# STAT387 Group 4 Project Paper

Mina Mehdinia, Tim Luo, & Will McIntosh

# Introduction

We conducted an analysis on the German credit dataset. The dataset contains quantifiable predictors such as "CheckingStatus1" and "purpose" of the loan as well as quantitative predictors like "Age" of the client and "Duration" of the loan[1]. The target variable in the dataset is a boolean whether the client did or not did not "Default" on their loan, and we focused on the primary metric for selecting the optimal statistical model being OEC for all of the five models we developed.

We utilized several modeling methods such as LDA, QDA, Naive Bayes, KNN, and Logistic Regression models to discover the optimal configuration for precise predictions on which clients are most likely to default on their loans. This would allow the German bank to know which clients they should give loans to and which they should not. We computed the confusion matrices, sensitivities, specificities, overall misclassification rates, plotted the ROC curves for each of the five models, and more.

| | Default | checkingstatus1 | duration | history | purpose | amount | savings |
|---|---|---|---|---|---|---|---|
| 1 | 0 | A11 | 6 | A34 | A43 | 1169 | A65 |
| 2 | 1 | A12 | 48 | A32 | A43 | 5951 | A61 |
| 3 | 0 | A14 | 12 | A34 | A46 | 2096 | A61 |
| 4 | 0 | A11 | 42 | A32 | A42 | 7882 | A61 |
| 5 | 1 | A11 | 24 | A33 | A40 | 4870 | A61 |
| 6 | 0 | A14 | 36 | A32 | A46 | 9055 | A65 |

# Literature Review (Methods)

## LDA

An LDA (Linear Discriminant Analysis) classifier is a machine learning algorithm that is used to classify data into different categories. The algorithm works by finding the best linear combination of features that separates the data points into their respective categories[2]. The LDA method approximates the Bayes classifier by plugging estimates for $\pi_k$, $\mu_k$, and $\sigma^2$ into the formula[3]:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

[1] https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
[2] PSU Online , *9.2 - Discriminant Analysis*. 9.2 - discriminant analysis. Retrieved March 13, 2023, from https://online.stat.psu.edu/stat508/book/export/html/645
[3] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning: with applications in R (Vol. 112). New York: Springer. Page 144.

In particular, the following estimates are used:

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i$$

$$\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} \left(x_i - \hat{\mu}_k\right)^2$$

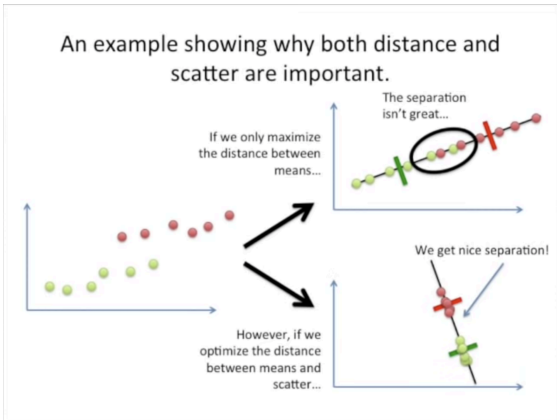
where:

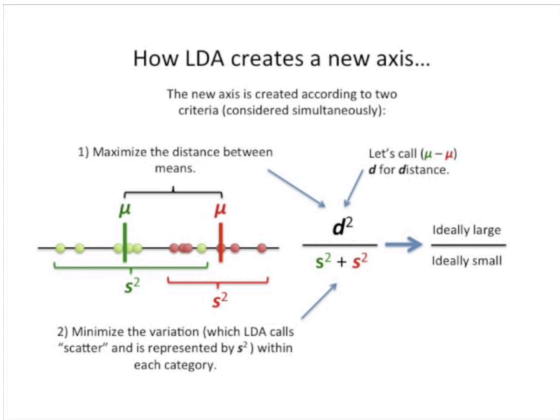
- $n$ is the total number of training observations
- $n_k$ is the number of training observations in the kth class.
- $\mu_k$ is simply the average of all the training observations from the kth class
- $\hat{\sigma}^2$ can be seen as a weighted average of the sample variances for each of the K classes.
- $\hat{\pi}_k = \frac{n_k}{n}$ LDA estimates $\pi_k$ using the proportion of the training observations that belong to the kth class

Culminating in:

$$\hat{\delta}_k(x) = x \cdot \frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}^2} + \log\left(\hat{\pi}_k\right)$$

## Visual Example of LDA




# QDA

QDA (Quadratic Discriminant Analysis) is a type of machine learning classifier used to separate objects or observations into different groups based on their characteristics.

To use QDA, we first need to have a set of data points that are already classified into different groups. QDA then uses mathematical equations to find patterns in the data that differentiate the groups from each other.

QDA assumes that each group of data points follows a specific type of distribution, called a quadratic distribution. It then uses these distributions to calculate the probability that a new observation belongs to each group.

