Statistical Learning Assignment

MIT 801

Connor McDonald
u16040725

Introduction To Machine And Statistical Learning

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Department of Computer Science
University of Pretoria
18 June 2021

1 Model Discussion

In this assignment we used marketing data from a Portuguese bank to determine if a client would subscribe to a "term deposit" or not. One of the biggest problems faced, was that roughly 90% of the observations resulted in a 0 (no), meaning that a model that only predicts 0 (no) would achieve an accuracy of 90% which is not a true reflection of the models performance, nor is it a robust model. The details of these challenges and the results that followed will be discussed further in section 2 and 3.

The first statistical learning technique used to model the output variable was linear regression. Linear regression is not typically used for classification as it predicts a continuous output. To combat this problem, a threshold of 0.5 was used in conjunction with the modelled outputs. If a value fell below the threshold it was classified as a 0 or no, and if it fell above the threshold it was classified as a 1 or yes. However, with our dataset being so imbalanced and using a threshold midway between the two classes, we often find that the model is extra conservative when predicting 0's or no's, and thus results in an extremely high accuracy of predicting 0's. With that being said, the model performed poorly when predicting 1's since it was so heavily skewed towards 0's. By lowering the threshold to 0.165, more favourable results were observed, this is discussed in detail in section 3. An example of this is shown in figure 1 below. As you can see if we used a threshold of 0.5 in this model, almost half of the observations that recorded a "yes" would have been classified as a "no", however the accuracy would have still been extremely high as 90% of the observations were "no's" and our model would have predicted nearly all of them correctly.

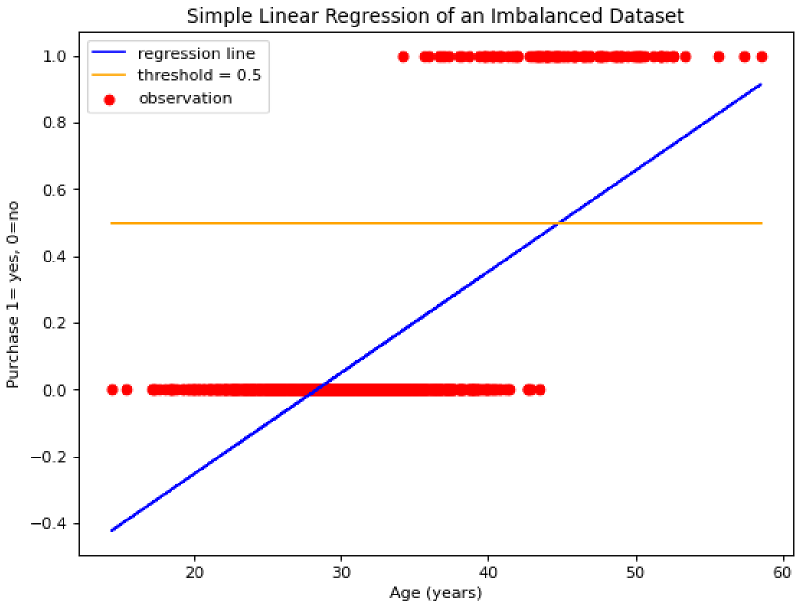

Figure 1

The second technique used was logistic regression. Logistic regression is used to model the probability of a dichotomous dependent variable such as (yes/no), (0/1), (pass/fail) etc. The underlying mathematical concept used in logistic regression is that of the logit function which is the natural logarithm of the odds of each outcome, where p is a probability. This function is shown below in equation 1.

$$\text{logit(p)} = \ln(\frac{p}{1-p}) = \beta_0 + \beta_1 X_1 +, ..., +\beta_n X_N \tag{1}$$

Much like the linear regression model, the logistic regression model also uses a threshold value to decide which class an observation belongs to. However since, logistic regression models the probability of an observation belonging to a specific class this threshold is often set to 0.5 by default in most Python libraries. This can cause problems with imbalanced datasets, and this was observed when trying to model the *insurance.xlsx* file, as the model was once again conservative when predicting "no's" which resulted in "yes's" being predicted poorly. This led to a misleading accuracy metric of 89% which seems good to the untrained eye, but what the model is actually doing is predicting "no" for almost every observation.

