Project Deliverable 4: Bank Marketing Dataset: EDA, Modeling, Clustering, and Pattern Mining

Sagar Bhetwal, Umesh Dhakal, Nirajan Acharya, Suresh Ghimire, Enjal Chauhan

University of the Cumberlands

Department of Computer Science

Advanced Big Data and Data Mining(MSCS634)

Dr. Satish Penmatsa

February 15th, 2026

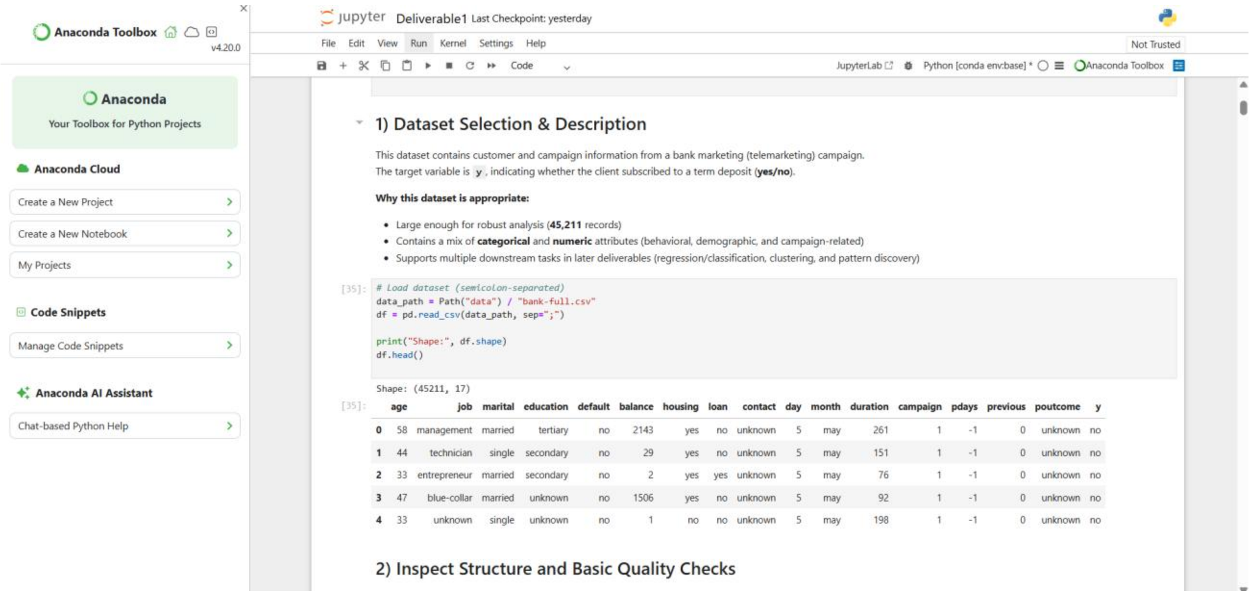
Project Deliverable 4: Bank Marketing Dataset: EDA, Modeling, Clustering, and Pattern Mining

Introduction

Our group chose a banking marketing dataset that looks at whether a customer decides to subscribe to a term deposit. We picked this dataset because it represents a real business problem and includes a large number of records with both numerical and categorical features. This made it a good fit for applying different data mining techniques covered in this course. The project was not just about building prediction models, but also about understanding the data, analyzing the results, and connecting what we found to real-world decision making.

Why we Chose Banking Data

Our team selected the bank marketing dataset because it provides a realistic example of how companies use customer and campaign data to make decisions. The dataset contains a large number of records along with mixed data types (numeric and categorical), which made it useful for practicing data preprocessing, feature engineering, and different machine learning techniques. Since the outcome is whether a customer subscribed to a term deposit, it also supports clear evaluation and practical insights. One thing we liked about this dataset is that it has a mix of customer info and campaign info, so we can study both customer behavior and marketing strategy. Also, the target column is clear (subscribe or not), so it makes it easier to evaluate models and compare methods.



1) Dataset Selection & Description

This dataset contains customer and campaign information from a bank marketing (telemarketing) campaign.
The target variable is `y`, indicating whether the client subscribed to a term deposit (**yes/no**).

Why this dataset is appropriate:

- Large enough for robust analysis (**45,211** records)
- Contains a mix of **categorical** and **numeric** attributes (behavioral, demographic, and campaign-related)
- Supports multiple downstream tasks in later deliverables (regression/classification, clustering, and pattern discovery)

[35]:
```
# Load dataset (semicolon-separated)
data_path = Path("data") / "bank-full.csv"
df = pd.read_csv(data_path, sep=";")

print("Shape:", df.shape)
df.head()
```

```
Shape: (45211, 17)
```

[35]:

	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	y
0	58	management	married	tertiary	no	2143	yes	no	unknown	5	may	261	1	-1	0	unknown	no
1	44	technician	single	secondary	no	29	yes	no	unknown	5	may	151	1	-1	0	unknown	no
2	33	entrepreneur	married	secondary	no	2	yes	yes	unknown	5	may	76	1	-1	0	unknown	no
3	47	blue-collar	married	unknown	no	1506	yes	no	unknown	5	may	92	1	-1	0	unknown	no
4	33	unknown	single	unknown	no	1	no	no	unknown	5	may	198	1	-1	0	unknown	no

