Modelling Creditability: A Bayesian Approach

Quan Tran

January 2024

Contents

1 Introduction

1.1 Loan and Loan Process Explained

A loan is essentially a sum of money or other assets given to a party with the expectation of repayment in the future. This includes repaying the original loan amount plus any accrued interest and finance charges. Loans vary in type, including personal, commercial, secured, and unsecured, and can be a one-time amount or an open-ended credit line within a specified limit.

Loans serve various purposes, such as purchasing goods, consolidating debt, funding business ventures, or investing. They contribute to economic growth by increasing the money supply and fostering competition through new business financing.

When a bank receives a loan application, it has to decide whether to approve or reject the loan based on the applicant's profile. There are two types of risks associated with this decision: if the applicant is a good credit risk and is likely to repay the loan, then not approving the loan results in a loss of business to the bank; if the applicant is a bad credit risk and is not likely to repay the loan, then approving the loan results in a financial loss to the bank.

1.2 Motivation

In the intricate world of financial behavior, creditability stands as a cornerstone, reflecting a myriad of socio-economic factors. Our investigation is underpinned by a sophisticated logistic regression model, meticulously crafted within a Bayesian framework. This methodological choice not only allows us to capture the nuanced interplay between individual predictors and creditability but also to discern the subtle yet significant influences exerted by occupational categories.

As we navigate through this labyrinth of data, our analysis is not just about crunching numbers; it's a quest to uncover the hidden stories behind creditworthiness. Each statistical insight offers a deeper understanding of the financial fabric that binds individuals and occupations, providing valuable perspectives for both risk assessment and strategic financial planning.

1.3 Outline

The report consists of the following parts: Introduction, Data Description, Models, Results, Discussion, Conclusion, and Appendices. First, we will formulate the problem and do some explanatory variable analysis on the train data. Subsequently, in the Models section, we present two BRMS models, one pooled and one hierarchical, and justify their priors. Next, we analyze the stability of the models through some convergence diagnostics and prior sensitivity analysis, evaluate the predictive performance through posterior predictive checks and assessments, and then compare the two models. Finally, in the Conclusion section, we will summarize this project by concluding the performance and pointing out the issues and potential improvements.

2 Data and Analysis Problem

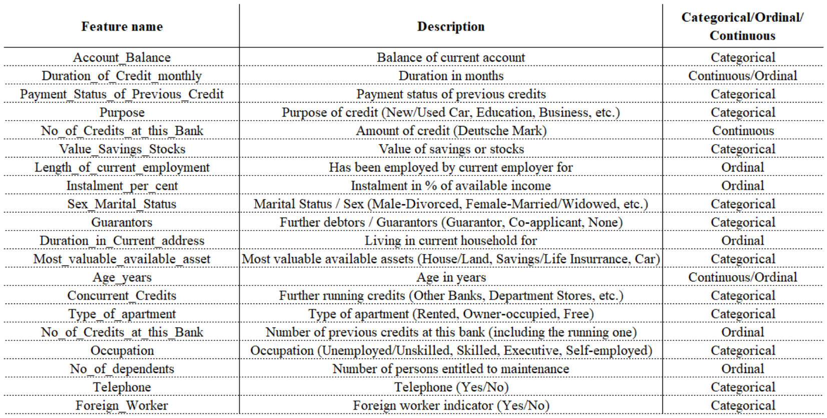
2.1 Dataset and Data Preprocessing

The German Credit Data was obtained from the Kaggle website. The data contains data 20 explanatory variables and 1 response variable, which is the classification whether an applicant is considered a Good or a Bad credit risk, for 1000 loan applicants.

Subsequently, we standardized the numerical variables in order to minimize the posterior dependency as the values are for from zero. The input explanatory numerical variables are on different scales so standardization is appropriate in this context.

Finally, we split 70% of the dataset into training dataset and 30% into testing dataset. While the training set is used for the explanatory data analysis and model fitting, the test set is utilized for predictions and accuracy testing purposes.

Table 1: Descriptions of all features in the datapoints



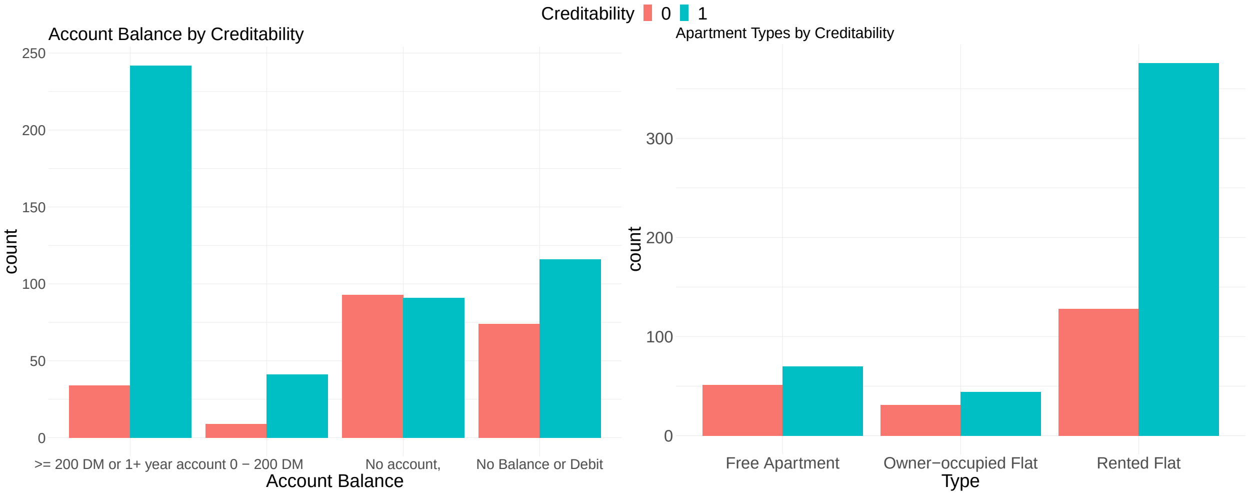
Feature name	**Description**	**Categorical/Ordinal/ Continuous**
Account_Balance	Balance of current account	Categorical
Duration_of_Credit_monthly	Duration in months	Continuous/Ordinal
Payment_Status_of_Previous_Credit	Payment status of previous credits	Categorical
Purpose	Purpose of credit (New/Used Car, Education, Business, etc.)	Categorical
No_of_Credits_at_this_Bank	Amount of credit (Deutsche Mark)	Continuous
Value_Savings_Stocks	Value of savings or stocks	Categorical
Length_of_current_employment	Has been employed by current employer for	Ordinal
Instalment_per_cent	Instalment in % of available income	Ordinal
Sex_Marital_Status	Marital Status / Sex (Male-Divorced, Female-Married/Widowed, etc.)	Categorical
Guarantors	Further debtors / Guarantors (Guarantor, Co-applicant, None)	Categorical
Duration_in_Current_address	Living in current household for	Ordinal
Most_valuable_available_asset	Most valuable available assets (House/Land, Savings/Life Insurrance, Car)	Categorical
Age_years	Age in years	Continuous/Ordinal
Concurrent_Credits	Further running credits (Other Banks, Department Stores, etc.)	Categorical
Type_of_apartment	Type of apartment (Rented, Owner-occupied, Free)	Categorical
No_of_Credits_at_this_Bank	Number of previous credits at this bank (including the running one)	Ordinal
Occupation	Occupation (Unemployed/Unskilled, Skilled, Executive, Self-employed)	Categorical
No_of_dependents	Number of persons entitled to maintenance	Ordinal
Telephone	Telephone (Yes/No)	Categorical
Foreign_Worker	Foreign worker indicator (Yes/No)	Categorical

Response variable:

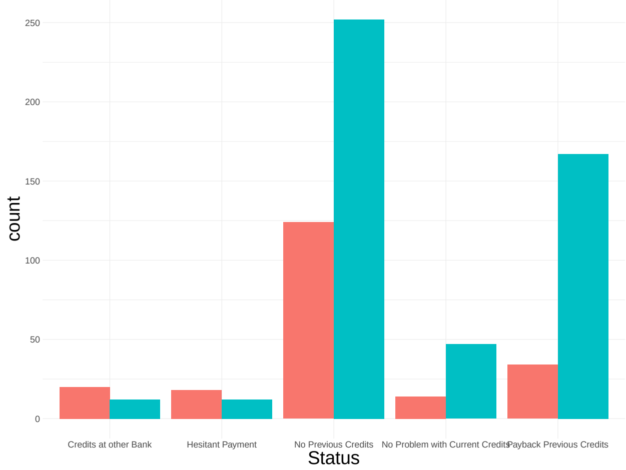
- `Creditability`: The indicator whether the subject has repaid the loan. The values is `1` if the subject repaid on time and is `0` if the subject has defaulted.

