Credit Allocation Mix in Peer-to-Peer Lending: A Network Study on Bondora

Social Network Analysis - Group 7

Floris Vermeulen   Martijn van Iterson   Niek Fleerakkers

Patryk Grodek   Samir Sabitli

2025-11-16

Table of Contents

List of Figures

List of Tables

1 Introduction

With the advent of online peer-to-peer (P2P) lending platforms, the traditional methods of financial intermediation have been usurped by individual choice; with no stringent third parties involved in transactions, individuals have greater power in sourcing credit. This presents new realities for individuals disenfranchised by the traditional financial system (Alistair Milne & Paul Parboteeah, 2017). Traditional financial literature shows that individually, investors exhibit behavioural biases (Agrawal, 2012; Ayal, Hochman, & Zakay, 2011) in how they choose their investments, however, numerous studies demonstrate that these behavioural biases are present among non-professional P2P borrowers. For example, Herzenstein, Dholakia, & Andrews (2011) and Lee & Lee (2012) suggest that users cluster around popular loans, exhibiting 'herding' behaviour. Literature focuses predominantly on funding success (Yao, Chen, Wei, Chen, & Yang, 2019), however, Ayal, Bar-Haim, & Ofir (2018) suggests that there is insufficient attention on how borrowers form their loan portfolios. As the authors indicate, this area of research is important in understanding why less professional investors deviate from known models of rational investor behaviour. Our study aims to model and understand how P2P borrowers' behaviours influence what kinds of loans they acquire.

Investigating behavioural biases, Ayal et al. (2018) find that investors' familiarity with specific assets can make them seem less risky, making recognisable assets more attractive. However, such assets lead to undiversified portfolios. Evidence with P2P platforms supports the familiarity bias in lenders, as Galak, Small, & Stephen (2010) demonstrate that lenders choose borrowers who are similar across demographic attributes such as gender, occupation, and ethnicity. Based on these notions, we formulate our first study, asking:

Research Question 1: *Do borrowers tend to choose similar loan uses based on similarities across their attributes?*

This question is answered using linear Quadratic Assignment Procedure (QAP) regressions, as we can represent similarities across loan uses and attributes as matrices that can be used as inputs in the model. Further, we integrate loan use frequencies to understand how the prevalence of specific loan types affects borrowers' portfolio compositions.

Using data from LendingClub, Serrano-Cinca, Gutiérrez-Nieto, & López-Palacios (2015) study funding success, finding that the reported loan use is a significant factor explaining differences in default rates. Specifically, the authors argue that there is heterogeneity in the credit riskiness of individuals seeking specific loan purposes, indicating that credit rating can explain some variation in borrowers' choice of loan use. Therefore, we formulate the following hypothesis:

Hypothesis 1: Borrowers with the same credit ratings choose similar loan uses

Supporting the familiarity bias, Ravina (2019) studies personal characteristics in P2P markets, finding that loans tend to be awarded to those sharing personal characteristics such as occupation. As Ayal et al. (2018) implies, this can be generalised to suggest that lenders' loan grants reflect a wider network of similar borrower preferences, which can reflect the homogeneity in loan use observed by Serrano-Cinca et al. (2015). Thus,

Hypothesis 2: Borrowers sharing the same occupation area tend to share loan uses

Further, behavioural finance suggests that individuals' financial decisions are not independent,

but rather, shaped by bounded rationality and psychological preferences (Kahneman & Tversky, 1979). In the P2P context, Ayal et al. (2018) shows that these preferences can influence how borrowers choose loan types and how this can overlap with other borrowers' decisions. Using the Exponential Random Graph Model (ERGM) framework, we can identify how these behaviours manifest as structural regularities in a borrower-to-loan-use network. Therefore, we ask:

Research Question 2: *What behavioural mechanisms affect the structure of borrower-to-loan-use connections?*

The aforedescribed section outlined that the familiarity bias can lead to similarities across borrower portfolios in terms of their characteristics. However, in P2P settings, this can manifest itself as familiarity *clustering*, wherein borrowers who have previously been granted certain loan types are more likely to repeat their choices, or select other categories that appear familiar to different borrowers. Herzenstein et al. (2011), studying borrower decisions on Prosper.com, identifies a 'strategic herding' behaviour, where individuals gravitate towards popular or socially-validated loans. Considering that loan popularity is directly observable through its number of bids, we assert that borrowers linked to one loan purpose are likely to connect with others sharing the same loan use. In network terms, this behaviour results in cyclical substructures where there are overlapping nodes of each of the bipartite partitions. We capture a conservative form of this cycle through the ERGM term cycle(4), formulating:

Hypothesis 3: Borrowers who share one loan purpose are likely to share another

Further, according to Amar, Ariely, Ayal, Cryder, & Rick (2011), the notion of Debt Account Aversion can also explain how investors choose debt. Specifically, investors tend to avoid holding multiple debt accounts because per Prospect Theory, borrowers mentally segregate wins and losses to minimise perceived distress from debt (Kahneman & Tversky, 1979). Ayal & Zakay (2009) expands with the Theory of Perceived Diversification, explaining that borrowers' distress increases with the perceived *distinctiveness* of each debt type, suggesting that in the P2P context, debt mentally differentiated by purpose would also be seen as distressing (Ayal & Zakay, 2009). Therefore, we assert that borrowers try to minimise loan distinctiveness by requesting loans of a single type. This can be captured by the b1degree(1) term, which models whether borrowers are only linked to one loan purpose. Therefore, we formulate:

Hypothesis 4: Borrowers exhibit preferences for minimal debt distinctiveness, sharing one loan use

Notwithstanding, credit choices relate to non-structural factors. Cooper, Gorbachev, & Luengo-Prado (2023) show that younger borrowers face constraints in the types of loans they can access in typical credit markets. The authors argue that this constraint shapes how these borrowers form loan portfolios, often concentrating around different use cases such as education, personal consumption, or small business loans. Considering the systemic differences in how age affects credit consumption, we assert that younger borrowers have different loan use portfolios. The ERGM term b1cov("age") can capture this by assessing how changes in age affect how borrowers form connections with different loan types. Therefore:

Hypothesis 5: Younger borrowers have different loan use mixes than older borrowers

Alongside age, the borrower's gender is known to be a significant factor in influencing credit use decisions. Aliano, Alnabulsi, Cestari, & Ragni (2023) study the role of gender and other factors on borrowers' probability of default on the Bondora.com, finding a persistent gender effect on different

loan uses; women tend to default less on health, home, and business loans. Croson & Gneezy (2009) shows that these differences can be due to risk perception and self-selection effects, where women exhibit greater risk aversion, leading to different borrowing patterns. We expect that these differences in risk tolerance and perceived creditworthiness by lenders affect loan use composition in terms of gender. The term `b1nodematch("gender")` captures this effect by assessing homophily in loan use selection. Our premise is supported by a negative estimate for the term, indicating heterophily. Therefore:

Hypothesis 6: Gender differences lead to different loan use mixes

Table 1: Hypotheses related to Study 2

Hypothesis	Term	Explanation
H_1^a Borrowers who share one loan purpose are likely to share another	*cycle(4)*	This captures the tendency of borrowers and loan uses to be cyclical, where people cluster around shared uses of loans.
H_2^a Borrowers exhibit preferences for minimal debt distinctiveness, sharing one loan use	*b1degree(1)*	This captures the tendency of borrowers to prefer minimal distinctiveness in their loan uses, preferring to hold only one distinct type to minimise their perceived risk.
H_3^a Younger borrowers have different loan use mixes than older borrowers	*b1cov("age")*	This captures the tendency for younger borrowers to choose different kinds of loans based on how differently they consume credit.
H_4^a Gender differences lead to different loan use mixes	*b1nodematch("gender")*	This captures the idea that there are systemic differences in what kinds of loan uses are preferred the genders due to their perceived riskiness and creditworthiness.

Following this section, we outline the methodology, explaining our dataset and network construction. To facilitate this, descriptive statistics are analysed for both the dataset and resulting network. Subsequently, we elaborate on our choice of models, discussing how network models can be used to conduct the study. Then, modelling results are shown and explained to evaluate our hypotheses. Simultaneously, the robustness of the study is evaluated through the models' diagnostics. Finally, we arrive at our conclusions, summarising the study, its results, and implications. Supporting items such as the source code and specific data processing steps are shown in section A.

2 Methodology

2.1 Dataset

The study utilises publicly-available data from Bondora, a European P2P lending platform primarily operating in the Baltics and Spain. The dataset contains detailed information on defaulted and non-defaulted loans granted to users between February 2009 and July 2021. Specifically, contained is a range of numeric, binary, categorical, and time-series attributes across 85,087 unique users and 179,235 individual loans. Since the company's API is no longer accessible, we utilised a publicly available repository compiled by Manu Siddhartha (2021) on kaggle.com. The user made a series of API calls to collect the data. Following this, the dataset was processed according to Appendix A.1.

