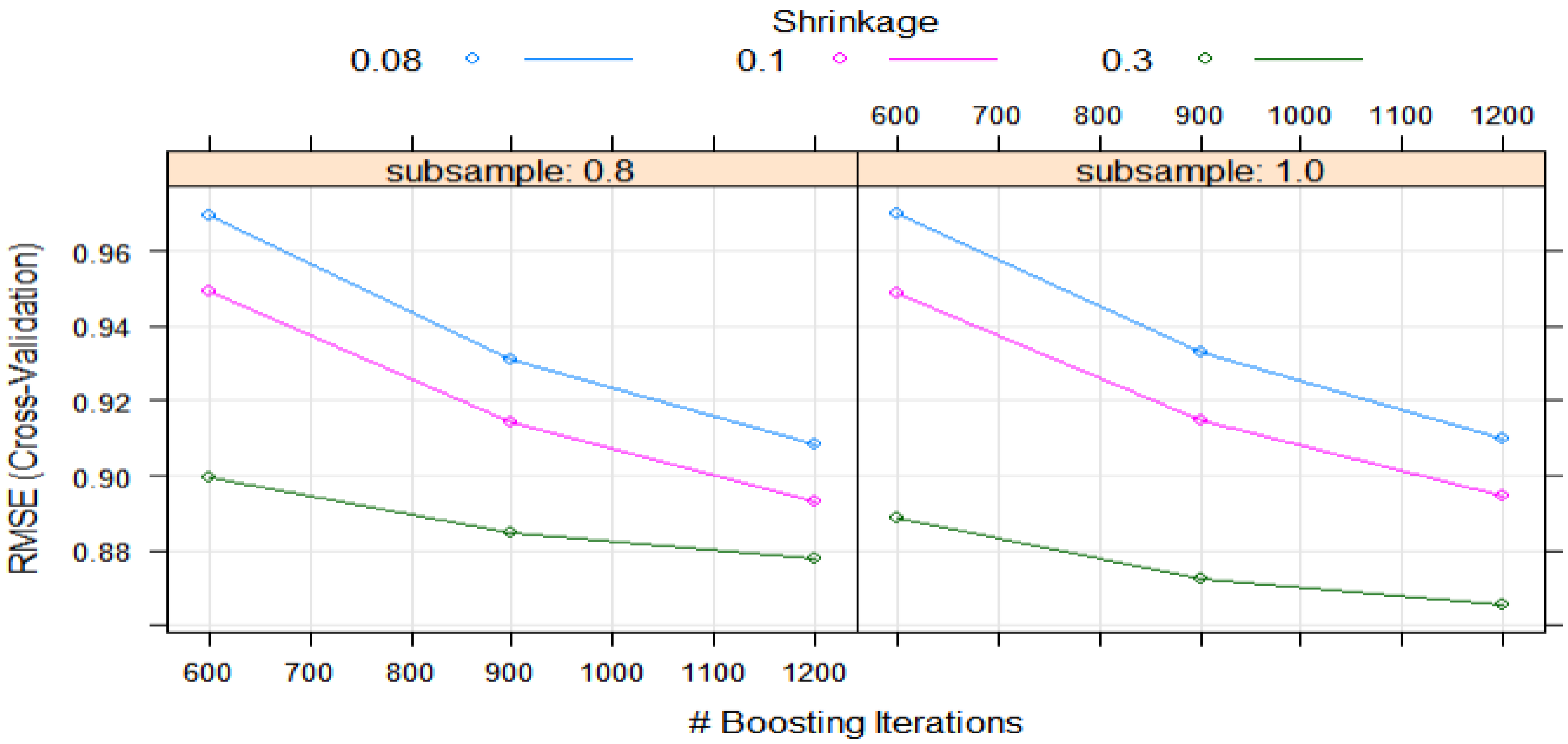
Model Building



- Best combination of tree depth and number of observations per node
  - Deeper trees with max tree depth of 4
  - Choose min child weight of 2. Using 1 is overfitting.

