Catalog Sales Data Mining

Zachary Anglin
Steven Lin
Jason Silverman

Fall 2014

EXECUTIVE SUMMARY

The goal of the project was to predict how much money each customer spent during the period of interest (September 1, 2012 – December 1, 2012) based on historical data. In order to accomplish this objective, a classification model was built to predict the buyers and multiple regression model was built to predict the amount of sales.

The most significant variables that were found to be key predictor of responder vs. non-responders were the years since last purchase and the orders this year (2012). These results are expected since these variables are an indicator of recency of last purchase, which is known in data base marketing to be one of the best predictors for deciding whether a customer will respond to a catalog or not. Other key predictors that were identified were the order last year, orders 2 years ago, the years since the customer was added to the file, an indicator of whether the customer had fall orders, the "life-to-date" (LTD) dollars spent, the consistency of past purchase measured by orders, and the interactions of "life-to-date" orders with spring order and years since added to the file.

For the sales amount, the key predictors were found to be indicators of orders for this year and the past three years, the sales for this years and the past three years, the "life-to-date" dollars spent, the consistency of past purchase captured by the interaction of sales this year and sales last year, and the interaction of LTD dollars spent with the years since added to the file.

The strategy of targeting the top 5,000 customers (out of the +51,000 customers in test data) based on expected sales results in a payoff of $112,905, which represents about 50% of the maximum possible amount.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

1.1 Problem Statement

On September 1, 2012, a retail company sent a catalog mailing to its existing population of customers. On December 1, 2012, the company recorded the number of dollars spent by each customer in the three-month period since the catalog mailing. The client provided us with a dataset including this information, along with several other variables related to prior customer purchases that might be useful in predicting future purchases.

1.2 Overall Approach

The task at hand, as specified by the client, was for our team of statistical analysts to predict how much money each customer spent during the period of interest (September 1, 2012 – December 1, 2012). The dataset provided by the client was randomly broken down into two subsets, a training set and a test set, so that the models used to make these predictions could be trained and tested. In order to complete the task at hand, the team broke the problem into two sub-problems: firstly, we sought to be build a logistic regression model that would return each customer's probability of making a purchase during the period of interest, and secondly, we sought to build a multiple regression model that would return each customer's expected amount of purchases in dollars, given that the customer would indeed make a purchase in the period of interest. Finally, once the models had provided us with both of these prediction values for each customer, our final step was simply to multiply these values together to determine the expected amount of purchases in dollars for each customer, during the period of interest.

The team had approached the assignment with a couple of a priori hypotheses regarding the predictors to be included in the logistic regression models and multiple regression models, respectively. With regards to the logistic models, the team expected that variables signifying recency of last purchase and consistency of past purchases would be included as significant predictors in each of the finalist models. With regards to the multiple regression models, the team expected only that the predictors included in the finalist models would differ substantially from the predictors in the best logistic model. Specifically, the team expected that historical "sales" variables (e.g. **slstyr**) would be more prevalent in the multiple regression models, whereas historical "order" variables (e.g. **ordtyr**) would be more prevalent in the logistic model.

1.3 Outline of Report

In the next section of the report, entitled Model Fitting, the team will discuss how we cleansed the data, created new variables, conducted exploratory data analyses, and created interaction terms. In the latter stages of this section, we will also discuss the processes through which we arrived at our finalist logistic regression and multiple regression models, respectively, along with the diagnostics we performed on each type of model. In the following section of the report, entitled Model Validation, the team will explain how we validated the models against our test data set, and we will also explain how we ultimately assessed the efficacy of our models using statistical and financial criteria. Finally, in the Conclusions section of the report, the team will discuss conclusions regarding the models and their significant predictors, along with steps that could have been taken to conceivably improve the performance of our models in terms of the designated criteria. An appendix will be attached to the end of the report and it will include all relevant code and output that cannot be included in the report.

2. DATA PREPARATION

2.1 Data Cleansing

Before any model fitting could be conducted, the data provided in the dataset needed to be cleansed substantially. Many relationships among variables that we would expect to hold did not hold, indicating that many records in the dataset had not been properly updated since being added to the file. In order to correct for such inconsistencies, the team made an assumption: if a relationship between variables was violated in a given record, the violation occurred because one or more of the relevant variables was not updated at all in an instance in which it should have been updated. Therefore, any time we engaged in data manipulation to alleviate an inconsistency, we always added to a variable's value rather than subtracting from the value. Please refer to Appendix A for detailed discussion regarding data cleansing.

2.2 Creating New Variables

Several new variables were created that the team suspected might be significant predictors in the logistic regression model, the multiple regression model, or both. Some of these variables would later be used in interaction terms for one or both of the regression models as well. Firstly, yearly order binary variables were created for each of the past four years (i.e. **ordtyr_bin**, **ordlyr_bin**, **ord2ago_bin**, **ord3ago_bin**). For instance, in cases in which **ordtyr** was greater than zero, **ordtyr_bin** was set to one; otherwise, **ordtyr_bin** was set to zero. In cases in which **ordlyr** was greater than zero, **ordlyr_bin** was set to one, and so on. A binary variable was also created for life-to-date fall orders, given that we are ultimately trying to predict purchases during a fall order period. In cases in which **falord** was greater than zero, **falord_bin** was set to one; otherwise, **falord_bin** was set to zero.

New variables were also created to reflect number of years since a customer's latest purchase (**yrs_since_lp**) and number of years since a customer was added to the file (**yrs_since_add**). In reality, these variables are simply transformations of the two newly created variables described in the previous section: **lpuryear_new** and **dateadyr**. Quite simply, **yrs_since_lp** was calculated by subtracting **lpuryear_new** from 2012, while **yrs_since_add** was calculated by subtracting **dateadyr** from 2012. For instance, when **lpuryear_new** was 2012, **yrs_since_lp** was set to zero, and so on.

Lastly, a binary **targdol** variable (**targdol_bin**) was created to be used as the dependent variable in the logistic regression model. In cases in which **targdol** exceeded zero, **targdol_bin** was set to one; otherwise, **targdol_bin** was set to zero. Furthermore, a new variable called **targdol_pur** was created to be used as the dependent variable in the multiple regression model. In cases in which **targdol** equaled zero, **targdol_pur** was set to NA; otherwise **targdol_pur** was set equal to **targdol**.

3. EXPLORATORY ANALYSIS

3.1 Descriptive Statistics

Before performing a rigorous analysis, an exploratory analysis was conducted by computing descriptive statistics to provide a quantitative and graphical summary of the data, to detect patterns in the data, assess the nature of the relations, and to help avoid incorrect assumptions for the statistical analysis. For the categorical variables, the descriptive statistics included frequencies and proportions for the various categories of the variables. For continuous variables, the descriptive analysis consisted of computing central tendency and distribution measures (e.g. skewness). All outputs from this section can be found in Appendix A.

The descriptive statistics for the continuous variables show large positive skewness values (right skewed distribution), suggesting that a transformation is needed to symmetrize and reduce the range as well. For the date variables, the minimum values for **datead6** and **datelp6** were 01/06/1931 and 01/01/1980 respectively (17 data points with **datelp6** = 01/01/1980 and 4 data points with **datead6** = 01/06/1931 were removed). For the categorical variables, frequency tables were constructed to assess the distribution. The results show that the train set and test set were evenly split 50-50 (Total Number of records= 101,532). However, the dataset is highly imbalanced, where only 9.427% of the customers responded by making a purchase. Thus, transformations might be needed and classification rates of the model should be interpreted with caution as they can be misleading. The variable **lpuryear** for the most part showed increasing frequency as the year increased from 2003 to 2012. There were also 0.72% records (N=728) with **lpuryear** missing. This issue was addressed in the data cleansing section.

In multiple regression, the normality of residuals implies that the response should be continuous and reasonably normally distributed. There are no assumptions in linear regression where the predictor variables have to be normally distributed, continuous, or even symmetric. However, transformations were considered to symmetrize the distribution that will help with the fit, reduce influence of outliers and overall improve the predictive capability of the model. In order to determine if transformations are needed and if outliers should be removed (e.g. 3 data points with **ordhist1** > 100 were removed), the distribution of continuous variables were assessed by plotting histograms and boxplots. Figure 1 shows the distribution of **targdol**, which is clearly right skewed. The log10 transformation shows that it worked pretty well in normalizing the variable. Potential outliers can also be seen from the boxplot. Similar analysis was done for the predictor variables in which log10, square root and inverse transformations were considered during the model selection process. The distribution of the dates show increasing number of records with later **datelp6**, and for the most part two distinct dates in the calendar year for **datead6**. The full details are shown in Appendix A.

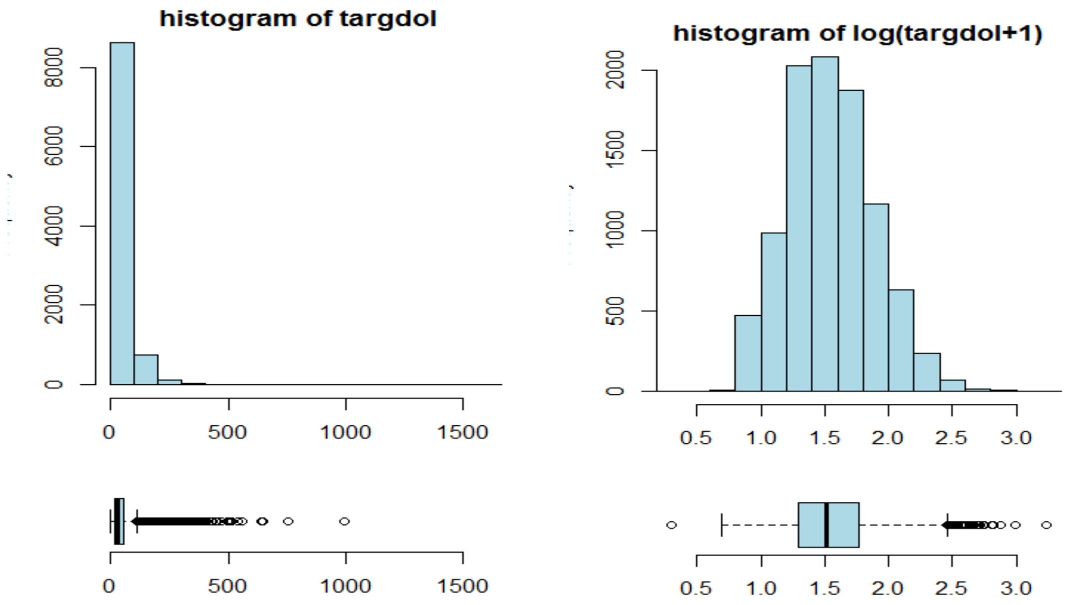


Figure 1. Distribution of targdol

Correlation and scatter plots were also constructed to assess the relationship between variables. Although correlations measure only linear relationship, they also provide some insight on the structure of the data. From the correlation plot (Figure 2), it can be seen that as expected, **yrs_since_lp** has a negative relationship with the rest of the variables. All the sales seemed positively correlated with **saleshist** and the orders, and all the orders have some positive relationship with **ordhist1**. The variables **falord** and **sprord** have a moderate to strong positive correlation with **ordhist1**, suggesting that all three cannot be in the model due to multicollinearity. All these relationships are expected since some variables are derived from each other or are multiple of each other (e.g. **falord** + **sprord** = **ordhist1**)

The box plot shows that the distribution of **yrs_since_lp** is different for people that responded and the people that did not. This is an indication that **yrs_since_lp** might be an important predictor in the model. Similar analysis suggests that the order variables might be more important predictors than sales variables for the classification model (see Appendix A).

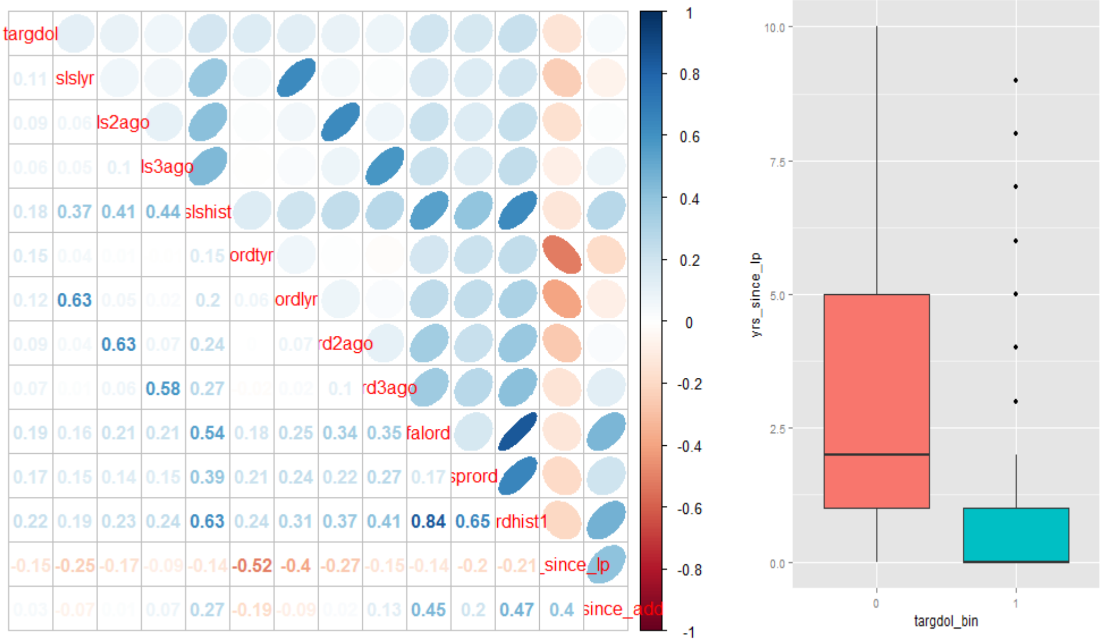


Figure 2. Correlation plot and Box plot

4. CLASSIFICATION MODEL

4.1 Model Selection

The first step was to fit a big model including all the transformed variables, derived variables and interactions that might be relevant in predicting the response. The overall approach was then to use a sequential method by looking at insignificant terms, VIF, deviance tests and model selection criteria, such as AIC and BIC, to determine what variables to add and drop. A stepwise regression was also conducted to arrive at the candidate models.

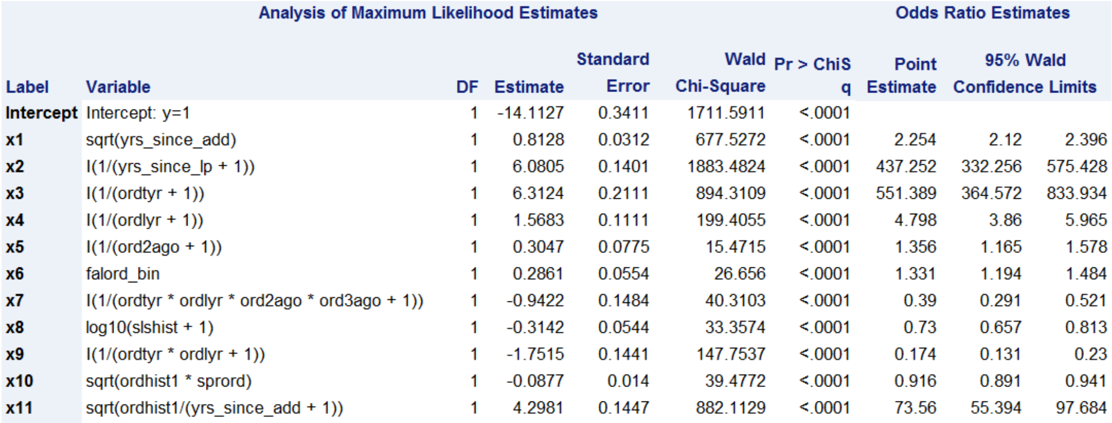
For example, the first model that was fitted had large VIF values and many insignificant terms. A stepwise BIC was selected since it penalizes model complexity more heavily, but VIF issues were still present. At this point, looking at correlation plots and using the meaning of the variables, some variables were dropped from the model. For example, in an intermediate step, the sales variables were dropped while order variables were kept in the model. This process was repeated until satisfactory results were achieved, that is all variables were significant and VIF problems were not present. The VIF of the final model (**log5**) indicates that multicollinearity is not an issue, with the exception a borderline VIF value for one variable (VIF = 10.75). An alternative model (l**og9**) was explored where VIF was not a problem. Detailed steps of the model selection process and outputs for **log9** can be found in Appendix 0.

Model **log5** was finally chosen as the best model over **log9** because it performed better in model selection criteria and VIF was not regarded as a significant problem

4.2 Final Model

The summary of model **log5** is shown below in Figure 3 All coefficients are significant given the rest are in the model. The coefficients are hard to interpret due to the transformations. A more intuitive interpretation is that the exponential of the coefficient of the transformed predictor variables is the multiplicative factor by which the odds of the outcome (i.e. customer will be a buyer) is expected to change given a one-unit increase in the transformed predictor variable.

As expected from the exploratory analysis, there are more binary variables and order variables than sales variables. The interaction that represents consistency of last purchases (products of the orders) appears to be a significant predictor. Also there seems to be a negative interaction between **ordhist1** and **sprord**, indicating that if a customer has lot of orders in the past that are spring orders, then the customer is less likely to purchase in the fall, after controlling for the other factors. By far the most significant predictors with the greatest effect (see odds ratio) are the inverses of the **ordtyr** and **yrs_since_lp**, which capture the recency effect and were pointed out in the exploratory analysis. All signs of the coefficients are as expected, with the exception of a few that cannot be interpreted as main effect because of their presence in an interaction term. The indicator of fall orders was also found to be significant, which makes sense since we are predicting buyers in the fall.



Analysis of Maximum Likelihood Estimates							Odds Ratio Estimates		
Label	Variable	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Point Estimate	95% Wald Confidence Limits	
Intercept	Intercept: y=1	1	-14.1127	0.3411	1711.5911	<.0001			
x1	sqrt(yrs_since_add)	1	0.8128	0.0312	677.5272	<.0001	2.254	2.12	2.396
x2	I(1/(yrs_since_lp + 1))	1	6.0805	0.1401	1883.4824	<.0001	437.252	332.256	575.428
x3	I(1/(ordtyr + 1))	1	6.3124	0.2111	894.3109	<.0001	551.389	364.572	833.934
x4	I(1/(ordlyr + 1))	1	1.5683	0.1111	199.4055	<.0001	4.798	3.86	5.965
x5	I(1/(ord2ago + 1))	1	0.3047	0.0775	15.4715	<.0001	1.356	1.165	1.578
x6	falord_bin	1	0.2861	0.0554	26.656	<.0001	1.331	1.194	1.484
x7	I(1/(ordtyr * ordlyr * ord2ago * ord3ago + 1))	1	-0.9422	0.1484	40.3103	<.0001	0.39	0.291	0.521
x8	log10(slshist + 1)	1	-0.3142	0.0544	33.3574	<.0001	0.73	0.657	0.813
x9	I(1/(ordtyr * ordlyr + 1))	1	-1.7515	0.1441	147.7537	<.0001	0.174	0.131	0.23
x10	sqrt(ordhist1 * sprord)	1	-0.0877	0.014	39.4772	<.0001	0.916	0.891	0.941
x11	sqrt(ordhist1/(yrs_since_add + 1))	1	4.2981	0.1447	882.1129	<.0001	73.56	55.394	97.684

Figure 3. Output log5 model (Estimates)

4.3 Model Diagnostics

Various tests of overall goodness of fit show that both models indicate rejection of the null hypothesis that all the coefficients are equal to zero (Appendix 0). The ROC curve and AUC (or concordance index) are shown in Appendix 0 and test the discriminatory power of the model. In this case, both models (fit1: log5, fit2: log9) have an AUC > 80% meaning the models can be regarded as very good in terms of discriminatory power. The p-value suggests that the AUC's are statistically significantly different, however, in practical terms there is not much difference. Appendix B also shows the classification tables on the training set for different cutoffs. Both models perform well in terms of overall classification rate (about 92%), but poorly on correct classification of true buyers (about 23%), which translates into a high Type II error.

Logistic regression models do not require any assumptions of normality of the residuals, but residuals and influential points should still be checked to assess the goodness of the model. Diagnostics were conducted to assess the validity of the model. The assessment included influence diagnostics to identify highly influential points by considering differences between fitted and observed values. Predicted probability diagnostics and leverage diagnostics were also conducted. The diagnostic plots are shown below in Appendix 0. The diagnostics might indicate some high leverage points. However, due to the large sample size, they might not be influential. The predicted probability plots did not show any suspicious point not following the trends. The displacement measures and deviance deletion differences, which capture the effect of influential points in the coefficients, were not found to be a significant problem. Therefore, based on the diagnostics and goodness of fit, it was determined that the model was adequate. Similar conclusions were reached for **log9** model (see Appendix B).