2.2   Data Processing & Network Formation

Initially, the dataset was filtered to reduce the user-base into the most active individuals. Zhao, Liu, Wang, Ge, & Chen (2016) shows that using a sub-sample of the most active lenders on P2P platforms can produce results that model theoretical behaviour more accurately, making inferences less noisy. We extend this to borrowers, assuming that the most active users are also knowledgeable of the platform. As such, we only keep users with more than 5 loans throughout their tenure. Since we are studying interactions across two distinct set of nodes, a bipartite network was formed with unique users as the first partition and the purpose of loans as the second. This allows us to evaluate how borrowers interact with loan purposes; a one-mode projection onto users is not meaningful for P2P networks as it produces no discernible structure. Ultimately, we have 138 and 9 nodes in the first and second partitions, respectively.

2.3   Descriptive Statistics & Preliminary Analysis

The table below summarises the numeric variables between the original and cleaned datasets. First, borrowers take out relatively small loans which becomes lower after processing. The average borrower is assigned a relatively high interest rate of 25%, but this figure is 35% in the original sample. On average, users are around age 40 but this varies highly. The average loan is also relatively long, at 46 months out of a maximum 60. Figure 8 shows that most borrowers are relatively creditworthy, with few being very highly or badly rated[1]. However, borrowers are rated proportionally worse in the initial sample. The processed sample resembles the original sample in most metrics, however, the study could be biased by the fact that the sub-sample of active users are more creditworthy and perceived as less risky, which skew loan purpose connections relative to the wider population of P2P users.

Table 2: Descriptive Statistics of Numeric Variables

DataFrame	Variable	Mean	Median	Std. Dev.	Min	Max
Original Dataset	Amount	2580.07	2125	2189.33	6.39	10632
Original Dataset	Interest	34.59	30.22	24.18	7.26	264.31
Original Dataset	Age	40.58	39	12.35	0	71
Original Dataset	LoanDuration	48.24	60	14.58	1	60
Cleaned Dataset	Amount	1765.33	1100	1496.62	500	9000
Cleaned Dataset	Interest	25.04	23	12.52	9.15	143.63
Cleaned Dataset	Age	39.35	38	10.6	21	70
Cleaned Dataset	LoanDuration	45.96	60	17.19	3	60

Table 3 shows a network density of 0.025, indicating that only 2.5% of all possible borrower–loan-type connections exist. This low density suggests that borrowers typically engage with few loan types, reflecting specialization in borrowing behaviour. The centralization value of 0.512 further indicates an uneven distribution across loan types, where a few popular categories attract many

---

[1]Bondora uses their own proprietary credit rating system where the best rating is AA and worst is HR (Bondora, 2024). The company bases these ratings on their calculated expected probability of loss on the loan. For example, someone rated AA ranges from a expected loss of 0% to 2%, whereas for HR this is 25% - >25%.
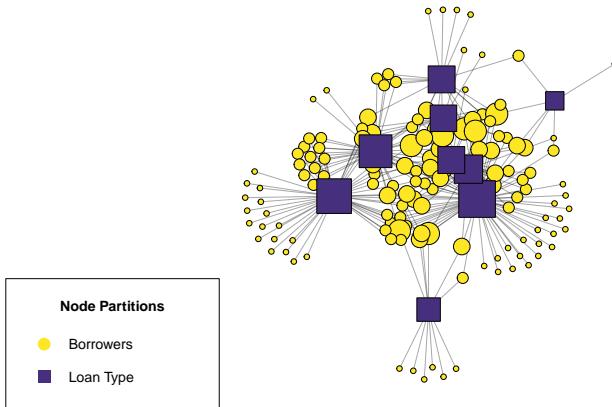
borrowers while most remain peripheral. The mean distance of approximately three implies that borrowers are closely connected, typically separated by only two loan-use steps.

Centrality distributions reinforce this structure Figure 2. Betweenness is highest among loan-type nodes, though several borrowers also bridge distinct loan clusters. Closeness centrality remains low but scattered, suggesting localized clustering constrained by the bipartite structure. Degree distributions show wide variation among loan types, consistent with a hub-like structure dominated by a few popular uses. Finally, the homogeneous eccentricity distribution indicates that no borrowers are highly isolated, reflecting a compact and moderately cohesive network overall.

Table 3: Bipartite Network Descriptive Statistics
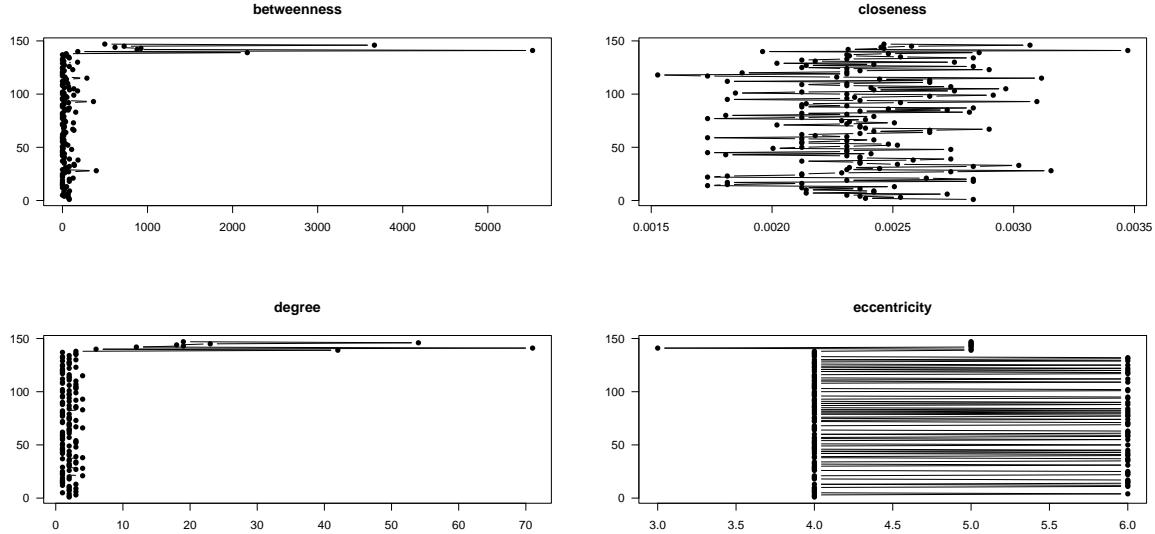
Statistic	Partition 1	Partition 2	Overall Network
Vertex Count	138	9	147
Edge Count	-	-	264
Density	-	-	0.025
Mean Distance	-	-	2.952
Reciprocity	-	-	1.000
Centralization	-	-	0.512

Figure 1: Network Visualisation

Note: Each colour represents a different partition of the network. The partitions' nodes are scaled to their relative degrees. Users with many different loan types are larger than those with single loan uses. Similarly, the size of second partition nodes represents how many users belong to each.

Figure 2: Network Centrality Distribution Plots

3 Research Rationale

Our overall study analyses borrowers' behaviours, however, the studies are differentiated in their scope; the first study observes dyad-level relations among borrowers to understand how shared attributes affect borrowers' behaviours. This is inherently useful because behavioural finance aims to understand how individuals behave in *aggregate* (Ayal et al., 2018). Typically, the Quadratic Assignment Procedure is selected for dyad-level observations because basic linear regressions such as Ordinary Least Squares are fundamentally flawed and biased when using non-independent observations (Simpson, 2001).

Recognising this, studies such as Zuo (n.d.) assume that groups of borrowers have dependencies, and accordingly, use clustering methods such as K-Means or Fuzzy Clustering to understand behaviour. However, these methods are unsuitable when we require a distribution of estimates to make inferences from our results. Moreover, studying individuals as dyads rather than large clusters provides finer granularity in results. Therefore, QAP regression is the most appropriate choice for this study: it allows us to control for the non-independence among dyads, derive a statistical distribution of estimates through matrix permutation, and draw valid inferences about the relationship between borrower similarities and loan-use behaviour. This makes it both theoretically and methodologically well-suited to our research objectives and dataset.

The second study extends the analysis from a dyadic similarities to the structural formation of borrower-to-loan-use connections, which can identify the behavioural mechanisms that influence borrowers' specific loan use mixes. While QAP can account for exogenous influences, it cannot analyse higher-order network patterns such as clustering while accounting for simultaneous inter-actions between borrowers and loan uses. Using the Exponential Random Graph Model allows us to analyse how local behavioural tendencies aggregates to a global network structure.

Recent evidence highlights the need to focus on structural approaches in P2P lending. Liu, Baals,

Osterrieder, & Hadji-Misheva ([2024](#)) studies borrowers' centrality with the Bondora P2P dataset, modelling edges through borrower-to-borrower similarity metrics. Ultimately, they find that a borrower's position is highly significant for their probability of default, using centrality as an exogenous variable in a logistic regression. In contrast, ERGM allows us to move away from predictive association and instead account for endogenous factors, allowing us to understand the likelihood that borrowers' selection of loan uses are random. Consequently, we can better understand how behavioural phenomena such as familiarity and debt account aversion can translate into *systemic* patterns of credit use.