2.2 Explanatory Data Analysis

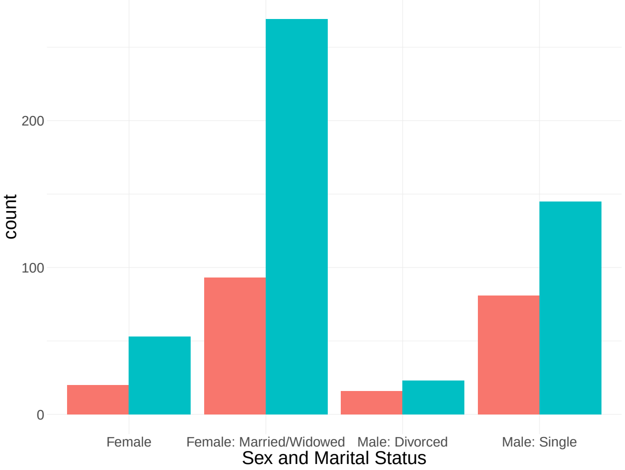
Plotting the histograms for the features that correspond to the response variable from the training dataset, we can gain insights into the correlation between the explanatory variables and the response variable.



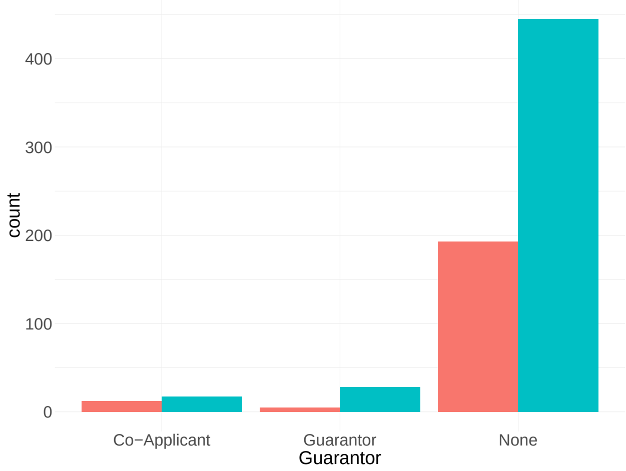
Payment Status of Previous Credit by Creditability



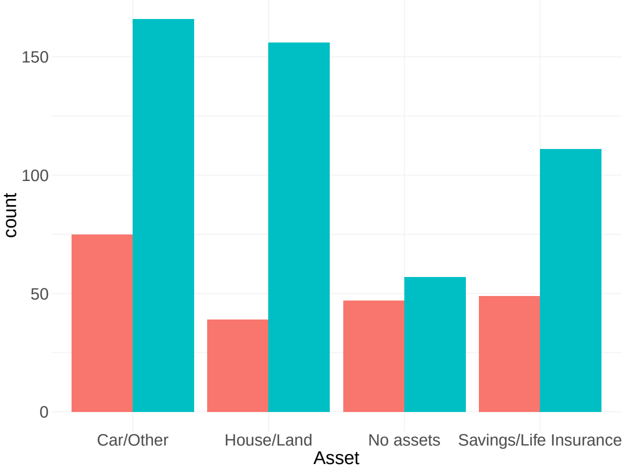
Sex and Marital Status by Creditability



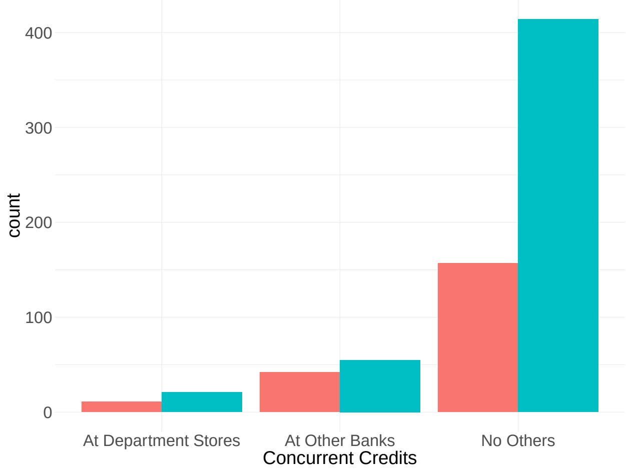
Guarantor by Creditability



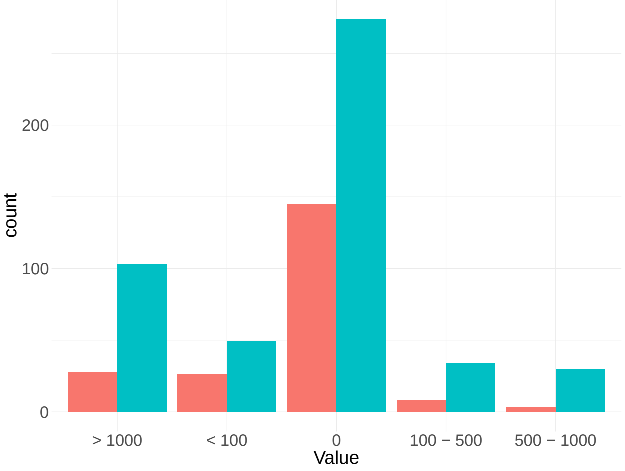
Most Valuable Available Asset by Creditability