5. MULTIPLE REGRESSION MODEL

5.1 Model Selection

The multiple regression model selection process was similar as the classification model selection process, but other relevant criteria and tests, such as Mallow's Cp, adjusted R^2 and F-tests, were considered. In addition to stepwise regression, a best subsets method was used to identify the best models for different sizes. Appendix C has the full details including the subset matrix of the variables and model sizes with the different criteria. The results are summarized graphically in Figure 4. As expected, BIC criteria suggests a model with fewer predictors. It can also be seen that for model sizes beyond 11 parameters the Mallow's Cp, adjusted R^2 and AIC remain almost the same. However, moving away from a model size between 9 and 12 worsens BIC significantly. Thus, BIC was chosen as one of the primary criteria since it gives a more parsimonious model with about the same value of the other criteria than the larger models.

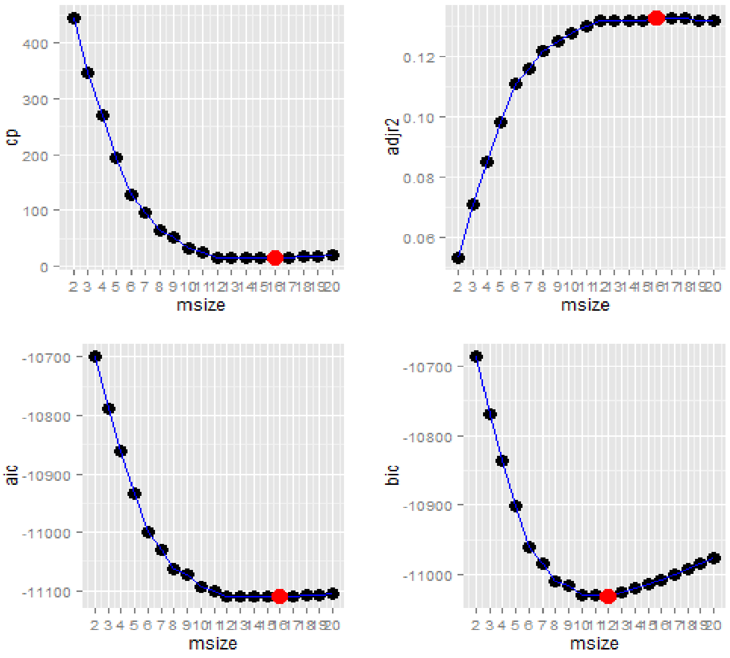


Figure 4. Multiple Regression Model Selection

5.2 Final Model

Detailed steps of the model selection process can be found in Appendix C. The final model (**lm4**) is shown below in Table 1, while the alternative model (**lm6**) can be found in Appendix C. The chosen model has higher adjusted R^2 (0.136 vs. 0.0996) but VIF might be problematic, which is discussed in the next section. The overall p-value <0.05 for the overall F-test indicates that not all the coefficients of the predictors are equal to zero. The summary shows that all coefficients are significant given the rest are in the model. The coefficients are hard to interpret due to the transformations. The value of the coefficient can be interpreted as the expected to change in the transformed response variable as a results of one unit change in the transformed predictor.

As expected, the multiple regression model contains more sales variable than the classification model. The interaction that represents consistency of last purchases (**slstyr*slslyr**) appears to be a significant predictor. The effect of **slshist** on the response variable also seems to depend on the level of **yrs_since_add**, which is capture by the interaction. The signs of the coefficients match our intuition for the most part, since higher sales in the past or consistency is expected to have higher purchase amount. The binary order variables have opposite signs of what we expected, suggesting multicollinearity problems or a missing variable.

Table 1. lm4 model

Variable	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	1.378	0.029	46.883	0
ordtyr_bin	-0.316	0.031	-10.058	1.44E-23
ordlyr_bin	-0.241	0.033	-7.318	2.94E-13
ord2ago_bin	-0.328	0.035	-9.330	1.57E-20
ord3ago_bin	-0.194	0.036	-5.461	4.96E-08
log10(slshist + 1)	0.072	0.015	4.749	2.10E-06
log10(slstyr + 1)	0.176	0.021	8.406	5.52E-17
log10(slslyr + 1)	0.122	0.021	5.782	7.84E-09
log10(sls2ago + 1)	0.190	0.022	8.740	3.18E-18
log10(sls3ago + 1)	0.115	0.022	5.196	2.12E-07
log10(slstyr * slslyr + 1)	0.020	0.006	3.340	0.00084306
I(log10(slshist + 1)/sqrt(yrs_since_add + 1))	0.074	0.020	3.780	0.00015893

5.3 Model Diagnostics

For the purposes of model validation, diagnostics were applied to each candidate model (**lm4** and **lm6**) to test the assumptions of predictor linearity, error normality, homoscedasticity, and predictor linear independence. Notably, because we were not dealing with time-series data, tests of error independence were not necessary. The discussion that follows is for **lm4**. The analysis was repeated for **lm6** and is shown in Appendix 0.

Figure 5 show various diagnostic plots. In order to check the assumption that each predictor variable was linear with respect to the response variable, each predictor variable was plotted against the residuals (see Appendix 0 for additional plots). Plots for each variable seem random, so assumptions about the form of the model (linear in the regression parameters) is satisfied, indicating that a transformation or a higher degree term is therefore not necessary.

The Normal Q-Q plot shows that standardized residuals mostly fall on the line, indicating that the normality assumption of errors is satisfied. The plot of residuals against fitted values shows that the data points are random and form a parallel band, suggesting that the common error variance (homoscedasticity) assumption is valid without further modification of the model. The same conclusions are reached for **lm6**.

As indicated above, the primary candidate suffers from some degree of multicollinearity which must be further examined. As expected, the correlation plots show strong correlation among some predictors. In addition, the variance inflation factor is greater than 10 for some of the predictor variables, so the assumption of the linear independence of each predictor may be violated.

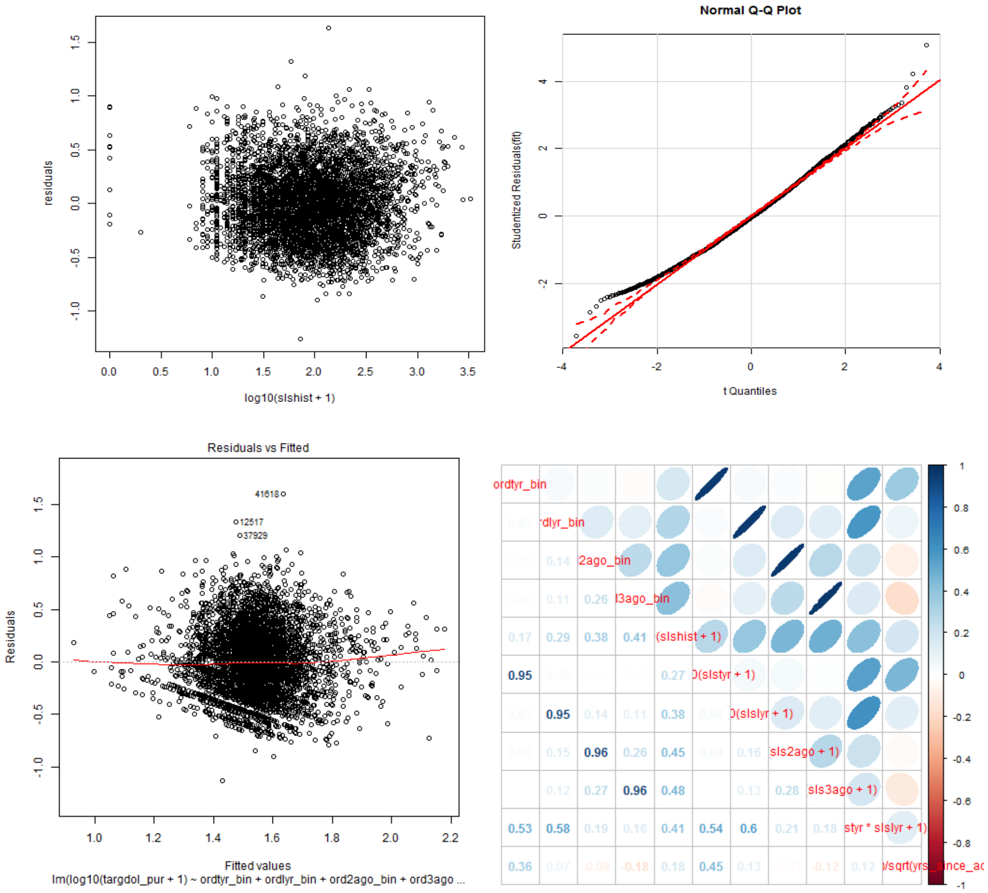


Figure 5. Residual vs Predictor, Q-Q Plot, Residual vs Fitted, Correlation Plot

However, the VIFs are not extremely large (capping at 14.5), so there is not a severe multicollinearity problem. In this case, the high VIFs are not a problem and can safely be ignored for the following reasons:

1. The high VIFs are caused by the inclusion of interactions – including the variables as well as their products implies that a given predictor will be correlated to its product interaction term.
2. Many of the variables with VIF > 10 are dummy variables (e.g., **ordtyr_bin**), which are highly correlated with the corresponding sales in that year (e.g., **slstyr**)

Because, for the purposes of this problem, we are not concerned with determining the precise effect of each variable, but rather with obtaining a model with the best projective power, we are comfortable dismissing multicollinearity concerns for the sake of a model that approaches the highest observed R-squared value. For comparison, the scatter plots and correlation coefficients for the alternative model **lm6** do not show strong correlation among any of the predictors, and no multicollinearity is indicated by high VIF values. Because of its predictive power and our willingness to set aside its multicollinearity concerns, we settled on **lm4** as the final multiple regression model.

All Cook distances are less than 1, so there are no influential points when using the cutoff of D greater than 1 (see Appendix C). There might be a couple of potential outliers since the studentized residuals are greater than |2|. However, conducting a statistical test only shows that observation 41618 is an outlier. Using the rule of thumb (hii>2(p+1)/2), there are a few observations with high leverage (i.e. extreme value on a predictor variable). However, the number of high leverage points is small compared to the size of the training set, and because the binary predictor variables are unbalanced and can potentially create high leverage points. The identified high leverage and outliers points do not seem to be influential (i.e. substantially change the coefficient estimates) according to Cook's D, which combines the information of leverage and outlierness of the residual. Therefore, no points were removed from the dataset.

6. MODEL VALIDATION

6.1 Classification

The finalist logistic classification model was run on the test set of customer records and its predictions for whether or not a given customer would buy compared with actual **targdol** values in order to obtain a rate of misclassification. The following tables break down the logistic model’s correct predictions that a customer would buy or not buy, along with its incorrect predictions of the same:

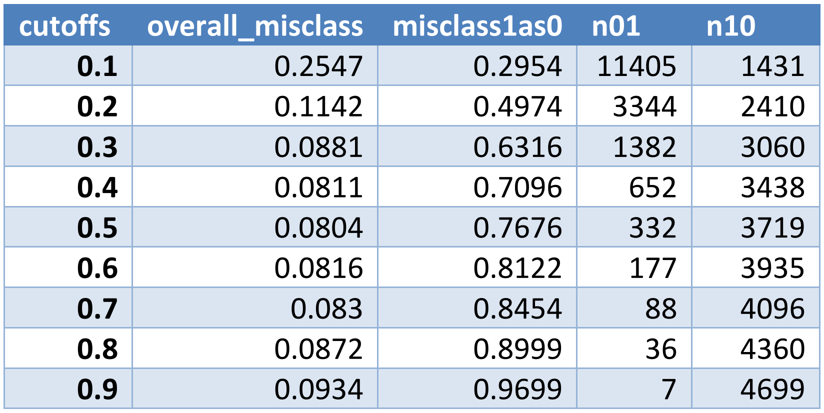
		Predicted	
		0	1
Response	0	46059	318
	1	3549	1177

		Predicted	
		0	1
Response	0	99.31%	0.69%
	1	75.10%	24.90%

Overall, these predictions indicate a misclassification rate of 7.57% of all cases. However, as indicated previously, this number is misleading due the imbalance in the data (i.e. high proportion of 0’s compared to 1’s). The cross tabulation on the right shows the row percent, which indicated the percentage conditioned on the observed true value. Thus, a customer that did not buy from the catalog (response = 0) was misclassified as buying (predicted = 1) 0.69% of the times. This number represents the false positives (or type I error). On the other hand, of all the customer that actually bought from the catalog (response = 1), the model misclassified (predicted

= 0) 24.9% of the cases. This number represents the false negatives (or type II error). The specificity of the model (proportion of 0’s correctly classified) is very high (99.31% = 1 – type I error). On the other hand, the sensitivity of the model (proportion of 1’s correctly classified) is not very high (75.1% = 1- type II error).

For our purposes, the type II error corresponds to not sending a given customer a catalog when they would make a purchase if sent one. In this initial calculation, misclassification rate was calculated with a cut-off point of 0.5, so that any logistic model prediction above 0.5 was indicative of a predicted buyer, and any logistic model prediction below 0.5 was indicative of a predicted non-buyer. If we were to account for an increased cost for Type-II errors by reducing this cut-off point (and therefore classifying more customers as buyers), our total cost of misclassification would decrease. This idea is illustrated when we allow the cut-off point to range over a set of values, as shown below in a table detailing misclassification rates for different cut-off points from our training set:



cutoffs	overall_misclass	misclass1as0	n01	n10
0.1	0.2547	0.2954	11405	1431
0.2	0.1142	0.4974	3344	2410
0.3	0.0881	0.6316	1382	3060
0.4	0.0811	0.7096	652	3438
0.5	0.0804	0.7676	332	3719
0.6	0.0816	0.8122	177	3935
0.7	0.083	0.8454	88	4096
0.8	0.0872	0.8999	36	4360
0.9	0.0934	0.9699	7	4699

Of course, determining the true cut-off point would require precise knowledge of the cost of each type of error. In this problem, it is clear that the costs of misclassifications are not equal, because it is more costly to miss a potential buyer (type II error) than to send a catalog to someone who is not a buyer (type I error). In practice, knowing the relative costs one can select the cutoff that minimizes the total cost of misclassification. As an example, Appendix D shows the results if the cost of type II error was twice as the cost of type I error, indicating that a lower cutoff of about 0.3 should be chosen for classification purposes.

6.2 Financial Criterion

Our approach was to find the expected **targdol** value of each customer in the test set by multiplying the predicted probabilities from the logistic regression by the predicted **targdol** amount from the multiple logistic regression. Then, for the top 5,000 customers with highest expected **targdol**, the actual **targdol** values were totaled up, which results in $112,905.9. This will be the actual total sales that will result if the models were applied to the test set and catalogs were mailed to the top 5,000 prospects, which we are defining by targeting the top 5,000 prospect with the highest expected value. In other words, had we used the model to target these top 5000 prospects, then that would have resulted in $112,905.9 actual sales revenue. Adding up the total dollar amount of **targdol** in the test set, we obtain a possible $226,456.3 that can be netted by sending catalogs to the right buyers. Thus, the payoff of our model represents about 50% of the maximum possible amount (as there are fewer than 5,000 buyers in the test set).

Note that we used a different approach to target the 5,000 customers by basing it on expected value instead of finding "buyers" predicted by the classification model. If we instead use this approach, then we would need to use a cutoff to determine these buyers. Thus, suppose we use a cutoff of 0.2 (because this will ensure more than 5,000 buyers). If we select the top 5,000 **targdol** amounts as predicted from by the finalist multiple regression model **lm4** out of the total number of customers from the test set predicted to be buyers by the finalist logistic regression model **fit.log5**, the total sum of the actual **targdol** values is $108,697 or 48% of the maximum possible amount (as there are fewer than 5,000 buyers in the test set). Note that this targeting method gives a lower payoff of the model than that of the expected value method.

6.3 Statistical Criterion

The final fitted regression model (**lm4**) met the usual criteria such as it contained all significant coefficients, the residual plots were satisfactory and different model selection criteria were considered. Parsimony and easiness of interpretability were also taken into account. In addition, the mean square error of prediction (MSEP) was computed to evaluate the fitted models (derived from the training set) on the test set.

To validate the models derived from the training set against the test set, first the classification model was applied to the test set to identify the buyers. Using a cutoff of 0.2, this gives 5,698 buyers. Then their purchase amounts were predicted by applying the regression model, resulting in a MSEP = 2269.732. Similarly, using a cutoff of 0.1 gives a MSEP = 1158.787.

7. CONCLUSIONS

As we expected prior to fitting any logistic regression models, recency of last purchase and consistency of past purchases proved to be very important predictors of **targdol_bin**. In our final logistic regression model, 1/(**yrs_since_lp** + 1) appears to be the most significant predictor, with a z-value equal to 43.399 (p < 2e-16). Two interaction terms measuring consistency of past purchases are also highly significant in the final logistic regression model: 1/(**ordtyr*****ordlyr** + 1), with a z-value equal to -12.155 (p < 2e-16), and 1/(**ordtyr*****ordlyr*****ord2ago*****ord3ago** + 1), with a z-value equal to -6.349 (p = 2.16e-10). One more interaction term turned out to be highly significant in the final logistic model: sqrt(**ordhist**/(**yrs_since_add** + 1)), with a z-value equal to 29.701 (p < 2e-16). The significance of the variable indicates that *volume of past purchases* is also a very important predictor of **targdol_bin**; that is, the higher a customer's average annual number of orders is since being added to the file, the more likely it is that the customer will order again in the upcoming period.

As we expected prior to fitting any multiple regression models, the significant predictors of **targdol_pur** (i.e. those in the multiple regression model) turned out to be quite different than the significant predictors of **targdol_bin** (i.e. those in the logistic regression model). While none of the yearly sales variables (i.e. **slstyr**, **slslyr**, **sls2ago**, **sls3ago**) were included in the final logistic model, all four were included in the final multiple regression model. This result makes a lot of sense intuitively, of course, because while the logistic model is intended to predict whether or not a given customer will place an *order* in the upcoming period, the multiple regression model is intended to predict the amount of *sales* a customer will contribute in the upcoming period, given that he or she will place an order in the first place. Interestingly, only two interaction terms were ultimately included in the final multiple regression model: log10(**slstyr*slslyr** + 1), with a t-value of 3.340 (p = 0.000843), and log10(**slshist** + 1)/sqrt(**yrs_since_add** + 1), with a t-value of 3.870 (p = 0.000159). Moreover, the team was surprised to find that no term involving **yrs_since_lp**, **targdol_bin**'s most significant predictor, was included in the multiple regression model at all.

Despite our efforts to fit the most appropriate models for predicting **targdol_bin** and **targdol_pur**, respectively, the team understands that our performance on the statistical and financial criteria may well have been limited by the manner in which we cleansed the data. Specifically, the team ultimately discovered the **lpuryear** variable from the original dataset may be unreliable at best, and systematically flawed at worst. The team discovered that in more than 2,000 instances, **lpuryear** returned a "3" when **datelp6** returned a date with the year 2012; thus, it seems highly likely that in such cases, the **lpuryear** value is referring to the year 2013 rather than 2003, as we initially assumed. Of course, if the variable is truly indicating that a customer's year of last purchase was 2013, such records should be thrown out of dataset entirely; otherwise, we are using data to fit our models that we would not have had if we were fitting the models in August of 2012 (prior to the start of the Fall 2012 sales period). In the end, the team decided to set **yrs_since_lp** equal to zero in such cases, hoping that the **lpuryear** values were simply imprecise.