Fortunately the *Sci-Kit Learn* library allows for adjustment in the weightings of each class, and after setting this argument to "balanced", a slight decrease in accuracy was observed, however it resulted in a massive improvement in the prediction of "yes's" and ultimately created a more robust model.

2 Interpretation of Estimated Results

Before removing any of the insignificant data, a regression summary was constructed using the *statsmodel.api* library to get an idea of which variables could be dropped due to insignificance. These summaries can be seen in appendix A, for linear regression and appendix B, for logistic regression. Variable names are formatted as "D_n" due to a number of dummy variables being added in the encoding process. Finally, one dummy variable was removed from each set to prevent any multicollinearity which may have made it difficult to interpret regression coefficients.

When determining the significance of variables in the linear regression model, a significance level of 0.05 was used and a process of backwards elimination was followed. After each run of the model, the variable with the highest p-value was eliminated until all variables had p-values below the significance level.

A total of 13 columns were eliminated due to this process, and the remaining columns can be seen in appendix C, where D_n represents a dummy variable. The top 3 independent variables (ranked on absolute coefficient value) will be discussed below. For the linear regression summary the "day" column shows a positive correlation, this could indicate people being more likely to subscribe towards the end of the month when pay day is approaching. Dummy variable 23 also shows relatively high correlation, this dummy variable was derived from the "poutcome" column which represents the outcome of the previous marketing campaign. This could be because a persons previous experience with marketing campaigns from this bank will likely determine their attitude towards a new campaign. For example if they had a terrible experience with a previous campaign, there is a low probability that they will accept a new subscription in the following campaign. Lastly, dummy variable 8 shows a negative correlation, this dummy variable was derived from the education column. This may be because higher education and higher salaries often go hand in and, and may lead to a person being more likely to spend money on a subscription.

Similarly, a significance level of 0.05 and backwards elimination were used to determine which columns were statistically insignificant for the logistic regression model. Roughly the same columns were eliminated, however, with the addition of one more column for a total of 14 columns which were deemed insignificant. The remaining variables are shown in appendix D. Once again, the "day" column shows the highest positive correlation to our dependent variable in relation to the rest of the dependent variables. However, unlike linear regression, D12 which is derived from the marital column is the second most "influential" variable and shows a strong negative correlation, this may be due to married individuals having a combined income being more likely to subscribe. Lastly, D24 is our third most influential variable, this column is derived from the "poutcome" column, this column also had a relatively large influence on our linear regression model. Again this is most likely because of a persons attitude towards a marketing campaign is determined based on their previous experiences.

3 Assessment of Accuracy

Before making adjustments, both models appear to perform exceptionally well, achieving an accuracy metric around 90%. This is a rather misleading metric since the data set is highly imbalanced where one class makes up 90% of the data. This is why specificity and sensitivity of each model was assessed. In our linear regression model a specificity of 99.06% was achieved when using the default threshold of 0.5, this means that the model could correctly predict "no" 99.06% of the time. On the other end of the spectrum was sensitivity, which the linear regression model achieved a score of 20.69%. This implies that the model could only correctly predict a "yes" 20.69% of the time and thus we do not have a very robust model at the default threshold since it can only accurately predict one of our two classes.

To address this issue the model was tested at different thresholds to find a balance between specificity and sensitivity. It was observed that a threshold between 0.16 and 0.17 produced favourable results, with a specificity of 84.05%, a sensitivity of 83.82% and an overall accuracy of 84.02%. Using this threshold may have lowered the overall accuracy, but the "robustness" of our model is drastically increased as we can now correctly predict both "yes" and "no" for more that 80% of the observations. A summary of results is provided in Table 1.

Table 1: Summary of Linear Regression Results at Different Thresholds

Threshold	**Accuracy (%)**	**Specificity (%)**	**Sensitivity (%)**
0.5 (default)	90.80	99.06	20.69
0.2	86.65	87.96	77.45
0.17	84.73	84.86	83.55
0.165	**84.02**	**84.05**	**83.82**
0.16	83.41	83.27	84.62
0.10	72.97	70.50	93.90

In the first version of the logistic regression model, where no "model-tuning" was done, a high accuracy (91%) and specificity (97.87%) were observed whilst a low sensitivity (32.63%) was observed. This was once again caused by the class imbalance in the dataset. By setting the *class_weight* argument to "balanced" in our logistic regression function, a higher weight was assigned to "yes's" in our dataset. This will influence the classification of the classes during the training phase by penalizing misclassification of the minority class.

