DATA 100 Final Project: Analysis of German Credit Data

Carter Fillingham (169064112), Karim Nasereddin (169119978)

2025-12-10

```
library(tidyverse); library(patchwork); library(tidymodels);
library(rsample); library(dplyr); library(ggrepel)
```

Introduction To The German Credit Dataset

The German Credit Data was downloaded from the University of California Irvine's Machine Learning Repository (Hofmann 1994). Where the dataset originates from Prof. Hans Hofmann of the University of Hamburg, (Michie, Spiegelhalter, and Taylor 1994) who provided the data to a European research project (STATLOG) that aimed to evaluate "the performance of machine learning, neural and statistical algorithms" on different datasets in order to provide guidance on which models to use in the future . (European Commission 1994)

Goals

For this project, our goal is to model the response variable credit amount given an individuals credit history, age, existing credit lines, etc...

```
credit <- tibble(read.table("german.data"))
```

Data Cleaning Procedure

The German Credit Data contains both numerical and categorical data. The names of the variables are a "V" followed by a single or double digit number. The categorical data is encoded

such that the values for each variable is a character consisting of a "A" followed by either a two digit or three digit number.

For example,

```
credit %>% select(V10) %>% unique()
```

```
# A tibble: 3 x 1
  V10
  <chr>
1 A101
2 A103
3 A102
```

Where "V10" is the variable name of the category of whether there is or is not a guarantor for the credit. "A101" translates to no guarantor, "A102" to a co-applicant, and "A103" to a guarantor.

First, we will rename the columns to help with understanding the data and we will use appendix D in the paper (Grömping 2019) which goes into the history of the German Credit Data and provides a translation for the variables.

```
credit <- credit %>%
  rename("account_status" = V1, "credit_duration" = V2, "credit_history" = V3,
         "credit_purpose" = V4, "credit_amount" = V5, "savings_category" = V6,
         "employment_length" = V7, "credit_rate" = V8,
         "mariage_sex_status" = V9, "external_creditors" = V10,
         "residence_years" = V11, "valuable_property" = V12,
         "years_old" = V13, "other_credit_plans" = V14, "housing_type" = V15,
         "credit_lines" = V16, "job_type" = V17, "dependants" = V18,
         "landline" = V19, "foreign_worker" = V20, "credit_risk" = V21)
credit_reduced_variables <- credit
```

Following the tidy data principles, we want each column to be a variable, each row to be an observation, and each entry to be a single value. For the guarantor example, we will create three variables, called none, co-applicant, and guarantor where the value is 1 if there is a none, co-applicant, or guarantor respectively. And are a 0 if there is no none, co-applicant, or guarantor for each observation. Then we will proceed similarity for each variable.

```
pivot_wider_helper <- function(data, column) {
  data %>%
    pivot_wider(
```

```
      names_from = all_of(column),
      values_from = all_of(column),
      values_fn = ~1,
      values_fill = 0,
      names_prefix = str_c(column, "_")
    )
}
non_numeric_columns <- names(
  credit %>%
    select(!where(is.numeric))
)
for (column in non_numeric_columns) {
  credit <- pivot_wider_helper(credit, column)
}
credit <- credit %>%
  mutate(across(where(is.double), as.factor))
credit_clean <- credit
credit <- credit <- tibble(read.table("german.data"))
```

```
set.seed(123)  # ensures reproducible splits
split1 <- initial_split(credit_clean, prop = 0.85)
credit_train_valid <- training(split1)
credit_test <- testing(split1)
set.seed(123)
split2 <- initial_split(credit_train_valid, prop = 0.80)
credit_train <- training(split2)
credit_valid <- testing(split2)
```

