DATA MINING

German Credit Risk Analysis

By:
Asli Kurt

INTRODUCTION TO THE DATASET

German Credit

This case encapsulates a pivotal phase in the evolution of predictive modeling, characterized by the manual classification of Records into categories of 'ACCEPT’ or ‘REJECT’ credit by human assessors.

SELECTED DATASET

German Credit

German Credit dataset is a popular choice for machine learning and data analysis projects

Why

- ✓ Realistic Business Use Case
- ✓ Well-Structured for Classification
- ✓ Manageable Size
- ✓ Diverse Feature Types
- ✓ Widely Used in Academia
- ✓ Ethical and Interpretability Challenges

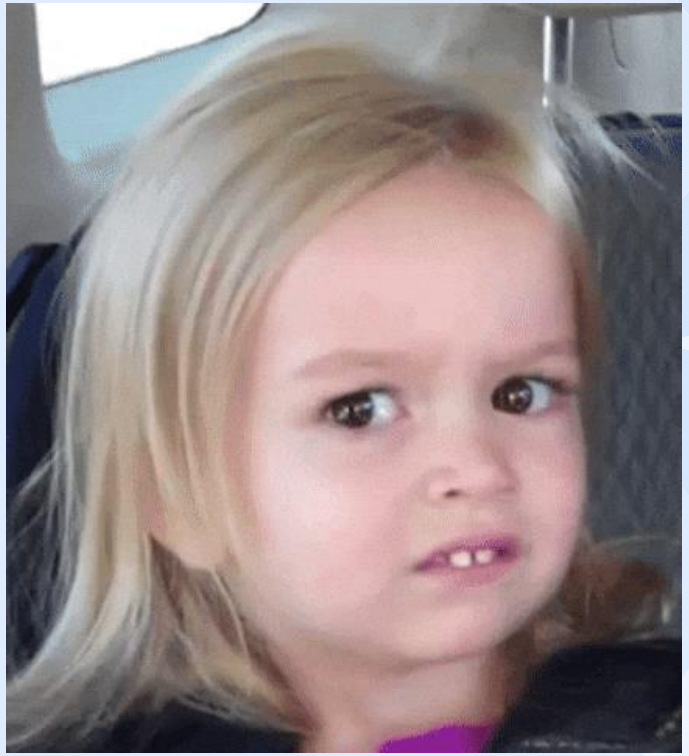
PREDICTOR VARIABLES

1000 Observations

30 Predictors **1 Response**

German Credit Dataset – Column Names

OBS.	EDUCATION	MALE_MAR_or WID
CHK_ACCT	RETRAINING	GUARANTOR
DURATION	AMOUNT	PRESENT_RESIDENT
HISTORY	SAV_ACCT	REAL_ESTATE
NEW_CAR	EMPLOYMENT	PROP_UNKN_NONE
USED_CAR	INSTALL_RATE	RENT
FURNITURE	MALE_DIV	NUM_CREDITS
RADIO.TV	MALE_SINGLE	JOB
ℓLCIATION	CLASS	NUM_DEPENDENTS



OUTLINE

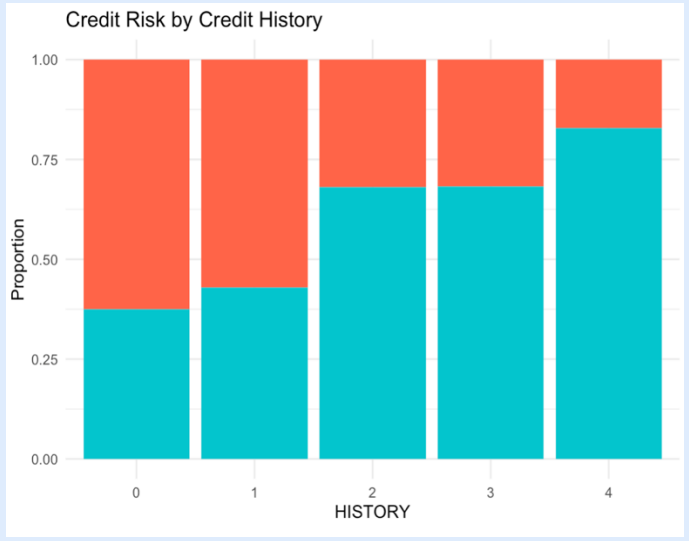
- **Objective**: Maximize net profit from loans
- **Data**: Split into training/validation

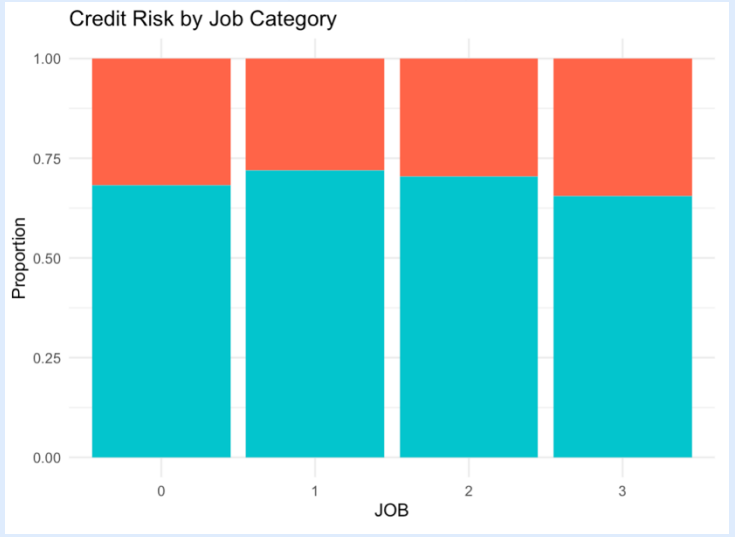
- **Models Tested**
  - Logistic Regression
  - Decision Tree
  - K-NN
  - Neural Network
- **Evaluation**
  - Compared confusion matrices
  - Metrics: Accuracy, Sensitivity, Specificity, Kappa
- **Threshold Tuning**
- **Net Profit**
- **Results**

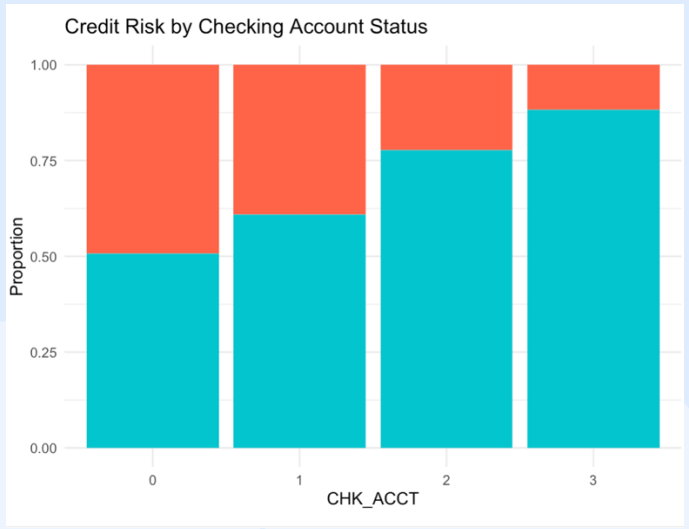
RESPONSE BY CATEGORICAL VARIABLES

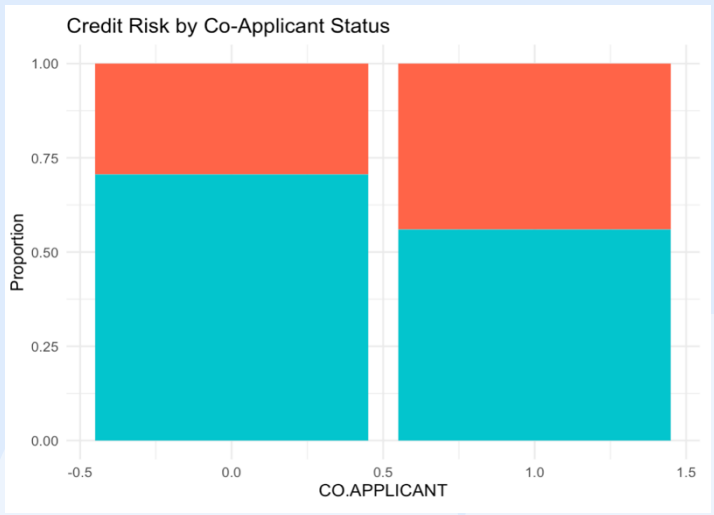
We used CHK_ACCT, HISTORY, and EMPLOYMENT in our final model as they show strong patterns related to credit outcomes.