The new results followed the same pattern as with linear regression, where overall accuracy and specificity decreased to 84.75% and 84.92%, respectively, and sensitivity drastically increased to 83.23%. To validate these results, a 10-fold cross validation procedure was carried out, which resulted in an average accuracy of 83.98% with a low standard deviation of only 0.50%. The cross-validated Receiver Operating Characteristic (ROC) curves were also plotted (Figure 2) and maintained a very tight spread. The mean Area Under Curve (AUC) score was calculated as 0.91, which indicates a high degree of separability between classes.

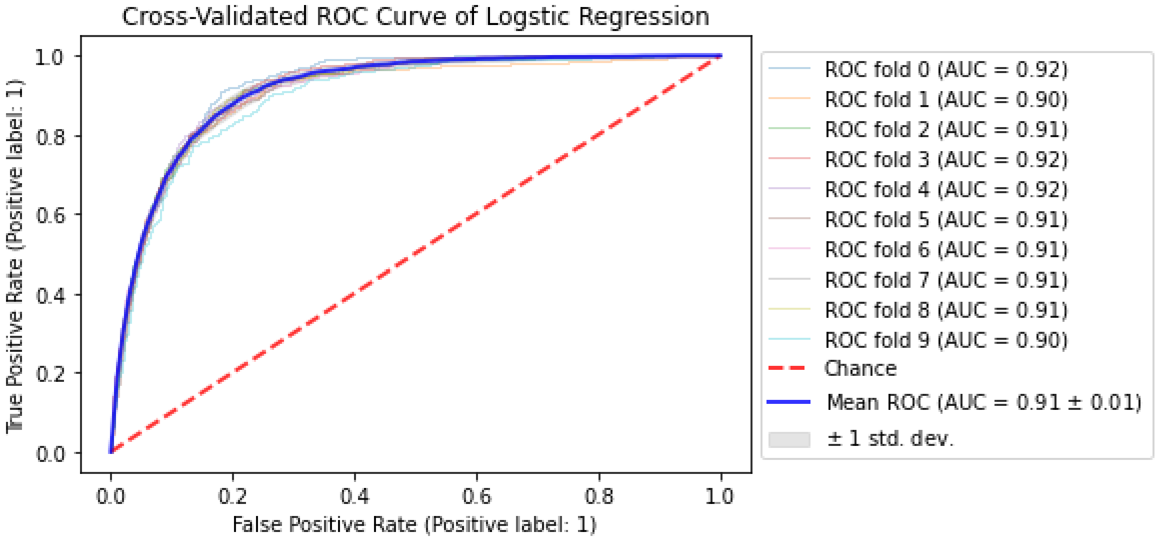

Figure 2

4 Model Predictions

please see the *results.xlsx* file attached

5 Model Selection

Both models models performed exceptionally well, achieving accuracy scores between 83% and 84%, however these metrics were achieved on the same training data, and if this section of training data contains a significantly different distribution of classes when compared to the rest of the dataset, our results may be negatively affected. To gain confidence in metrics observed, a 10-fold cross validation was performed on both models, and the results from this cross-validation are summarized in Table 2.

As discussed in Section 3 both models, exhibited similar specificity and sensitivity. Therefore the deciding factor in choosing which model would be better suited for predicting the given data was based on the cross-validated results. Logistic regression showed higher accuracy and lower standard deviation, furthermore, it produced a high average AUC score, seen in Figure 2 indicating that the model separates the classes effectively. Lastly, from a usability perspective, logistic regression is more commonly used for binary classification, and therefore has a bigger online support community. I found it much easier to find help when I got stuck creating the logistic regression model than when I was creating the linear regression model. Furthermore, most of the Python libraries used in classification cannot be used with linear regression, which is part of the reason I was unable to construct a cross-validated ROC curve for the linear regression model.

For the reasons stated above, **Logistic Regression** will be the best model for the given data.