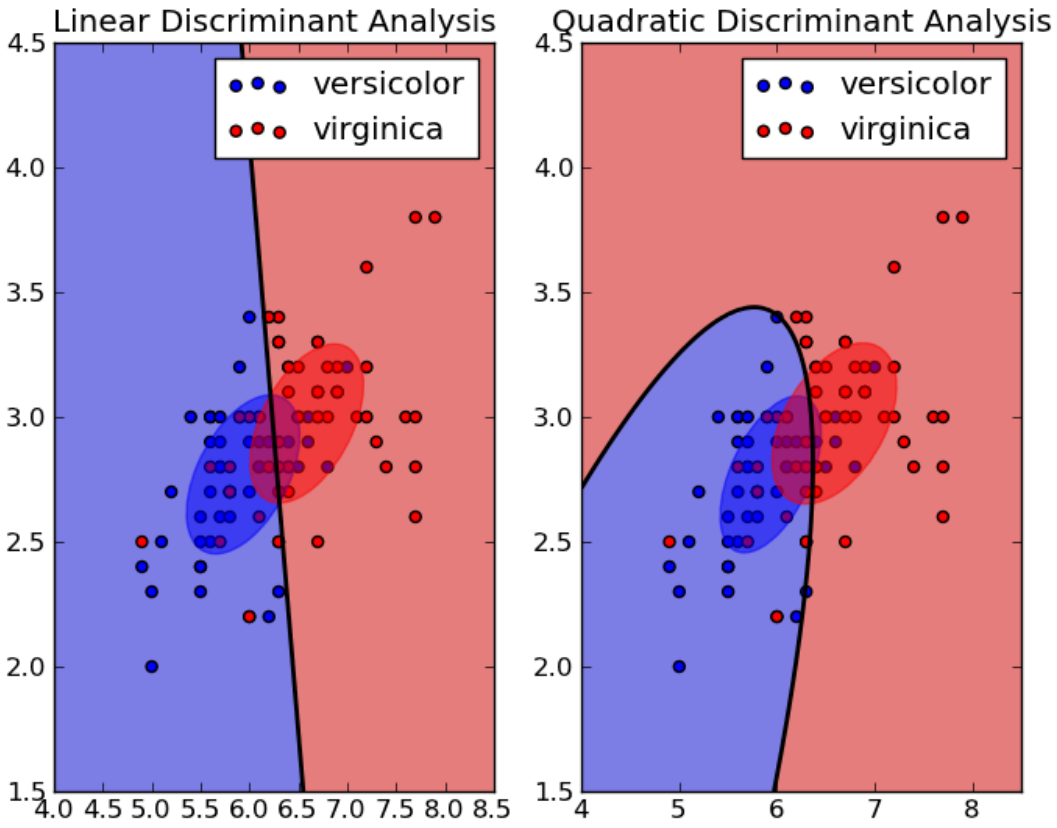
Like LDA, the QDA classifier results from assuming that the observations from each class are drawn from a Gaussian distribution, and plugging estimates for the parameters into Bayes' theorem in order to perform prediction. However, unlike LDA, QDA assumes that each class has its own covariance matrix. That is, it assumes that an observation from the kth class is of the form $X \sim N(\mu_k, \Sigma_k)$, where $\Sigma_k$ is a covariance matrix for the kth class[4]. Under this assumption, the Bayes classifier assigns an observation X = x to the class for which:

$$\begin{aligned} \delta_k(x) &= -\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}_k^{-1}(x-\mu_k) - \frac{1}{2}\log|\mathbf{\Sigma}_k| + \log \pi_k \\ &= -\frac{1}{2}x^T \mathbf{\Sigma}_k^{-1} x + x^T \mathbf{\Sigma}_k^{-1}\mu_k - \frac{1}{2}\mu_k^T \mathbf{\Sigma}_k^{-1}\mu_k - \frac{1}{2}\log|\mathbf{\Sigma}_k| + \log \pi_k \end{aligned}$$

## LDA vs QDA

| LDA | QDA |
|---|---|
| Assume has an underlying Gaussian distribution. | Assume has an underlying Gaussian distribution. |
| Estimate the means and covariances from just one class. | Estimate the means and covariances for each class. |
| Classification Rule: Assign a new observation to the class with the greatest likelihood. | Classification Rule: Assign a new observation to the class with the greatest likelihood. |
| Assume every covariance is the same. | Different classes can have different covariances. |

[4] (James et al., 2013) Page 152.




## Naive Bayes

Naive Bayes (NB) classifier is a simple machine learning algorithm used for classification tasks.

The "naive" part of the name comes from the assumption that all features in the data are independent of each other. In other words, the presence or absence of one feature does not affect the probability of any other feature being present or absent.

The classifier works by calculating the probability of each possible outcome, based on the input features. It does this by using Bayes' theorem, which tells us how to update the probability of a hypothesis (in this case, a particular classification) based on new evidence (the input features).

Once the classifier has calculated the probability of each possible outcome, it selects the outcome with the highest probability as its prediction[5].

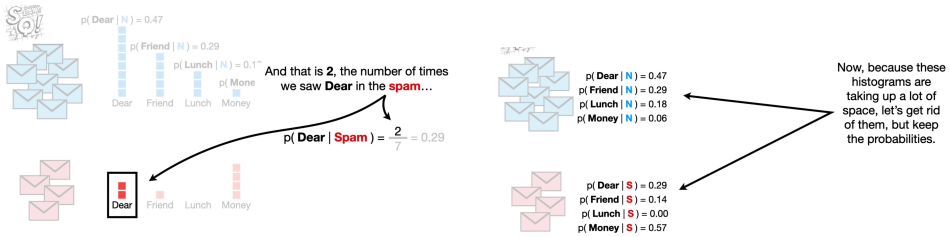
[5] (James et al., 2013) Page 154.

$$\Pr(Y = k|X = x) = \frac{\pi_k \times f_{k1}(x_1) \times f_{k2}(x_2) \times \cdots \times f_{kp}(x_p)}{\sum_{l=1}^{K} \pi_l \times f_{l1}(x_1) \times f_{l2}(x_2) \times \cdots \times f_{lp}(x_p)}$$

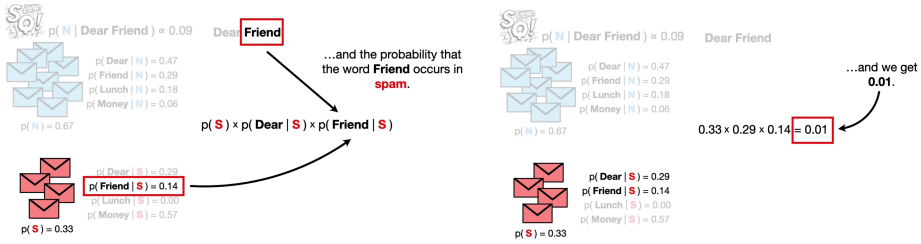
for $k = 1, \ldots, K$

## Example Using Spam Email Detection

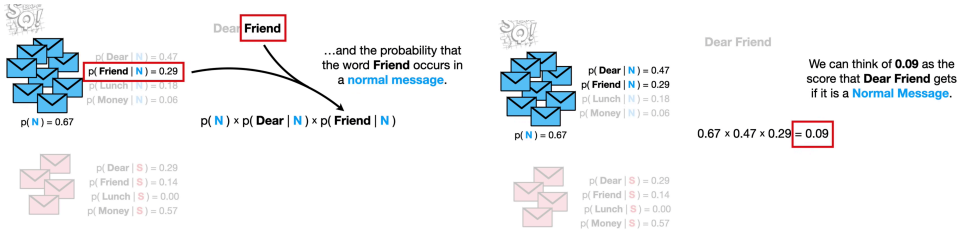
### Build Attribute Properties




### Build the Prior Probabilities




### Calculate the Likelihoods




# Logistic Regression

In the Machine Learning world, Logistic Regression is a kind of parametric classification model, despite having the word ‘regression’ in its name.