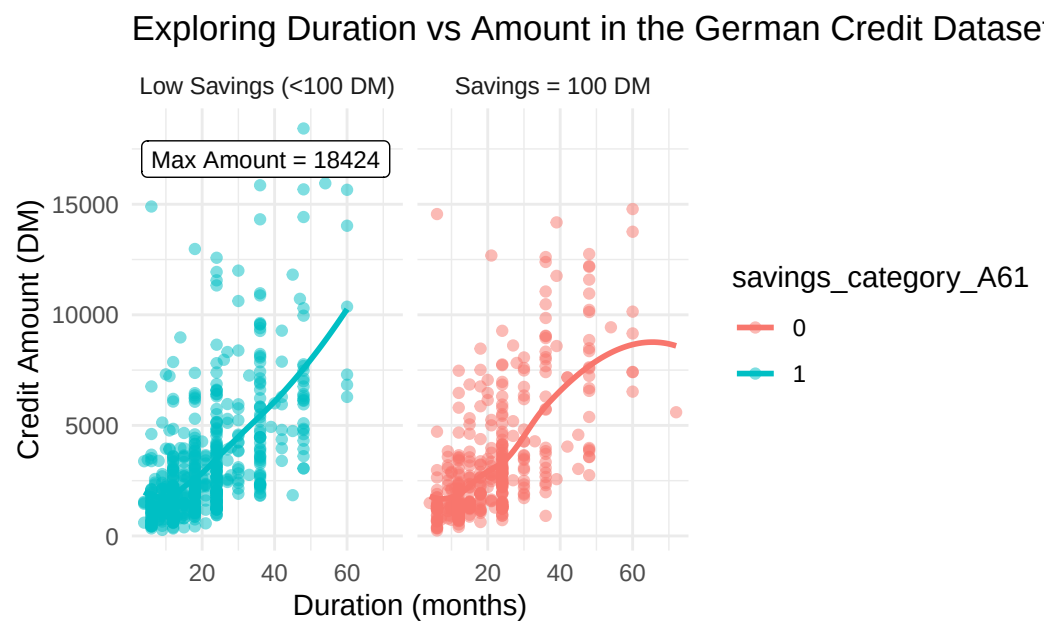
Exploratory Plot 1

Exploratory Plot 1 uses the **training set** to examine how credit duration relates to credit amount, with colour and faceting based on the **low-savings indicator (savings_category_A61)** (Hofmann 1994). A smoothing curve highlights the trend for each group, and a label marks the highest loan amount. The plot shows that applicants with very low savings tend to request higher credit amounts and often over longer durations. These relationships are important for modelling because duration and amount are key continuous predictors, while savings level may help explain variation in borrowing behaviour.

```
credit_train <- credit_clean
credit_train$savings_label <- ifelse(
```

```
  credit_train$savings_category_A61 == 1,
  "Low Savings (<100 DM)",
  "Savings   100 DM"
)
ggplot(credit_train, aes(x = credit_duration, y = credit_amount)) +
  geom_smooth(aes(colour = savings_category_A61), method = "loess", se = FALSE) +
  geom_point(aes(colour = savings_category_A61), alpha = 0.5) +
  geom_label_repel(
    data = credit_train |> slice_max(credit_amount, n = 1),
    aes(label = paste("Max Amount =", credit_amount)),
    size = 3
  ) +
  facet_wrap(~ savings_label) +
  labs(
    title = "Exploring Duration vs Amount in the German Credit Dataset",
    x = "Duration (months)",
    y = "Credit Amount (DM)",
    caption = "Training data only.Relationship between savings, duration, and loan amount."
  ) +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```

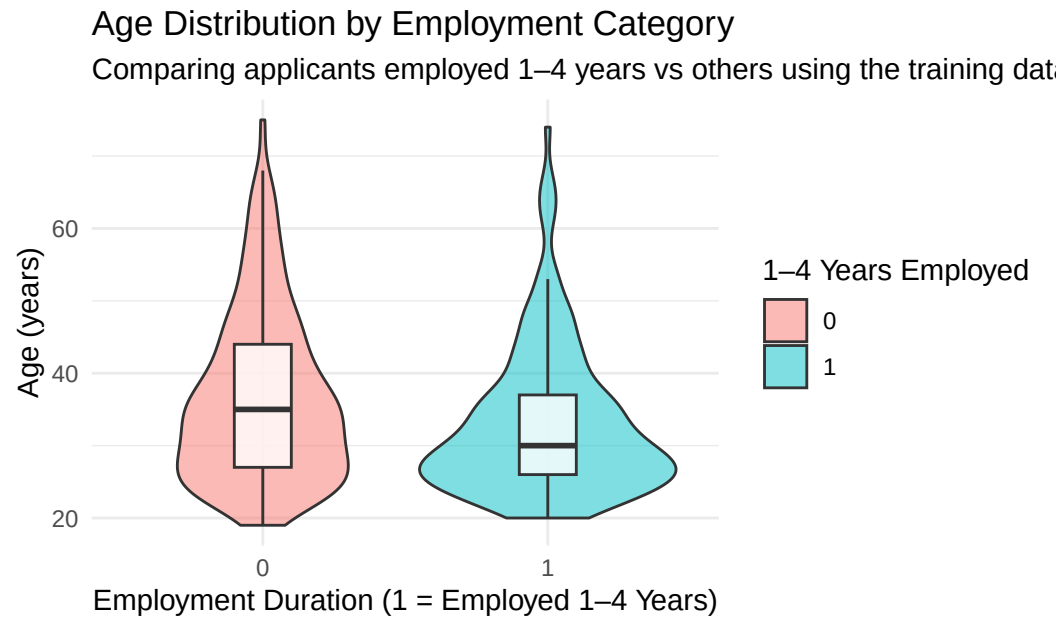


a only.Relationship between savings, duration, and loan amount.

Exploratory Plot 2

Exploratory Plot 2 compares applicant age across employment categories using **only the training data**. The violin and boxplot combination shows that applicants employed for 1–4 years tend to be slightly younger and have a more concentrated age range, while the other group spans early, mid, and late career stages, thus a wider and older distribution. This relationship is relevant **for modelling** because both age and employment stability influence credit behaviour and inform which features should be included in later