Table 2: 10-fold Cross-validated Summary of Model Results

Model	**Accuracy (%)**	**Standard Deviation (%)**
Linear Regression	83.24	0.68
Logistic Regression	83.98	0.50

A

Dep. Variable:	y	**R-squared:**	0.302
Model:	OLS	**Adj. R-squared:**	0.301
Method:	Least Squares	**F-statistic:**	411.3
Date:	Thu, 17 Jun 2021	**Prob (F-statistic):**	0.00
Time:	16:30:56	**Log-Likelihood:**	-4134.6
No. Observations:	40000	**AIC:**	8355.
Df Residuals:	39957	**BIC:**	8725.
Df Model:	42		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
const	0.1168	0.001	86.976	0.000	0.114	0.119
D0	-0.0073	0.002	-3.448	0.001	-0.011	-0.003
D1	-0.0041	0.002	-2.674	0.007	-0.007	-0.001
D2	-0.0055	0.002	-3.593	0.000	-0.008	-0.002
D3	-0.0038	0.002	-1.638	0.101	-0.008	0.001
D4	0.0060	0.002	3.441	0.001	0.003	0.009
D5	-0.0042	0.002	-2.753	0.006	-0.007	-0.001
D6	-0.0035	0.002	-2.010	0.044	-0.007	-8.64e-05
D7	0.0112	0.002	7.403	0.000	0.008	0.014
D8	-0.0043	0.002	-2.195	0.028	-0.008	-0.000
D9	-0.0012	0.002	-0.799	0.424	-0.004	0.002
D10	-0.0015	0.001	-1.051	0.293	-0.004	0.001
D11	-0.0058	0.002	-2.695	0.007	-0.010	-0.002
D12	0.0040	0.002	1.748	0.080	-0.000	0.008
D13	0.0032	0.002	1.459	0.145	-0.001	0.007
D14	0.0112	0.002	4.517	0.000	0.006	0.016
D15	0.0018	0.002	1.158	0.247	-0.001	0.005
D16	-0.0002	0.001	-0.130	0.897	-0.003	0.002
D17	-0.0238	0.002	-14.670	0.000	-0.027	-0.021
D18	-0.0091	0.001	-6.566	0.000	-0.012	-0.006
D19	-0.0030	0.001	-2.144	0.032	-0.006	-0.000
D20	-0.0398	0.002	-18.969	0.000	-0.044	-0.036
D21	-0.0241	0.002	-10.210	0.000	-0.029	-0.020
D22	0.0086	0.001	6.134	0.000	0.006	0.011
D23	-0.0035	0.002	-1.832	0.067	-0.007	0.000
D24	-0.0185	0.002	-11.249	0.000	-0.022	-0.015
D25	-0.0252	0.002	-10.714	0.000	-0.030	-0.021
D26	0.0051	0.003	2.052	0.040	0.000	0.010
D27	0.0296	0.001	20.351	0.000	0.027	0.032
D28	-0.0135	0.003	-4.628	0.000	-0.019	-0.008

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
D29	-0.0209	0.002	-10.460	0.000	-0.025	-0.017
D30	0.0197	0.002	12.991	0.000	0.017	0.023
D31	0.0193	0.001	12.940	0.000	0.016	0.022
D32	0.0048	0.002	3.084	0.002	0.002	0.008
D33	0.0739	0.002	46.530	0.000	0.071	0.077
D34	-0.0062	0.003	-1.862	0.063	-0.013	0.000
age	0.0018	0.002	1.013	0.311	-0.002	0.005
balance	0.0021	0.001	1.544	0.123	-0.001	0.005
day	0.0070	0.002	4.522	0.000	0.004	0.010
duration	0.1216	0.001	89.784	0.000	0.119	0.124
campaign	-0.0047	0.001	-3.310	0.001	-0.007	-0.002
pdays	-0.0064	0.003	-2.198	0.028	-0.012	-0.001
previous	0.0019	0.002	1.226	0.220	-0.001	0.005

Omnibus:	11429.506	**Durbin-Watson:**	2.021
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	39055.937
Skew:	1.433	**Prob(JB):**	0.00
Kurtosis:	6.901	**Cond. No.**	6.50

B

Dep. Variable:	y	**No. Observations:**	40000
Model:	Logit	**Df Residuals:**	39957
Method:	MLE	**Df Model:**	42
Date:	Thu, 17 Jun 2021	**Pseudo R-squ.:**	0.3385
Time:	16:31:48	**Log-Likelihood:**	-9536.8
converged:	True	**LL-Null:**	-14416.