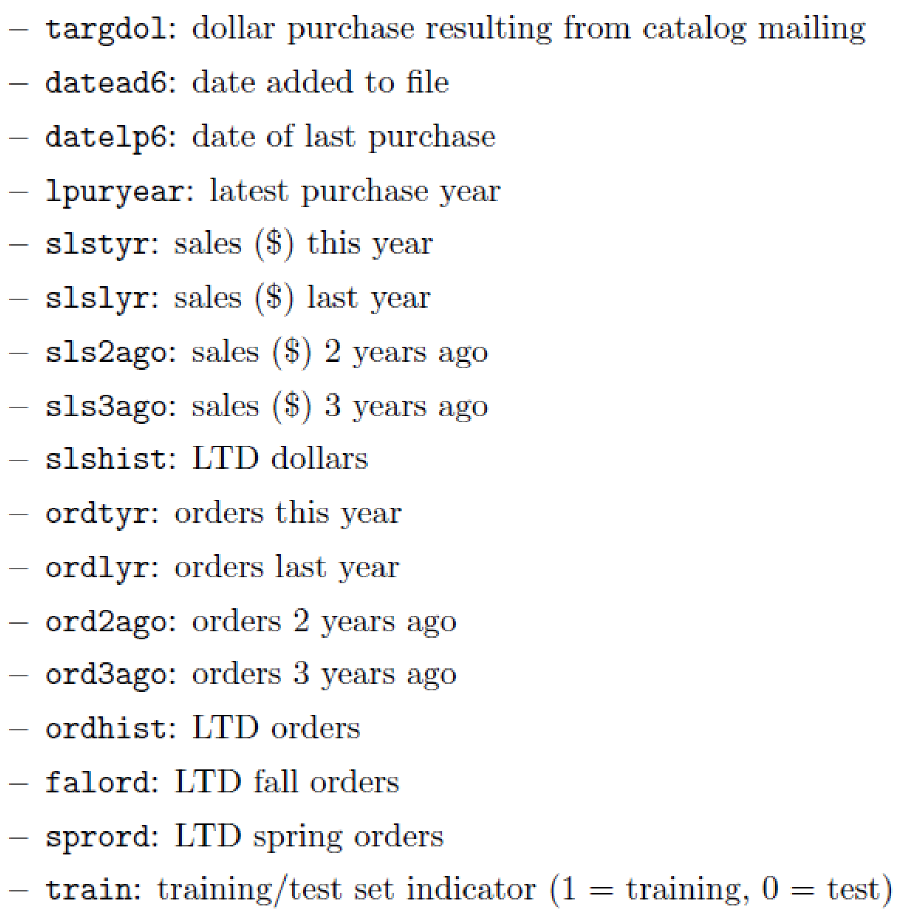
REFERENCES

Samprit Chatterjee, Ali S. Hadi. “Regression Analysis by Example”, 5th Edition

APPENDIX

A. Exploratory Analysis

Original Variables



- `targdol`: dollar purchase resulting from catalog mailing
- `datead6`: date added to file
- `datelp6`: date of last purchase
- `lpuryear`: latest purchase year
- `slstyr`: sales ($) this year
- `slslyr`: sales ($) last year
- `sls2ago`: sales ($) 2 years ago
- `sls3ago`: sales ($) 3 years ago
- `slshist`: LTD dollars
- `ordtyr`: orders this year
- `ordlyr`: orders last year
- `ord2ago`: orders 2 years ago
- `ord3ago`: orders 3 years ago
- `ordhist`: LTD orders
- `falord`: LTD fall orders
- `sprord`: LTD spring orders
- `train`: training/test set indicator (1 = training, 0 = test)

Data Cleansing

<u>Updating Life-To-Date Orders</u>

As stated previously, life-to-date fall orders (**falord**) plus life-to-date spring orders (**sprord**) should equal total life-to-date orders (**ordhist**) in every record. Therefore, in cases in which **falord** plus **sprord** exceeded **ordhist**, **ordhist** was updated to equal **falord** plus **sprord**. By contrast, however, in cases in which **ordhist** exceeded **falord** plus **sprord**, neither **falord** nor

sprord was updated to alleviate the inconsistency. In the latter set of cases, given that a true **ordhist** value is greater than or equal to the stated **ordhist** value, we can state with certainty that either the true **falord** value is greater than the stated **falord** value or the true **sprord** value is greater than the stated **sprord** value. Unfortunately, while we know that at least one of these values should be increased in the record of interest, we cannot state with certainty that any one in particular should be updated. Therefore, in cases in which **ordhist** exceeded **falord** plus **sprord**, the team was unable to resolve the violation. (Note: the updated **ordhist** variable was renamed as "**ordhist1**".)

Updating Yearly Order Variables

If a sale was made to a particular customer this year, then sales this year (**slstyr**) in the customer's record should be greater than zero and by definition, orders this year (**ordtyr**) should be greater than zero as well. Similarly, if sales last year (**slslyr**) is greater than zero then orders last year (**ordlyr**) should be greater than zero, and so on. Therefore, for each year, if the sales by year value exceeded zero but the order by year value equaled zero, the order by year value was updated to equal one. For instance, if **slstyr** was greater than zero but **ordtyr** equaled zero, **ordtyr** was updated to equal one, and so on.

Updating Latest Purchase Year

Two variables in the dataset indicate the latest purchase year for each customer: date of last purchase year (**datelp6**), and latest purchase year (**lpuryear**). The **lpuryear** variable has a value between zero and nine, rather than a full year; therefore, the team inferred that **lpuryear** = 2 refers to 2012, **lpuryear** = 0 refers to 2010, **lpuryear** = 8 refers to 2008, and so on (it appears that if **datelp6** is accurate and

datelp6 < 1/1/2003, **lpuryear** is blank in that row). After updating **lpuryear** to reflect an actual year, these two variables should return the same year. Therefore, to correct for inconsistencies between these two variables, a new latest purchase year variable (**lpuryear_new**) was created. In cases in which **lpuryear** exceeded **datelp6**, **lpuryear_new** was set equal to **lpuryear**; otherwise, **lpuryear_new** was set equal to **datelp6**.

Once differences between **datelp6** and **lpuryear** had been reconciled by creating the **lpuryear_new** variable, **lpuryear_new** was updated further by comparing it against the four (previously updated) yearly order variables. For example, in cases in which orders this year (**ordtyr**) was greater than zero, we assume that the customer made a purchase in 2012; therefore, if **lpuryear_new** was less than 2012 in such cases, **lpuryear_new** was updated to equal 2012. Similarly, in cases in which **ordtyr** = 0 but **ordlyr** was greater than zero, we assume that the customer last made a purchase no earlier than 2011; therefore, if **lpuryear_new** was less than 2011 in such cases, **lpuryear_new** was updated to equal 2011.

(Note: Due to concerns about the precision of dates recorded in the dataset, we chose to use years, rather than specific dates, as values of our updated date variables.)

Updating Year Added to File

Year added to file (**dateadyr**) was created by extracting the year from each date added to file (**datead6**) value. Then, **dateadyr** was updated by comparing it against the order by year variables, similarly to how **lpuryear_new** was updated. For instance, in cases in which **ord3ago** was greater than zero, we assume that the customer was added to the file no later than 2009; therefore, if **dateadyr** was greater than 2009 in such cases, **dateadyr** was updated to equal 2009. Similarly, in cases in which **ord3ago** = 0 but **ord2ago** was greater than zero, we assume that the customer was added to the file no later than 2010; therefore, if **dateadyr** was greater than 2010 in such cases, **dateadyr** was updated to equal 2010.

Descriptive Statistics

	n	n_na	mean	sd	median	min	max	range	skew	kurtosis	se
targdol	101532	0	4.5	21.45	0	0	1720	1720	14.1	569	0.07
slstyr	101532	0	11.22	35.42	0	0	1748	1748	9.71	224.3	0.11
slslyr	101532	0	12.52	38.29	0	0	2290	2290	11.63	382.32	0.12
sls2ago	101532	0	12.6	38.5	0	0	2942	2942	14.01	583.49	0.12
sls3ago	101532	0	11.67	40.75	0	0	2844	2844	20.11	938.26	0.13
slshist	101532	0	89.37	133.6	50	0	5091	5091	8.25	148.77	0.42
ordtyr	101532	0	0.26	0.53	0	0	8	8	2.4	8.76	0
ordlyr	101532	0	0.29	0.57	0	0	11	11	2.38	9.31	0
ord2ago	101532	0	0.29	0.56	0	0	9	9	2.3	8.07	0
ord3ago	101532	0	0.27	0.55	0	0	10	10	2.39	8.57	0
ordhist	101532	0	2.14	1.98	1	0	39	39	3.38	18.23	0.01
falord	101532	0	1.38	1.56	1	0	106	106	10.25	514.87	0
sprord	101532	0	0.72	1.03	0	0	21	21	2.88	17.75	0

datead6	datelp6
Min. :1931-06-01	Min. :1980-01-01
1st Qu.:2005-09-27	1st Qu.:2007-11-17
Median :2008-01-11	Median :2009-11-15
Mean :2007-02-23	Mean :2009-05-22
3rd Qu.:2010-01-18	3rd Qu.:2011-03-13
Max. :2012-11-30	Max. :2012-07-20

Train	Frequency	Percent
0	51114	50.34275
1	50418	49.65725
Total	101532	100

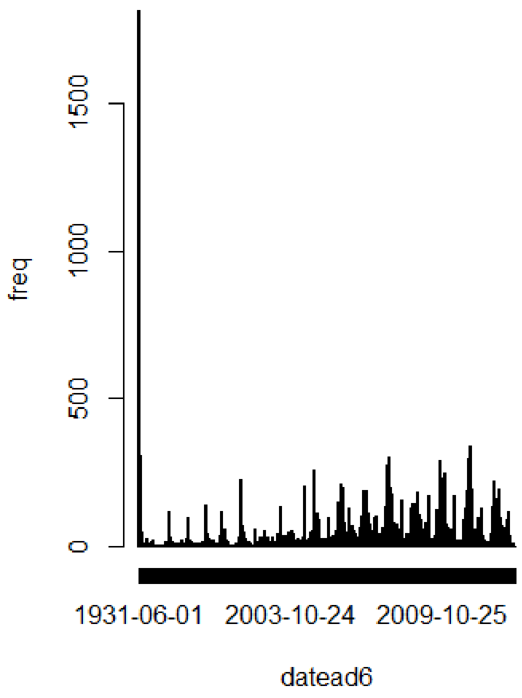
Targdol_bin	Frequency	Percent
0	91961	90.57342
1	9571	9.426585
Total	101532	100

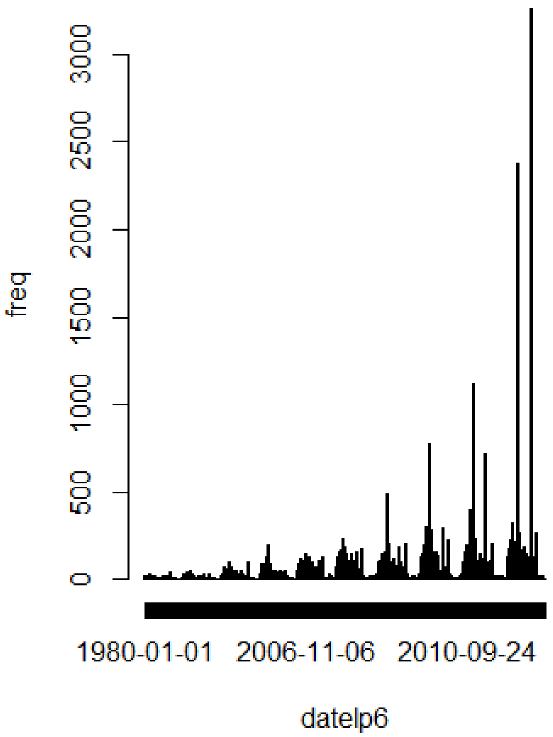
lpuryear	Frequency	Percent	Valid Percent
0	14931	14.70571	14.81191223
1	19733	19.43525	19.57561208
2	8812	8.679037	8.741716599

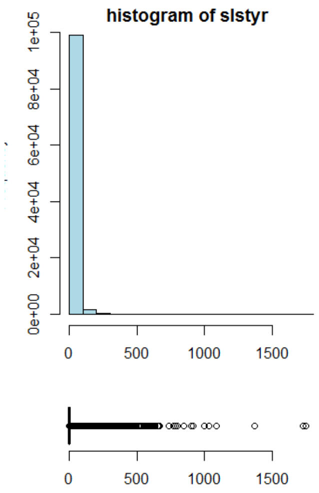
3	4347	4.281409	4.312328876
4	3152	3.10444	3.126860045
5	5519	5.435725	5.474981152
6	6833	6.729898	6.778500853
7	9545	9.400977	9.468870283
8	12327	12.141	12.2286814
9	15605	15.36954	15.48053649
NA's	728	0.717015	NA
Total	101532	100	100

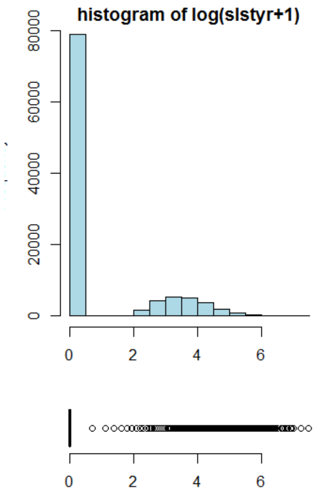
Distributions

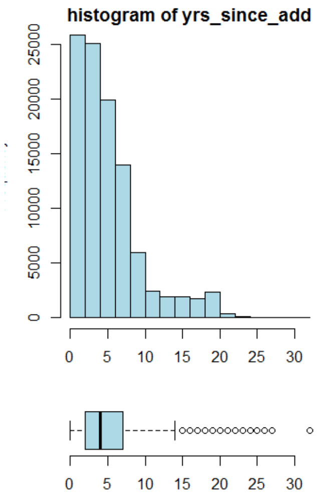
After data cleansing and with new variables created.

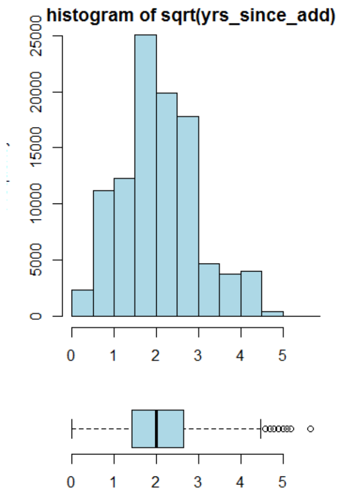


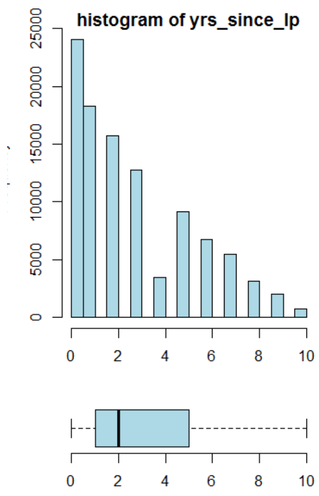


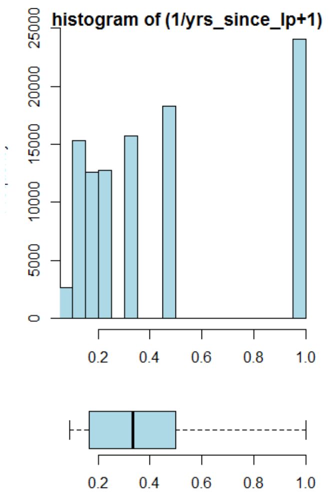

histogram of slstyr
0e+00
2e+04
4e+04
6e+04
8e+04
1e+05
0
500
1000
1500
0
500
1000
1500

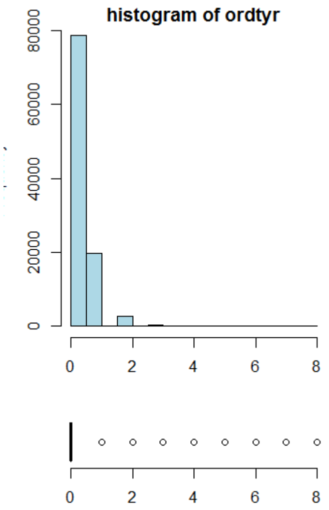

histogram of log(slstyr+1)
0
20000
40000
60000
80000
0
2
4
6
0
2
4
6

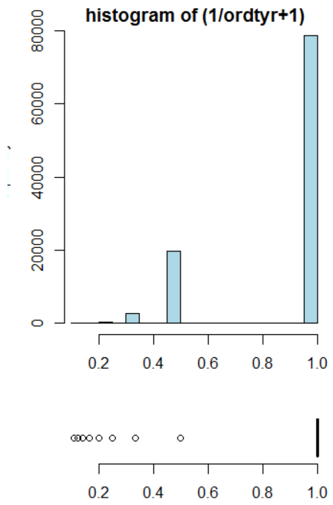

histogram of yrs_since_add
0
5000
10000
15000
20000
25000
0
5
10
15
20
25
30
0
5
10
15
20
25
30

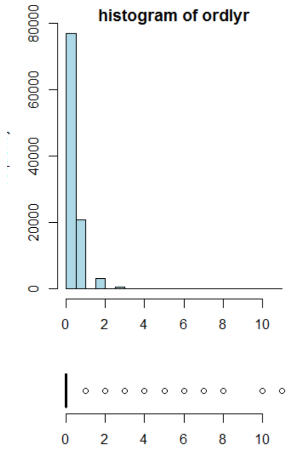

histogram of sqrt(yrs_since_add)
0
5000
10000
15000
20000
25000
0
1
2
3
4
5
0
1
2
3
4
5

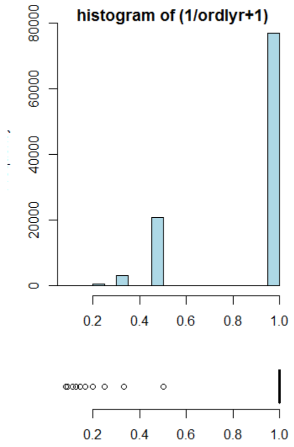

histogram of yrs_since_lp
0
5000
10000
15000
20000
25000
0
2
4
6
8
10
0
2
4
6
8
10

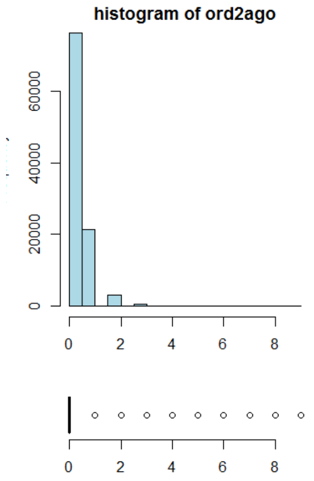

histogram of (1/yrs_since_lp+1)
0
5000
10000
15000
20000
25000
0.2
0.4
0.6
0.8
1.0
0.2
0.4
0.6
0.8
1.0

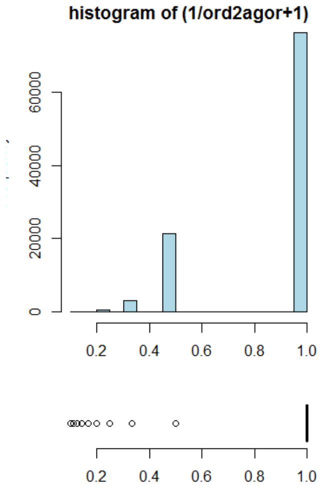

histogram of ordtyr
0
20000
40000
60000
80000
0
2
4
6
8
0
2
4
6
8

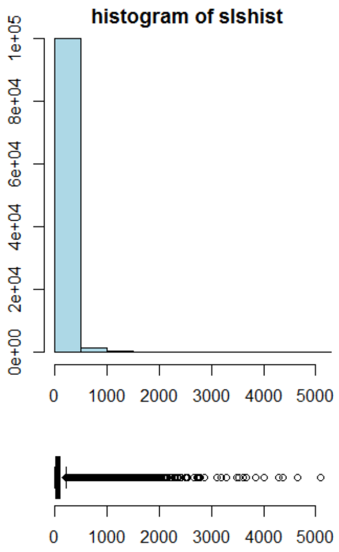

histogram of (1/ordtyr+1)
0
20000
40000
60000
80000
0.2
0.4
0.6
0.8
1.0
0.2
0.4
0.6
0.8
1.0

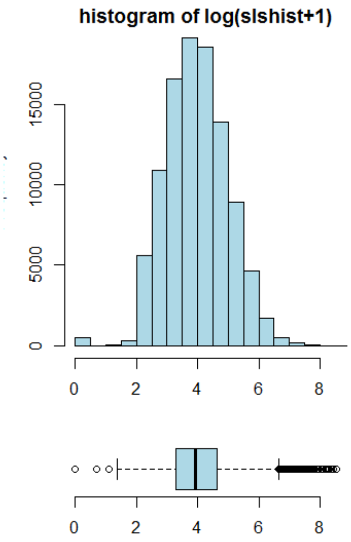

histogram of ordlyr
0
20000
40000
60000
80000
0
2
4
6
8
10
0
2
4
6
8
10

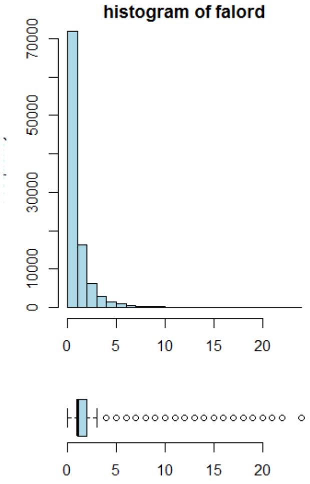

histogram of (1/ordlyr+1)
0
20000
40000
60000
80000
0.2
0.4
0.6
0.8
1.0
0.2
0.4
0.6
0.8
1.0

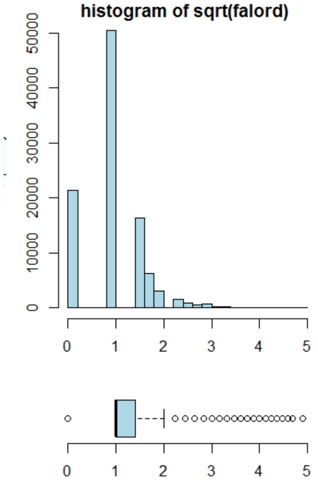

histogram of ord2ago
0
20000
40000
60000
0
2
4
6
8
0
2
4
6
8

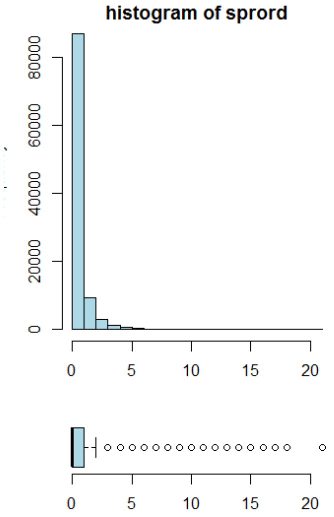

histogram of (1/ord2agor+1)
0
20000
40000
60000
0.2
0.4
0.6
0.8
1.0
0.2
0.4
0.6
0.8
1.0

