Loan Approval

Trillyon Earl, Owen Jefcoat

2025-03-01

Abstract

Using a 45 000 observeration Kaggle Loan Approval Classification data set with 14 applicant and loan variables, we set out to build a model that classifies applications as accepted (1) or rejected (0). After standardising monetary variables and splitting the data 80 : 20 into training and test sets, we fitted three families of logistic models: (i) an ordinary full model (and its null counterpart), (ii) Firth's bias-reduced logistic regression to remedy the perfect / quasi-complete separation produced by the "previous loan defaults" factor, and (iii) ridge-penalised logistic regression with λ chosen by 10-fold cross-validation to mitigate multicollinearity. Problematic observations were diagnosed with hat-values, studentised residuals and Cook's D; 504 outliers (all influential) were removed before refitting. On the held-out test data, the ridge model achieved the lowest misclassification rate (10.5 %) and an MSPE of 0.0768, edging out both the full ordinary model (10.6%) and the optimised Firth variants (10.6%). Thus, while Firth regression stabilised coefficients under separation and stepwise AIC removed uneeded complexity, continuous shrinkage via ridge offered the best trade-off between bias and variance, making it the preferred tool for loan-approval screening in this study. The predictor variables for this data set are:

Predictors	Type
Age	Float
Gender	Categorical
Education	Categorical
Income	Float
Employment Years	Integer
Home Ownership	Categorical
Loan Amount	Float
Loan Intent	Categorical
Loan Interest Rate	Float
Loan Percent Income	Float
Credit History Length	Float
Credit Score	Integer
Previous Loans	Categorical
Loan Status	Integer

Objective

The objective of this analysis is to accurately predict whether or not a person given some information about them would be approved for a loan. A value of 0 indicates rejecton and a values of 1 indicates acceptance for a loan in loan status.

Data Preprocessing

Aside from converting categorical variables to factors, the only preprocessing we performed was scaling the **Loan Amount** and **Income** variables, following feedback from Dr. Gong during our presentation of this analysis. This step is sensible because those two variables have magnitudes that differ substantially from the others in the dataset. We then split the data into an $80-20$ train-test partition, using 80% for training and 20% for testing.

Methods

Full Ordinary Logistic Model

This is the fitted model of all predictors. The results of the measurements we got from this model are below. We expected this model to do relatively well as our variables seems to have strong correlation with either approval or rejection already so there is not much to filter or search for.

Null Ordinary Logistic Model

This is the model of only the intercept. We expected this model to do relatively poor as the intercept alone is too little information to make accurate predictions and the results of the measurements reflected this hypothesis.

Null Model vs. Full Model

We determined that the null model is inadequate not only from its poor performance metrics but also from the drop-in-deviance test. The χ^2 ANOVA produced a very low p-value, indicating that at least one predictor variable is essential for the model.

Model	AIC	BIC	MSPE	Error
Full	15898.15	16093.45	0.07304491	10.61%
Null	38138.34	38146.83	0.1729012	22.23%

Table 1: Full vs. Null model statistics for Loan Approval

Problematic Sample Points

Leverage

We recieved around 5954 out of the 36000 oberservations that are considered high leverage. An oberservation is considered high leverage in our analysis if the leverage is greater than $2(k+1)/n$, where k is 14 and n is 36000.

