**University of Southern California**



**DSO 562: Fraud Analytics**

**Project 2: Credit Card Fraud Detection Model Development**

**Professor:** Stephen Coggeshall

**Table of contents**

[**1. Executive Summary**](#_j4lgm84qflbh) **3**

[**2. Description of Data**](#_xosch62javsc) **4**

[**3. Data Cleaning**](#_1l3frjr7d4w2) **8**

[3.1 Removing Outliers](#_gqo2swoqmsin) 8

[3.2 Filling Missing value](#_qheysmc0yooj) 9

[**4. Candidate Variable Creation**](#_5ku1w0gymbdn) **10**

[4.1 Amount variables](#_89dixa6a14dn) 10

[4.2 Frequency variables](#_i7tkogew2o44) 20

[4.3 Days since variables](#_gx6yzzw2mi7s) 22

[4.4 Velocity change variables](#_jrhxxmx8kcam) 22

[**5. Feature selection**](#_x5et59xwex0p) **27**

[5.1 Filtering with Kolmogorov–Smirnov (KS) and Fraud Detection Rate (FDR)](#_4oct472k0emy) 27

[5.2 Wrapper with Logistic Regression and backward feature selection](#_xt4z1wfjqf3r) 27

[**6. Supervised Fraud Algorithms**](#_18kx94g4nmfs) **30**

[6.1 Model Development](#_vmz7136ef7as) 30

[6.1.1 Logistic Regression (baseline model)](#_4frtngg51q8y) 30

[6.1.2 Random Forest](#_ybyhaz7kec62) 31

[6.1.3 Boosted Decision Tree](#_1z04lchm0v0v) 31

[6.1.4 SVM](#_531o5j95k153) 32

[6.1.5 Neural Network](#_ioz0jxgvlo6i) 33

[**7. Results**](#_nscsqr49j35r) **34**

[**8 Conclusion**](#_i3gp0dlziu0q) **37**

[**Appendix**](#_jtoe22qpum0l)**:** Credit Card Transaction Data Quality Report 38

# 

# 1. Executive Summary

The objective of this project is two-fold: 1) to gain insights into the fraud modes of credit card transactions by *Tennessee City Government* with supervised learning techniques and 2) prescribe reasonable actions based on a fraud detection model that maximize overall fraud savings. This study details a methodology to build a robust model that detects such credit card frauds in government spending.

The model is built on a dataset provided by the *City of Tennessee*. The dataset contains 98752 records of credit card transactions, each with 10 fields including fraud label, card number, merchant number and etc. The pipeline of our approach includes:

* data exploration
* data cleaning
* candidate variable creation
* feature selection
* fraud model development
* results / recommended actions

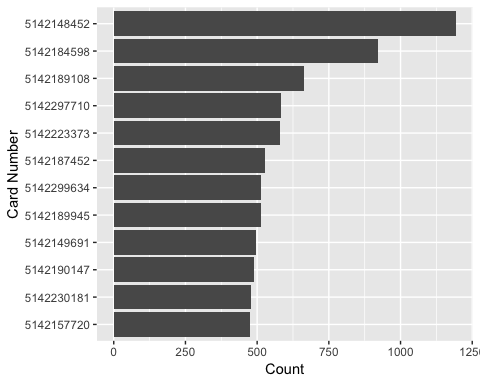
The final fraud detection model is selected from 5 candidate models including Logistic Regression, Neural Network, Boosted Decision Tree, Random Forest and Support Vector Machine models. Resampling and hyperparameter tuning were performed to ensure optimal prediction accuracy in terms of fraud detection rate within top 3% predicted frauds. After comparing FDR across models, the Neural Network model was selected to be the final fraud detection model as it performs best among all 5 algorithms.

# 2. Description of Data

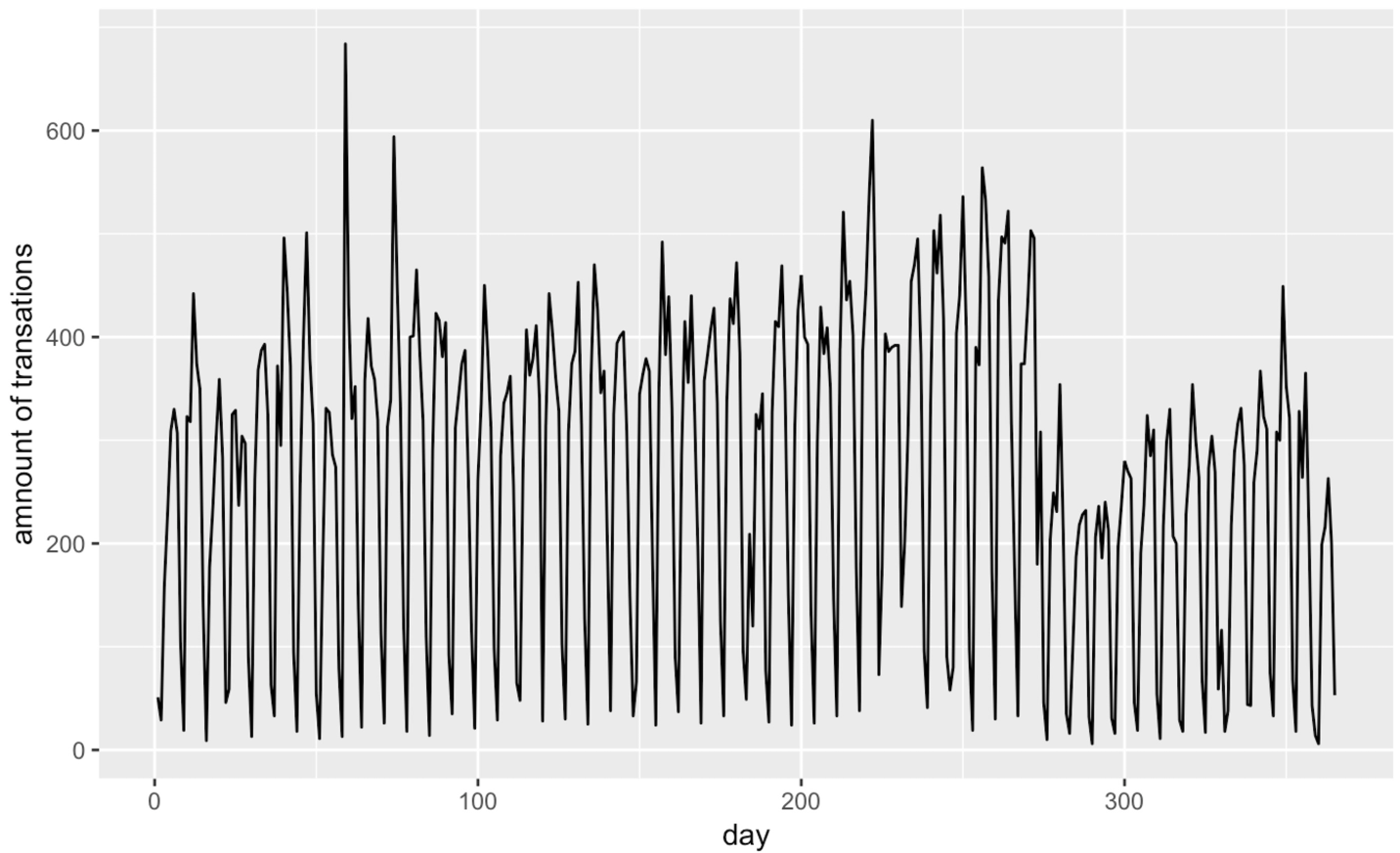
Credit Card Transaction dataset in 2010 has 10 variables and 96,753 observations in total. Except for the variable Recnum and amount are numerical variables, all the other variables are categorical, including *Card number*, *Date of transaction*, *Merchant number*, *Merchant description*, *Merchant state*, *Merchant zip*, Transaction type and Fraud label.

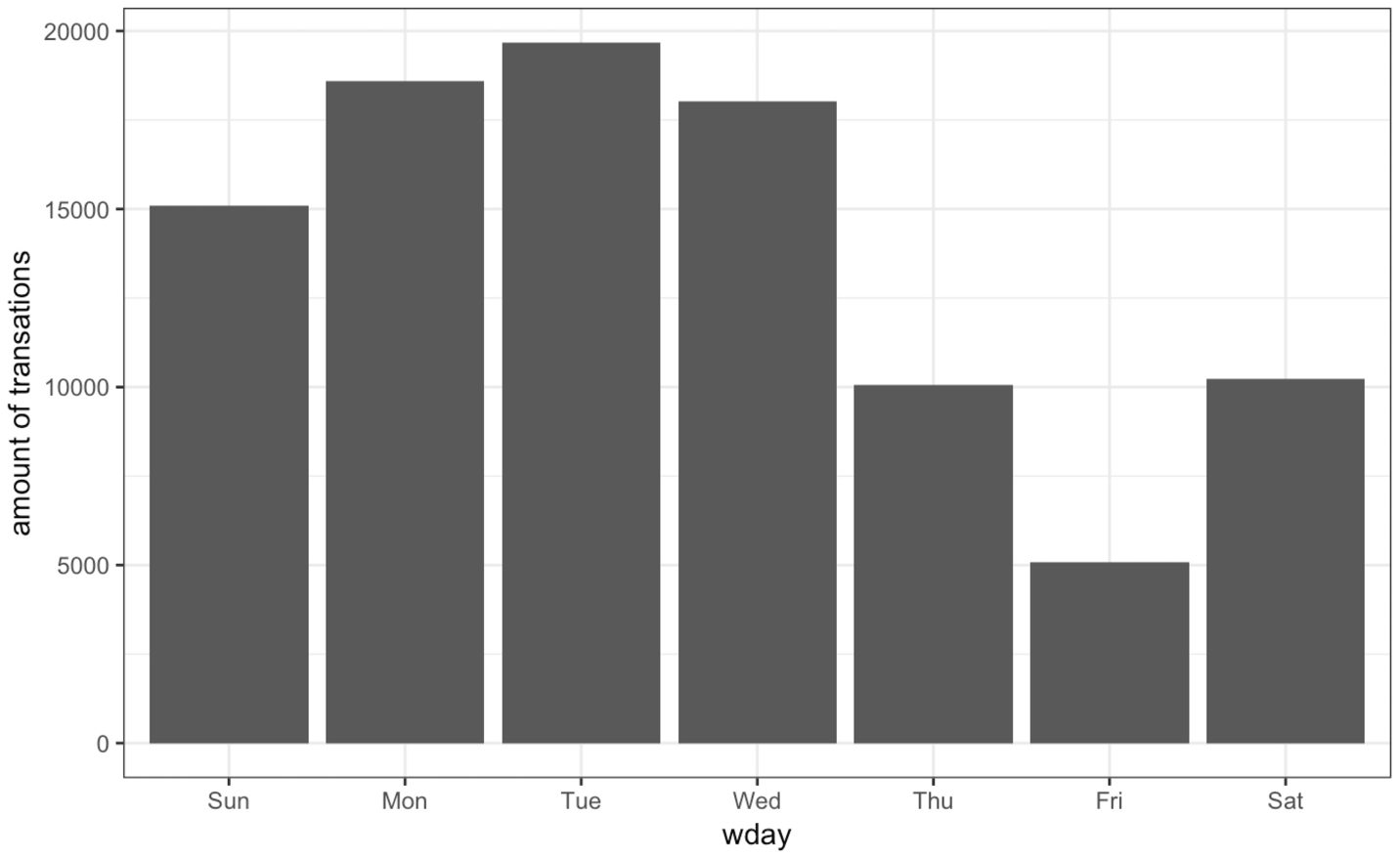
Some key variables are described below. All fields are discussed in greater detail in a Data Quality Report attached in the appendix.