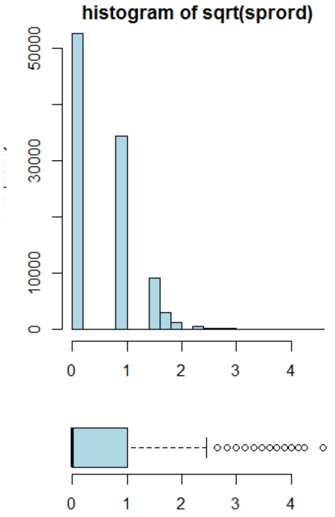

histogram of slshist


histogram of log(slshist+1)

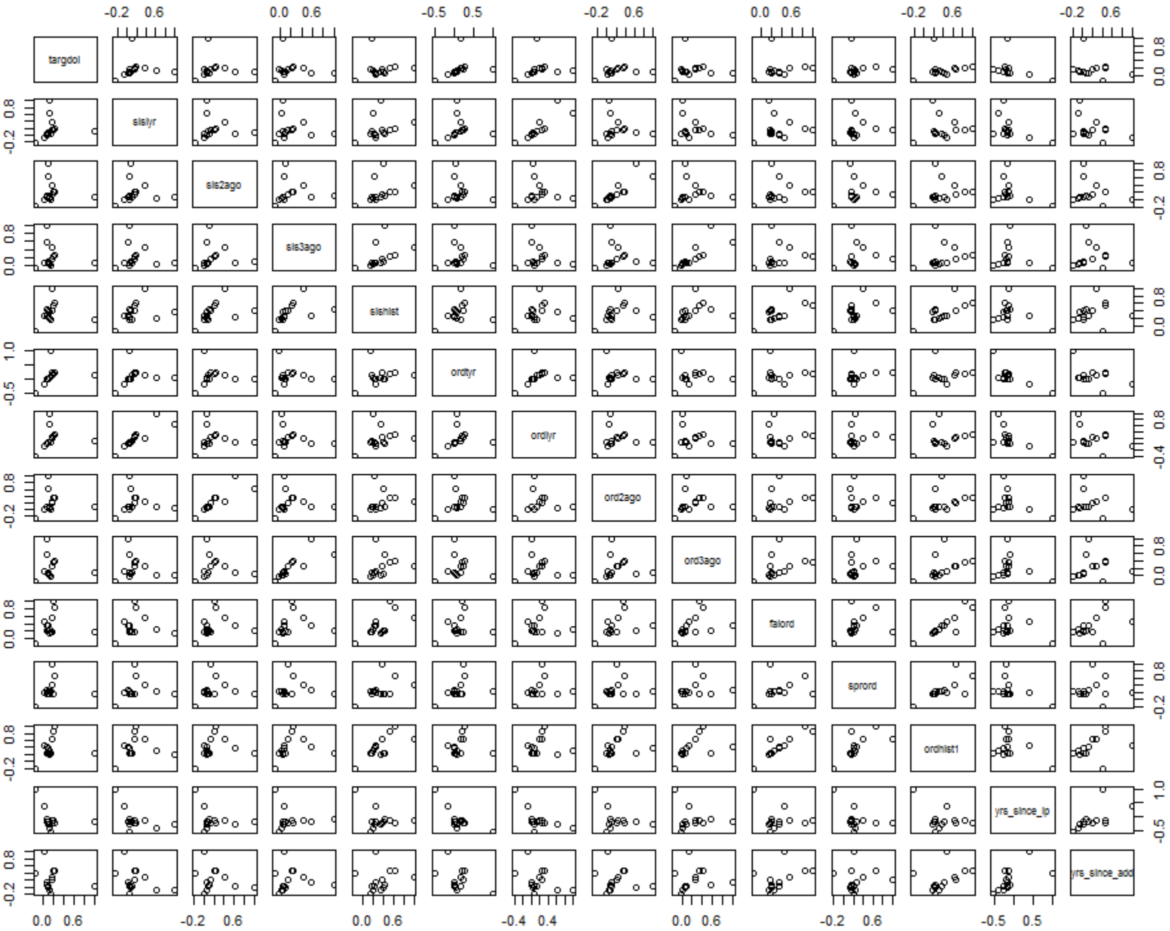

histogram of falord


histogram of sqrt(falord)

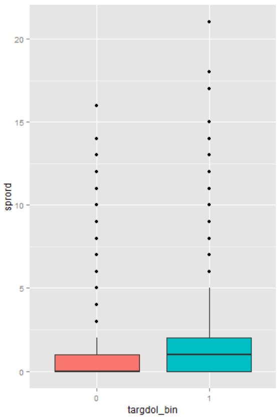

histogram of sprord

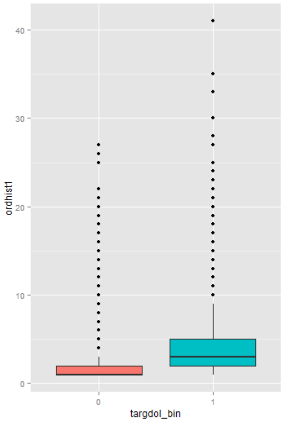

histogram of sqrt(sprord)

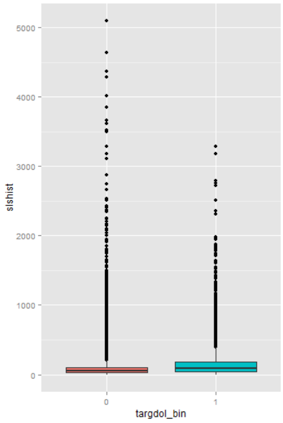
Correlations

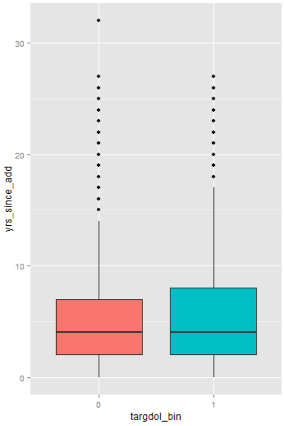


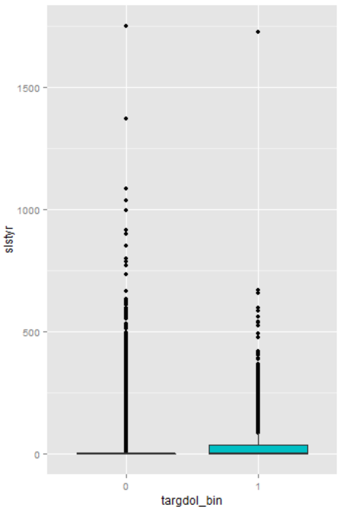
Boxplots

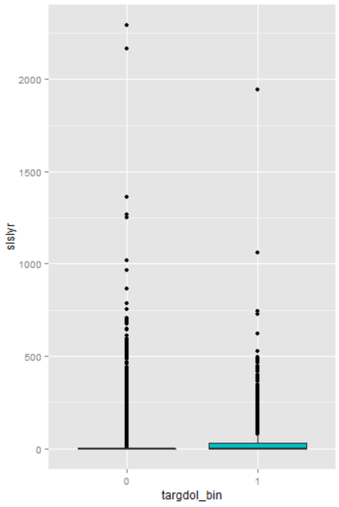


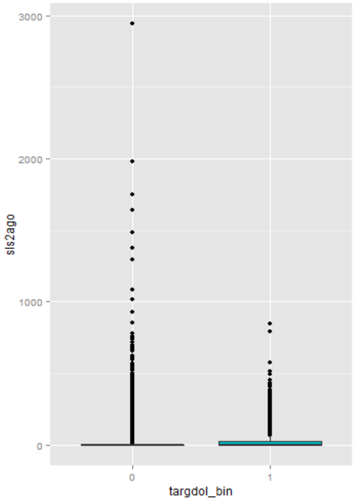


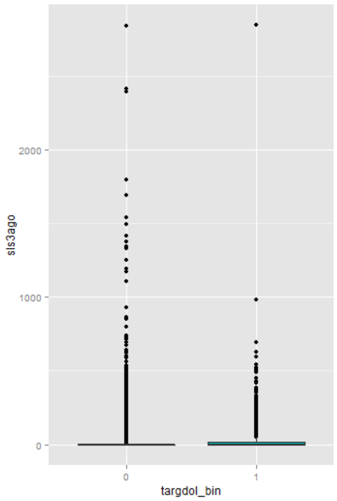


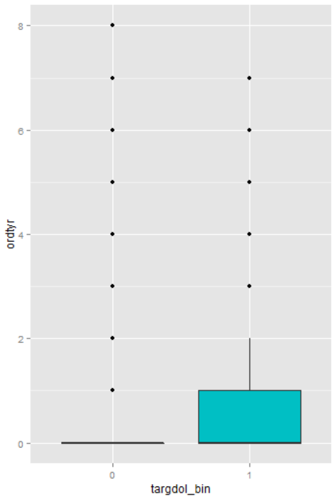


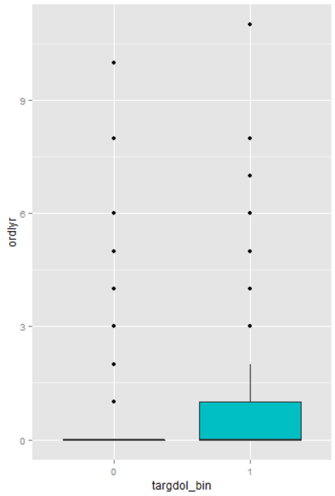

slstyr
1500
1000
500
0
0
1
targdol_bin

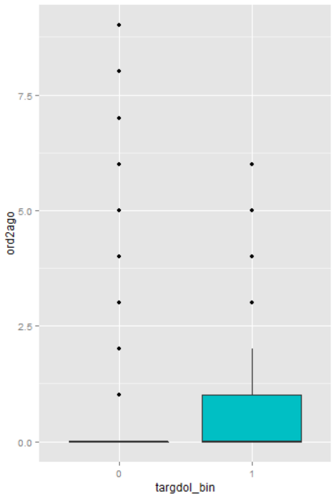

slslyr
2000
1500
1000
500
0
0
1
targdol_bin

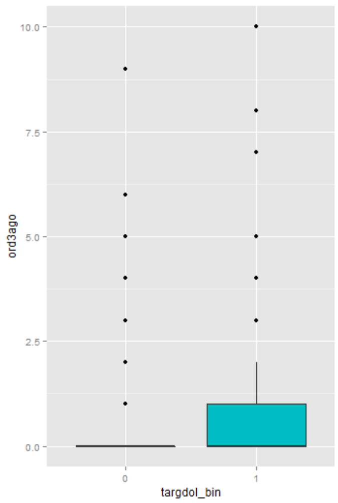

sls2ago
3000
2000
1000
0
0
1
targdol_bin

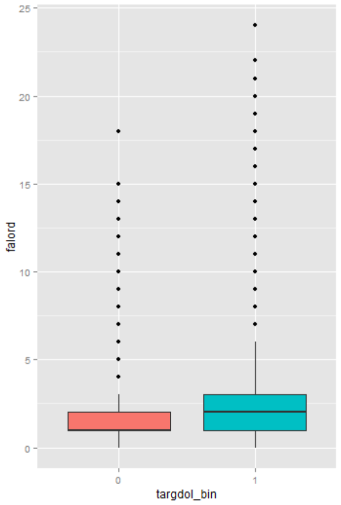

sls3ago
2000
1000
0
0
1
targdol_bin


ordtyr
8
6
4
2
0
0
1
targdol_bin


ordlyr
9
6
3
0
0
1
targdol_bin


ord2ago
7.5
5.0
2.5
0.0
0
1
targdol_bin


ord3ago
10.0
7.5
5.0
2.5
0.0
0
1
targdol_bin


falord
25
20
15
10
5
0
0
1
targdol_bin

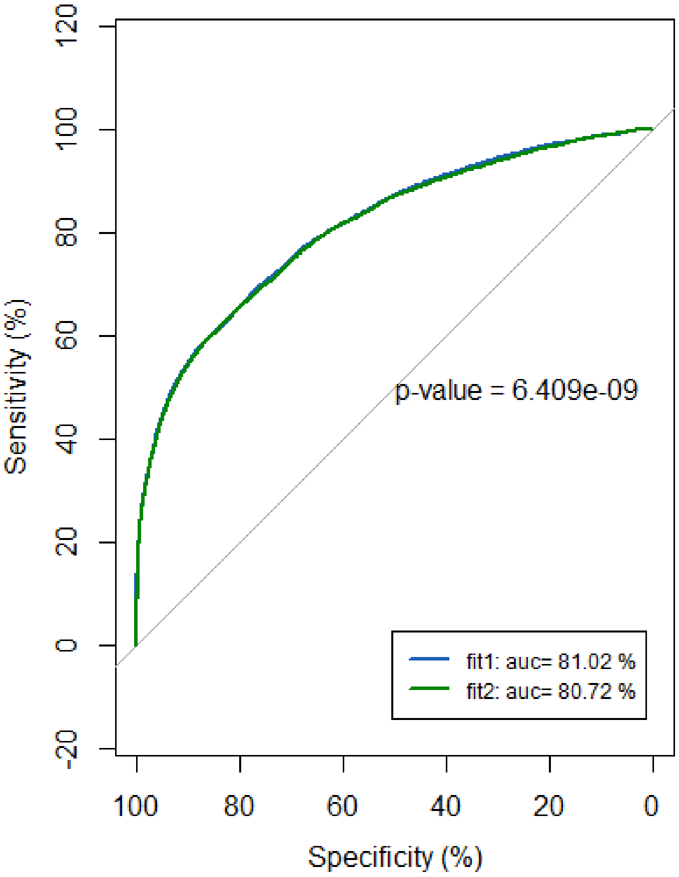
B. Classification Model

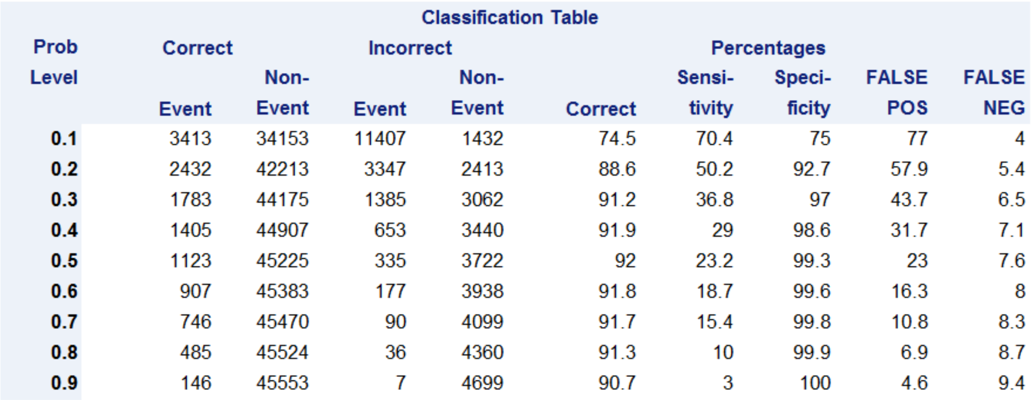
Model selection

```
fit.log1 = glm(formula = targdol_bin ~ sqrt(yrs_since_add) + I(1/(yrs_since_lp+1)) +
                 I(1/(ordtyr+1)) + I(1/(ordlyr+1)) + I(1/(ord2ago+1)) +
                 I(1/(ord3ago+1)) + falord_bin + ordtyr_bin + ordlyr_bin + ord2ago_bin +
                 ord3ago_bin + log10(slstyr+1) + log10(slslyr+1) + log10(sls2ago+1) +
                 log10(sls3ago+1) + I(1/(ordlyr*ord2ago + 1)) + I(1/(ord2ago*ord3ago + 1)) +
                 I(1/(ordtyr*ordlyr*ord2ago*ord3ago + 1)) + I(falord/(ordhist1+1)) +
                 I(1/(ordhist1+1)) +
                 log10(slshist + 1) + I(1/(ordtyr * ordlyr + 1)) +
                 I(1/(ordtyr*ordlyr*ord2ago + 1)) +
                 I(1/(ordlyr*ord2ago*ord3ago + 1)) + sqrt(ordhist1*sprord) +
                 sqrt(falord_bin/(yrs_since_lp+1)) +
                 sqrt(ordhist1/(yrs_since_add + 1)),
               family = binomial, data = Catalog.log.train, na.action = na.exclude)

summary(fit.log1)

# This will take some time
fit.log2 = step(fit.log1, k = log(n), direction = "both")

fit.log2 = glm(formula = targdol_bin ~ sqrt(yrs_since_add) + I(1/(yrs_since_lp + 1)) +
                 I(1/(ordtyr + 1)) + I(1/(ordlyr + 1)) + I(1/(ord2ago + 1)) +
                 I(1/(ord3ago + 1)) + falord_bin + ordtyr_bin + log10(slstyr + 1) +
                 log10(slslyr + 1) + log10(sls3ago + 1) +
                 I(1/(ordtyr * ordlyr * ord2ago * ord3ago + 1)) +
                 log10(slshist + 1) + I(1/(ordtyr * ordlyr + 1)) +
                 sqrt(ordhist1 * sprord) + sqrt(ordhist1/(yrs_since_add + 1)),
               family = binomial, data = Catalog.log.train, na.action = na.exclude)

summary(fit.log2)
# AIC = 24225 (16 predictors)

vif(fit.log2)
# Still have some big VIF issues

# Removing ordtyr_bin (due to high VIF)
fit.log3 = glm(formula = targdol_bin ~ sqrt(yrs_since_add) + I(1/(yrs_since_lp + 1)) +
                 I(1/(ordtyr + 1)) + I(1/(ordlyr + 1)) + I(1/(ord2ago + 1)) +
                 I(1/(ord3ago + 1)) + falord_bin + log10(slstyr + 1) +
                 log10(slslyr + 1) + log10(sls3ago + 1) +
                 I(1/(ordtyr * ordlyr * ord2ago * ord3ago + 1)) +
                 log10(slshist + 1) + I(1/(ordtyr * ordlyr + 1)) +
                 sqrt(ordhist1 * sprord) + sqrt(ordhist1/(yrs_since_add + 1)),
               family = binomial, data = Catalog.log.train, na.action = na.exclude)

summary(fit.log3)
# AIC = 24374 (15 predictors)

vif(fit.log3)
# Still have some VIF issues

# Removing slstyr, slslyr, and sls3ago (due to VIF)
fit.log4 = glm(formula = targdol_bin ~ sqrt(yrs_since_add) + I(1/(yrs_since_lp + 1)) +
                 I(1/(ordtyr + 1)) + I(1/(ordlyr + 1)) + I(1/(ord2ago + 1)) +
                 I(1/(ord3ago + 1)) + falord_bin +
                 I(1/(ordtyr * ordlyr * ord2ago * ord3ago + 1)) +
                 log10(slshist + 1) + I(1/(ordtyr * ordlyr + 1)) +
                 sqrt(ordshist1 * sprord) + sqrt(ordhist1/(yrs_since_add + 1)),
               family = binomial, data = Catalog.log.train, na.action = na.exclude)

summary(fit.log4)
# AIC = 24398 (12 predictors)

vif(fit.log4)
```

```
# ordtyr's VIF = 11.167369

# Removing ord3ago (due to relatively low significance)
fit.log5 = glm(formula = targdol_bin ~ sqrt(yrs_since_add) + I(1/(yrs_since_lp + 1)) +
                 I(1/(ordtyr + 1)) + I(1/(ordlyr + 1)) + I(1/(ord2ago + 1)) +
                 falord_bin +
                 I(1/(ordtyr * ordlyr * ord2ago * ord3ago + 1)) +
                 log10(slshist + 1) + I(1/(ordtyr * ordlyr + 1)) +
                 sqrt(ordhist1 * sprord) + sqrt(ordhist1/(yrs_since_add + 1)),
               family = binomial, data = Catalog.log.train, na.action = na.exclude)

summary(fit.log5)
# AIC = 24402 (11 predictors)

vif(fit.log5)
# ordtyr's VIF = 10.743075 (good enough I think)


fit.log6 = glm(formula = targdol_bin ~ sqrt(yrs_since_add) + sqrt(yrs_since_lp) +
                 sqrt(ordtyr) + sqrt(ordlyr) + sqrt(ord2ago) +
                 sqrt(ord3ago) + falord_bin + ordtyr_bin + ordlyr_bin + ord2ago_bin +
                 ord3ago_bin + log10(slstyr+1) + log10(slslyr+1) + log10(sls2ago+1) +
                 log10(sls3ago+1) + sqrt(ordlyr*ord2ago) + sqrt(ord2ago*ord3ago) +
                 sqrt(ordtyr*ordlyr*ord2ago*ord3ago) + I(falord/(ordhist1+1)) +
                 sqrt(ordhist1) +
                 log10(slshist + 1) + sqrt(ordtyr * ordlyr) +
                 sqrt(ordtyr*ordlyr*ord2ago) +
                 sqrt(ordlyr*ord2ago*ord3ago) + sqrt(ordhist1*sprord) +
                 sqrt(falord_bin/(yrs_since_lp+1)) +
                 sqrt(ordhist1/(yrs_since_add + 1)),
               family = binomial, data = Catalog.log.train, na.action = na.exclude)

summary(fit.log6)

fit.log7 = step(fit.log6, k = log(n), direction = "both")

fit.log7 = glm(formula = targdol_bin ~ sqrt(yrs_since_add) + sqrt(yrs_since_lp) +
                 sqrt(ordtyr) + sqrt(ordlyr) + sqrt(ord2ago) + sqrt(ord3ago) +
                 falord_bin + ordtyr_bin + log10(slstyr + 1) + log10(slslyr + 1) +
                 log10(sls3ago + 1) + sqrt(ordlyr * ord2ago) + sqrt(ord2ago * ord3ago) +
                 sqrt(ordtyr * ordlyr * ord2ago * ord3ago) + log10(slshist + 1) +
                 sqrt(ordtyr * ordlyr) + sqrt(ordhist1 * sprord) +
                 sqrt(falord_bin/(yrs_since_lp + 1)) + sqrt(ordhist1/(yrs_since_add + 1)),
               family = binomial, data = Catalog.log.train, na.action = na.exclude)

summary(fit.log7)
# AIC = 24681 (19 predictors)

vif(fit.log7)
# Pretty big VIF issues again

# Removing ordtyr_bin (due to VIF issues)
fit.log8 = glm(formula = targdol_bin ~ sqrt(yrs_since_add) + sqrt(yrs_since_lp) +
                 sqrt(ordtyr) + sqrt(ordlyr) + sqrt(ord2ago) + sqrt(ord3ago) +
                 falord_bin + log10(slstyr + 1) + log10(slslyr + 1) +
                 log10(sls3ago + 1) + sqrt(ordlyr * ord2ago) + sqrt(ord2ago * ord3ago) +
                 sqrt(ordtyr * ordlyr * ord2ago * ord3ago) + log10(slshist + 1) +
                 sqrt(ordtyr * ordlyr) + sqrt(ordhist1 * sprord) +
                 sqrt(falord_bin/(yrs_since_lp + 1)) + sqrt(ordhist1/(yrs_since_add + 1)),
               family = binomial, data = Catalog.log.train, na.action = na.exclude)

summary(fit.log8)
# AIC = 24773 (18 predictors)

vif(fit.log8)
# Still have a bunch of issues

# Removing slstyr, slslyr, sls3ago, and falord_bin (due to VIF issues)
fit.log9 = glm(formula = targdol_bin ~ sqrt(yrs_since_add) + sqrt(yrs_since_lp) +
                 sqrt(ordtyr) + sqrt(ordlyr) + sqrt(ord2ago) + sqrt(ord3ago) +
         sqrt(ordlyr * ord2ago) + sqrt(ord2ago * ord3ago) +
         sqrt(ordtyr * ordlyr * ord2ago * ord3ago) + log10(slshist + 1) +
         sqrt(ordtyr * ordlyr) + sqrt(ordhist1 * sprord) +
         sqrt(falord_bin/(yrs_since_lp + 1)) + sqrt(ordhist1/(yrs_since_add + 1)),
        family = binomial, data = Catalog.log.train, na.action = na.exclude)

summary(fit.log9)
# AIC = 24826 (14 predictors)

vif(fit.log9)
# All VIFs < 10
```