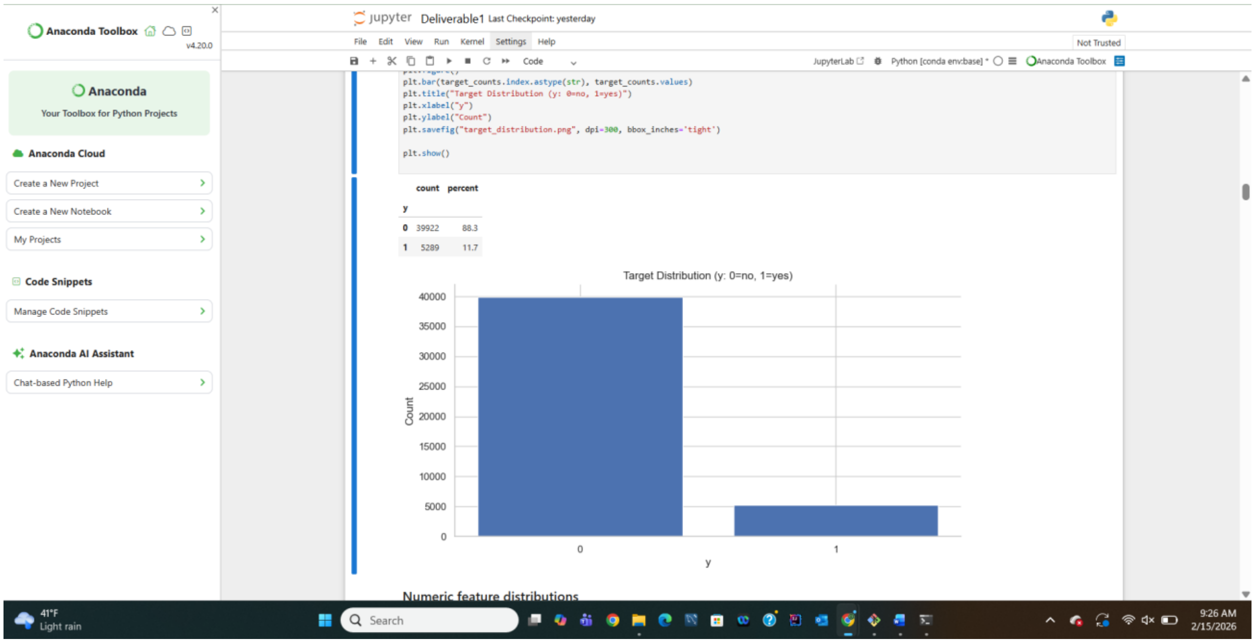
2) Inspect Structure and Basic Quality Checks

Figure 1. Bank marketing dataset

Data Cleaning and Data Analysis

During Deliverable 1, we focused on preparing the dataset so it could be used for modeling. We handled missing values carefully, especially in the pdays column. In this dataset, a missing pdays value means the customer was not contacted before. Instead of removing those rows, we created a flag variable to keep that information and filled the missing values in a way that would not affect the model. We also checked for duplicate records and looked at outliers in numerical features such as age, balance, duration, and campaign. Exploratory Data Analysis helped us better understand how the data is distributed, how features relate to each other, and how they connect to the target variable.

Some of the screenshots below show the target distribution, correlation heatmap, and numeric distributions.



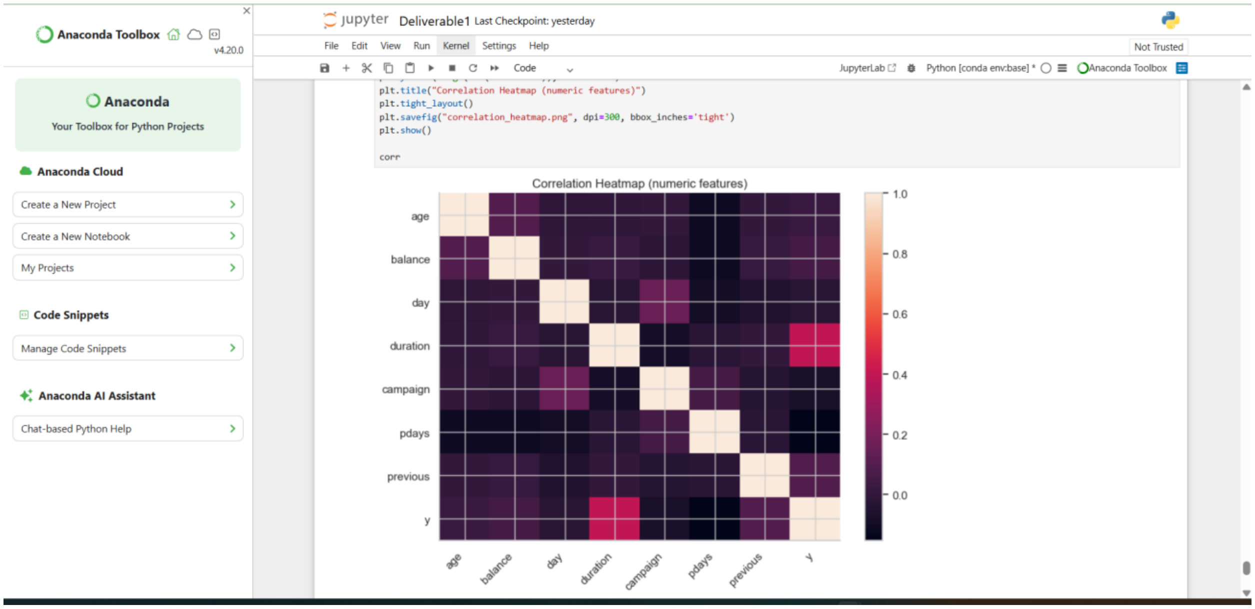
```
plt.bar(target_counts.index.astype(str), target_counts.values)
plt.title("Target Distribution (y: 0=no, 1=yes)")
plt.xlabel("y")
plt.ylabel("Count")
plt.savefig("target_distribution.png", dpi=300, bbox_inches='tight')

plt.show()
```