|  |  |
| --- | --- |
| *Field Name* | *Description* |
| **Cardnum** | The number of the credit card that used in the transaction.  There are 1645 unique values. No missing values exist. |

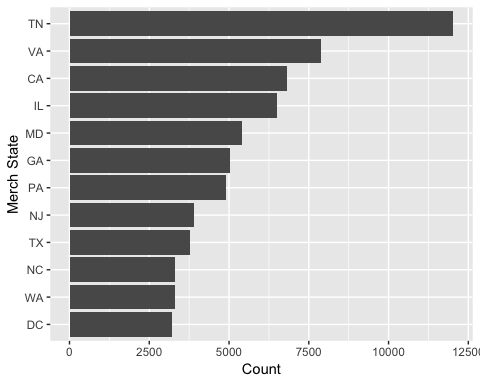
****

|  |  |
| --- | --- |
| *Field Name* | *Description* |
| **Date** | The date of each credit card transaction records.  There are 365 unique values from 01/01/2010 to 12/31/2010. No missing values exist.  Daily and weekly transaction frequencies are shown below. |

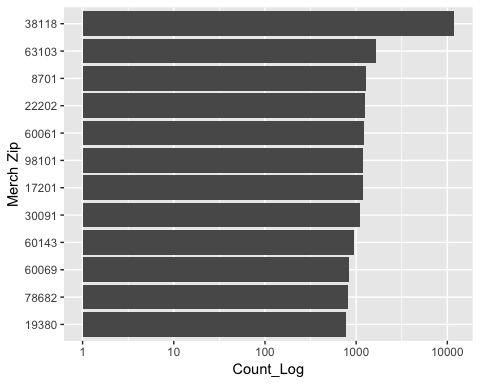




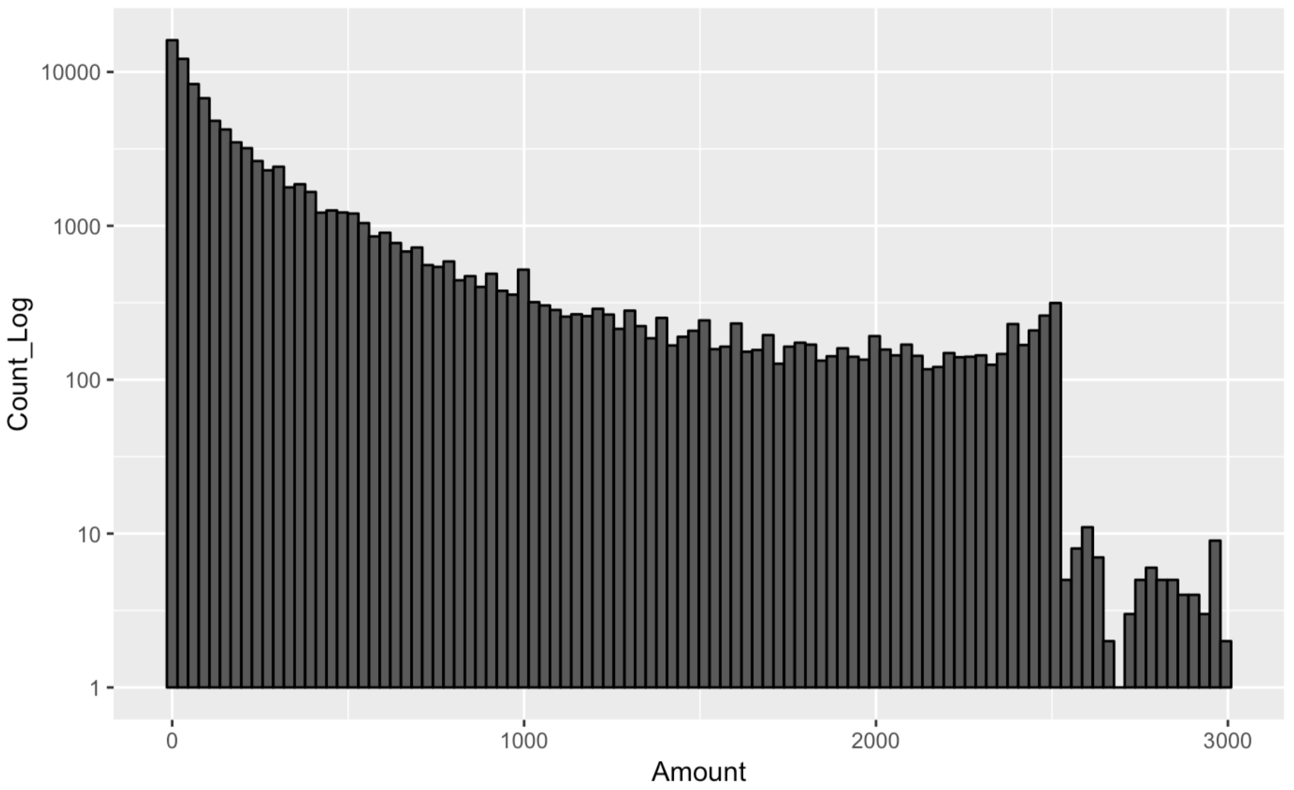
|  |  |
| --- | --- |
| *Field Name* | *Description* |
| **Merch State** | The state of each merchant in credit card transaction records.  There are 95,558 records with 228 unique values and 1177 missing values. |



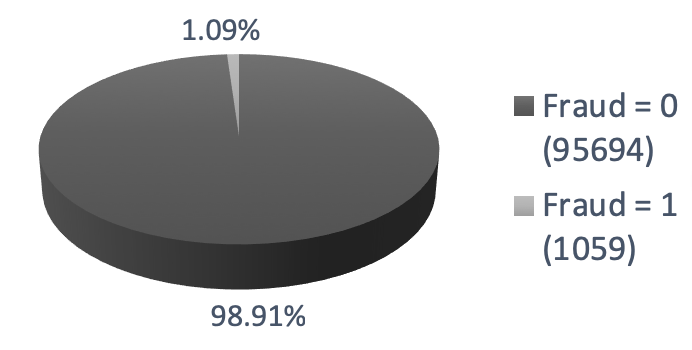
|  |  |
| --- | --- |
| *Field Name* | *Description* |
| **Merch zip** | The zip code of each merchant in credit card transaction records.  There are 92097 records with 4568 unique values and 4638 missing values. |



|  |  |
| --- | --- |
| *Field Name* | *Description* |
| **Amount** | The amount of each credit card transaction record.  There are 34909 unique values. No missing values exist.  Mean is 428, standard deviation is 10,006, max value is 3,102,046, min value is 0.01. |



|  |  |
| --- | --- |
| *Field Name* | *Description* |
| **Fraud** | Label of fraud definition. There are 2 unique values.  Label “0” means believable, label “1” means fraudulent.  No missing value exist. |

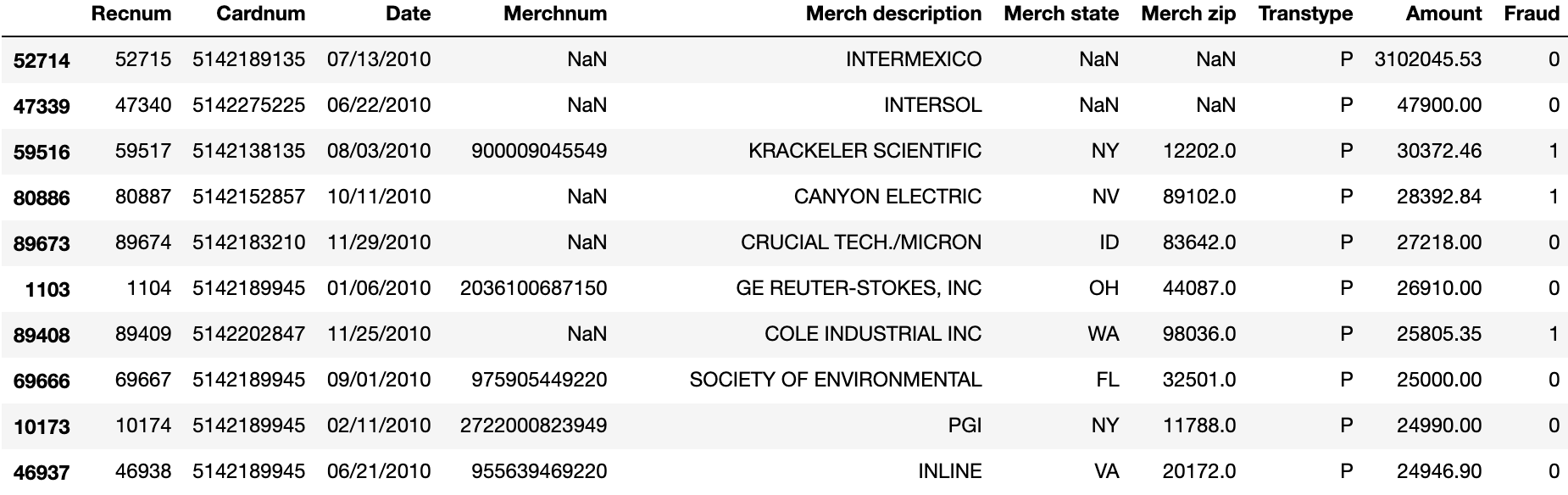


# 

# 3. Data Cleaning

## 3.1 Removing Outliers

Based on the description of dataset, we found there was a large number (3102045.53) in the Amount field. Thus, we removed the outlier.



## 3.2 Filling Missing value

According to data quality analysis, 3 variables – Merchnum, Merch state and Merch zip include missing values in the form of either blank or 0. The following table details the severity of missing values for each variable.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Variable name* | *Description* | *Missing records* | *% missing of total* | *Mode* |
| **Merchnum** | # Merchant | 3375 | 3.49 % | 930090121224 |
| **Merch state** | State of Merchant | 1195 | 1.24 % | TN |
| **Merch zip** | 5-digit zip code | 4656 | 4.81 % | 38118 |

### 3.2.1 Merchnum

Aggregate by cardnum and merchant state, use the modal value of the group. Then aggregate by merchant state only, use the modal value of that group. If still have missing values, fill with mode of all, which is ‘930090121224’.

### 3.2.2 Merch state

Aggregate by Merchant zip, use the modal value of that group. If still have missing values, fill with modal value of all, which is ‘TN’.

### 3.3.3 Merch zip

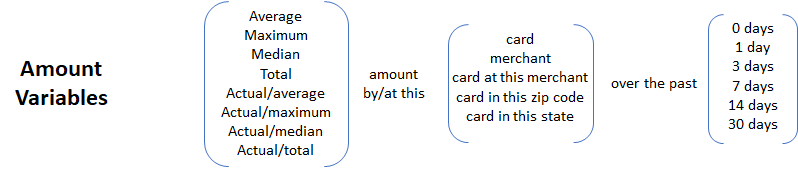
Aggregate by cardnum and merchant state, use the modal value of the group. Then aggregate by merchant state only, use the modal value of that group. If still have missing values, fill with modal value of all, which is ‘38118’.