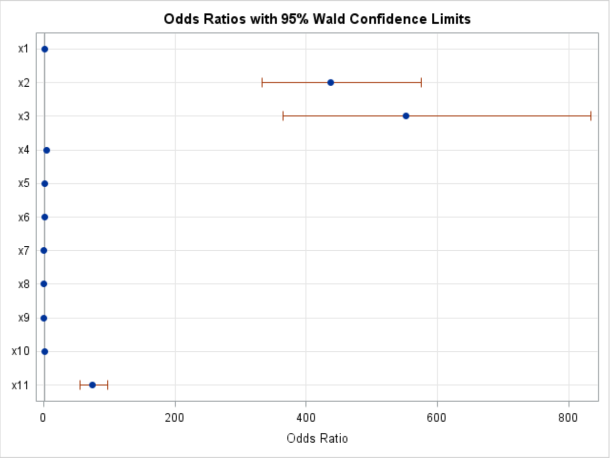
log5

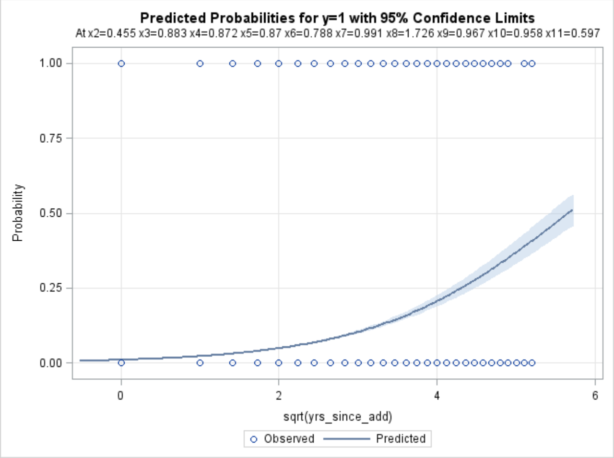


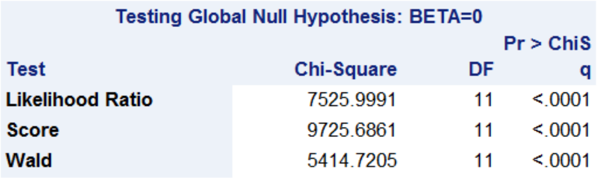


Classification Table

Prob	Correct		Incorrect		Percentages				
Level		Non-		Non-		Sensi-	Speci-	FALSE	FALSE
	Event	Event	Event	Event	Correct	tivity	ficity	POS	NEG
0.1	3413	34153	11407	1432	74.5	70.4	75	77	4
0.2	2432	42213	3347	2413	88.6	50.2	92.7	57.9	5.4
0.3	1783	44175	1385	3062	91.2	36.8	97	43.7	6.5
0.4	1405	44907	653	3440	91.9	29	98.6	31.7	7.1
0.5	1123	45225	335	3722	92	23.2	99.3	23	7.6
0.6	907	45383	177	3938	91.8	18.7	99.6	16.3	8
0.7	746	45470	90	4099	91.7	15.4	99.8	10.8	8.3
0.8	485	45524	36	4360	91.3	10	99.9	6.9	8.7
0.9	146	45553	7	4699	90.7	3	100	4.6	9.4

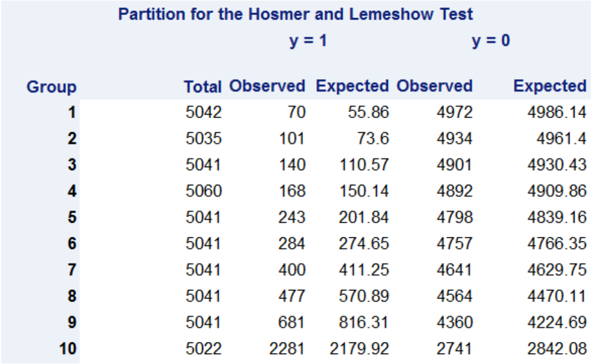

Odds Ratios with 95% Wald Confidence Limits


Predicted Probabilities for y=1 with 95% Confidence Limits
At x2=0.455 x3=0.883 x4=0.872 x5=0.87 x6=0.788 x7=0.991 x8=1.726 x9=0.967 x10=0.958 x11=0.597



Testing Global Null Hypothesis: BETA=0

Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	7525.9991	11	<.0001
Score	9725.6861	11	<.0001
Wald	5414.7205	11	<.0001



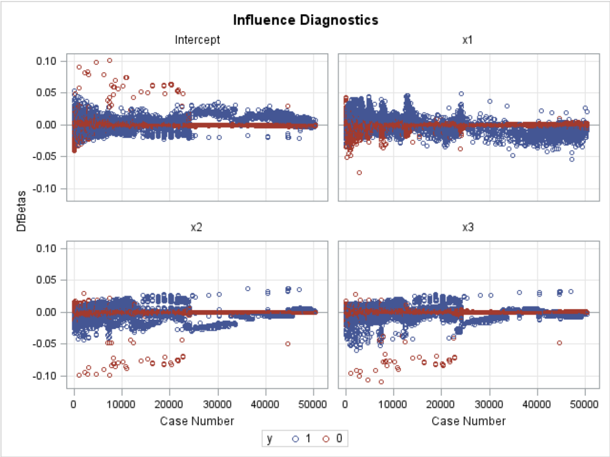
Partition for the Hosmer and Lemeshow Test

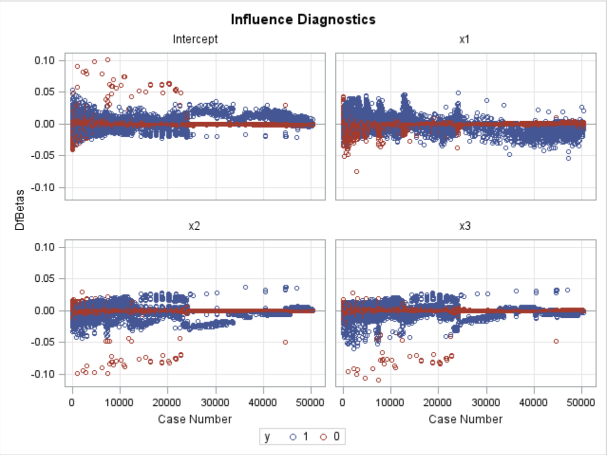
		y = 1		y = 0	
Group	Total	Observed	Expected	Observed	Expected
1	5042	70	55.86	4972	4986.14
2	5035	101	73.6	4934	4961.4
3	5041	140	110.57	4901	4930.43
4	5060	168	150.14	4892	4909.86
5	5041	243	201.84	4798	4839.16
6	5041	284	274.65	4757	4766.35
7	5041	400	411.25	4641	4629.75
8	5041	477	570.89	4564	4470.11
9	5041	681	816.31	4360	4224.69
10	5022	2281	2179.92	2741	2842.08

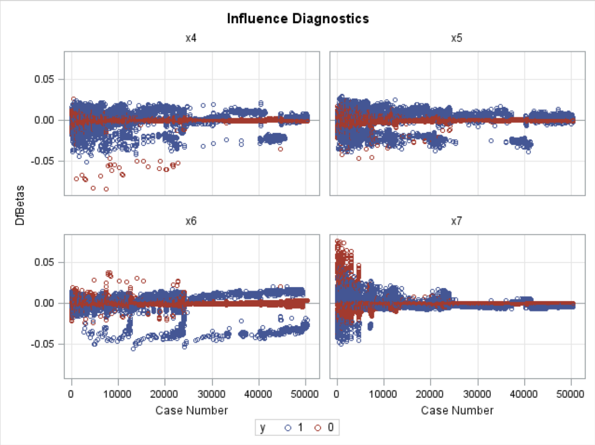
Hosmer and Lemeshow Goodness-of-Fit Test		
Chi-Square	DF	Pr > ChiSq
86.0456	8	<.0001

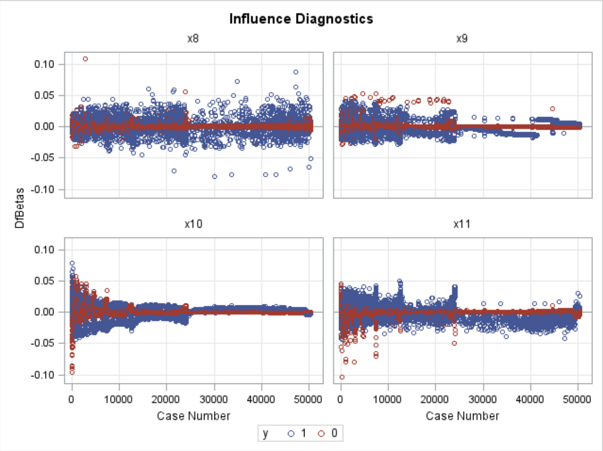
Association of Predicted Probabilities and Observed Responses			
Percent Concordant	81	**Somers' D**	0.62
Percent Discordant	19	**Gamma**	0.62
Percent Tied	0	**Tau-a**	0.108
Pairs	220738200	**c**	0.81

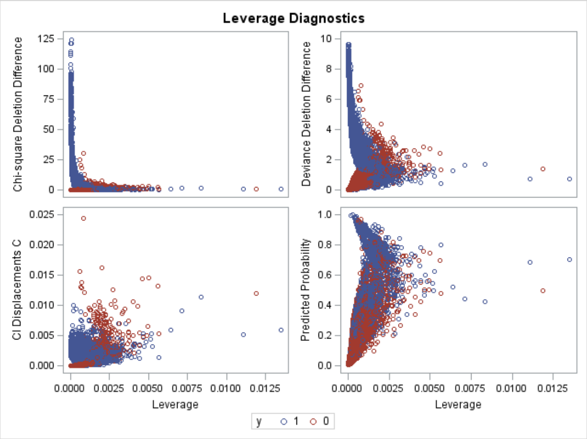
Variable	VIF
sqrt(yrs_since_add)	3.85
I(1/(yrs_since_lp + 1))	7.51
I(1/(ordtyr + 1))	10.74
I(1/(ordlyr + 1))	3.02
I(1/(ord2ago + 1))	1.39
falord_bin	1.29
I(1/(ordtyr * ordlyr * ord2ago * ord3ago + 1))	1.50
log10(slshist + 1)	2.15
I(1/(ordtyr * ordlyr + 1))	3.50
sqrt(ordhist1 * sprord)	2.50
sqrt(ordhist1/(yrs_since_add + 1))	3.95

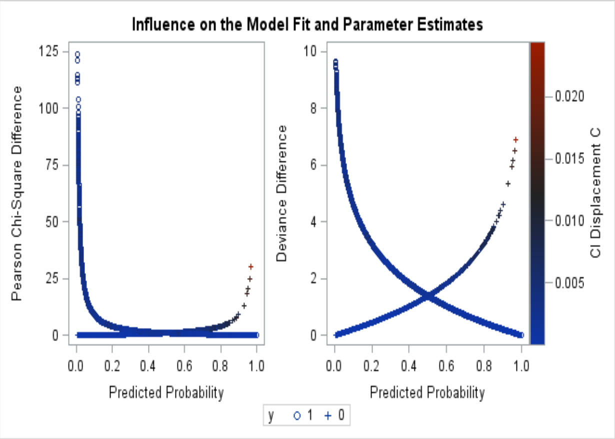


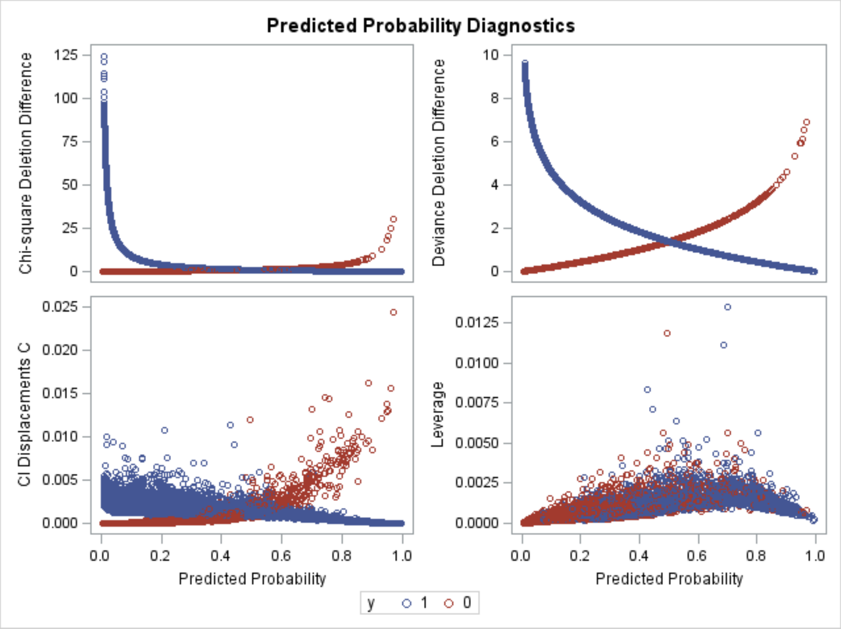



Influence Diagnostics
x4
x5
x6
x7
DfBetas
0.05
0.00
-0.05
0
10000
20000
30000
40000
50000
Case Number
y ○ 1 ○ 0


Influence Diagnostics
x8
x9
x10
x11
DfBetas
0.10
0.05
0.00
-0.05
-0.10
0
10000
20000
30000
40000
50000
Case Number
y ○ 1 ○ 0

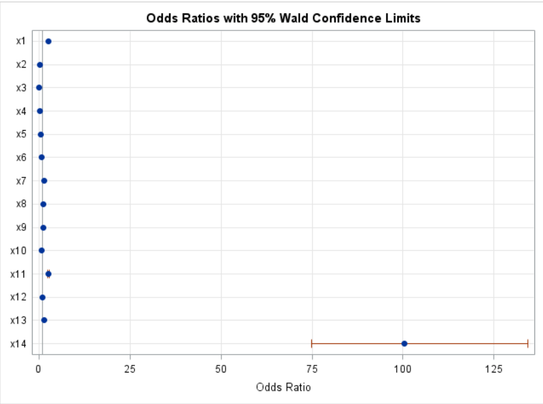

Leverage Diagnostics
Chi-square Deletion Difference
0
25
50
75
100
125
Deviance Deletion Difference
0
2
4
6
8
10
CI Displacements C
0.000
0.005
0.010
0.015
0.020
0.025
Predicted Probability
0.0
0.2
0.4
0.6
0.8
1.0
0.0000
0.0025
0.0050
0.0075
0.0100
0.0125
Leverage
y ○ 1 ○ 0

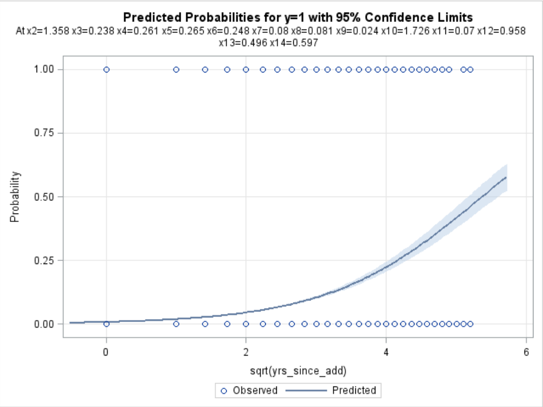

Influence on the Model Fit and Parameter Estimates
Pearson Chi-Square Difference
0
25
50
75
100
125
Deviance Difference
0
2
4
6
8
10
0.0
0.2
0.4
0.6
0.8
1.0
Predicted Probability
CI Displacement C
0.005
0.010
0.015
0.020
y ○ 1 + 0


Predicted Probability Diagnostics
Chi-square Deletion Difference
0
25
50
75
100
125
Deviance Deletion Difference
0
2
4
6
8
10
CI Displacements C
0.000
0.005
0.010
0.015
0.020
0.025
Leverage
0.0000
0.0025
0.0050
0.0075
0.0100
0.0125
0.0
0.2
0.4
0.6
0.8
1.0
Predicted Probability
y ○ 1 ○ 0