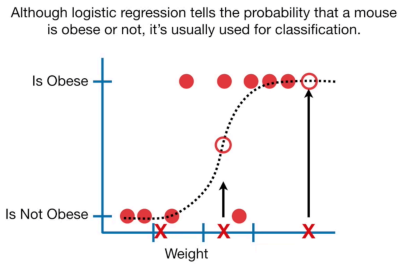
This means that logistic regression models are models that have a certain fixed number of parameters that depend on the number of input features, and they output categorical prediction, like for example if a plant belongs to a certain species or not.

In Logistic Regression, we don't directly fit a straight line to our data like in linear regression. Instead, we fit a S shaped curve, called Sigmoid, to our observations[6].

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}$$

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X.$$

### Example Using Obese Mice




## K-Nearest Neighbors (KNN)

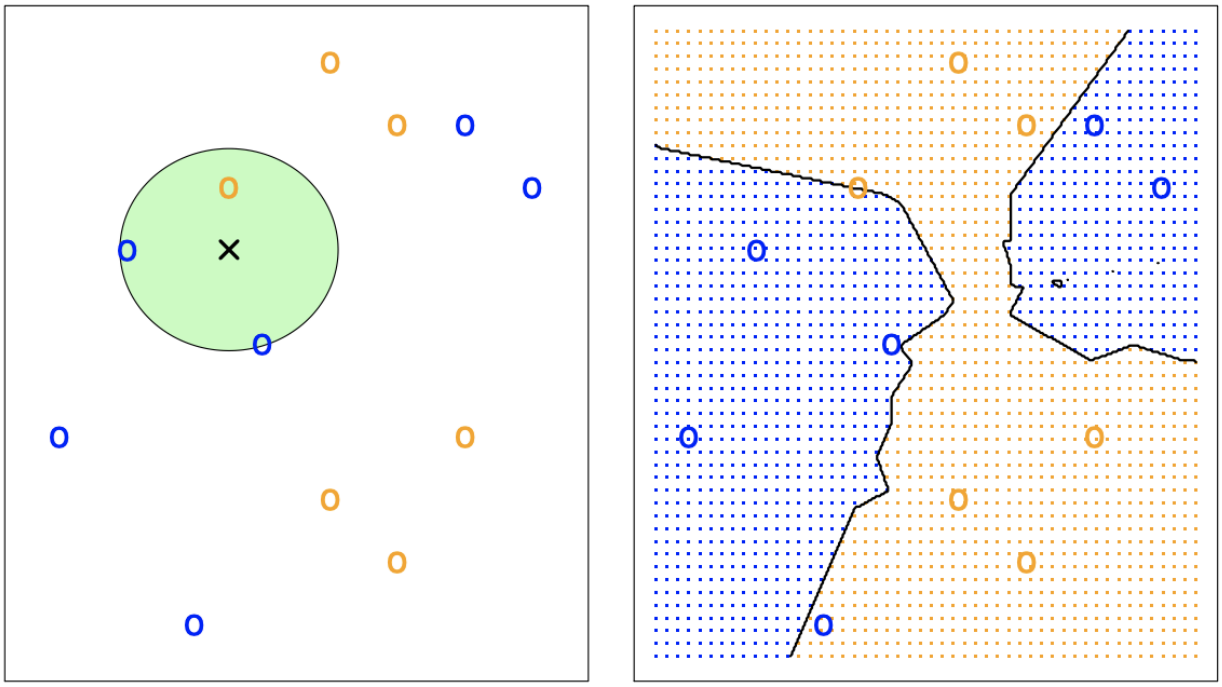
KNN is a type of machine learning algorithm that can be used for classification problems. In a classification problem, the goal is to predict which class a new observation belongs to, based on a set of features or characteristics.

The KNN algorithm works by finding the K nearest neighbors of the new observation in the dataset (where K is a pre-defined number). The "nearest neighbors" are defined as the observations that have the most similar characteristics to the new observation, based on some distance metric (such as Euclidean distance[7]).

$$\Pr(Y = j|X = x_0) = \frac{1}{K}\sum_{i\in\mathcal{N}_0} I(y_i = j).$$

[6] (James et al., 2013) Page 133.
[7] (James et al., 2013) Page 39.