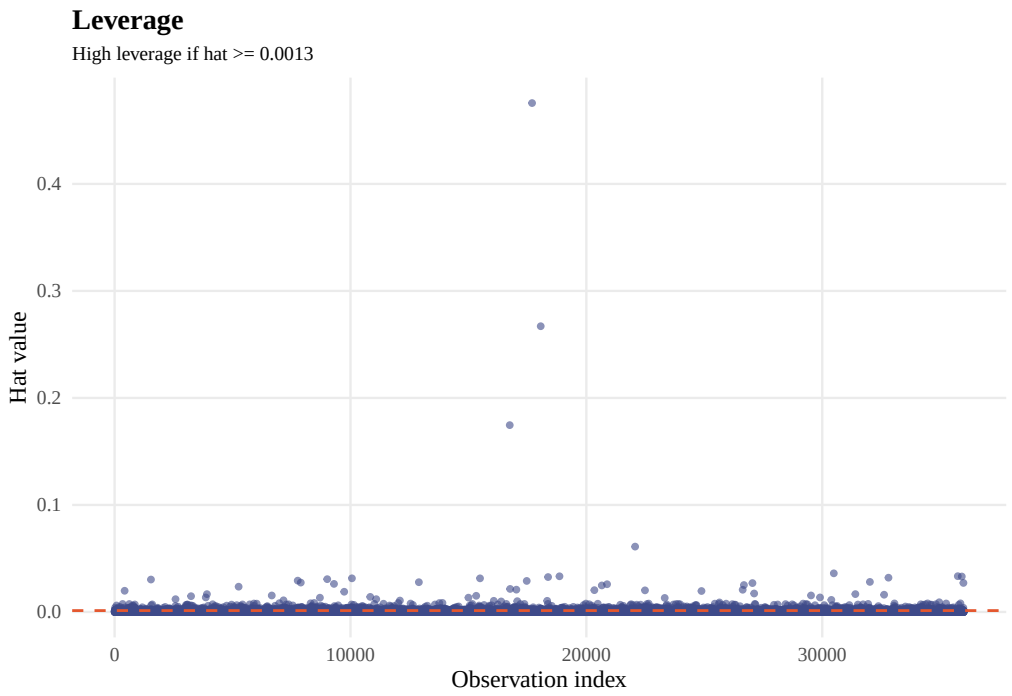


Figure 1: Leverage values with dashed cutoff.

Outliers

We found 504 outliers within this data. A oberservation was determined an outlier if its the absolute value of its studentized residual was greater than or equals 1.97

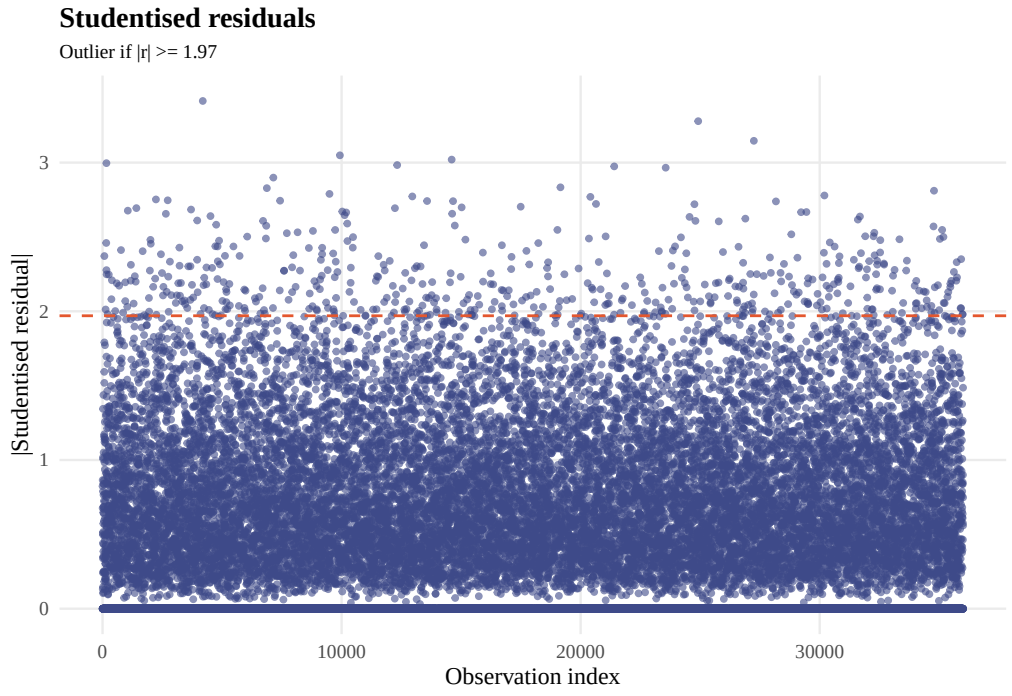


Figure 2: Absolute value of studentized residuals with dashed cutoff.