	count	percent
y		
0	39922	88.3
1	5289	11.7

Numeric feature distributions

Figure 2. Subscription outcome distribution



```
plt.title("Correlation Heatmap (numeric features)")
plt.tight_layout()
plt.savefig("correlation_heatmap.png", dpi=300, bbox_inches='tight')
plt.show()

corr
```

Figure 3. Correlation heatmap of numeric features

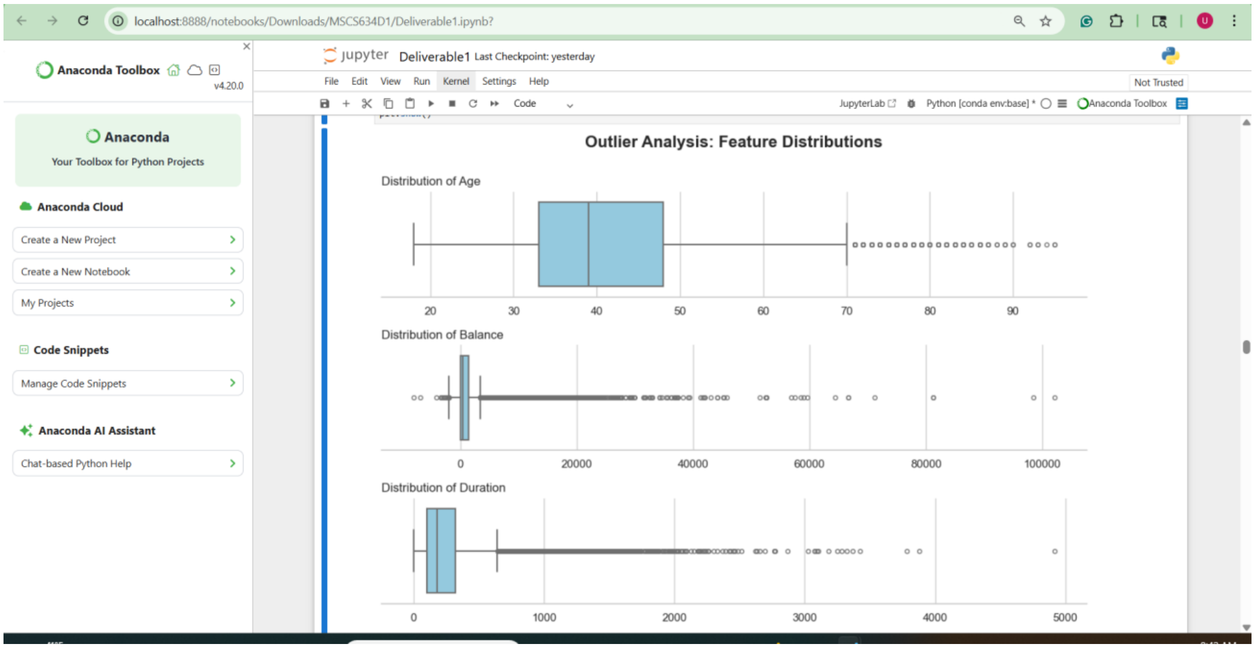
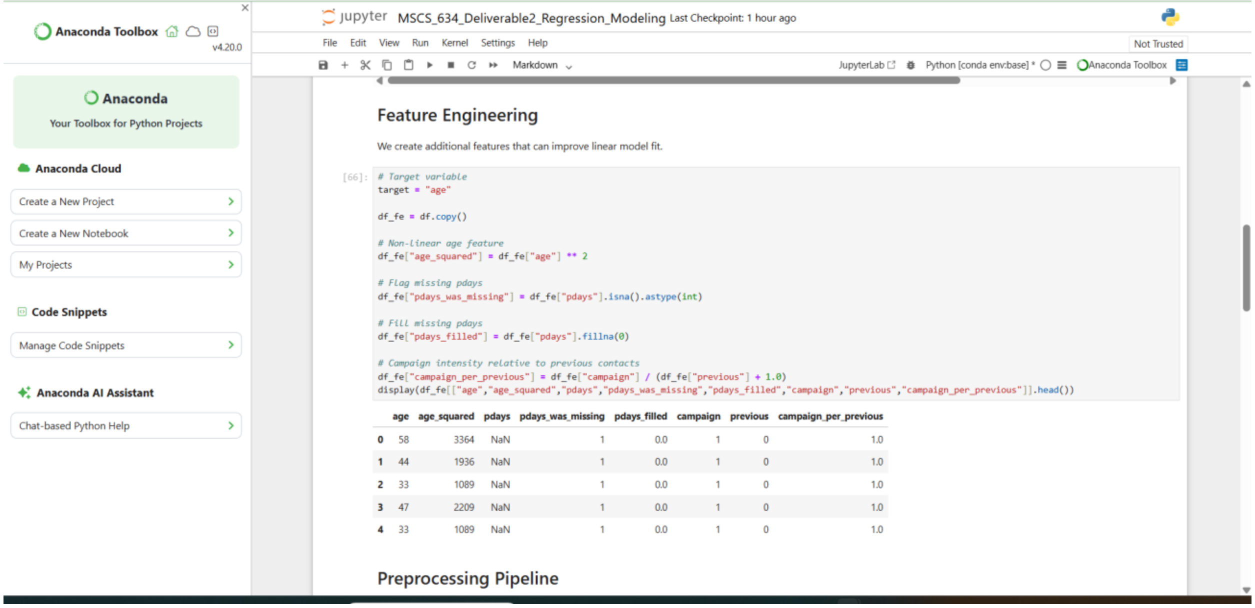