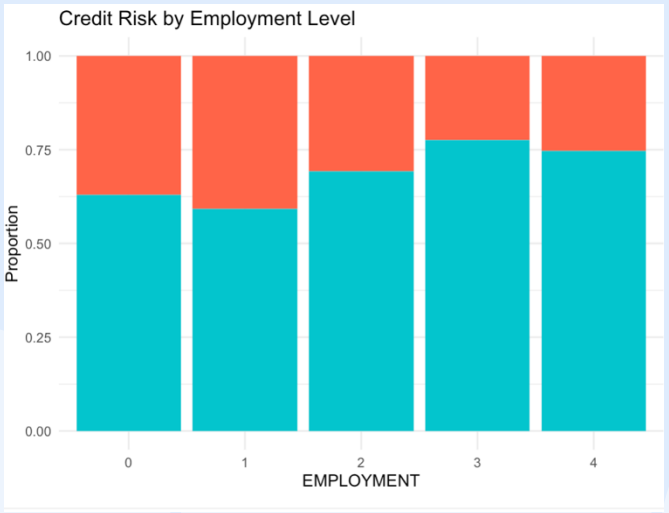
- We considered including SAV_ACCT and CO.APPLICANT based on model testing.
- JOB is not a strong variable because it does not significantly affect the responses made by credit risks.











INSIGHTS REVEALED BY THE DATASET

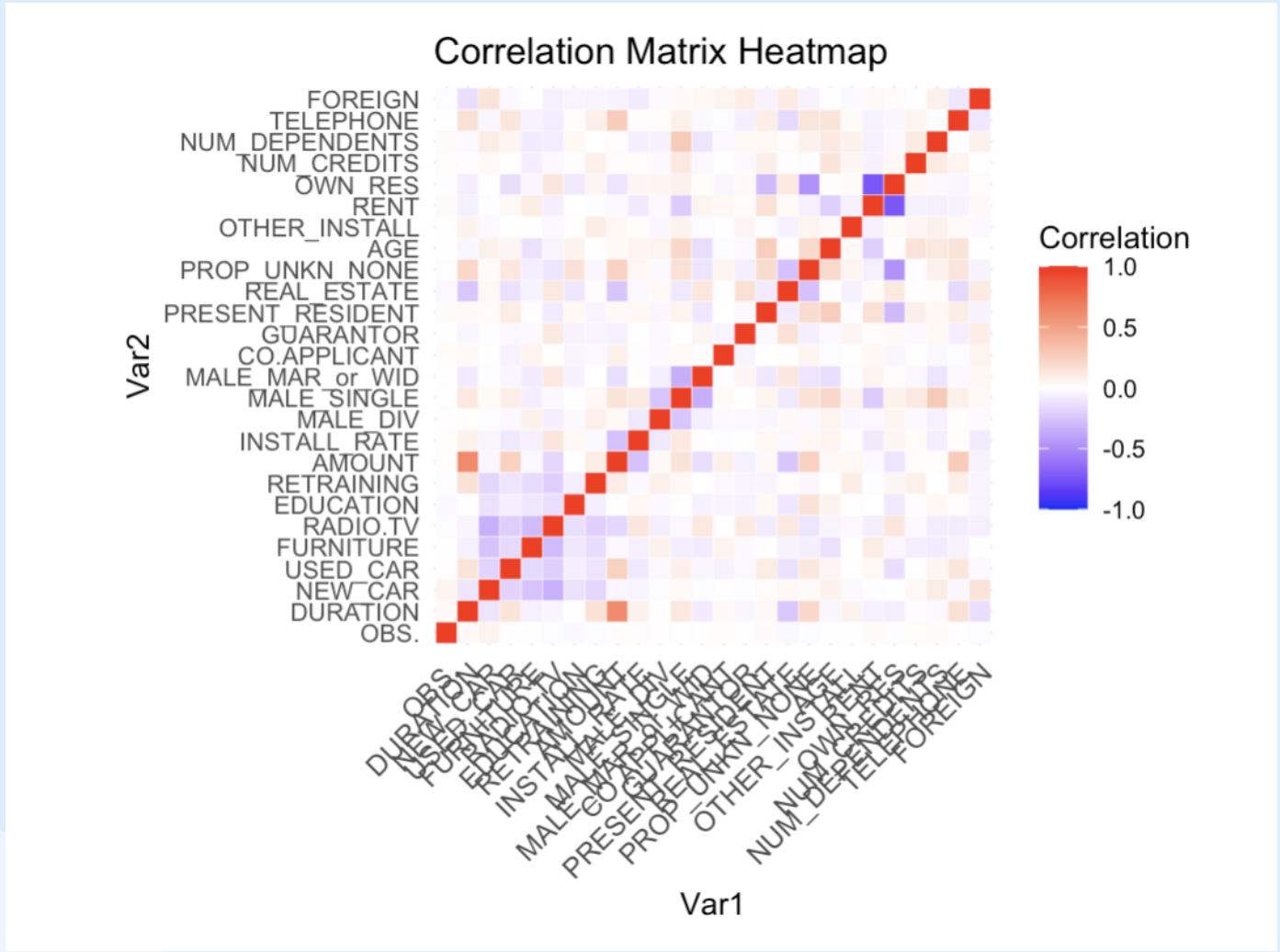
These 3 numeric variables (AGE, AMOUNT, DURATION) are strong variables for our classification models. Each one shows meaningful trends when compared against credit risks.

- AMOUNT and DURATION show the strongest separation between Accept/Reject.
- AGE also helps, especially in classifying more reliable clients.

CORRELATION ANALYSIS SUMMARY

In this analysis, we visualized the correlation matrix of all numeric variables using a heatmap.

This helped us understand how features relate to each other and identify multicollinearity risks.



CORRELATION OUTCOMES

- OWN_RES and RENT are highly negatively correlated (~ -0.74), which makes sense: you usually don't own and rent at the same time.
- DURATION and AMOUNT show a moderate positive correlation (~0.62), which is expected — longer durations often involve larger loan amounts.
- MALE_SINGLE and MALE_MAR_or_WID are negatively related (~ -0.35), since these categories are logically exclusive.
- INSTALL_RATE and AMOUNT are negatively correlated (~ -0.27), meaning loans with higher amounts tend to have lower installment rates.
- AGE shows mild positive correlation with variables like PRESENT_RESIDENT and NUM_CREDITS — older people tend to stay longer and may have more credit experience.
- Modeling Insight:
- No pairs were found with high correlation above 0.85, so no immediate need to drop variables.
- However, it's still good to check multicollinearity in logistic regression using VIF (Variance Inflation Factor).