Cook's Distance & Influential Points

When using a cutoff value of 1 for our Cook's Distance we recieved no influential points and after some research we switched this cutoff to $\frac{4}{n}$ as used in many papers when dealing with relatively large datasets. After loosening the cutoff, we found 2478 influential points as shown in Figure 3.

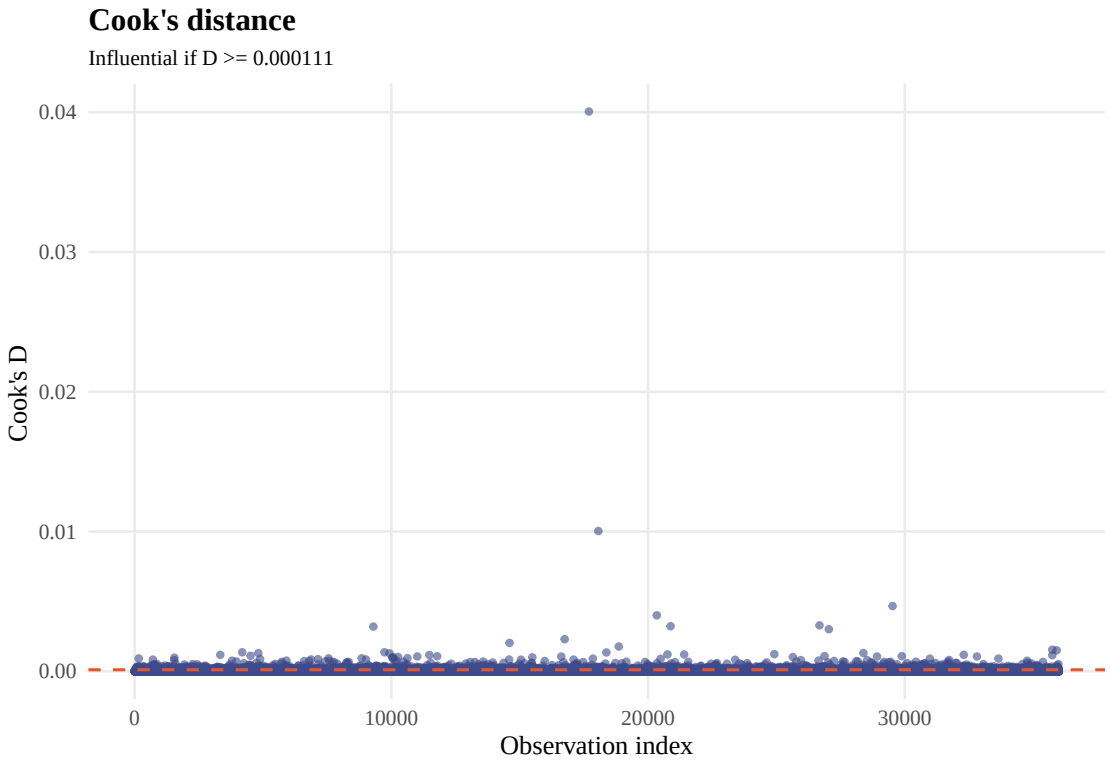


Figure 3: Cook's Distance values with dashed cutoff.

Problematic Points

We discovered that every outlier was also an influential point, so we removed all outliers from the model. Next, we rebuilt the full model with the updated training set and evaluated its performance. This revision caused only a slight increase in the error rate and MSPE, but it produced a substantial decrease in AIC and BIC.

Model	AIC	BIC	MSPE	Error
Full (Removed Outliers)	13010.07	13205.05	0.07450024	10.63%

Table 2: Full model with no outliers statistics for Loan Approval

Perfect and Quasi Complete Seperation

As we were analyzing the output of our model we encountered the warning message "glm.fit: fitted probabilities numerically 0 or 1 occurred". After researching we found out that this is because we have Perfect or Quasi-Complete Seperation. This means that one or more of our predictor varibles almost perfectly predicts our outcome variable. From anaylzing the summary of the full model we can spot multiple problem covariates. This is a problem because it can lead to overfitting, and it pushes the maximum likelihood and standard error towards infinity.

Number of Previous Loan Defaults On File We can see that this covariate's estimate has a large magnitude around 20. The cause of this could be from perfect and quasi complete seperation or multicollinearity.