Figure 4. Outlier and numeric feature distribution analysis

Feature Engineering

We did feature engineering to help improve the model performance. For example, we created age_squared so the model can capture a non-linear relationship between age and the target variable. We also made a few features related to previous campaign activity to better represent customer engagement and past contact history.

We also encoded categorical variables so they can work in machine learning models. These steps helped the models learn patterns from the data more clearly instead of ignoring important information.



Feature Engineering

We create additional features that can improve linear model fit.

```
[66]: # Target variable
target = "age"

df_fe = df.copy()

# Non-linear age feature
df_fe["age_squared"] = df_fe["age"] ** 2

# Flag missing pdays
df_fe["pdays_was_missing"] = df_fe["pdays"].isna().astype(int)

# Fill missing pdays
df_fe["pdays_filled"] = df_fe["pdays"].fillna(0)

# Campaign intensity relative to previous contacts
df_fe["campaign_per_previous"] = df_fe["campaign"] / (df_fe["previous"] + 1.0)
display(df_fe[["age","age_squared","pdays","pdays_was_missing","pdays_filled","campaign","previous","campaign_per_previous"]].head())
```

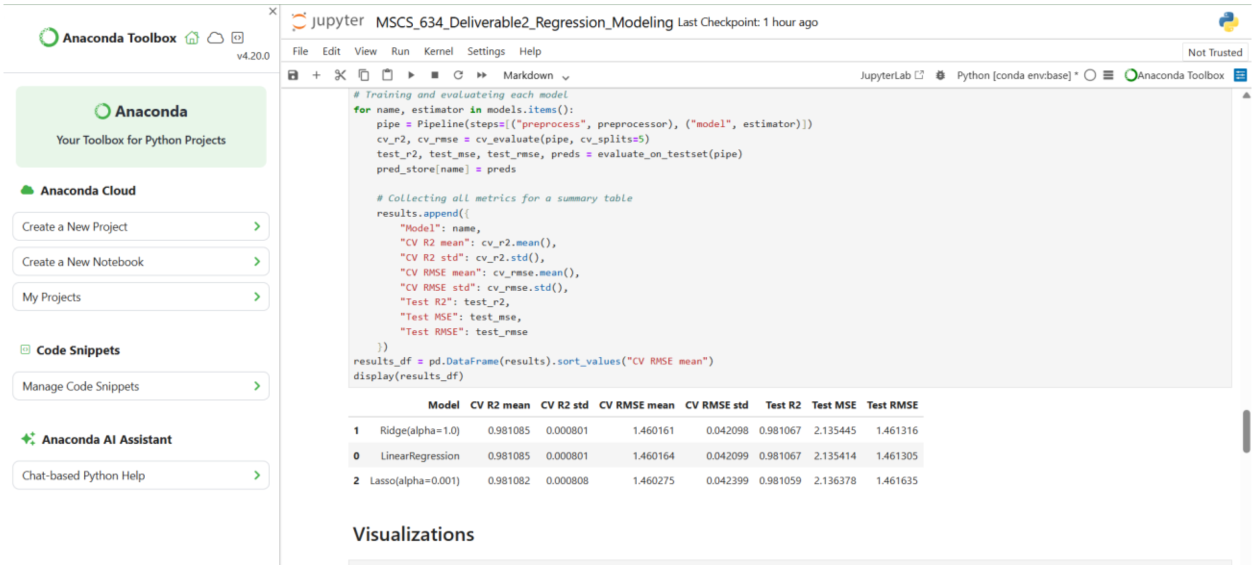
	age	age_squared	pdays	pdays_was_missing	pdays_filled	campaign	previous	campaign_per_previous
0	58	3364	NaN	1	0.0	1	0	1.0
1	44	1936	NaN	1	0.0	1	0	1.0
2	33	1089	NaN	1	0.0	1	0	1.0
3	47	2209	NaN	1	0.0	1	0	1.0
4	33	1089	NaN	1	0.0	1	0	1.0