	Var1 <fctr>	Var2 <fctr>	value <dbl>	Strength <fctr>
542	OWN_RES	RENT	-0.7359677	Strong
35	AMOUNT	DURATION	0.6249842	Moderate
464	OWN_RES	PROP_UNKN_NONE	-0.4764963	Moderate

Model #1 - Logistic Regression

Variance Inflation Factor (VIF)

Based on manual VIF analysis, most predictors show no multicollinearity issues. However, 'RENT' and 'OWN_RES' show moderate collinearity (VIF ~5-6), likely due to their inverse relationship.

OBS.	DURATION	NEW_CAR	USED_CAR	FURNITURE	RADIO.TV	EDUCATION
1.034774	2.060540	4.571889	2.861766	3.769703	4.886764	1.898842
RETRAINING	AMOUNT	INSTALL_RATE	MALE_DIV	MALE_SINGLE	MALE_MAR_or_WID	CO.APPLICANT
2.671560	2.425032	1.369939	1.165336	1.614019	1.233057	1.060047
GUARANTOR	PRESENT_RESIDENT	REAL_ESTATE	PROP_UNKN_NONE	AGE	OTHER_INSTALL	RENT
1.084916	1.228502	1.243205	2.609581	1.326559	1.041665	5.036730
OWN_RES	NUM_CREDITS	NUM_DEPENDENTS	TELEPHONE	FOREIGN		
6.006377	1.064626	1.175805	1.188569	1.135827		

LOGISTIC REGRESSION MODEL SUMMARY

- Accuracy: 73% – decent
- Sensitivity: 83.8% – good at finding *Accepts*
- Specificity: 47.8% – weak at finding *Rejects*
- Balanced Accuracy: 65.8%
- Kappa: 0.33 – moderate agreement

Potential Issues:

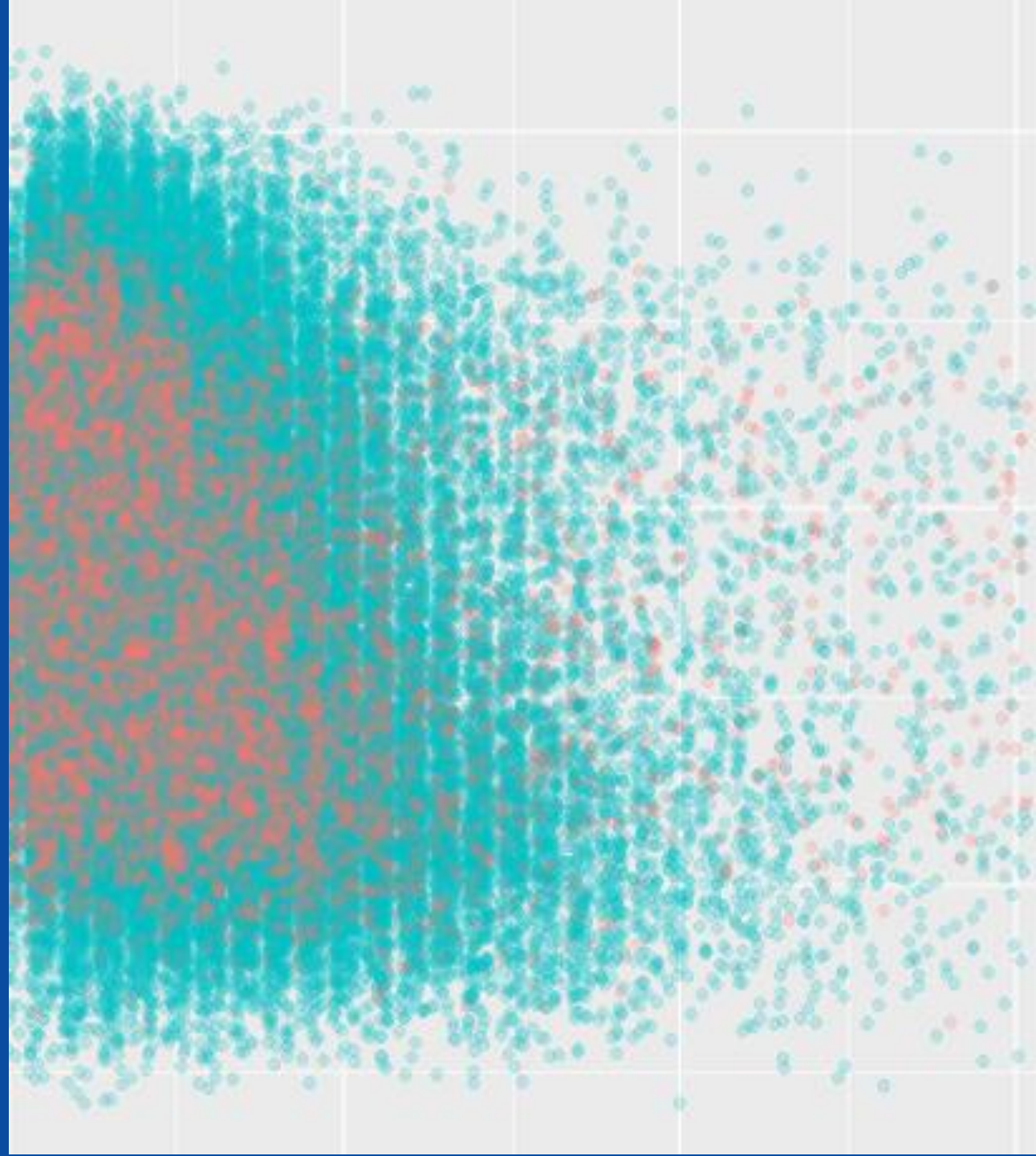
•Bias toward "Accept" decisions
•Class imbalance (70% Accept / 30% Reject) skews results
•Risk of approving bad credit – potential financial loss

Conclusion:

