Bank Customer

Churn Prediction and Lifetime Value Optimization

Business Intelligence

Using **11 machine learning** models
(e.g., Logistic Regression, Random Forest, XGBoost, CatBoost).
Includes EDA, churn prediction, CLV analysis, and model evaluation
with metrics like **Precision, Recall, F1-score**, and Confusion Matrix.

Hijir Della Wirasti

https://github.com/hijirdella/Bank-Customer-Churn-Prediction-and-CLV-Optimization

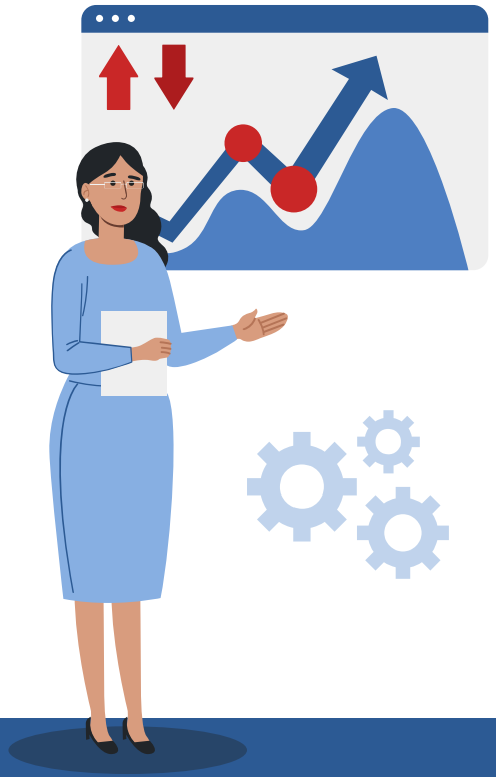
LINKS

Dataset	https://www.kaggle.com/datasets/gauravtopre/bank-customer-churn-dataset/data
Google Collab	https://colab.research.google.com/drive/1xTTv02XhxYjcd8XIryYi8MtWH8kwq1ZW?usp=sharing
GitHub	https://github.com/hijirdella/Bank-Customer-Churn-Prediction-and-CLV-Optimization
LinkedIn	https://www.linkedin.com/in/hijirdella/
Email	hijirdw@gmail.com

Hijir Della Wirasti

Business Intelligence

TABLE OF CONTENTS



01

Problem Definition

02

Key Insights from Data Exploration

03

Model Development and Evaluation

04

Business Recommendations

05

Conclusion

01

Problem Definition

Problem Definition

Banks face significant challenges in retaining customers due to increased competition and diverse customer needs. Customer churn, the process of customers leaving a service, leads to lost revenue and increased acquisition costs. Additionally, not all customers contribute equally to the bank's profitability, making it crucial to identify high-value customers and allocate retention efforts effectively.

Key Questions:

1. **Churn Prediction**: Which customers are most likely to churn, and what factors drive their decision to leave?
2. **Customer Lifetime Value (CLV)**: How can the bank estimate the long-term value of a customer and prioritize retention strategies for high-value customers?

Objectives:

1. Develop a **churn prediction model** using machine learning to identify at-risk customers.
2. Perform **Customer Lifetime Value (CLV) analysis** to evaluate customer profitability and retention impact.
3. Provide actionable insights for proactive customer retention and maximize lifetime value.

02

Key Insights from Data Exploration

DATA EXPLORATION

Data

	customer_id	credit_score	country	gender	age	tenure	balance	products_number	credit_card	active_member	estimated_salary	churn
0	15634602	619	France	Female	42	2	0.00	1	1	1	101348.88	1
1	15647311	608	Spain	Female	41	1	83807.86	1	0	1	112542.58	0
2	15619304	502	France	Female	42	8	159660.80	3	1	0	113931.57	1
3	15701354	699	France	Female	39	1	0.00	2	0	0	93826.63	0
4	15737888	850	Spain	Female	43	2	125510.82	1	1	1	79084.10	0
5	15574012	645	Spain	Male	44	8	113755.78	2	1	0	149756.71	1
6	15592531	822	France	Male	50	7	0.00	2	1	1	10062.80	0
7	15656148	376	Germany	Female	29	4	115046.74	4	1	0	119346.88	1
8	15792365	501	France	Male	44	4	142051.07	2	0	1	74940.50	0
9	15592389	684	France	Male	27	2	134603.88	1	1	1	71725.73	0

Numerical

	customer_id	credit_score	age	tenure	balance	products_number	credit_card	active_member	estimated_salary	churn
count	1.000000e+04	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.00000	10000.000000	10000.000000	10000.000000
mean	1.569094e+07	650.528800	38.921800	5.012800	76485.889288	1.530200	0.70550	0.515100	100090.239881	0.203700
std	7.193619e+04	96.653299	10.487806	2.892174	62397.405202	0.581654	0.45584	0.499797	57510.492818	0.402769
min	1.556570e+07	350.000000	18.000000	0.000000	0.000000	1.000000	0.00000	0.000000	11.580000	0.000000
25%	1.562853e+07	584.000000	32.000000	3.000000	0.000000	1.000000	0.00000	0.000000	51002.110000	0.000000
50%	1.569074e+07	652.000000	37.000000	5.000000	97198.540000	1.000000	1.00000	1.000000	100193.915000	0.000000
75%	1.575323e+07	718.000000	44.000000	7.000000	127644.240000	2.000000	1.00000	1.000000	149388.247500	0.000000
max	1.581569e+07	850.000000	92.000000	10.000000	250898.090000	4.000000	1.00000	1.000000	199992.480000	1.000000

Data Info

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   customer_id       10000 non-null  int64
 1   credit_score      10000 non-null  int64
 2   country           10000 non-null  object
 3   gender            10000 non-null  object
 4   age               10000 non-null  int64
 5   tenure            10000 non-null  int64
 6   balance           10000 non-null  float64
 7   products_number   10000 non-null  int64
 8   credit_card       10000 non-null  int64
 9   active_member     10000 non-null  int64
 10  estimated_salary  10000 non-null  float64
 11  churn             10000 non-null  int64
dtypes: float64(2), int64(8), object(2)
memory usage: 937.6+ KB
```

Missing & Duplicate

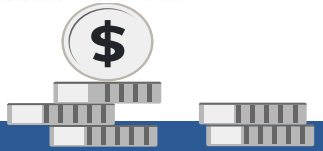
```
Missing Values:
customer_id        0
credit_score       0
country            0
gender             0
age                0
tenure             0
balance            0
products_number    0
credit_card        0
active_member      0
estimated_salary   0
churn              0
dtype: int64

