Churn Prediction - PowerCo.

Fasih Atif

1/4/2021

Background

PowerCo is a major gas and electricity utility that supplies to corporate, SME (Small & Medium Enterprise), and residential customers. The power-liberalization of the energy market in Europe has led to significant customer churn, especially in the SME segment. They have approached me in my capacity as a Data Science Consultant to find out which customers are most likely to churn and what drivers lead to their churn. I met with PowerCo and discussed the various ways we could go about resolving the problems. One of the hypotheses under consideration is that churn is driven by the customers' price sensitivities and that it is possible to predict customers likely to churn using a predictive model. I have received the raw data from the management and have begun working.

Understanding Data

The raw data consisted of 33 columns and 16096 observations in each column. The data represents the several characteristics of the company that could play an important part in their churning and retention. The data contains variables such as historical usage, dates, forecasted consumption and prices, net/gross margins and churn status. It is often said that 80% of the time is spent in data cleaning and processing while just 20% is spent on running the analytics on the clean data. This saying was applicable in our case as well. There were several data quality issues such as missing values, outliers, incorrect negative values, multicollinearity, and skewed variables. Our dependent variable would be the binary churn variable while the rest of the variables would act as the independent variables. **Figure 1** presents an overview of the raw data.

Exploring Data Analysis

The missing data (**Table 1**) was very small for majority of the variables, so i was able to easily replace the missing values with mean and median approximations. Any column that had more than 30% missing observations were removed. I then moved on to conducted Feature Engineering to draw more useful variables from our existing ones. I converted dates into monthly durations to make them more useful (**Figure 2**). After getting done with our initial data cleaning and feature Engineering, i checked the distributions of the variables through histograms (**Figure 1-9**).

Our exploratory data analysis shows that around 10% of the of total customers have churned (**Figure 3**). All Consumption related variables (cons_12m ,cons_last_month,imp_cons,cons_gas_12m) and 2 of the forecast variables (forecast_cons_12m,forecast_meter_rent_12m) are extremely right skewed (**Figure 4,5**). The values on the higher end and lower ends of the distribution are potential outliers. Next we checked a non-parametric estimator loess smoother to have a general idea about the functional form between a variable and churn. We will specifically focus on those variables who had skewed distributions (Figure). As expected, the functional forms are very non linear and hence we took log of the 6 variables (4 consumption and 2 forecast variables) to give the loess line a more linear shape (**Figure 6,7**). Before i took the logs, i converted the negative values to NA values in the consumption related variables. Firstly it will allows us to take logs and secondly these negative values seem to be corrupted data. The energy consumption variables negative state means that the customers were now returning/creating the energy instead of buying from PowerCo. The missing values were then umputed with mean values. The transformation of the log variables can be seen in (**Figure 9**). I used boxplots to tell us more about the outliers and what their values are. We then removed these values and replaced them with the mean of the respective columns values excluding the outliers.

Multicollinearity and Dummy Variables

When you have two independent variables that are very highly correlated, you definitely should remove one of them because you run into the multicollinearity conundrum and your regression model's regression coefficients related to the two highly correlated variables will be unreliable. I checked for multicollinearity between the explanatory variables. I drew a correlated matrix (**Figure 10**) to observe which pair of variables were highly correlated. Once I noticed some highly correlated pairs, we calculated the Variance Inflation Factor (VIF) (**Table 2**) to confirm the correlation and remove one of the variable from the correlated pairs.This was done to reduce multicollinearity. Categorical columns ('channel_sales' and 'has_gas') were converted into dummy columns and the reference columns were removed.

Machine Learning, Probabilities, and Predictions

I devised the following equation that will be used in the models:

$$
\begin{gathered}
churn = \beta_0 + \beta_1 cons\_12m_i + \beta_2 log(cons\_last\_month)_i + \beta_3 log(imp\_cons)_i + \beta_4 forecast\_price\_energy\_p1_i + \\
\beta_5 forecast\_price\_energy\_p2_i + \beta_6 forecast\_price\_pow\_p1_i + \beta_7 forecast\_discount\_energy_i + \beta_8 log(forecast\_meter\_rent\_12m_i) + \\
beta_9 months\_active_1 + \beta_{10} months\_modif_i + \beta_{11} months\_renewal_i + \beta_{12} channel\_usil_i + \beta_{13} channel\_lmke_i + \beta_{14} channel\_ewpa_i + \\
\beta_{15} channel\_foos_i + \beta_{16} channel\_sddi_i + \beta_{17} channel\_epum_i + \beta_{18}\_months\_end + i + \beta_{19} b\_prod\_act_i + \beta_{20}\_pow\_max_i \\
+ \beta_{21}\_has\_gas\_1_i + \beta_{22}\_net\_margin_i + \beta_{23} margin\_net\_pow\_ele_i + \beta_{24} log(forecast\_cons\_12m_i)
\end{gathered}
$$

Linear probability Model

Now that our data is ready for analysis we will start off with same basic probability models such as the Linear Probability Model.We split the data into train and test groups in a 75/25 ratio. The produced coefficients for this model can been see in (**Table 3**). We used the model to predict on the test set. The probability distribution is centered around 1.1 (**Figure 11**). Probability values are between 0 and 1 but in our model, half of our probabilities are greater than 1. This is a drawback of the model. Nevertheless, we will try to study the characteristics of the top 1% customers who have the highest and lowest probability of leaving. The results are shown in (**Table 4**). The AIC for the model was 4893.13 and BIC was 5092.895.

Logistic Regression

Logistic Regression belongs to the family of generalized linear models. It is a binary classification algorithm used when the response variable is dichotomous (1 or 0). Inherently, it returns the set of probabilities of target class. But, we can also obtain response labels using a probability threshold value.

Logistic Regression with Logit/Probit models will suit our research better as they limit the probability between 0 and 1. We first ran a logit model with all variables and looked at the results. The most insignificant variables were removed and a second logit model was performed. The second logit model performed slightly better with lesser coefficients. We calculated the marginal difference for logit model. Then we ran Probit model and Probit marginal difference models to see if it performed better than logit. The coefficients of the 4 models can be compared in (**Table 5**).

The AIC for logit model is 7593.75 and BIC was 7786.119. The AIC for Probit model was 7595.39 and BIC was 7787.75. The logit model performed better overall.

The confusion matrix is shown below:

```
           Reference
Prediction    0    1
         0 3625  398
         1    0    0
```

It shows that our model predicted correctly with an accuracy of 90.1%. We also drew an ROC curve (**Figure 12**) and computed the AUC validated which turned out to be 0.5. The model can be greatly improved if more work is done on this.