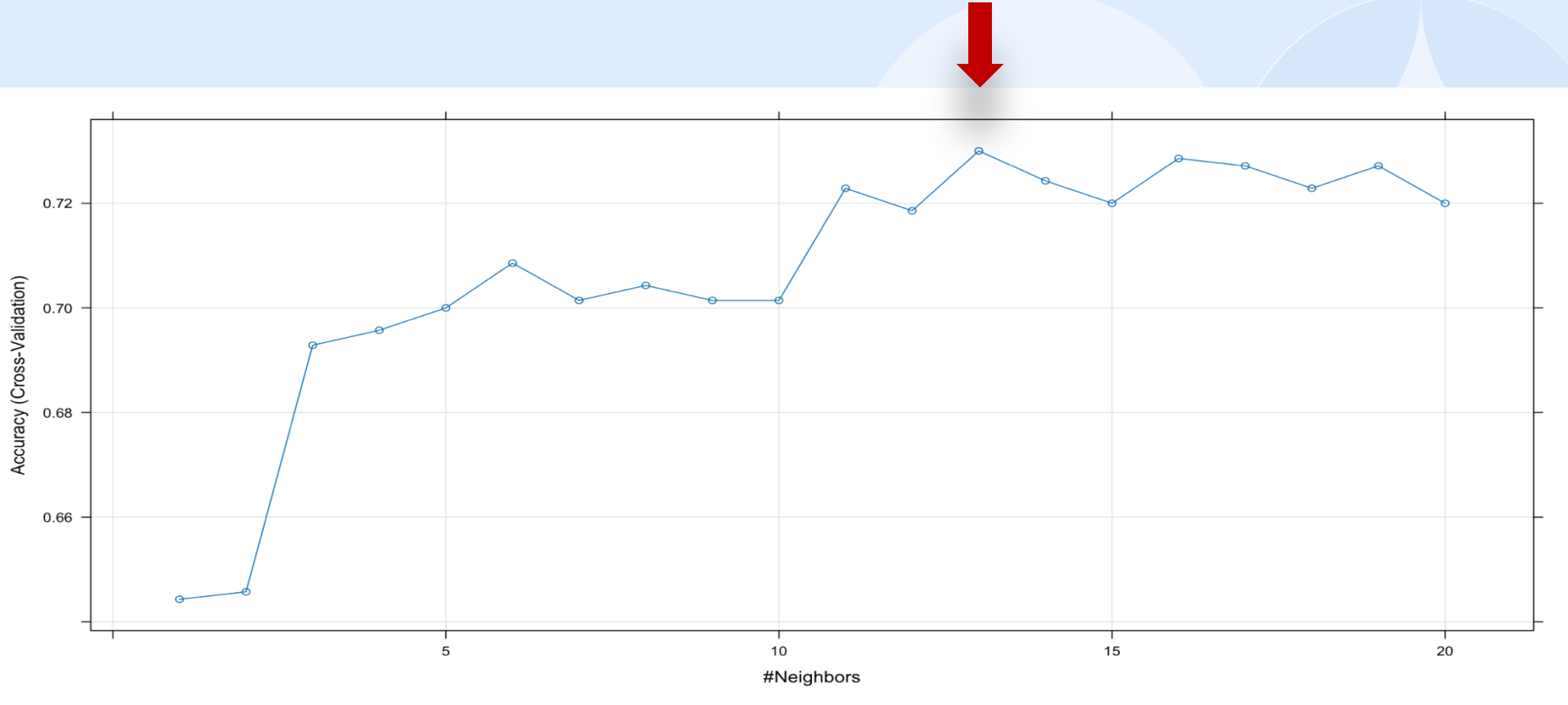
✓ Good for **low-risk lending**
✓ Not ideal for **high-risk or high-value** credit decisions

```
[1] "CONFUSION MATRIX: TEST/VALIDATION DATA"
Confusion Matrix and Statistics

          Reference
Prediction Reject Accept
    Reject     43     34
    Accept     47    176

               Accuracy : 0.73
                 95% CI : (0.676, 0.7794)
    No Information Rate : 0.7
    P-Value [Acc > NIR] : 0.1418

                  Kappa : 0.3295

 Mcnemar's Test P-Value : 0.1824

            Sensitivity : 0.8381
            Specificity : 0.4778
         Pos Pred Value : 0.7892
         Neg Pred Value : 0.5584
             Prevalence : 0.7000
         Detection Rate : 0.5867
   Detection Prevalence : 0.7433
      Balanced Accuracy : 0.6579

       'Positive' Class : Accept
```

Model #2 – K-Nearest Neighbors



BEST K VALUE



BEST K= 13

K-NN Model Summary

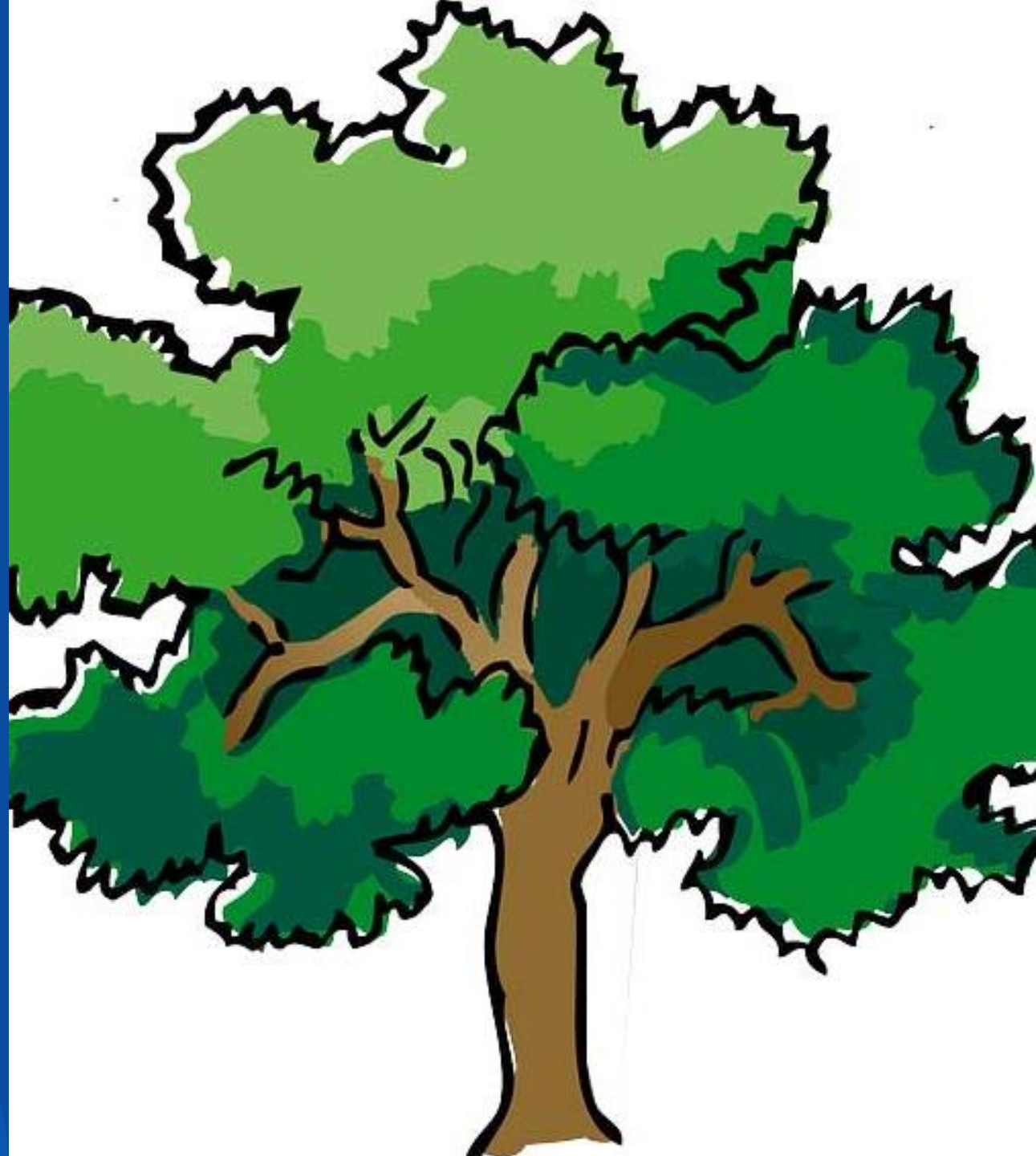
- Accuracy: 70.3% – Decent
- Sensitivity: 92.4% – Strong at classifying good credit clients
- Specificity: 18.9% – Weak at classifying bad credit clients

Potential Issues:
•Strong bias toward predicting "Accept"
•Very poor at identifying "Reject" cases
•Class imbalance (70% Accept / 30% Reject) amplifies this bias