Duplicate Values:
0
```

Categorical

	country	gender
count	10000	10000
unique	3	2
top	France	Male
freq	5014	5457

Churn Percentage

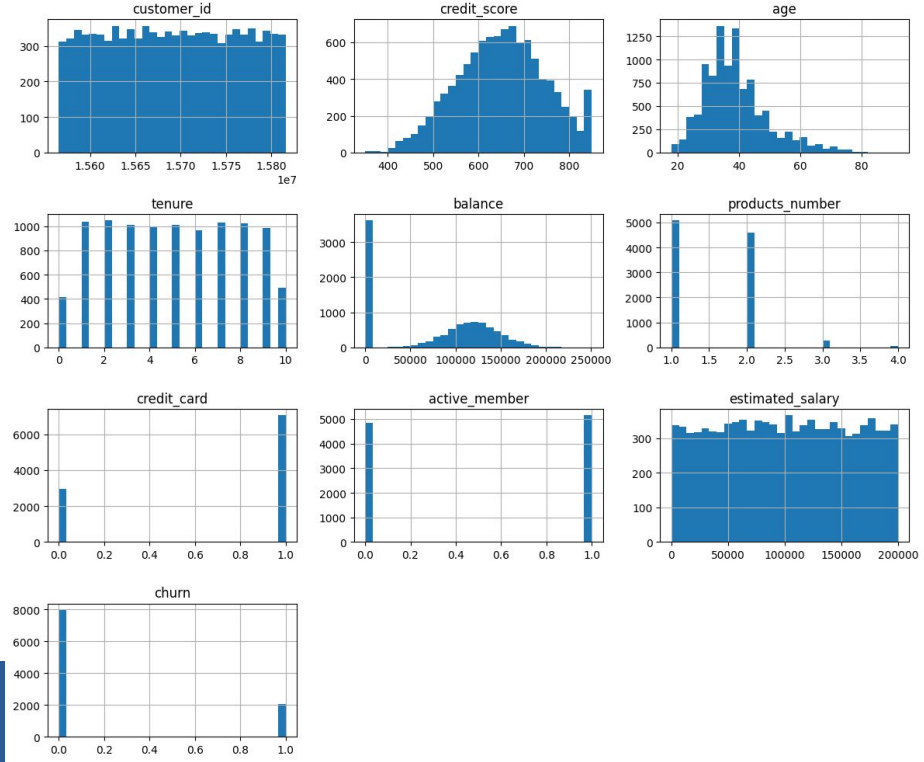
```
Churn Percentage:
churn
0    79.63
1    20.37
Name: proportion, dtype: float64
```



The dataset contains 10,000 records with **no missing** or **duplicate values**. Churn is imbalanced, with 20.37% of customers churned. Numerical features like balance and age show significant variation, while France dominates as the most common country (50.14%). The class imbalance suggests a need for handling to improve prediction accuracy.

EDA (Exploratory Data Analysis)

Univariate Analysis for Numerical Data



Insights from Histograms:

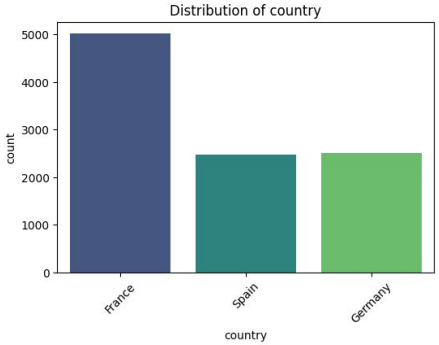
1. `credit_score`:
   - The distribution is approximately normal, with most customers having a credit score between 500 and 750.
   - A small number of customers have very high or very low credit scores.
2. `age`:
   - Most customers are in their 30s and 40s.
   - A few outliers represent customers older than 70.
3. `tenure`:
   - Tenure is evenly distributed, indicating no specific bias toward customers with longer or shorter relationships.
4. `balance`:
   - Many customers have a balance close to zero, but a significant group has balances spread between 50,000 and 200,000.
5. `products_number`:
   - Most customers have 1 or 2 products, with very few having 3 or 4 products.
6. `credit_card`:
   - Most customers hold a credit card.
7. `active_member`:
   - Slightly more than half of the customers are active members.
8. `estimated_salary`:
   - Salaries are evenly distributed across the range, with no visible outliers.
9. `churn`:
   - There is a class imbalance, with the majority of customers labeled as non-churners.

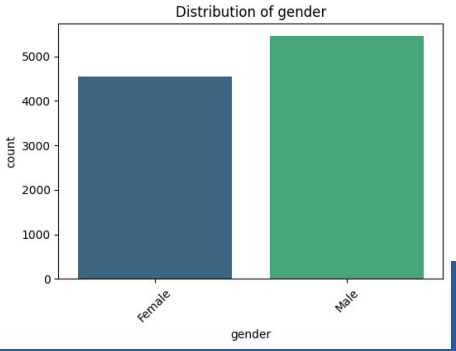
Key Takeaways:

- Features like `credit_score`, `balance`, and `age` show significant variation and are likely strong predictors of churn.
- The class imbalance in `churn` may need to be addressed to improve model performance.
- Features like `products_number` and `credit_card` exhibit limited variation, which might reduce their predictive power.

EDA (Exploratory Data Analysis)

Univariate Analysis for Categorical Data





Insights from Categorical Feature Distribution

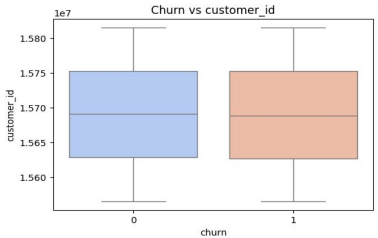
1. **Country Distribution:**
   - Majority of customers are from **France** (~50%).
   - Customers from **Spain** and **Germany** are almost equal, contributing to the other 50%.
2. **Gender Distribution:**
   - Slightly more **Male** customers than Female (~55% Male, ~45% Female).

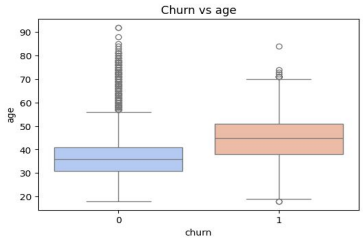
Key Takeaways:

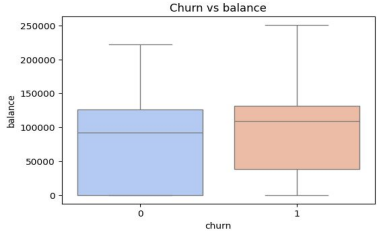
- The dominance of French customers might skew model predictions unless properly balanced.
- Gender is relatively balanced, so it may not have a strong predictive influence.

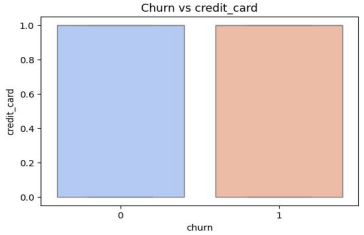
EDA (Exploratory Data Analysis)

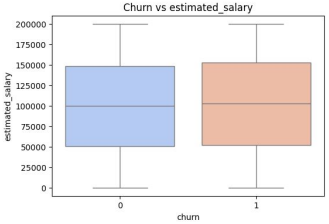
Bivariate Analysis for Numerical Data

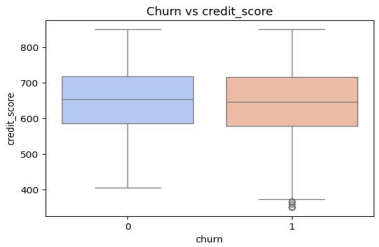


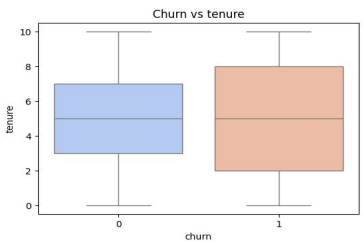


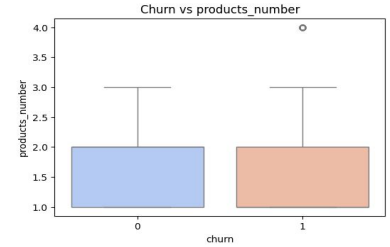


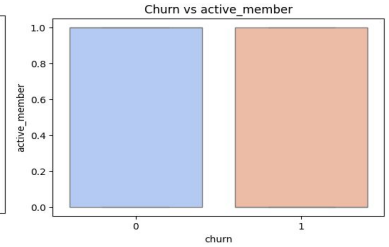










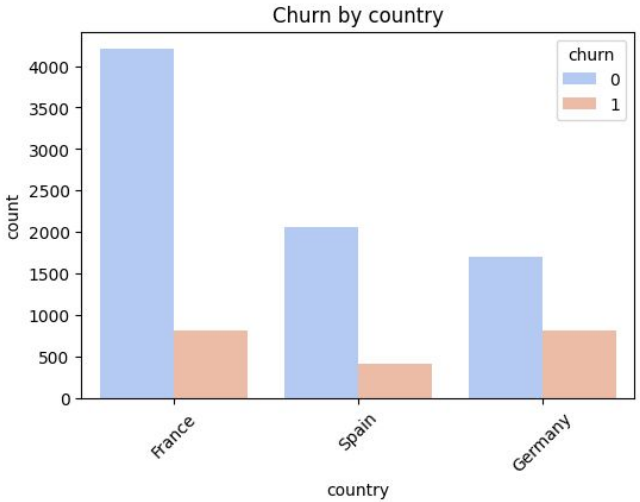


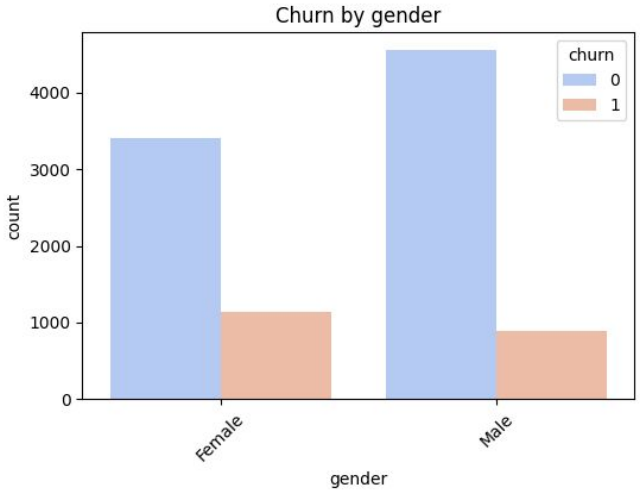
Key Takeaways:

- **Credit Score, Age, and Balance** show significant patterns with churn and are likely strong predictors.