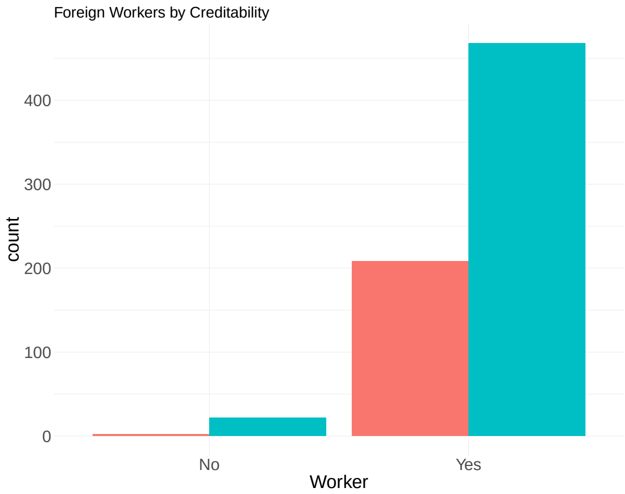


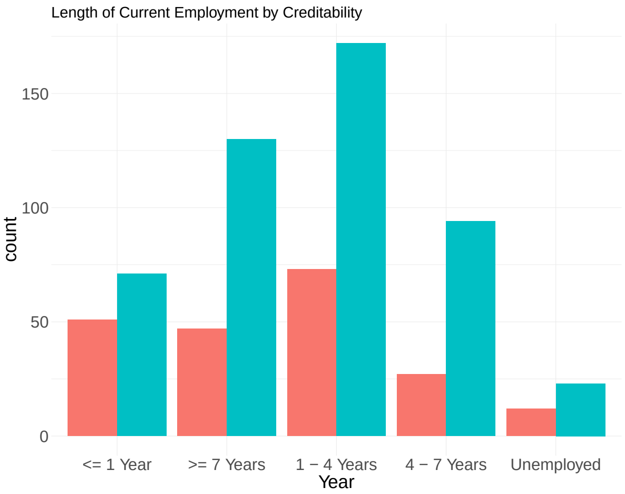
Concurrent Credits by Creditability

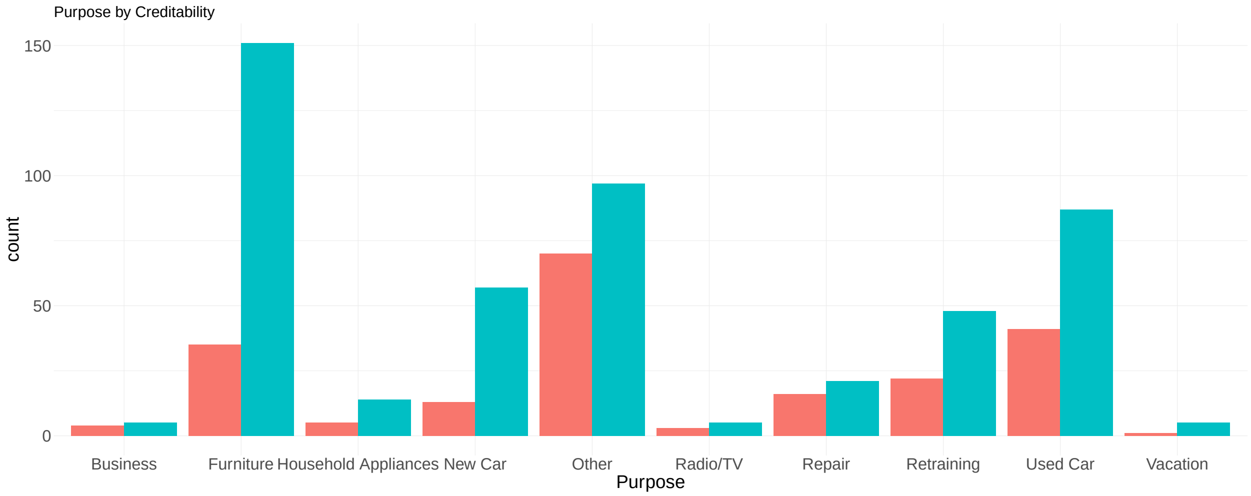


Value of Savings and Stocks by Creditability









There is one noticeable concern that can be readily pointed out from the histograms: There are under-representation of many classes in various predictor variables, including `Foreign_Worker`, `Concurrent_Credits`, `Guarantor`. For example, for the variable `Foreign_Worker`, there are 678 foreign workers, while there are only 22 natives.

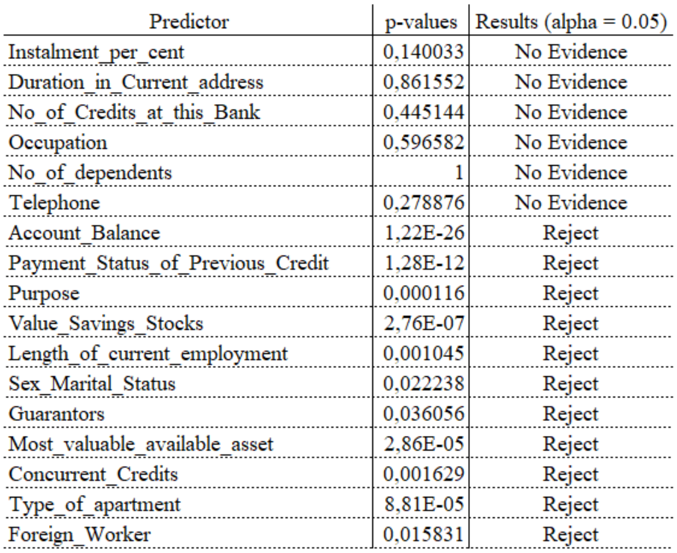
2.3 Feature Engineering and Data Processing

We can examine the dependency of the response variable on explanatory variables using Chi-Square Test for categorical variables and Correlation Matrix for continuous variables, from which we can determine the final explanatory variables for the models.

2.3.1 Chi-Square Test:

- **Hypotheses**:
  - Null Hypothesis (H0): There is no association between the predictor variable and target variable.
  - Alternate Hypothesis (H1): There is a significant association between the the predictor variable and target variable.
- **Significance level (α)**: $(\alpha) = 0.05$.

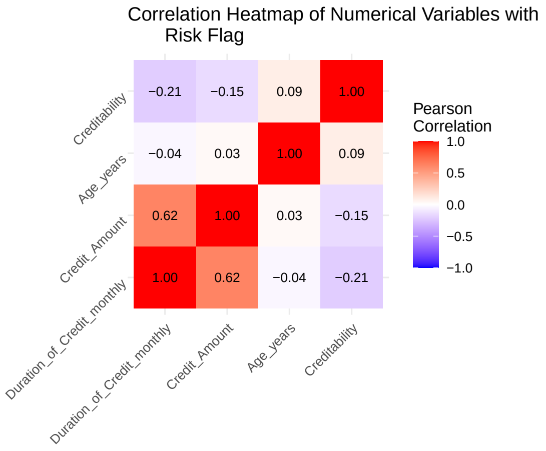
Table 2: Chi-square test p value results



Predictor	p-values	Results (alpha = 0.05)
Instalment_per_cent	0,140033	No Evidence
Duration_in_Current_address	0,861552	No Evidence
No_of_Credits_at_this_Bank	0,445144	No Evidence
Occupation	0,596582	No Evidence
No_of_dependents	1	No Evidence
Telephone	0,278876	No Evidence
Account_Balance	1,22E-26	Reject
Payment_Status_of_Previous_Credit	1,28E-12	Reject
Purpose	0,000116	Reject
Value_Savings_Stocks	2,76E-07	Reject
Length_of_current_employment	0,001045	Reject
Sex_Marital_Status	0,022238	Reject
Guarantors	0,036056	Reject
Most_valuable_available_asset	2,86E-05	Reject
Concurrent_Credits	0,001629	Reject
Type_of_apartment	8,81E-05	Reject
Foreign_Worker	0,015831	Reject

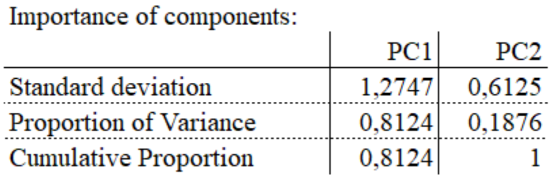
From the result table, we conclude that `Instalment_per_cent`, `Duration_in_Current_Address`, `No_of_Credits_at_this_Bank`, `Occupation`, `No_of_dependents`, and `Telephone` do not show a statistically significant association with the response variable `Creditability`. This outcome indicates that, based on our Chi-square analysis, these factors may not be critical predictors for `Creditability` in the context of this dataset.

2.3.2 Correlation Matrix



From the correlation plot, it can be observed that there is some correlation between `Credit_Amount` and `Duration_of_Credit_monthly`. The effects of multicollinearity can be severe. For example, it can cause the coefficient estimates to be unstable and sensitive to small changes in the model. This means that the coefficients can swing wildly based on which other independent variables are in the model. Multicollinearity can also make it difficult for the model to estimate the relationship between each independent variable and the dependent variable independently because the independent variables tend to change in unison. Jim Frost et al. 2017

One of the methods to deal with multicollinearity is to linearly combine the features `Credit_Amount` and `Duration_of_Credit_monthly` by Principal Component Analysis (PCA). PCA is a technique transforming the two original variables into a new set of variables called principal components, which are the linear combinations of the original variables. Each component is orthogonal to the others.



Importance of components:

	PC1	PC2
Standard deviation	1,2747	0,6125
Proportion of Variance	0,8124	0,1876
Cumulative Proportion	0,8124	1

From the above table, we can observe that the first component has already explained 81% of the total variance. It is reasonable to use this principal component as a feature instead of `Credit_Amount` and `Duration_of_Credit_monthly` and call it `PCA1`.

3 Models Description

In this project, two models will be used and analyzed: a Pooled model and a Hierarchical model. For the statistical model, Logistic Regression is chosen as the project aims to classify a binary outcome (Good Creditability: "1", Bad Creditability: "0").

Mathematical Expression for Both Models

The Logistic Regression model used to classify creditability y can be expressed as follows:

$$\Pr[Y_i = y | x_{1,i}, \dots, x_{m,i}] = \begin{cases} \frac{1}{1+e^{-LP_i}}, & \text{if } y = 1 \\ -\frac{1}{1+e^{LP_i}}, & \text{if } y = 0 \end{cases}$$

where the linear predictor LP_i is the weighted sum of the independent variables, representing the log-odds of an observation i for our Bernoulli-distributed random variable Y_i

$$LP_i = \boldsymbol{\beta} \cdot \mathbf{X}_i$$

with $\boldsymbol{\beta} \in \mathbb{R}^{m+1}$ is the vector of the regression coefficients. $\mathbf{X}_i \in \mathbb{R}^{m+1}$ are the explanatory variables indexed by i. Then we can logit transform the first function to obtain the linear predictors:

$$logit(Pr[Y_i = y]) = \boldsymbol{\beta} \cdot \mathbf{X}_i$$
$$log(\frac{Pr[Y_i = y]}{1 - Pr[Y_i = y]}) = \boldsymbol{\beta} \cdot \mathbf{X}_i$$

then the likelihood is

$$y_i \sim Bernoulli[log(\frac{Pr[Y_i = y]}{1 - Pr[Y_i = y]})]$$

In practical terms, parameters $\boldsymbol{\beta}$ will estimate the change in the log-odds of the outcome when the variable changes from its reference level (usually 0) to the other level for categorical variables, or the change in the log-odds of the outcome for continuous numerical variables.