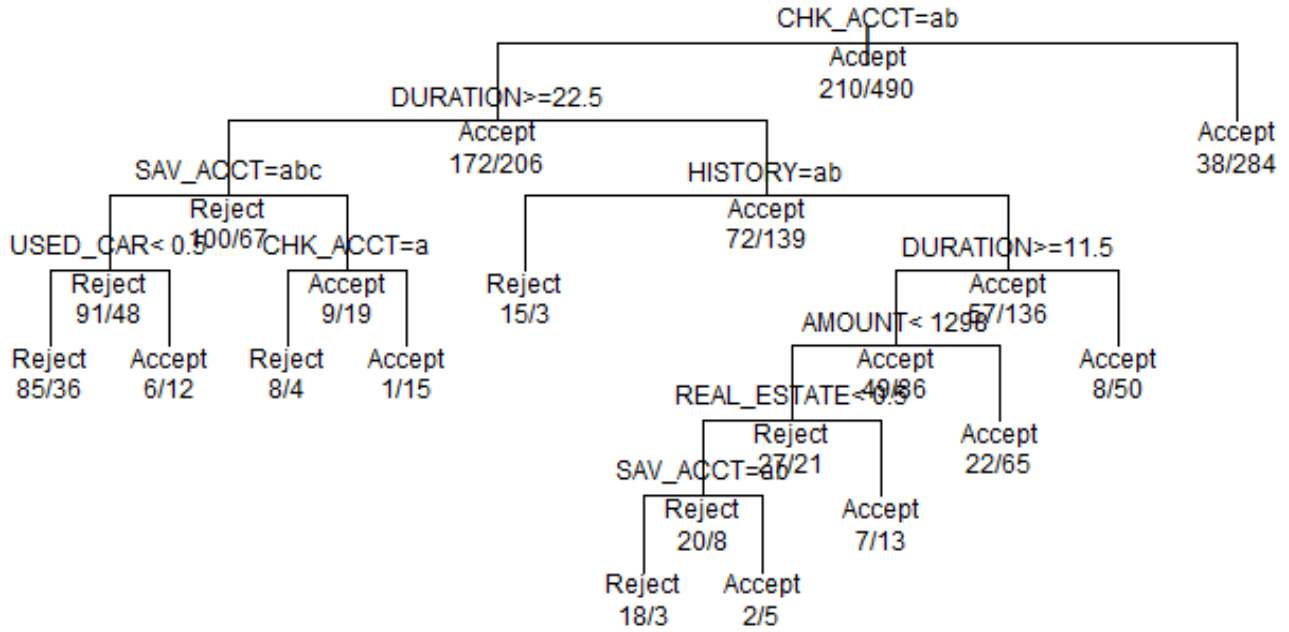
Conclusion:
✓ Useful for maximizing approvals in low-risk scenarios
✓ Not suitable for credit risk control — too risky for high-value or sensitive decisions

```
[1] "CONFUSION MATRIX: VALIDATION DATA (KNN)"
Confusion Matrix and Statistics

          Reference
Prediction Reject Accept
    Reject     17     16
    Accept     73    194

               Accuracy : 0.7033
                 95% CI : (0.6481, 0.7545)
    No Information Rate : 0.7
    P-Value [Acc > NIR] : 0.4782

                  Kappa : 0.1376

 Mcnemar's Test P-Value : 2.921e-09

            Sensitivity : 0.9238
            Specificity : 0.1889
         Pos Pred Value : 0.7266
         Neg Pred Value : 0.5152
             Prevalence : 0.7000
         Detection Rate : 0.6467
   Detection Prevalence : 0.8900
      Balanced Accuracy : 0.5563

       'Positive' Class : Accept
```

Model #3 – Decision Tree



CLASSIFICATION TREE



DECISION TREE MODEL SUMMARY

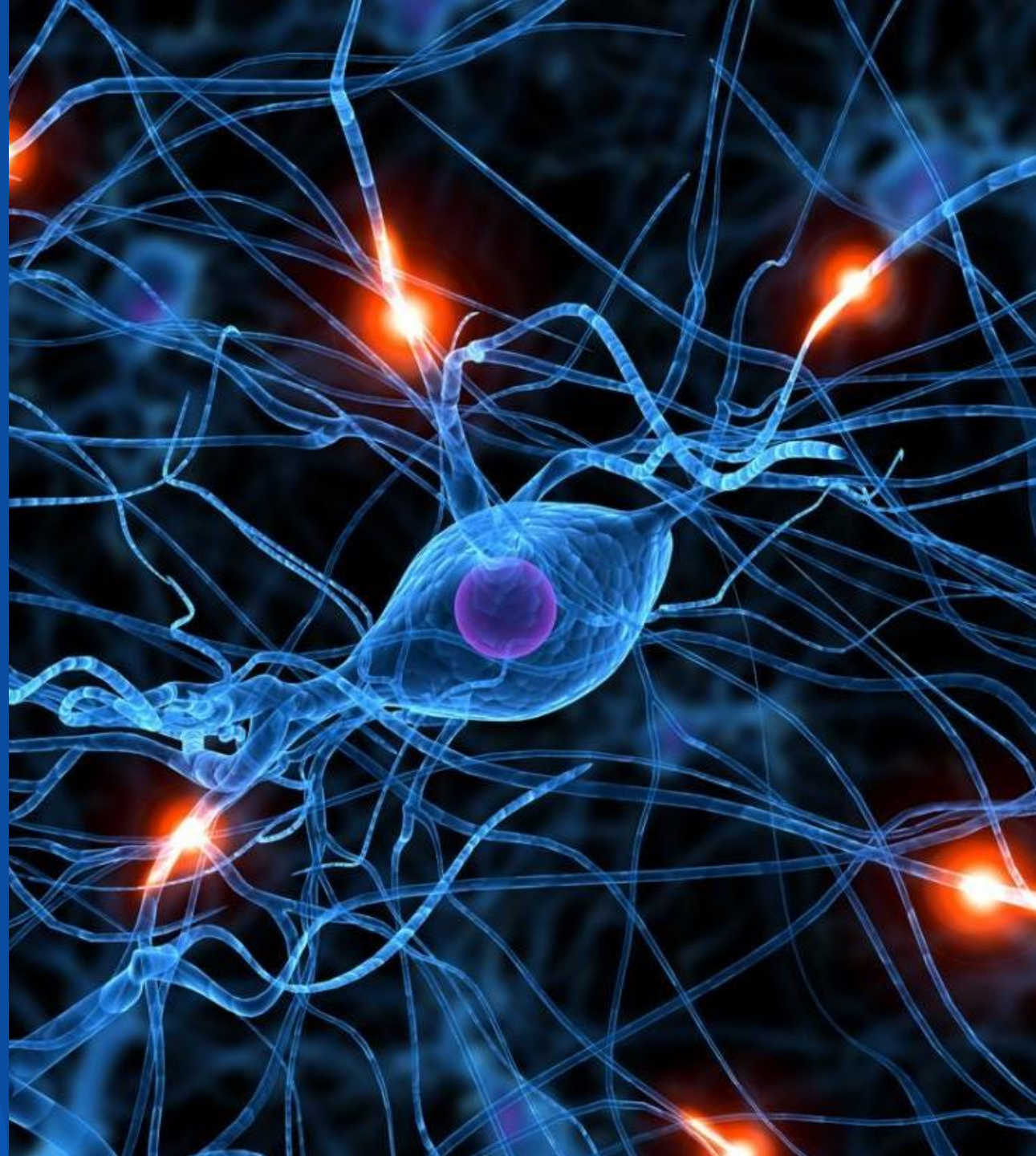
- Accuracy: 72.3% – decent
- Sensitivity: 84.8% – good at finding *Accepts*
- Specificity: 43.3% – weak at finding *Rejects*

Potential Issues: Bias toward "Accept" predictions
•Class imbalance (70% Accept / 30% Reject) affects performance
•High false negative classification rate – risk of rejecting good applicants