- Features like **tenure, products number, credit card, and salary** may have limited predictive power. These features might require deeper analysis or could be deprioritized.

EDA (Exploratory Data Analysis)

Bivariate Analysis for Categorical Data





Churn by Country:

- France has the highest churn count but also the largest customer base.
- Churn in Spain and Germany is proportionally smaller.

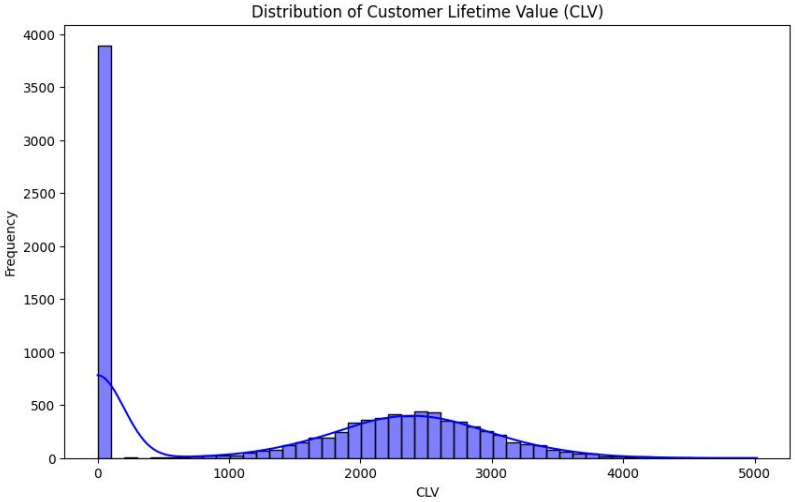
Churn by Gender:

- Females have a slightly higher churn rate compared to males.
- Males represent the majority of the customer base.

Key Insight:

- Targeted retention strategies may be needed for France and female customers.

Customer Lifetime Value



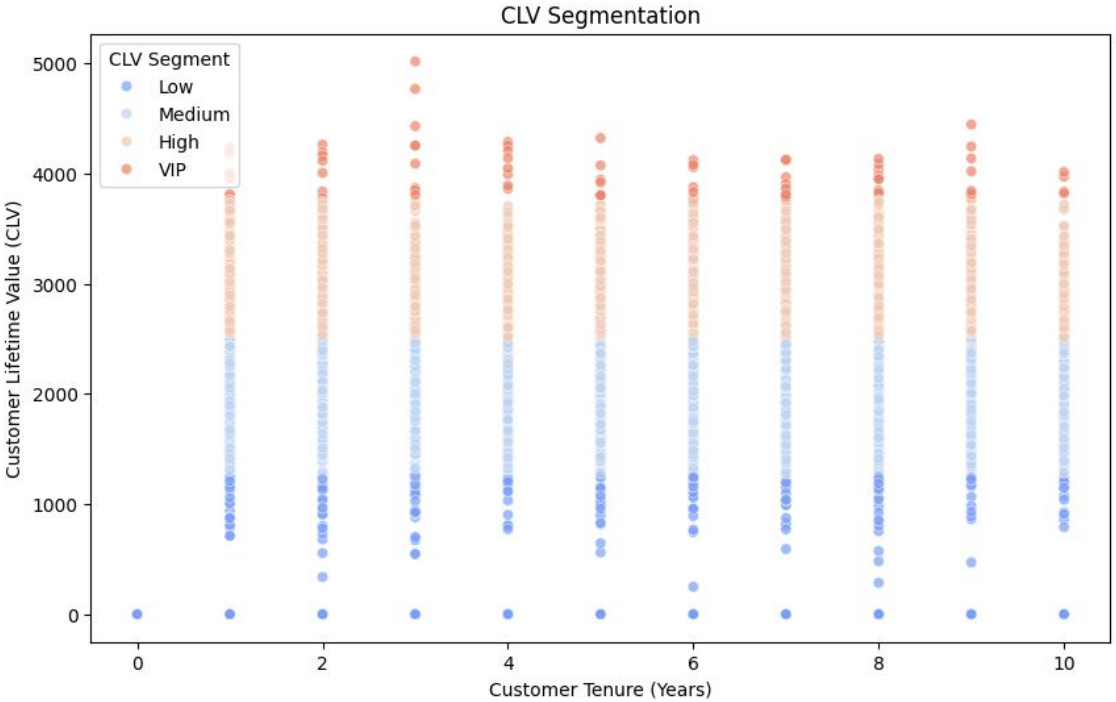
CLV Segment Analysis (EDA):

	CLV Segment	CLV Mean	CLV Sum	Revenue Mean	Tenure Mean
0	Low	46.688768	1.904902e+05	208.404405	4.730392
1	Medium	2049.266225	6.856845e+06	2049.266225	5.205021
2	High	2915.419408	7.273971e+06	2915.419408	5.202806
3	VIP	4000.907256	3.160717e+05	4000.907256	5.455696

CLV Insights and Segment Analysis:

1. **Distribution of CLV**:
   - The majority of customers fall into the **Low** CLV category.
   - Higher CLV values are observed in a smaller subset, including **Medium**, **High**, and **VIP** segments.
2. **Segment Analysis**:
   - **Low CLV** customers have a mean CLV of 46.69, with the shortest tenure and lowest revenue contributions.
   - **Medium CLV** customers contribute significantly with an average CLV of 2049.27.
   - **High CLV** and **VIP** segments have the highest CLV and revenue, highlighting their importance for retention.
3. **Key Takeaways**:
   - Focus retention efforts on **Medium**, **High**, and **VIP** customers to maximize profitability.
   - Develop strategies to increase tenure and revenue for **Low CLV** customers to boost their lifetime value.

Customer Lifetime Value

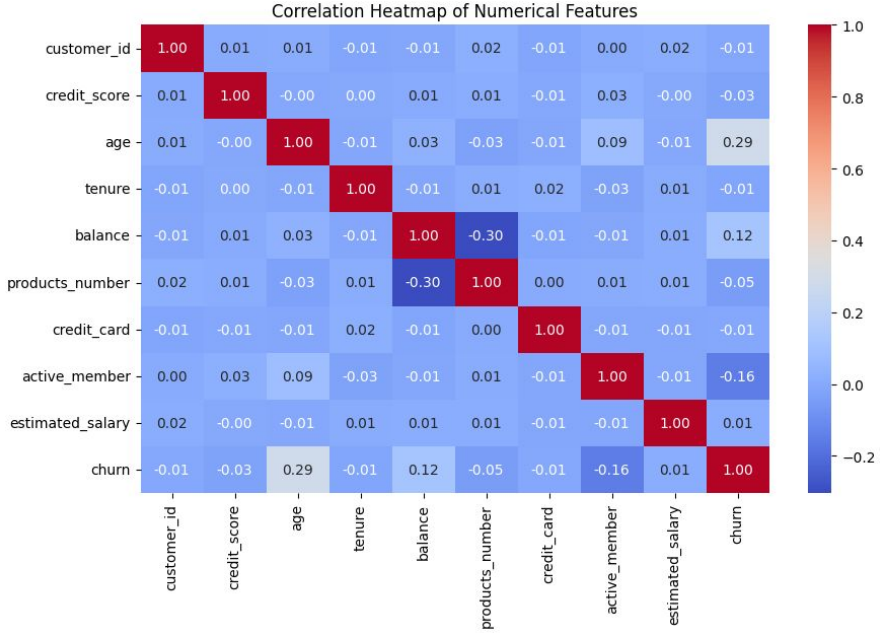


Insights from CLV Segmentation vs Tenure Plot

1. **CLV Distribution Across Tenure**:
   - CLV increases with customer tenure, with VIP customers showing the highest CLV consistently across all tenure periods.
2. **Segment Patterns**:
   - **Low CLV** customers are concentrated at lower tenure levels.
   - **Medium**, **High**, and **VIP** customers are distributed more evenly but become more prominent as tenure increases.
3. **Key Takeaway**:
   - Retention strategies should focus on converting **Low** and **Medium** CLV customers to higher segments by increasing their tenure through personalized offers and better customer engagement.

EDA (Exploratory Data Analysis)

Multivariate Analysis



Insights from Correlation Heatmap

1. **Strongest Correlations**:
   - `age` has the highest positive correlation with churn (0.29), suggesting older customers are more likely to churn.
   - `balance` shows a smaller positive correlation with churn (0.12).
2. **Negative Correlations**:
   - `active_member` has a negative correlation with churn (-0.16), indicating active members are less likely to churn.
   - `products_number` and `credit_card` have weak negative correlations with churn.
3. **Feature Independence**:
   - Most numerical features have low intercorrelations, reducing concerns about multicollinearity.
4. **Key Takeaway**:
   - Focus on `age`, `balance`, and `active_member` as important predictors for churn, while weaker features like `products_number` and `credit_card` may contribute less to the model.

03

Model Development and Evaluation

Pre-Processing

1 Handle Outliers