	coef	**std err**	**z**	**P> |z|**	**[0.025**	**0.975]**
const	-2.8423	0.027	-105.117	0.000	-2.895	-2.789
x1	-0.1303	0.032	-4.105	0.000	-0.193	-0.068
x2	-0.0614	0.024	-2.593	0.010	-0.108	-0.015
x3	-0.0867	0.024	-3.640	0.000	-0.133	-0.040
x4	-0.0704	0.032	-2.226	0.026	-0.132	-0.008
x5	0.0539	0.022	2.394	0.017	0.010	0.098
x6	-0.0616	0.022	-2.791	0.005	-0.105	-0.018
x7	-0.0657	0.026	-2.550	0.011	-0.116	-0.015
x8	0.0612	0.016	3.711	0.000	0.029	0.093
x9	-0.0762	0.027	-2.780	0.005	-0.130	-0.022
x10	-0.0335	0.020	-1.678	0.093	-0.073	0.006
x11	-0.0191	0.019	-0.984	0.325	-0.057	0.019
x12	-0.0862	0.031	-2.806	0.005	-0.146	-0.026
x13	0.0438	0.032	1.357	0.175	-0.019	0.107
x14	0.0886	0.035	2.567	0.010	0.021	0.156
x15	0.1775	0.037	4.853	0.000	0.106	0.249
x16	0.0450	0.022	2.047	0.041	0.002	0.088
x17	-0.0038	0.023	-0.164	0.870	-0.049	0.042
x18	-0.3280	0.023	-14.148	0.000	-0.373	-0.283
x19	-0.1608	0.023	-6.853	0.000	-0.207	-0.115
x20	-0.0418	0.020	-2.123	0.034	-0.080	-0.003
x21	-0.7339	0.035	-20.810	0.000	-0.803	-0.665
x22	-0.2315	0.029	-8.008	0.000	-0.288	-0.175
x23	0.0440	0.013	3.370	0.001	0.018	0.070
x24	-0.0340	0.022	-1.522	0.128	-0.078	0.010
x25	-0.2059	0.022	-9.233	0.000	-0.250	-0.162
x26	-0.2846	0.030	-9.627	0.000	-0.343	-0.227
x27	0.1501	0.032	4.640	0.000	0.087	0.213
x28	0.1688	0.013	12.899	0.000	0.143	0.194
x29	-0.1671	0.035	-4.739	0.000	-0.236	-0.098

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
x30	-0.2406	0.026	-9.432	0.000	-0.291	-0.191
x31	0.1133	0.015	7.751	0.000	0.085	0.142
x32	0.1029	0.014	7.205	0.000	0.075	0.131
x33	0.0431	0.019	2.288	0.022	0.006	0.080
x34	0.4075	0.016	26.010	0.000	0.377	0.438
x35	-0.0354	0.038	-0.927	0.354	-0.110	0.039
x36	0.0008	0.025	0.033	0.973	-0.048	0.050
x37	0.0226	0.017	1.319	0.187	-0.011	0.056
x38	0.0682	0.022	3.089	0.002	0.025	0.111
x39	1.0829	0.018	61.181	0.000	1.048	1.118
x40	-0.2783	0.033	-8.341	0.000	-0.344	-0.213
x41	-0.0107	0.033	-0.327	0.743	-0.075	0.053
x42	0.0188	0.015	1.271	0.204	-0.010	0.048