Combatting Perfect and Quesi-Complete Seperation

From our research a useful way to combat Quesi-Complete Seperation is to merge or drop factor levels with too few observations ($< 0.5\%$ of data size) or drop levels that are heavily sided towards one outcome. The reason for this is that if these factor levels with few observations and heavily lean towards one side of the binary outcome then when building a model it will see this almost perfect predictor for the outcome variable and thus give us an esitmate with a very large magnitude for that factor level. Factor levels with too few observations also may have a wide confidence interval. Thus we collpase these factor levels into a single factor level that accounts for a larger group of observations. Below is a count of how many times each factor level occurs in the dataset of 36000 observations and their corresponding outcome.

Status	PERSONAL	MEDICAL	EDUCATION	VENTURE	HOME	DEBT
0	4 835	4 889	6 041	5 316	2 806	3 965
1	1 083	1 862	1 138	864	987	1 710

Table 3: Loan intent by loan status

Status	High School	Associate	Bachelor	Master	Doctorate
0	7 404	7 452	8 301	4 316	379
1	2 064	2 048	2 272	1 157	103

Table 4: Education level by loan status

Status	RENT	OWN	MORTGAGE	OTHER
0	12 574	2 172	13 045	61
1	5 977	157	1 480	30

Table 5: Home-ownership status by loan status

Status	Yes	No
0	18 259	9 593
1	0	7 644

Table 6: Previous loan defaults on file by loan status

We see that we actually have complete seperation. Every observation that has had a previous loan default has been rejected for a loan. We dont want to just remove this level because it still contains very important information thus we dedcided to combat this by using firth logistic regression as recommended in the sources provided.

Firth's Bias Reduced Logistic Regression

Firth logistic regression augments the ordinary log-likelihood with the **Firth penalty**

$$L_{\text{Firth}} = L_{\text{Logit}} - \frac{1}{2}\,\text{Penalty}$$

where the penalty term is made to prevent the log-likelihood from becoming infinite in cases of seperation.

Formula

$$L_{\text{Firth}} = L_{\text{Logit}} \; - \; \frac{1}{2} \log\left|I(\boldsymbol{\beta})\right|,$$

where $I(\boldsymbol{\beta})$ is the observed Fisher–information matrix.

As shown in the appendix (M.IV), there are still estimates with large magnitudes however the standard error for these estimates have significantly decreased which now informs us that it has a strong effect on our data. For example the p value of previous loan defaults was large because of it's standard error, however now the p-value is near zero. Firth Regression, however, does not handle multi-collinearity like ridge regression thus we can still test for multi-collinearity which we will do utilizing VIF.

Model	AIC	BIC	MSPE	Error
Firth Model	13011.3	13206.28	0.07447477	10.63%

Table 7: Firth model statistics for Loan Approval

Multi-Collinearity (VIF)

The two variables with VIFs greater than 10 were **age** and **employment experience**. After comparing the deviance of both, removing employment experience resulted in a lower deviance thus we chose to remove that variable. After that removal there were no more VIFs > 10. When removing the problem covariate with didn't see any noticeable change in our measurements in accuracy.

Stepwise AIC Variable Selection on Firth Logistic Regression

We also applied a stepwise AIC variable selection to our original firth model in multiple directions which resulted in the measurements below. Again there was no noticeable change in accuracy besides our error percentage decreasing from 10.63% to 10.60% in our backward and both model.

Forward Forward Stepwise AIC resulted in the original Firth model, no variabels were removed.

Backward Backward Stepwise AIC resulted in the removal of gender, education, and income variables.

Both Similar to Backward Stepwise AIC, Alternating Stepwise AIC resulted in the same removals as Backward Stepwise AIC.