```
Data shape before handling outliers: (10000, 12)
```

```
Data shape after handling outliers: (10000, 12)
```

The shape of the dataset before and after handling outliers is the same, indicating that **no rows were removed**, and the outliers were capped (adjusted) to fall within the calculated bounds using the IQR method.

Why is the Shape Unchanged?

1. **Capping, Not Removing**:
   - The outliers were capped to the `lower_bound` or `upper_bound`, so no rows were dropped, only their values were adjusted.
2. **Dataset Integrity Maintained**:
   - The process ensured that the dataset retains the same number of observations while making the feature values more robust to extreme data points.

2 Correlation Check (2) - Chi Square

```
Uji Chi-Square untuk country
Chi2 Statistik: 301.25533682434536, p-value: 3.8303176053541544e-66

Uji Chi-Square untuk gender
Chi2 Statistik: 112.91857062096116, p-value: 2.2482100097131755e-26

Uji Chi-Square untuk products_number
Chi2 Statistik: 1503.6293615070408, p-value: 0.0

Uji Chi-Square untuk credit_card
Chi2 Statistik: 0.47133779904440803, p-value: 0.49237236141554686

Uji Chi-Square untuk active_member
Chi2 Statistik: 242.98534164287963, p-value: 8.785858269303703e-55

Uji Chi-Square untuk churn
Chi2 Statistik: 9993.835961897581, p-value: 0.0

Features to drop:
       Feature      Chi2   P-value    Chi2 %
3  credit_card  0.471338  0.492372  0.003878
```

Actionable Recommendations:

1. **Keep Important Features**
   - Retain high Chi2 value features like `country`, `gender`, `products_number`, and `active_member` for modeling.
2. **Remove `credit_card` Feature**
   - This feature does not significantly impact churn prediction and can be excluded to simplify the model.
3. **Further Analysis**
   - Conduct additional analysis on the key features to explore deeper relationships with `churn`.
   - Use these relevant features in machine learning models to enhance prediction accuracy.

Pre-Processing

3 Drop Feature

✅ customer_id is dropped because it is just an identifier, not a predictive feature.

✅ credit_card is dropped because statistical tests show it is not related to churn.

✅ Feature selection improves model efficiency and accuracy by keeping only relevant variables.

4 Feature Encoding

	credit_score	country	gender	age	tenure	balance	products_number	active_member	estimated_salary	churn
0	619.0	France	Female	42.0	2.0	0.00	1.0	1.0	101348.88	1
1	608.0	Spain	Female	41.0	1.0	83807.86	1.0	1.0	112542.58	0
2	502.0	France	Female	42.0	8.0	159660.80	3.0	0.0	113931.57	1
3	699.0	France	Female	39.0	1.0	0.00	2.0	0.0	93826.63	0
4	850.0	Spain	Female	43.0	2.0	125510.82	1.0	1.0	79084.10	0

	credit_score	gender	age	tenure	balance	products_number	\
0	619.0	0	42.0	2.0	0.00	1.0	
1	608.0	0	41.0	1.0	83807.86	1.0	
2	502.0	0	42.0	8.0	159660.80	3.0	
3	699.0	0	39.0	1.0	0.00	2.0	
4	850.0	0	43.0	2.0	125510.82	1.0	

	active_member	estimated_salary	churn	country_Germany	country_Spain
0	1	101348.88	1	False	False
1	1	112542.58	0	False	True
2	0	113931.57	1	False	False
3	0	93826.63	0	False	False
4	1	79084.10	0	False	True

Actionable Recommendations:

1. **Keep Important Features**
   - Retain high Chi2 value features like `country`, `gender`, `products_number`, and `active_member` for modeling.
2. **Remove `credit_card` Feature**
   - This feature does not significantly impact churn prediction and can be excluded to simplify the model.
3. **Further Analysis**
   - Conduct additional analysis on the key features to explore deeper relationships with `churn`.
   - Use these relevant features in machine learning models to enhance prediction accuracy.

Pre-Processing

5 Feature Scalling

✅ No need for Min-Max Normalization before StandardScaler because the data has no extreme outliers

```
   credit_score    gender       age    tenure   balance  products_number  \
0     -0.326878 -1.095988  0.342615 -1.041760 -1.225848        -0.924827
1     -0.440804 -1.095988  0.240011 -1.387538  0.117350        -0.924827
2     -1.538636 -1.095988  0.342615  1.032908  1.333053         2.583620
3      0.501675 -1.095988  0.034803 -1.387538 -1.225848         0.829397
4      2.065569 -1.095988  0.445219 -1.041760  0.785728        -0.924827

   active_member  estimated_salary  country_Germany  country_Spain
0       0.970243          0.021886        -0.578736      -0.573809
1       0.970243          0.216534        -0.578736       1.742740
2      -1.030670          0.240687        -0.578736      -0.573809
3      -1.030670         -0.108918        -0.578736      -0.573809
4       0.970243         -0.365276        -0.578736       1.742740
```

Explanation

1. **Feature Scaling**:
   - X: Features without the target (churn).
   - y: Target (churn).
2. **StandardScaler**:
   - Standardizes features (mean = 0, std = 1) for consistent scaling.
3. **Why?**:
   - Prevents large-scale features from dominating.
   - Optimizes performance for scaling-sensitive models (e.g., SVM, Logistic Regression).

6 Handle Class Imbalance

Oversampling

```
Distribusi kelas sebelum SMOTE: Counter({0: 7963, 1: 2037})
Distribusi kelas setelah SMOTE: Counter({0: 7963, 1: 3981})
   credit_score    gender       age    tenure   balance  products_number  \
0     -0.326878 -1.095988  0.342615 -1.041760 -1.225848        -0.924827
1     -0.440804 -1.095988  0.240011 -1.387538  0.117350        -0.924827
2     -1.538636 -1.095988  0.342615  1.032908  1.333053         2.583620
3      0.501675 -1.095988  0.034803 -1.387538 -1.225848         0.829397
4      2.065569 -1.095988  0.445219 -1.041760  0.785728        -0.924827

   active_member  estimated_salary  country_Germany  country_Spain
0       0.970243          0.021886        -0.578736      -0.573809
1       0.970243          0.216534        -0.578736       1.742740
2      -1.030670          0.240687        -0.578736      -0.573809
3      -1.030670         -0.108918        -0.578736      -0.573809
4       0.970243         -0.365276        -0.578736       1.742740
```

Undersampling

```
Distribusi kelas sebelum undersampling: Counter({0: 7963, 1: 2037})
Distribusi kelas setelah undersampling: Counter({0: 2546, 1: 2037})
   credit_score    gender       age    tenure   balance  products_number  \
0      0.905595 -1.095988 -0.580820 -0.695982  0.731614         0.829397
1     -0.171524 -1.095988  0.650427 -1.041760 -1.225848        -0.924827
2     -0.647942  0.912419 -0.888632 -0.695982  0.617383        -0.924827
3     -1.352212  0.912419 -0.478216 -0.695982 -1.225848         0.829397
4      1.019521 -1.095988  0.547823 -1.733315 -0.079945         0.829397