```
ggplot(credit_train, aes(x = employment_length_A73, y = years_old)) +
  geom_violin(aes(fill = employment_length_A73), alpha = 0.5) +
  geom_boxplot(width = 0.2, alpha = 0.8, outlier.shape = NA) +
  labs(
    title = "Age Distribution by Employment Category",
    subtitle = "Comparing applicants employed 1-4 years vs others using the training data",
    x = "Employment Duration (1 = Employed 1-4 Years)",
    y = "Age (years)",
    fill = "1-4 Years Employed",
    caption = "Exploratory Plot 2: visualizing feature relationships relevant for modelling"
  ) +
  theme_minimal()
```



oratory Plot 2: visualizing feature relationships relevant for modelling

Exploratory Linear Model 1

The full model includes credit duration, the low-savings indicator, and their interaction to test whether the effect of duration on credit amount differs by savings level. Duration is strongly significant, while the savings and interaction terms are weaker, but together they still provide the best predictive performance based on validation RMSE.

```
model_full <- lm(credit_amount ~ credit_duration * savings_category_A61,
                 data = credit_train)
summary(model_full)
```

```
Call:
lm(formula = credit_amount ~ credit_duration * savings_category_A61,
    data = credit_train)

Residuals:
    Min      1Q  Median      3Q     Max
-5308.3 -1261.7  -433.8   662.4 13790.6

Coefficients:
                                    Estimate Std. Error t value Pr(>|t|)
(Intercept)                          180.443    221.236   0.816    0.415
credit_duration                      148.928      8.865  16.800   <2e-16 ***
savings_category_A611                 59.819    285.543   0.209    0.834
credit_duration:savings_category_A611  -4.730     11.716  -0.404    0.686
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2207 on 996 degrees of freedom
Multiple R-squared:  0.3908,    Adjusted R-squared:  0.3889
F-statistic: 212.9 on 3 and 996 DF,  p-value: < 2.2e-16
```

The model is used to generate predicted credit amounts for the **validation set**, and the RMSE is calculated by comparing these predictions to the actual values. This RMSE score measures how well the full model predicts unseen data, providing an objective way to assess its performance.

```
predict_full <- predict(model_full, newdata = credit_valid)
rmse_full <- sqrt(mean((credit_valid$credit_amount - predict_full)^2))
rmse_full
```

```
[1] 1831.186
```

This model removes the **categorical savings variable** and uses only credit duration to predict credit amount. The RMSE calculated on the validation set shows how much predictive accuracy is lost when the categorical feature is removed.

```
model_no_cat <- lm(credit_amount ~ credit_duration,
                   data = credit_train)
predict_no_cat <- predict(model_no_cat, newdata = credit_valid)
rmse_no_cat <- sqrt(mean((credit_valid$credit_amount - predict_no_cat)^2))
rmse_no_cat
```

```
[1] 1833.651
```

This version removes the interaction term but keeps both predictors. By calculating the validation RMSE, we can see how much predictive performance changes when the interaction between duration and savings level is excluded, helping assess whether the interaction meaningfully improves the model.