3.1 Pooled Logistic Regression Model

For the Pooled Logistic Regression Model, we will use the following features: `Account_Balance`, `Payment_Status_of_Previous_Credit`, `Purpose`, `Value_Savings_Stocks`, `Length_of_current_employment`, `Sex_Marital_Status`, `Guarantors`, `Most_valuable_available_asset`, `Concurrent_Credits`, `Type_of_apartment`, `PCA1`, `Age_years`. You can inspect the BRMS code for the model here.

3.1.1 Priors Justification

For the β coefficients, a common choice is to use normal distributions as priors due to their mathematical convenience and the central limit theorem. For example, $\beta_i \sim \mathcal{N}(0, \sigma^2)$, where σ^2 is the variance for most of the parameters. In this project, we will choose 0 as location parameter and 1 as the variance for most of the parameters, expressing our humble prior knowledge about how the input variables can affect the target variable.

$$\beta_j \sim \mathcal{N}(0, 1) \forall j \in \{1, 2, 3, \ldots, k\}$$

However, for the variables `Account_Balance` and `Value_Savings_Stocks`, we assume a informative prior:

$$\beta_{AccountBalance}, \beta_{ValueSavingStocks} \sim \mathcal{N}(-1.5, 0.5)$$

The justification for our choice of this prior follows. According to Treece, K. et al. (2023), capital, representation of the commitment to contribute some of the borrower's own funds, is taken into account by the lenders. For the variable Account_Balance, the baseline category is the one having highest value (`> 1000`). Similarly, for the variable Value_Savings_Stocks, the baseline category is also the category has highest value (`>= 200 DM or 1+ year account`). Therefore, it is reasonable to assume that the log-odds of creditability decreases when these variables' values move from the baseline categories to the other categories.

3.2 Hierarchical Logistic Regression Model

For the Hierarchical Logistic Regression Model, we will use the same features as the Pooled Model as predictors. However, Hierarchical Model is more complex, as the data is categorized into G groups, with each category is a class in the variable Occupation. We believe that for each class of `Occupation`, the influences of `Account_Balance`, `Payment_Status_of_Previous_Credit`, and `Length_of_current_employment` on the log-odds of `Creditability` are different. For example, for two people having the same status of previous credit, we believe the person who is a skilled worker is possibly more creditable.

Let G: D → G is a surjective mapping from the set of datapoints D to the set of categories G of variable `Occupation`. Then we have the following Linear Predictor LP_i:

$$LP_i = \boldsymbol{\beta_0} + \mathbf{X}_i \boldsymbol{\beta} + \mathbf{Z}_i \boldsymbol{u}_{G(i)}$$

- α: The global intercept (Intercept).
- $\mathbf{X}_i$: The vector of predictor values for observation i.
- $\boldsymbol{\beta}$: The vector of fixed-effects coefficients (class "b").
- $\mathbf{Z}_i$: The vector of group-level predictor values for observation i.
- $\boldsymbol{u}_{G(i)}$: The vector of random effects for group $G(i)$.

You can inspect the BRMS code for the model here.

```r
model2 <- brm(
  # Specify the formula used
  model2form,
  # Specify the dataset used to train the model
  data = train_data,
  # Specify the observation model family
  family = bernoulli("logit"),
  # Pass the prior to the model
  prior = priors_model2,
  # Specify the number of chains used in the MCMC algorithm
  chains=4,
  # Specify the number of total draws and warm-up draws
  iter=4000, warmup=1000,
  # Cache the result
  file="cache/model2",
  # Declare the backend used
  backend="cmdstanr",
  # Specify the number of CPU cores used
  cores=8)
```

3.2.1 Priors Justification

For fixed effects (`class="b"`), we use the same set of priors for the coefficients of predictors

$$\beta_j \sim \mathcal{N}(0,1), \forall j \in \{0,1,2,3,\ldots,k\}$$

The logical reasoning remains the same.

For Random Effects Standard Deviations (`class = "sd"`), we use a weakly informative Cauchy priors

$$\sigma_{\text{Occupation}} \sim \text{Cauchy}(0,2)$$

.

For Correlation (`class = "cor"`), we use LKJ prior for the correlation matrix of random slopes within each `Occupation`

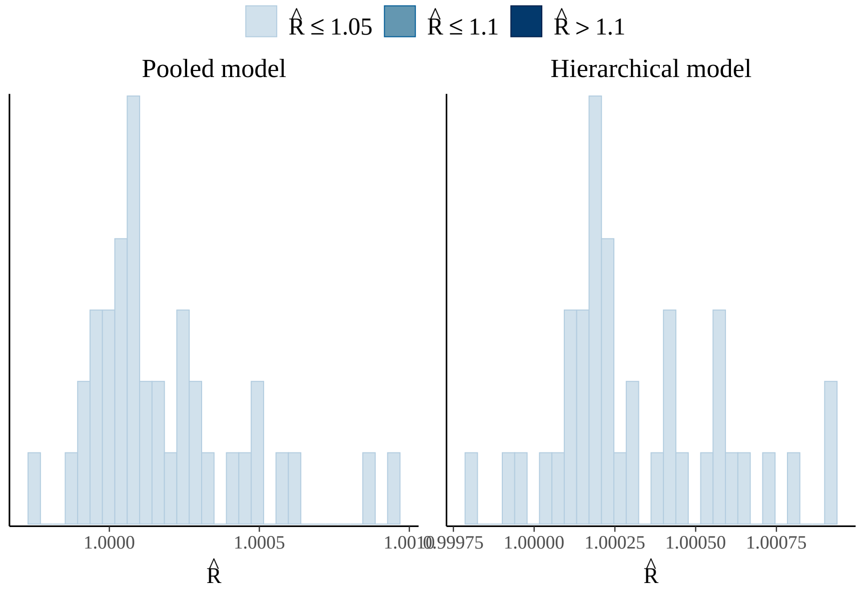
$$\Omega_u \sim \text{LKJ}(2)$$

4 Results and Comparisons

4.1 Convergence Diagnostics

4.1.1 $\hat{R}$ Diagnostics

The $\hat{R}$ values of the parameters for two models are plotted into two histograms as follows:



From the histograms, we can observe that the all $\hat{R}$ values are smaller than 1.05. This suggests that the good convergence across chains.

4.1.2 Divergences Diagnostics

A divergent transition occurs when the Hamiltonian Monte Carlo sampler is unable to accurately simulate the trajectory through the parameter space. This typically happens in regions of high curvature in the posterior distribution, which are difficult for the sampler to navigate. Divergent transitions are a sign that the sampler may not be exploring the posterior distribution effectively, which can lead to biased estimates and unreliable inference. They indicate that the sampler's trajectory had to be prematurely halted to avoid numerical errors.

In this section, we will examine the number of divergent draws of the two models.

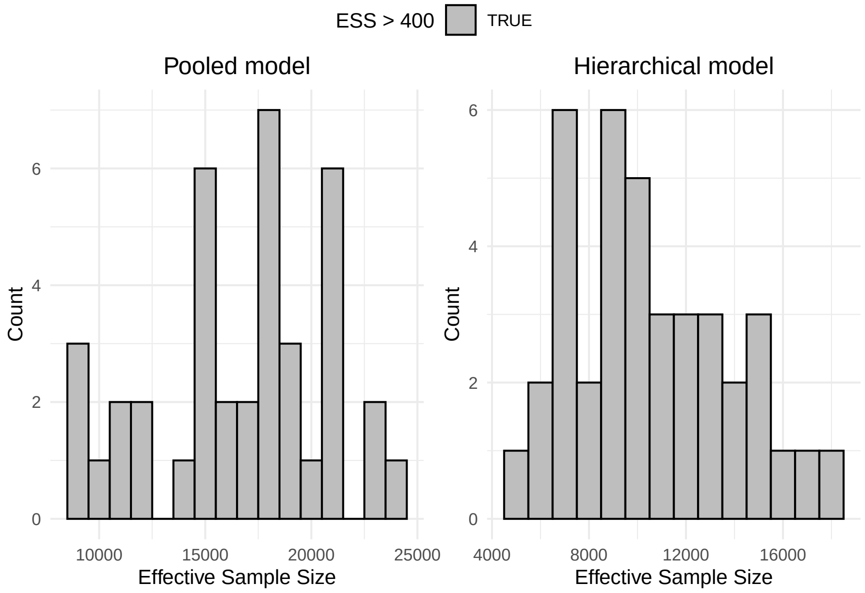
Number of Divergent draws:

```
Pooled model:


           Transitions
Divergent            0
Convergent       12000


Hierarchical model:


           Transitions
Divergent           50
Convergent       11950
```

For the pooled model, we have 0 divergent transitions. However, on the contrary, hierarchical model has $\sim$ 100 divergent transitions. While this number should not be ignored, occasional divergences might not significantly impact the overall inference, considering the huge number of total draws of 12000.