   active_member  estimated_salary  country_Germany  country_Spain
0       0.970243         -1.255880        -0.578736       1.742740
1       0.970243          0.754127        -0.578736      -0.573809
2      -1.030670          1.334923        -0.578736      -0.573809
3       0.970243          0.080228        -0.578736      -0.573809
4      -1.030670          0.886727         1.727904      -0.573809
```

Explanation of Oversampling and Undersampling:

1. **Oversampling (SMOTE)**:
   - **Purpose**: Balances the dataset by increasing the minority class samples.
   - **How**: SMOTE generates synthetic samples for the minority class using interpolation.
   - **Result**: The minority class is increased to 50% of the majority class, improving model performance on the minority class.
2. **Undersampling (RandomUnderSampler)**:
   - **Purpose**: Balances the dataset by reducing the majority class samples.
   - **How**: RandomUnderSampler randomly removes samples from the majority class.
   - **Result**: The majority class is reduced to 80% of the minority class, ensuring balance without oversampling.

Key Differences:

- **SMOTE** increases the dataset size by adding synthetic samples, while **RandomUnderSampler** reduces the dataset size by removing majority class samples.
- Oversampling is beneficial for retaining all data points, while undersampling avoids synthetic data but loses some original majority class data.

Model Development

1 Split Data

1. **Purpose**: The `train_test_split` function splits the dataset into training (80%) and testing (20%) sets to evaluate model performance on unseen data.
2. **Scenarios**:
   - **Original Dataset**: `X_scaled, y` split to retain original imbalance.
   - **SMOTE Dataset**: `X_smote, y_smote` split to use the oversampled balanced dataset.
   - **Undersampled Dataset**: `X_under, y_under` split to use the reduced balanced dataset.
3. **Stratification**:
   - Ensures the class distribution in training and testing sets is proportional to the original dataset.
4. **Random State**:
   - Ensures reproducibility of splits by setting `random_state=42`.

Each split prepares data for evaluating how balancing methods (SMOTE or undersampling) affect model performance.

2 Modeling

Models Used and Their Advantages:

1. **Logistic Regression**:
   - **Type**: Linear model.
   - **Advantages**: Simple, interpretable, and effective for linearly separable data.
2. **Random Forest**:
   - **Type**: Ensemble of decision trees.
   - **Advantages**: Handles non-linear data, reduces overfitting, and provides feature importance.
3. **Gradient Boosting**:
   - **Type**: Boosting algorithm.
   - **Advantages**: High accuracy, handles non-linear data well, and reduces bias.
4. **AdaBoost**:
   - **Type**: Boosting algorithm.
   - **Advantages**: Focuses on misclassified samples and improves weak learners.
5. **Support Vector Machine (SVM)**:
   - **Type**: Kernel-based algorithm.
   - **Advantages**: Effective for high-dimensional spaces and handles non-linear boundaries.

6. K-Nearest Neighbors (KNN):
- **Type**: Instance-based algorithm.
- **Advantages**: Simple, no assumption about data distribution, and effective for small datasets.

7. Naive Bayes:
- **Type**: Probabilistic classifier.
- **Advantages**: Fast, works well with small datasets, and effective for categorical data.

8. Decision Tree:
- **Type**: Tree-based model.
- **Advantages**: Simple, interpretable, and captures non-linear relationships.

9. XGBoost:
- **Type**: Gradient boosting framework.
- **Advantages**: Highly efficient, supports regularization, and handles missing values.

10 LightGBM:
- **Type**: Gradient boosting framework.
- **Advantages**: Faster than XGBoost, efficient on large datasets, and handles categorical features.

11. CatBoost:
- **Type**: Gradient boosting framework.
- **Advantages**: Handles categorical features natively, avoids overfitting, and has fast training.

Model Evaluation of Normal Data

	Dataset	Model	Accuracy	Precision	Recall	F1 Score
0	Normal	Logistic Regression	0.815500	0.626667	0.230958	0.337522
1	Normal	Random Forest	0.865500	0.787500	0.464373	0.584235
2	Normal	Gradient Boosting	0.872500	0.804000	0.493857	0.611872
3	Normal	AdaBoost	0.855000	0.725869	0.461916	0.564565
4	Normal	Support Vector Machine	0.859000	0.841530	0.378378	0.522034
5	Normal	K-Nearest Neighbors	0.839000	0.676349	0.400491	0.503086
6	Normal	Naive Bayes	0.820000	0.595918	0.358722	0.447853
7	Normal	Decision Tree	0.784500	0.473913	0.535627	0.502884
8	Normal	XGBoost	0.848500	0.673333	0.496314	0.571429
9	Normal	LightGBM	0.861000	0.750973	0.474201	0.581325
10	Normal	CatBoost	0.866000	0.779116	0.476658	0.591463

Key Observations:

1. **Best Accuracy:**
   - The highest accuracy is achieved by the Gradient Boosting model (87.25%), closely followed by CatBoost (86.6%) and Random Forest (86.55%).
2. **Best Precision:**
   - The Support Vector Machine (SVM) model has the highest precision (84.15%), indicating that it minimizes false positives better than the other models.
3. **Best Recall:**
   - Gradient Boosting has the highest recall (49.39%), suggesting it is better at identifying true positives (churn cases).
4. **Best F1-Score:**
   - Gradient Boosting also has the highest F1-score (61.19%), which balances both precision and recall effectively, making it a robust choice for imbalanced datasets.
5. **Logistic Regression Performance:**
   - Logistic Regression has moderate accuracy (81.55%) but performs poorly on recall (23.09%), indicating it struggles to identify churn cases.
6. **Weak Models:**