Preprocessing Pipeline

Figure 5. Feature engineering

Regression Modeling and Evaluation

In Deliverable 2, our group created and compared multiple regression models to predict the subscription outcome. We started with baseline regression models and then improved them by adding engineered features. To measure how well each model performed, we used evaluation metrics such as R-squared, Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). We also used 5-fold cross-validation to test how consistent the models were across different splits of the data. This was helpful because it gave us a stronger idea of how the models might perform on new data and it reduced the chance of overfitting. The results table helped us see the difference between models in a simple way. Instead of picking a model from one split, cross-validation gave us a more consistent view of performance across multiple folds.



```
# Training and evaluateing each model
for name, estimator in models.items():
    pipe = Pipeline(steps=[("preprocess", preprocessor), ("model", estimator)])
    cv_r2, cv_rmse = cv_evaluate(pipe, cv_splits=5)
    test_r2, test_mse, test_rmse, preds = evaluate_on_testset(pipe)
    pred_store[name] = preds

    # Collecting all metrics for a summary table
    results.append({
        "Model": name,
        "CV R2 mean": cv_r2.mean(),
        "CV R2 std": cv_r2.std(),
        "CV RMSE mean": cv_rmse.mean(),
        "CV RMSE std": cv_rmse.std(),
        "Test R2": test_r2,
        "Test MSE": test_mse,
        "Test RMSE": test_rmse
    })
results_df = pd.DataFrame(results).sort_values("CV RMSE mean")
display(results_df)
```

	Model	CV R2 mean	CV R2 std	CV RMSE mean	CV RMSE std	Test R2	Test MSE	Test RMSE
1	Ridge(alpha=1.0)	0.981085	0.000801	1.460161	0.042098	0.981067	2.135445	1.461316
0	LinearRegression	0.981085	0.000801	1.460164	0.042099	0.981067	2.135414	1.461305
2	Lasso(alpha=0.001)	0.981082	0.000808	1.460275	0.042399	0.981059	2.136378	1.461635

Visualizations

Figure 6. cross-validation/test results table.

Classification, Clustering and Pattern Mining

For Deliverable 3, our group worked on three main parts: classification, clustering, and pattern mining. First, we built multiple classification models such as Logistic Regression, Decision Trees, and Support Vector Machines (SVM) to predict whether a customer would subscribe. To compare the models, we used confusion matrices, ROC curves, accuracy, and F1 score. We also did hyperparameter tuning (mainly to improve the SVM model), which helped us get better results compared to the baseline settings. Next, we used K-Means clustering to group customers with similar characteristics. To choose the best number of clusters, we used the elbow method and silhouette scores, since they helped us see which cluster size made the most sense.

Finally, we applied association rule mining using the Apriori algorithm to find frequent patterns in the data. These rules helped us understand which combinations of customer features and campaign details were commonly linked with subscription outcomes, which can be useful for planning better marketing strategies. The confusion matrix and ROC curve made it easier to see what kind of errors the model makes. This helped us understand whether the model was missing too many "yes" cases or incorrectly predicting "yes" when it should be "no."