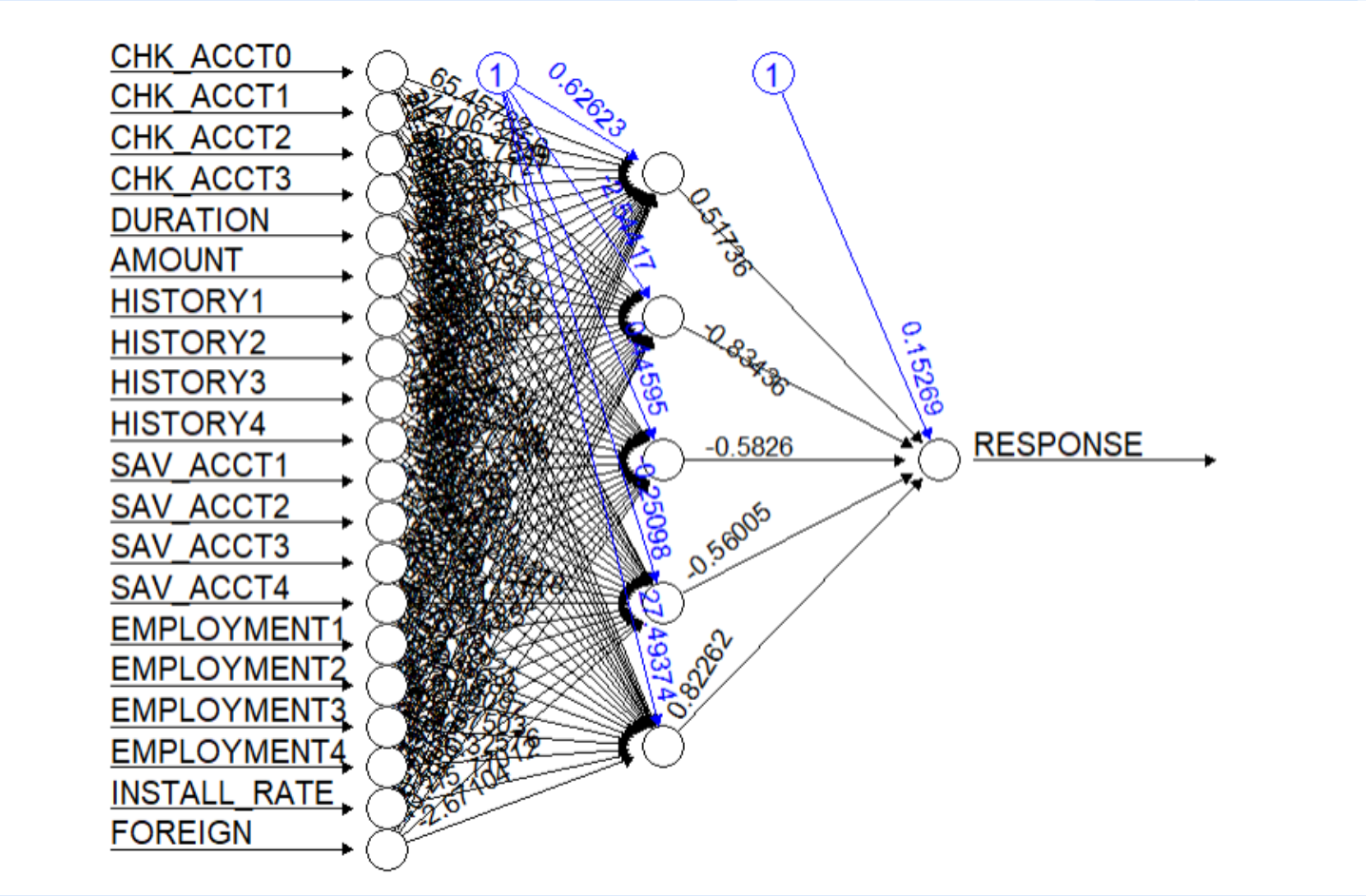
Conclusion:
✓ Suitable for campaigns focused on approvals or growth
✓ **Not reliable** for strict credit risk screening or high-stakes lending

```
[1] "CONFUSION MATRIX: TEST/VALIDATION DATA (Decision Tree)"
Confusion Matrix and Statistics

          Reference
Prediction Reject Accept
    Reject     39     32
    Accept     51    178

               Accuracy : 0.7233
                 95% CI : (0.669, 0.7732)
    No Information Rate : 0.7
    P-Value [Acc > NIR] : 0.20722

                  Kappa : 0.299

 Mcnemar's Test P-Value : 0.04818

            Sensitivity : 0.8476
            Specificity : 0.4333
         Pos Pred Value : 0.7773
         Neg Pred Value : 0.5493
             Prevalence : 0.7000
         Detection Rate : 0.5933
   Detection Prevalence : 0.7633
      Balanced Accuracy : 0.6405

       'Positive' Class : Accept
```

Model #4 – Neural Network



NEURAL NETWORK



NEURAL NETWORK MODEL SUMMARY

- Accuracy: 71.3% – decent
- Sensitivity: 77.6% – strong at finding *Accepts*
- Specificity: 56.7% – moderate at finding *Rejects*
- Balanced Accuracy: 67.1%
- Kappa: 0.33 – moderate agreement

Potential Issues:

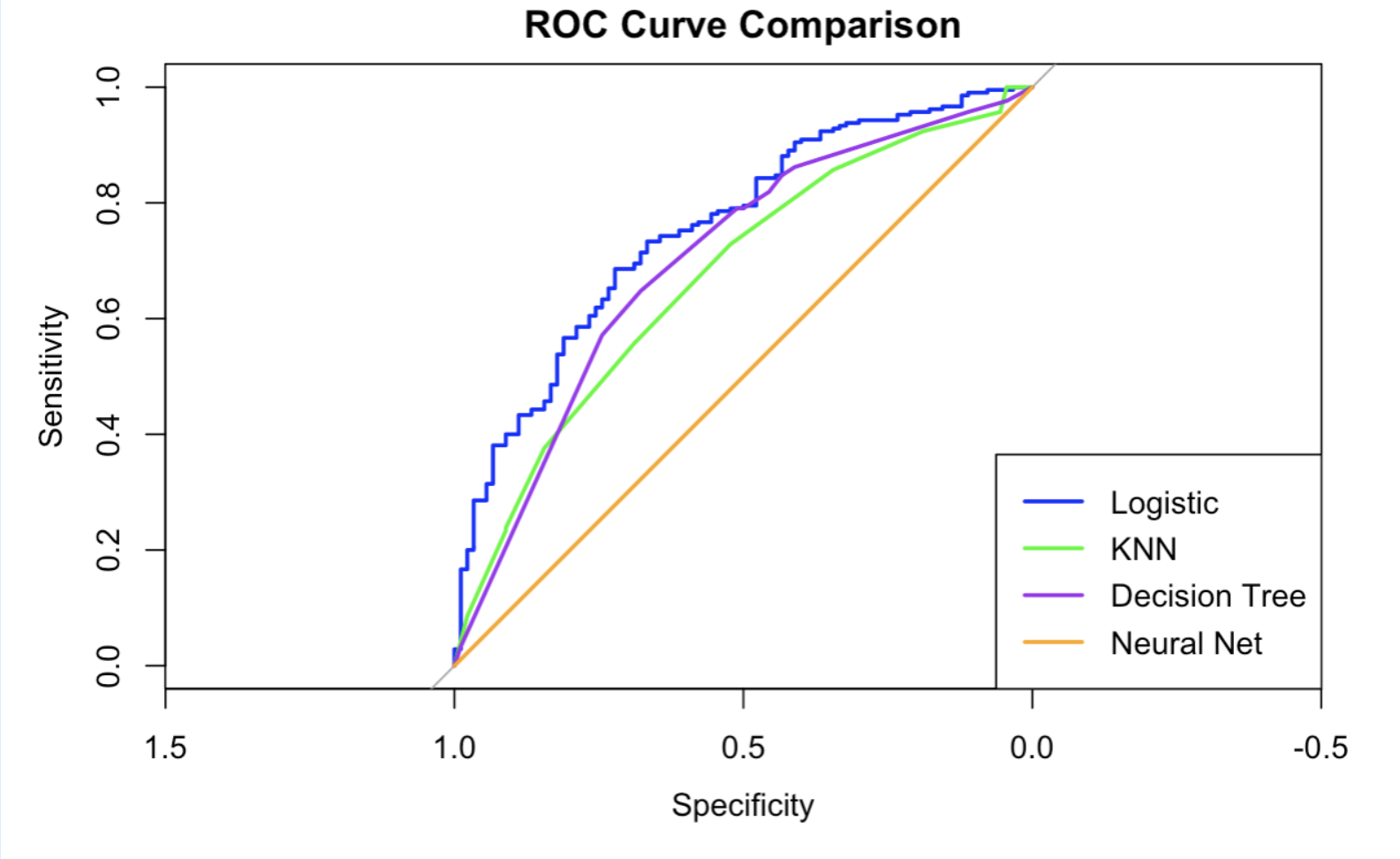
- Slight bias toward predicting "Accept"
- Better balance between classes compared to Decision Tree and KNN
- Some false positives remain – potential for approving unqualified applicants