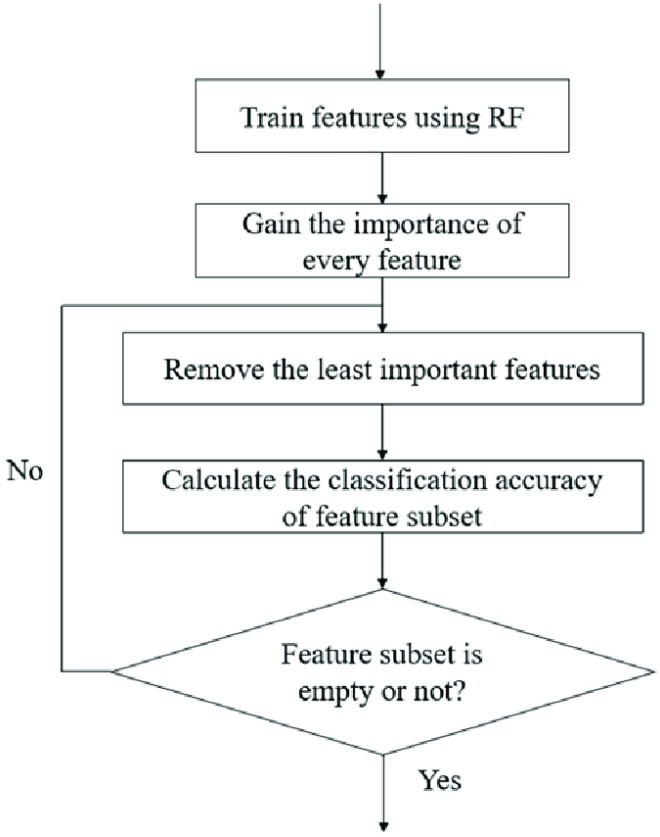
Example Using K=3



## Recursive Feature Elimination (RFE)

Recursive Feature Elimination (RFE for short, is a feature selection algorithm. A machine learning dataset for classification or regression consists of rows and columns, like an excel spreadsheet. Rows are often referred to as samples and columns are referred to as features, e.g. features of an observation in a problem domain[8]. It uses a random forest model (RF) to find the feature importances and provides the optimal combination of features based on the number of desire features as the input argument presented by the user.

[8] Guyon I.; Weston J.; Barnhill S.; Vapnik V. (2002). "Gene selection for cancer classification using support vector machines". Machine Learning. 46 (1–3): 389–422. doi:10.1023/A:1012487302797.




# Data Analysis

The german dataset contains data about loan applicants (borrower) of a German credit institution from 1994. There are 1,000 observations. Each observation represents one borrower. No missing or duplicate data. There are 21 variables total (20 predictors), 15 Categorical variables, 6 Numeric variables. Response variable is called “Default” and is categorical, it indicates if the borrower defaulted on their loan (not paying back) or not, 0 = Did not default and 1 = Did default.

## Use Case

The primary use case of this data would be that maybe some bank is paying us as data science consultants to help the bank predict if a future client is likely to default on their loan or not for the main purpose of selecting clients that will pay back their loans. This way, the banks get more money since clients pay back their loans with interest, as opposed to walking away with the cash.

## Categorical Predictors

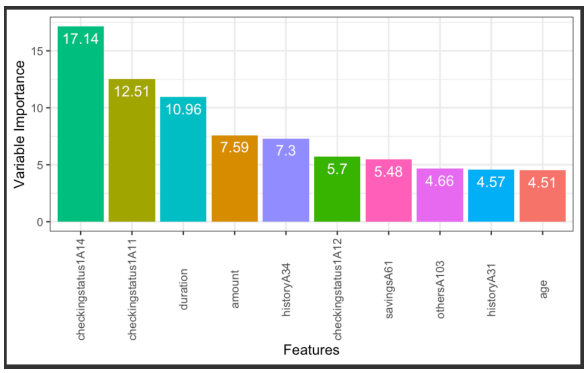
- Checking - status of existing checking account
- History - credit history of borrower
- Purpose - purpose of loan application
- Savings - status of savings/bonds account
- Employ - length of current employment
- Status - marital status and sex
- Others - indicate if someone else will be involved in the repayment or is guaranteeing the loan
- Installment - percentage of amount borrowed that will be charged by a lender to a borrower
- Property - most valuable asset
- Otherplans - type of installment plans the borrower already has
- Housing - residential status of borrower
- Job - current type of employment
- Tele - indicate if phone number is registered under the borrower's name
- Foreign - indicate if the borrower is a foreign worker

## Numerical Predictors

- Duration - repayment duration of the loan (in months)
- Amount - amount of money borrowed (in DM)
- Residence - length of time living at current residence (in years, 4 = 4+)
- Age - age of borrower (in years)
- Cards - number of existing credits at this bank
- Liable - number of people being liable to provide maintenance for

## Finding Important Features

We used both Recursive Feature Elimination and the summary results of running a Logistic Regression model on the entire dataset to discover which were the most prominent predictors that indicate a client's likelihood to default on their loan.

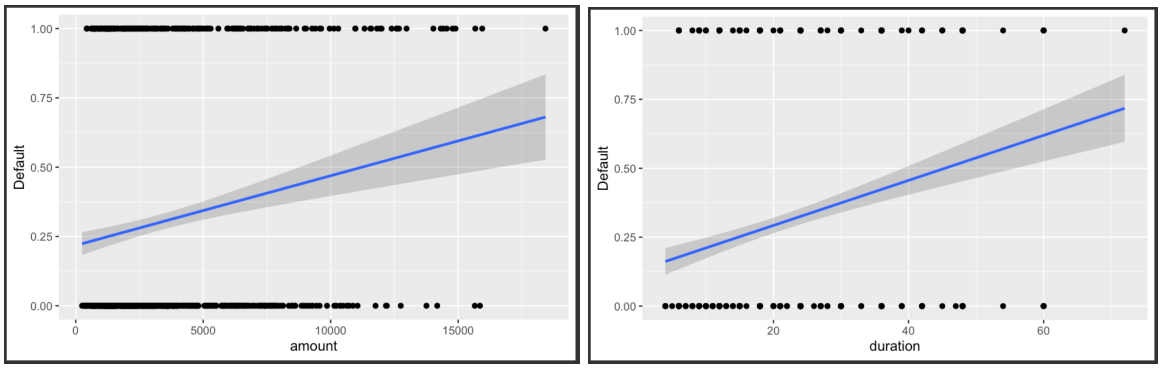



```
Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)        4.005e-01  1.084e+00   0.369 0.711869
checkingstatus1A12 -3.749e-01  2.179e-01  -1.720 0.085400 .
checkingstatus1A13 -9.657e-01  3.692e-01  -2.616 0.008905 **
checkingstatus1A14 -1.712e+00  2.322e-01  -7.373 1.66e-13 ***
duration           2.786e-02  9.296e-03   2.997 0.002724 **
historyA31         1.434e-01  5.489e-01   0.261 0.793921
historyA32        -5.861e-01  4.305e-01  -1.362 0.173348
historyA33        -8.532e-01  4.717e-01  -1.809 0.070470 .
historyA34        -1.436e+00  4.399e-01  -3.264 0.001099 **
purposeA41        -1.666e+00  3.743e-01  -4.452 8.51e-06 ***
purposeA410       -1.489e+00  7.764e-01  -1.918 0.055163 .
purposeA42        -7.916e-01  2.610e-01  -3.033 0.002421 **
purposeA43        -8.916e-01  2.471e-01  -3.609 0.000308 ***
purposeA44        -5.228e-01  7.623e-01  -0.686 0.492831
purposeA45        -2.164e-01  5.500e-01  -0.393 0.694000
purposeA46         3.628e-02  3.965e-01   0.092 0.927082
```