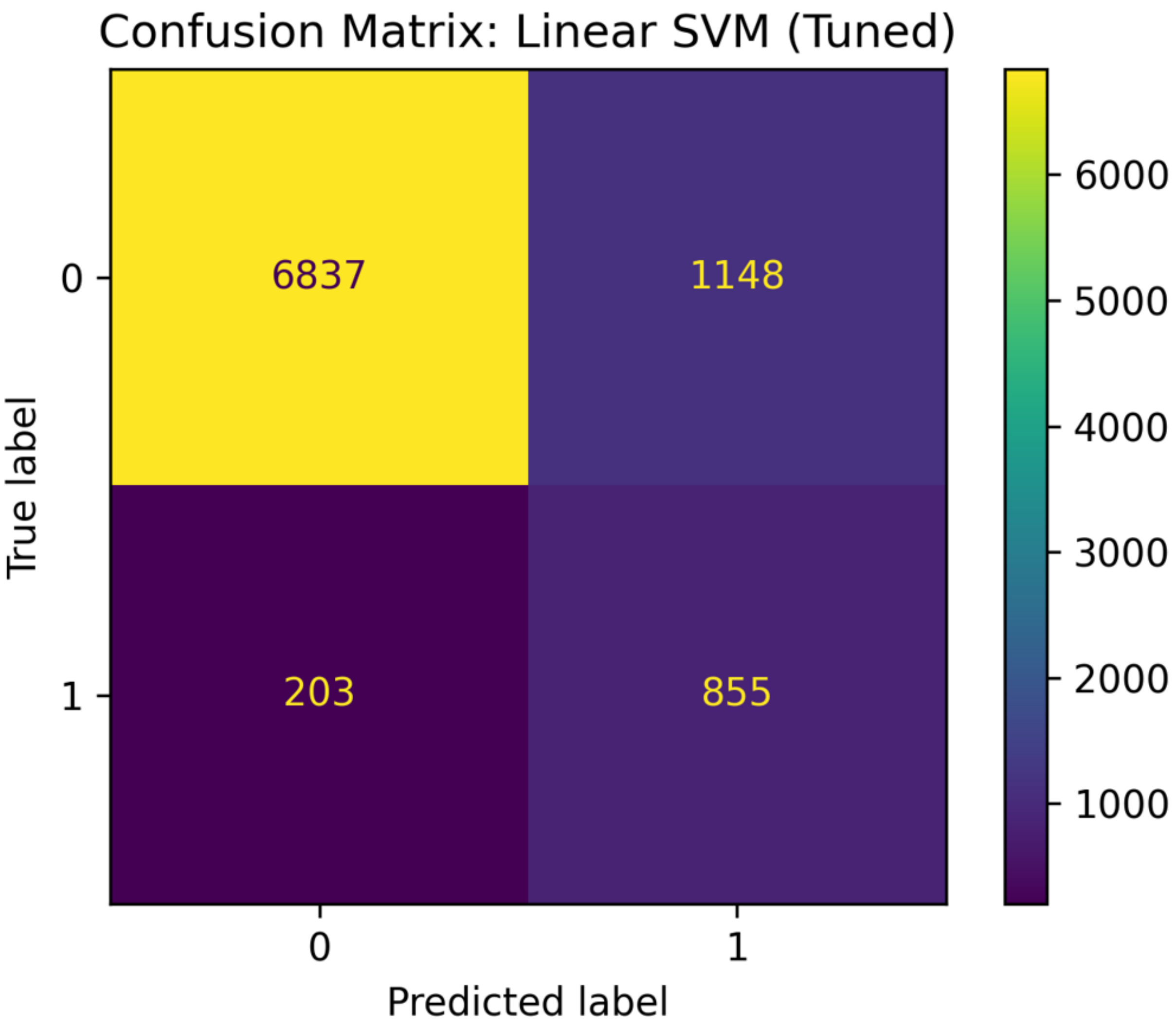


Figure 7. Confusion matrix for tuned Linear SVM classifier

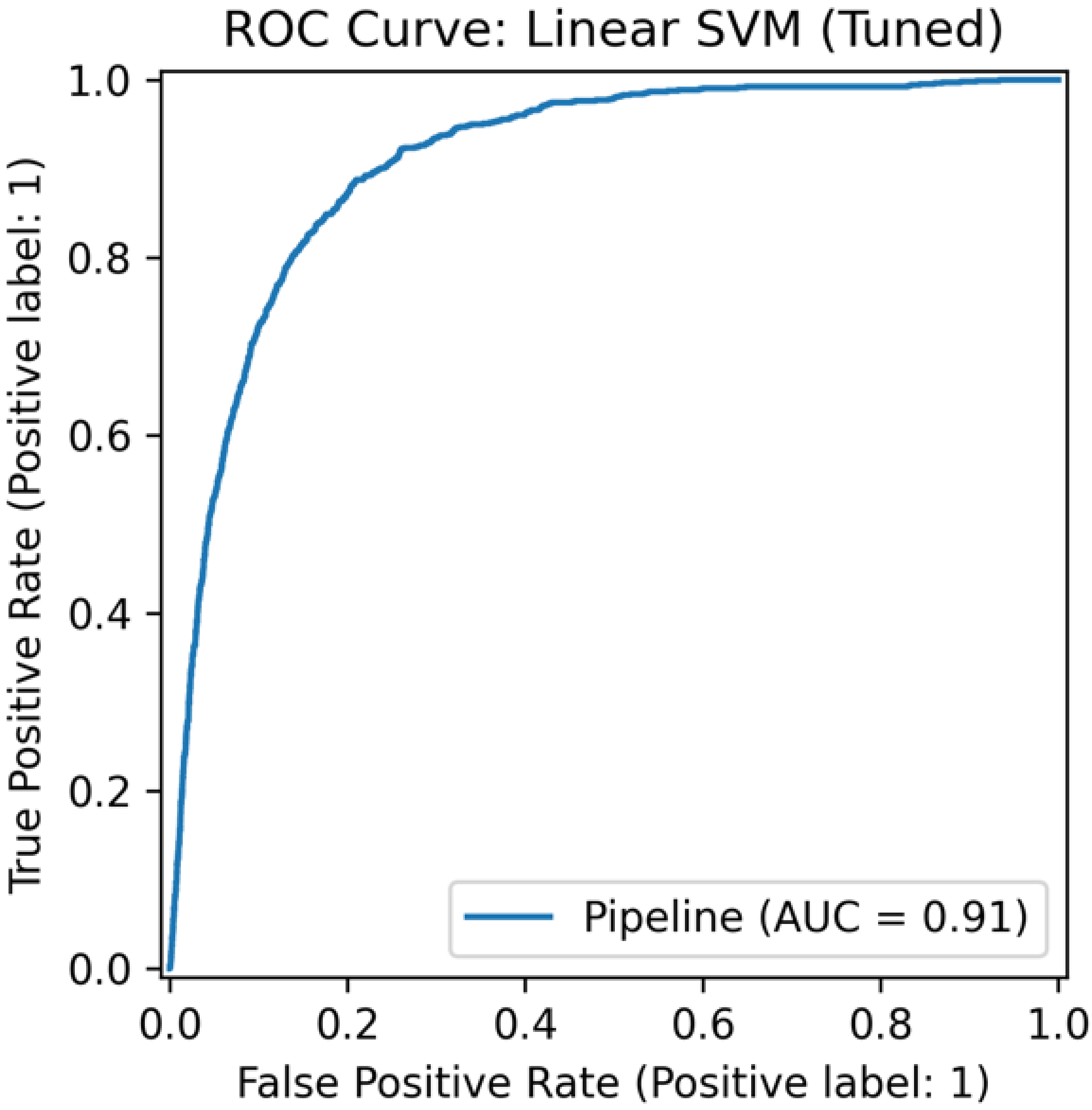