   - Decision Tree and Naive Bayes models have lower F1-scores (50.29% and 44.78%, respectively), suggesting they are less reliable for predicting churn compared to ensemble methods.

Conclusion:

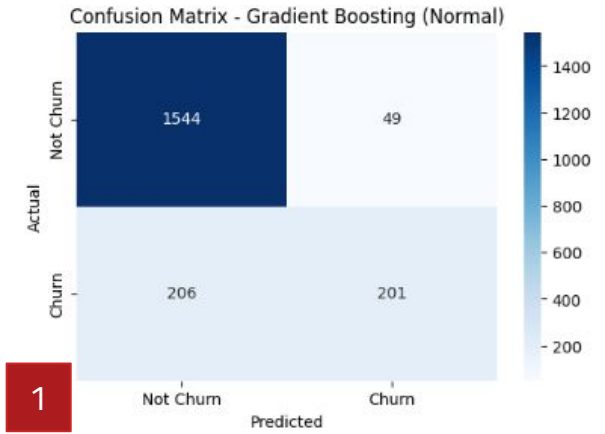
- **Top Performers:** Gradient Boosting, CatBoost, and Random Forest emerge as the best models due to their balanced performance across all metrics.
- **Improvements Needed:** Logistic Regression, Naive Bayes, and Decision Tree could benefit from further tuning or enhanced feature engineering to improve recall and F1-scores.
- **Recommended Model:** Gradient Boosting offers the best trade-off between precision and recall, making it ideal for this churn prediction task.

Check out google collab to see 33 model results: Link

Top Performer of Normal Data

Dataset: Normal | Model: Gradient Boosting

	precision	recall	f1-score	support
0	0.88	0.97	0.92	1593
1	0.80	0.49	0.61	407
accuracy			0.87	2000
macro avg	0.84	0.73	0.77	2000
weighted avg	0.87	0.87	0.86	2000



1

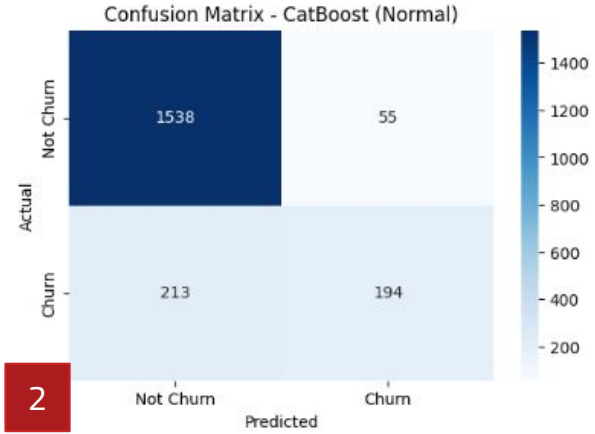
Confusion Matrix Insight for Gradient Boosting:

- Correctly predicted 1544 non-churn cases and 201 churn cases.
- 206 churn cases were misclassified as non-churn, showing some limitations in recall for churn prediction.

Summary: High accuracy and F1-score, making it the most balanced model among the three.

Dataset: Normal | Model: CatBoost

	precision	recall	f1-score	support
0	0.88	0.97	0.92	1593
1	0.78	0.48	0.59	407
accuracy			0.87	2000
macro avg	0.83	0.72	0.76	2000
weighted avg	0.86	0.87	0.85	2000



2

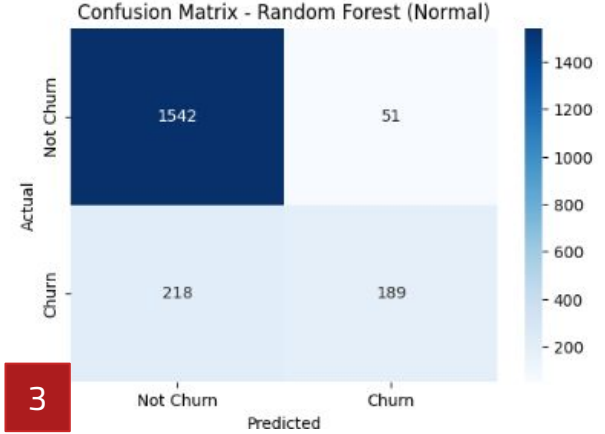
Confusion Matrix Insight for CatBoost:

- Correctly predicted 1538 non-churn cases and 194 churn cases.
- 213 churn cases were misclassified as non-churn, slightly worse recall than Gradient Boosting.

Summary: Very similar to Gradient Boosting in performance, slightly lower F1-score for churn cases.

Dataset: Normal | Model: Random Forest

	precision	recall	f1-score	support
0	0.88	0.97	0.92	1593
1	0.79	0.46	0.58	407
accuracy			0.87	2000
macro avg	0.83	0.72	0.75	2000
weighted avg	0.86	0.87	0.85	2000



3

Confusion Matrix Insight for Random Fores:

- Correctly predicted 1542 non-churn cases and 189 churn cases.
- 218 churn cases were misclassified as non-churn, showing the lowest recall among the three models.

Summary: While overall accuracy is high, the recall for churn cases is slightly worse, which could lead to missed churn predictions.

Model Evaluation of Oversampled Data

	Dataset	Model	Accuracy	Precision	Recall	F1 Score
11	Oversampled	Logistic Regression	0.744244	0.665474	0.467337	0.549077
12	Oversampled	Random Forest	0.868146	0.850073	0.733668	0.787593
13	Oversampled	Gradient Boosting	0.844286	0.829193	0.670854	0.741667
14	Oversampled	AdaBoost	0.807451	0.766667	0.606784	0.677419
15	Oversampled	Support Vector Machine	0.816241	0.802030	0.595477	0.683490
16	Oversampled	K-Nearest Neighbors	0.822101	0.729295	0.741206	0.735202
17	Oversampled	Naive Bayes	0.767267	0.683486	0.561558	0.616552
18	Oversampled	Decision Tree	0.791545	0.682598	0.699749	0.691067
19	Oversampled	XGBoost	0.867308	0.843615	0.738693	0.787676
20	Oversampled	LightGBM	0.873587	0.863235	0.737437	0.795393
21	Oversampled	CatBoost	0.882378	0.882615	0.746231	0.808713

Key Observations:

1. **Best Accuracy:**
   - **CatBoost** achieves the highest accuracy (88.24%), demonstrating its ability to predict most cases correctly.
2. **Best Precision:**
   - **CatBoost** also excels in precision (88.26%), indicating it minimizes false positives effectively.
3. **Best Recall:**
   - **LightGBM** achieves the highest recall (73.74%), making it the best at identifying true churn cases.
4. **Best F1-Score:**
   - **CatBoost** leads with the highest F1-Score (0.808713), showcasing its balanced performance across precision and recall.
5. **Logistic Regression Performance:**
   - Logistic Regression shows modest performance (F1-Score = 0.549077) with poor recall (46.73%), struggling to identify churn cases effectively.
6. **Weak Models:**
   - Models like **Naive Bayes** (F1-Score = 0.616552) and **Logistic Regression** underperform compared to ensemble methods, primarily due to lower recall and inability to handle oversampled data optimally

Conclusion:

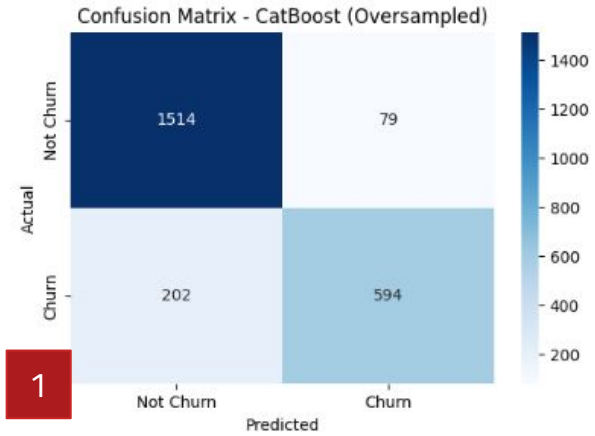
- **Top Performers:**
  - **CatBoost** is the best overall model due to its superior F1-Score and balanced metrics.
  - **LightGBM** and **XGBoost** are strong alternatives, particularly in recall and F1-Score.
- **Improvements Needed:**
  - Focus on enhancing Logistic Regression and Naive Bayes through feature engineering or hyperparameter optimization to improve recall and F1-Score.
- **Recommended Model:**
  - **CatBoost** is the top choice for oversampled data, offering the best trade-off between precision and recall, making it highly effective for churn prediction tasks.

Check out google collab to see 33 model results: Link

Top Performer of Oversampled Data

Dataset: Oversampled | Model: CatBoost

	precision	recall	f1-score	support
0	0.88	0.95	0.92	1593
1	0.88	0.75	0.81	796
accuracy			0.88	2389
macro avg	0.88	0.85	0.86	2389
weighted avg	0.88	0.88	0.88	2389



1

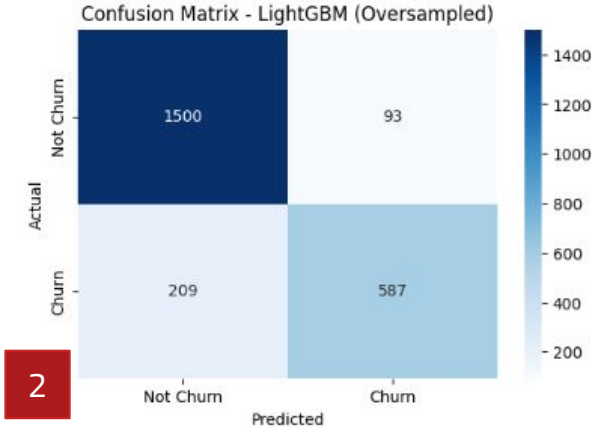
Confusion Matrix Insight for CatBoost:

- Achieves the highest F1-score (0.808713) with balanced precision (88%) and recall (75%).
- Effectively identifies churn cases with fewer false negatives (202 instances).

Summary: Best performance with F1-score 0.8087, showing balanced precision (88%) and recall (75%). Effectively handles churn cases.

Dataset: Oversampled | Model: LightGBM

	precision	recall	f1-score	support
0	0.88	0.94	0.91	1593
1	0.86	0.74	0.80	796
accuracy			0.87	2389