Ridge Logistic Regression

The goal of ridge regression is to exchange increased bias for decreased variance. Ridge regression takes the ordinary least squares method and adds a penalty to it which minimizes the sum of squared residuals (RSS) and the penalty, which is the sum of coefficients (B_j) squared times lambda, where lambda determines the severity of the penalty. This results in a smaller coefficients as we can in the below R code. In the case of Logistic Regression, Ridge Regression minimizes the sum of the likelihoods plus the sum of coefficients squared times lambda. But how do we choose a value of lambda that doesnt minimize the coefficients too much but just enough? We use Cross-Validation, specifically 10-fold cross validation to determine which lambda results in the lowest SSE. Another useful thing about Ridge Regression is that we do not need to do any variable selection as Ridge Regression already minimizes coefficients that would be removed from various variable selection methods. The coefficients found from our Ridge Regression can be viewed in the appendix.

Why Not LASSO

We decided to use Ridge regression instead of LASSO regression soley because we wanted to keep all of our coefficients even if they aren't impactful as they still provide some sense of inference.

Model	MSPE	Error
Ridge Model	0.0767909	10.51%

Table 8: Ridge model statistics for Loan Approval

Analysis & Results

The different regression methods we used were Ordinary Logistic Regression, Firth Logistic Regression, and Ridge Regression. We applied different variable selection techniques such as Stepwise AIC and VIF.

Ordinary Logistic Regression

- The logistic full model (loan_status ~ all variables) with the original training data.
- The logistic null model (loan_status ~ intercept) with the original training data and Drop-in Deviance test.
- The logistic model (loan_status ~ all variables) with the omitted outliers training data.

Firth's Logistic Regression

- The firth logistic model (loan_status ~ all variables - person_emp_exp) with the omitted outliers training data
- The firth logistic model (loan_status ~ all variables - person_emp_exp) with the omitted outliers training data and variable selection using VIF and StepwiseAIC

Ridge Logistic Regression

- The Ridge logistic model (loan_status ~ all variables) with the ommitted outliers and with 10-fold cross-validation.

Results

To evaluate predictive performance we applied each fitted model to the 20% test data and recorded the mean-squared prediction error (MSPE) and the classification error rate at a 0.5 cut-off.

Model (training setup)	MSPE	Prediction Error Rate
Ordinary Logistic Regression		
Full model, original data	0.07304491	0.1061111
Null model, Drop-in-Deviance test	0.1729012	0.2223333
Full model, outliers removed	0.07450024	0.1063333
Firth's Logistic Regression		
Firth, outliers removed	0.07447477	0.1063333
Firth (− person_emp_exp), outliers removed	0.0745102	0.1066667
Firth + VIF + Stepwise AIC	0.07443141	0.106
Ridge Logistic Regression		
Ridge (10-fold CV, $\lambda_{\min}$), outliers removed	0.0767909	0.1051111

Conclusion

Best-performing model

Based off of prediction error rate, which is the most important to us for this analysis, ridge regression produced the lowest error rate and thus was our best performing model.

Handling separation

During the study we encountered both perfect and quasi-complete separation—most prominently for the "previous loan defaults" variable. Ordinary logistic regression produced huge, unstable coefficients and warnings. Firth's bias-reduced logistic regression resolved this by adding the a penalty, giving us more interpretable estimates.

Practical insight

Separation and multicollinearity require different remedies. Firth regression fixes the former, while ridge regression fixes the latter. In our loan-approval data, we can assume multicollinearity was the stronger driver of error, thus ridge regression emerged as the overall winner.

Sources

- https://stats.oarc.ucla.edu/other/mult-pkg/faq/general/faqwhat-is-complete-or-quasi-complete-separation-in-logistic-regression-and-what-are-some-strategies-to-deal-with-the-issue/
- https://medium.datadriveninvestor.com/firths-logistic-regression-classification-with-datasets-that-are-small-imbalanced-or-separated-49d7782a13f1