# 4. Candidate Variable Creation

A total of 371 variables are created from the original 10 fields. The following section details the reasoning behind the creation of such variables and exact formulas to calculate them. Overall, four categories of variables are created: amount variables, frequency variables, days since variables, and velocity change variables.

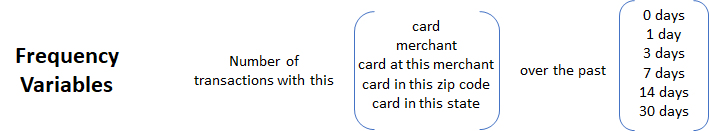
## 4.1 Amount variables

This set of variables measures the dollar value of each credit card purchase.



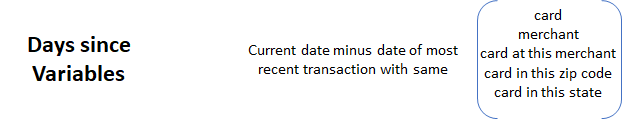
|  |  |
| --- | --- |
| ***Index*** | ***Amount variables*** |
| 1 | amt\_mean\_card\_1d |
| 2 | Actual/amt\_mean\_card\_1d |
| 3 | amt\_mean\_card\_2d |
| 4 | Actual/amt\_mean\_card\_2d |
| 5 | amt\_mean\_card\_4d |
| 6 | Actual/amt\_mean\_card\_4d |
| 7 | amt\_mean\_card\_8d |
| 8 | Actual/amt\_mean\_card\_8d |
| 9 | amt\_mean\_card\_15d |
| 10 | Actual/amt\_mean\_card\_15d |
| 11 | amt\_mean\_card\_31d |
| 12 | Actual/amt\_mean\_card\_31d |
| 13 | amt\_max\_card\_1d |
| 14 | Actual/amt\_max\_card\_1d |
| 15 | amt\_max\_card\_2d |
| 16 | Actual/amt\_max\_card\_2d |
| 17 | amt\_max\_card\_4d |
| 18 | Actual/amt\_max\_card\_4d |
| 19 | amt\_max\_card\_8d |
| 20 | Actual/amt\_max\_card\_8d |
| 21 | amt\_max\_card\_15d |
| 22 | Actual/amt\_max\_card\_15d |
| 23 | amt\_max\_card\_31d |
| 24 | Actual/amt\_max\_card\_31d |
| 25 | amt\_median\_card\_1d |
| 26 | Actual/amt\_median\_card\_1d |
| 27 | amt\_median\_card\_2d |
| 28 | Actual/amt\_median\_card\_2d |
| 29 | amt\_median\_card\_4d |
| 30 | Actual/amt\_median\_card\_4d |
| 31 | amt\_median\_card\_8d |
| 32 | Actual/amt\_median\_card\_8d |
| 33 | amt\_median\_card\_15d |
| 34 | Actual/amt\_median\_card\_15d |
| 35 | amt\_median\_card\_31d |
| 36 | Actual/amt\_median\_card\_31d |
| 37 | amt\_sum\_card\_1d |
| 38 | Actual/amt\_sum\_card\_1d |
| 39 | amt\_sum\_card\_2d |
| 40 | Actual/amt\_sum\_card\_2d |
| 41 | amt\_sum\_card\_4d |
| 42 | Actual/amt\_sum\_card\_4d |
| 43 | amt\_sum\_card\_8d |
| 44 | Actual/amt\_sum\_card\_8d |
| 45 | amt\_sum\_card\_15d |
| 46 | Actual/amt\_sum\_card\_15d |
| 47 | amt\_sum\_card\_31d |
| 48 | Actual/amt\_sum\_card\_31d |
| 49 | amt\_mean\_merch\_1d |
| 50 | Actual/amt\_mean\_merch\_1d |
| 51 | amt\_mean\_merch\_2d |
| 52 | Actual/amt\_mean\_merch\_2d |
| 53 | amt\_mean\_merch\_4d |
| 54 | Actual/amt\_mean\_merch\_4d |
| 55 | amt\_mean\_merch\_8d |
| 56 | Actual/amt\_mean\_merch\_8d |
| 57 | amt\_mean\_merch\_15d |
| 58 | Actual/amt\_mean\_merch\_15d |
| 59 | amt\_mean\_merch\_31d |
| 60 | Actual/amt\_mean\_merch\_31d |
| 61 | amt\_max\_merch\_1d |
| 62 | Actual/amt\_max\_merch\_1d |
| 63 | amt\_max\_merch\_2d |
| 64 | Actual/amt\_max\_merch\_2d |
| 65 | amt\_max\_merch\_4d |
| 66 | Actual/amt\_max\_merch\_4d |
| 67 | amt\_max\_merch\_8d |
| 68 | Actual/amt\_max\_merch\_8d |
| 69 | amt\_max\_merch\_15d |
| 70 | Actual/amt\_max\_merch\_15d |
| 71 | amt\_max\_merch\_31d |
| 72 | Actual/amt\_max\_merch\_31d |
| 73 | amt\_median\_merch\_1d |
| 74 | Actual/amt\_median\_merch\_1d |
| 75 | amt\_median\_merch\_2d |
| 76 | Actual/amt\_median\_merch\_2d |
| 77 | amt\_median\_merch\_4d |
| 78 | Actual/amt\_median\_merch\_4d |
| 79 | amt\_median\_merch\_8d |
| 80 | Actual/amt\_median\_merch\_8d |
| 81 | amt\_median\_merch\_15d |
| 82 | Actual/amt\_median\_merch\_15d |
| 83 | amt\_median\_merch\_31d |
| 84 | Actual/amt\_median\_merch\_31d |
| 85 | amt\_sum\_merch\_1d |
| 86 | Actual/amt\_sum\_merch\_1d |
| 87 | amt\_sum\_merch\_2d |
| 88 | Actual/amt\_sum\_merch\_2d |
| 89 | amt\_sum\_merch\_4d |
| 90 | Actual/amt\_sum\_merch\_4d |
| 91 | amt\_sum\_merch\_8d |
| 92 | Actual/amt\_sum\_merch\_8d |
| 93 | amt\_sum\_merch\_15d |
| 94 | Actual/amt\_sum\_merch\_15d |
| 95 | amt\_sum\_merch\_31d |
| 96 | Actual/amt\_sum\_merch\_31d |
| 97 | amt\_mean\_cardmerch\_1d |
| 98 | Actual/amt\_mean\_cardmerch\_1d |
| 99 | amt\_mean\_cardmerch\_2d |
| 100 | Actual/amt\_mean\_cardmerch\_2d |
| 101 | amt\_mean\_cardmerch\_4d |
| 102 | Actual/amt\_mean\_cardmerch\_4d |
| 103 | amt\_mean\_cardmerch\_8d |
| 104 | Actual/amt\_mean\_cardmerch\_8d |
| 105 | amt\_mean\_cardmerch\_15d |
| 106 | Actual/amt\_mean\_cardmerch\_15d |
| 107 | amt\_mean\_cardmerch\_31d |
| 108 | Actual/amt\_mean\_cardmerch\_31d |
| 109 | amt\_max\_cardmerch\_1d |
| 110 | Actual/amt\_max\_cardmerch\_1d |
| 111 | amt\_max\_cardmerch\_2d |
| 112 | Actual/amt\_max\_cardmerch\_2d |
| 113 | amt\_max\_cardmerch\_4d |
| 114 | Actual/amt\_max\_cardmerch\_4d |
| 115 | amt\_max\_cardmerch\_8d |
| 116 | Actual/amt\_max\_cardmerch\_8d |
| 117 | amt\_max\_cardmerch\_15d |
| 118 | Actual/amt\_max\_cardmerch\_15d |
| 119 | amt\_max\_cardmerch\_31d |
| 120 | Actual/amt\_max\_cardmerch\_31d |
| 121 | amt\_median\_cardmerch\_1d |
| 122 | Actual/amt\_median\_cardmerch\_1d |
| 123 | amt\_median\_cardmerch\_2d |
| 124 | Actual/amt\_median\_cardmerch\_2d |
| 125 | amt\_median\_cardmerch\_4d |
| 126 | Actual/amt\_median\_cardmerch\_4d |
| 127 | amt\_median\_cardmerch\_8d |
| 128 | Actual/amt\_median\_cardmerch\_8d |
| 129 | amt\_median\_cardmerch\_15d |
| 130 | Actual/amt\_median\_cardmerch\_15d |
| 131 | amt\_median\_cardmerch\_31d |
| 132 | Actual/amt\_median\_cardmerch\_31d |
| 133 | amt\_sum\_cardmerch\_1d |
| 134 | Actual/amt\_sum\_cardmerch\_1d |
| 135 | amt\_sum\_cardmerch\_2d |
| 136 | Actual/amt\_sum\_cardmerch\_2d |
| 137 | amt\_sum\_cardmerch\_4d |
| 138 | Actual/amt\_sum\_cardmerch\_4d |
| 139 | amt\_sum\_cardmerch\_8d |
| 140 | Actual/amt\_sum\_cardmerch\_8d |
| 141 | amt\_sum\_cardmerch\_15d |
| 142 | Actual/amt\_sum\_cardmerch\_15d |
| 143 | amt\_sum\_cardmerch\_31d |
| 144 | Actual/amt\_sum\_cardmerch\_31d |
| 145 | amt\_mean\_cardzip\_1d |
| 146 | Actual/amt\_mean\_cardzip\_1d |
| 147 | amt\_mean\_cardzip\_2d |
| 148 | Actual/amt\_mean\_cardzip\_2d |
| 149 | amt\_mean\_cardzip\_4d |
| 150 | Actual/amt\_mean\_cardzip\_4d |
| 151 | amt\_mean\_cardzip\_8d |
| 152 | Actual/amt\_mean\_cardzip\_8d |
| 153 | amt\_mean\_cardzip\_15d |
| 154 | Actual/amt\_mean\_cardzip\_15d |
| 155 | amt\_mean\_cardzip\_31d |
| 156 | Actual/amt\_mean\_cardzip\_31d |
| 157 | amt\_max\_cardzip\_1d |
| 158 | Actual/amt\_max\_cardzip\_1d |
| 159 | amt\_max\_cardzip\_2d |
| 160 | Actual/amt\_max\_cardzip\_2d |
| 161 | amt\_max\_cardzip\_4d |
| 162 | Actual/amt\_max\_cardzip\_4d |
| 163 | amt\_max\_cardzip\_8d |
| 164 | Actual/amt\_max\_cardzip\_8d |
| 165 | amt\_max\_cardzip\_15d |
| 166 | Actual/amt\_max\_cardzip\_15d |
| 167 | amt\_max\_cardzip\_31d |
| 168 | Actual/amt\_max\_cardzip\_31d |
| 169 | amt\_median\_cardzip\_1d |
| 170 | Actual/amt\_median\_cardzip\_1d |
| 171 | amt\_median\_cardzip\_2d |
| 172 | Actual/amt\_median\_cardzip\_2d |
| 173 | amt\_median\_cardzip\_4d |
| 174 | Actual/amt\_median\_cardzip\_4d |
| 175 | amt\_median\_cardzip\_8d |
| 176 | Actual/amt\_median\_cardzip\_8d |
| 177 | amt\_median\_cardzip\_15d |
| 178 | Actual/amt\_median\_cardzip\_15d |
| 179 | amt\_median\_cardzip\_31d |
| 180 | Actual/amt\_median\_cardzip\_31d |
| 181 | amt\_sum\_cardzip\_1d |
| 182 | Actual/amt\_sum\_cardzip\_1d |
| 183 | amt\_sum\_cardzip\_2d |
| 184 | Actual/amt\_sum\_cardzip\_2d |
| 185 | amt\_sum\_cardzip\_4d |
| 186 | Actual/amt\_sum\_cardzip\_4d |
| 187 | amt\_sum\_cardzip\_8d |
| 188 | Actual/amt\_sum\_cardzip\_8d |
| 189 | amt\_sum\_cardzip\_15d |
| 190 | Actual/amt\_sum\_cardzip\_15d |
| 191 | amt\_sum\_cardzip\_31d |
| 192 | Actual/amt\_sum\_cardzip\_31d |
| 193 | amt\_mean\_cardstate\_1d |
| 194 | Actual/amt\_mean\_cardstate\_1d |
| 195 | amt\_mean\_cardstate\_2d |
| 196 | Actual/amt\_mean\_cardstate\_2d |
| 197 | amt\_mean\_cardstate\_4d |
| 198 | Actual/amt\_mean\_cardstate\_4d |
| 199 | amt\_mean\_cardstate\_8d |
| 200 | Actual/amt\_mean\_cardstate\_8d |
| 201 | amt\_mean\_cardstate\_15d |
| 202 | Actual/amt\_mean\_cardstate\_15d |
| 203 | amt\_mean\_cardstate\_31d |
| 204 | Actual/amt\_mean\_cardstate\_31d |
| 205 | amt\_max\_cardstate\_1d |
| 206 | Actual/amt\_max\_cardstate\_1d |
| 207 | amt\_max\_cardstate\_2d |
| 208 | Actual/amt\_max\_cardstate\_2d |
| 209 | amt\_max\_cardstate\_4d |
| 210 | Actual/amt\_max\_cardstate\_4d |
| 211 | amt\_max\_cardstate\_8d |
| 212 | Actual/amt\_max\_cardstate\_8d |
| 213 | amt\_max\_cardstate\_15d |
| 214 | Actual/amt\_max\_cardstate\_15d |
| 215 | amt\_max\_cardstate\_31d |
| 216 | Actual/amt\_max\_cardstate\_31d |
| 217 | amt\_median\_cardstate\_1d |
| 218 | Actual/amt\_median\_cardstate\_1d |
| 219 | amt\_median\_cardstate\_2d |
| 220 | Actual/amt\_median\_cardstate\_2d |
| 221 | amt\_median\_cardstate\_4d |
| 222 | Actual/amt\_median\_cardstate\_4d |
| 223 | amt\_median\_cardstate\_8d |
| 224 | Actual/amt\_median\_cardstate\_8d |
| 225 | amt\_median\_cardstate\_15d |
| 226 | Actual/amt\_median\_cardstate\_15d |
| 227 | amt\_median\_cardstate\_31d |
| 228 | Actual/amt\_median\_cardstate\_31d |
| 229 | amt\_sum\_cardstate\_1d |
| 230 | Actual/amt\_sum\_cardstate\_1d |
| 231 | amt\_sum\_cardstate\_2d |
| 232 | Actual/amt\_sum\_cardstate\_2d |
| 233 | amt\_sum\_cardstate\_4d |
| 234 | Actual/amt\_sum\_cardstate\_4d |
| 235 | amt\_sum\_cardstate\_8d |
| 236 | Actual/amt\_sum\_cardstate\_8d |
| 237 | amt\_sum\_cardstate\_15d |
| 238 | Actual/amt\_sum\_cardstate\_15d |
| 239 | amt\_sum\_cardstate\_31d |
| 240 | Actual/amt\_sum\_cardstate\_31d |