```
model_no_interaction <- lm(credit_amount ~ credit_duration + savings_category_A61,
                           data = credit_train)
predict_no_inter <- predict(model_no_interaction, newdata = credit_valid)
rmse_no_inter <- sqrt(mean((credit_valid$credit_amount - predict_no_inter)^2))
rmse_no_inter
```

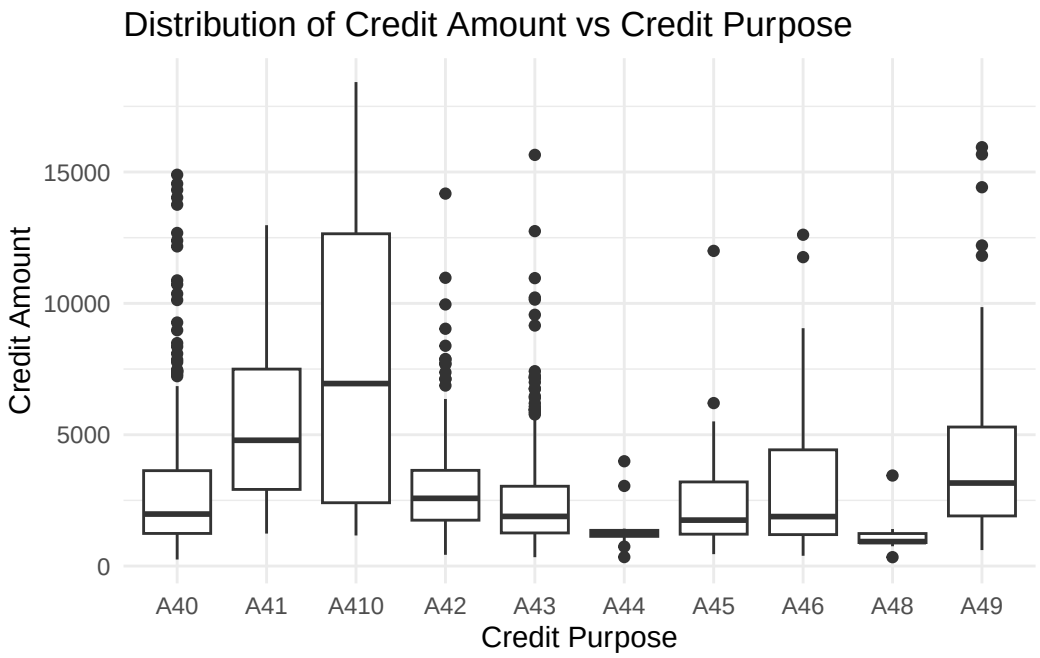
```
[1] 1834.262
```

Model 2 and Advanced Data Cleaning

Given that the independent variables are mostly categorical, 54 categorical variables vs 8 numeric variables, a multiple linear regression model over the binary 0 and 1 encoding of the categorical variables may not be the best idea. Consider the credit_purpose variable, in the unclean data there are 11 possible values for the variable from A40 to A410, this creates 10 new variables in the cleaned dataset. And across the whole dataset which originally had 21 variables we now have 62 variables. From the lecture content, we know that there is a trade off between bias and variance, more variables may give more predictive power through overfitting (not good), but also introduces an increase in variability. A good model should find a balanced middle ground.

To reduce the amount of variables we will follow chapter 17.3: of the text "Tidy Modeling with R" which introduces likelihood encoding (Max Kuhn 2023). Instead of encoding categorical variables with a 1 or 0, we will encode the variables with the mean of the response variable credit_amount for each category. For example, the credit_purpose variable:

```
credit %>%
  ggplot(aes(x = V4, y = V5)) +
  geom_boxplot() +
  labs(
    title = "Distribution of Credit Amount vs Credit Purpose",
    y = "Credit Amount",
    x = "Credit Purpose") +
  theme_minimal()
```



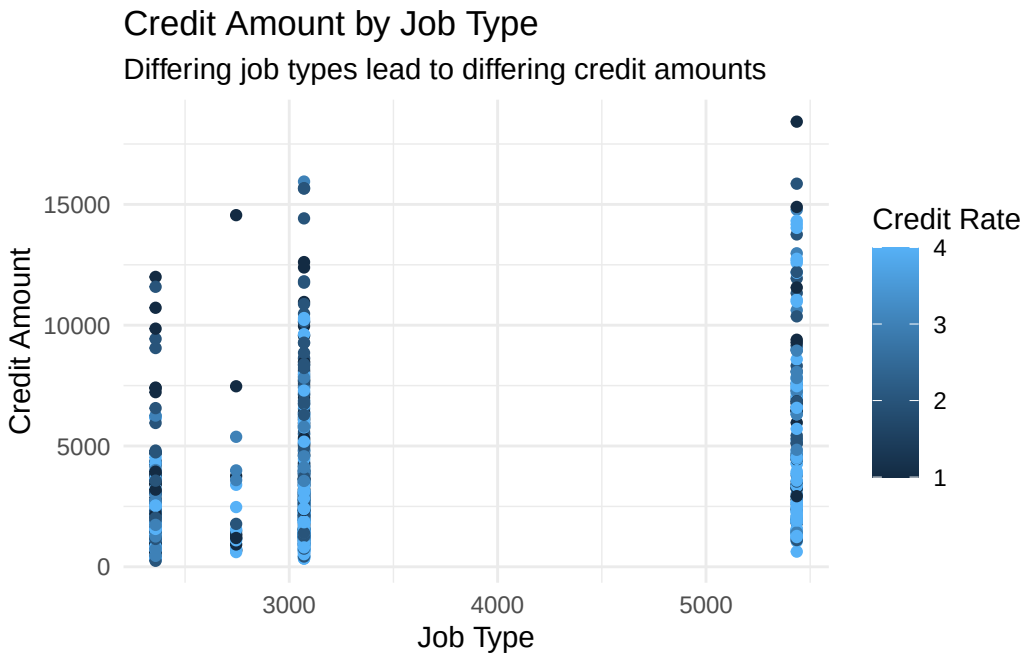
This gives us much more insight about the data compared to just 1s and 0s. We can see that for credit purpose A44 (Domestic Appliances) tends to correspond to lower credit amount vs A49 (Business).

```
credit_reduced_variables <- credit_reduced_variables %>%
  mutate(across(all_of(non_numeric_columns),
                ~ave(credit_amount, ., FUN = mean)))
```

Intuitively, we know that good credit in the past may be a good predictor for your credit in the future. Other intuitive credit predictors may also be, job type where higher paying and more

reputation may lead a lender to lend more, credit rate where cheaper credit may encourage more borrowing, and credit type because you may need to borrow more for a house instead of a car.

```
credit_reduced_variables %>% ggplot(aes(x = job_type, y = credit_amount,
                                        color = credit_rate)) +
  geom_point() +
  labs(title = "Credit Amount by Job Type",
       subtitle = "Differing job types lead to differing credit amounts",
       y = "Credit Amount",
       x = "Job Type",
       color = "Credit Rate") +
  theme_minimal()
```




```
model_reduced_variables <- lm(credit_amount ~ job_type + credit_duration
                              + credit_rate + credit_purpose, credit_reduced_variables)
summary(model_reduced_variables)
```

```
Call:
lm(formula = credit_amount ~ job_type + credit_duration + credit_rate +
    credit_purpose, data = credit_reduced_variables)
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-5061.8 -1061.7  -158.1   659.7 12164.8

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)     -508.89764  303.69883  -1.676   0.0941 .
job_type           0.62089    0.06556   9.470  < 2e-16 ***
credit_duration  134.82120    5.10330  26.418  < 2e-16 ***
credit_rate     -777.45652   53.72534 -14.471  < 2e-16 ***
credit_purpose     0.37976    0.06041   6.287 4.85e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1865 on 995 degrees of freedom
Multiple R-squared:  0.5653,    Adjusted R-squared:  0.5636
F-statistic: 323.5 on 4 and 995 DF,  p-value: < 2.2e-16
```


European Commission. 1994. "Comparative Testing and Evaluation of Statistical and Logical Learning Algorithms for Large-Scale Applications in Classification, Prediction and Control (STATLOG)." CORDIS. https://cordis.europa.eu/project/id/5170.

Grömping, Ulrike. 2019. "South German Credit Data: Correcting a Widely Used Data Set." https://www1.beuth-hochschule.de/FB_II/reports/Report-2019-004.pdf.

Hofmann, H. 1994. "Statlog (German Credit Data)." UCI Machine Learning Repository. https://archive.ics.uci.edu/dataset/144/statlog+german+credit+data.

Max Kuhn, Julia Silge. 2023. "Tidy Modeling with r." https://www.tmwr.org/.

Michie, D., D. J. Spiegelhalter, and C. C. Taylor. 1994. *Machine Learning, Neural and Statistical Classification*. https://www.researchgate.net/publication/2335004_Machine_Learning_Neural_and_Statistical_Classification.