Conclusion:

✓ Good all-around model for **balanced lending decisions**
✓ Suitable for **moderate-risk credit environments**
✓ More reliable than KNN when both approval and rejection accuracy matter

```
[1] "CONFUSION MATRIX: TEST/VALIDATION DATA (Neural Net)"
Confusion Matrix and Statistics

          Reference
Prediction Reject Accept
    Reject     51     47
    Accept     39    163

               Accuracy : 0.7133
                 95% CI : (0.6586, 0.7638)
    No Information Rate : 0.7
    P-Value [Acc > NIR] : 0.3321

                  Kappa : 0.3344

 Mcnemar's Test P-Value : 0.4504

            Sensitivity : 0.7762
            Specificity : 0.5667
         Pos Pred Value : 0.8069
         Neg Pred Value : 0.5204
             Prevalence : 0.7000
         Detection Rate : 0.5433
   Detection Prevalence : 0.6733
      Balanced Accuracy : 0.6714

       'Positive' Class : Accept
```

ROC CURVE



Based on the ROC curves, the **Decision Tree and Logistic Regression** models are preferred choices for classification performance on this dataset.

For further improvements, ensemble approaches or hyperparameter tuning (especially for Neural Net) can be considered.

AUC Score Summary

Accuracy measures how many predictions we got right after setting a decision threshold, while AUC measures how well we ranked positives higher than negatives, no matter where we set the cutoff — meaning AUC focuses on the model's ability to separate the two groups, while accuracy focuses on final decisions. Focuses on the true-positives.

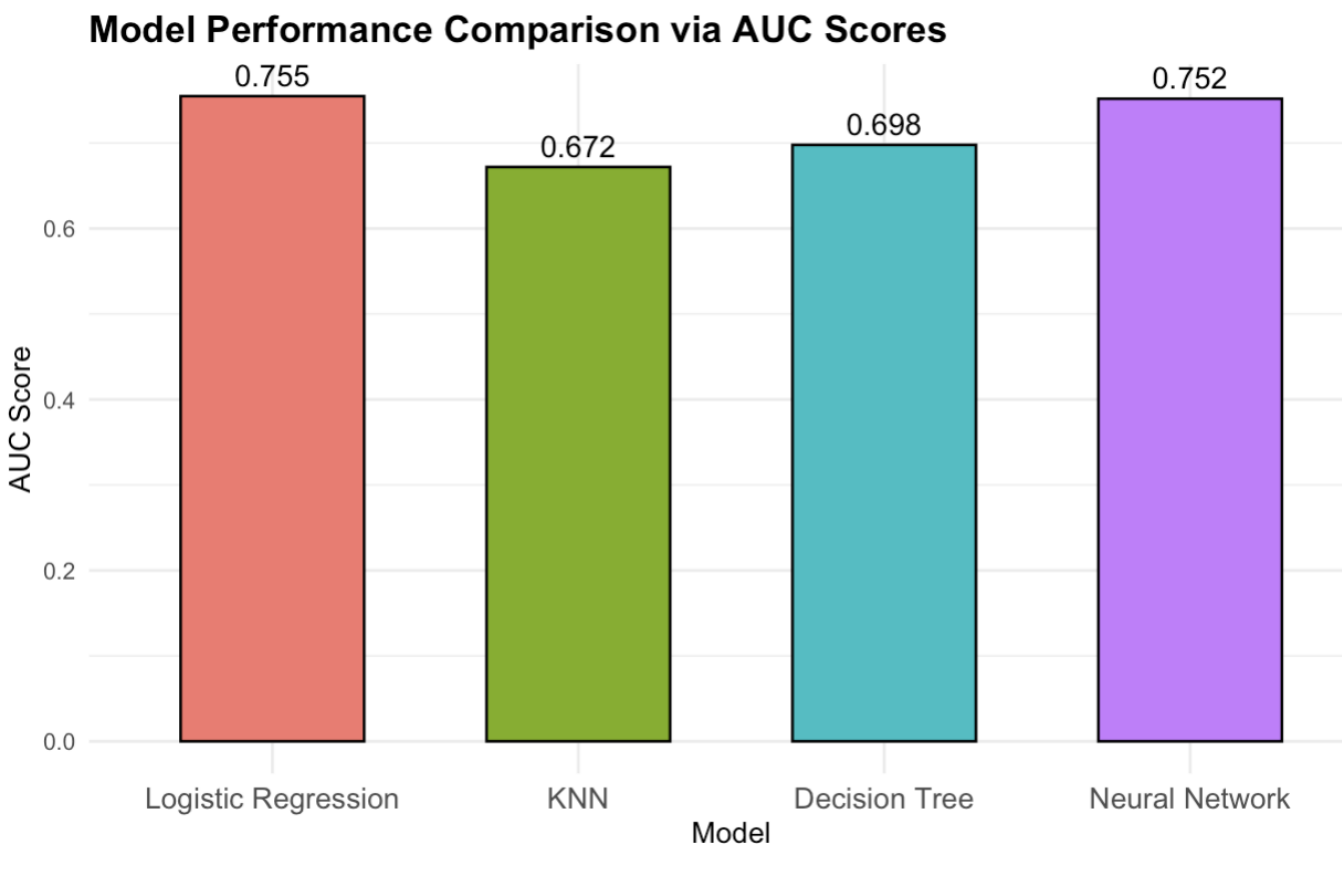
Model <chr>	AUC <dbl>
Logistic Regression	0.7548677
K-Nearest Neighbors	0.6719841
Decision Tree	0.6978571
Neural Network	0.7517989

Conclusion

- ✓ **Logistic Regression is the most reliable in terms of classification ability.**
- ✓ **ROC and AUC analysis confirm it's the strongest performer on this dataset.**

- Logistic Regression AUC: Highest, best overall separation of classes.
- Decision Tree & Neural Net: Strong models, but have slightly lower readings.
- KNN: Lowest AUC, likely impacted by class imbalance or parameter tuning.