XGBoost Model

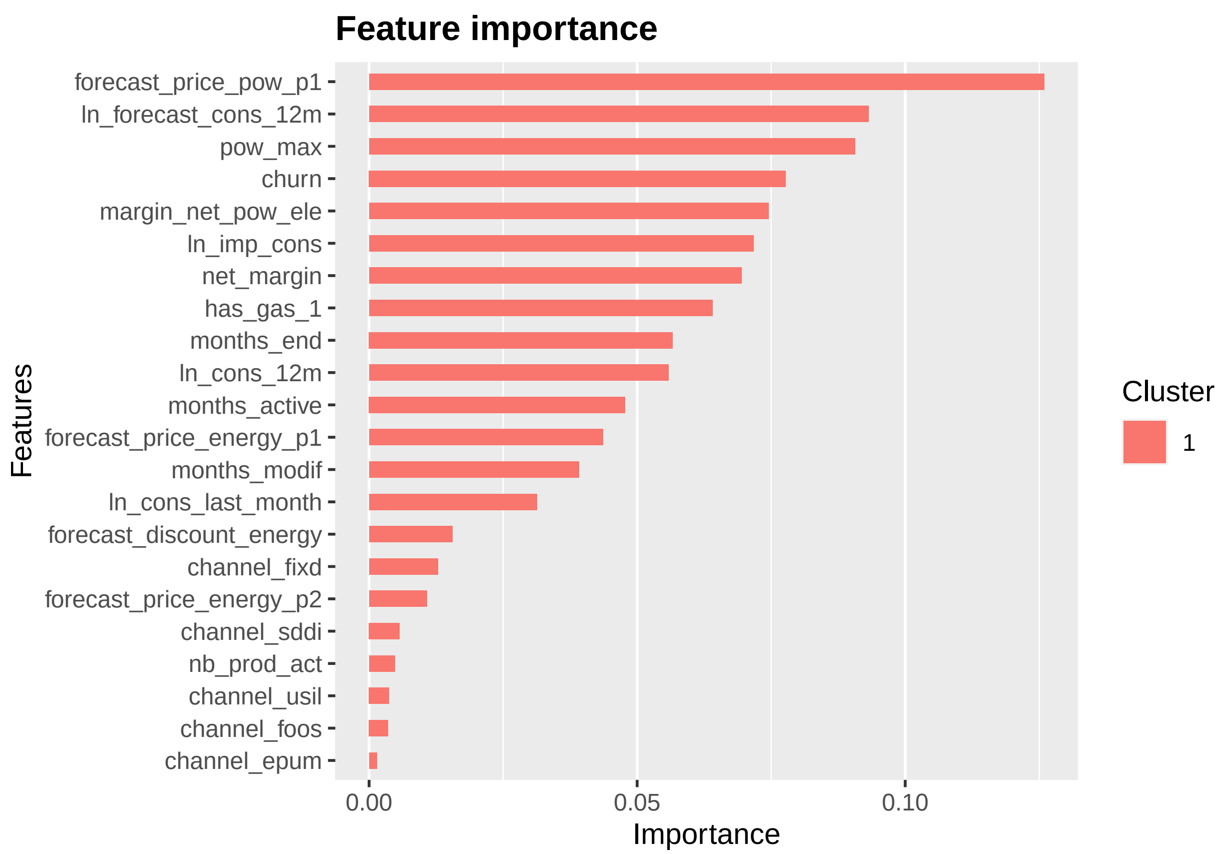
It is known for its good performance as compared to all other machine learning algorithms and has been a winner in many Data Science competitions. I wanted to run and try a bigger complex model to see if they perform any better or is simpler the better? I set the parameters for 10 k folds and 500 rounds and used cross validation to check on which round we would get the lowest test mean error. The model stopped on round 47 and and achieved an accuracy of 90.32%. I used 47 rounds in our main XGBoost training model and predicted our test set. The accuracy on our test set was approximately 90.73$ with confidence intervals of 89 and 91%. The confusion matrix for the XGboost model is below:

```
           Reference
Prediction    0    1
         0 3612  360
         1   13   38
```

.

We can see that the model correctly predicted that a majority of the customers wont churn and in reality they didnt. This shows that our model has a very high sensitivity of 99% based on our test set.

Its not exactly easy to interpret information from XGBoost model without processing it further but we can use the data to show which are the most influential variables:



The chart shows that forecast_price_pow_p1 , ln_forecast_cons_12m, pow_max are some of the most important variables that the management should look at when dealing with customers.

The ROC (**Figure 13**) of the XGboost model fares better than the logit model as the AUC comes out to be 0.5684.

Conclusion

PowerCo appraoched me to help them predict which customers are most likely to leave. I did some exploratory data analysis and transformed the variables. I then conducted some machine learning classification and analysed probabilities. Both Logit and XGboost models performed very well and had accuracy above 90%. However, the AUC , AIC, and BIC scores were beter in XGBoost model. I would recommend using both logistic and XGboost models. Logistic has comparatively easier interpretations and easier to use.

APPENDIX A

Table 1 - Missing Values

	na_count	na_percent
id	0	0.00
activity_new	9545	59.30
campaign_disc_ele	16096	100.00
channel_sales	4218	26.21
cons_12m	0	0.00
cons_gas_12m	0	0.00
cons_last_month	0	0.00
date_activ	0	0.00
date_end	2	0.01
date_first_activ	12588	78.21
date_modif_prod	157	0.98
date_renewal	40	0.25
forecast_base_bill_ele	12588	78.21
forecast_base_bill_year	12588	78.21
forecast_bill_12m	12588	78.21
forecast_cons	12588	78.21
forecast_cons_12m	0	0.00
forecast_cons_year	0	0.00
forecast_discount_energy	126	0.78
forecast_meter_rent_12m	0	0.00
forecast_price_energy_p1	126	0.78
forecast_price_energy_p2	126	0.78
forecast_price_pow_p1	126	0.78
has_gas	0	0.00
imp_cons	0	0.00
margin_gross_pow_ele	13	0.08
margin_net_pow_ele	13	0.08
nb_prod_act	0	0.00
net_margin	15	0.09
num_years_antig	0	0.00
origin_up	87	0.54
pow_max	3	0.02
churn	0	0.00

**Table 2 - Variance Inflation Factor (VIF) **

	VIF
forecast_discount_energy	1.256417
nb_prod_act	1.242218
num_years_antig	41.724615
contract_duration	2733.747467
months_active	2579.186182
months_end	90.275012
months_modif	1.410913
months_renewal	3.979559
channel_epum	1.005128
channel_ewpa	1.378587
channel_fixd	1.002336
channel_foos	2.301143
channel_lmke	1.574024
channel_sddi	1.011849
channel_usil	1.569362
has_gas_1	15.877874
forecast_price_energy_p1	3.285631
forecast_price_energy_p2	2.965002
forecast_price_pow_p1	5.430790
margin_gross_pow_ele	151.607239
margin_net_pow_ele	151.419021
net_margin	2.024534
pow_max	2.807251
ln_cons_12m	3.056839
ln_cons_gas_12m	15.732625
ln_cons_last_month	4.963743
ln_imp_cons	4.182487
ln_forecast_cons_12m	2.632955
ln_forecast_meter_rent_12m	2.326036

Table 3 - Linear Probability Model Coefficients

```
Residuals:
     Min       1Q   Median       3Q      Max 
-0.26768 -0.12172 -0.09047 -0.05592  1.02581 

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)    
(Intercept)                1.309e+00  1.420e-01   9.217  < 2e-16 ***
forecast_discount_energy  -1.742e-04  5.833e-04  -0.299 0.765233    
nb_prod_act                1.062e-04  2.005e-03   0.053 0.957782    
months_active             -1.021e-03  1.829e-04  -5.583 2.41e-08 ***
months_end                -1.473e-04  1.396e-03  -0.106 0.915946    
months_modif              -1.089e-04  1.046e-04  -1.041 0.297875    
months_renewal            -1.792e-03  1.349e-03  -1.328 0.184131    
channel_epum              -1.378e-01  1.714e-01  -0.804 0.421145    
channel_ewpa              -2.874e-02  1.324e-02  -2.170 0.030026 *  
channel_fixd              -1.528e-01  2.096e-01  -0.729 0.465908    
channel_foos               1.430e-02  8.180e-03   1.748 0.080552 .  
channel_lmke              -3.670e-02  9.951e-03  -3.688 0.000227 ***
channel_sddi              -1.048e-01  1.214e-01  -0.863 0.388020    
channel_usil              -2.293e-03  1.181e-02  -0.194 0.846085    
has_gas_1                 -1.970e-02  7.739e-03  -2.546 0.010910 *  
forecast_price_energy_p1  -5.016e-01  2.513e-01  -1.996 0.045924 *  
forecast_price_energy_p2   2.826e-01  9.583e-02   2.949 0.003197 ** 
forecast_price_pow_p1     -2.780e-03  3.227e-03  -0.861 0.389000    
margin_net_pow_ele         2.073e-03  2.567e-04   8.074 7.47e-16 ***
net_margin                 1.928e-05  2.702e-05   0.713 0.475627    
pow_max                   -1.424e-03  9.597e-04  -1.484 0.137778    
ln_cons_12m                2.325e-03  2.646e-03   0.879 0.379547    
ln_cons_last_month        -3.414e-03  1.550e-03  -2.202 0.027662 *  
ln_imp_cons                1.391e-04  2.215e-03   0.063 0.949929    
ln_forecast_cons_12m       1.849e-03  2.938e-03   0.630 0.528985    
ln_forecast_meter_rent_12m 2.083e-04  3.142e-03   0.066 0.947144    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.296 on 12047 degrees of freedom
Multiple R-squared:  0.02126,	Adjusted R-squared:  0.01923 
F-statistic: 10.47 on 25 and 12047 DF,  p-value: < 2.2e-16
```