# Discovering Trustworthy Clients

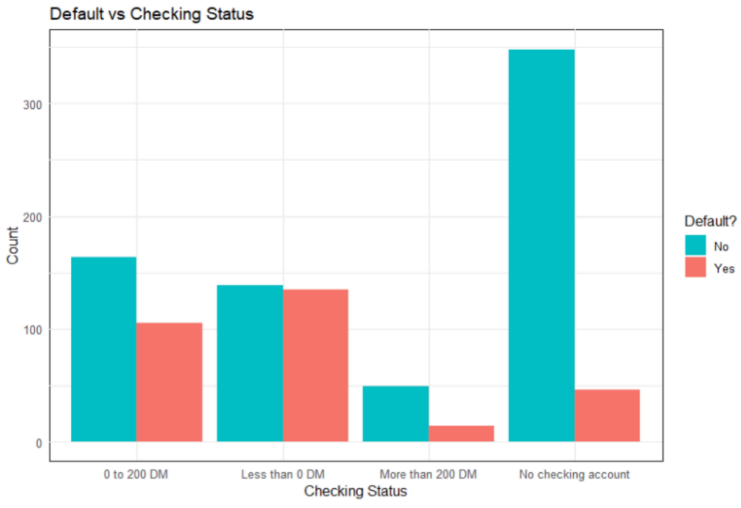
We then used the information from the important features to discover what the most trustworthy client would look like. Meaning, if we presented this information to a bank who was interested in discovering which clients are less likely to default on their loans, the CEO, CTO, and CFO wouldn't be interested about hearing anything regarding p-values and formulas as mentioned above, they'd want to know what a trustworthy client looks like, meaning specifically which behaviors do they exhibit. They'd be asking us data scientists the question, "before we approve someone for a loan, what spending habits should we be looking for?" Our answers are below.

## Amount and Duration




We can see from the images above that as the amount of the loan and duration of the loan increase, the likelihood of a client default on their loan increases. A trustworthy client is one who wants a short loan and for a shorter duration of time, around the first quartile is the optimal spot for both.

## Checking Status




If a client does not have a checking account with the bank, they are much less likely to default on their loan.

Statistics:

- χ-squared = 123.72
- df = 3
- p-value < 2.2e-16

A trustworthy client is one who currently does not have a checking account. Though this is difficult to explain, maybe there is a clear separation between those without checking accounts who did default and those who did not.

## Savings Account Status

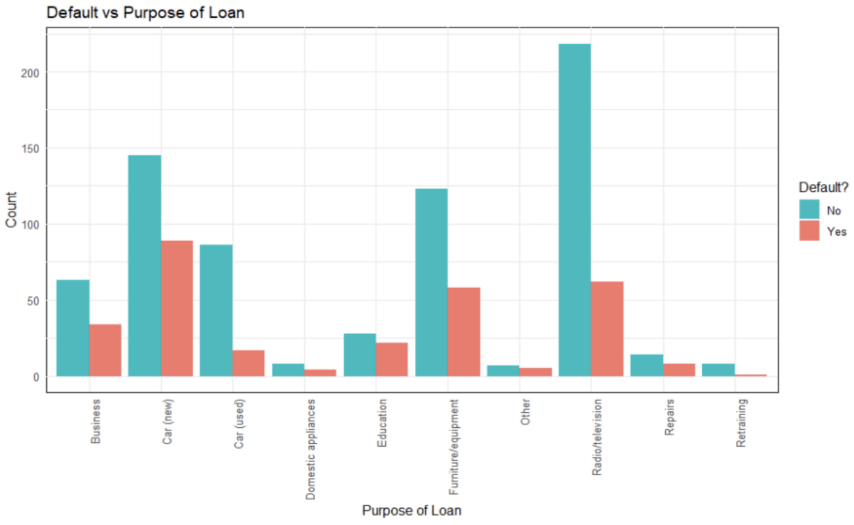

If the client has in their savings account less than 100 DM, they are more likely to not default on their loan.

Statistics:

- χ-squared = 36.099
- df = 4
- p-value = 2.761e-07

A trustworthy client is one who currently has less than 100 Deutschmarks (DM) in their savings account. Though this is difficult to explain, maybe there is a clear separation between those without checking accounts who did default and those who did not.

## Purpose of Loan




If the purpose of the loan is to get a car, furniture/equipment, or radio/television, they are more likely to not default on their loan.