4.1.3 Effective Sample Size (ESS) diagnostics

The Effective sample size (ESS) values of the parameters for two models are plotted into two histograms as follows:



In overall, we have obtained good ESS for all parameters since they are all greater than 100 times the number of chains (=4) Vehtari et al. 2021. Effective Sample Sizes across our parameters were robust, providing confidence in the precision of our posterior estimates. The absence of significant divergences further supported the adequacy of our sampling process.

4.1.4 Summary

Given these satisfactory diagnostic results, we did not make adjustments to the model's specifications or priors. This decision was based on the overall adequacy of the convergence indicators, suggesting that the model was capturing the underlying data structure effectively.

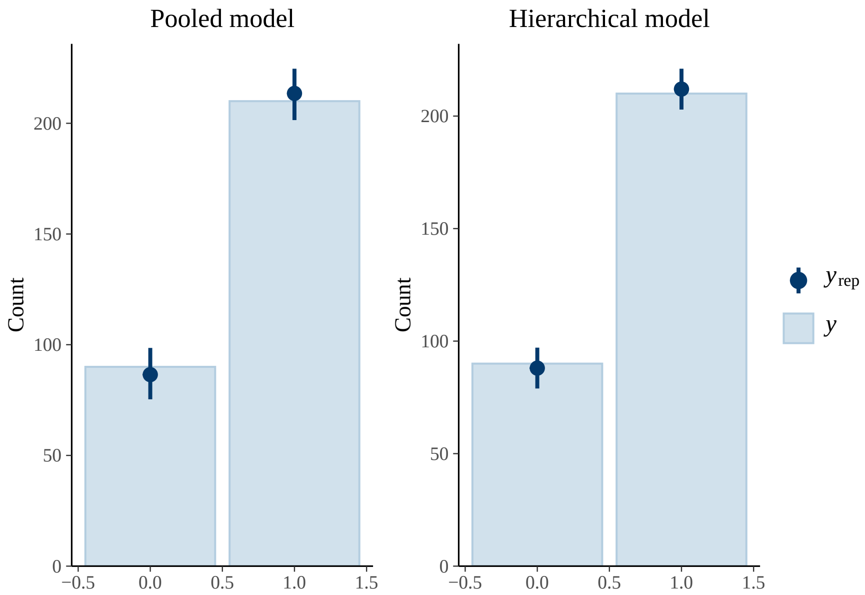
Therefore, we are confident in the reliability of our model's results. The diagnostics support the validity of our findings, though we acknowledge the importance of continual vigilance for potential convergence issues in Bayesian modeling. Future studies may explore further refinements in model specifications, particularly if working with more complex datasets or different model structures.

4.2 Model Comparisons

4.2.1 Posterior Predictive Performance Checks

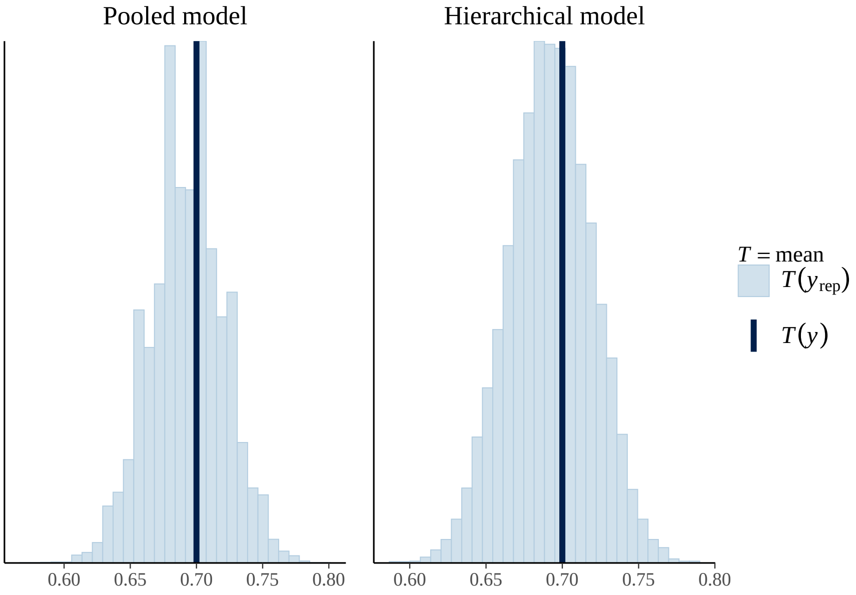
The following plot displays the observed frequencies of the response variable in the data to the frequencies simulated from the model. It's particularly useful for categorical data in this context. The bars show the distribution of creditability (by count) and the predicted number of each category (`Good` or `Bad`) from the model, with the median as a dot and error bar.

Ideally, the observed and simulated data should be similar. If the models fit the data well, the bars for the observed data should fall within the range of variation of the bars for the simulated data.



From the plots, it can be observed that the two models replicate the original data quite well, with bars of the observed data all fall within the range of the variation of the simulated data. In the predictive performance assessment section, we will provide numerical metrics to evaluate the true performance of both models' predicting ability.

The two following plots will show the distribution of mean calculated from the simulated posterior predictive data. The observed statistic from `test_data` is highlighted and overlaid on this distribution. If the model fits the data well, the observed mean should fall within the range of the distribution of the simulated statistic. This would indicate that the model can reproduce summary statistics similar to what is observed in the actual data.



Upon examining our PPC plots, we observe a commendable alignment in the central tendencies between the observed and simulated data, suggesting that our model effectively captures the primary patterns in the creditability data. However, notable discrepancies in the tails of the distribution indicate potential underestimation of extreme values. This could imply model misspecification, perhaps due to not fully accounting for high-risk profiles in the data.

Further, the underrepresentation of certain creditability outcomes in the simulated data suggests that incorporating additional predictors or exploring interaction effects could enhance our model's accuracy. For instance, incorporating economic indicators might improve the model's ability to capture these tail behaviors.

4.2.2 Predictive Performance Assesments

To gain a more comprehensive understanding of the predictive performance of our models, we construct a confusion matrix for each model. For every observation in the test dataset, we calculate the mean prediction based on 12,000 samples. Using a threshold of 0.5, we classify each data point as Good Creditability if the mean prediction exceeds this threshold; otherwise, we classify it as Bad Creditability.

The predictions are then compared to the true labels to compute the number of True Positive (TP), False Positive (FP), True Negative (NG), False Negative (NG).

The Confusion Matrix:

```
Model 1:

                Actual False Actual True
Predicted False           46          29
Predicted True            44         181

Model 2:

                Actual False Actual True
Predicted False           48          29
Predicted True            42         181
```

Then we will evaluate the classification accuracy of both models using some of the common metrics: Accuracy, Precision, Recall, and F1 Score.

- **Accuracy**: Accuracy measures how often a classifier makes the correct prediction. It is the ratio of the number of correct predictions to the total number of predictions.
- **Precision**: Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. It's a measure of a classifier's exactness. High precision relates to a low rate of false positives, and it is particularly important in cases where the cost of a false positive is high.
- **Specificity**: Specificity measures the proportion of actual negatives that are correctly identified as such (e.g., the percentage of non-creditworthy individuals who are correctly identified by the model).
- **Recall (or Sensitivity)**: Recall is the ratio of correctly predicted positive observations to all observations in the actual class. It's a measure of a classifier's completeness. High recall relates to a low rate of false negatives, and it is important in cases where the cost of a false negative is high.
- **F1 Score**: The F1 Score is the weighted average of Precision and Recall. It's a measure of a test's accuracy and considers both the precision and the recall. This is useful when seeking a balance between Precision and Recall, especially if there is an uneven class distribution.

```
Pooled model:
Accuracy:    0.76
Precision:   0.80
Specificity: 0.51
Recall:      0.86
F1 Score:    0.83

Hierarchical model:
Accuracy:    0.76
Precision:   0.81
Specificity: 0.53
Recall:      0.86
F1 Score:    0.84
```

From the metrics obtained, it is apparent that there is no considerable differences between the two models. However, the Hierarchical model is performing strictly better in some aspects.