As we aim to understand how two distinct groups of nodes interact, a bipartite approach is highly suitable. Specifically, Stivala, Wang, & Lomi ([2025](#)) shows that in bipartite networks, terms such as `cycle(4)` can be effective in capturing clustering as in hypothesis 3, whereas it would fail in one-mode projected networks. Furthermore, the flexibility of ERGM also allows for modelling exogenous impacts such as heterophily in gender-based loan use mixes, as in hypothesis 6, and covariate age effects as in hypothesis 5.

A   Supplements

A.1   Data Preprocessing Steps

This section outlines the specific steps taken to process the data into the network object that was used in later analysis, outlining the rationale, where needed.

- Following the data import, only the following attributes were kept to make our processing steps more focused

```r
keep_cols <- c("LoanId", "UserName","Age", "Gender",
               "Country", "Amount", "Interest","LoanDuration",
               "UseOfLoan", "Rating", "Restructured", "MonthlyPayment",
               "OccupationArea")
```

- We removed rows with any `NA` values for the selected attributes to ensure that we have a complete dataset

- To ensure that our study is focused around active and possibly knowledgeable users of the platform, we kept only the loans from users with 5+ granted loans

- When adding labels to each loan use and occupation area, we keep any possible missing values because these do not indicate lower quality data. Instead, they can carry information about lenders distribute loans according to the quality of information presented by borrowers (Yao et al., 2019). Additionally, keeping missing labels is important to ensure that the edges are not superficially limited and that nodes can be isolates if they are indeed that way in reality.

- To create the network object for ERGM models:

    - We first create an incidence matrix of size $n\ \text{\OE}\ m$, borrower usernames by the loan use, where the values are the number of loans the user obtained for each loan use category. These values are then turned into binary because our modelling does not support the use of weighted edges.

    - A bipartite network is created from this incidence matrix, where the first partition are unique users, and the second partition is the loan use associated among all loans of the user. Users can share more than more loan use type.

    - We add age and gender attributes to each node of the first partition

- To create the relevant networks/matrices for QAP Regressions:

    - We first re-create the loan use incidence matrix, however, we keep the counts as the linear regression supports this.

        * We then convert the loan use incidence matrix into an adjacency matrix of shape $n\,\text{\OE}\,n$ through the transformation $X X^{\mathrm{T}}$, which results in a matrix of shared loan uses weighted by the frequency of each user's loan count within each loan use category.

        * We then set the diagonal values of X to be zero to ensure that there are no loops within the network.

    - In accordance with this procedure, we create adjacency matrices for Credit Rating and Occupation Area which will be used as the main predictors in the QAP models.

* Credit Rating is a weighted adjacency matrix based on the loan count belonging to each user for each credit rating category

* Occupation Area is a binary adjacency matrix based on which occupation each user has reported to have held in the past when applying for each of their loans.

– Additionally, we create a set of control variables to improve the validity and performance of the linear regression using Loan Amounts, Age, Gender, Loan Duration, and whether the loans have been restructured.

* Loan Amounts is a weighted adjacency matrix based on five bins across the range of possible loan amounts

* Age is an adjacency matrix based on differences in users' ages

* Gender is a binary adjacency matrix based on users' shared gender.

* Loan Duration is an adjacency matrix based on the differences in users' loans average durations

* Restructured is a binary adjacency matrix based on whether the users share the fact that they have defaulted on any loans in the past. The users can either have shared both defaults and non-defaults, or both.

A.2   Distributions of Variables across Samples

Figure 3: Distribution of Loan Amount across Samples
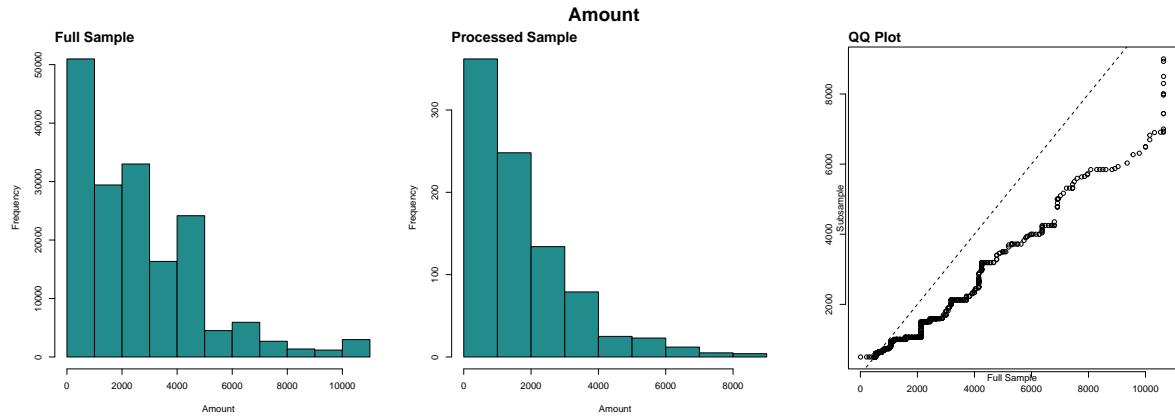
Figure 4: Distribution of Interest Rates across Samples
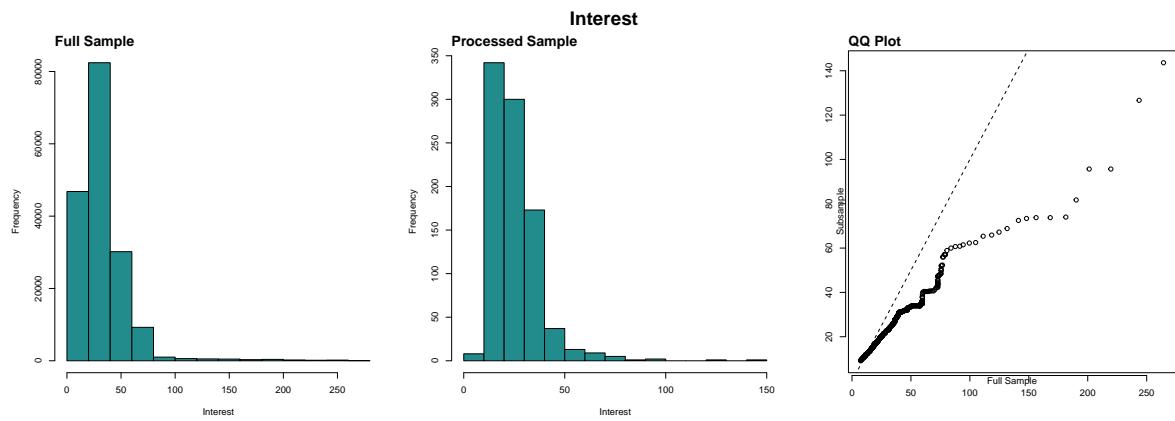
Figure 5: Distribution of Age across Samples
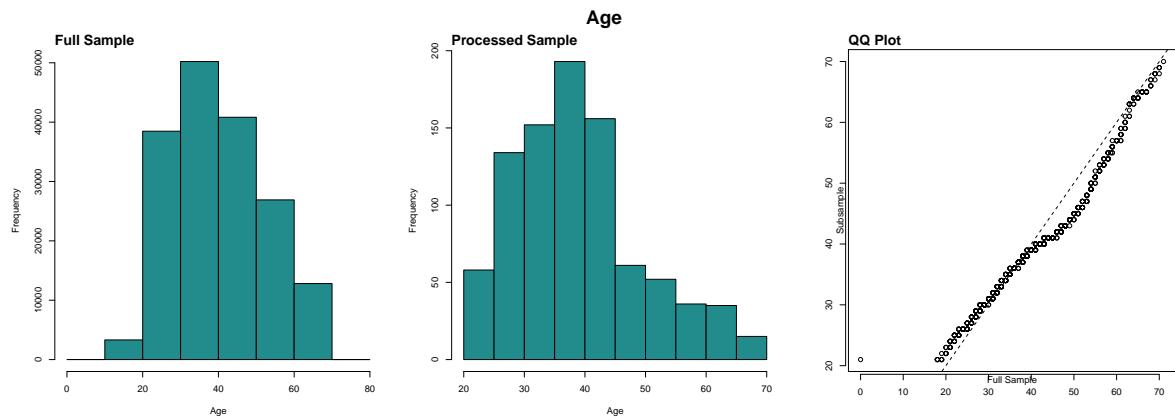
Figure 6: Distribution of Loan Duration across Samples