Step 5: (MAE = 0.4407)

Model Building



- Lower the learning rate and increase the number of trees

The dangers of overfitting

- Although the out-of-fold cross validation MAE had decreased, the MAE on the validation set had increased
- R^2 and RMSE follow similar patterns

	MAE		RMSE		R^2	
Step	Validation	Training	Validation	Training	Validation	Training
1	0.7818	0.7699	1.1189	1.0936	0.6758	0.6875
2	0.6069	0.6002	0.9075	0.8918	0.7858	0.7912
3	0.5231	0.5191	0.7949	0.7836	0.8356	0.8387
4	0.5371	0.4407	0.8517	0.6586	0.8109	0.8866

Variable Importance

Finding the most influential predictors

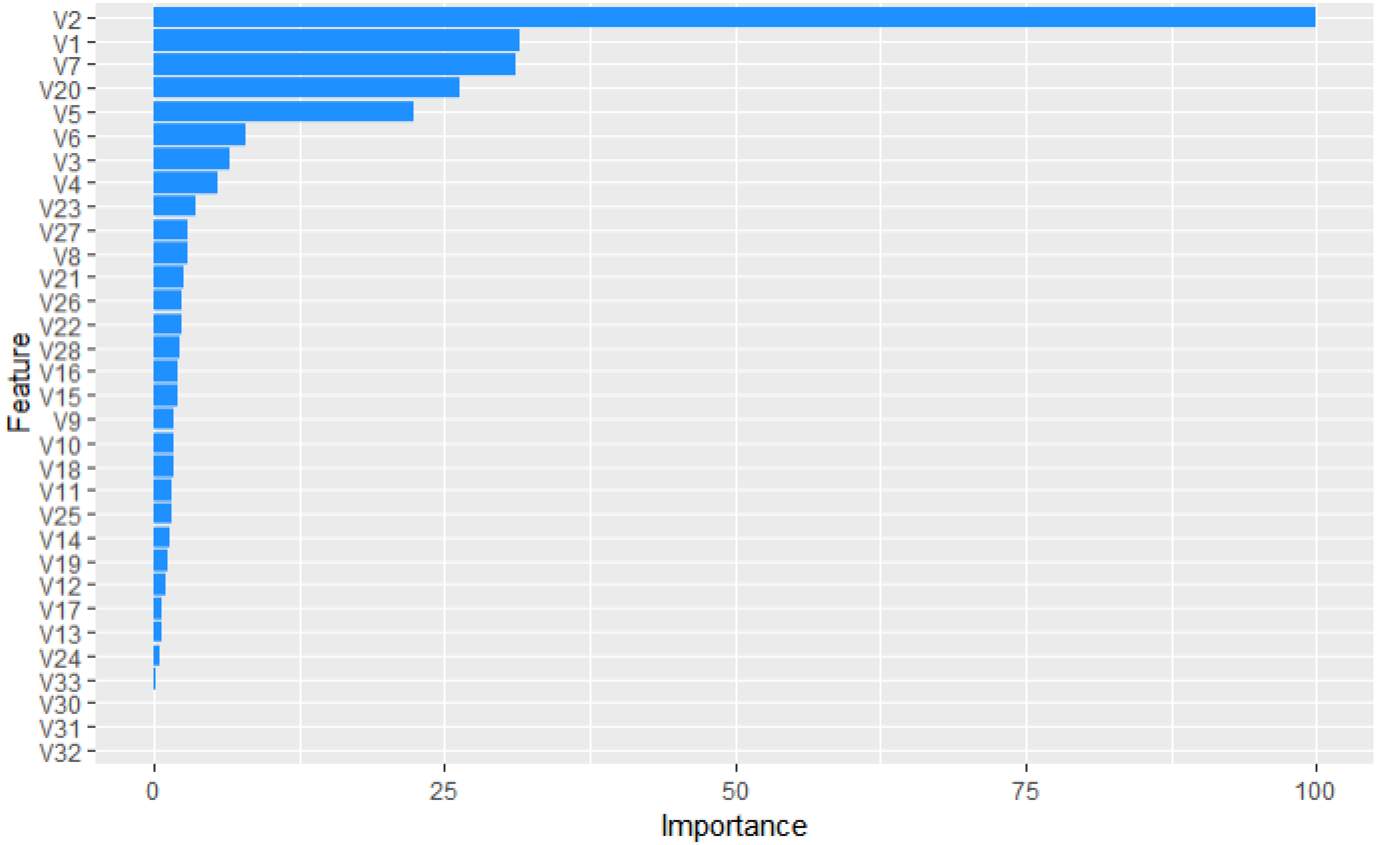
Definition of importance

To get the relative variable importance for a given predictor,

1. Start with a single tree
2. Look over all internal nodes where this is used as a split
3. Take the sum of improvement measures (Information Gain)
4. Do this for all trees
5. Rescale all variables by the one with the highest score