Table 4 - Top and Bottom 1% companies

statistics	forecast_price_pow_p1	ln_forecast_cons_12m	pow_max	margin_net_pow_ele	net_margin
mean	40.6856298	7.112611	21.311415	45.884750	221.741
median	40.6067010	7.700162	20.604131	46.985000	217.987
sd	0.1954818	1.282639	2.544473	6.499514	172.243

statistics	forecast_price_pow_p1	ln_forecast_cons_12m	pow_max	margin_net_pow_ele	net_margin
mean	42.352418	7.111064	18.24505	0.7714634	225.1347
median	43.088515	7.771301	20.60413	-0.3600000	217.9870
sd	1.789522	1.944410	4.74957	8.3326387	141.1079

TABLE 5 - Logistic Regression Model Summaries

(From left to right: logit__model__2, logit__marg, probit__model__1,probit__marg)

	Model 1	Model 2	Model 3	Model 4	Model 5
Constant	0.204	-1.213*		-0.091	
	(1.676)	(0.493)		(0.855)	
forecast_discount_energy	-0.002	-0.004	-0.000	-0.001	-0.000
	(0.006)	(0.006)	(0.001)	(0.003)	(0.001)
nb_prod_act	-0.032		-0.003	-0.018	-0.003
	(0.049)		(0.002)	(0.023)	(0.002)
months_active	-0.013**	-0.013**	-0.001**	-0.006**	-0.001**
	(0.002)	(0.002)	(0.000)	(0.001)	(0.000)
months_end	-0.003		-0.000	-0.002	-0.000
	(0.018)		(0.001)	(0.009)	(0.001)
months_modif	-0.002	-0.001	-0.000	-0.001	-0.000
	(0.001)	(0.001)	(0.000)	(0.001)	(0.000)
months_renewal	-0.021	-0.018*	-0.002	-0.010	-0.002
	(0.018)	(0.009)	(0.001)	(0.009)	(0.001)
channel_epum	-11.868		-0.099**	-3.884	-0.099**
	(300.302)		(0.003)	(81.864)	(0.003)