The two metrics we would be focus on are Precision and Specificity. Since the cost false positivity is high (the loss of lending who would default eventually outweigh the profit gained), the Precision and Specificity metrics are appropriate since they put emphasis on classifier's exactness. Even if the two models have quite good Precision scores (0.8 and 0.81 for Pooled model and Hierarchical model respectively), the Specificity scores are quite humble (0.51 and 0.53 for Pooled model and Hierarchical model respectively). Therefore, these models should be refined before deployed in practical use.

4.2.3 Model Comparisons Using ELPD

ELPD stands for Expected Log Pointwise Predictive Density. It is a measure of a model's predictive accuracy, where higher values indicate better predictive performance. ELPD estimates the (log) probability density of the observed data under the model, averaged over the posterior distribution. A higher ELPD value suggests that the model is more likely to predict new, similar data accurately. The LOO-CV estimation of ELPD for both models is given and compared as follows:

Pooled model:

```
Computed from 12000 by 700 log-likelihood matrix

         Estimate   SE
elpd_loo   -356.6 15.7
p_loo        34.4  2.0
looic       713.2 31.4
------
Monte Carlo SE of elpd_loo is 0.1.

All Pareto k estimates are good (k < 0.5).
See help('pareto-k-diagnostic') for details.
```

Hierarchical model:

```
Computed from 12000 by 700 log-likelihood matrix

         Estimate   SE
```

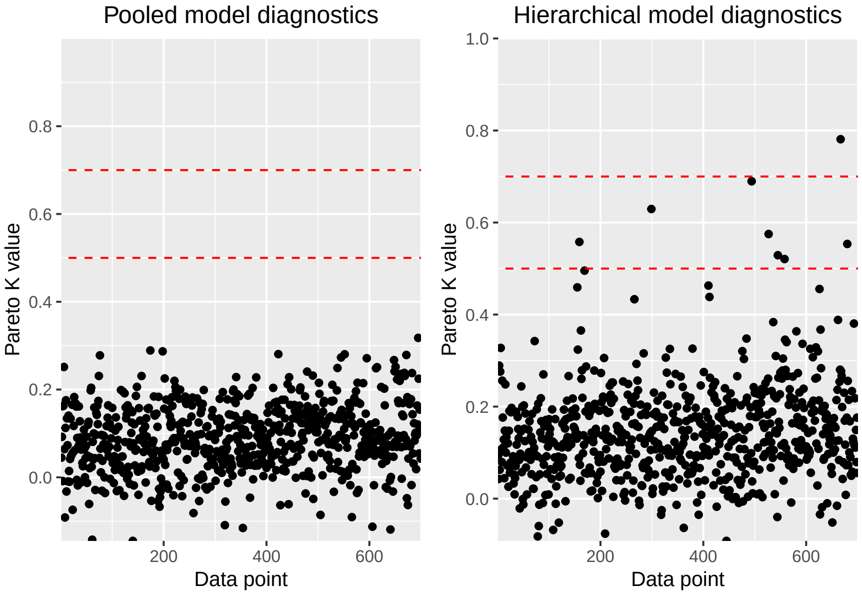
```
elpd_loo   -358.6 16.4
p_loo        50.8  3.0
looic       717.2 32.8
------
Monte Carlo SE of elpd_loo is NA.

Pareto k diagnostic values:
                         Count Pct.    Min. n_eff
(-Inf, 0.5]   (good)     692   98.9%   2912
 (0.5, 0.7]   (ok)         7    1.0%   2023
   (0.7, 1]   (bad)        1    0.1%   480
   (1, Inf)   (very bad)   0    0.0%   <NA>
See help('pareto-k-diagnostic') for details.

       elpd_diff se_diff
model1  0.0       0.0
model2 -2.0       4.4
```

The metric `se_diff` stands for the Standard Error of the Difference in ELPD between two models. It is used when comparing the ELPD of two different models to assess if the difference in their predictive performances is statistically significant. Since the ratio $\frac{se\_diff}{elpd\_diff}$ is quite large, this suggests that the different might not be statistically significant.

While the ELPD estimation (scaled by standard error) of the pooled model yields a higher value than that of the hierarchical model (as indicated by a negative value of `elpd_diff`), this result might not be statistically significant due to the aforementioned reason.

Then we will plot the Pareto $\hat{k}$ values in order to assess the reliability of LOO approximation:

- $\hat{k} \leq 0.5$ suggests high accuracy in estimating the corresponding LOO-CV component.
- $0.5 < \hat{k} \leq 0.7$ indicates lower but still acceptable accuracy.
- $\hat{k} \geq 0.7$ suggests that the PSIS-LOO approximation may not be reliable for that component/observation.



While most of the Pareto $\hat{k}$ values for the Pooled model is below 0.5, there is one $\hat{k}$ values for the Hierarchical model higher than 0.7, and couples of $\hat{k}$ values fall within the range of [0.5, 0.7]. This suggests that the observation corresponding to this $\hat{k}$ values is highly influential. This is expected since the Hierarchical model is complex and the introduction of random effects may lead to overfitting. These high $\hat{k}$ values may stem from the fact that some explanatory variables are imbalanced, with some categories have a smaller number of observations than the others.

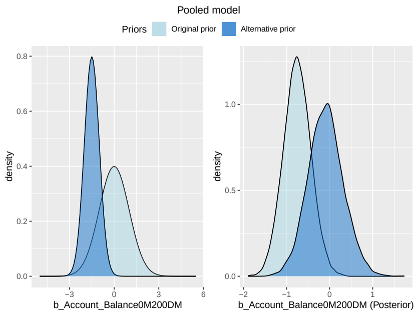
4.3 Prior Sensitivity Analysis

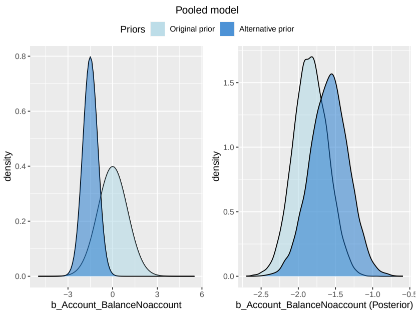
In this section, we will inspect how the marginal posterior distributions of the parameters for the variables `Account_Balance` and `Value_Savings_Stocks` change when we adopt weakly informative priors $\beta_i \sim \mathcal{N}(0,1)$ instead of the currently informative priors $\beta_i \sim \mathcal{N}(-1.5, 0.5)$. This is done by fitting another model with modification only in the prior settings.

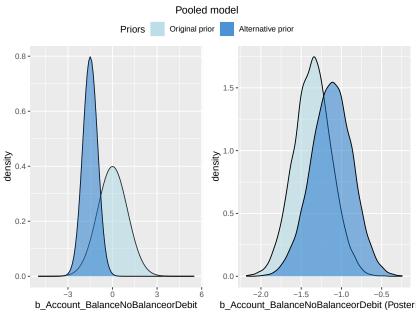
4.3.1 Pooled Logistics Regression models

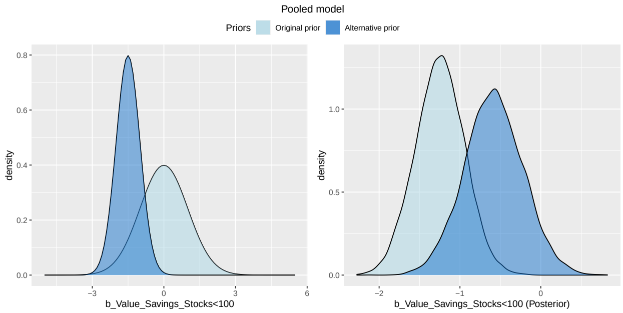
You can inspect the BRMS code for the alternative model here.