C

Dep. Variable:	y	**R-squared:**	0.300
Model:	OLS	**Adj. R-squared:**	0.300
Method:	Least Squares	**F-statistic:**	591.2
Date:	Thu, 17 Jun 2021	**Prob (F-statistic):**	0.00
Time:	19:25:18	**Log-Likelihood:**	-4181.2
No. Observations:	40000	**AIC:**	8422.
Df Residuals:	39970	**BIC:**	8680.
Df Model:	29		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
const	0.1168	0.001	86.889	0.000	0.114	0.119
D0	-0.0058	0.001	-3.884	0.000	-0.009	-0.003
D1	-0.0028	0.001	-2.069	0.039	-0.006	-0.000
D2	-0.0049	0.001	-3.592	0.000	-0.008	-0.002
D3	0.0073	0.001	5.224	0.000	0.005	0.010
D4	-0.0031	0.001	-2.281	0.023	-0.006	-0.000
D5	-0.0060	0.002	-2.816	0.005	-0.010	-0.002
D6	0.0058	0.002	2.710	0.007	0.002	0.010
D7	0.0074	0.001	5.078	0.000	0.005	0.010
D8	-0.0249	0.002	-15.592	0.000	-0.028	-0.022
D9	-0.0096	0.001	-7.016	0.000	-0.012	-0.007
D10	-0.0398	0.002	-19.113	0.000	-0.044	-0.036
D11	-0.0249	0.002	-10.574	0.000	-0.029	-0.020
D12	0.0087	0.001	6.230	0.000	0.006	0.011
D13	-0.0037	0.002	-1.969	0.049	-0.007	-1.66e-05
D14	-0.0189	0.002	-11.467	0.000	-0.022	-0.016
D15	-0.0260	0.002	-11.107	0.000	-0.031	-0.021
D16	0.0049	0.003	1.962	0.050	5.4e-06	0.010
D17	0.0300	0.001	20.575	0.000	0.027	0.033
D18	-0.0138	0.003	-4.716	0.000	-0.020	-0.008
D19	-0.0212	0.002	-10.640	0.000	-0.025	-0.017
D20	0.0197	0.002	13.038	0.000	0.017	0.023
D21	0.0197	0.001	13.208	0.000	0.017	0.023
D22	0.0051	0.002	3.308	0.001	0.002	0.008
D23	0.0742	0.002	46.726	0.000	0.071	0.077
D24	-0.0080	0.003	-2.474	0.013	-0.014	-0.002
balance	0.0070	0.002	4.469	0.000	0.004	0.010
day	0.1216	0.001	89.742	0.000	0.119	0.124
duration	-0.0050	0.001	-3.501	0.000	-0.008	-0.002
campaign	-0.0071	0.003	-2.447	0.014	-0.013	-0.001

D

Dep. Variable:	y	**No. Observations:**	40000
Model:	Logit	**Df Residuals:**	39971
Method:	MLE	**Df Model:**	28
Date:	Thu, 17 Jun 2021	**Pseudo R-squ.:**	0.3372
Time:	20:41:08	**Log-Likelihood:**	-9554.3
converged:	True	**LL-Null:**	-14416.

	coef	**std err**	**z**	**P> |z|**	**[0.025**	**0.975]**
const	-2.8361	0.027	-105.314	0.000	-2.889	-2.783
D0	-0.1537	0.027	-5.761	0.000	-0.206	-0.101
D1	-0.0676	0.023	-3.006	0.003	-0.112	-0.024
D2	-0.1036	0.022	-4.635	0.000	-0.147	-0.060
D3	-0.0769	0.027	-2.873	0.004	-0.129	-0.024
D4	-0.0648	0.021	-3.119	0.002	-0.105	-0.024
D5	-0.0690	0.023	-2.945	0.003	-0.115	-0.023
D6	0.0638	0.015	4.293	0.000	0.035	0.093
D7	-0.0775	0.023	-3.323	0.001	-0.123	-0.032
D8	-0.1159	0.020	-5.929	0.000	-0.154	-0.078
D9	0.1084	0.024	4.529	0.000	0.061	0.155
D10	-0.3212	0.022	-14.423	0.000	-0.365	-0.278
D11	-0.1600	0.023	-6.869	0.000	-0.206	-0.114
D12	-0.7428	0.034	-21.627	0.000	-0.810	-0.676
D13	-0.2075	0.024	-8.507	0.000	-0.255	-0.160
D14	0.0505	0.013	3.987	0.000	0.026	0.075
D15	-0.1998	0.022	-9.234	0.000	-0.242	-0.157
D16	-0.2712	0.026	-10.507	0.000	-0.322	-0.221
D17	0.1749	0.028	6.158	0.000	0.119	0.231
D18	0.1774	0.012	14.444	0.000	0.153	0.201
D19	-0.1372	0.030	-4.524	0.000	-0.197	-0.078
D20	-0.2243	0.023	-9.764	0.000	-0.269	-0.179
D21	0.1231	0.014	9.038	0.000	0.096	0.150
D22	0.1137	0.013	8.546	0.000	0.088	0.140
D23	0.0593	0.016	3.624	0.000	0.027	0.091
D24	0.4235	0.012	33.916	0.000	0.399	0.448
balance	0.0800	0.021	3.839	0.000	0.039	0.121
day	1.0836	0.018	61.480	0.000	1.049	1.118
duration	-0.2938	0.033	-8.837	0.000	-0.359	-0.229