Statistics:

- χ-squared = 33.356
- df = 9
- p-value = 0.0001157

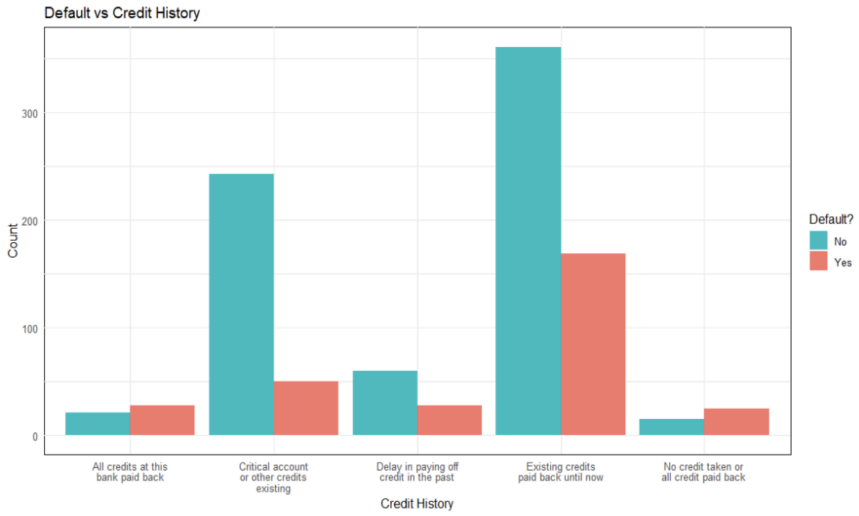
A trustworthy client is one who wants to purchase furniture, television or a radio. This might be highly correlated with the next feature in that people who want to purchase furniture or a television might be more inclined to own their home.

## Homeownership Status

`

A trustworthy client is one who owns their home as opposed to renting or free-loading off of someone else. This makes sense, if someone owns their home they are familiar with the processes of obtaining a loan (a large one at that for a home) and how to regularly make responsible payments.

## Existing Loans




Clients who have other existing credits are more likely to not default.

Statistics:

- χ-squared = 61.691
- df = 4
- p-value = 1.279e-12

A trustworthy client is one who currently has a few loans out. This makes sense in that these are responsible individuals who are familiar with the process of making payments on their loans and have shown a reputation for being careful with their money.

The Optimal Client

- Does not have a checking account.
- The purpose of the loan is to get a car, furniture, or radio/television.
- Savings account less than 100 DM.
- Has installment payments that are smaller than 20% of their take home salary.
- Owns their home.
- Clients who have other existing credits are more likely to not default.

# Conclusions

## Implementation and Results

The models we implemented are the ones listed above (except for RFE which was only used for data analysis):

- LDA
- QDA
- Logistic Regression
- Naïve Bayes
- K Nearest Neighbors (KNN)

## LDA Results

**Confusion Matrix:**

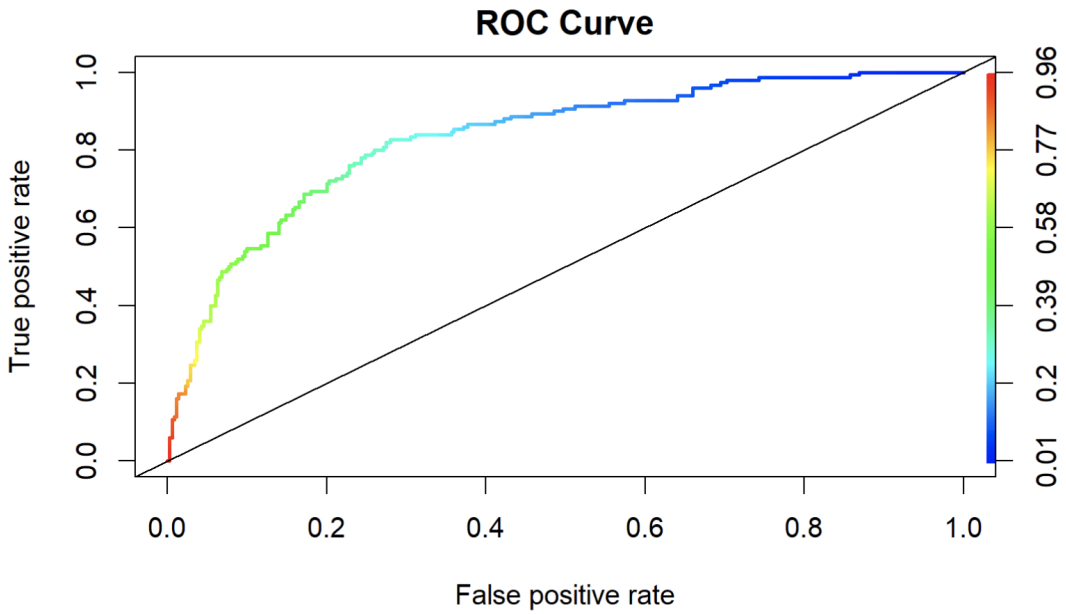
| Prediction \ Reference | 0 | 1 |
|---|---|---|
| 0 | 427 | 93 |
| 1 | 58 | 122 |

**Sensitivity: 0.5674**

**Specificity: 0.8804**

**OER: 0.216**

**AUC: 0.8479**




## QDA Results

**Confusion Matrix:**

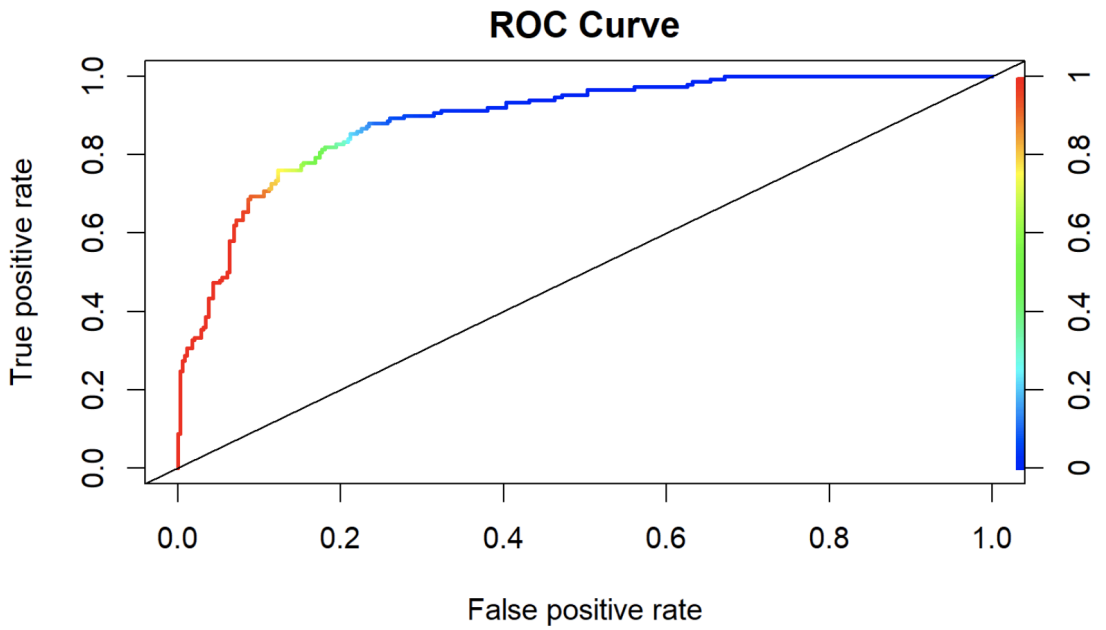
| Prediction \ Reference | 0 | 1 |
|---|---|---|
| 0 | 407 | 52 |
| 1 | 78 | 163 |