## 4.2 Frequency variables



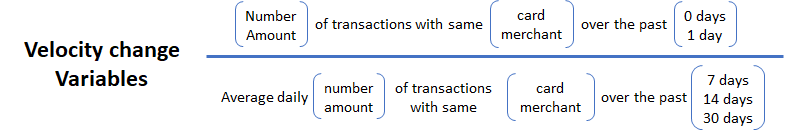
|  |  |
| --- | --- |
| ***Index*** | ***Frequency variables*** |
| 241 | freq\_card\_1d |
| 242 | freq\_card\_2d |
| 243 | freq\_card\_4d |
| 244 | freq\_card\_8d |
| 245 | freq\_card\_15d |
| 246 | freq\_card\_31d |
| 247 | freq\_merch\_1d |
| 248 | freq\_merch\_2d |
| 249 | freq\_merch\_4d |
| 250 | freq\_merch\_8d |
| 251 | freq\_merch\_15d |
| 252 | freq\_merch\_31d |
| 253 | freq\_cardmerch\_1d |
| 254 | freq\_cardmerch\_2d |
| 255 | freq\_cardmerch\_4d |
| 256 | freq\_cardmerch\_8d |
| 257 | freq\_cardmerch\_15d |
| 258 | freq\_cardmerch\_31d |
| 259 | freq\_cardzip\_1d |
| 260 | freq\_cardzip\_2d |
| 261 | freq\_cardzip\_4d |
| 262 | freq\_cardzip\_8d |
| 263 | freq\_cardzip\_15d |
| 264 | freq\_cardzip\_31d |
| 265 | freq\_cardstate\_1d |
| 266 | freq\_cardstate\_2d |
| 267 | freq\_cardstate\_4d |
| 268 | freq\_cardstate\_8d |
| 269 | freq\_cardstate\_15d |
| 270 | freq\_cardstate\_31d |

## 4.3 Days since variables



|  |  |
| --- | --- |
| ***Index*** | ***Days Since variables*** |
| 367 | Days\_since\_per\_Cardnum |
| 368 | Days\_since\_per\_Merchnum |
| 369 | Days\_since\_per\_Cardnum\_Merchnum |
| 370 | Days\_since\_per\_Cardnum\_Merch zip |
| 371 | Days\_since\_per\_Cardnum\_Merch state |

## 4.4 Velocity change variables



|  |  |
| --- | --- |
| ***Index*** | ***Velocity Change variables*** |
| 271 | tran\_count\_card\_1d/avgdaily\_tran\_count\_Merchnum\_8d |
| 272 | tran\_count\_card\_1d/avgdaily\_tran\_count\_Cardnum\_8d |
| 273 | tran\_count\_card\_1d/avgdaily\_tran\_sum\_Merchnum\_8d |
| 274 | tran\_count\_card\_1d/avgdaily\_tran\_sum\_Cardnum\_8d |
| 275 | tran\_count\_card\_1d/avgdaily\_tran\_count\_Merchnum\_15d |
| 276 | tran\_count\_card\_1d/avgdaily\_tran\_count\_Cardnum\_15d |
| 277 | tran\_count\_card\_1d/avgdaily\_tran\_sum\_Merchnum\_15d |
| 278 | tran\_count\_card\_1d/avgdaily\_tran\_sum\_Cardnum\_15d |
| 279 | tran\_count\_card\_1d/avgdaily\_tran\_count\_Merchnum\_31d |
| 280 | tran\_count\_card\_1d/avgdaily\_tran\_count\_Cardnum\_31d |
| 281 | tran\_count\_card\_1d/avgdaily\_tran\_sum\_Merchnum\_31d |
| 282 | tran\_count\_card\_1d/avgdaily\_tran\_sum\_Cardnum\_31d |
| 283 | tran\_count\_card\_2d/avgdaily\_tran\_count\_Merchnum\_8d |
| 284 | tran\_count\_card\_2d/avgdaily\_tran\_count\_Cardnum\_8d |
| 285 | tran\_count\_card\_2d/avgdaily\_tran\_sum\_Merchnum\_8d |
| 286 | tran\_count\_card\_2d/avgdaily\_tran\_sum\_Cardnum\_8d |
| 287 | tran\_count\_card\_2d/avgdaily\_tran\_count\_Merchnum\_15d |
| 288 | tran\_count\_card\_2d/avgdaily\_tran\_count\_Cardnum\_15d |
| 289 | tran\_count\_card\_2d/avgdaily\_tran\_sum\_Merchnum\_15d |
| 290 | tran\_count\_card\_2d/avgdaily\_tran\_sum\_Cardnum\_15d |
| 291 | tran\_count\_card\_2d/avgdaily\_tran\_count\_Merchnum\_31d |
| 292 | tran\_count\_card\_2d/avgdaily\_tran\_count\_Cardnum\_31d |
| 293 | tran\_count\_card\_2d/avgdaily\_tran\_sum\_Merchnum\_31d |
| 294 | tran\_count\_card\_2d/avgdaily\_tran\_sum\_Cardnum\_31d |
| 295 | tran\_sum\_card\_1d/avgdaily\_tran\_count\_Merchnum\_8d |
| 296 | tran\_sum\_card\_1d/avgdaily\_tran\_count\_Cardnum\_8d |
| 297 | tran\_sum\_card\_1d/avgdaily\_tran\_sum\_Merchnum\_8d |
| 298 | tran\_sum\_card\_1d/avgdaily\_tran\_sum\_Cardnum\_8d |
| 299 | tran\_sum\_card\_1d/avgdaily\_tran\_count\_Merchnum\_15d |
| 300 | tran\_sum\_card\_1d/avgdaily\_tran\_count\_Cardnum\_15d |
| 301 | tran\_sum\_card\_1d/avgdaily\_tran\_sum\_Merchnum\_15d |
| 302 | tran\_sum\_card\_1d/avgdaily\_tran\_sum\_Cardnum\_15d |
| 303 | tran\_sum\_card\_1d/avgdaily\_tran\_count\_Merchnum\_31d |
| 304 | tran\_sum\_card\_1d/avgdaily\_tran\_count\_Cardnum\_31d |
| 305 | tran\_sum\_card\_1d/avgdaily\_tran\_sum\_Merchnum\_31d |
| 306 | tran\_sum\_card\_1d/avgdaily\_tran\_sum\_Cardnum\_31d |
| 307 | tran\_sum\_card\_2d/avgdaily\_tran\_count\_Merchnum\_8d |
| 308 | tran\_sum\_card\_2d/avgdaily\_tran\_count\_Cardnum\_8d |
| 309 | tran\_sum\_card\_2d/avgdaily\_tran\_sum\_Merchnum\_8d |
| 310 | tran\_sum\_card\_2d/avgdaily\_tran\_sum\_Cardnum\_8d |
| 311 | tran\_sum\_card\_2d/avgdaily\_tran\_count\_Merchnum\_15d |
| 312 | tran\_sum\_card\_2d/avgdaily\_tran\_count\_Cardnum\_15d |
| 313 | tran\_sum\_card\_2d/avgdaily\_tran\_sum\_Merchnum\_15d |
| 314 | tran\_sum\_card\_2d/avgdaily\_tran\_sum\_Cardnum\_15d |
| 315 | tran\_sum\_card\_2d/avgdaily\_tran\_count\_Merchnum\_31d |
| 316 | tran\_sum\_card\_2d/avgdaily\_tran\_count\_Cardnum\_31d |
| 317 | tran\_sum\_card\_2d/avgdaily\_tran\_sum\_Merchnum\_31d |
| 318 | tran\_sum\_card\_2d/avgdaily\_tran\_sum\_Cardnum\_31d |
| 319 | tran\_count\_Merch\_1d/avgdaily\_tran\_count\_Merchnum\_8d |
| 320 | tran\_count\_Merch\_1d/avgdaily\_tran\_count\_Cardnum\_8d |
| 321 | tran\_count\_Merch\_1d/avgdaily\_tran\_sum\_Merchnum\_8d |
| 322 | tran\_count\_Merch\_1d/avgdaily\_tran\_sum\_Cardnum\_8d |
| 323 | tran\_count\_Merch\_1d/avgdaily\_tran\_count\_Merchnum\_15d |
| 324 | tran\_count\_Merch\_1d/avgdaily\_tran\_count\_Cardnum\_15d |
| 325 | tran\_count\_Merch\_1d/avgdaily\_tran\_sum\_Merchnum\_15d |
| 326 | tran\_count\_Merch\_1d/avgdaily\_tran\_sum\_Cardnum\_15d |
| 327 | tran\_count\_Merch\_1d/avgdaily\_tran\_count\_Merchnum\_31d |
| 328 | tran\_count\_Merch\_1d/avgdaily\_tran\_count\_Cardnum\_31d |
| 329 | tran\_count\_Merch\_1d/avgdaily\_tran\_sum\_Merchnum\_31d |
| 330 | tran\_count\_Merch\_1d/avgdaily\_tran\_sum\_Cardnum\_31d |
| 331 | tran\_count\_Merch\_2d/avgdaily\_tran\_count\_Merchnum\_8d |
| 332 | tran\_count\_Merch\_2d/avgdaily\_tran\_count\_Cardnum\_8d |
| 333 | tran\_count\_Merch\_2d/avgdaily\_tran\_sum\_Merchnum\_8d |
| 334 | tran\_count\_Merch\_2d/avgdaily\_tran\_sum\_Cardnum\_8d |
| 335 | tran\_count\_Merch\_2d/avgdaily\_tran\_count\_Merchnum\_15d |
| 336 | tran\_count\_Merch\_2d/avgdaily\_tran\_count\_Cardnum\_15d |
| 337 | tran\_count\_Merch\_2d/avgdaily\_tran\_sum\_Merchnum\_15d |
| 338 | tran\_count\_Merch\_2d/avgdaily\_tran\_sum\_Cardnum\_15d |
| 339 | tran\_count\_Merch\_2d/avgdaily\_tran\_count\_Merchnum\_31d |
| 340 | tran\_count\_Merch\_2d/avgdaily\_tran\_count\_Cardnum\_31d |
| 341 | tran\_count\_Merch\_2d/avgdaily\_tran\_sum\_Merchnum\_31d |
| 342 | tran\_count\_Merch\_2d/avgdaily\_tran\_sum\_Cardnum\_31d |
| 343 | tran\_sum\_Merch\_1d/avgdaily\_tran\_count\_Merchnum\_8d |
| 344 | tran\_sum\_Merch\_1d/avgdaily\_tran\_count\_Cardnum\_8d |
| 345 | tran\_sum\_Merch\_1d/avgdaily\_tran\_sum\_Merchnum\_8d |
| 346 | tran\_sum\_Merch\_1d/avgdaily\_tran\_sum\_Cardnum\_8d |
| 347 | tran\_sum\_Merch\_1d/avgdaily\_tran\_count\_Merchnum\_15d |
| 348 | tran\_sum\_Merch\_1d/avgdaily\_tran\_count\_Cardnum\_15d |
| 349 | tran\_sum\_Merch\_1d/avgdaily\_tran\_sum\_Merchnum\_15d |
| 350 | tran\_sum\_Merch\_1d/avgdaily\_tran\_sum\_Cardnum\_15d |
| 351 | tran\_sum\_Merch\_1d/avgdaily\_tran\_count\_Merchnum\_31d |
| 352 | tran\_sum\_Merch\_1d/avgdaily\_tran\_count\_Cardnum\_31d |
| 353 | tran\_sum\_Merch\_1d/avgdaily\_tran\_sum\_Merchnum\_31d |
| 354 | tran\_sum\_Merch\_1d/avgdaily\_tran\_sum\_Cardnum\_31d |
| 355 | tran\_sum\_Merch\_2d/avgdaily\_tran\_count\_Merchnum\_8d |
| 356 | tran\_sum\_Merch\_2d/avgdaily\_tran\_count\_Cardnum\_8d |
| 357 | tran\_sum\_Merch\_2d/avgdaily\_tran\_sum\_Merchnum\_8d |
| 358 | tran\_sum\_Merch\_2d/avgdaily\_tran\_sum\_Cardnum\_8d |
| 359 | tran\_sum\_Merch\_2d/avgdaily\_tran\_count\_Merchnum\_15d |
| 360 | tran\_sum\_Merch\_2d/avgdaily\_tran\_count\_Cardnum\_15d |
| 361 | tran\_sum\_Merch\_2d/avgdaily\_tran\_sum\_Merchnum\_15d |
| 362 | tran\_sum\_Merch\_2d/avgdaily\_tran\_sum\_Cardnum\_15d |
| 363 | tran\_sum\_Merch\_2d/avgdaily\_tran\_count\_Merchnum\_31d |
| 364 | tran\_sum\_Merch\_2d/avgdaily\_tran\_count\_Cardnum\_31d |
| 365 | tran\_sum\_Merch\_2d/avgdaily\_tran\_sum\_Merchnum\_31d |
| 366 | tran\_sum\_Merch\_2d/avgdaily\_tran\_sum\_Cardnum\_31d |