APPENDIX B

Figure: 1 - Data Overview

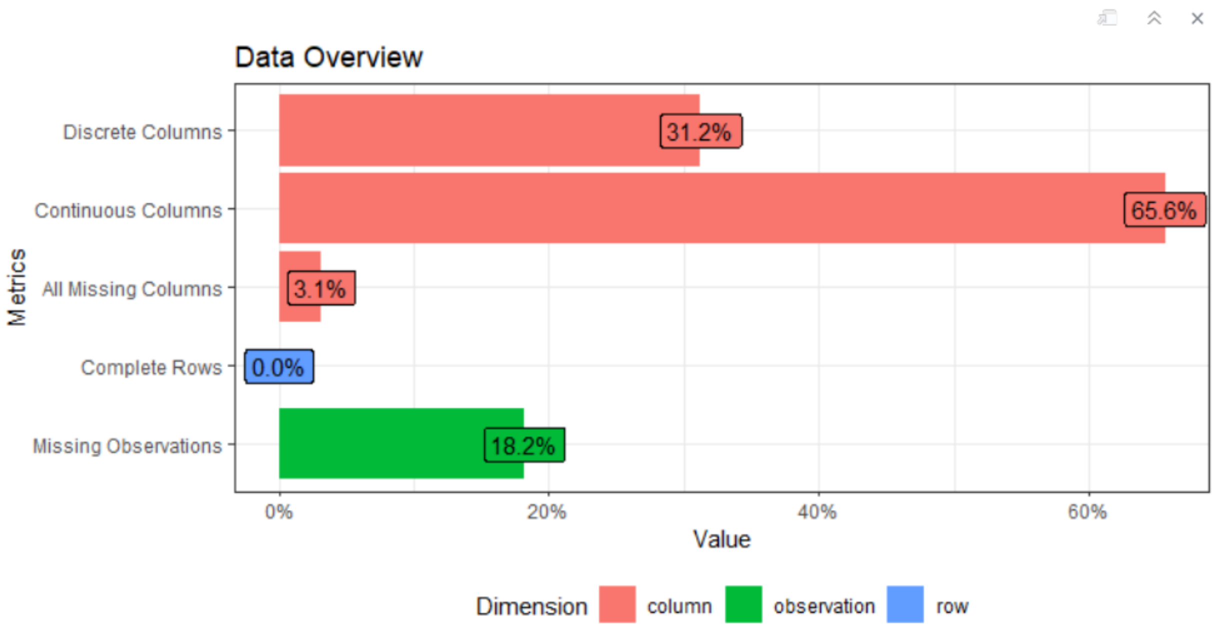


Figure: 2 - Monthly Duration charts

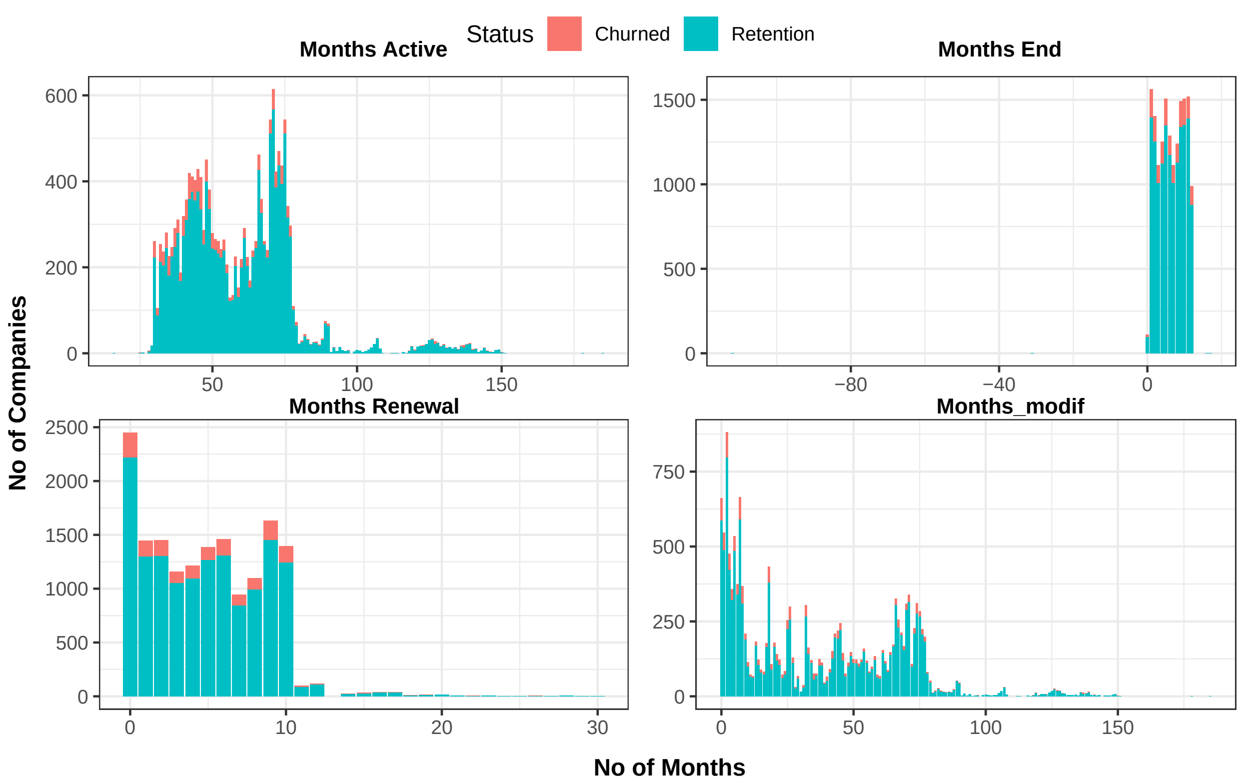


Figure: 3 - Churn Rate

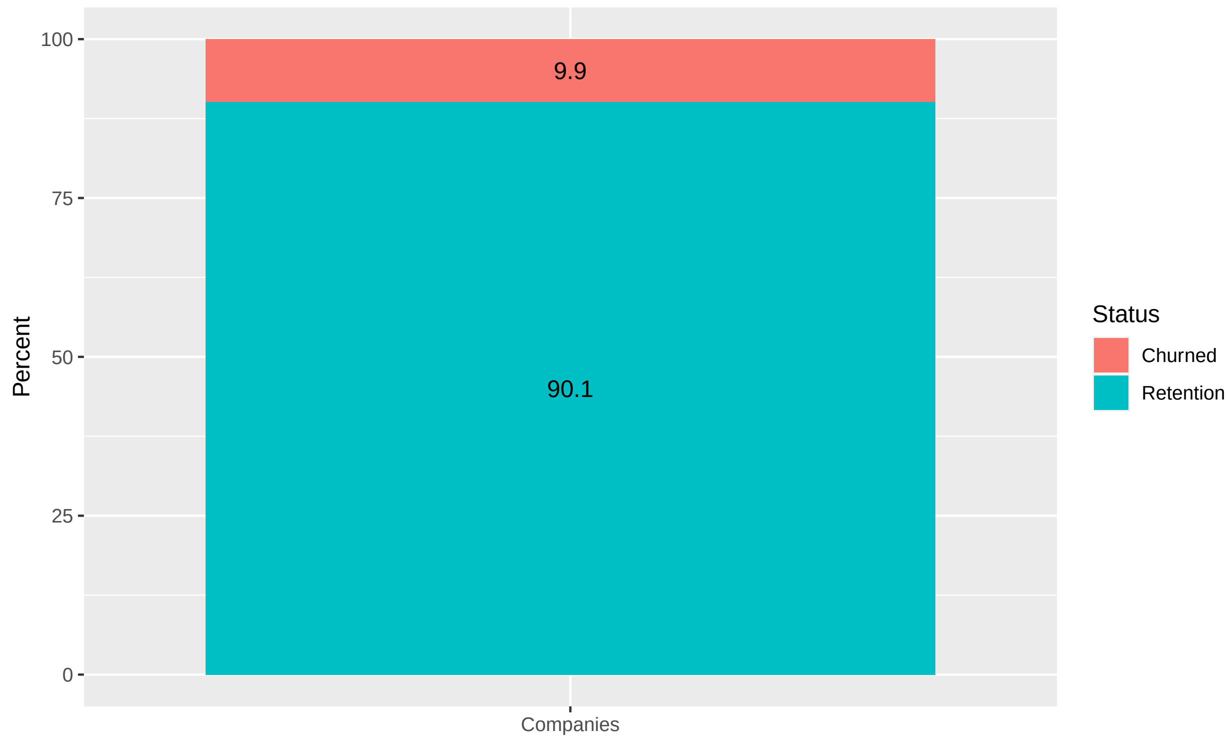

100
75
50
25
0
Percent
9.9
90.1
Companies
Status
Churned
Retention

Figure: 4 - Consumption Variables Exploration

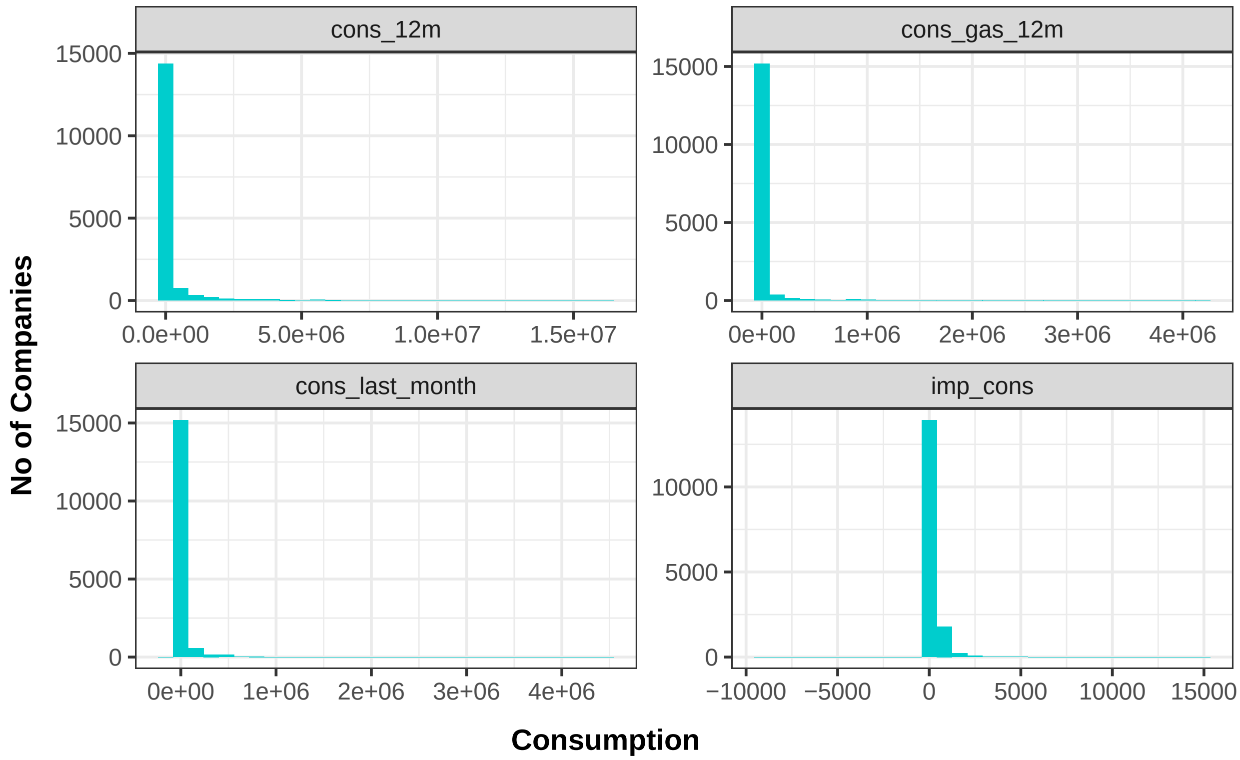