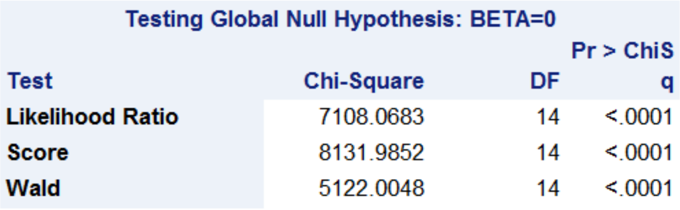
log9

Analysis of Maximum Likelihood Estimates							Odds Ratio Estimates		
				Standard	Wald			95% Wald	
					Chi-	Pr > ChiS	Point		
Label	Variable	DF	Estimate	Error	Square	q	Estimate	Confidence Limits	
Intercept	Intercept: y=1	1	-3.5799	0.1406	648.2175	<.0001			
x1	sqrt(yrs_since_add)	1	0.9083	0.0313	840.5749	<.0001	2.48	2.332	2.637
x2	sqrt(yrs_since_lp)	1	-1.9267	0.0485	1576.3	<.0001	0.146	0.132	0.16
x3	sqrt(ordtyr)	1	-2.4084	0.0882	746.0222	<.0001	0.09	0.076	0.107
x4	sqrt(ordlyr)	1	-1.0165	0.0623	266.4579	<.0001	0.362	0.32	0.409
x5	sqrt(ord2ago)	1	-0.5369	0.0537	99.9308	<.0001	0.585	0.526	0.649
x6	sqrt(ord3ago)	1	-0.3377	0.049	47.4292	<.0001	0.713	0.648	0.785
x7	sqrt(ordlyr * ord2ago)	1	0.2785	0.0601	21.4414	<.0001	1.321	1.174	1.486
x8	sqrt(ord2ago * ord3ago)	1	0.2344	0.0634	13.6526	0.0002	1.264	1.116	1.432
x9	sqrt(ordtyr * ordlyr * ord2ago * ord3ago)	1	0.2476	0.0628	15.5246	<.0001	1.281	1.133	1.449
x10	log10(slshist + 1)	1	-0.3499	0.0538	42.2198	<.0001	0.705	0.634	0.783
x11	sqrt(ordtyr * ordlyr)	1	0.9435	0.0681	192.2507	<.0001	2.569	2.248	2.936
x12	sqrt(ordhist1 * sprord)	1	-0.088	0.014	39.7012	<.0001	0.916	0.891	0.941
x13	sqrt(falord_bin/(yrs_since_lp + 1))	1	0.4098	0.0643	40.6827	<.0001	1.507	1.328	1.709
x14	sqrt(ordhist1/(yrs_since_add + 1))	1	4.6083	0.1488	958.9919	<.0001	100.318	74.939	134.291



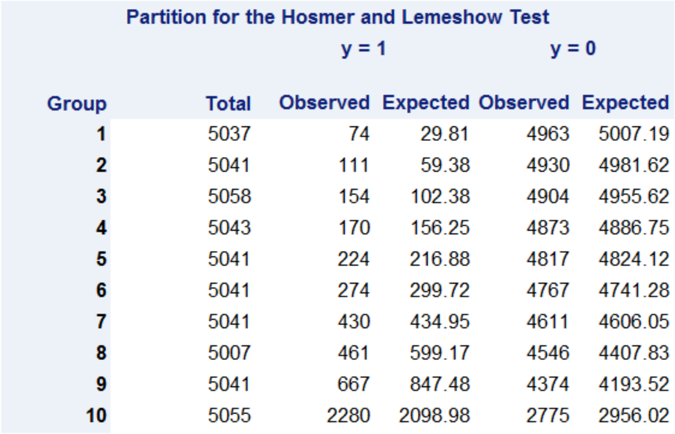


Classification Table									
Prob	Correct		Incorrect		Percentages				
Level		Non-		Non-		Sensi-	Speci-	FALSE	FALSE
	Event	Event	Event	Event	Correct	tivity	ficity	POS	NEG
0.1	3452	33368	12192	1393	73	71.2	73.2	77.9	4
0.2	2445	42035	3525	2400	88.2	50.5	92.3	59	5.4
0.3	1825	44063	1497	3020	91	37.7	96.7	45.1	6.4
0.4	1386	44918	642	3459	91.9	28.6	98.6	31.7	7.2
0.5	1075	45261	299	3770	91.9	22.2	99.3	21.8	7.7
0.6	727	45411	149	4118	91.5	15	99.7	17	8.3
0.7	447	45485	75	4398	91.1	9.2	99.8	14.4	8.8
0.8	206	45528	32	4639	90.7	4.3	99.9	13.4	9.2
0.9	40	45555	5	4805	90.5	0.8	100	11.1	9.5



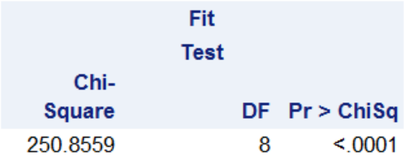
Testing Global Null Hypothesis: BETA=0

			Pr > ChiS
Test	**Chi-Square**	**DF**	**q**
Likelihood Ratio	7108.0683	14	<.0001
Score	8131.9852	14	<.0001
Wald	5122.0048	14	<.0001



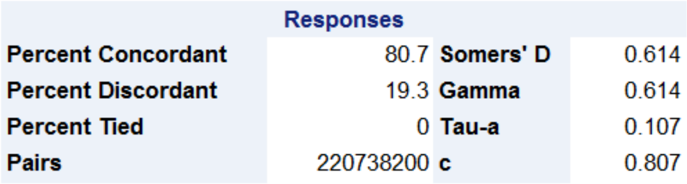
Partition for the Hosmer and Lemeshow Test

		y = 1		y = 0	
Group	**Total**	**Observed**	**Expected**	**Observed**	**Expected**
1	5037	74	29.81	4963	5007.19
2	5041	111	59.38	4930	4981.62
3	5058	154	102.38	4904	4955.62
4	5043	170	156.25	4873	4886.75
5	5041	224	216.88	4817	4824.12
6	5041	274	299.72	4767	4741.28
7	5041	430	434.95	4611	4606.05
8	5007	461	599.17	4546	4407.83
9	5041	667	847.48	4374	4193.52
10	5055	2280	2098.98	2775	2956.02



Fit Test

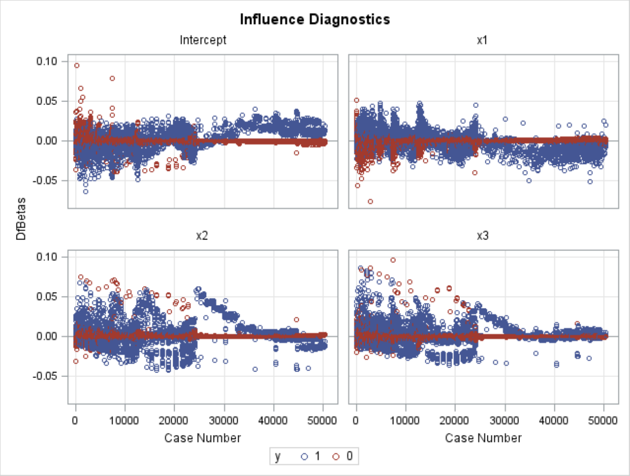
Chi-Square	DF	Pr > ChiSq
250.8559	8	<.0001

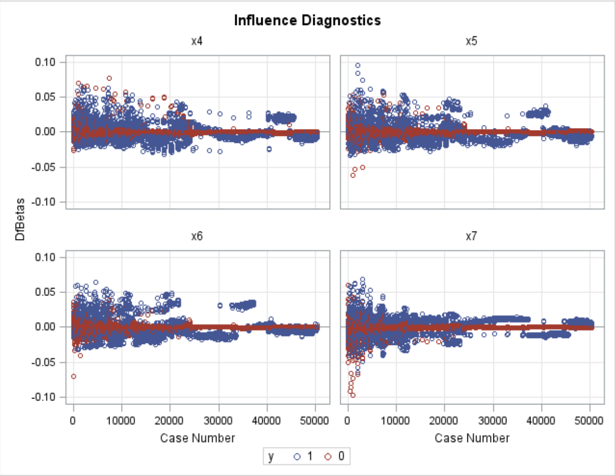


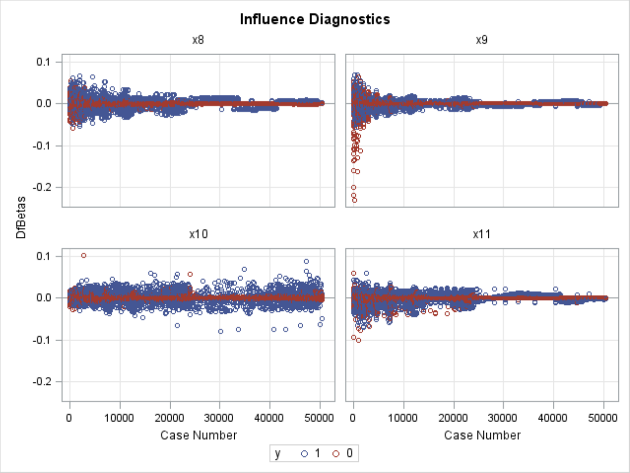
Responses

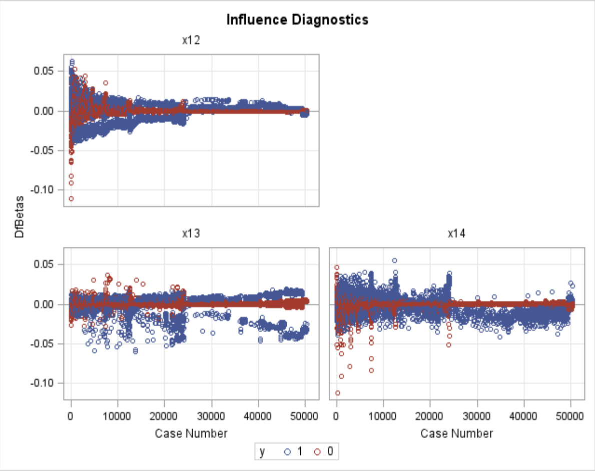
Percent Concordant	80.7	**Somers' D**	0.614
Percent Discordant	19.3	**Gamma**	0.614
Percent Tied	0	**Tau-a**	0.107
Pairs	220738200	**c**	0.807

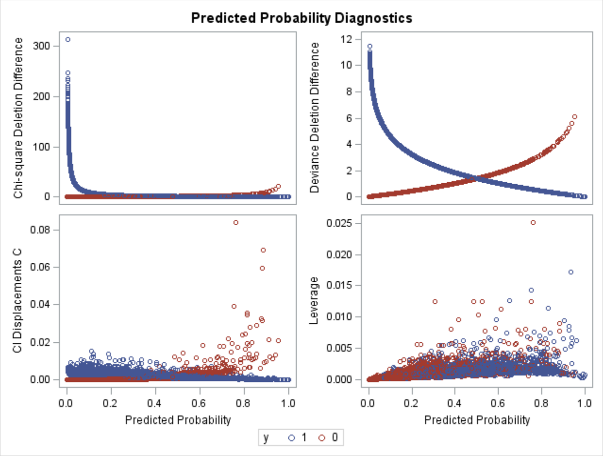
Variable	VIF
sqrt(yrs_since_add)	4.02989881
sqrt(yrs_since_lp)	5.04803261
sqrt(ordtyr)	8.12994359
sqrt(ordlyr)	4.17804593
sqrt(ord2ago)	2.96222612
sqrt(ord3ago)	2.27003811
sqrt(ordlyr * ord2ago)	2.93157133
sqrt(ord2ago * ord3ago)	3.13919223
sqrt(ordtyr * ordlyr * ord2ago * ord3ago)	1.93817058
log10(slshist + 1)	2.17246406
sqrt(ordtyr * ordlyr)	3.73807094
sqrt(ordhist1 * sprord)	2.49170136
sqrt(falord_bin/(yrs_since_lp + 1))	1.72529446
sqrt(ordhist1/(yrs_since_add + 1))	4.13730424

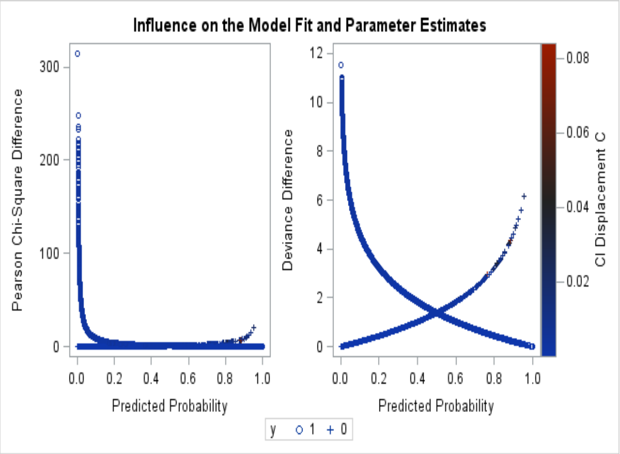

Influence Diagnostics
Intercept
x1
x2
x3
DfBetas
0.10
0.05
0.00
-0.05
0
10000
20000
30000
40000
50000
Case Number
y ○ 1 ○ 0


Influence Diagnostics
x4
x5
x6
x7
DfBetas
0.10
0.05
0.00
-0.05
-0.10
0
10000
20000
30000
40000
50000
Case Number
y ○ 1 ○ 0


Influence Diagnostics
x8
x9
x10
x11
DfBetas
0.1
0.0
-0.1
-0.2
0
10000
20000
30000
40000
50000
Case Number
y ○ 1 ○ 0

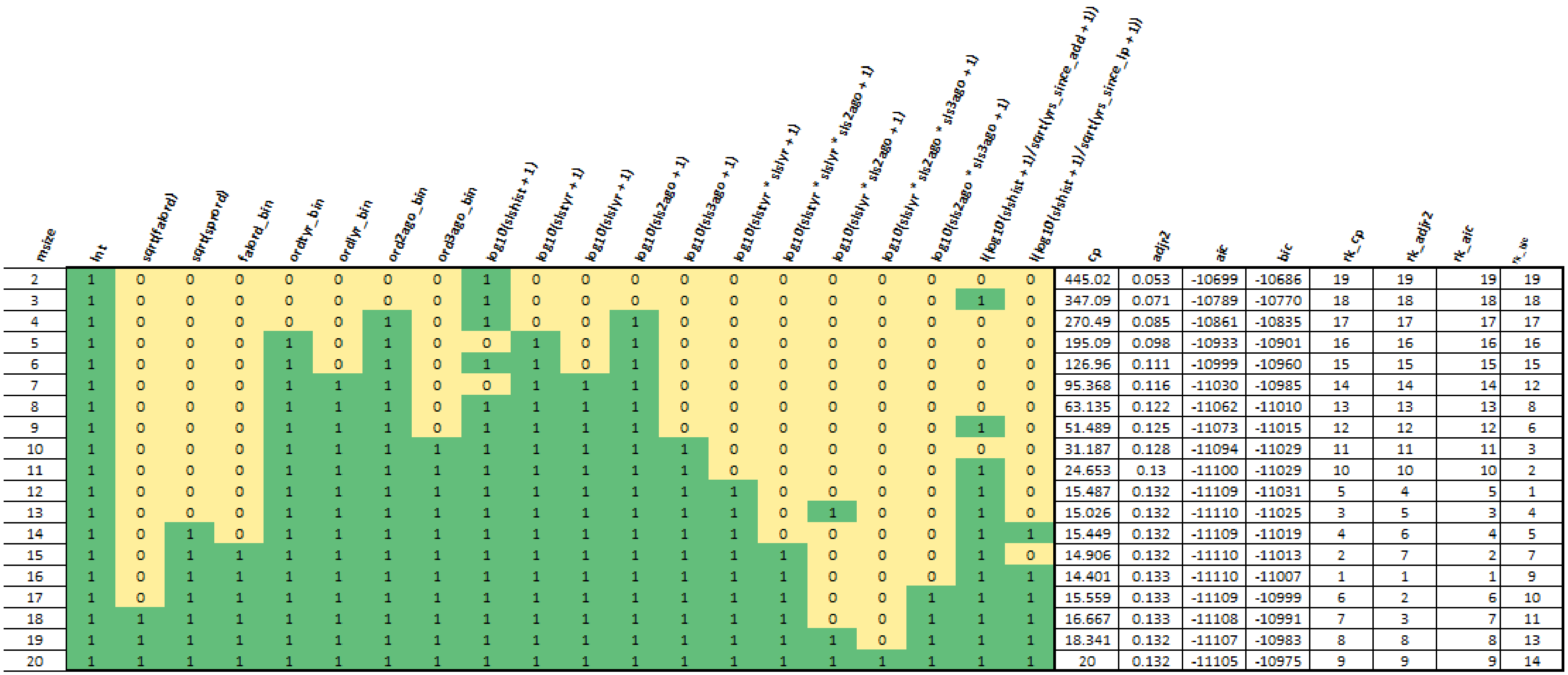

Influence Diagnostics
x12
x13
x14
DfBetas
0.05
0.00
-0.05
-0.10
0
10000
20000
30000
40000
50000
Case Number
y ○ 1 ○ 0


Predicted Probability Diagnostics
Chi-square Deletion Difference
300
200
100
0
Deviance Deletion Difference
12
10
8
6
4
2
0
CI Displacements C
0.08
0.06
0.04
0.02
0.00
Leverage
0.025
0.020
0.015
0.010
0.005
0.000
0.0
0.2
0.4
0.6
0.8
1.0
Predicted Probability
y ○ 1 ○ 0

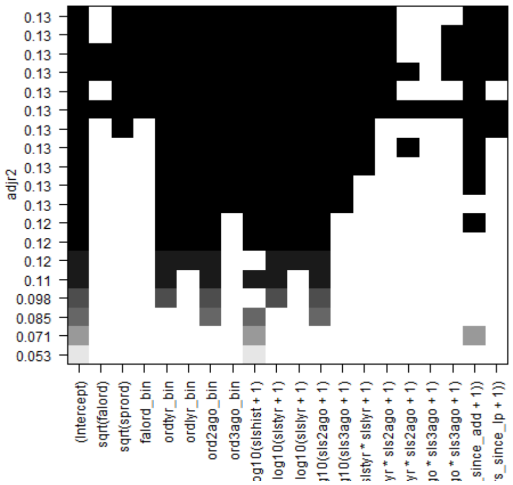

Influence on the Model Fit and Parameter Estimates
Pearson Chi-Square Difference
300
200
100
0
Deviance Difference
12
10
8
6
4
2
0
0.0
0.2
0.4
0.6
0.8
1.0
Predicted Probability
CI Displacement C
0.08
0.06
0.04
0.02
y ○ 1 + 0

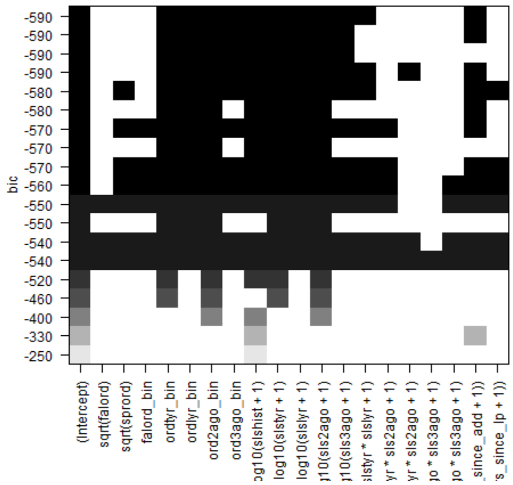
C. Multiple Regression

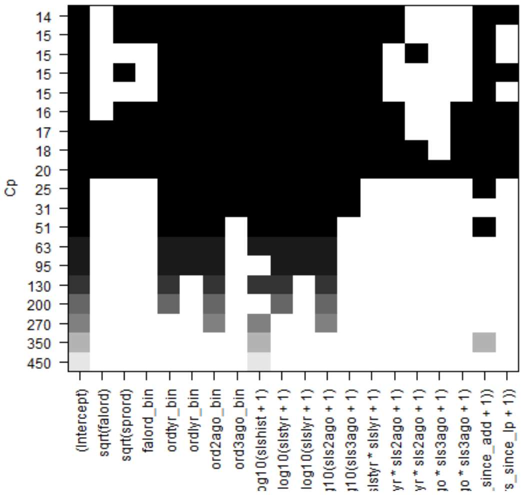
Subsets



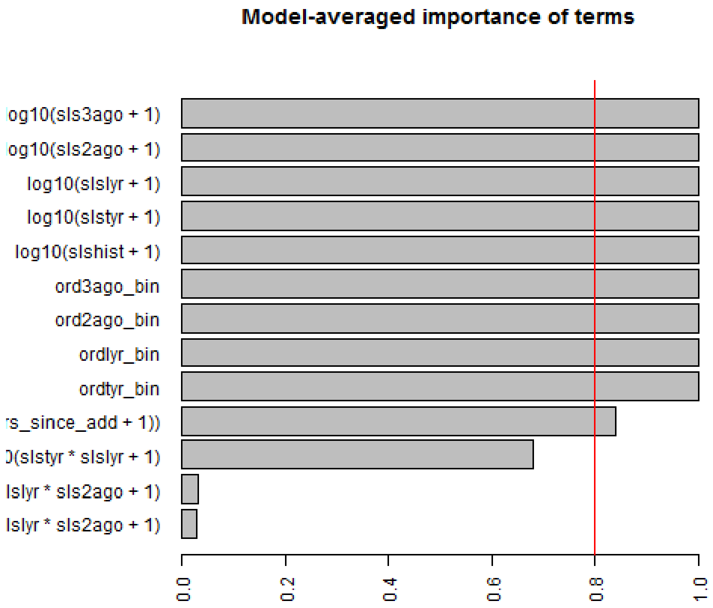
msize	Int	sqrt(fabord)	sqrt(sprord)	fabord_bin	ordtyr_bin	ordlyr_bin	ord2ago_bin	ord3ago_bin	log10(slshist + 1)	log10(slstyr + 1)	log10(slslyr + 1)	log10(sls2ago + 1)	log10(sls3ago + 1)	log10(slstyr * slslyr + 1)	log10(slstyr * slslyr * sls2ago + 1)	log10(slslyr * sls2ago + 1)	log10(slslyr * sls2ago * sls3ago + 1)	log10(sls2ago * sls3ago + 1)	I(log10(slshist + 1)/sqrt(yrs_since_add + 1))	I(log10(slshist + 1)/sqrt(yrs_since_lp + 1))	cp	adjr2	aic	bic	rk_cp	rk_adjr2	rk_aic	rk_bic
2	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	445.02	0.053	-10699	-10686	19	19	19	19
3	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	347.09	0.071	-10789	-10770	18	18	18	18
4	1	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	270.49	0.085	-10861	-10835	17	17	17	17
5	1	0	0	0	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	195.09	0.098	-10933	-10901	16	16	16	16
6	1	0	0	0	1	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	126.96	0.111	-10999	-10960	15	15	15	15
7	1	0	0	0	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	0	95.368	0.116	-11030	-10985	14	14	14	12
8	1	0	0	0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	63.135	0.122	-11062	-11010	13	13	13	8
9	1	0	0	0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	1	0	51.489	0.125	-11073	-11015	12	12	12	6
10	1	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	31.187	0.128	-11094	-11029	11	11	11	3
11	1	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0	24.653	0.13	-11100	-11029	10	10	10	2
12	1	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	1	0	15.487	0.132	-11109	-11031	5	4	5	1
13	1	0	0	0	1	1	1	1	1	1	1	1	1	1	0	1	0	0	1	0	15.026	0.132	-11110	-11025	3	5	3	4
14	1	0	1	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	1	1	15.449	0.132	-11109	-11019	4	6	4	5
15	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	14.906	0.132	-11110	-11013	2	7	2	7
16	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	1	14.401	0.133	-11110	-11007	1	1	1	9
17	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1	1	15.559	0.133	-11109	-10999	6	2	6	10
18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1	1	16.667	0.133	-11108	-10991	7	3	7	11
19	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	18.341	0.132	-11107	-10983	8	8	8	13
20	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	20	0.132	-11105	-10975	9	9	9	14


adjr2
0.13
0.13
0.13
0.13
0.13
0.13
0.13
0.13
0.13
0.13
0.13
0.12
0.12
0.12
0.11
0.098
0.085
0.071
0.053
(Intercept)
sqrt(falord)
sqrt(sprord)
falord_bin
ordtyr_bin
ordlyr_bin
ord2ago_bin
ord3ago_bin
og10(slshist + 1)
log10(slstyr + 1)
log10(slslyr + 1)
j10(sls2ago + 1)
j10(sls3ago + 1)
slstyr * slslyr + 1)
yr * sls2ago + 1)
yr * sls2ago + 1)
jo * sls3ago + 1)
jo * sls3ago + 1)
_since_add + 1))
's_since_lp + 1))