# 

# 5. Feature selection

Many variables created in the previous section are highly correlated to one another; also, we hope to distill only the most useful predictor candidates to train our fraud models with. Thus, we resorted to the following two methods to screen candidate variables prior to fraud model development.

## 5.1 Filtering with Kolmogorov–Smirnov (KS) and Fraud Detection Rate (FDR)

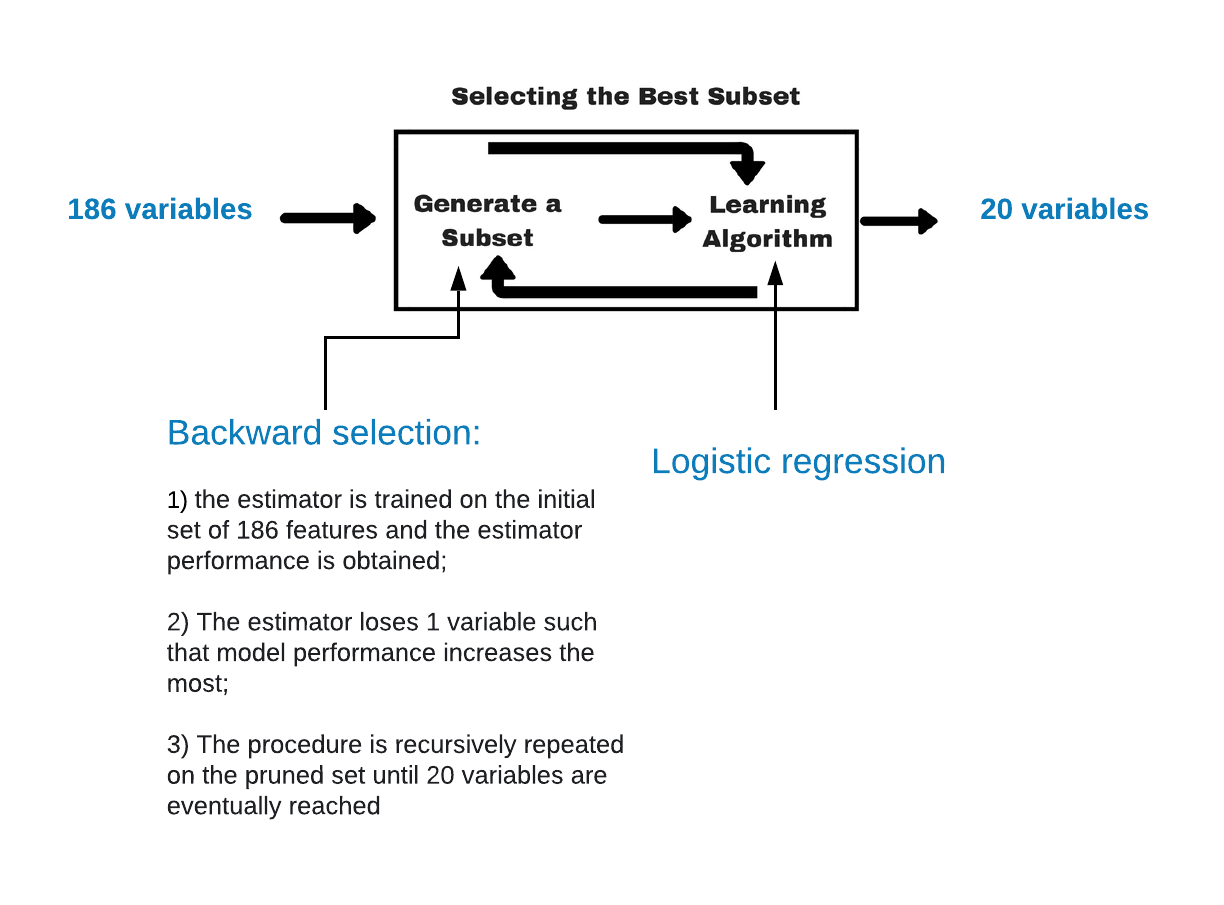
This approach serves to rank order all candidate variables by individual predictive power. The goal is to select the top 50% candidate variables (186 out of 371) in terms of individual predictive power. Such individual predictive power is measured by a final score, the formula being described below:

* 1. . This measures how well a given variable separates goods and bads.
  2. **FDR with 3% cutoff**: the percentage of real frauds within the top 3% predicted frauds.

**Final score = ranking by KS score descent + ranking by FDR 3% descent**

## 5.2 Wrapper with Logistic Regression and backward feature selection

Through the filtering process exist only 186 (top 50%) candidate variables to be further selected here in this step. Compared to the filter method which sorts out variables with high individual predictive power, this wrapper method seeks to find the optimal set of candidate variables collectively. As a result, 20 candidate variables are selected for model development.