Figure: 5 - Forecast Variables Exploration

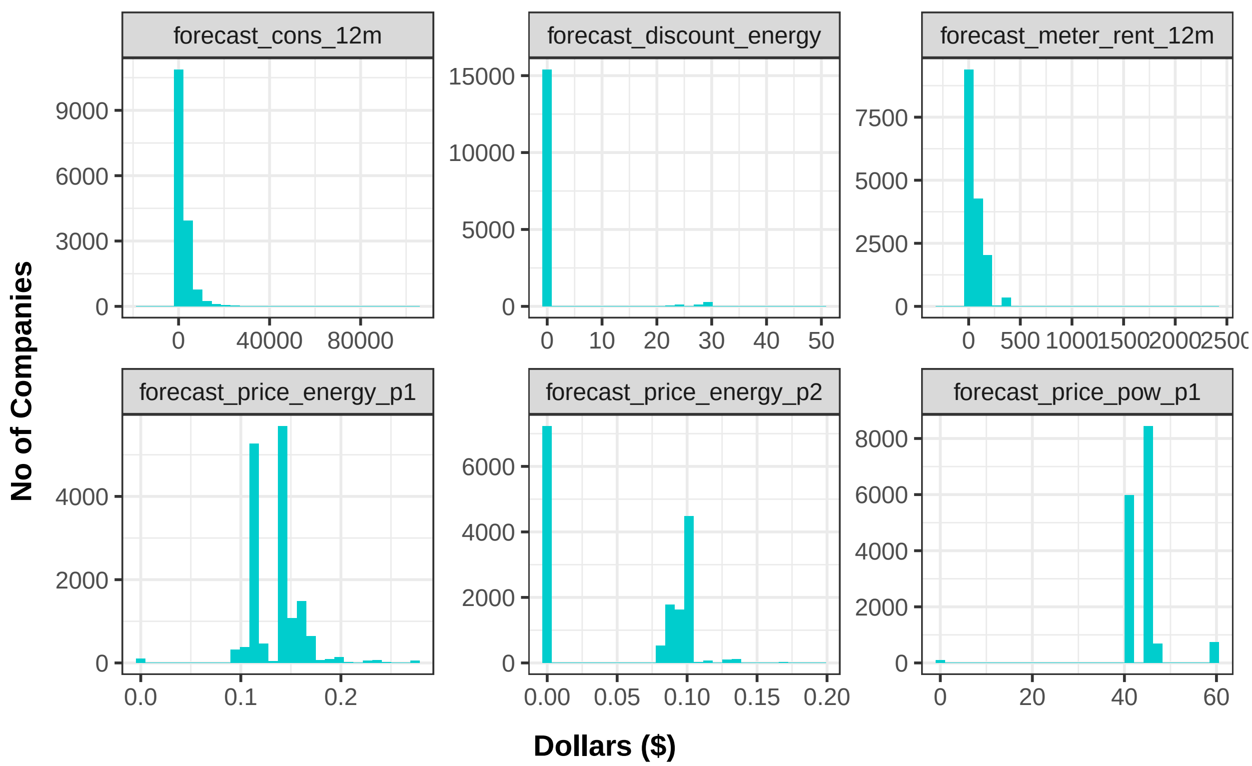


Figure: 6 - Margin Variables Exploration

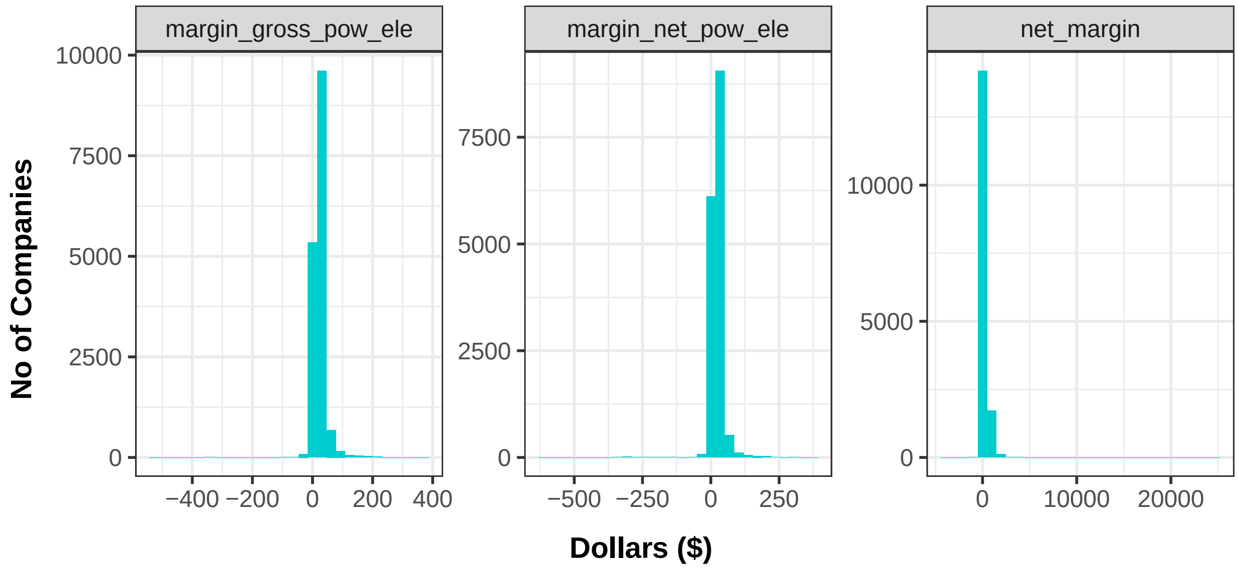


Figure: 7 - Other Variables Exploration

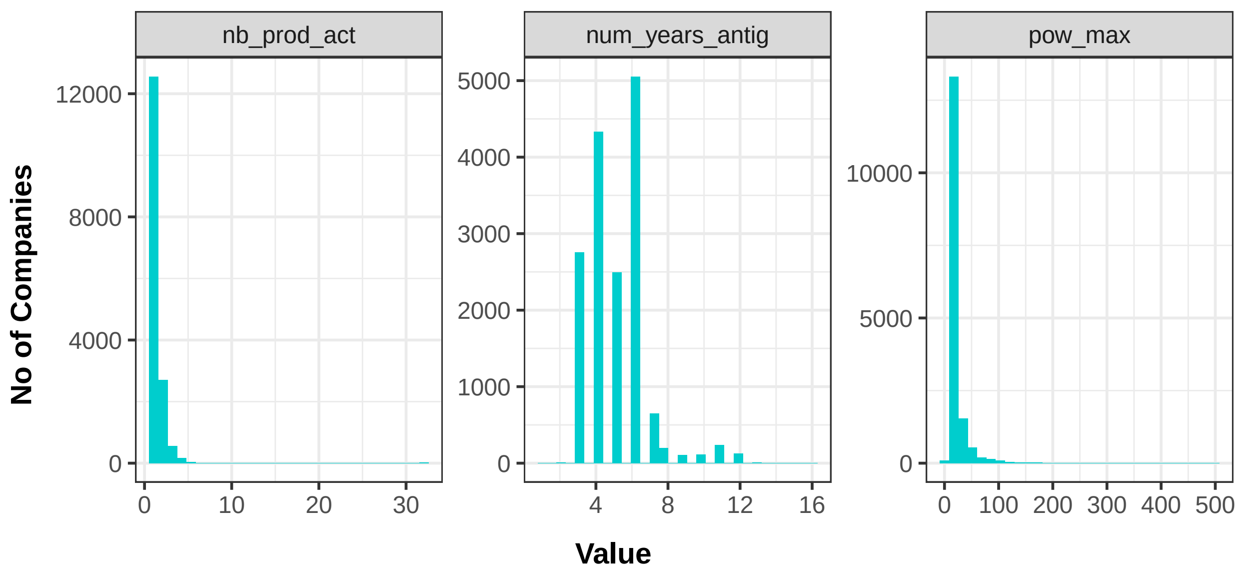
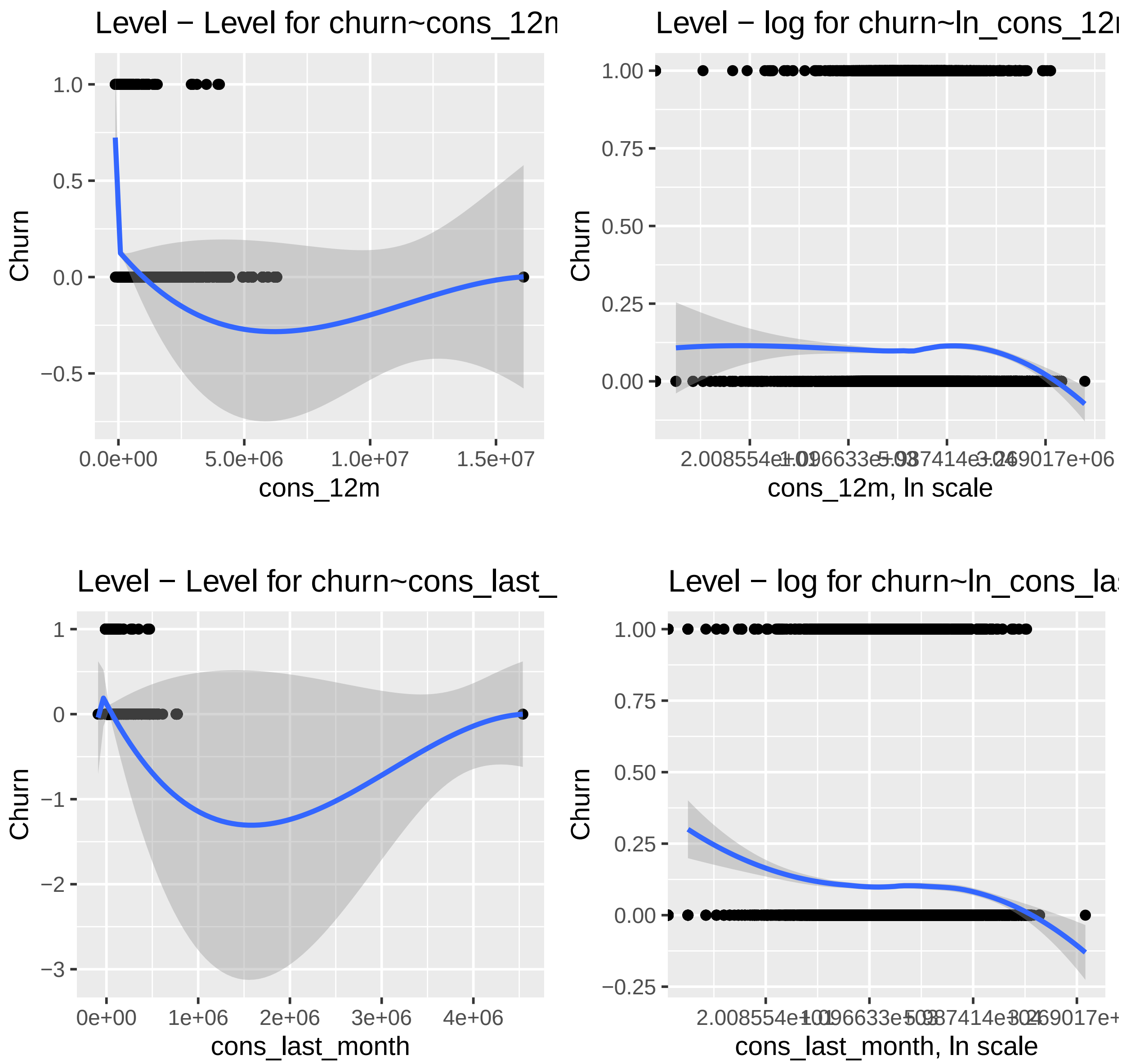
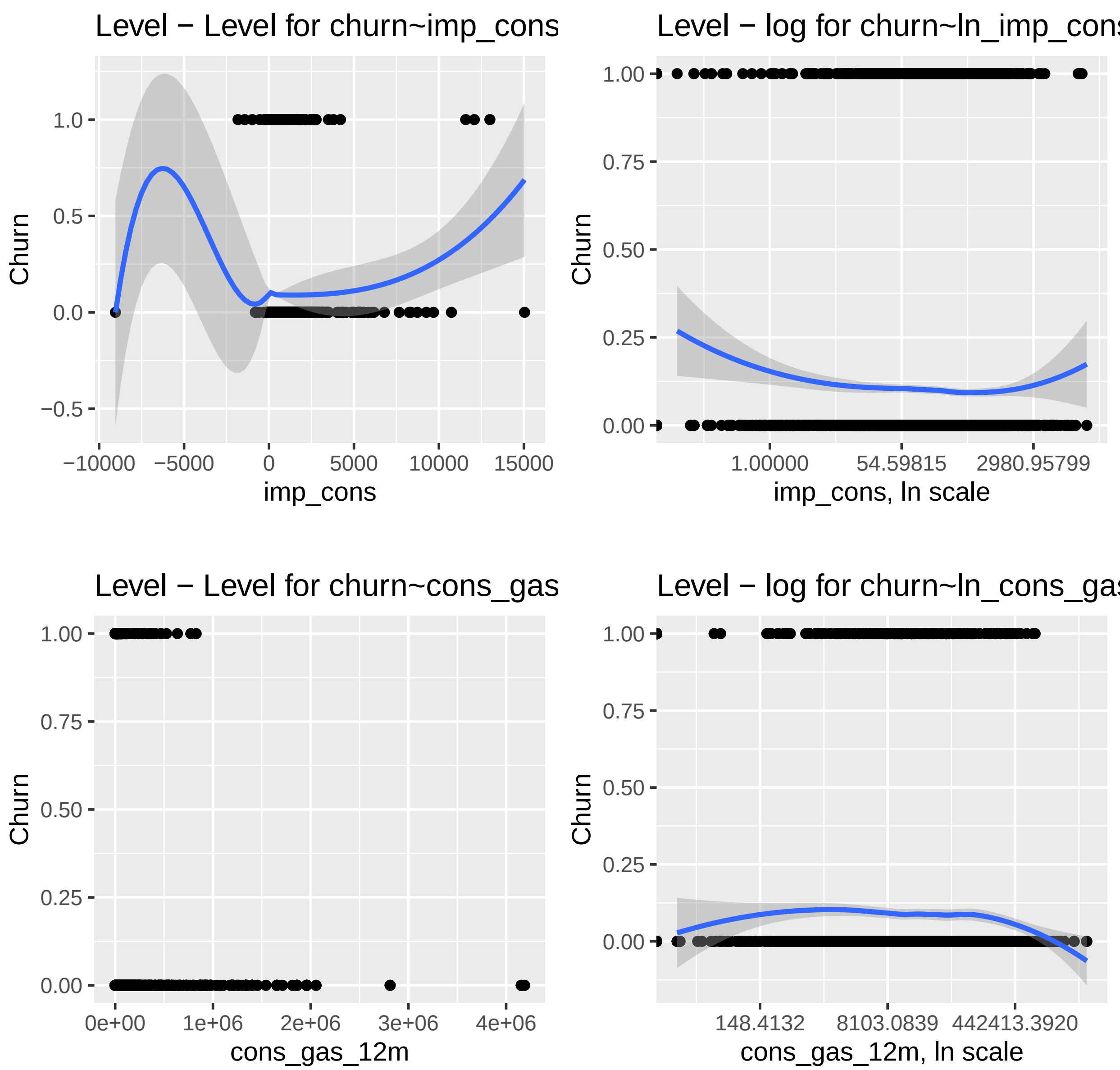


Figure: 8 - Loess/Scatterplot of Variables





Level − Level for churn~imp_cons
Churn
imp_cons
−10000
−5000
0
5000
10000
15000
1.0
0.5
0.0
−0.5
Level − log for churn~ln_imp_cons
imp_cons, ln scale
1.00000
54.59815
2980.95799
1.00
0.75
0.50
0.25
0.00
Level − Level for churn~cons_gas
cons_gas_12m
0e+00
1e+06
2e+06
3e+06
4e+06
Level − log for churn~ln_cons_gas
cons_gas_12m, ln scale
148.4132
8103.0839
442413.3920

