Customer Lifetime Value (CLV) Prediction Project Report

Objective

The objective of this project is to analyze customer purchasing behavior and predict **Customer Lifetime Value (CLV)** using SQL and Python. This analysis helps businesses identify high-value customers, enhance retention, and optimize marketing strategies.

---

1. Data Source

- **Dataset:** Online Retail CLV.csv
- **Records:** ~540K transactions
- **Fields:** Invoice_No, Stock_Code, Description, Quantity, Invoice_Date, Unit_Price, Customer_ID, Country

---

2. SQL-Based Data Analysis

Step 1: Database Creation & Data Loading

```
CREATE DATABASE online_retail_clv;
USE online_retail_clv;

CREATE TABLE online_retail (
   Invoice_No VARCHAR(20),
   Stock_Code VARCHAR(20),
   Description TEXT,
   Quantity INT,
   Invoice_Date DATETIME,
   Unit_Price DECIMAL(10, 2),
   Customer_ID INT NULL,
   Country VARCHAR(100)
);

LOAD DATA INFILE 'Online Retail CLV.csv'
INTO TABLE online_retail
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
IGNORE 1 ROWS;
```

Step 2: Data Understanding

- Total Transactions: ~541,909

- Unique Customers: ~4,300

```
SELECT COUNT(DISTINCT Invoice_No), COUNT(DISTINCT Customer_ID)
FROM online_retail;
```

Step 3: Profit Calculation

```
ALTER TABLE online_retail ADD COLUMN Profit DECIMAL(10,2);
UPDATE online_retail SET Profit = Quantity * Unit_Price;
```

Step 4: Revenue Analysis

- Top 10 products by total sold
- Country-wise revenue
- Monthly revenue trends

```
SELECT Country, SUM(Profit) AS Revenue
FROM online_retail
GROUP BY Country
ORDER BY Revenue DESC;
```

Insights: - The **United Kingdom** contributes ~85% of total sales. - Holiday months (November–December) show significant revenue spikes.

Step 5: Customer Insights

```
SELECT Customer_ID, SUM(Profit) AS Total_Spent
FROM online_retail
GROUP BY Customer_ID
ORDER BY Total_Spent DESC
LIMIT 10;
```

Findings: A small fraction of customers generate the majority of sales (Pareto Principle).

Step 6: RFM Analysis

```
SELECT @ref_date := MAX(Invoice_Date) FROM online_retail;

SELECT
   Customer_ID,
   DATEDIFF(@ref_date, MAX(Invoice_Date)) AS RecencyDays,
   COUNT(DISTINCT Invoice_No) AS Frequency,
   ROUND(SUM(Profit), 2) AS Monetary
FROM online_retail
GROUP BY Customer_ID;
```

Step 7: Feature View Creation

```
CREATE OR REPLACE VIEW customer_ltv_features AS
SELECT
   Customer_ID,
```

```sql
    DATEDIFF((SELECT MAX(Invoice_Date) FROM online_retail), MAX(Invoice_Date)) AS RecencyDays,
    COUNT(DISTINCT Invoice_No) AS Frequency,
    ROUND(SUM(Profit)/COUNT(DISTINCT Invoice_No), 2) AS AOV,
    ROUND(SUM(Profit), 2) AS TotalSpend
FROM online_retail
WHERE Customer_ID IS NOT NULL
GROUP BY Customer_ID;
```

---

3. Python-Based LTV Modeling

Libraries Used

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
```

Step 1: Load Data

```python
ltv_df = pd.read_csv('final_ltv_predictions.csv')
ltv_df.head()
```

Step 2: Feature Engineering

Features: RecencyDays, Frequency, AOV, TotalSpend
Target: Predicted_LTV

```python
X = ltv_df[['RecencyDays', 'Frequency', 'AOV', 'TotalSpend']]
y = ltv_df['Predicted_LTV']
```

Step 3: Model Training

```python
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(X_train, y_train)
```

Step 4: Model Evaluation

```python
y_pred = model.predict(X_test)
print('R2:', r2_score(y_test, y_pred))
print('MSE:', mean_squared_error(y_test, y_pred))
```

Results: - $R^2 \approx 0.85$ (Strong correlation) - MSE: Low (Accurate model)

Step 5: Visualization

```python
sns.scatterplot(x=y_test, y=y_pred)
plt.xlabel('Actual LTV')
```

```
plt.ylabel('Predicted LTV')
plt.title('Actual vs Predicted LTV')
plt.show()
```

---

4. Business Insights

Metric	Description	Business Use
Recency	Days since last purchase	Identify dormant customers
Frequency	Number of purchases	Target loyal buyers
AOV	Average Order Value	Segment premium buyers
TotalSpend	Overall revenue	Rank high-value customers
Predicted_LTV	Expected lifetime value	Guide retention marketing

Findings: - 20% of customers drive ~80% of total revenue. - Frequent, high-spend users have the highest predicted LTV.

---

5. Recommendations

1. Segment customers using RFM features to identify high-value groups.
2. Introduce loyalty programs for frequent buyers.
3. Re-engage inactive customers through targeted campaigns.
4. Offer upsell recommendations for customers with high AOV.

---

6. Conclusion

This project integrates **SQL, Python, and Machine Learning** to predict Customer Lifetime Value accurately. The insights enable data-driven marketing, targeted retention, and improved profitability.

---

7. Tools & Technologies

Category	Tools
Database	MySQL
Programming	Python (Pandas, Scikit-learn)
Visualization	Power BI, Matplotlib, Seaborn
ML Model	Linear Regression
Dataset	Online Retail (UCI Repository)
Output	final_ltv_predictions.csv