Loan Duration
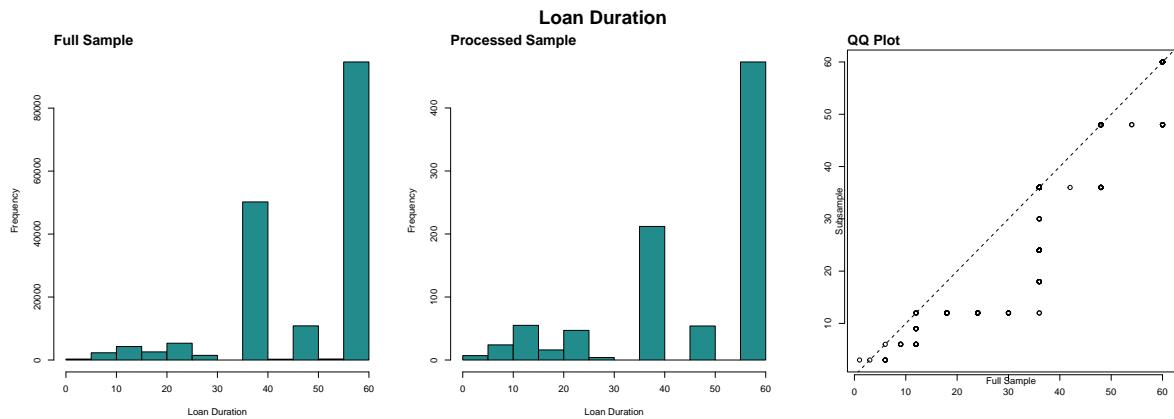
Figure 7: Distribution of Loan Purpose across Samples

Loan Purpose
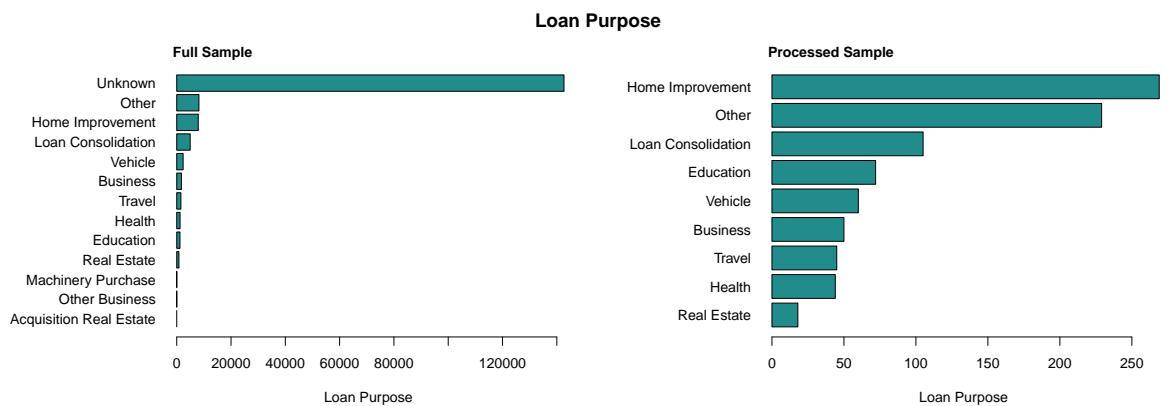
Figure 8: Distribution of Credit Ratings across Samples

Credit Rating
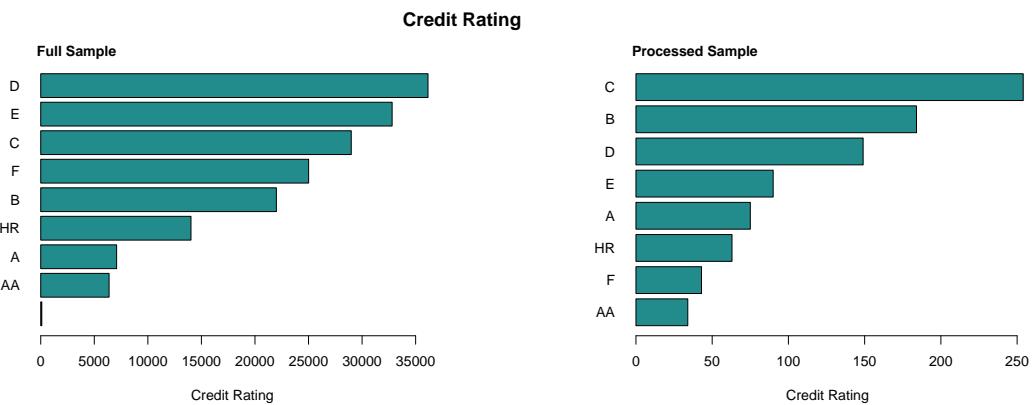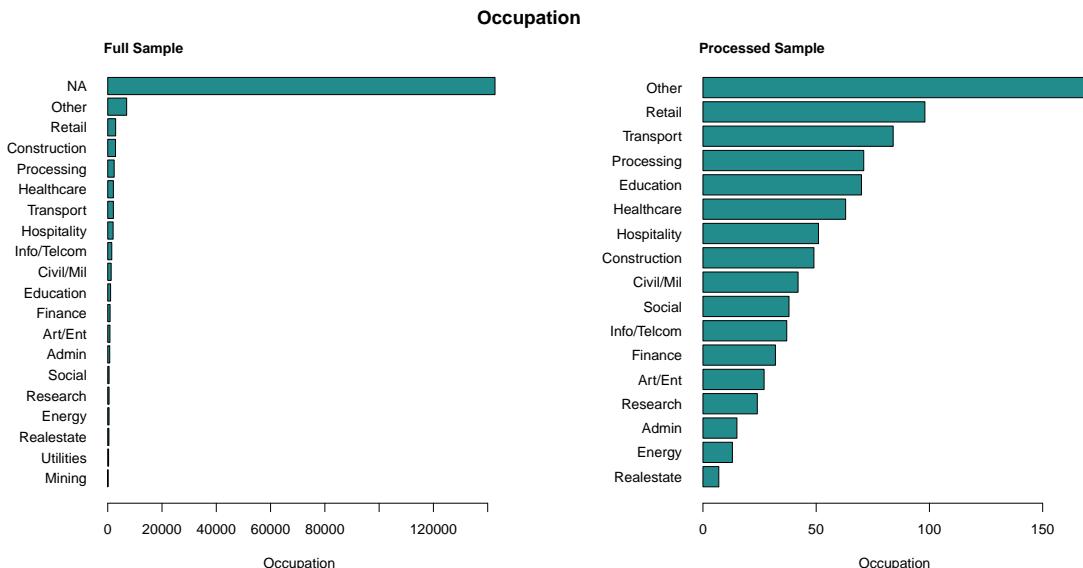
Figure 9: Distribution of Occupation Types across Samples