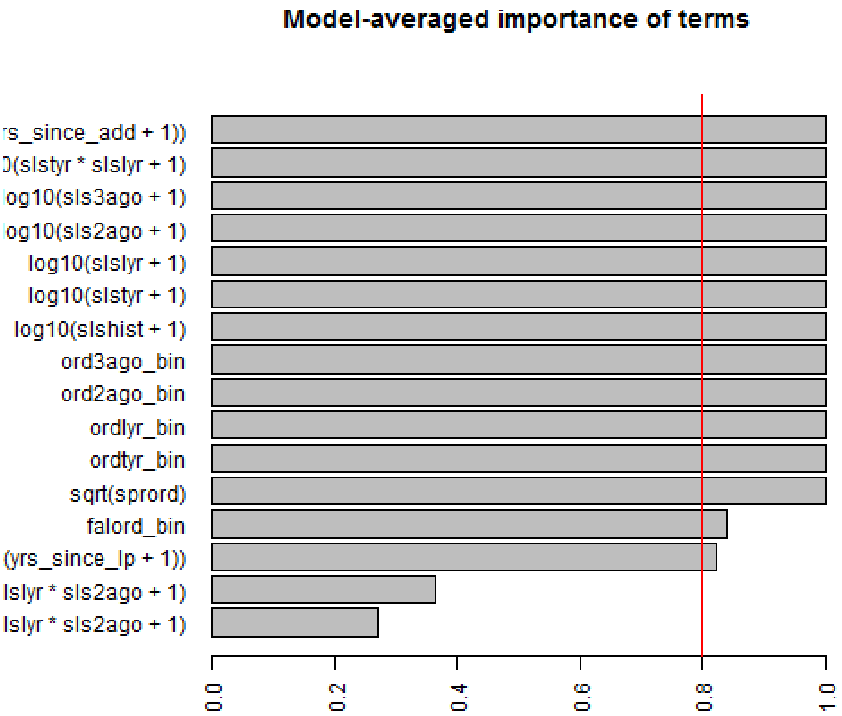

bic
-590
-590
-590
-590
-580
-580
-570
-570
-570
-560
-550
-550
-540
-540
-520
-460
-400
-330
-250
(Intercept)
sqrt(falord)
sqrt(sprord)
falord_bin
ordtyr_bin
ordlyr_bin
ord2ago_bin
ord3ago_bin
og10(slshist + 1)
log10(slstyr + 1)
log10(slslyr + 1)
j10(sls2ago + 1)
j10(sls3ago + 1)
slstyr * slslyr + 1)
yr * sls2ago + 1)
yr * sls2ago + 1)
jo * sls3ago + 1)
jo * sls3ago + 1)
_since_add + 1))
's_since_lp + 1))


Cp
14
15
15
15
15
16
17
18
20
25
31
51
63
95
130
200
270
350
450
(Intercept)
sqrt(falord)
sqrt(sprord)
falord_bin
ordtyr_bin
ordlyr_bin
ord2ago_bin
ord3ago_bin
og10(slshist + 1)
log10(slstyr + 1)
log10(slslyr + 1)
j10(sls2ago + 1)
j10(sls3ago + 1)
slstyr * slslyr + 1)
yr * sls2ago + 1)
yr * sls2ago + 1)
jo * sls3ago + 1)
jo * sls3ago + 1)
_since_add + 1))
's_since_lp + 1))

BIC:


Model-averaged importance of terms
og10(sls3ago + 1)
og10(sls2ago + 1)
log10(slslyr + 1)
log10(slstyr + 1)
log10(slshist + 1)
ord3ago_bin
ord2ago_bin
ordlyr_bin
ordtyr_bin
's_since_add + 1))
)(slstyr * slslyr + 1)
lslyr * sls2ago + 1)
lslyr * sls2ago + 1)
0.0
0.2
0.4
0.6
0.8
1.0

AIC:


Model-averaged importance of terms
's_since_add + 1))
)(slstyr * slslyr + 1)
og10(sls3ago + 1)
og10(sls2ago + 1)
log10(slslyr + 1)
log10(slstyr + 1)
log10(slshist + 1)
ord3ago_bin
ord2ago_bin
ordlyr_bin
ordtyr_bin
sqrt(sprord)
falord_bin
(yrs_since_lp + 1))
lslyr * sls2ago + 1)
lslyr * sls2ago + 1)
0.0
0.2
0.4
0.6
0.8
1.0

Model Selection

For the predicted variable, we used the base-10 log of one plus the value for **targdol**. This transformation was made in order to better normalize the predicted variable and bring in extreme values. One was added in assure that no negative values were given.

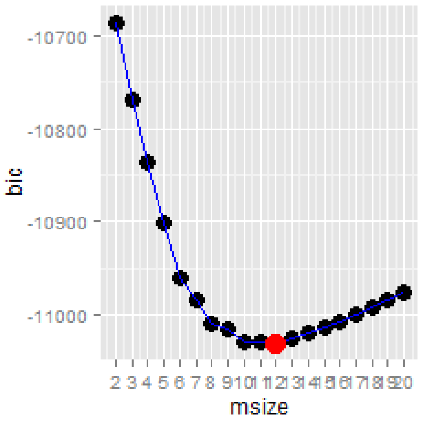
The predictors we used were: the square root of the created variable **yrs_since_add**, the square root of the inverse of one plus the derived variable **yrs_since_lp**, the square root of **ordtyr**, the square root of **ord2ago**, the square root of **ord3ago**, the square root of **falord**, the square root of **sprord**, the square root of the modified version of **ordhist** (**ordhist1**), the binary indicator variable **falord_bin**, the binary indicator variable **ordtyr_bin**, the binary indicator variable **ord2ago_bin**, the binary indicator variable **ord3ago_bin**, the base-10 log of **slshist** plus one, the base-10 log of **slslyr** plus one, the base-10 log of **sls2ago** plus one, the base-10 log of **sls3ago** plus one, the square root of the inverse of one plus the interaction between **ordtyr** and **ordlyr**, the square root of the inverse of one plus the interaction of **ordtyr, ordlyr,** and **ord2ago**, the base-10 log of the inverse of one plus the interaction between **slstyr** and **slslyr**, the base-10 log of the inverse of one plus the interaction between **slstyr**, **slslyr**, and **sls2ago**, the interaction between **falord** and the inverse of **ordhist1**, and the square root of the interaction between **ordhist1** and the inverse of one plus **yrs_since_add**. The square root and base-10 log transformation were made to predictor variables in order to normalize them, and interactions were chosen by modeler expertise as potentially-powerful predictors of dollars spent.

1. This first model, **lm1**, had a multiple R-squared value of 0.143 and adjusted R-squared value of 0.139, and included 8 of 25 predictors which were not significant at the $\alpha = 0.05$ level. Furthermore, the model suffered from a high degree of multicollinearity, as indicated by variance inflation factor (VIF) values for significant predictor variables as high as 31.13. For these reasons, the model was deemed not acceptable
2. The next model, **lm2**, was obtained by applying a stepwise regression procedure to the full model expounded upon above. The stepwise regression procedure was applied was applied in both the forward and backward directions. The output of this procedure was a new model, titled **lm2**, which included only 19 of the original 25 predictor variables from **lm1**, and achieved a multiple R-squared value of 0.0939 and adjusted R-squared value of 0.09036. However, **lm2** also suffered from multicollinearity problems, with VIF values

exceeding 25, as well as 6 of 19 predictor variables not significant at the α = 0.05 level. Thus, **lm2** was also deemed unacceptable.

3. The next model, **lm3**, was obtained by starting over from scratch with a full model that did not include square root of the order history terms, as none of these predictors had been statistically significant in previous models and suffered from multicollinearity problems. The model also included more interaction terms, most notably the log-transformed interaction between **sls2ago** and **sls3ago**, and the interaction between the log-transformed value of one plus **slshist** and the inverse of the square root of one plus the derived variable **yrs_since_lp**. This model obtained a multiple R-squared value of 0.136 and adjusted R-squared value of 0.1323, only slightly behind the original full model, **lm1**. Most importantly, **lm3** drastically improved its situation with regard to multicollinearity, with the highest predictor VIF values in the mid-teens. To improve upon this, a stepwise regression procedure was again applied in both directions, to obtain **lm4**.
4. The next model, selected from **lm3** by a stepwise regression procedure, consisted of only 12 predictor variables (the binary indicators for each year of order history, the log transformed values of one plus each sales variable, the log-transformed value of one plus the interaction between **slstyr** and **slslyr,** and the interaction between the log-transformed value of one plus **slshist** and the inverse of the square root of one plus the derived variable **yrs_since_lp**). This model achieved a multiple R-squared value of 0.317 and an adjusted R-squared value of 0.132, indicating very little loss of predictive power from the full model **lm3** from which it was derived. An analysis of predictor variance inflation factors showed slight problems with VIF values approaching 14, but nothing as major as had been seen in previous models.

   In order to ensure that **lm4** was indeed composed of the best predictors from the full model given in **lm3**, a best-subsets regression procedure was applied to the full model and cross-referenced with the model chosen by stepwise regression. The results of this procedure analyzed with respect to the Bayesian information criterion are given below:



○

As indicated, the optimal value for BIC is found in the subset of **lm3** with 12 predictors, which is notably the number of predictors in our candidate model, **lm4**. After examining the 12 predictors chosen as the most significant, we see that they are the same as those in **lm4**, providing support for our theory that it provides the best combination of predictive power and parsimony of the models available to us. It was thus decided that this model was likely to be a powerful predictor for **targdol**, and should be treated as a candidate for the final predictive model, but that alternatives without VIF problems were needed.

5. The alternative full model, **lm5**, was created using multiple least-squares regression for the base-10 log value of **targdol**, modeled by the square root of **falord**, the square root of **sprord**, the **falord** binary indicator, the **ordtyr** binary indicator**,** the **ordlyr** binary indicator, the **ord2ago** binary indicator, the **ord3ago** binary indicator, the base-10 log of **slshist** plus one, the base-10 log of one plus the interaction between **slstyr** and **slslyr**, the base-10 log of one plus the interaction between **slstyr**, **slslyr**, and **sls2ago**, the base-10 log of one plus the interaction between **slslyr** and **sls2ago**, the base-10 log of one plus the interaction between **slslyr**, **sls2ago**, and **sls3ago**, the base-10 log of one plus the interaction between **sls2ago** and **sls3ago**, the interaction between the base-10 log of one plus **slshist** and the inverse of the square root of one plus **yrs_since_add**, and the interaction between the base-10 log of one plus **slshist** and the inverse of the square root of one plus the derived variable **yrs_since_lp**. The model achieved a multiple R-squared value of 0.104 and an adjusted R-squared value of 0.101, indicating that it suffered from a possible loss of explanatory power when compared to **lm3**. However, the strength of this alternative model its lack of multicollinearity issues, as evidenced by a maximum

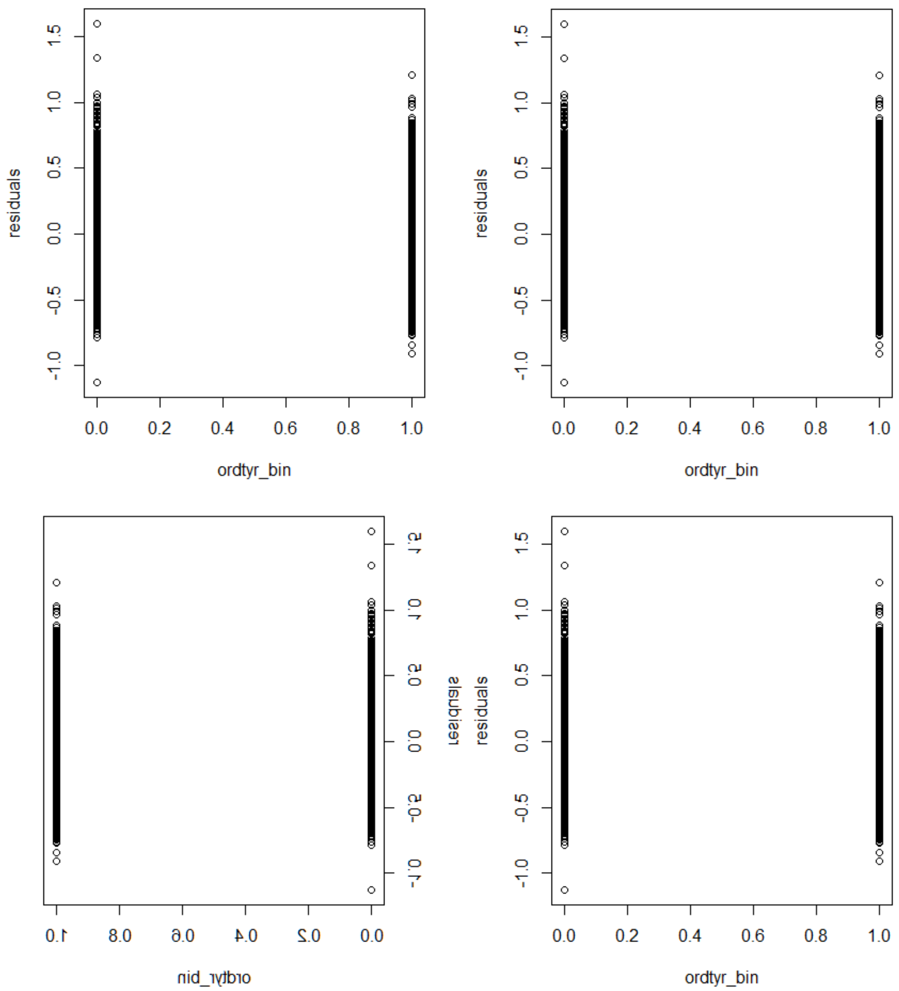
predictor VIF value of 5.778. Not all of its predictors were significant at the α= 0.05 level, though, so a stepwise regression protocol was applied in order to select which variables would be used in the final alternative model.

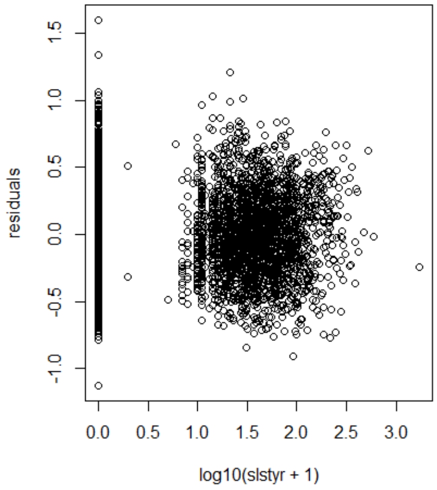
6. The final alternative model, **lm6**, was obtained via a two-direction stepwise protocol applied to **lm5**. This procedure eliminated four of **lm5**’s 15 predictor variables, leaving only the square root of **sprord**, each of the indicator variables for past orders, the log-transformed value for **slshist** + 1, the log-transformed interaction between **slstyr** and **slslyr**, the log-transformed interaction between **sls2ago** and **sls3ago**, and the two interaction terms of log(**slshist** + 1) and both the inverse of the square root of one plus **yrs_since_add** and the inverse of the square root of one plus **yrs_since_lp**. This model achieved a multiple R-squared value of 0.101 and an adjusted R-squared value of 0.0996, indicating that it is roughly as strong a predictor of **targdol** as the full model from which it was derived, but not quite as strong as our primary candidate, **lm4**. We proceeded to run model diagnostics on both **lm4** and **lm6** before settling on one as the final predictive model for **targdol** to be used in conjunction with the logistic model described above.