macro avg	0.87	0.84	0.85	2389
weighted avg	0.87	0.87	0.87	2389



2

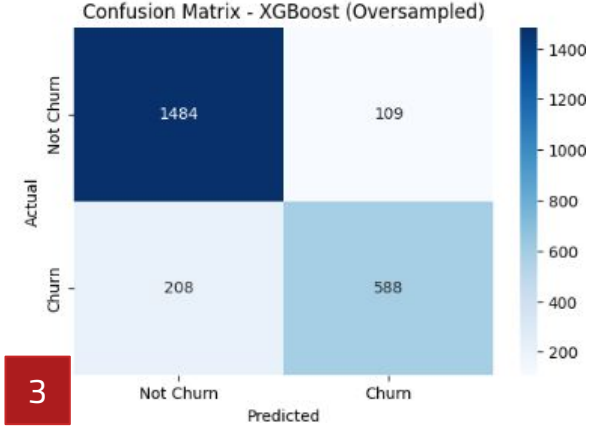
Confusion Matrix Insight for LightGBM:

- F1-score of 0.795393, slightly lower than CatBoost, but strong recall (74%).
- Handles class imbalance well, minimizing misclassifications between classes.

Summary: Strong F1-score 0.7954 with high recall (74%), slightly behind CatBoost but handles class imbalance effectively.

Dataset: Oversampled | Model: XGBoost

	precision	recall	f1-score	support
0	0.88	0.93	0.90	1593
1	0.84	0.74	0.79	796
accuracy			0.87	2389
macro avg	0.86	0.84	0.85	2389
weighted avg	0.87	0.87	0.86	2389



3

Confusion Matrix Insight for XGBoost:

- F1-score of 0.787676, with good precision (84%) and recall (74%).
- Performs reliably but generates slightly higher false positives than CatBoost and LightGBM.

Summary: Reliable F1-score 0.7877, with good precision (84%) but higher false positives than CatBoost and LightGBM.

Model Evaluation of Undersampled Data

	Dataset	Model	Accuracy	Precision	Recall	F1 Score
22	Undersampled	Logistic Regression	0.696838	0.682584	0.595588	0.636126
23	Undersampled	Random Forest	0.766630	0.759358	0.696078	0.726343
24	Undersampled	Gradient Boosting	0.782988	0.781671	0.710784	0.744544
25	Undersampled	AdaBoost	0.770992	0.761905	0.705882	0.732824
26	Undersampled	Support Vector Machine	0.790622	0.805085	0.698529	0.748031
27	Undersampled	K-Nearest Neighbors	0.745911	0.743733	0.654412	0.696219
28	Undersampled	Naive Bayes	0.750273	0.739946	0.676471	0.706786
29	Undersampled	Decision Tree	0.701200	0.665025	0.661765	0.663391
30	Undersampled	XGBoost	0.767721	0.745592	0.725490	0.735404
31	Undersampled	LightGBM	0.779716	0.769634	0.720588	0.744304
32	Undersampled	CatBoost	0.792803	0.785340	0.735294	0.759494

Key Observations:

1. **Best Accuracy:**
   - **CatBoost** achieves the highest accuracy (79.28%), demonstrating its ability to predict most cases accurately.
2. **Best Precision:**
   - **Support Vector Machine (SVM)** excels in precision (80.51%), effectively minimizing false positives.
3. **Best Recall:**
   - **Gradient Boosting** achieves the highest recall (71.07%), indicating it is the best at identifying true churn cases.
4. **Best F1-Score:**
   - **CatBoost** has the highest F1-Score (0.759494), making it the most balanced model for precision and recall.
5. **Logistic Regression Performance:**
   - Logistic Regression shows moderate performance (Accuracy = 69.68%, F1-Score = 0.636126) but struggles with recall (59.56%), indicating difficulty in identifying churn cases.
6. **Weak Models:**
   - **Decision Tree** (F1-Score = 0.663391) and **Naive Bayes** (F1-Score = 0.706786) underperform compared to ensemble models, particularly in recall and overall balance.

Conclusion:

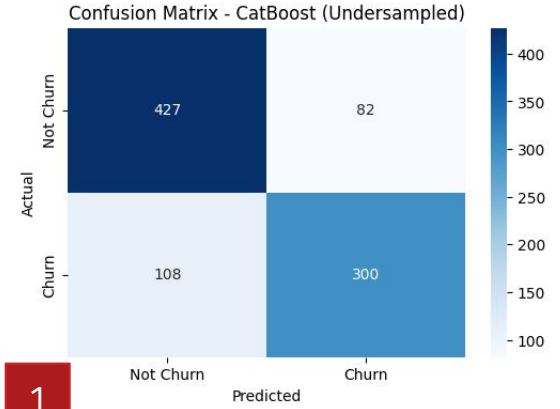
- **Top Performers:**
  - **CatBoost** emerges as the top model due to its superior F1-Score and balanced metrics.
  - **Gradient Boosting** and **LightGBM** are strong alternatives, especially in recall and F1-Score.
- **Improvements Needed:**
  - Models like Logistic Regression and Decision Tree need further optimization to enhance their recall and overall performance on undersampled data.
- **Recommended Model:**
  - **CatBoost** is recommended for its consistent performance and balanced precision-recall trade-off, making it ideal for churn prediction on undersampled data.

Check out google collab to see 33 model results: Link

Top Performer of Undersampled Data

Dataset: Undersampled | Model: CatBoost

	precision	recall	f1-score	support
0	0.80	0.84	0.82	509
1	0.79	0.74	0.76	408
accuracy			0.79	917
macro avg	0.79	0.79	0.79	917
weighted avg	0.79	0.79	0.79	917



1

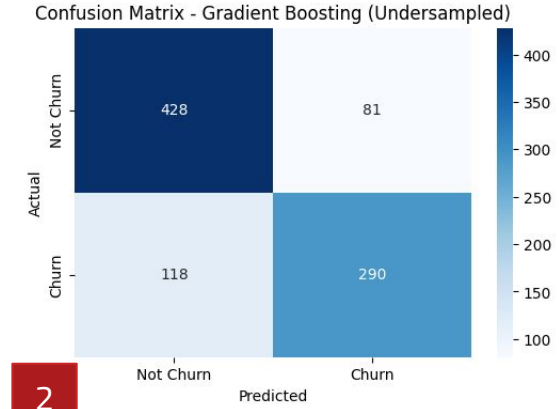
Confusion Matrix Insight for Catboost:

- Achieves the highest F1-score (0.759) among the models, with balanced precision (79%) and recall (74%).
- Effectively handles churn cases but slightly higher false negatives (108 instances).

Summary: Handles churn cases effectively but has 108 false negatives.

Dataset: Undersampled | Model: Gradient Boosting

	precision	recall	f1-score	support
0	0.78	0.84	0.81	509
1	0.78	0.71	0.74	408
accuracy			0.78	917
macro avg	0.78	0.78	0.78	917
weighted avg	0.78	0.78	0.78	917



2

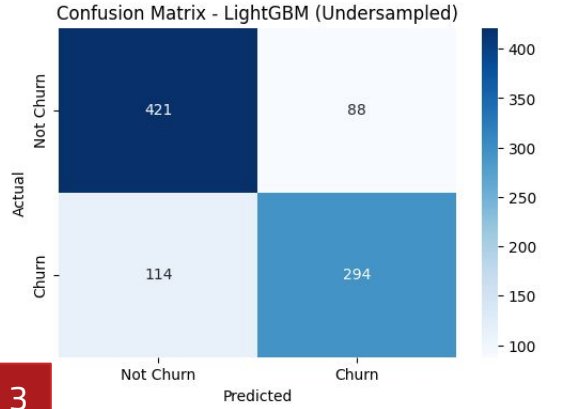
Confusion Matrix Insight for Gradient Boosting:

- F1-score of 0.744, slightly lower than CatBoost but maintains good precision (78%) and recall (71%).
- Slightly more false negatives (118 instances) compared to CatBoost.

Summary: Slightly more false negatives (118) than CatBoost.

Dataset: Undersampled | Model: LightGBM

	precision	recall	f1-score	support
0	0.79	0.83	0.81	509
1	0.77	0.72	0.74	408
accuracy			0.78	917
macro avg	0.78	0.77	0.78	917
weighted avg	0.78	0.78	0.78	917



3

Confusion Matrix Insight for LightGBM

- F1-score of 0.744, comparable to Gradient Boosting, with precision (77%) and recall (72%).
- Struggles with slightly more false positives and false negatives than CatBoost.

Summary: Slightly higher misclassification than CatBoost and Gradient Boosting.

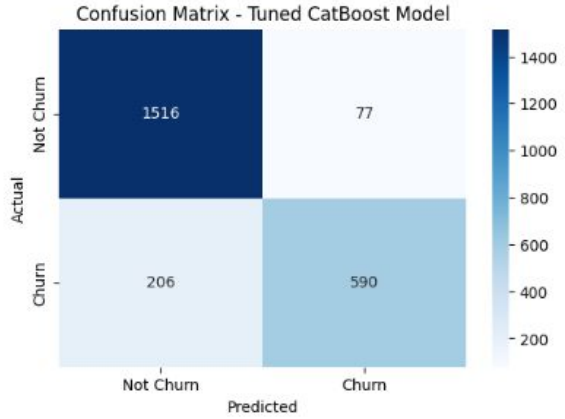
Hyperparameter Tuning

Top Models per Dataset Type:

- **Normal Data:**
  - **Gradient Boosting:** F1 = **0.611872**
  - **CatBoost:** F1 = **0.591463**
  - **Random Forest:** F1 = **0.584235**
- **Oversampled Data (SMOTE):**
  - **CatBoost:** F1 = **0.808713** ✅ *Best Overall*
  - **LightGBM:** F1 = **0.795393**
  - **XGBoost:** F1 = **0.787676**
- **Undersampled Data:**
  - **CatBoost:** F1 = **0.759494**
  - **Gradient Boosting:** F1 = **0.744544**
  - **LightGBM:** F1 = **0.744304**

Do hyperparameters on the best model