Occupation

A.3 Source Code - Data Preprocessing

```r
# ----------------------------------------------------------------------------------- #
install.packages("viridis") # For Colours
install.packages("here") # To locate files from RProj
# ----------------------------------------------------------------------------------- #
# Import the Bondora P2P Dataset
obj_paths = "resources/objects/"
bondora_raw <- read.csv("dataset/LoanData_Bondora.csv")
raw_cols <- colnames(bondora_raw)
# ----------------------------------------------------------------------------------- #
# Select Columns to Keep
keep_cols <- c("LoanId", "UserName","Age", "Gender",
               "Country", "Amount", "Interest","LoanDuration",
               "UseOfLoan", "Rating", "Restructured", "MonthlyPayment",
               "OccupationArea")

bondora <- bondora_raw[keep_cols]

# Remove Rows with any NAs -> Complete Dataset Preferred
print(paste("NA Count |",sum(is.na(bondora)),"rows"))
bondora <- na.omit(bondora)
print(paste("NA Count |",sum(is.na(bondora)),"rows"))

# Remove users with only -1 Stated UseofLoan
# NOTE: STEP REMOVED TO PRESERVE SOME GENUINE SPARSITY
#bondora_clean <- bondora[bondora$UseOfLoan != -1, ]

# Remove Users with only One Loan
user_counts <- table(bondora_clean$UserName)
multi_users <- names(user_counts[user_counts > 5])
bondora_clean <- bondora_clean[bondora_clean$UserName %in% multi_users, ]
#bondora_test <- bondora_raw[bondora_raw$UserName %in% multi_users, ]

# See if Ratings are Properly Encoded
unique(bondora_clean$Rating)

# See distribution of UserName Counts
hist(table(bondora_clean$UserName))
barplot(table(bondora_clean$UseOfLoan))

# Extract UseofLoan Types and Turn into Factor
bondora_clean$UseOfLoan_factor <- as.factor(bondora_clean$UseOfLoan)
unique(bondora_clean$UseOfLoan_factor)

loan_use_labels <- c(
  "-1" = "NA",
  "0" ="Loan Consolidation",
  "1" = "Real Estate",
  "2" = "Home Improvement",
  "3" = "Business",
  "4" = "Education",
  "5" = "Travel",
  "6" = "Vehicle",
  "7" = "Other",
  "8" = "Health",
  "110" = "Other Business",
  "102" = "Undefined Business",
  "108" = "Undefined Business"
)
# Change Labels for Cleaned Dataset
bondora_clean$UseOfLoan_factor <- loan_use_labels[
  as.character(bondora_clean$UseOfLoan)]

# Change Labels for Uncleaned Dataset
bondora$UseOfLoan_factor <- loan_use_labels[
  as.character(bondora$UseOfLoan)]
```

```r
66
67 # Add labels to the OccupationArea Variable
68 levels(as.factor(bondora_clean$OccupationArea)) # view codes
69
70 occupation_labels <- c(
71   "-1" = "NA",
72   "1" = "Other",
73   "2" = "Mining",
74   "3" = "Processing",
75   "4" = "Energy",
76   "5" = "Utilities",
77   "6" = "Construction",
78   "7" = "Retail",
79   "8" = "Transport",
80   "9" = "Hospitality",
81   "10" = "Info/Telcom",
82   "11" = "Finance",
83   "12" = "Realestate",
84   "13" = "Research",
85   "14" = "Admin",
86   "15" = "Civil/Mil",
87   "16" = "Education",
88   "17" = "Healthcare",
89   "18" = "Social",
90   "19" = "Art/Ent",
91   "20" = "Agriculture",
92   "21" = "Forestry/Fish"
93 )
94 # store original
95 bondora_clean$occupation_code <- bondora_clean$OccupationArea
96 bondora_clean$occupation_label <- occupation_labels[
97   as.character(bondora_clean$OccupationArea)]
98
99 bondora$occupation_code <- bondora$OccupationArea
100 bondora$occupation_label <- occupation_labels[
101   as.character(bondora$OccupationArea)]
102 # ----------------------------------------------------------------------------- #
103 # Observe Descriptive Statistics
104
105 cols <- viridis::viridis(30)
106
107 # Make function to save plots
108 save_plot <- function(plt_nam) {
109   plt <- recordPlot()
110   saveRDS(plt, here::here("resources","objects","preprocessing",
111                           paste0(plt_nam,".Rds")))
112 }
113
114 # Make function to consistently plot comparisons
115 plot_desc_hists <- function(df1, df2, col_name, type) {
116
117   par(mfrow=c(1,3))
118
119   hist(df1[[col_name]], xlab=type, col=cols[15], main="", breaks=10)
120   mtext("Full Sample", side=3, adj=0, line=0.25, cex=1, font=2)
121
122   hist(df2[[col_name]], xlab=type, col=cols[15], main="", breaks=10)
123   mtext("Processed Sample", side=3, adj=0, line=0.25, cex=1, font=2)
124
125   qqplot(df1[[col_name]], df2[[col_name]], main="", cex=1,
126          xlab="Full Sample", ylab="Subsample", line=0.25)
127   abline(0, 1, lty=2)
128
129   mtext("QQ Plot", side=3, adj=0, line=0.25, cex=1, font=2)
130
131   mtext(type, outer = TRUE, line = -2, side=3, cex = 1.3, font = 2)
```

```r
132
133    # Reset plot window
134    par(mfrow=c(1,1), mar=c(5,4,4,2)+0.1)
135  }
136
137  plot_desc_bar <- function(df1, df2, col_name, type) {
138
139    par(mfrow=c(1,2), mar=c(5,10,4,2))
140
141    barplot(sort(table(df1[[col_name]]), decreasing = F),
142            xlab=type, col=cols[15], horiz=TRUE, las=1)
143    mtext("Full Sample", side=3, adj=0, line=0.25, cex=1, font=2)
144
145    barplot(sort(table(df2[[col_name]]), decreasing = F),
146            xlab=type, col=cols[15], horiz=TRUE, las=1)
147    mtext("Processed Sample", side=3, adj=0, line=0.25, cex=1, font=2)
148
149    mtext(type, outer = TRUE, line = -2, side=3, cex = 1.3, font = 2)
150
151    # Reset plot window
152    par(mfrow=c(1,1), mar=c(5,4,4,2)+0.1)
153  }
154
155  plot_desc_hists(bondora, bondora_clean, "Amount", "Amount")
156  save_plot("hist_amt")
157
158  plot_desc_hists(bondora, bondora_clean, "Interest", "Interest")
159  save_plot("hist_int")
160
161  plot_desc_hists(bondora, bondora_clean, "LoanDuration", "Loan Duration")
162  save_plot("hist_loandur")
163
164  plot_desc_hists(bondora, bondora_clean, "MonthlyPayment", "Monthly Payment")
165  save_plot("hist_monpmt")
166
167  plot_desc_hists(bondora, bondora_clean, "Age", "Age")
168  save_plot("hist_age")
169
170  plot_desc_bar(bondora, bondora_clean, "UseOfLoan_factor", "Loan Purpose")
171  save_plot("bar_loanuse")
172
173  plot_desc_bar(bondora, bondora_clean, "Rating", "Credit Rating")
174  save_plot("bar_rating")
175
176  plot_desc_bar(bondora, bondora_clean, "occupation_label", "Occupation")
177  save_plot("bar_occupation")
178
179  # Get Tabular Summary Statistics
180  tab_comps <- function(df1, df2, cols) {
181    stats <- c("Mean"=mean, "Median"=median, "Std. Dev."=sd, "Min"=min, "Max"=max)
182
183    get_stats <- function(d) {
184      t(sapply(d[cols], function(x)
185        sapply(stats, function(f) round(f(x, na.rm=TRUE), 2))
186      ))
187    }
188
189    df1_stats <- get_stats(df1)
190    df2_stats <- get_stats(df2)
191
192    out <- rbind(
193      cbind(DataFrame = "Original Dataset",
194            Variable = rownames(df1_stats), df1_stats),
195      cbind(DataFrame = "Cleaned Dataset",
196            Variable = rownames(df2_stats), df2_stats)
197    )
```

```r
198     rownames(out) <- NULL
199     as.data.frame(out)
200 }
201
202 tab_results <- tab_comps(bondora, bondora_slim,
203                          c("Amount","Interest","Age","LoanDuration"))
204
205 knitr::kable(tab_results)
206
207 # Save table for use in the Report
208 saveRDS(tab_results,
209         file=paste0(obj_paths,"preprocessing/","summary_table.Rds"))
210 # ---------------------------------------------------------------------------- #
211 # Convert Dataset into Incidence Matrix to form Network Object (for ERGM)
212 bondora_slim <- bondora_clean
213
214 # Create the Incidence Matrix for Use of Loan
215 bondora_matrix <- table(
216     bondora_slim$UserName, bondora_slim$UseOfLoan)
217 bondora_matrix[bondora_matrix > 0] <- 1 # Given that ergm.counts fails with GOF
218
219 # Create network object with counts as Edge attribute
220 bondora_net <- network::network(
221     bondora_matrix, directed=FALSE, bipartite=nrow(bondora_matrix),
222     ignore.eval = FALSE, names.eval="frequency", loops=FALSE)
223
224 # Set the bipartite Attribute - UNNCESSARY GIVEN bipartite=length(n)
225 len <- dim(bondora_matrix)[1]
226 len_b2 <- dim(bondora_matrix)[2]
227 b_indicator <- c(rep(1,len),rep(2,len_b2))
228 #network::set.vertex.attribute(
229 #   bondora_net, "bipartite", value = rep(len,len), v=1:len
230 #)
231
232 # Extract Partition 2 Labels
233 loan_use <- levels(bondora_clean$UseOfLoan_factor)
234
235 # Create Loan Type Attribute for Partition 2
236 b2_loantype <- rep(NA, len)
237 b2_loantype <- c(b2_loantype, loan_use)
238
239 if (length(b2_loantype) == network::network.size(bondora_net)) {
240     network::set.vertex.attribute(
241         bondora_net, "b2_loantype", value = b2_loantype)
242     }
243
244 # Add Age Vertex Attribute to B1
245 age <- bondora_clean$Age[match(
246     rownames(bondora_matrix), bondora_clean$UserName)]
247 b1_age <- c(age, rep(NA, len_b2))
248 network::set.vertex.attribute(
249     bondora_net, "b1_age", value=b1_age
250 )
251
252 # Add Gender Vertex Attribute to B1
253 gender <- bondora_clean$Gender[match(
254     rownames(bondora_matrix), bondora_clean$UserName)]
255 unique(gender) # Check to see if encoded properly
256 gender <- ifelse(gender == 0, "male","female")
257 b1_gender <- c(gender, rep(NA, len_b2))
258 network::set.vertex.attribute(
259     bondora_net, "b1_gender", value = b1_gender
260 )
261
262 # Save the network and data frame object
263 saveRDS(bondora_slim, file=paste0(obj_paths,"preprocessing/","bondora_df.Rds"))
```

```r
264  saveRDS(bondora_net, file=paste0(obj_paths,"preprocessing/","bondora_net.Rds"))
265  # --------------------------------------------------------------------------- #
266  # First, create Incidence Matrix again but with Frequency Counts
267  loan_use_matrix <- table(
268    bondora_slim$UserName, bondora_slim$UseOfLoan)
269
270  # Get the Adjacency Matrix for Loan Use Similarity (Dependent QAP Variable)
271  adj_mat_loan_use <- loan_use_matrix %*% t(loan_use_matrix)
272  # Remove self weights to remove any loops
273  diag(adj_mat_loan_use) <- 0
274
275  # Get the Adjacency Matrix for Credit Rating Similarity (Predictor in QAP)
276  incidence_rating <- table(bondora_slim$UserName, bondora_slim$Rating)
277  adj_mat_rating <- incidence_rating %*% t(incidence_rating)
278  diag(adj_mat_rating) <- 0
279
280  # Get Adjacency Matrix for Occupation Similarity (Predictor in QAP, Binary)
281  incidence_occupation <- table(bondora_slim$UserName,
282                                bondora_slim$occupation_label)
283  adj_mat_occupation <- incidence_occupation %*% t(incidence_occupation)
284  adj_mat_occupation[adj_mat_occupation > 0] <- 1
285  diag(adj_mat_occupation) <- 0
286
287  # Get the Adjacency Matrix for Loan Amount (Control in QAP) - BINARY
288  # First, bin the Loan Amounts
289  bondora_slim$Amount_bins <- cut(
290    bondora_slim$Amount, breaks=c(0,2000,4000,6000,8000,10000),
291    labels = c(1:5)
292  )
293  incidence_amount_bins <- table(bondora_slim$UserName, bondora_slim$Amount_bins)
294  adj_mat_amount_bins <- incidence_amount_bins %*% t(incidence_amount_bins)
295  diag(adj_mat_amount_bins) <- 0
296
297  # Continous Absolute Difference Approach
298  avg_amount <- tapply(bondora_slim$Amount, bondora_slim$UserName, mean)
299  adj_mat_amount_diff <- outer(avg_amount, avg_amount,
300                               FUN = function(x,y) abs(x - y))
301  diag(adj_mat_amount_diff) <- 0
302
303  # Get Matrix for Differences in Age
304  incidence_age <- table(bondora_slim$UserName, bondora_slim$Age)
305  borrower_ages <- as.numeric(colnames(incidence_age)[max.col(incidence_age)])
306  names(borrower_ages) <- rownames(incidence_age)
307  adj_mat_age <- outer(borrower_ages, borrower_ages,
308                       FUN = function(x, y) abs(x - y))
309  rownames(adj_mat_age) <- colnames(adj_mat_age) <- names(borrower_ages)
310
311  # Get Adjacency Matrix for (same) Gender
312  incidence_gender <- table(bondora_slim$UserName, bondora_slim$Gender)
313  adj_mat_gender <- incidence_gender %*% t(incidence_gender)
314  # Make the matrix binary for homophily
315  adj_mat_gender <- ifelse(adj_mat_gender > 0, 1, 0)
316  diag(adj_mat_gender) <- 0
317
318  # Get Adjacency Matrix for Differences in Average Loan Duration
319  borrower_loandur <- sapply(tapply(bondora_slim$LoanDuration,
320                                    bondora_slim$UserName, unique),
321                             mean)
322  adj_mat_loandur_diff <- outer(borrower_loandur, borrower_loandur,
323                                FUN=function(x,y) abs(x-y))
324  diag(adj_mat_loandur_diff) <- 0
325
326  # Get Adjacency Matrix for Homophily in Restructure of Loans
327  incidence_restructure <- table(bondora_slim$UserName, bondora_slim$Restructured)
328  adj_mat_rest <- incidence_restructure %*% t(incidence_restructure)
329  diag(adj_mat_rest) <- 0
```

```
330
331 # Save the objects for the QAP Regression in different Script
332 qap_paths = paste0(obj_paths,"/qap/")
333 saveRDS(b_indicator, file=paste0(
334   "resources/objects/preprocessing/indicator.Rds"))
335
336 saveRDS(adj_mat_loan_use, file=paste0(qap_paths,"adj_mat_loanuse.Rds"))
337 saveRDS(adj_mat_occupation, file=paste0(qap_paths,"adj_mat_occup.Rds"))
338 saveRDS(adj_mat_rating, file=paste0(qap_paths,"adj_mat_rating.Rds"))
339 saveRDS(adj_mat_amount_diff, file=paste0(qap_paths,"adj_mat_amtdiffs.Rds"))
340 saveRDS(adj_mat_age, file=paste0(qap_paths,"adj_mat_agediffs.Rds"))
341 saveRDS(adj_mat_gender, file=paste0(qap_paths,"adj_mat_gender.Rds"))
342 saveRDS(adj_mat_loandur_diff, file=paste0(qap_paths,"adj_mat_loandurdiffs.Rds"))
343 saveRDS(adj_mat_rest, file=paste0(qap_paths,"adj_mat_rest.Rds"))
344 # ------------------------------------------------------------------------ #
```