Figure 8. ROC curve for tuned Linear SVM classifier

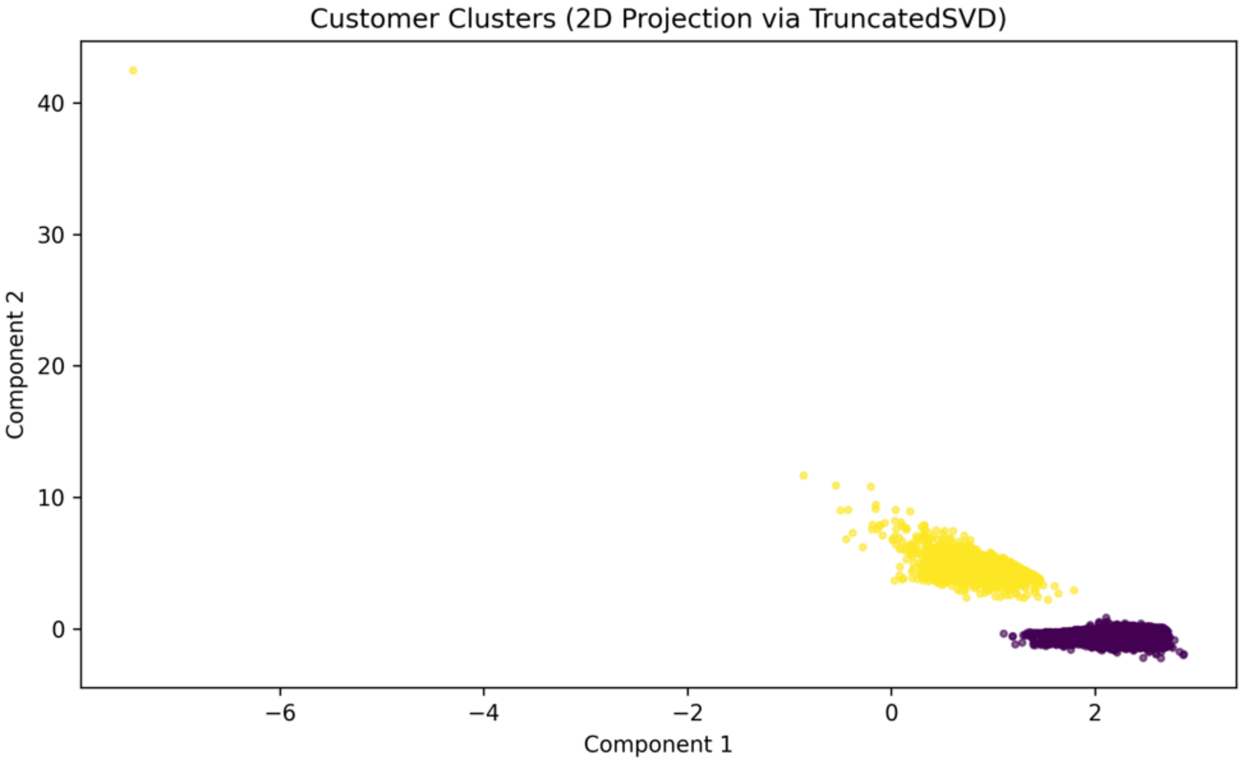
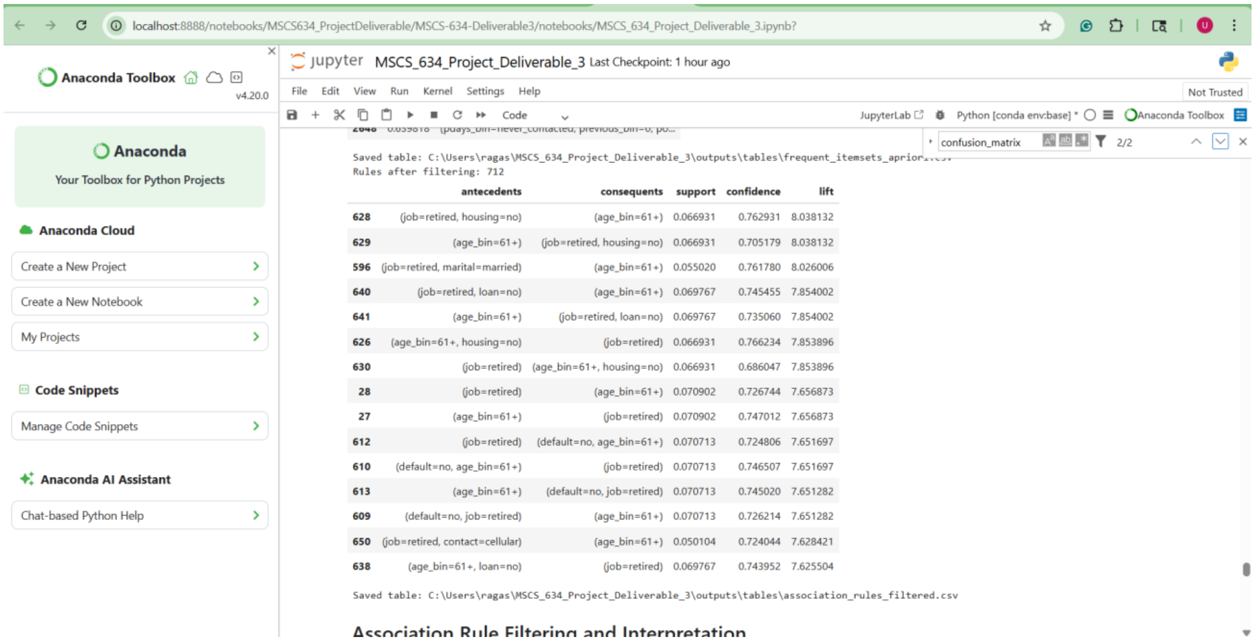