[](https://www.lucidchart.com/documents/edit/3eb9a073-163a-48e3-b986-703d165a2f42/0?callback=close&name=docs&callback_type=back&v=690&s=612)

|  |  |
| --- | --- |
| ***Index*** | ***Final variables*** |
| **1** | amt\_max\_cardstate\_15d |
| **2** | Actual/amt\_median\_merch\_31d |
| **3** | Actual/amt\_median\_merch\_15d |
| **4** | freq\_cardzip\_2d |
| **5** | freq\_cardstate\_4d |
| **6** | freq\_cardmerch\_4d |
| **7** | freq\_cardzip\_4d |
| **8** | Actual/amt\_median\_card\_4d |
| **9** | tran\_count\_Merch\_2d/avgdaily\_tran\_sum\_Cardnum\_31d |
| **10** | amt\_max\_cardstate\_31d |
| **11** | freq\_cardstate\_2d |
| **12** | freq\_cardmerch\_2d |
| **13** | Actual/amt\_mean\_merch\_31d |
| **14** | Actual/amt\_mean\_merch\_15d |
| **15** | Days\_since\_per\_Cardnum\_Merch state |
| **16** | Actual/amt\_median\_card\_31d |
| **17** | Actual/amt\_mean\_merch\_8d |
| **18** | freq\_card\_2d |
| **19** | freq\_card\_1d |
| **20** | freq\_card\_8d |

# 6. Supervised Fraud Algorithms

## 6.1 Model Development

Out of all trained 5 models, the Neural Network model was selected to be the final fraud detection model. We summarized the fraud detection rate at 3% for each model below. The Neural network model is the most performant in terms of FDR on the out-of-time set; it also produces relatively steady results across 10 trials.

|  |  |  |  |
| --- | --- | --- | --- |
| ***Model*** | ***Training*** | ***Testing*** | ***Out of Time*** |
| **Logistic Regression** | 54.70% | 49.01% | 34.60% |
| **Random Forest** | 83.16% | 57.83% | 33.58% |
| **Boosted Tree** | 98.07% | 55.34% | 32.07% |
| **SVM** | 61.56% | 52.96% | 37.99% |
| **Neural Network**  **(Final model)** | **70.49%** | **58.58%** | **39.00%** |

### 6.1.1 Logistic Regression (baseline model)

Logistic Regression is one the most popular classification algorithms that predict the probability of binary outcomes by using sigmoid function. From the outcome we have, the difference between testing dataset and out of time dataset is more than 10%, so it tends to be overfitting with this model. Compared to other models, our baseline model having better performance than two other models (Random Forest and Boosted Decision Tree) based on out of time dataset. Since the Logistic Regression model is a linear model, the outcome of running several time remain the same.

|  |  |  |  |
| --- | --- | --- | --- |
| ***FDR @3%*** | ***Train*** | ***Test*** | ***OOT*** |
|  | 54.70% | 49.01% | 34.60% |

### 6.1.2 Random Forest

Random Forest is a modified version of decision tree. It combines multiple small trees, and considers a subset of features to fit the model. After trying different parameters of the Random Forest model, we choose the optimal combination of parameters, which has 100 trees to grow and 4 variables randomly sampled as candidates at each split.Using random forest model, we found that the FDR of train dataset and FDR of test dataset have a difference of 25%, which indicates a overfitting problem. Also, we captured only about 34% frauds in the end. Thus, Random Forest is not a very good model.

|  |  |  |  |
| --- | --- | --- | --- |
| ***FDR@3%*** | ***Train*** | ***Test*** | ***OOT*** |
| 1 | 83.09% | 58.89% | 32.40% |
| 2 | 82.46% | 59.29% | 31.28% |
| 3 | 83.41% | 55.34% | 33.52% |
| 4 | 84.05% | 58.89% | 37.43% |
| 5 | 83.57% | 59.68% | 34.08% |
| 6 | 83.09% | 58.50% | 33.52% |
| 7 | 83.73% | 58.10% | 34.08% |
| 8 | 82.46% | 62.06% | 39.11% |
| 9 | 83.25% | 49.01% | 27.37% |
| 10 | 82.45% | 58.50% | 32.96% |
| Average | 83.16% | 57.83% | 33.58% |

### 6.1.3 Boosted Decision Tree

The boosted trees is a way to seek a strong model by combining several weak models and the most common weak learner is the decision trees. The Adaptive Boosting algorithm trains all records and the next generation will focus on those records which have strong misclassified that have a high weight.

Optimizing model by changing important parameters including the depth of the tree, the number of weak learners, and the loss function. Finally, the Boosted Decision Tree performed best when the number of trees equals to 300 and the depth of trees equals to 6 with a square loss function.

However, Boosted Decision Tree doesn’t perform well on overfitting problem. We can find this from the result of the values of train and test.

|  |  |  |  |
| --- | --- | --- | --- |
| ***FDR @3%*** | ***Train*** | ***Test*** | ***OOT*** |
| 1 | 98.56% | 46.64% | 32.40% |
| 2 | 93.14% | 35.57% | 19.55% |
| 3 | 99.84% | 63.24% | 31.84% |
| 4 | 99.84% | 63.63% | 40.78% |
| 5 | 99.84% | 52.17% | 35.75% |
| 6 | 99.68% | 65.22% | 44.13% |
| 7 | 95.06% | 58.50% | 20.11% |
| 8 | 98.88% | 52.17% | 32.96% |
| 9 | 96.13% | 56.13% | 28.49% |
| 10 | 99.68% | 60.08% | 34.64% |
| Average | 98.07% | 55.34% | 32.07% |

### 6.1.4 SVM

SVM is abbreviation for “Support Vector Machines”, a supervised learning models used for classification and regression analysis. SVM training algorithm usually works as a non-probabilistic binary linear classifier, although methods such as Platt scaling exist to use SVM in a probabilistic classification setting, which is what we use to get fraud score.

|  |  |  |  |
| --- | --- | --- | --- |
| ***FDR @3%*** | ***Train*** | ***Test*** | ***OOT*** |
|  | 61.56% | 52.96% | 37.99% |

### 6.1.5 Neural Network

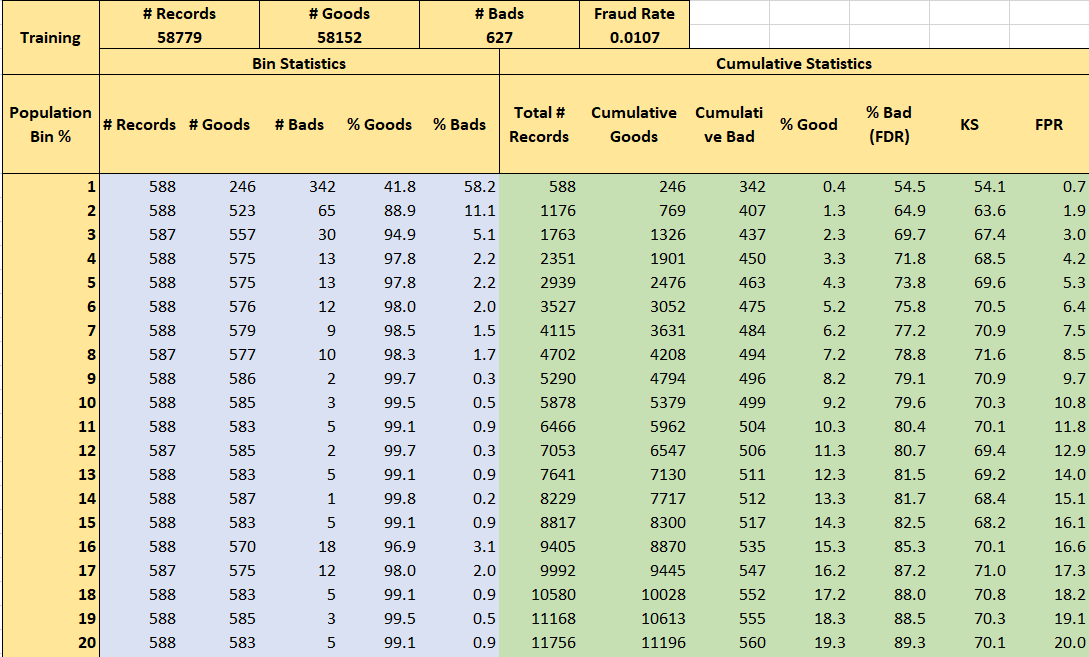
Neural network is an excellent complement to other techniques and improves with exposure to data. It’s a part of cognitive computing technology where the machine mimics how the human brain works and how it observes patterns. The neural networks are completely adaptive; able to learn from patterns of legitimate behavior. These can adapt to the change in the behavior of normal transactions and identify patterns of fraud transactions. For Neural Network model, we used 5 input layer nodes and 5 hidden layer nodes.

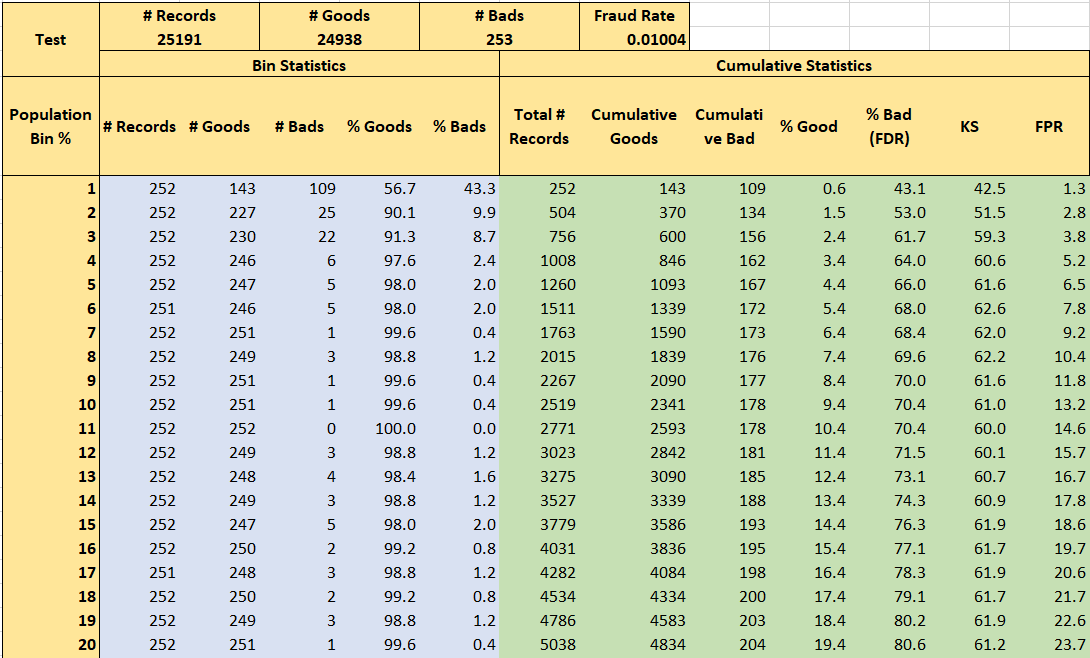
|  |  |  |  |
| --- | --- | --- | --- |
| ***FDR @3%*** | ***Train*** | ***Test*** | ***OOT*** |
| 1 | 70.49% | 59.29% | 38.55% |
| 2 | 70.49% | 56.92% | 39.11% |
| 3 | 70.65% | 57.71% | 39.11% |
| 4 | 70.49% | 58.10% | 38.55% |
| 5 | 70.49% | 57.71% | 38.55% |
| 6 | 70.49% | 58.10% | 39.11% |
| 7 | 70.49% | 59.29% | 39.11% |
| 8 | 70.65% | 58.89% | 39.11% |
| 9 | 70.02% | 62.06% | 40.22% |
| 10 | 70.65% | 57.71% | 38.55% |
| Average | 70.49% | 58.58% | 39.00% |

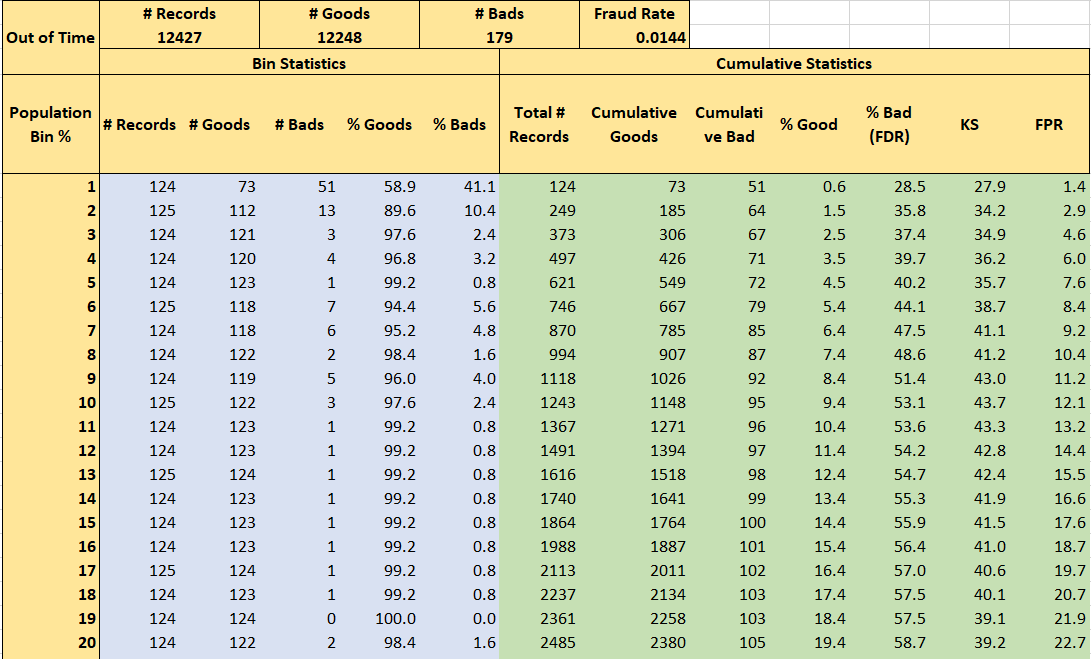
# 7. Results

As we concluded that Neural Network to be the final fraud detection model, we report a more in-depth view of its performance on training set, test set and out-of-time set below.