Appendix

Models

M.I Full

```
full_model <- glm(loan_status ~ .,
  data = train,
  family = binomial(link = "logit")
)
summary(full_model)$coefficients
```

```
##                                    Estimate   Std. Error     z value      Pr(>|z|)
## (Intercept)                    -21.758111451 1.146103e+02  -0.1898443  8.494311e-01
## person_age                       0.030423954 1.222692e-02   2.4882753  1.283643e-02
## person_gendermale                0.024334719 3.970059e-02   0.6129562  5.399053e-01
## person_education.L              -0.039743530 1.075995e-01  -0.3693655  7.118553e-01
## person_education.Q              -0.027970014 9.241857e-02  -0.3026450  7.621605e-01
## person_education.C              -0.063123340 6.583484e-02  -0.9588136  3.376527e-01
## person_education^4              -0.055860860 4.394815e-02  -1.2710630  2.037062e-01
## person_income                    0.035459150 1.747334e-02   2.0293281  4.242489e-02
## person_emp_exp                  -0.022685160 1.088906e-02  -2.0832985  3.722403e-02
## person_home_ownershipOWN        -2.130840264 1.071923e-01 -19.8786661  6.226728e-88
## person_home_ownershipMORTGAGE   -0.732437127 4.494166e-02 -16.2975100  1.027953e-59
## person_home_ownershipOTHER      -0.388868356 3.568798e-01  -1.0896341  2.758743e-01
## loan_amnt                       -0.638481146 2.780679e-02 -22.9613408 1.135183e-116
## loan_intentMEDICAL               0.479978344 6.519124e-02   7.3626202  1.803350e-13
```

```
## loan_intentEDUCATION              -0.128676204 6.729748e-02  -1.9120509  5.586967e-02
## loan_intentVENTURE                -0.474036914 7.253698e-02  -6.5351069  6.356411e-11
## loan_intentHOMEIMPROVEMENT         0.775404590 7.563178e-02  10.2523649  1.154828e-24
## loan_intentDEBTCONSOLIDATION       0.794361144 6.737327e-02  11.7904493  4.372024e-32
## loan_int_rate                      0.337524154 7.378969e-03  45.7413733  0.000000e+00
## loan_percent_income               15.890251428 3.441044e-01  46.1785814  0.000000e+00
## cb_person_cred_hist_length        -0.011379714 1.010452e-02  -1.1262007  2.600806e-01
## credit_score                      -0.009073938 4.587094e-04 -19.7814528  4.301182e-87
## previous_loan_defaults_on_fileNo  20.383743689 1.146096e+02   0.1778537  8.588378e-01
```

M.II Null

```
null_model <- glm(
  loan_status ~ 1,
  data   = train,
  family = binomial(link = "logit")
)
summary(null_model)$coefficients
```

```
##              Estimate Std. Error  z value Pr(>|z|)
## (Intercept) -1.252924 0.01267787 -98.8276        0
```

M.III Full (No-Outlier)

```
train_no_outlier <- train[-outliers, ]
no_outlier_model <- glm(loan_status ~ .,
  data = train_no_outlier,
  family = binomial(link = "logit")
)
summary(no_outlier_model)$coefficients
```

```
##                                    Estimate   Std. Error     z value      Pr(>|z|)
## (Intercept)                     -23.59527039 1.082451e+02  -0.2179801  8.274446e-01
## person_age                        0.05779288 1.368629e-02   4.2226831  2.414112e-05
## person_gendermale                 0.05046379 4.432191e-02   1.1385744  2.548807e-01
## person_education.L               -0.09887846 1.222630e-01  -0.8087356  4.186672e-01
## person_education.Q               -0.08258983 1.049970e-01  -0.7865919  4.315208e-01
## person_education.C               -0.11336240 7.440740e-02  -1.5235366  1.276245e-01
## person_education^4               -0.07267642 4.925117e-02  -1.4756282  1.400437e-01
## person_income                     0.02754032 2.656103e-02   1.0368697  2.997966e-01
## person_emp_exp                   -0.04280294 1.218940e-02  -3.5114893  4.456034e-04
## person_home_ownershipOWN         -2.78159177 1.253855e-01 -22.1843109 4.868338e-109
## person_home_ownershipMORTGAGE    -0.92224542 5.076969e-02 -18.1652760  9.721684e-74
## person_home_ownershipOTHER       -0.66817937 4.015350e-01  -1.6640628  9.609989e-02
## loan_amnt                        -0.85695622 3.327371e-02 -25.7547569 2.851257e-146
## loan_intentMEDICAL                0.67979997 7.320739e-02   9.2859476  1.602745e-20
## loan_intentEDUCATION             -0.14958946 7.571544e-02  -1.9756798  4.819105e-02
## loan_intentVENTURE               -0.46730487 8.142677e-02  -5.7389585  9.526057e-09
## loan_intentHOMEIMPROVEMENT        1.10420995 8.503417e-02  12.9854846  1.478991e-38
## loan_intentDEBTCONSOLIDATION      1.08247022 7.585798e-02  14.2696951  3.380532e-46
## loan_int_rate                     0.45805178 9.111818e-03  50.2700761  0.000000e+00
## loan_percent_income              21.31205102 4.369673e-01  48.7726495  0.000000e+00
## cb_person_cred_hist_length       -0.02553008 1.130353e-02  -2.2585945  2.390862e-02
## credit_score                     -0.01179624 5.201396e-04 -22.6789862 7.223422e-114
```

```
## previous_loan_defaults_on_fileNo  20.96196544 1.082441e+02   0.1936545  8.464464e-01
```