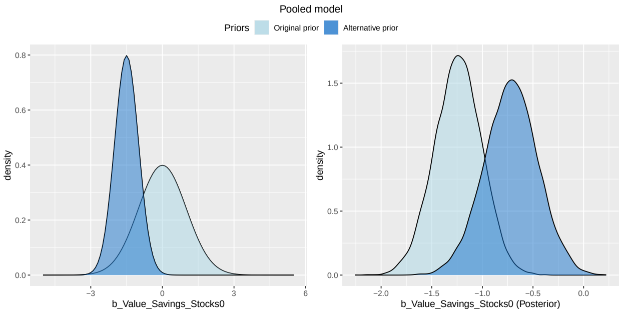
All the chains for this model have converged, and no divergence warning are given. By plotting the obtained posterior and the original posterior, we can analyse the discrepancies between them.

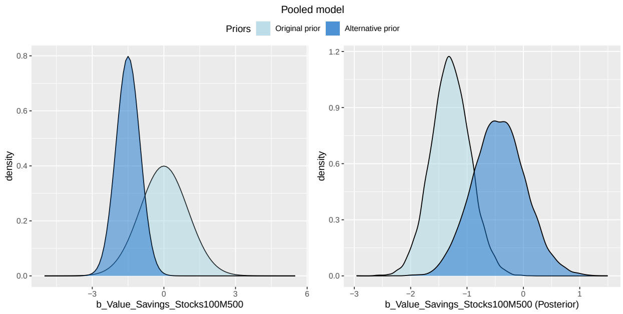


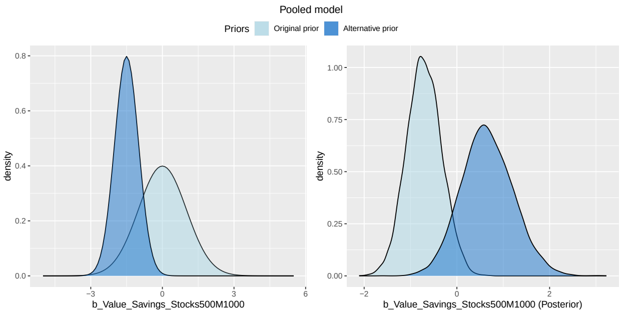










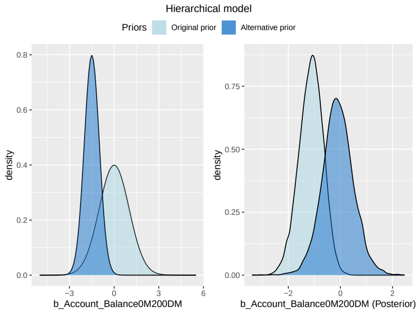


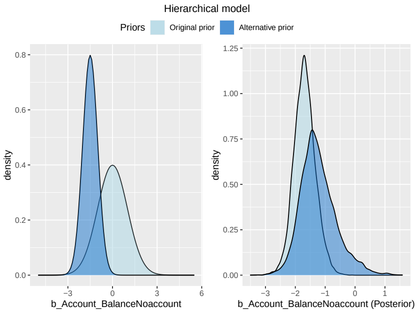
It is expected that the posteriors will be shifted away due to the change in location parameter from the prior. On the one hand, for the parameter `b_Account_BalanceNoaccount` and `b_Account_BalanceNoBalanceorDebit`, the shape of the posterior distributions do not change significantly. This suggests that these two parameters are quite robust to prior choices. On the other hand, for the rest of parameters, we can notice that there are noticeable changes in the resulting shape. To sum up, even though there are some trivial discrepancies, we can conclude that the Pooled model is robust to prior choices.

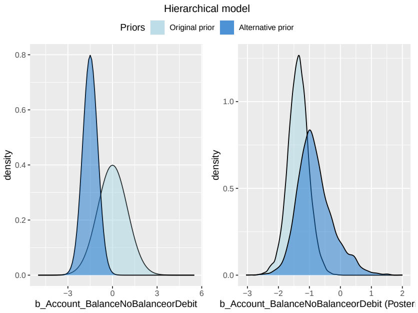
4.3.2 Hierarchical Logistic Regression model

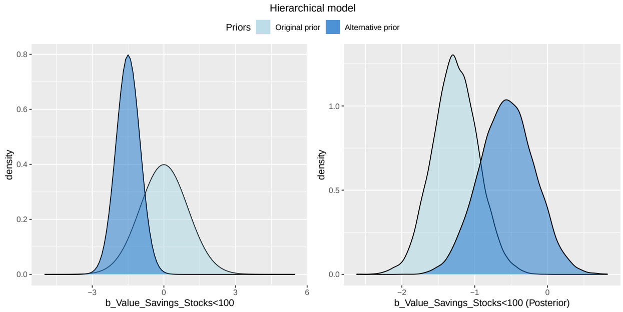
You can inspect the BRMS code for the model here.

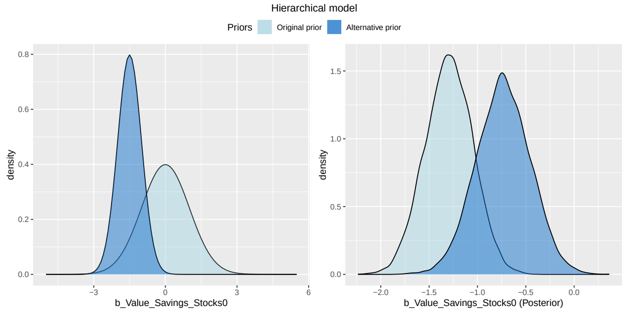
All the chains for this model have converged, and no divergence warning are given. By plotting the obtained posterior and the original posterior, we can analyse the discrepancies between them.

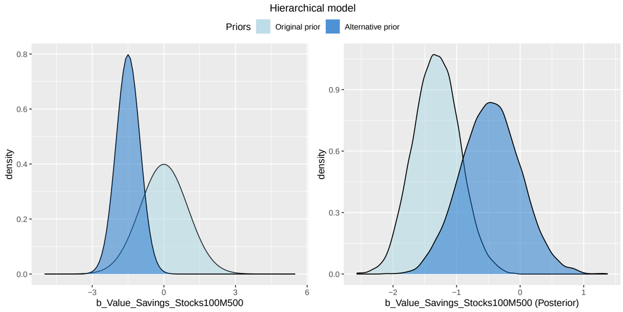


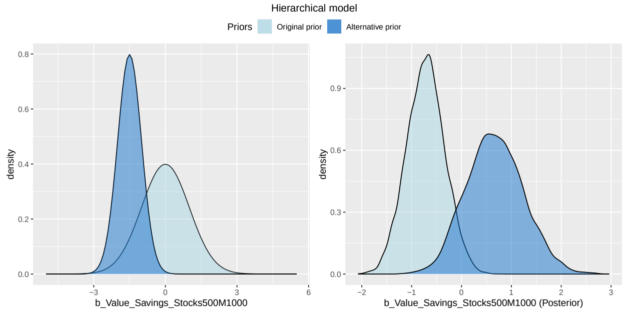












For the Hierarchical model, even with a significantly more informative prior used, there is no substantial change in the shape of the posterior distribution but only minor changes in the location. Therefore, it can be concluded that the Hierarchical model is quite robust to prior changes.

5 Conclusion and potential improvements

5.1 Conclusion

This project aimed to enhance the accuracy and reliability of credit risk assessment models using advanced statistical techniques. Our investigation involved a comparative analysis of the Pooled Logistic Regression Model and the Hierarchical Logistic Regression Model. The findings revealed that while both models exhibit robust predictive capabilities, the Hierarchical Logistic Regression Model demonstrated a slightly superior performance in terms of predictive accuracy. This improvement is attributed to its ability to account for the hierarchical nature of financial data, offering a more nuanced understanding of credit risk factors.

The significance of these findings lies in their potential application within the financial sector. By adopting more sophisticated models like the Hierarchical Logistic Regression Model, financial institutions can achieve a more accurate assessment of credit risk. This advancement is crucial in reducing the probability of loan defaults, thereby enhancing the stability and profitability of these institutions. Additionally, our findings contribute to the broader field of financial analytics by demonstrating the efficacy of hierarchical modeling approaches in complex data environments.

In conclusion, this project represents a step forward in the field of credit risk assessment. By leveraging advanced statistical models, we can provide more accurate and reliable tools for financial institutions, ultimately contributing to a more robust financial system. Future research in this area holds great promise for continuing to refine and enhance these tools, adapting them to the evolving landscape of credit risk.

5.2 Issues and potential improvements

While our models demonstrate robust predictive capabilities, we acknowledge limitations such as the dataset's scope and potential biases inherent in historical lending data. These aspects underline the need for continual refinement and testing of our models against diverse and updated datasets.