The following tables summarize the top 1% to 20% records of high fraud likelihood. It is clear that the final model does a decent job on catching frauds on the training set and the performance degrades on the test set and even further on the out-of-time set.

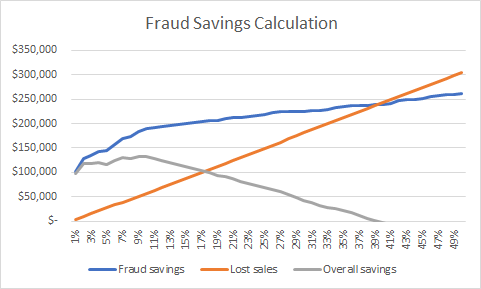






**Recommendation**

As a result, we recommend the fraud threshold to be around 10%. Assuming $2000 fraud saving for each fraud successfully caught and $50 loss for each false positive, we plotted the overall savings at each cut-off from 1% to 50%. Right around 10%, overall saving is maximized - $132,600. Choosing a threshold greater than 10% risks costing too much for fraud savings.



# 8 Conclusion

In this section, we briefly reflect on the steps we took to arrive at the the recommendation above.

**Process flow**

* **Data exploration**. Exploratory data analysis was done and Data Quality Report produced.
* **Data cleaning**. Outliers were removed and missing values filled in.
* **Candidate variable creation**. We created 371 candidate variables out the original 10 fields.
* **Feature selection**. We selected the top 20 candidate variables in terms of predicted power.
* **Fraud model development**. A neural network was chosen to be the final model among 5 candidate models
* **Results / recommended actions**. A 10% cut-off on predicted fraud likelihood was recommended for fraud classification.

**Limitations**

* The dataset is a bit small to train complicated algorithms on such as neural network. We see some fluctuations across different model runs even on the same data.
* The wrapper feature selection process was conducted on logistic regression, which is a linear algorithm. Some important non-linear relationships might not be sufficiently captured by wrapping with logistic regression.

**Migitations**

* To solve the problem of small sample size, the best way to collect more data. If feasible, we would like to wait until there are sufficient records to build our fraud models.
* To extract non-linear predictive power of features, we could use algorithms such as random forests to conduct feature selection.

# Appendix

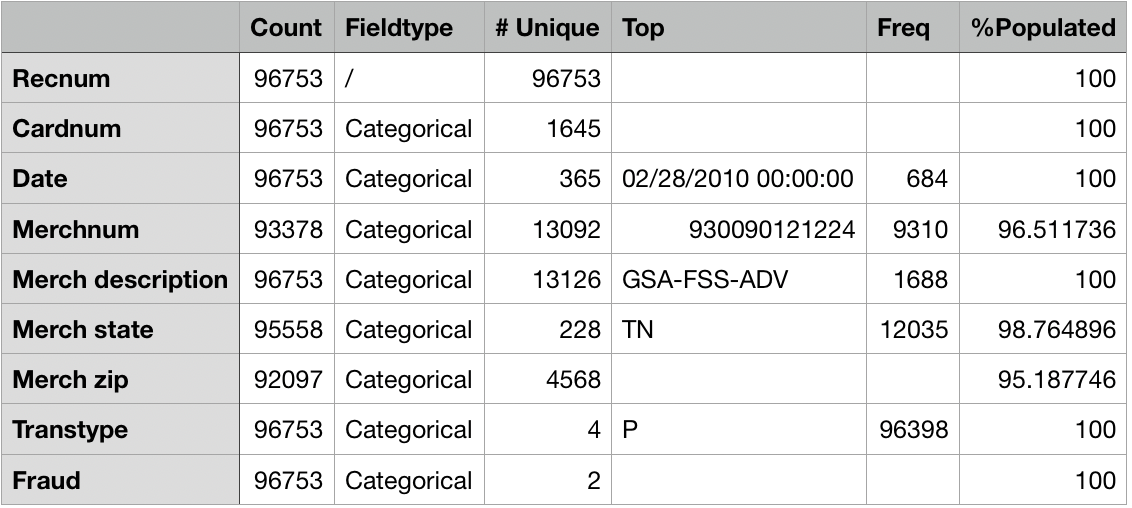
**Credit Card Transaction Data Quality Report**

# **Data Description**

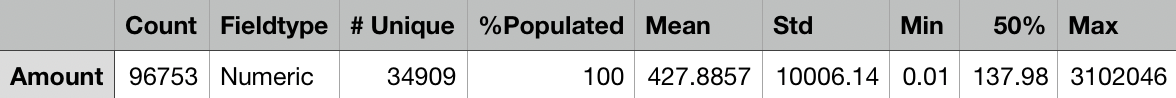
This data contains card transactions of merchandise purchased in 2010, for purpose to calculate the frequency of card transaction fraud, collect characteristic of fraud transactions and predict future fraud transactions. This dataset provides data by date, with 9 meaningful fields and 96,753 records.

# **Table of Fields**

Categorical fields



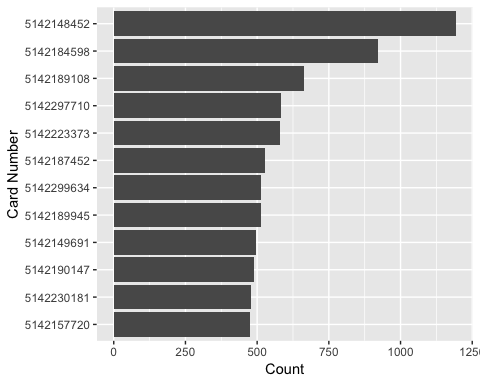
Numeric field



# **Field Description**

### **Cardnum**

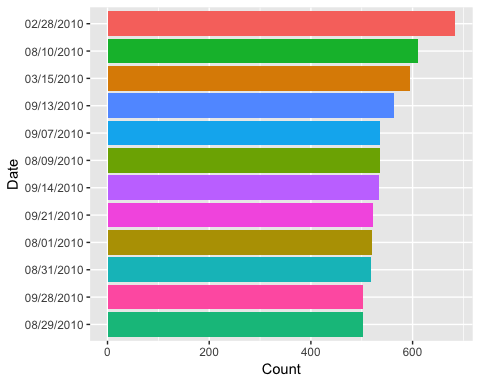
This field is categorical, each record of it has a value. It has 1645 unique values.



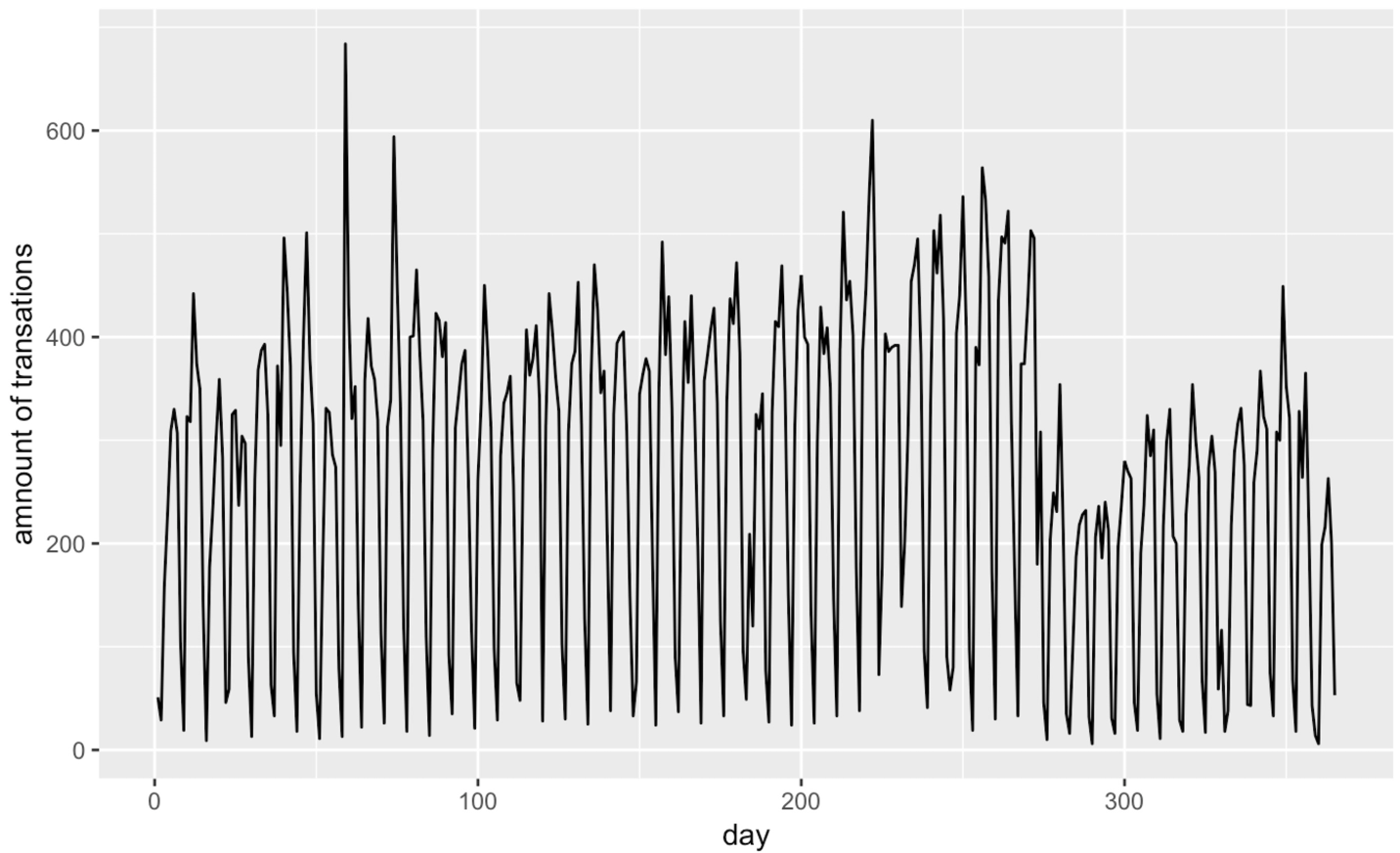
### **Date**

This field is categorical, each record of it has a value. It represents transactions in all 365 days. The most common transaction date is 2/28/10.