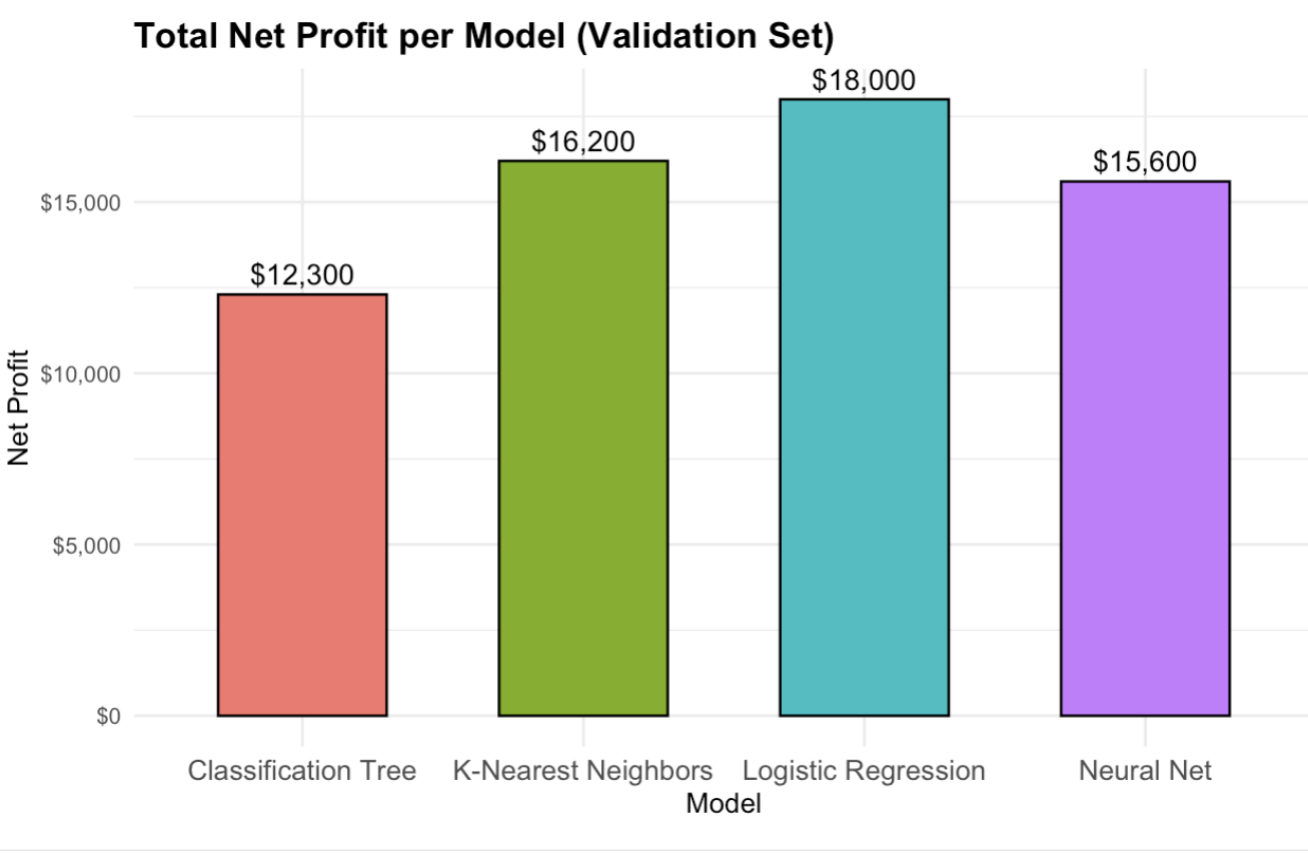
MODEL PERFORMANCE COMPARISON via AUC SCORES



- Logistic Regression achieved the highest AUC score of 0.755, suggesting strong classification performance.
- The Neural Network also performed well with an AUC of 0.752, indicating it may be a viable alternative.
- Decision Tree had a moderate performance with an AUC of 0.698.
- KNN performed the worst with an AUC of 0.672, suggesting that it may not be suitable for this dataset.

Overall, Logistic Regression appears to be the most effective model for this classification task. Depending on the context and need for model interpretability or complexity, Logistic Regression or Neural Network can be considered.

TOTAL NET PROFIT for PER MODEL



Total Net Profit Comparison (Validation Set)

- **Logistic Regression**: **$18,000** – Highest profit
- **K-Nearest Neighbors**: **$16,200** – Strong profit, but low specificity
- **Neural Net**: **$15,600** – Balanced model with solid returns
- **Classification Tree**: **$12,300** – Lowest profit, but interpretable

Conclusion:

Logistic Regression delivers the highest net profit, suggesting strong overall predictive performance and a favorable balance of approvals and rejections.

LOGISTIC REGRESSION DECILE VISUALIZATION

Decile 10: People the model was most confident about
Decile 1: People the model was least confident about

Focus lending on the top deciles and avoid or reconsider the lower ones to maximize profit.

Each bar shows the total profit the company would make from lending to applicants in that group.

Green bars = profitable groups
Red bars = groups that lose money

Top-scoring applicants (Deciles 10 to 6) are consistently profitable — the model's predictions are working well here.

#Bottom-scoring applicants (Deciles 1 to 4) lead to big losses — too many risky approvals.

PROFIT vs. THRESHOLD (Logistic Regression)

Maximum Profit Threshold: ~0.1–0.2
Conclusion: Approving more applicants (even with lower confidence) maximizes profit

Issue: However, this means that we will be approve all clients including clients that do not have good credit scores, which can lead to a loss of profit.

	Threshold <dbl>	Accuracy <dbl>	Sensitivity <dbl>	Specificity <dbl>	Net_Profit <dbl>
Accuracy	0.1	0.7066667	1.0000000	0.02222222	21000
Accuracy1	0.2	0.7166667	0.9904762	0.07777778	20400
Accuracy2	0.3	0.7366667	0.9523810	0.23333333	18000
Accuracy3	0.4	0.7500000	0.9095238	0.37777778	15300
Accuracy4	0.5	0.7300000	0.8380952	0.47777778	10800
Accuracy5	0.6	0.7100000	0.7666667	0.57777778	6300
Accuracy6	0.7	0.6900000	0.6904762	0.68888889	1500
Accuracy7	0.8	0.6433333	0.5809524	0.78888889	-5400
Accuracy8	0.9	0.5466667	0.3904762	0.91111111	-17400

9 rows

Why Profit Drops
Higher thresholds approve fewer people

Missed good customers → lower overall profit

However, less risk of approving bad credit clients.

Thus, we should aim for a profit threshold of around ~0.3

In that scenario, having a sense of caution while aiming for profitability leads to minimized risks.

CONCLUSION

Model <chr>	Accuracy <dbl>	Sensitivity <dbl>	Specificity <dbl>	True_Positives <int>	False_Positives <int>	Net_Profit <dbl>
Logistic Regression	0.7366667	0.9523810	0.23333333	200	10	18000
Classification Tree	0.7266667	0.8619048	0.41111111	181	29	12300
Neural Net	0.7366667	0.9142857	0.32222222	192	18	15600
K-Nearest Neighbors	0.7033333	0.9238095	0.02222222	194	16	16200

4 rows

- Here, we set a threshold of 0.3, meaning that the model will be under the assumption that 30% of clients will be approved instead of the default of 0.5 (50%).
- Based on our net profit model, we can see that Logistic Regression having the highest accuracy and sensitivity.
  - Despite its specificity not being the highest, it still yields the highest net profit
- Logistic Regression has classified the most amount of True_Positives while minimizing losses from False Positives which has led to it having a total Net_Profit of 18000.
- Thus, Logistic Regression should be our best model to use in a business standpoint.


THANK
YOU