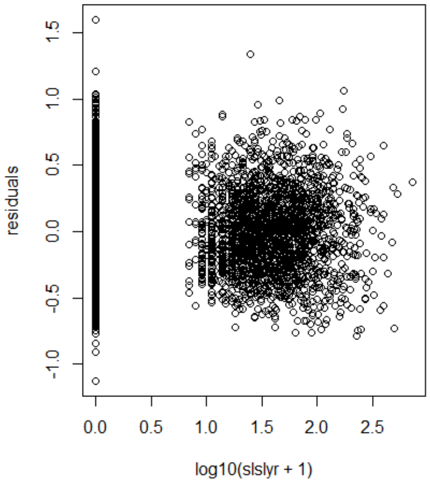
```
f1 = log10(targdol_pur+1) ~ sqrt(yrs_since_add)+sqrt(1/(yrs_since_lp+1))+sqrt(ordtyr) +
  sqrt(ordlyr)+sqrt(ord2ago)+sqrt(ord3ago)+sqrt(falord)+sqrt(sprord) +
  sqrt(ordhist1)+falord_bin+ordtyr_bin+ordlyr_bin+ord2ago_bin +
  ord3ago_bin+log10(slshist+1)+log10(slstyr+1)+log10(slslyr+1)+
  log10(sls2ago+1)+log10(sls3ago+1)+sqrt(1/(ordtyr*ordlyr+1))+
  sqrt(1/(ordtyr*ordlyr*ord2ago+1))+log10(1/(slstyr*slslyr+1))+
  log10(1/(slstyr*slslyr*sls2ago+1))+I(falord/ordhist1)+I(sqrt(ordhist1/(yrs_since_add+1)))

fit.lm1 = lm(f1, data = Catalog.lm.train)

summary(fit.lm1)
vif(fit.lm1)
# Adj.R2 = 0.1386
# Lots of insignificant coefficients; obvious multicollinearity issues

fit.lm2 = stepAIC(fit.lm1, direction = "both")

fit.lm2 = lm(formula = targdol_pur ~ sqrt(ordlyr) + sqrt(ord2ago) + sqrt(ord3ago) +
             sqrt(falord) + sqrt(sprord) + sqrt(ordhist1) + falord_bin +
             ordtyr_bin + ordlyr_bin + ord2ago_bin + ord3ago_bin + log10(slshist + 1) +
             log10(slstyr + 1) + log10(slslyr + 1) + log10(sls2ago + 1) +
             log10(sls3ago + 1) + log10(1/(slstyr * slslyr + 1)) +
             log10(1/(slstyr * slslyr * sls2ago + 1)) +
             I(sqrt(ordhist1/(yrs_since_add + 1))), data = Catalog.lm.train)

summary(fit.lm2)
vif(fit.lm2)
# Adj.R2 = 0.09036
# Still have insignificant predictors and plenty of multicollinearity
AIC(fit.lm2)
# 51910.31
# Starting over w/o sqrt(ord)'s and w/ more interactions
f3 = log10(targdol_pur+1) ~ sqrt(falord) + sqrt(sprord) +
  falord_bin + ordtyr_bin + ordlyr_bin + ord2ago_bin + ord3ago_bin +
  log10(slshist + 1) + log10(slstyr + 1) + log10(slslyr + 1) +
  log10(sls2ago + 1) + log10(sls3ago + 1) + log10(slstyr * slslyr + 1) +
  log10(slstyr * slslyr * sls2ago + 1) + log10(slslyr*sls2ago+1) +
  log10(slslyr*sls2ago*sls3ago+1) + log10(sls2ago*sls3ago+1) +
  I(log10(slshist+1)/sqrt(yrs_since_add + 1)) +
  I(log10(slshist+1)/sqrt(yrs_since_lp+1))

fit.lm3 = lm(formula = f3, data = Catalog.lm.train)

summary(fit.lm3)
vif(fit.lm3)

# Adj.R2 = 0.1323

n = dim(Catalog.lm.train)[1]

fit.lm4 = step(fit.lm3, k = log(n), direction = "both")

summary(fit.lm4)
# Adj.R2 = 0.1316 (Best Model)
AIC(fit.lm4)
# 2641.076

vif(fit.lm4)

fit.lm5 = lm(formula = log10(targdol_pur+1) ~ sqrt(falord) + sqrt(sprord) +
               falord_bin + ordtyr_bin + ordlyr_bin + ord2ago_bin + ord3ago_bin +
               log10(slshist + 1) + log10(slstyr * slslyr + 1) +
               log10(slstyr * slslyr * sls2ago + 1) + log10(slslyr*sls2ago+1) +
               log10(slslyr*sls2ago*sls3ago+1) + log10(sls2ago*sls3ago+1) +
               I(log10(slshist+1)/sqrt(yrs_since_add + 1)) +
               I(log10(slshist+1)/sqrt(yrs_since_lp+1)), data = Catalog.lm.train)

summary(fit.lm5)
# Adj.R2 = 0.101

vif(fit.lm5)

fit.lm6 = step(fit.lm5, k = log(n), direction = "both")

summary(fit.lm6)
# Adj.R2 = 0.09956 (Alternative Model)

vif(fit.lm6)

# # step using AIC (both LR and LM) ####
# library(MASS)
# fit_stepAIC = step(object = full, direction = "both")  # AIC
#
# # step using BIC (both LR and LM) ####
# fit_stepAIC = step(object = full, direction = "both", k=log(n))  # BIC
#
# # use single drop/add (both LR and LM) ####
# drop1(fit, test="F",data=mydata) # LRT on individual terms
# add1(fit, scope= ~ X1 + X2, test="F",data=mydata) # LRT on individual terms

# use regsubsets (LM only) ####
source("ModelSelection.R")
lm_select1 = regsubsetsF2(f3,Catalog.lm.train)
lm_select1$subsets

write.csv(lm_select1$subsets, "subsets_lm.csv")

# use glmulti (both LM and LR) ####
source("ModelSelection.R")
lm_select2 = glmulti_gaussian(f3,Catalog.lm.train)
lm_select2
```

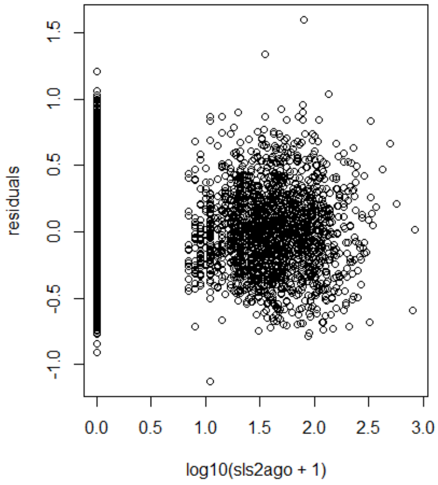
```
# use bic.glm (both LR and LM) ####
# http://www.medicine.mcgill.ca/epidemiology/joseph/courses/EPIB-621/logselect.pdf
# Bayesian Model Averaging accounts for the model uncertainty inherent in the variable
# selection problem by averaging over the best models in the model class according
# to approximate posterior model probability.
# install.packages("BMA")
library(BMA)
output=bic.glm(f1,data=Catalog.lm.train,glm.family=gaussian)
summary(output)
names(output)

output$label # variables in the model
output$postprob # best model = highest posterior probability
output$probne0 # probability variable should be in the model
write.csv(output$probne0, "probInModel_lm.csv")
```

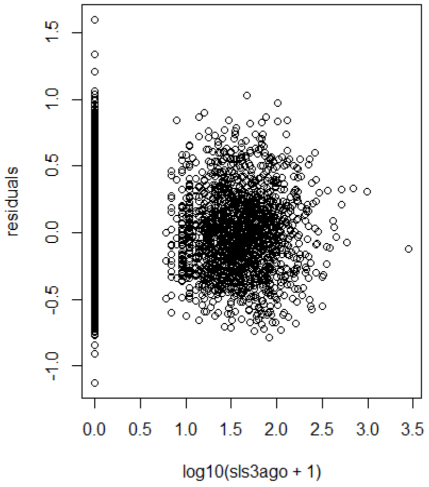
lm4

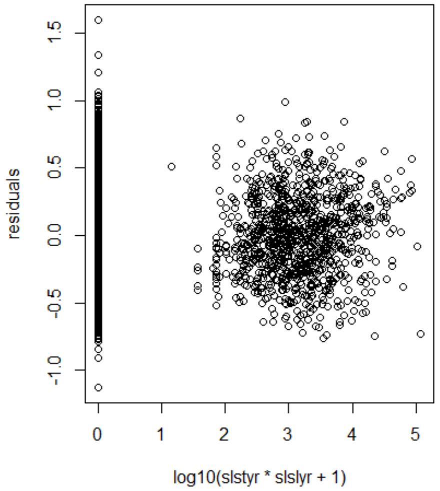


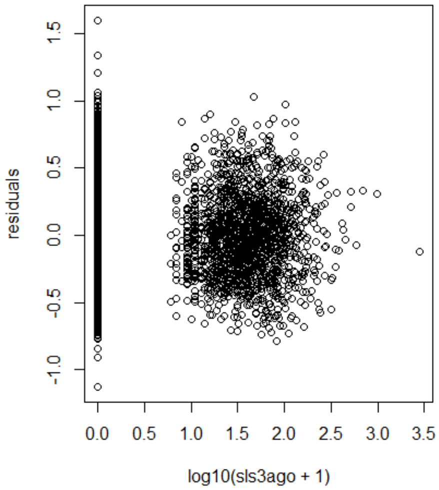

residuals
-1.0
-0.5
0.0
0.5
1.0
1.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
log10(slstyr + 1)

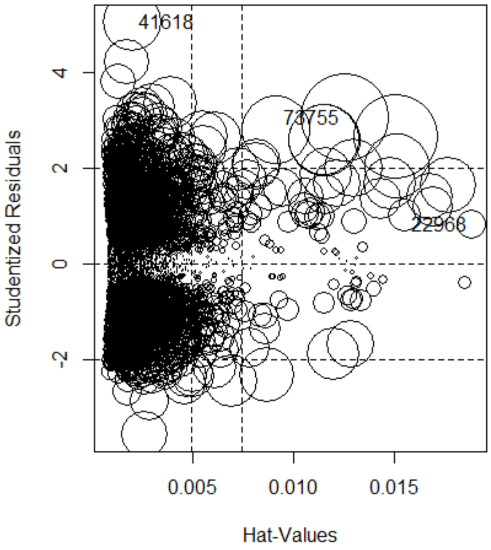

residuals
-1.0
-0.5
0.0
0.5
1.0
1.5
0.0
0.5
1.0
1.5
2.0
2.5
log10(slslyr + 1)

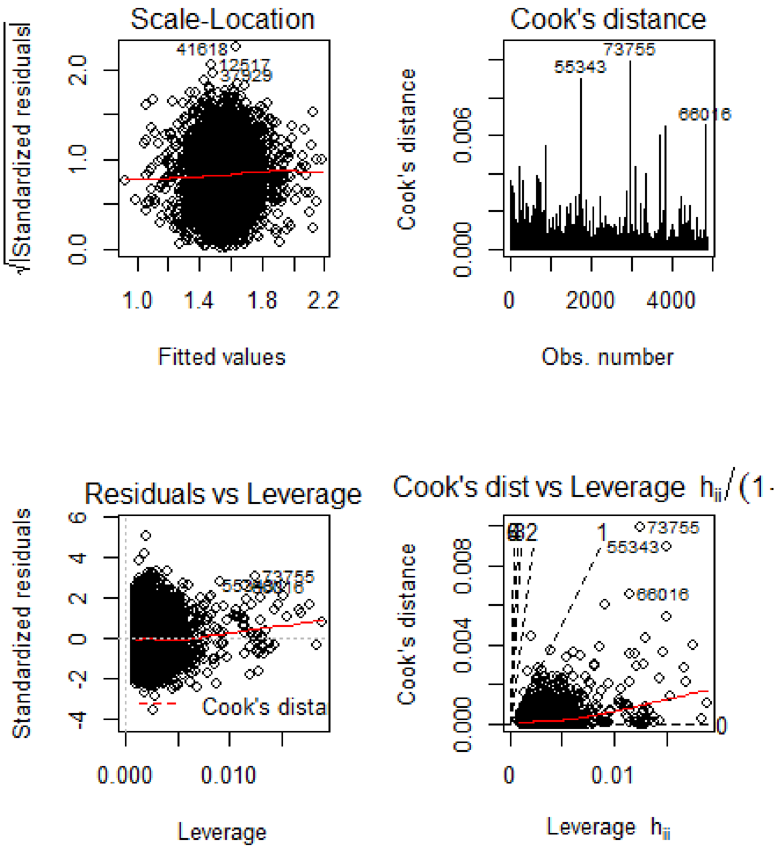

residuals
-1.0
-0.5
0.0
0.5
1.0
1.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
log10(sls2ago + 1)

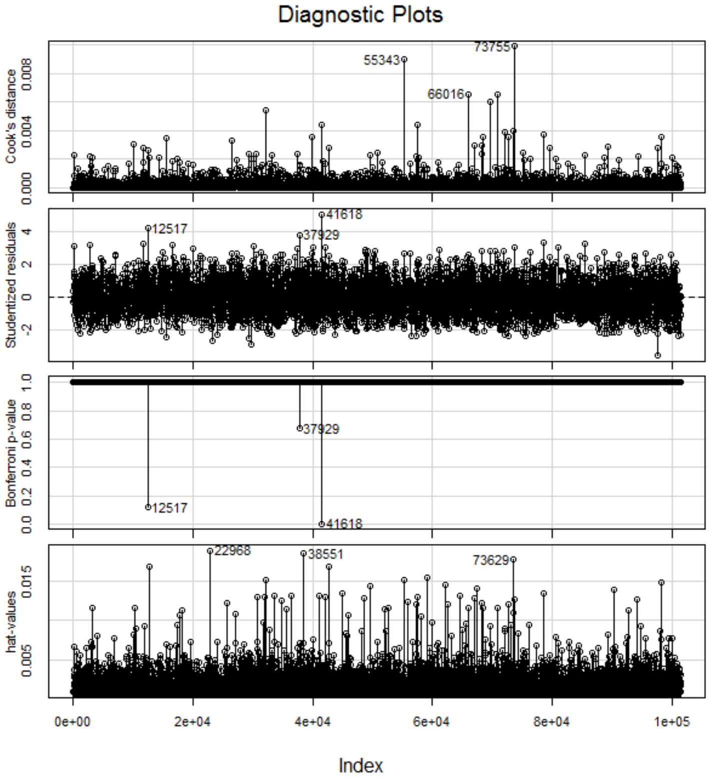

residuals
-1.0
-0.5
0.0
0.5
1.0
1.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
log10(sls3ago + 1)


residuals
-1.0
-0.5
0.0
0.5
1.0
1.5
0
1
2
3
4
5
log10(slstyr * slslyr + 1)


residuals
-1.0
-0.5
0.0
0.5
1.0
1.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
log10(sls3ago + 1)


41618
73755
22968
Studentized Residuals
4
2
0
-2
0.005
0.010
0.015
Hat-Values

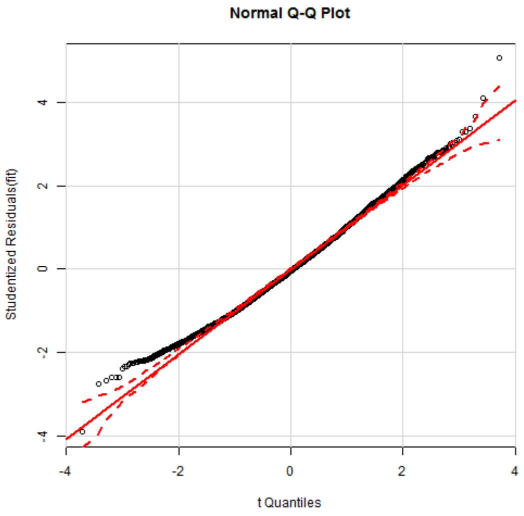

Scale-Location
√|Standardized residuals|
41618
12517
37929
0.0
1.0
2.0
1.0
1.4
1.8
2.2
Fitted values
Cook's distance
Cook's distance
73755
55343
66016
0.000
0.006
0
2000
4000
Obs. number
Residuals vs Leverage
Standardized residuals
73755
55343
66016
Cook's dista
-4
-2
0
2
4
6
0.000
0.010
Leverage
Cook's dist vs Leverage $h_{ii}/(1-$
Cook's distance
73755
55343
66016
2
1
0
0.000
0.004
0.008
0
0.01
Leverage h_{ii}

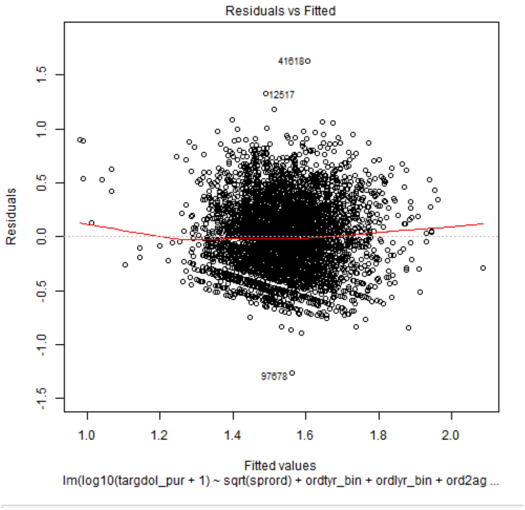

Diagnostic Plots
Cook's distance
73755
55343
66016
0.000
0.004
0.008
Studentized residuals
41618
12517
37929
-2
0
2
4
Bonferroni p-value
37929
12517
41618
0.0
0.2
0.4
0.6
0.8
1.0
hat-values
22968
38551
73629
0.005
0.015
0e+00
2e+04
4e+04
6e+04
8e+04
1e+05
Index

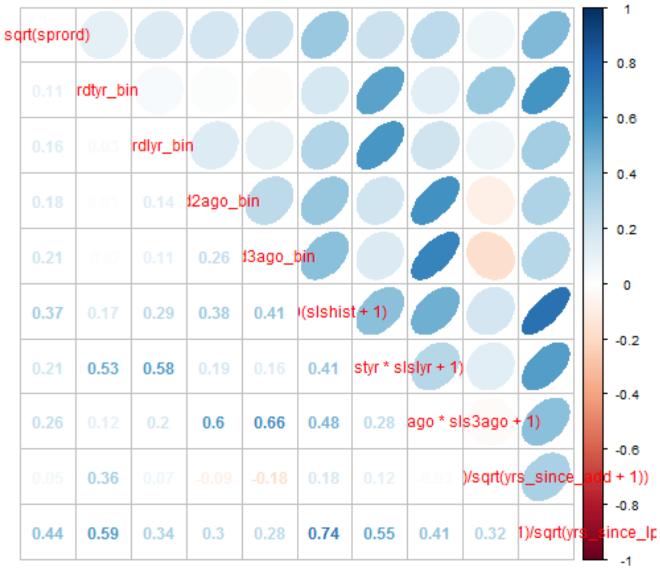
Variable	VIF
ordtyr_bin	11.82866
ordlyr_bin	12.83138
ord2ago_bin	14.04109
ord3ago_bin	13.63637
log10(slshist + 1)	2.385936
log10(slstyr + 1)	14.42153
log10(slslyr + 1)	14.50465
log10(sls2ago + 1)	14.76474
log10(sls3ago + 1)	14.27897
log10(slstyr * slslyr + 1)	3.140899
I(log10(slshist + 1)/sqrt(yrs_since_add + 1))	1.621298

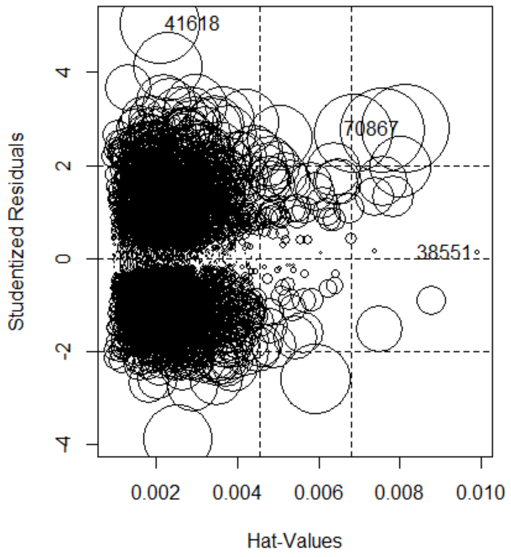
lm6

Variable	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	1.166916	0.025329	46.07074	0
sqrt(sprord)	-0.02285	0.007196	-3.17485	0.001509
ordtyr_bin	-0.15431	0.017468	-8.8334	1.40E-18
ordlyr_bin	-0.12468	0.013361	-9.3317	1.55E-20
ord2ago_bin	-0.07828	0.012815	-6.10824	1.09E-09
ord3ago_bin	-0.06717	0.014216	-4.72465	2.37E-06
log10(slshist + 1)	0.141267	0.018753	7.532933	5.89E-14
log10(slstyr * slslyr + 1)	0.047099	0.005922	7.952595	2.26E-15
log10(sls2ago * sls3ago + 1)	0.018158	0.006181	2.93753	0.003324
I(log10(slshist + 1)/sqrt(yrs_since_add + 1))	0.170937	0.018227	9.378458	1.00E-20
I(log10(slshist + 1)/sqrt(yrs_since_lp + 1))	0.069024	0.018017	3.831054	0.000129



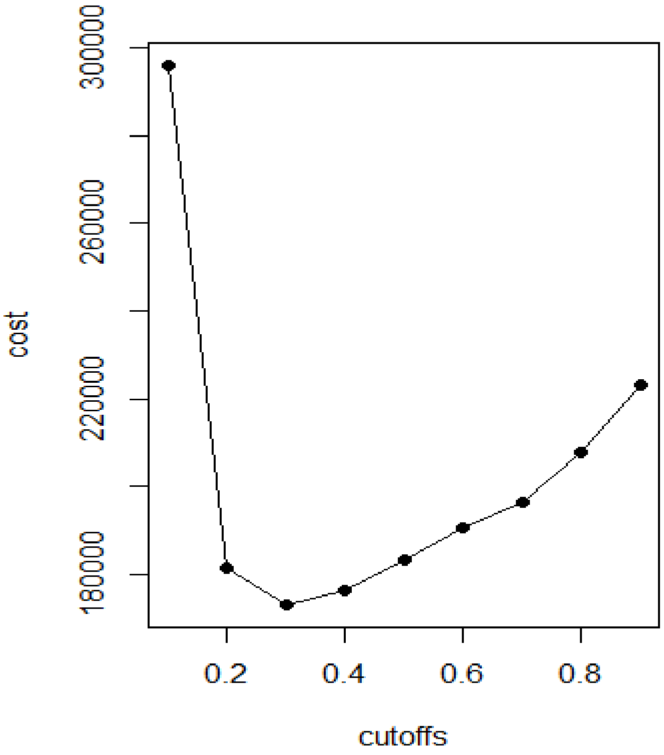






Variable	VIF
sqrt(sprord)	1.324338
ordtyr_bin	3.533825
ordlyr_bin	2.038148
ord2ago_bin	1.794188
ord3ago_bin	2.101263
log10(slshist + 1)	3.537125
log10(slstyr * slslyr + 1)	2.846507
log10(sls2ago * sls3ago + 1)	2.897314
I(log10(slshist + 1)/sqrt(yrs_since_add + 1))	1.348305
I(log10(slshist + 1)/sqrt(yrs_since_lp + 1))	5.701023

D. Model Validation


cost
cutoffs