scripts/bondora_preprocessing.R

A.4 Source Code - QAP Linear Regression

```r
# ------------------------------------------------------------------------------ #
# NOTE: NOT THE LATEST QAP ANALYSIS!

# If not already Installed
install.packages("viridis") # For Colours
install.packages("here")    # To locate files from RProj

# Set colour palette
cols <- viridis::viridis(30)
# ------------------------------------------------------------------------------ #
# Load Relevant Files
qap_path="resources/objects/qap/"

loan_use_mat <- readRDS(paste0(qap_path,"adj_mat_loanuse.RDS"))
rating_mat <- readRDS(paste0(qap_path,"adj_mat_rating.RDS"))
amt_diffs_mat <- readRDS(paste0(qap_path,"adj_mat_amtdiffs.RDS"))
age_diffs_mat <- readRDS(paste0(qap_path,"adj_mat_agediffs.RDS"))
gender_mat <- readRDS(paste0(qap_path,"adj_mat_gender.RDS"))
loandur_diffs_mat <- readRDS(paste0(qap_path,"adj_mat_loandurdiffs.RDS"))
rest_mat <- readRDS(paste0(qap_path,"adj_mat_rest.RDS"))
occup_mat <- readRDS(paste0(qap_path,"adj_mat_occup.Rds"))
# ------------------------------------------------------------------------------ #
# Create function to determine significance from t-value statistic
t_to_stars <- function(t) {
    stars <- rep("", length(t))
    stars[abs(t) >= 1.96]   <- "*"
    stars[abs(t) >= 2.576] <- "**"
    stars[abs(t) >= 3.291] <- "***"

    return(stars)
}

# Make function to save plots
save_qap_plot <- function(plt_nam) {
    plt <- recordPlot()
    saveRDS(plt, here::here("resources","objects","qap",
                            paste0(plt_nam,".Rds")))
}

var_names <- c("Intercept", "Rating", "Occupation",
               "Loan Amount", "Age","Gender", "Loan Duration", "Restructured")
pred_vars <- list(rating_mat, occup_mat, amt_diffs_mat, age_diffs_mat,
                  gender_mat, loandur_diffs_mat, rest_mat)
# ------------------------------------------------------------------------------ #
# Basic QAP Linear Regression
qap_m1 <- sna::netlm(y = loan_use_mat,
                     x = list(rating_mat, amt_diffs_mat, age_diffs_mat,
                              gender_mat, loandur_diffs_mat, rest_mat),
                     nullhyp = "qapspp", reps = 2500)
qap_m1$names <- var_names
summary(qap_m1)

results_m1 <- qap_m1$coefficients
names(results_m1) <- var_names
results_sig_m1 <- paste(round(results_m1,3), t_to_stars(qap_m1$tstat))

# Plot Residuals
hist(qap_m1$residuals, main="QAP Residuals", col = cols[15])

# Save Model
saveRDS(qap_m1, file = paste0(qap_path,"qap_m1.RDS"))
# ------------------------------------------------------------------------------ #
# Standardised QAP Linear Regression
scaled_dep <- scale(loan_use_mat)
scaled_pred <- lapply(pred_vars, scale)
```

```R
66
67  qap_m2 <- sna::netlm(y = scaled_dep,
68                       x = scaled_pred,
69                       nullhyp = "qapspp", reps = 2500)
70  qap_m2$names <- var_names
71  summary(qap_m2)
72
73  # Plot the result
74  results_m2 <- qap_m2$coefficients
75  names(results_m2) <- var_names
76  results_sig_m2 <- paste(round(results_m2,3), t_to_stars(qap_m2$tstat))
77
78  # Plot Residuals
79  hist(qap_m2$residuals, main="QAP Residuals", col = cols[15])
80
81  # Save Model
82  saveRDS(qap_m2, file = paste0(qap_path,"qap_m2.RDS"))
83  # ------------------------------------------------------------------------------ #
84  # Plot the result
85  par(mfrow=c(1,2))
86
87  qap_plot_m1 <- barplot(results_m1, col = cols[15],  border = cols[10],
88                         ylim = c(min(results_m1) + min(results_m1)*0.15,
89                                  max(results_m1) + max(results_m1)*0.15),
90                         main="QAP Model Results (Unstandardised)")
91  text(x = qap_plot_m1,
92       y = results_m1 + sign(results_m1)*(0.075*diff(range(results_m1))),
93       labels = results_sig_m1, font = 2)
94
95
96  qap_plot_m2 <- barplot(results_m2, col = cols[15],  border = cols[10],
97                         ylim = c(min(results_m2) + min(results_m2)*0.15,
98                                  max(results_m2) + max(results_m2)*0.15),
99                         main="QAP Model Results (Standardised)")
100 text(x = qap_plot_m2,
101      y = results_m2 + sign(results_m2)*(0.075*diff(range(results_m2))),
102      labels = results_sig_m2, font = 2)
103
104 save_qap_plot("unstd_std_plot")
105
106 par(mfrow=c(1,1))
```