Figure 9. Customer clusters from K-Means



```
2048    0.039818    (pdays_bin=never_contacted, previous_bin=0, po...

Saved table: C:\Users\ragas\MSCS_634_Project_Deliverable_3\outputs\tables\frequent_itemsets_aprior
Rules after filtering: 712
```

	antecedents	consequents	support	confidence	lift
628	(job=retired, housing=no)	(age_bin=61+)	0.066931	0.762931	8.038132
629	(age_bin=61+)	(job=retired, housing=no)	0.066931	0.705179	8.038132
596	(job=retired, marital=married)	(age_bin=61+)	0.055020	0.761780	8.026006
640	(job=retired, loan=no)	(age_bin=61+)	0.069767	0.745455	7.854002
641	(age_bin=61+)	(job=retired, loan=no)	0.069767	0.735060	7.854002
626	(age_bin=61+, housing=no)	(job=retired)	0.066931	0.766234	7.853896
630	(job=retired)	(age_bin=61+, housing=no)	0.066931	0.686047	7.853896
28	(job=retired)	(age_bin=61+)	0.070902	0.726744	7.656873
27	(age_bin=61+)	(job=retired)	0.070902	0.747012	7.656873
612	(job=retired)	(default=no, age_bin=61+)	0.070713	0.724806	7.651697
610	(default=no, age_bin=61+)	(job=retired)	0.070713	0.746507	7.651697
613	(age_bin=61+)	(default=no, job=retired)	0.070713	0.745020	7.651282
609	(default=no, job=retired)	(age_bin=61+)	0.070713	0.726214	7.651282
650	(job=retired, contact=cellular)	(age_bin=61+)	0.050104	0.724044	7.628421
638	(age_bin=61+, loan=no)	(job=retired)	0.069767	0.743952	7.625504

```
Saved table: C:\Users\ragas\MSCS_634_Project_Deliverable_3\outputs\tables\association_rules_filtered.csv
```

Association Rule Filtering and Interpretation

Figure 10. Top association rules from Apriori

Key Insights We Found

- Campaign features (especially duration and previous contact history) were some of the strongest signals for subscription.
- Classification models worked better than regression for this dataset, and results improved after hyperparameter tuning.
- K-Means clustering showed different customer groups, and some clusters had higher subscription rates than others.
- Association rules showed patterns where combinations of job type, education, and contact method appeared more often in successful subscriptions.

Overall, we noticed that campaign-related features like duration and previous contact history mattered a lot when predicting subscriptions. When we grouped customers using clustering, some segments were more likely to subscribe than others, which suggests that a targeted marketing strategy could work better than using the same approach for everyone. We also saw that classification models performed better than regression for this problem, especially after tuning the model settings. Finally, the association rules supported our findings by showing repeated combinations of customer attributes and contact method that showed up more often when subscriptions were successful.

Ethical Considerations

We kept ethical considerations in mind throughout this project. The dataset does not include personal identifying information like names, phone numbers, or addresses, which helps protect customer privacy. Even so, there can still be bias in the data because some features like age, job, and education might affect the outcome in a way that is not fully fair.

To focus on fairness, we paid attention to how the models behaved across different groups and we did not rely on features in a way that could lead to unfair or discriminatory decisions. Overall, our results should be used as a tool to support decisions, not as something that completely replaces human judgment.

Future Work

Based on our results, banks can make their marketing campaigns more effective by prioritizing customer groups that showed higher engagement and stronger subscription outcomes. Using past contact history and campaign response patterns to guide who to contact, when to contact, and how to contact them could improve the overall success rate. For future work, we could test more advanced approaches such as ensemble models like Random Forest or Gradient Boosting. It would also be useful to add time-based patterns such as seasonality, month trends and external factors like economic conditions to see if they improve prediction performance.

Conclusion

This project helped us go through the full data mining process from start to finish. We cleaned the data, handled missing values, and made sure everything was ready for modeling, then we did EDA to understand the distributions, relationships between features, and how different variables connect to the target outcome. After that, we built models and evaluated how well they worked, including both regression and classification so we could compare which approach fit this problem better. We also used K-Means clustering to group customers with similar characteristics, which helped us understand different customer segments, and we applied association rule mining to find common patterns that were often linked with subscription results.

References

Dua, D., & Graff, C. (2019). UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences. https://archive.ics.uci.edu/ml/datasets/bank+marketing

Han, J., Kamber, M., & Pei, J. (2012). Data mining: Concepts and techniques (3rd ed.). Morgan Kaufmann.

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90–95.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). An introduction to statistical learning: With applications in Python (2nd ed.). Springer.

Tan, P.-N., Steinbach, M., & Kumar, V. (2019). Introduction to data mining (2nd ed.). Pearson.