M.IV Firth

```
firth_model <- glm(
  formula = loan_status ~ ., data = train_no_outlier, family = "binomial",
  na.action = na.fail, method = "brglmFit"
)
summary(firth_model)$coefficients
```

```
##                                    Estimate   Std. Error     z value      Pr(>|z|)
## (Intercept)                     -13.92632226 1.5073303500  -9.2390645  2.486618e-20
## person_age                        0.05787478 0.0136680800   4.2343022  2.292625e-05
## person_gendermale                 0.05040087 0.0442814136   1.1381947  2.550392e-01
## person_education.L               -0.09829245 0.1221213833  -0.8048750  4.208918e-01
## person_education.Q               -0.08193838 0.1048774749  -0.7812772  4.346395e-01
## person_education.C               -0.11284000 0.0743284335  -1.5181269  1.289824e-01
## person_education^4               -0.07256562 0.0492045511  -1.4747746  1.402731e-01
## person_income                     0.03807323 0.0206319034   1.8453572  6.498560e-02
## person_emp_exp                   -0.04261261 0.0121771529  -3.4993904  4.663233e-04
## person_home_ownershipOWN         -2.77554249 0.1252055374 -22.1678893 7.012241e-109
## person_home_ownershipMORTGAGE    -0.92149684 0.0507039227 -18.1740739  8.281488e-74
## person_home_ownershipOTHER       -0.67597949 0.4009028098  -1.6861431  9.176825e-02
## loan_amnt                        -0.86029549 0.0322396701 -26.6843765 7.146450e-157
## loan_intentMEDICAL                0.67851609 0.0731406814   9.2768632  1.745414e-20
## loan_intentEDUCATION             -0.14952569 0.0756484704  -1.9765858  4.808847e-02
## loan_intentVENTURE               -0.46653944 0.0813436523  -5.7354129  9.727505e-09
## loan_intentHOMEIMPROVEMENT        1.10166435 0.0849473974  12.9687828  1.839278e-38
## loan_intentDEBTCONSOLIDATION      1.08005942 0.0757837324  14.2518636  4.364760e-46
## loan_int_rate                     0.45720807 0.0090958279  50.2656908  0.000000e+00
## loan_percent_income              21.32468844 0.4270426288  49.9357371  0.000000e+00
## cb_person_cred_hist_length       -0.02591025 0.0112694080  -2.2991666  2.149548e-02
## credit_score                     -0.01177072 0.0005194372 -22.6605313 1.098475e-113
## previous_loan_defaults_on_fileNo 11.28646992 1.4389093236   7.8437673  4.372269e-15
```

M.V Ridge

```
cv_ridge <- cv.glmnet(
  x_train, y_train,
  family = "binomial",
  alpha = 0,
  nfolds = 10,
  type.measure = "class"
)

lambda_min <- cv_ridge$lambda.min # lambda that minimises CV error
lambda_1se <- cv_ridge$lambda.1se
lambda_min
```

```
## [1] 0.02216404
```

```
ridge_model <- glmnet(
  x_train, y_train,
  family = "binomial",
  alpha = 0,
```