scripts/qap_network_analysis.R

A.5 Source Code - ERGM Network Analysis

```r
# ------------------------------------------------------------------------------ #
# NOTE: NOT THE LATEST ERGM ANALYSIS! FOR REFERENCE ONLY

# If not already Installed
install.packages("viridis")       # For Colours
install.packages("Rglpk")         # Additional solver for ERGMs
install.packages("here")          # To locate files from RProj

# Import the Network and Other Object
ergm_path <- "resources/objects/ergm/"
bondora_net <- readRDS(here::here(
  "resources","objects","preprocessing","bondora_net.Rds"))
b_indicator <- readRDS(here::here(
  "resources","objects","preprocessing","indicator.Rds"))

# Set colour palette
cols <- viridis::viridis(30)

# Determine acceptable core count
n_cores <- parallel::detectCores() - 3 # Leave some out for other processes
print(paste("You have",n_cores,"usable cores"))

# Repeatability
seed(42)

# Save plots
save_ergm_plot <- function(plt_nam) {
  plt <- recordPlot()
  saveRDS(plt, here::here("resources","objects","ergm",
                          paste0(plt_nam,".Rds")))
}
# ------------------------------------------------------------------------------ #
# Copy network for plotting
bondora_plot <- bondora_net

# Get node type for plotting
type_indicator <- ifelse(b_indicator == 2, TRUE,FALSE)
shape <- ifelse(type_indicator,"square","circle")
network::set.vertex.attribute(bondora_plot, "shape", shape)

# Get Category Count for Vertex Size
counts <- sna::degree(bondora_plot)
counts_att <- ifelse(type_indicator, log(counts)*4, counts*1.5)
network::set.vertex.attribute(bondora_plot, "size", counts_att)

# Colours for the Node Types
plot_cols <- ifelse(type_indicator, cols[5], cols[30])
network::set.vertex.attribute(bondora_plot, "color", plot_cols)

# Legend Plotting
type_legend <- ifelse(type_indicator, "Borrowers", "Loan Type")
type_legend <- as.factor(type_legend)

# Plot the Network
plot(snafun::to_igraph(bondora_plot),
     #main = "Bipartite User-LoanUse",
     edge.arrow.size = 0.3,
     edge.color = rgb(0,0,0, alpha = 0.35),
     vertex.frame.color = "black",
     vertex.label = NA,
     vertex.frame.size = 3,
     edge.curved = FALSE,
     layout=igraph::layout.fruchterman.reingold)
legend("bottomleft",
       legend = levels(type_legend),
```

```r
66          inset = c(0.15,  0.01),
67          col = c( cols [30],  cols [5]),
68          pch = c(16,  15),
69          title = "Node  Partitions",
70          title.font = 2,
71          cex = 1,                    # Increase  the  text  size
72          pt.cex = 2,                 # Increase  the  point  symbol  size
73          box.lwd = 1,                # Thin  box  border
74          box.col = "black",          # Box  color
75          bty = "o"                   # Use  a  box  around  legend
76  )
77  save_ergm_plot("network_plot")
78
79  # Summary  Statistics
80  snafun::g_density(bondora_net)
81  snafun::g_centralize(bondora_net)
82
83  # Plot  Network  Summary  Statistics
84  snafun::plot_centralities(bondora_net)
85  save_ergm_plot("network_plots")
86  # -------------------------------------------------------------------------------- #
87  # Make  function  to  calculate  probabilities  from  log  odds
88  lodds_to_prob <- function(l_odd) {
89    return(exp(l_odd) / (1 + exp(l_odd)))
90  }
91  # Make  function  to  save  ERGM  object
92  save_ergm <- function(object, id) {
93    saveRDS(object, file=here::here(
94      "resources","objects","ergm",id,".Rds"))
95  }
96  # Make  function  to  conduct  ERGMs  automatically
97  auto_ergm <- function(model, mcmc, name) {
98
99    # Conducts  the  GOF  Diagnostics  and  then  saves  the  model,
100   # mcmc  diagnostics  and  gof  object  in  a  list.
101   # This  list  can  be  imported  as  an  .RDS  object  into  the  R  environment
102
103   # Diagnostics
104   if (mcmc) {
105     ergm::mcmc.diagnostics(model)
106   }
107
108   # The  GOF  must  be  adjusted  otherwise  it  takes  too  long
109   # We  do  not  limit  the  GOF  by  changing  its  range  of  parameters
110   gof <- ergm::gof(model,
111                    control = ergm::control.gof.ergm(
112                      nsim = 200,
113                      MCMC.burnin = 5000,
114                      MCMC.interval = 1000,
115                      parallel = n_cores,
116                      parallel.type = "PSOCK"
117                    ))
118
119   # Return  List  to  view  each  item  separately
120   result <- list(model, gof)
121   names(result) <- c("model","gof")
122   save_ergm(result, paste0(name,"_panel"))
123
124   return(result)
125 }
126 # -------------------------------------------------------------------------------- #
127 # Find  max  degree
128 (max_deg <- max(summary(bondora_net ~ b2factor("b2_loantype"))))
129 # -------------------------------------------------------------------------------- #
130 # Base  Model  + GOF
131 formula_base_model <- bondora_net ~ edges
```

```r
132  base_ergm <- ergm::ergm(formula_base_model)
133  base_ergm_panel <- auto_ergm(base_ergm, mcmc = FALSE, name = "ergm_base")
134  snafun::stat_plot_gof(base_ergm_panel$gof)
135  models = list(base_ergm)
136  texreg::screenreg(models)
137  # ------------------------------------------------------------------------------ #
138  # Base Model + Edge Counts + GOF
139  #base_model_counts <- ergm::ergm(bondora_net ~ edges, response="frequency",
140  #                                 reference = ~ Poisson)
141  #basemodel_counts_gof <- ergm::gof(base_model_counts)
142  #snafun::stat_plot_gof(basemodel_counts_gof)
143  #
144  #texreg::screenreg(list(base_model, base_model_counts))
145  # ------------------------------------------------------------------------------ #
146  # Iteration 1 + MCMC Diagnostics + GOF
147  model_1_params <- bondora_net ~ edges +
148    # b1 decay can be very low since 9 b2
149    gwb1degree(decay=0.15, fixed=TRUE)
150
151  model_1 <- ergm::ergm(
152    model_1_params,
153
154    # Max b2 degree is 72, so this constraint is reasonable
155    # and helps convergence significantly.
156    # Technically in the Bondora population this can be
157    # far higher but we are studying a subsample.
158    #constraints = ~ bd(minout = 0, maxout = 80),
159
160    control = ergm::control.ergm(
161      # Greater burn-in for cleaner result
162      MCMC.burnin = 20000,
163      # Greater sample size for greater stability
164      MCMC.samplesize = 100000,
165      seed = 42,
166      MCMC.interval = 1000,
167      # Only needed for convergence pvals to improve
168      MCMLE.maxit = 45,
169      # Smaller steps for stability
170      MCMLE.steplength = 0.25,
171      parallel = n_cores,
172      parallel.type = "PSOCK"
173    )
174  )
175
176  model_1_panel <- auto_ergm(model=model_1, mcmc=TRUE, name="ergm_m1")
177  model_1_panel$gof
178  snafun::stat_plot_gof(model_1_panel$gof)
179  texreg::screenreg(list(base_ergm, model_1))
180  # ------------------------------------------------------------------------------ #
181  # Iteration 2 + MCMC Diagnostics + GOF
182  model_2_params <- bondora_net ~ edges +
183    # low decay important because there is high clustering around low degrees
184    gwb1degree(decay=0.15, fixed=TRUE) +
185    # decay should be higher due to wider variation in degree but
186    # too high of degree makes the traces concentrated around the tails.
187    gwb1dsp(decay=0.5, fixed=TRUE)
188
189  model_2 <- ergm::ergm(
190    model_2_params,
191
192    # Max b2 degree is 72, so this constraint is reasonable
193    # and helps convergence significantly.
194    # Technically in the Bondora population this can be
195    # far higher but we are studying a subsample.
196    constraints = ~ bd(minout = 0, maxout = 80),
197
```

```
198     control = ergm::control.ergm(
199       # Greater burn-in for cleaner result
200       MCMC.burnin = 20000,
201       # Greater sample size for greater stability
202       MCMC.samplesize = 100000,
203       seed = 42,
204       MCMC.interval = 1000,
205       # Only needed for convergence pvals to improve
206       MCMLE.maxit = 45,
207       # Smaller steps for stability
208       MCMLE.steplength = 0.25,
209       parallel = n_cores,
210       parallel.type = "PSOCK"
211       )
212     )
213
214 model_2_panel <- auto_ergm(model=model_2, mcmc=TRUE, name="ergm_m2")
215 snafun::stat_plot_gof(model_2_panel$gof)
216 model_2_panel$gof
217 models <- list(base_ergm, model_1, model_2)
218 texreg::screenreg(models)
219 # ------------------------------------------------------------------------------ #
220 # Iteration 3 + MCMC Diagnostics + GOF
221 model_3_params <- bondora_net ~ edges +
222     # low decay important because there is high clustering around low degrees
223     gwb1degree(decay=0.15, fixed=TRUE) +
224     # decay should be higher due to wider variation in degree but
225     # too high of degree makes the traces concentrated around the tails.
226     gwb1dsp(decay=0.5, fixed=TRUE) +
227     # See differences across genders (implicitly, since b1nodemix unavailable)
228     b1nodematch("b1_gender", diff=FALSE)
229
230 model_3 <- ergm::ergm(
231     model_3_params,
232
233     # Max b2 degree is 72, so this constraint is reasonable
234     # and helps convergence significantly.
235     # Technically in the Bondora population this can be
236     # far higher but we are studying a subsample.
237     constraints = ~ bd(minout = 0, maxout = 80),
238
239     control = ergm::control.ergm(
240       # Greater burn-in for cleaner result
241       MCMC.burnin = 20000,
242       # Greater sample size for greater stability
243       MCMC.samplesize = 100000,
244       seed = 42,
245       MCMC.interval = 1000,
246       # Only needed for convergence pvals to improve
247       MCMLE.maxit = 45,
248       # Smaller steps for stability
249       MCMLE.steplength = 0.25,
250       parallel = n_cores,
251       parallel.type = "PSOCK"
252     )
253 )
254
255 model_3_panel <- auto_ergm(model=model_3, mcmc=TRUE, name="ergm_m3")
256 snafun::stat_plot_gof(model_3_panel$gof)
257 model_3_panel$gof
258 models <- list(base_ergm, model_1, model_2, model_3)
259 texreg::screenreg(models)
260 # ------------------------------------------------------------------------------ #
261 # Iteration 4 + MCMC Diagnostics + GOF
262 model_4_params <- bondora_net ~ edges +
263     # low decay important because there is high clustering around low degrees
```

```r
264    gwb1degree(decay=0.15, fixed=TRUE) +
265    # decay should be higher due to wider variation in degree but
266    # too high of degree makes the traces concentrated around the tails.
267    gwb1dsp(decay=0.5, fixed=TRUE) +
268    # See differences across genders (implicitly, since b1nodemix unavailable)
269    b1nodematch("b1_gender", diff=TRUE) +
270    # See if higher ages make a difference
271    b1cov("b1_age")
272
273  model_4 <- ergm::ergm(
274    model_4_params,
275
276    # Max b2 degree is 72, so this constraint is reasonable
277    # and helps convergence significantly.
278    # Technically in the Bondora population this can be
279    # far higher but we are studying a subsample.
280    constraints = ~ bd(minout = 0, maxout = 80),
281
282    control = ergm::control.ergm(
283      # Greater burn-in for cleaner result
284      MCMC.burnin = 20000,
285      # Greater sample size for greater stability
286      MCMC.samplesize = 100000,
287      seed = 42,
288      MCMC.interval = 1000,
289      # Only needed for convergence pvals to improve
290      MCMLE.maxit = 45,
291      # Smaller steps for stability
292      MCMLE.steplength = 0.25,
293      parallel = n_cores,
294      parallel.type = "PSOCK"
295    )
296  )
297
298  model_4_panel <- auto_ergm(model=model_4, mcmc=TRUE, name="ergm_m4")
299  model_4_panel$gof
300  snafun::stat_plot_gof(model_4_panel$gof)
301  models <- list(base_ergm, model_1, model_2, model_3, model_4)
302  texreg::screenreg(models)
303  # ---------------------------------------------------------------------------- #
```