Top 5 most important predictors

Model Evaluation



- We also note that the top 5 by correlation with the target are V2, V5, V6, V20, and V16

Partial dependence on target

- Estimates the marginal effect of predictor X on target *after integrating out all other predictors*

Partial dependence on target

Model Evaluation

- Model = F(X1, X2)
- To estimate the effect of X1 on F when adjusting for X2,

Training Data

X1	X2	F(X1,X2)
1	3	10
1	4	20
2	5	50
2	6	70

F evaluated at all combinations of X1 and X2

X1	X2	F(X1,X2)
1	3	40
2	3	40
1	4	50
2	4	60
1	6	20
2	6	40

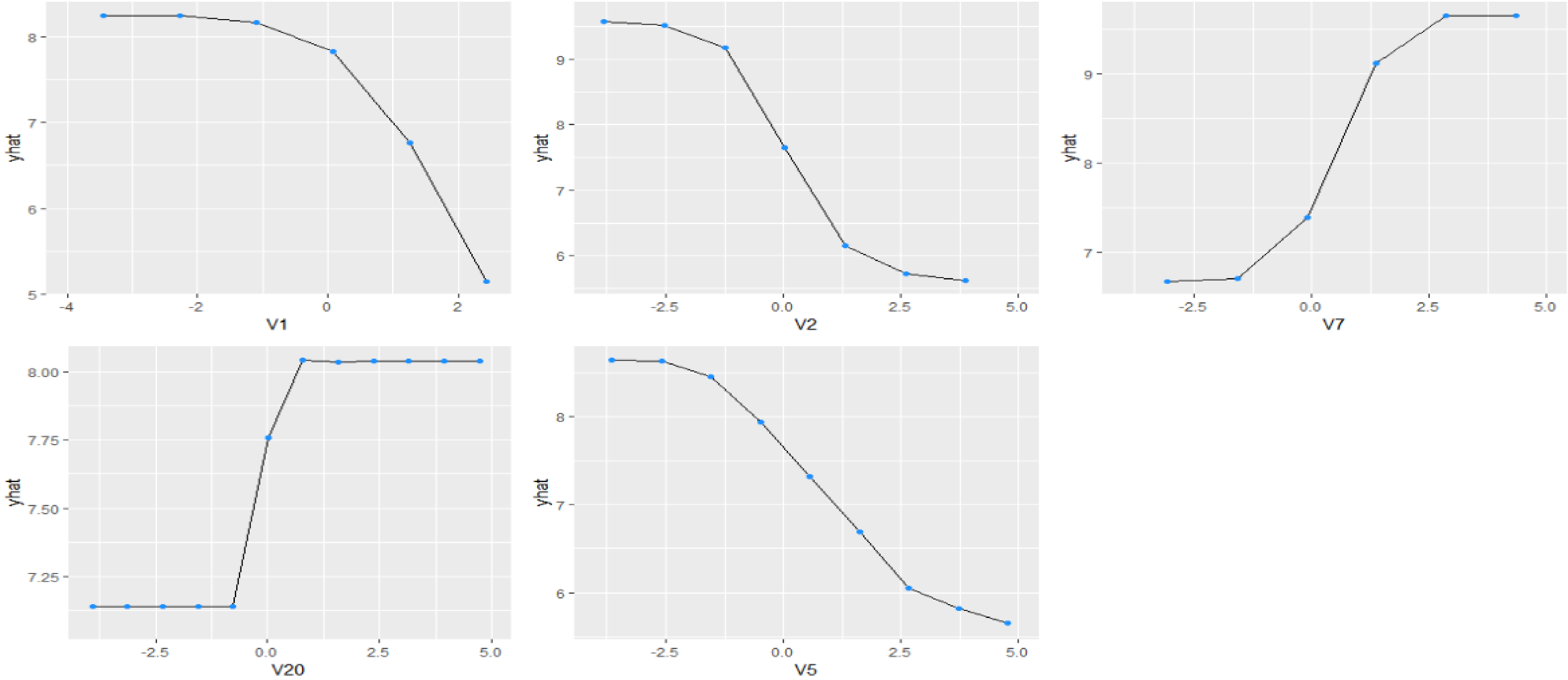
Marginal Average

X1	F(X1,X2)
1	36.6667
2	23.3333

`36.67 = (40 + 50 + 20)/3

Partial dependence plots

Model Evaluation

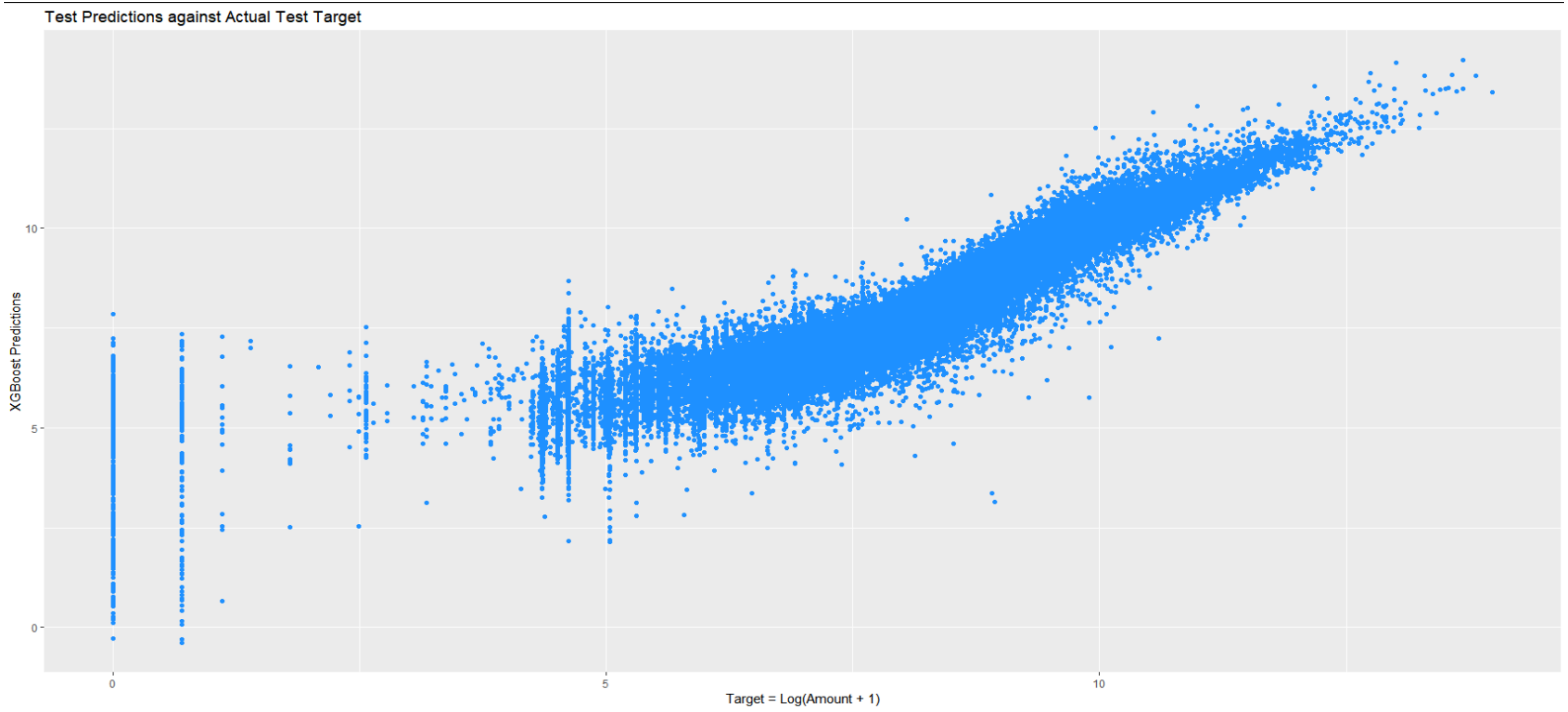


Actual verses expected

- Compare the actual target values from the validation set against the fitted values