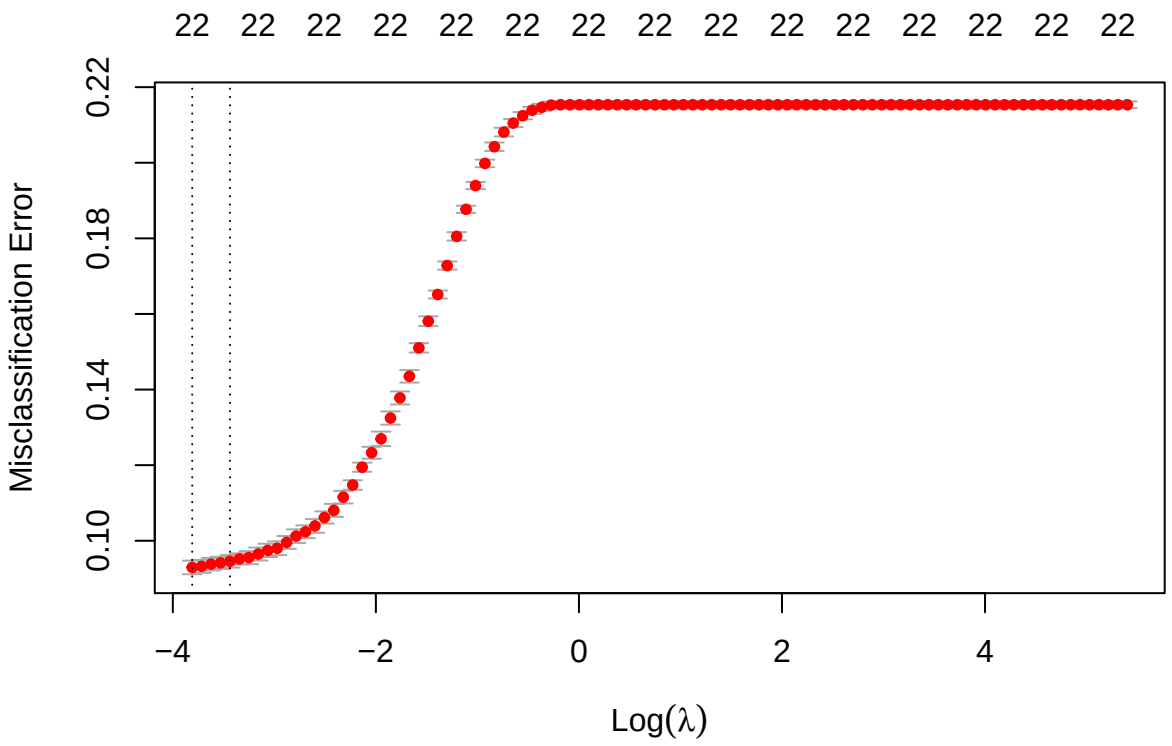
```
  lambda = lambda_min
)
coef(ridge_model)
```

```
## 23 x 1 sparse Matrix of class "dgCMatrix"
##                                         s0
## (Intercept)                     -4.591604682
## person_age                       0.002185293
## person_gendermale                0.015019924
## person_education.L              -0.034690058
## person_education.Q               0.007919038
## person_education.C              -0.019048617
## person_education^4              -0.022281856
## person_income                   -0.315897557
## person_emp_exp                  -0.005405100
## person_home_ownershipOWN        -1.390395863
## person_home_ownershipMORTGAGE   -0.750843115
## person_home_ownershipOTHER      -0.171525294
## loan_amnt                       -0.164841899
## loan_intentMEDICAL               0.336736920
## loan_intentEDUCATION            -0.223927016
## loan_intentVENTURE              -0.370796006
## loan_intentHOMEIMPROVEMENT       0.497792185
## loan_intentDEBTCONSOLIDATION     0.558104462
## loan_int_rate                    0.263097147
## loan_percent_income              9.640888494
## cb_person_cred_hist_length      -0.004235670
## credit_score                    -0.004942970
## previous_loan_defaults_on_fileNo  2.842972175
```

Plots

P.I Lambda CV

```
plot(cv_ridge)
```


22 22 22 22 22 22 22 22 22 22 22 22 22 22 22
Misclassification Error
0.10
0.14
0.18
0.22
−4
−2
0
2
4
Log(λ)