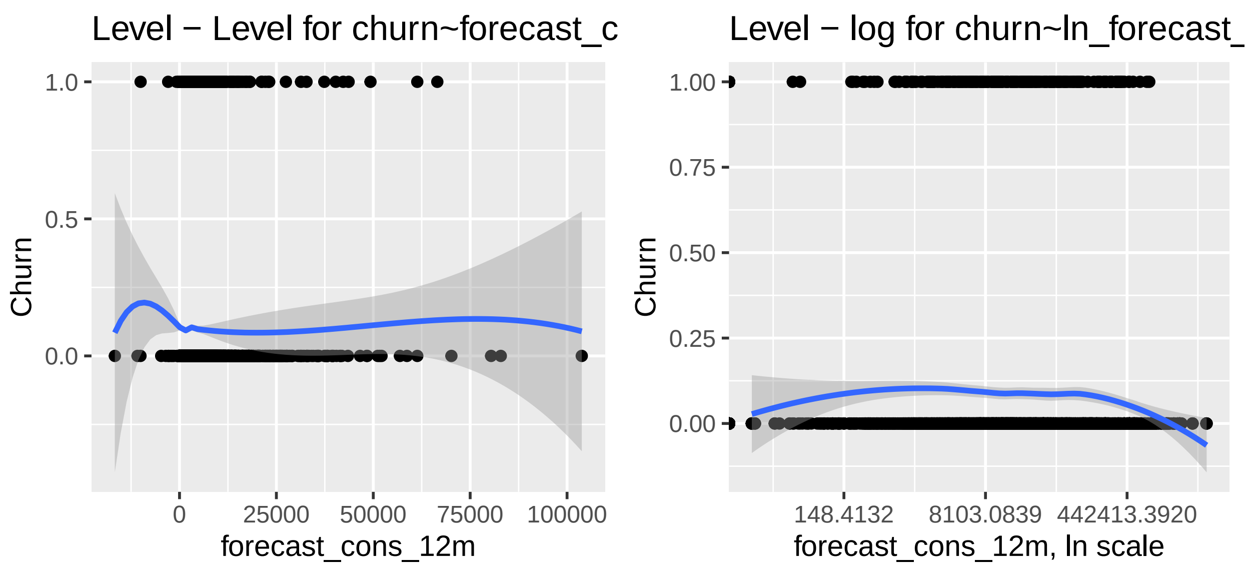


Figure: 9 - Transformed Variables Exploration

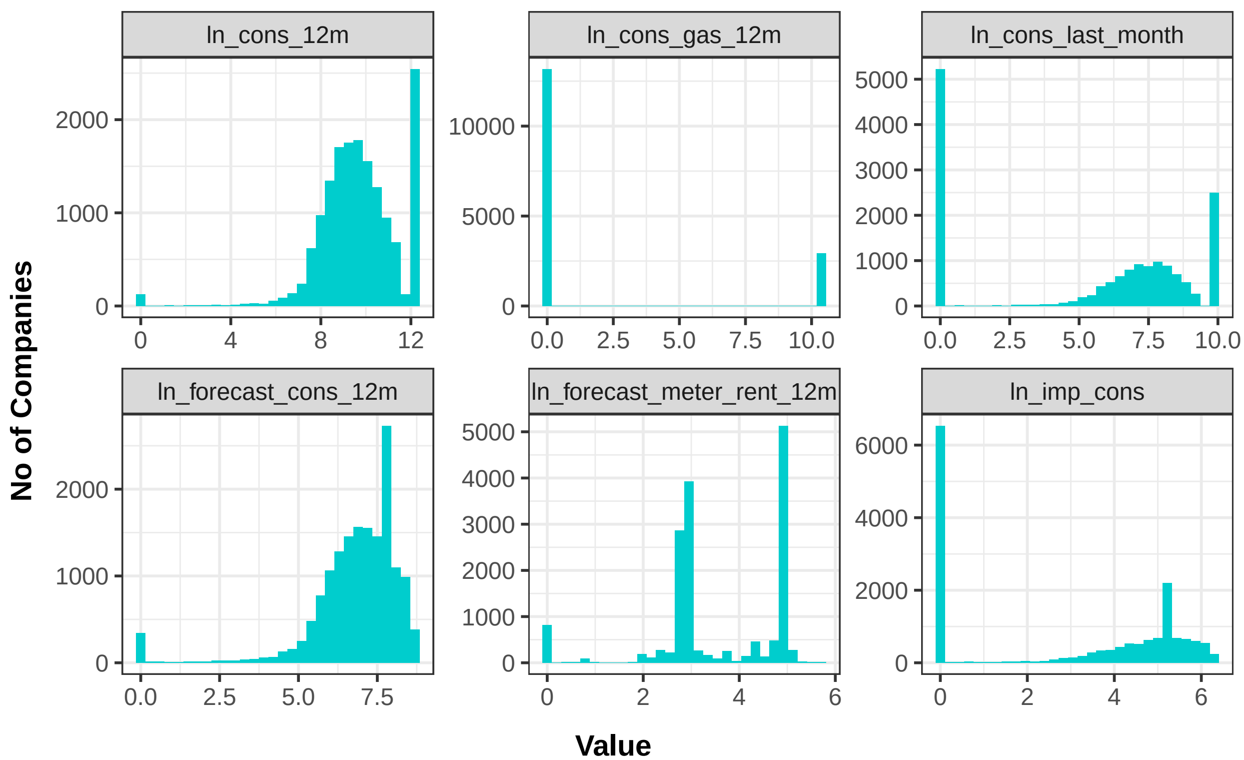


Figure 10 - Correlation Matrix

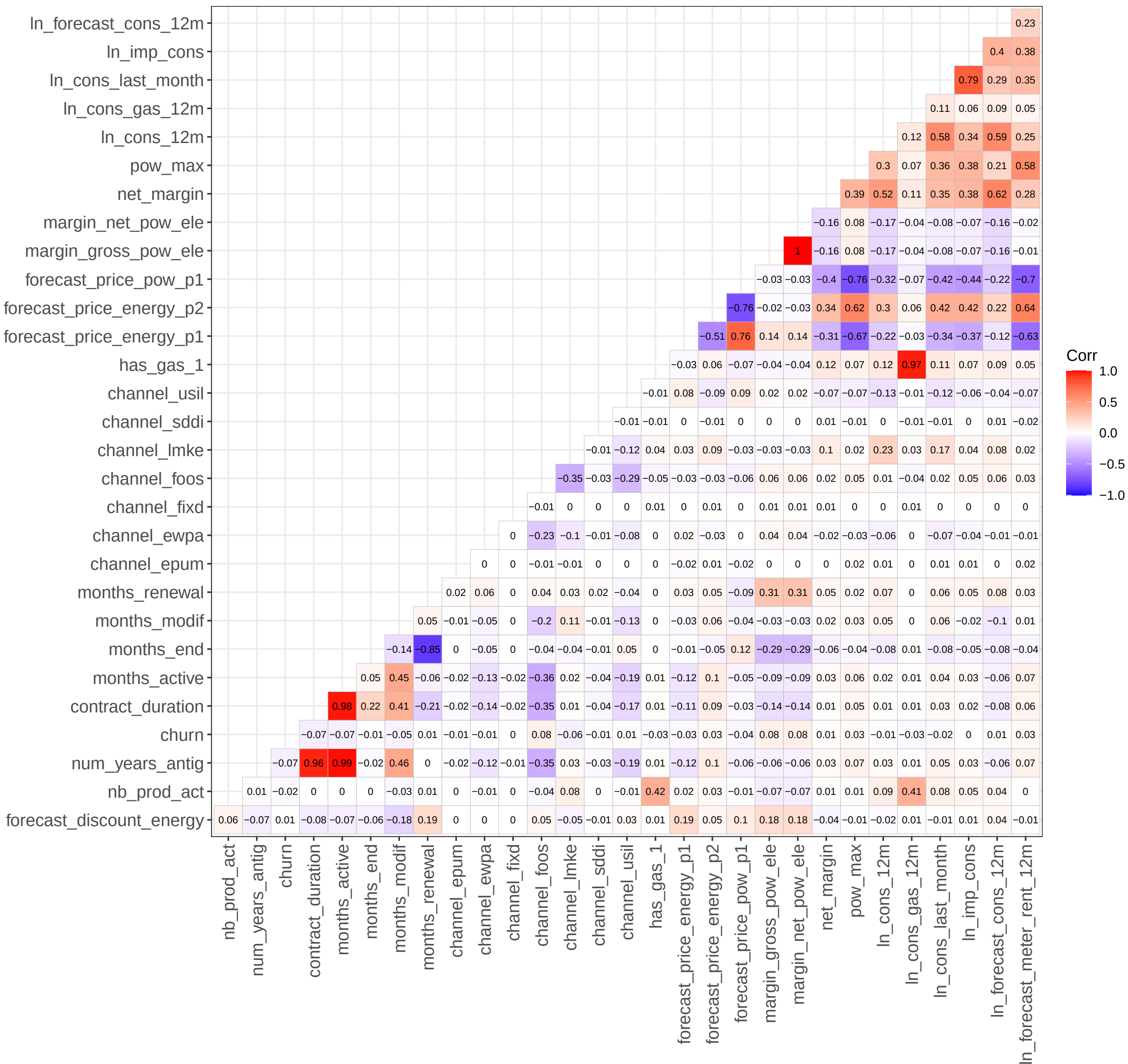


Figure 11 - Probability Distribution