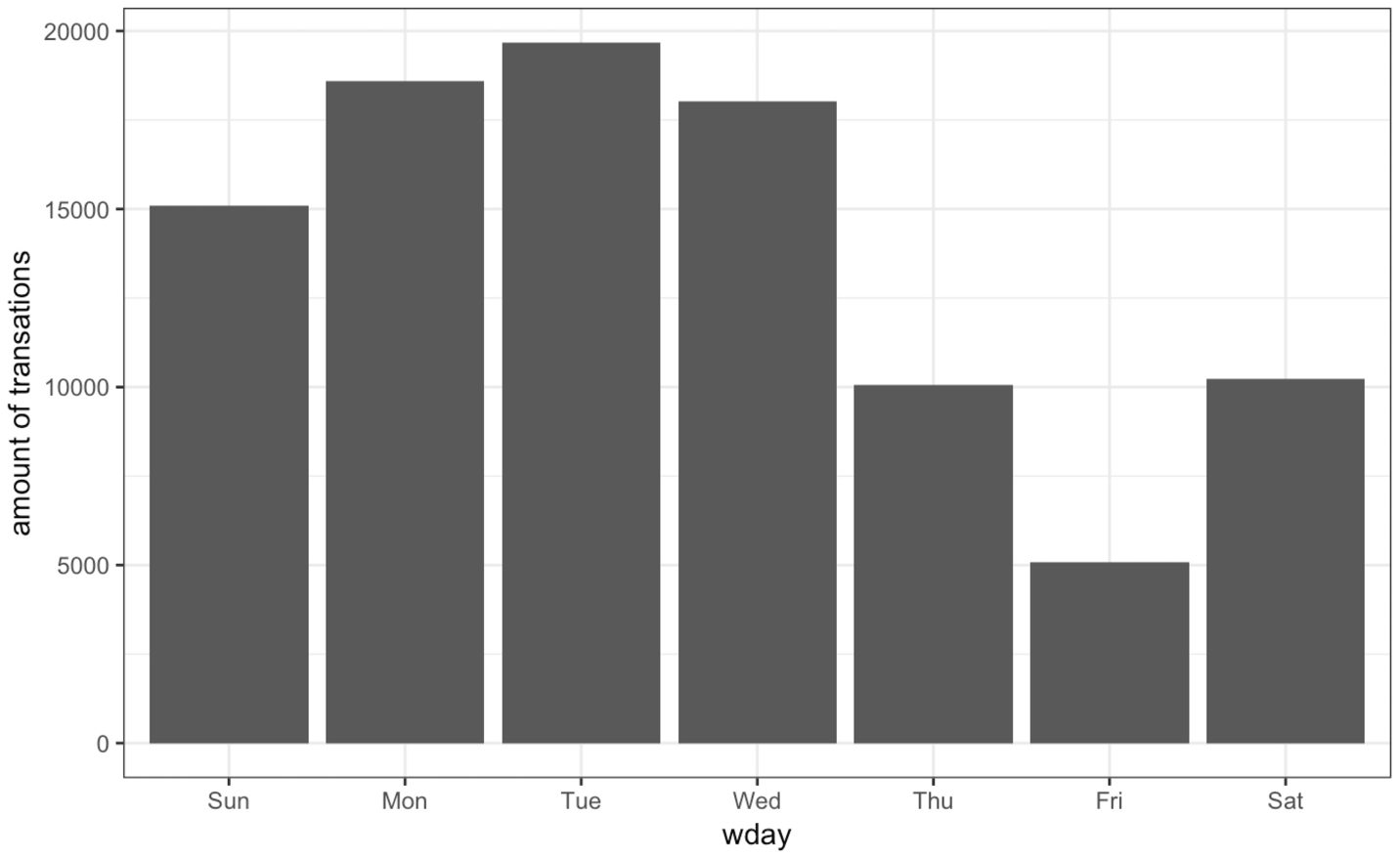
Number of transactions of most frequent date.



Number of transactions by day.

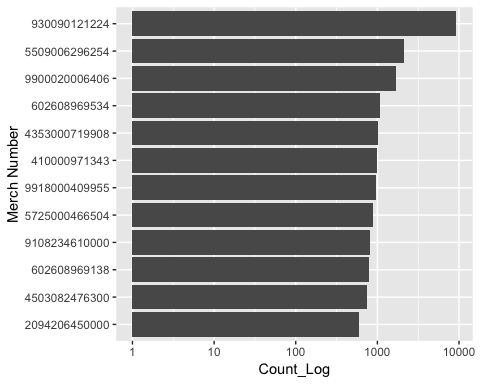


Number of transactions by weekday.



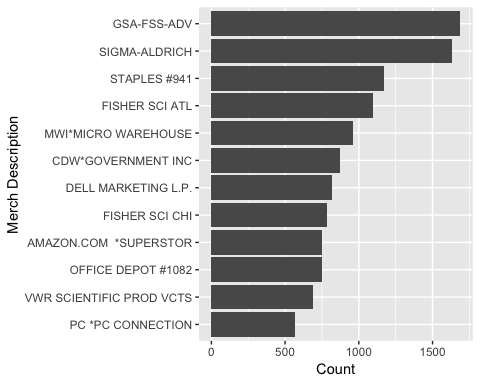
### **Merchnum**

This field is categorical, 93,378 records have a value, 96.5% populated. It has 13092 unique values. The most common field value is 930090121224.



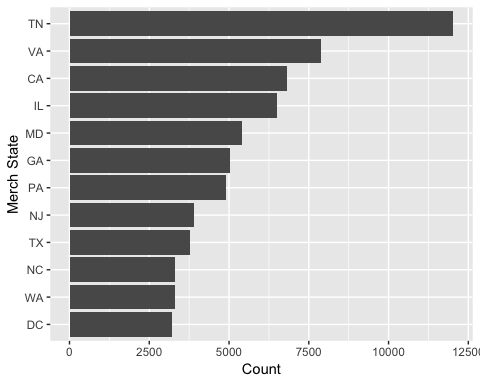
### **Merch description**

This field is categorical, each record of it has a value. It has 13,126 unique values. The most common field value is GSA-FSS-ADV.



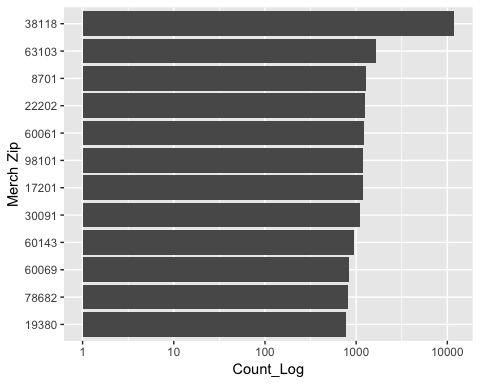
### **Merch state**

This field is categorical, 95,558 records have a value, 98.8% populated. It has 228 unique values. The most common field value is TN.



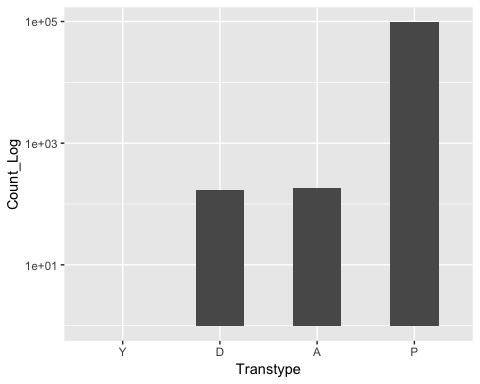
### **Merch zip**

This field is categorical, 92,097 records have a value, 95.2% populated. It has 4568 unique values. The most common field value is 38118.



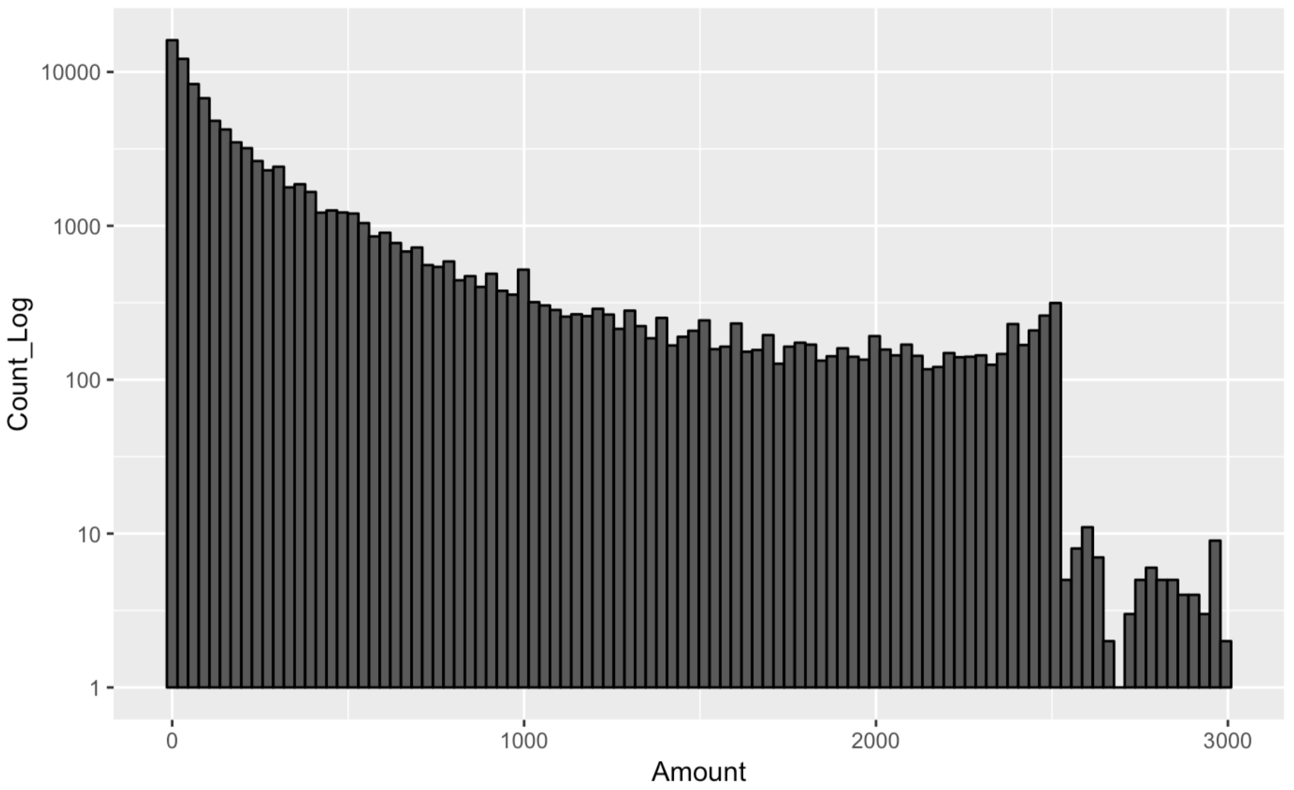
### **Transtype**

This field is categorical, each record of it has a value. It has 4 unique values. The most common field value is P.

****

### **Amount**

This field is numeric, each record of it has a value. It has 34,909 unique values. Mean is 428, standard deviation is 10,006, max value is 3,102,046, min value is 0.01.



* **Fraud**

This field is categorical, each record of it has a value. It has only 2 unique values. 0 represents the good records, while 1 means bad records which is a fraud. Most of the entries are labeled as not fraud.