**Sensitivity: 0.7581**

**Specificity: 0.8392**

**OER: 0.186**

**AUC: 0.8797**




Logistic Regression Results

**Confusion Matrix:**

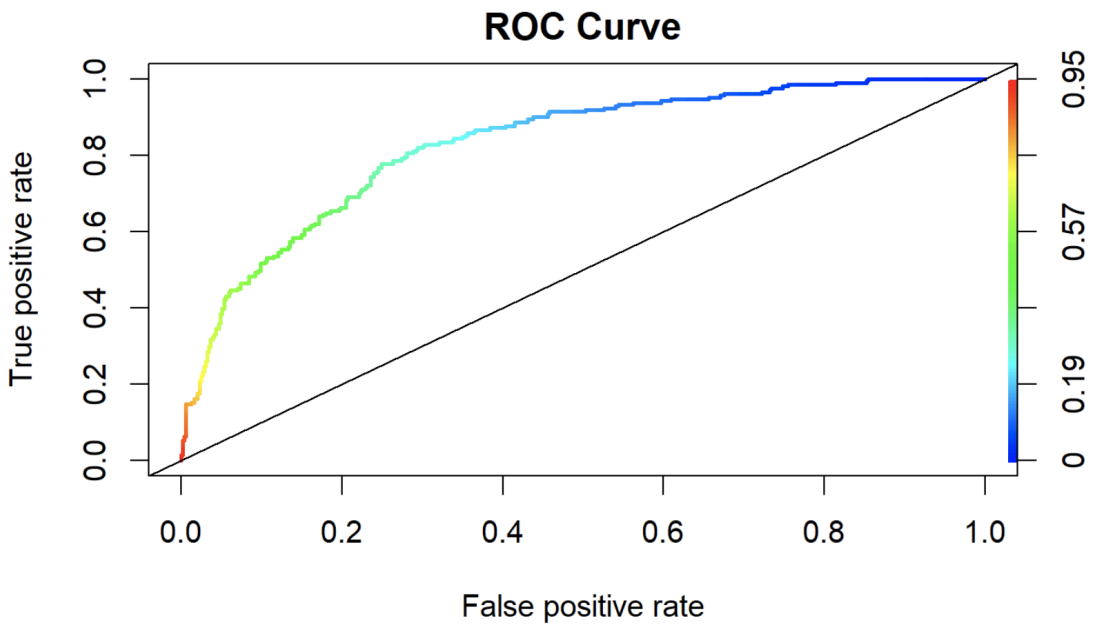
| Prediction \ Reference | 0 | 1 |
|---|---|---|
| 0 | 432 | 98 |
| 1 | 57 | 113 |

**Sensitivity: 0.883**

**Specificity: 0.535**

**OER: 0.221**

**AUC: 0.829**




Naive Bayes Results

**Confusion Matrix:**

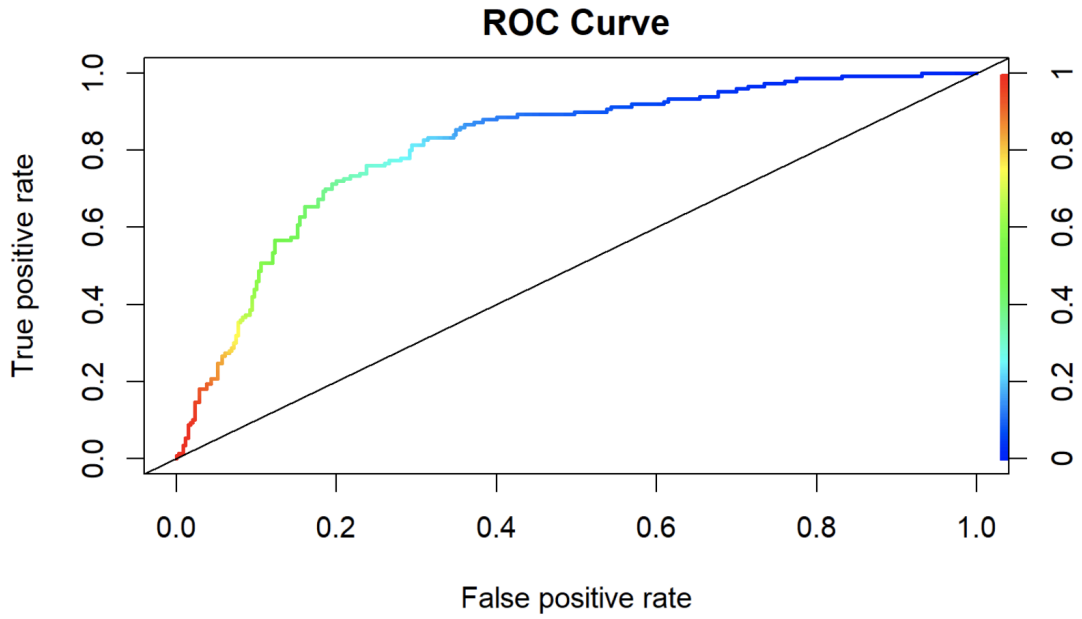
| Prediction \ Reference | 0 | 1 |
|---|---|---|
| 0 | 425 | 103 |
| 1 | 60 | 112 |