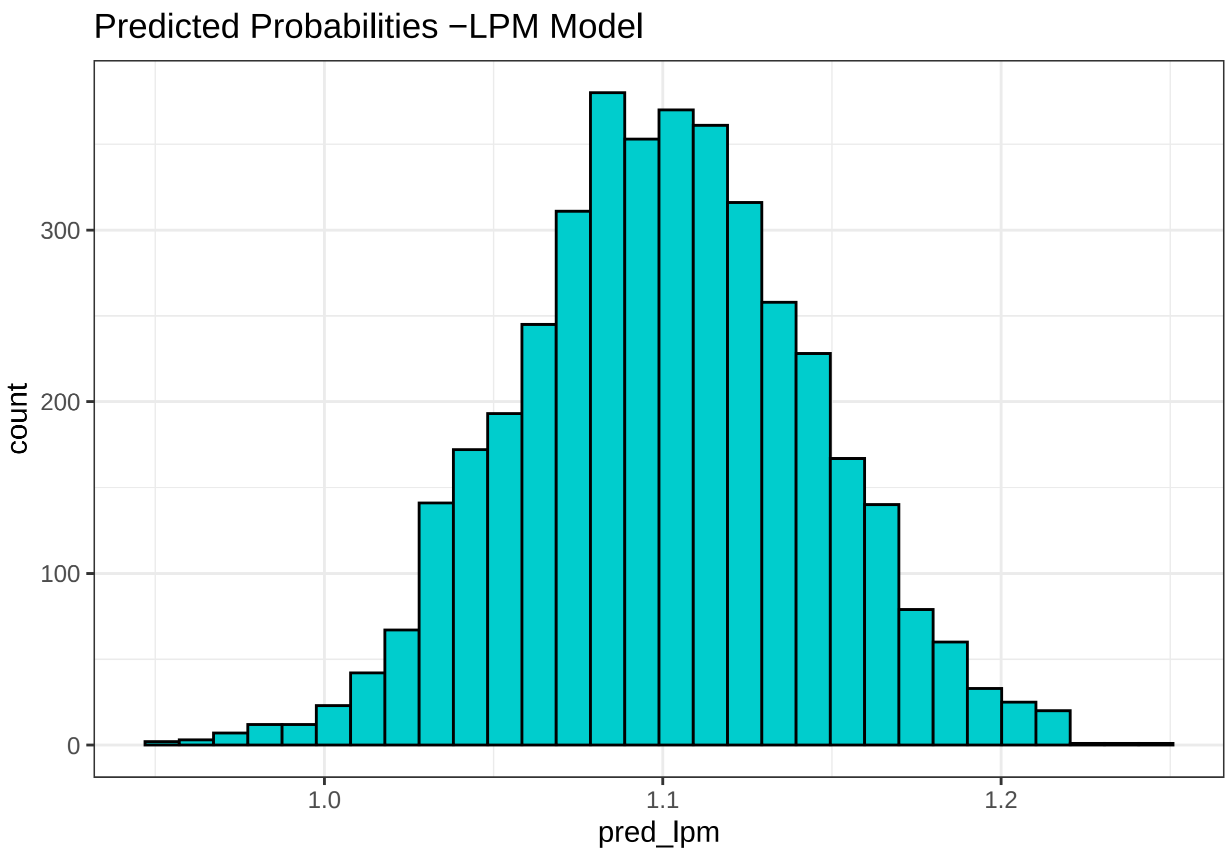


Figure 12 - ROC Plot

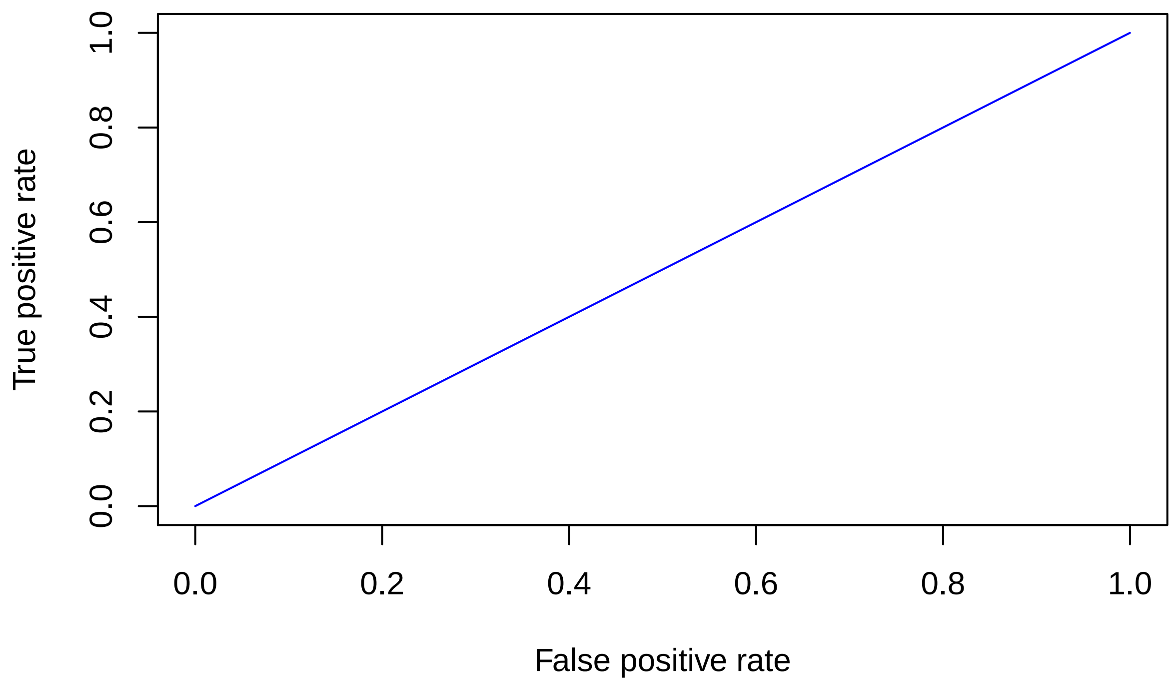
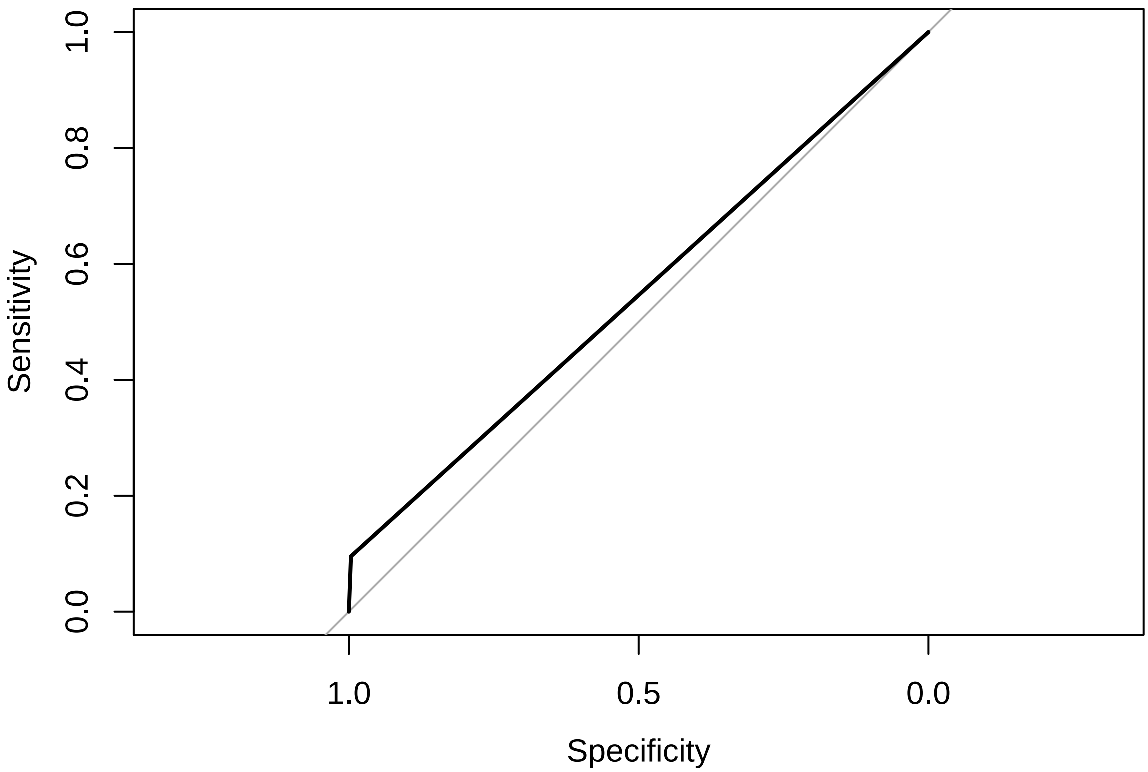


Figure 13 - ROC Plot XGBoost


Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
1.0
0.5
0.0
Specificity