Future research should focus on integrating emerging socio-economic variables, exploring alternative Bayesian models, and applying these models across varied financial landscapes. There lies an untapped potential in harnessing machine learning techniques to enhance model accuracy and interpretability further.

As the dataset suffers from the from the under-representation of some classes in some features, we can balance the dataset by keeping the data minority class and decreasing the size of the majority class. This technique is called undersampling. This way, we can extract more precise insights from the originally imbalanced dataset.

Another way to improve the project is to add a Receiver operating characteristics curve (ROC curve). This curve helps us understand how the predictive performance of this binary classification model changes when we change the threshold value by plotting True positive rate (TPR) against the False positive rate (FPR) at each threshold setting.

6 Self-reflection

This project helps me familiarize myself with the workflow of Bayesian inference. I had to come up with an idea, find the appropriate data, build a model, as well as familiarize with the background. I chose this topic because I have a minor in Business Analytics and I have a special interest in Quantitative Methods that can be applied in Finance and Business context. In overall, this project gives me a chance to hone my skills in time management, task management, problem-solving, as well as technical knowledge such as programming and statistics. I am pretty much content with invaluable learning outcome from this work.

7 Appendices and References

7.1 BRMS code

7.1.1 Pooled Logistic Regression Model

Model with Original Priors:

```r
# Priors
priors_model1 <- c(
  prior(normal(0, 1), class = "b"),
  prior(normal(0, 1), class = "Intercept"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Account_Balance0M200DM"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Account_BalanceNoaccount"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Account_BalanceNoBalanceorDebit"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Value_Savings_Stocks<100"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Value_Savings_Stocks0"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Value_Savings_Stocks100M500"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Value_Savings_Stocks500M1000"))

# Specify a Bayesian model formula
model1form <- bf(Creditability ~ 1 + Account_Balance + Payment_Status_of_Previous_Credit +
                   Purpose + Value_Savings_Stocks + Length_of_current_employment +
                   Sex_Marital_Status + Guarantors + Most_valuable_available_asset +
                   Concurrent_Credits + Type_of_apartment + PCA1 + Age_years)
```

```r
model1 <- brm(
  # Specify the formula used
  model1form,
  # Specify the dataset used to train the model
  data = train_data,
  # Specify the observation model family
  family = bernoulli("logit"),
  # Pass the prior to the model
  prior = priors_model1,
  # Specify the number of chains used in the MCMC algorithm
  chains=4,
  # Specify the number of total draws and warm-up draws
  iter=4000,
  warmup=1000,
  # Cache the result
  file="cache/model1",
  # Declare the backend used
  backend="cmdstanr",
  # Specify the number of CPU cores used
  cores=8)
```

Model with Alternative Priors:

```r
# Priors
priors_model1_alt <- c(prior(normal(0, 1), class = "b"),
                       prior(normal(0, 1), class = "Intercept"))
```

```
# Specify the Bayesian formula for the model
model1_alt_form <- bf(Creditability ~ 1 + Account_Balance + Payment_Status_of_Previous_Credit +
                        Purpose + Value_Savings_Stocks + Length_of_current_employment +
                        Sex_Marital_Status + Guarantors + Most_valuable_available_asset +
                        Concurrent_Credits + Type_of_apartment + PCA1 + Age_years)
```

```
model1_alt <- brm(
  # Specify the formula used
  model1_alt_form,
  # Specify the dataset used to train the model
  data = train_data,
  # Specify the observation model family
  family = bernoulli("logit"),
  # Pass the prior to the model
  prior = priors_model1_alt,
  # Specify the number of chains used in the MCMC algorithm
  chains=4,
  # Specify the number of total draws and warm-up draws
  iter=4000, warmup=1000,
  # Cache the result
  file="cache/model1_alt",
  # Declare the backend used
  backend="cmdstanr",
  # Specify the number of CPU cores used
  cores=8)
```

7.1.2 Hierarchical Logistic Regression Model

Model with Original Priors:

```
# Declare priors for the Hierarchiccal model
# Priors for non-grouped parameters
non_grouped_priors_model2 <- c(
  prior(normal(0, 1), class = "b"),
  prior(normal(0, 1), class = "Intercept"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Account_Balance0M200DM"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Account_BalanceNoaccount"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Account_BalanceNoBalanceorDebit"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Value_Savings_Stocks<100"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Value_Savings_Stocks0"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Value_Savings_Stocks100M500"),
  prior(normal(-1.5, 0.5), class = "b", coef = "Value_Savings_Stocks500M1000"))

# Priors for grouped parameters
grouped_priors_model2 <- c(
  # Prior for standard deviations of random effects
  prior(cauchy(0, 2), class = "sd"),
  # Prior for the correlation matrix of random effects
  prior(lkj(2), class = "cor"))


# Concatenate two parts into a complete set of priors
```

```
priors_model2 <- c(non_grouped_priors_model2, grouped_priors_model2)


# Specify a Bayesian model formula
model2form <- bf(Creditability ~ 1 + Account_Balance + Payment_Status_of_Previous_Credit +
          Purpose + Value_Savings_Stocks + Length_of_current_employment +
          Sex_Marital_Status + Guarantors + Most_valuable_available_asset +
          Concurrent_Credits + Type_of_apartment + PCA1 + Age_years +
          (1 + Length_of_current_employment + Account_Balance +
          Payment_Status_of_Previous_Credit | Occupation))
```

```
model2 <- brm(
  # Specify the formula used
  model2form,
  # Specify the dataset used to train the model
  data = train_data,
  # Specify the observation model family
  family = bernoulli("logit"),
  # Pass the prior to the model
  prior = priors_model2,
  # Specify the number of chains used in the MCMC algorithm
  chains=4,
  # Specify the number of total draws and warm-up draws
  iter=4000, warmup=1000,
  # Cache the result
  file="cache/model2",
  # Declare the backend used
  backend="cmdstanr",
  # Specify the number of CPU cores used
  cores=8)
```

Model with Alternative Priors:

```
# Declare priors for the Hierarchical model
# Priors for non-grouped parameters
non_grouped_priors_model2_alt <- c(prior(normal(0, 1), class = "b"),
                       prior(normal(0, 1), class = "Intercept"))

# Priors for grouped parameters
grouped_priors_model2_alt <- c(prior(cauchy(0, 2), class = "sd"), # Prior for standard deviations of ra
                       prior(lkj(2), class = "cor")) # Prior for the correlation matrix of random effec

# Concatenate two parts into a complete set of priors
priors_model2_alt <- c(non_grouped_priors_model2_alt, grouped_priors_model2_alt)

# Specify a Bayesian model formula
model2form <- bf(Creditability ~ 1 + Account_Balance + Payment_Status_of_Previous_Credit +
          Purpose + Value_Savings_Stocks + Length_of_current_employment +
          Sex_Marital_Status + Guarantors + Most_valuable_available_asset +
          Concurrent_Credits + Type_of_apartment + PCA1 + Age_years +
          (1 + Length_of_current_employment + Account_Balance +
          Payment_Status_of_Previous_Credit | Occupation))
```

```
model2_alt <- brm(
  # Specify the formula used
  model2form,
  # Specify the dataset used to train the model
  data = train_data,
  # Specify the observation model family
  family = bernoulli("logit"),
  # Pass the prior to the model
  prior = priors_model2_alt,
  # Specify the number of chains used in the MCMC algorithm
  chains=4,
  # Specify the number of total draws and warm-up draws
  iter=4000, warmup=1000,
  # Cache the result
  file="cache/model2_alt",
  # Declare the backend used
  backend="cmdstanr",
  # Specify the number of CPU cores used
  cores=8)
```

7.2 References

Frost, J. (2017) Multicollinearity in regression analysis: Problems, detection, and solutions, Statistics By Jim. Available at: https://statisticsbyjim.com/regression/multicollinearity-in-regression-analysis/#comments (Accessed: 13 January 2024).

Vehtari, Aki, Andrew Gelman, Daniel Simpson, Bob Carpenter, and Paul-Christian Bürkner. 2021. "Rank-Normalization, Folding, and Localization: An Improved $\widehat{R}$ for Assessing Convergence of Mcmc." Bayesian Analysis. https://doi.org/10.1214/20-BA1221.

Treece, K. (2023) Understand the 5 C's of credit before applying for a loan, Forbes. Available at: https://www.forbes.com/advisor/credit-score/5-cs-of-credit/ (Accessed: 14 January 2024).