scripts/ergm_network_analysis.R

References

Agrawal, V. (2012). Managing the diversified team: Challenges and strategies for improving performance. *Team Performance Management: An International Journal*, *18*(7), 384–400. https://doi.org/10.1108/13527591211281129

Aliano, M., Alnabulsi, K., Cestari, G., & Ragni, S. (2023). The role of gender and education in peer-to-peer lending activities: Evidence from a european cross-country study. *European Scientific Journal ESJ*, *2*. https://doi.org/10.19044/esipreprint.2.2023.p95

Alistair Milne, & Paul Parboteeah. (2017). *The business models and economics of peer-to-peer lending*. Retrieved from https://repository.lboro.ac.uk/articles/online_resource/The_business_models_and_economics_of_peer-to-peer_lending/9494891

Amar, M., Ariely, D., Ayal, S., Cryder, C. E., & Rick, S. I. (2011). Winning the battle but losing the war: The psychology of debt management. *Journal of Marketing Research*, *48*, S38–S50.

Ayal, S., Bar-Haim, D., & Ofir, M. (2018). Behavioral biases in peer-to-peer (P2P) lending. *Forthcoming in Behavioral Finance: The Coming of Age (Venezia I. Ed., World Scientific Publishers)*.

Ayal, S., Hochman, G., & Zakay, D. (2011). Two sides of the same coin: Information processing style and reverse biases. *Judgment and Decision Making*, *6*(4), 295–305. https://doi.org/10.1017/S193029750000190X

Ayal, S., & Zakay, D. (2009). The perceived diversity heuristic: The case of pseudodiversity. *Journal of Personality and Social Psychology*, *96*(3), 559–573. https://doi.org/10.1037/a0013906

Bondora. (2024, August 28). How are bondora risk ratings calculated? Bondora help center. Retrieved November 13, 2025, from https://help.bondora.com/hc/en-us/articles/14814705732881-How-are-Bondora-risk-ratings-calculated

Cooper, D., Gorbachev, O., & LuengoPrado, M. J. (2023). Consumption, credit, and the missing young. *Journal of Money, Credit and Banking*, *55*(2), 379–405.

Croson, R., & Gneezy, U. (2009). Gender differences in preferences. *Journal of Economic Literature*, *47*(2), 448–474.

Galak, J., Small, D. A., & Stephen, A. T. (2010). Micro-finance decision making: A field study of prosocial lending. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.1634949

Herzenstein, M., Dholakia, U. M., & Andrews, R. L. (2011). Strategic herding behavior in peer-to-peer loan auctions. *Journal of Interactive Marketing*, *25*(1), 27–36. https://doi.org/10.1016/j.intmar.2010.07.001

Kahneman, D., & Tversky, A. (1979). Prospect theory: An analysis of decision under risk. *Econometrica*, *47*(2), 263–291. https://doi.org/10.2307/1914185

Lee, E., & Lee, B. (2012). Herding behavior in online P2P lending: An empirical investigation. *Electronic Commerce Research and Applications*, *11*(5), 495–503. https://doi.org/10.1016/j.elerap.2012.02.001

Liu, Y., Baals, L. J., Osterrieder, J., & Hadji-Misheva, B. (2024). Network centrality and credit risk: A comprehensive analysis of peer-to-peer lending dynamics. *Finance Research Letters*, *63*, 105308. https://doi.org/10.1016/j.frl.2024.105308

Manu Siddhartha. (2021). Bondora peer to peer lending loan data. Kaggle. Retrieved November 12, 2025, from https://www.kaggle.com/datasets/sid321axn/bondora-peer-to-peer-lending-loan-data

Ravina, E. (2019). Love & loans: The effect of beauty and personal characteristics in credit markets. *Available at SSRN 1107307*.

Serrano-Cinca, C., Gutiérrez-Nieto, B., & López-Palacios, L. (2015). Determinants of default in

P2P lending. *PLOS ONE*, *10*(10), e0139427. https://doi.org/10.1371/journal.pone.0139427

Simpson, W. (2001). QAP: The quadratic assignment procedure. *North American STATA Users' Group Meeting*, *1*, 1–17.

Stivala, A., Wang, P., & Lomi, A. (2025). *Improving exponential-family random graph models for bipartite networks*. arXiv. https://doi.org/10.48550/ARXIV.2502.01892

Yao, J., Chen, J., Wei, J., Chen, Y., & Yang, S. (2019). The relationship between soft information in loan titles and online peer-to-peer lending: Evidence from RenRenDai platform. *Electronic Commerce Research*, *19*(1), 111–129. https://doi.org/10.1007/s10660-018-9293-z

Zhao, H., Liu, Q., Wang, G., Ge, Y., & Chen, E. (2016). Portfolio selections in P2P lending: A multi-objective perspective. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2075–2084. San Francisco California USA: ACM. https://doi.org/10.1145/2939672.2939861

Zuo, Y. (n.d.). *CLUSTERING ANALYSIS TO SUPPORT LENDER's DECISION-MAKING IN P2P LENDING*.

B Technology Statement

During the preparation of this work, we used ChatGPT in order to generate select parts of the R script utilised to process the dataset. Specifically, the tool was used to transform the processed dataset into a format that `igraph` would accept as a network object. Additionally, some R functions written by the group were improved with ChatGPT's suggestions. No AI tool was utilised to independently write parts of the report. The following parts of the assignment were affected/generated by AI tool usage: **INTRODUCTION;** The tool was utilised to evaluate the validity of certain theoretical concepts and refine them. **DATASET;** the data described within this section was processed partly by some code drafted by ChatGPT and edited by the group. After using this tool/service, **Group 7** evaluated the validity of the tool's outputs, including the sources that generative AI tools have used, and edited the content as needed. As a consequence, **Group 7** takes full responsibility for the content of their work.