```
Fitting 3 folds for each of 81 candidates, totalling 243 fits
✅ Best Hyperparameters: {'depth': 6, 'iterations': 300, 'l2_leaf_reg': 5, 'learning_rate': 0.1}
⭐ Best CatBoost Model - F1 Score: 0.8066

🔹 Classification Report:
              precision    recall  f1-score   support

           0       0.88      0.95      0.91      1593
           1       0.88      0.74      0.81       796

    accuracy                           0.88      2389
   macro avg       0.88      0.85      0.86      2389
weighted avg       0.88      0.88      0.88      2389
```



Hyperparameter Tuning:

- The best hyperparameters for CatBoost were optimized: depth=6, iterations=300, l2_leaf_reg=5, learning_rate=0.1.
- Tuning resulted in a slight improvement in F1-score from 0.8087 to 0.8066.

Performance Insights:

- Precision: 88% - Indicates that 88% of predicted churn cases were correct.
- Recall: 74% - The model identified 74% of actual churn cases, slightly better than the untuned version.
- F1-Score: 0.8066 - A balanced measure of precision and recall, maintaining its strength in handling churn cases.
- Accuracy: 88% - Strong overall predictive capability for churn and non-churn cases.

Confusion Matrix Observations:

- True Negatives: 1516 - Most non-churn cases were correctly predicted.
- False Positives: 77 - Slight reduction in false positives after tuning.
- True Positives: 590 - Slightly fewer churn cases identified than in the untuned version.
- False Negatives: 206 - False negatives slightly increased, indicating room for further improvement.

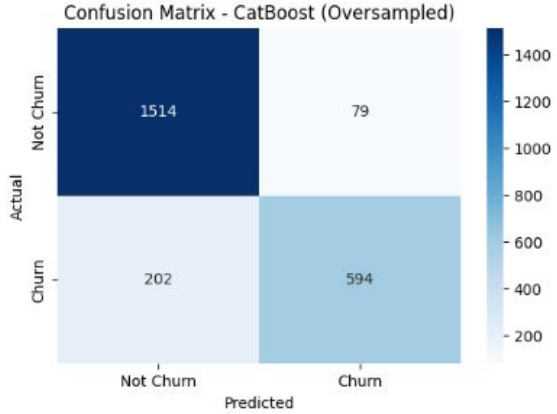
Conclusion:

- The tuned CatBoost model continues to demonstrate robust performance with high precision and accuracy.
- While recall and F1-score showed a marginal drop, the model remains the best performer overall, effectively balancing the prediction of churn and non-churn cases.

Best Model

Dataset: Oversampled | Model: CatBoost

	precision	recall	f1-score	support
0	0.88	0.95	0.92	1593
1	0.88	0.75	0.81	796
accuracy			0.88	2389
macro avg	0.88	0.85	0.86	2389
weighted avg	0.88	0.88	0.88	2389



	Dataset	Model	Accuracy	Precision	Recall	F1 Score
21	Oversampled	CatBoost	0.882378	0.882615	0.746231	0.808713

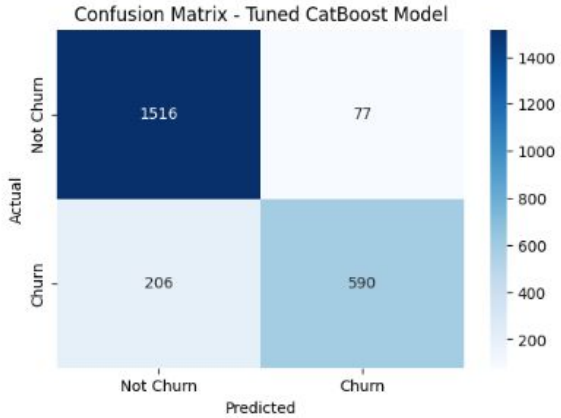
Before Hyperparameter - Best Model

1. **F1-Score:** 0.8087 (Best overall model).
2. **Performance:** Achieved optimal balance between precision (88.26%) and recall (74.62%).

Fitting 3 folds for each of 81 candidates, totalling 243 fits
✅ Best Hyperparameters: {'depth': 6, 'iterations': 300, 'l2_leaf_reg': 5, 'learning_rate': 0.1}
⭐ Best CatBoost Model - F1 Score: 0.8066

🔹 Classification Report:

	precision	recall	f1-score	support
0	0.88	0.95	0.91	1593
1	0.88	0.74	0.81	796
accuracy			0.88	2389
macro avg	0.88	0.85	0.86	2389
weighted avg	0.88	0.88	0.88	2389

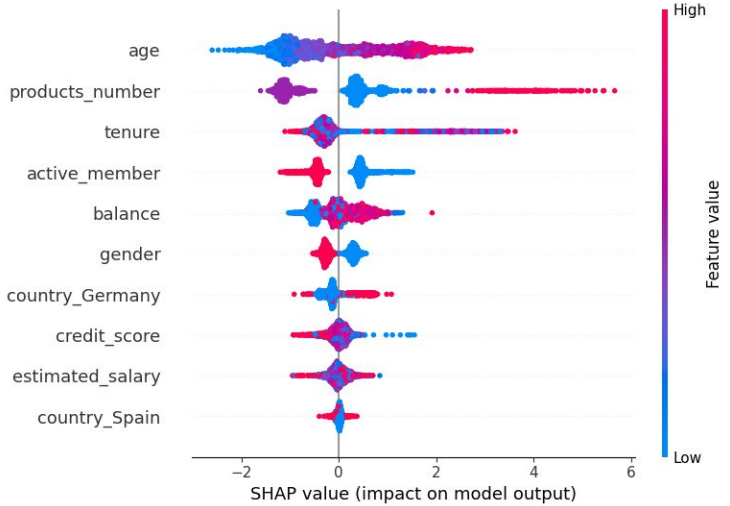


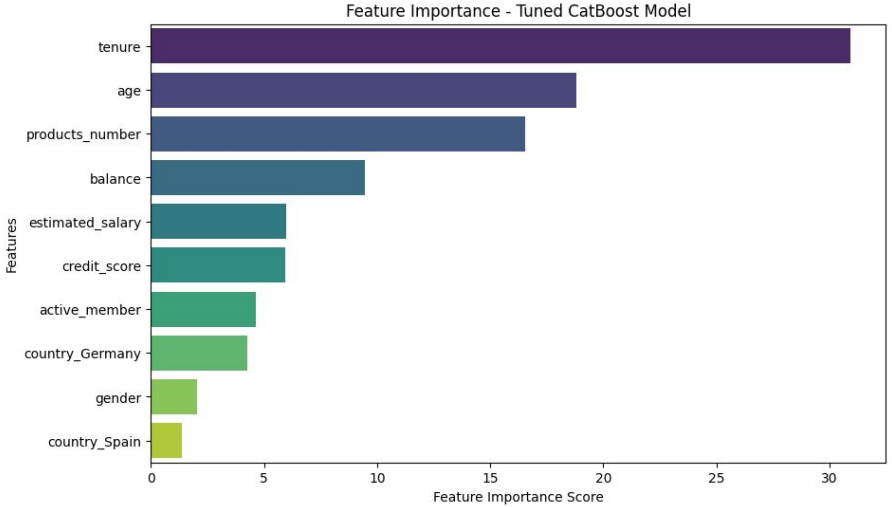
After Hyperparameter - Not Best Model

1. **F1-Score:** Slightly decreased to 0.8066.
2. **Performance:** Marginal improvement in reducing false positives but slight decline in recall.

Conclusion: The pre-tuned CatBoost model performs slightly better overall, demonstrating that default parameters can achieve optimal performance for this dataset.

Feature Importance





SHAP Values Insight:

- **Tenure** and **Age** are the most significant features impacting the model's predictions. Longer tenure and older age are associated with lower churn risk.
- **Balance** and **Products Number** also have a strong impact, where higher balance and owning fewer products influence churn probability.
- **Country and Gender** have less impact compared to financial and behavioral features.

Feature Importance Insight (CatBoost):

- **Tenure** is the most critical feature, indicating customer retention duration is key in predicting churn.
- **Age** and **Products Number** follow, highlighting demographic and product engagement as vital factors.
- Features like **Balance** and **Credit Score** remain influential but are secondary to tenure and age.
- Country and gender contribute minimally, suggesting churn is less region or gender-specific.

04

Business Recommendations

Reducing Churn

1. **Address High-Churn Age Groups:**
   - Older customers are more likely to churn based on their SHAP value impact. Provide special offers, personalized communication, or financial advisory services to retain this demographic.
   - Younger customers with low tenure should be targeted with onboarding programs to improve engagement.
2. **Focus on Low-Tenure Customers:**
   - Tenure is the most significant factor in churn prediction. Implement programs that incentivize loyalty in the early stages of a customer's lifecycle, such as welcome discounts or initial rewards for milestone completions.
3. **Improve Retention for Low-Balance Customers:**
   - Customers with lower balances are more prone to churn. Introduce flexible payment plans or rewards for increasing their account balance.
4. **Tailor Strategies Based on Country Segmentation:**
   - Customers from Spain and Germany show distinct churn behaviors. Develop country-specific campaigns, such as language-specific messaging or culturally relevant promotions, to engage these customer segments.
5. **Enhance Engagement for Low Product Usage:**
   - Customers with fewer products are at higher churn risk. Cross-sell relevant products or services to enhance their involvement and loyalty to the company.

Optimizing Customer Lifetime Value (CLV)

1. **Leverage High CLV Segments (VIP & High):**
   - Focus on upselling and premium offers for VIP and High CLV customers. For instance, offer premium services or tiered loyalty benefits that align with their high spending capacity.
2. **Promote Balanced Spending:**
   - Encourage Medium CLV customers to move into higher tiers by offering exclusive rewards for additional purchases or higher product engagement.
3. **Increase Tenure for Higher CLV:**
   - The relationship between tenure and CLV suggests a need to promote long-term contracts or retention campaigns that reward customers for staying longer.
4. **Monitor Credit Score Impact:**
   - Customers with lower credit scores tend to churn more. Provide financial education programs or personalized financial advice to help these customers improve their credit behavior.
5. **Incentivize Product Expansion:**
   - Customers with more products tend to have higher CLV. Use bundling strategies or discounts for adding more products to increase overall customer value.

Specific Campaign Suggestions

Cross-Selling Initiatives

Highlight complementary products to low-product customers through targeted recommendations.

Balance Boost Program

Incentivize customers with low balances by introducing cashback rewards for reaching higher balance thresholds.

Country-Specific Promotions

Launch campaigns tailored for Spain and Germany to address their unique churn risks and preferences.

“Stay Longer, Save More" Campaign

Offer progressive discounts or rewards for customers as their tenure increases.

05

Conclusion

Conclusion

This project effectively tackled the challenges of customer churn and profitability optimization in the banking sector. By developing a churn prediction model and conducting a Customer Lifetime Value (CLV) analysis, the following key outcomes were achieved:

1. **Churn Prediction:**
   - The **CatBoost model** proved to be the most effective, achieving an **F1-score of 0.808** on oversampled data without hyperparameter tuning.
   - Key drivers of churn include **tenure, age, and account balance**, with newer customers, older demographics, and low-balance accounts being at higher risk.
2. **Customer Lifetime Value (CLV):**
   - Customers were segmented into **Low, Medium, High, and VIP tiers**, enabling targeted and tier-specific retention strategies.
   - High-CLV customers exhibited **longer tenure, greater product engagement, and higher account balances**, signifying their importance to overall profitability.
3. **Business Recommendations:**
   - Focus on retaining **high-value customers** through tailored loyalty programs and personalized benefits.
   - Mitigate churn risk by proactively engaging at-risk customers, especially those with **shorter tenure and fewer product interactions**.
   - Implement **upsell and cross-sell strategies** for medium-value customers to boost their profitability.

"The cost of retaining a customer is far less than the cost of acquiring a new one. By understanding what keeps customers loyal, you maximize their lifetime value and build a foundation for sustainable growth."

THANK YOU!

Do you have any questions?

hijirdw@gmail.com
https://www.linkedin.com/in/hijirdella/

https://github.com/hijirdella/Bank-Customer-Churn-Prediction-and-CLV-Optimization

Hijir Della Wirasti
Business Intelligence