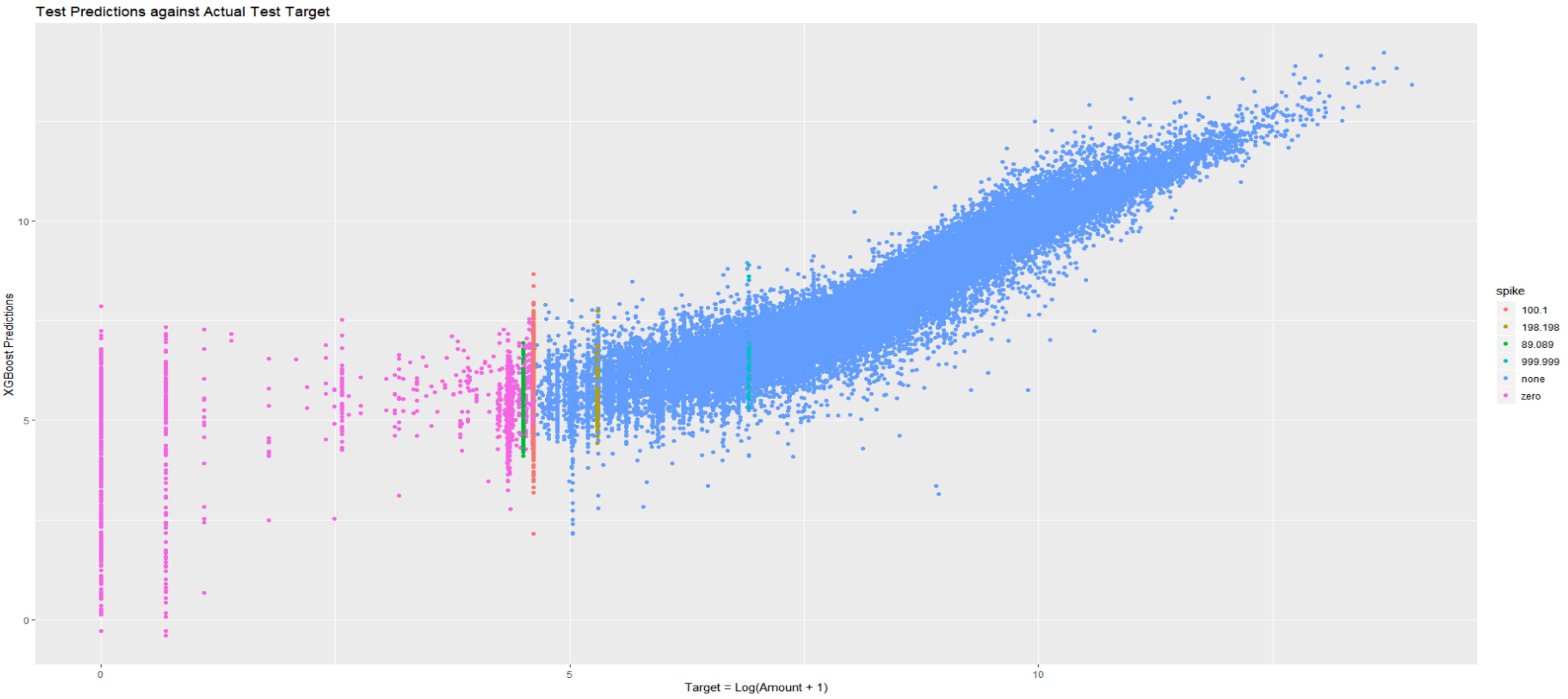
Actual verses expected

Model Evaluation



Actual verses expected

Model Evaluation



Tools Used

Reference texts and software packages

Tools Used

Tools Used

Reference Texts

- An Introduction to Statistical Learning
- The Elements of Statistical Learning

R Software Packages

- Everything by Hadley Wickham including but not limited to: ggplot2, dplyr, tidyr, purr (excellent package), broom, forcats
- The caret library for model fitting
- The XGBoost (Extreme Gradient Boosting) GBM implementation

Online Articles (More than can be listed)

- “Generalized Linear Models for Insurance Ratemaking” (https://www.casact.org/pubs/monographs/papers/05-Goldburd-Khare-Tevet.pdf)
- “A Gentle Introduction to XGBoost for Applied Machine Learning”. (https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/)
- “An End-to-End Guide to Understandign XGBoost”. (https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/)

Various Kaggle repositories

Questions?

[redacted]

?