**Sensitivity: 0.5209**

**Specificity: 0.8762**

**OER: 0.233**

**AUC: 0.8193**




K Nearest Neighbors Results

**Confusion Matrix:**

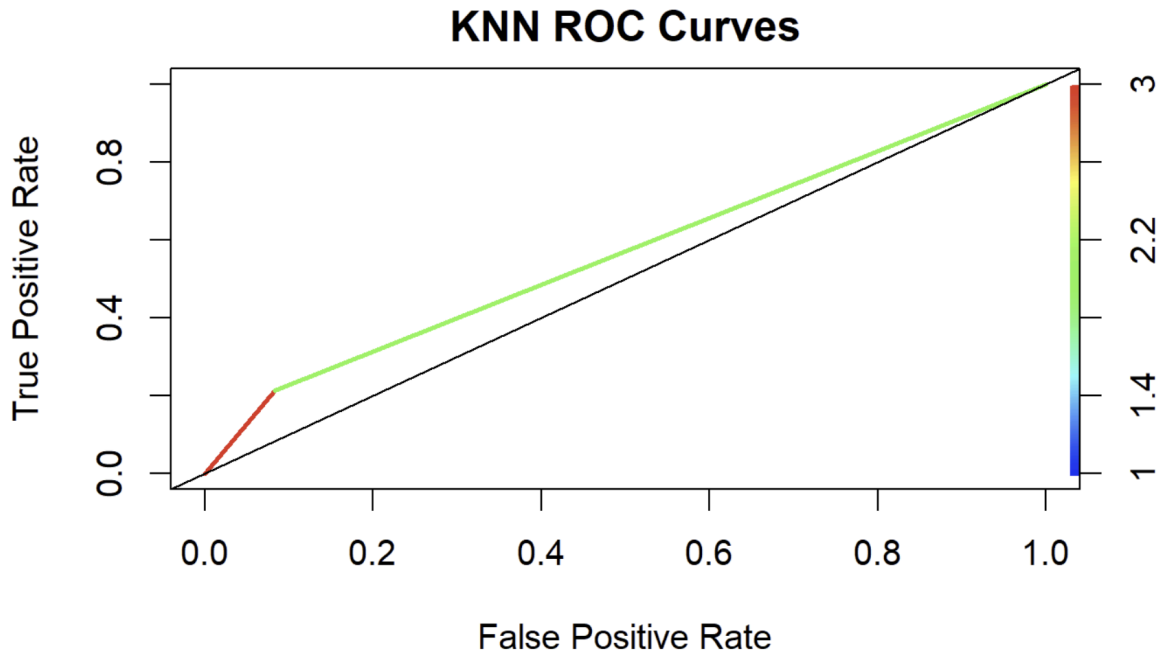
| Prediction \ Reference | 0 | 1 |
|---|---|---|
| 0 | 66 | 22 |
| 1 | 6 | 6 |

**Sensitivity: 0.2143**

**Specificity: 0.9167**

**OER: 0.28**

**AUC: 0.5655**




## Results Conclusion

# Model Results

| | SPEC | SENS | OER | AUC |
|---|---|---|---|---|
| LDA | 0.8804 | 0.5674 | 0.216 | 0.8479 |
| QDA | 0.8392 | 0.7581 | 0.186 | 0.8797 |
| NAIVE-B | 0.8763 | 0.5209 | 0.233 | 0.8193 |
| KNN | 0.9167 | 0.2143 | 0.280 | 0.566 |
| LOG | 0.8994 | 0.5070 | 0.220 | 0.8343 |

QDA performed the best out of the five models. It has the lowest OER so it has the lowest rate of misclassifying a prospective client. It has the highest AUC so this model fits the data the best out of the others.

## Alternative Results

### Additional Experiment Results (using only log reg significant variables)

| | SPEC | SENS | OER | AUC |
|---|---|---|---|---|
| LDA | 0.9228 | 0.3267 | 0.256 | 0.6248 |
| QDA | 0.8943 | 0.4133 | 0.250 | 0.6538 |
| NAIVE-B | 0.9867 | 0.9743 | 0.022 | 0.9805 |
| KNN | 0.9685 | 0.0933 | 0.294 | 0.5309 |
| LOG | 0.9286 | 0.3267 | 0.252 | 0.6276 |

Here we used just the significant variables discovered when running the summary function of a Logistic Regression model. This method preferred the Naive Bayes model.

## Group Challenges

We faced several challenges while conducting the analysis of the data, including selecting the proper platform to communicate on (Discord private chat), working with git + R Studio for the first time, selecting toolsets to whiteboard ideas (Excalidraw), selecting where to write down the project paper (Google docs), utilizing the data dictionary defining the classified variable name, building out the presentation slides (Google slides), and more.

Since novelty was initially expressed as being a goal of this project, our group at first focused efforts on utilizing dimensionality reduction methods like RFE to extract feature importances and PCA for dimensionality reduction, but later were told to scrap those parts of the project when checking in mid way with the professor. After discovering this, we reassessed on our group chat and realized that we were able to keep our originally established individual tasks.

It was challenging to explain the need to "tell the story of the data," which required detailing what are the prominent characteristics of a client that are most strongly correlated (positively or negatively) with a propensity to default. This is a bit of an ambiguous task that requires exploration of the data without knowing what the end goal will be. Taking on this task was challenging.

Furthermore, our group initially selected our primary metric for model selection as being whichever one causes a decrease in false positive rates since the official data documentation website that the professor gave them states, "It is worse to class a customer as good when they

are bad, than it is to class a customer as bad when they are good." After our check in with the professor we were told to scrap that using OEC as the most important metric. We corrected our approach after consulting with the professor.

Overall, the group individually and collectively worked hard together to utilize various methods to develop models and faced several challenges during the analysis process.