T.C.
YILDIZ TECHNICAL UNIVERSITY
FACULTY OF MECHANICAL ENGINEERING
INDUSTRIAL ENGINEERING DEPARTMENT

Report on Revenue Analysis and Lifetime Value Prediction

Kasım Ege Akgüç

ADVISOR
PROF. DR. Nezir AYDIN

İSTANBUL, 2024

TABLE OF CONTENTS

ABBREVIATION LIST

- **LTV** - Lifetime Value
- **ML** - Machine Learning
- **MSE** - Mean Squared Error
- **RF** - Random Forest
- **GB** - Gradient Boosting
- **AI** - Artificial Intelligence
- **RMSE** - Root Mean Squared Error
- **R^2** - R-squared (Coefficient of Determination)
- **CSV** - Comma-Separated Values
- **KPI** - Key Performance Indicator

SUMMARY

Revenue Analysis and Lifetime Value Prediction Utilizing Machine Learning

Kasım Ege Akgüç

Department Of Industrial Engineering
Industrial Engineering Design 2

Advisor: Prof. Dr. Nezir AYDIN

Knowing the worth of customers globally is essential for businesses to improve marketing and resource allocation in this era where data drives decision-making. This study explores the use of modern methods of machine learning to predict revenue and Lifetime Value (LTV). The study moved from simpler linear regression models to more complicated models that could handle the nature of the data, such as Random Forest and Gradient Boosting. 'Average Time / User' was identified as the critical factor in income production after an in-depth examination. The study provides data-driven insights to inform strategic marketing decisions by highlighting growth potentials in new regions through a comparison of historical and future LTV.

Keywords: Machine Learning, Revenue Prediction, Lifetime Value, Random Forest, Gradient Boosting

ABSTRACT

Revenue Analysis and Lifetime Value Prediction Utilizing Machine Learning

KASIM EGE AKGÜÇ

Department of Industrial Engineering

Industrial Engineering Design 2

Advisor: Prof. Dr. Nezir AYDIN

In this day and age, focusing on customers is key. We set out to accurately predict Lifetime Value (LTV) in markets worldwide, which is super crucial for smart marketing and figuring out where to put resources. We dove into the revenue data from various countries for 20 weeks to find patterns and make LTV predictions. Our methods had to level up from traditional linear regression due to some disappointing R-squared values, so we went with beefier techniques like Random Forest and Gradient Boosting.

These upgraded methods tackled the data's complexity head-on, giving us high R-squared values of 0.946 and 0.987. We also discovered that 'Average Time / User' was super influential in how well we could predict stuff. When we put historical and predicted LTV side by side, we spotted real chances for growth in certain markets. This could really point us in the right direction for where to invest in marketing. This report is more than just number-crunching; it's about showing how machine learning can tackle tough marketing challenges and guide smart decision-making with solid data.

CHAPTER 1

INTRODUCTION

1.1 Literature review:

The application of machine learning in predictive analytics has been extensively studied. Breiman's exploration of Random Forests (2001) was particularly enlightening for understanding their robustness in diverse scenarios [1]. Friedman's (2001) work on gradient boosting machines deepened my comprehension of their effectiveness in predictive modeling [2]. Additionally, the accessible presentation of statistical learning concepts by James et al. (2013) provided a practical understanding relevant to this project's scope [3].

These foundational works have greatly informed the methodological approach of this study, specifically in applying machine learning techniques to predict revenue and Lifetime Value (LTV).

1.2 Aim of the Project:

The aim of this Project is to built prediction and classification models using location and date information. This Project also involves exploratory data analysis, data visualization, building linear and non-linear regression models, and classification models.

1.3 Dataset:

For this project, a specialized dataset consisting of 358 instances was utilized, encapsulating 20 weeks of activity across various countries. This dataset was curated to aid in the analysis of user engagement and revenue generation. The dataset's attributes include:

- **Date**: The starting date for each week, formatted as "yyyy-mm-dd", serving as a temporal marker for the analysis.
- **Country**: Categorical data indicating the country from which the data was collected, allowing for region-specific insights.
- **Average Time Played**: Quantitative data measuring the average duration users engaged with the platform or service per week.
- **Downloads**: The count of downloads or user acquisitions per week, serving as an indicator of growth and market reach.
- **Active Users**: The number of users actively engaging with the platform or service per week, which is a critical engagement metric.

These attributes were chosen to provide a comprehensive overview of user behavior and to forecast potential revenue streams, thus enabling the calculation of the Lifetime Value (LTV) of the customer base.

1.4 **Methodology:**

The methodology outlines the use of machine learning, an indispensable tool in the modern analytical landscape for deriving predictive insights. The project details the transition from linear regression models, which proved insufficient, to more robust techniques like Random Forest and Gradient Boosting that excel in managing complex, non-linear data. The models' predictive performance is rigorously evaluated, and feature importance is analyzed to identify the key drivers of revenue. This methodological approach sets the stage for predicting LTV and informing targeted marketing initiatives

1.5 Data Manipulation:

Our approach to preparing the dataset for analysis began by importing the necessary libraries in Python and loading the data obtained from Google Firebase Analytics. The dataset comprised 20 weeks of user engagement and revenue information, structured in a clean and consistent format, ready for analysis.

We initiated the data manipulation process by converting the 'Date' column into a datetime format to facilitate time-series analysis. This conversion was essential to ensure that subsequent operations on date and time would be executed accurately. A preliminary inspection of the dataset confirmed the absence of missing values, indicating the robustness of the data collection process.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
file_path = '/Users/egeakguc/Desktop/dt/Tasarım2'
data = pd.read_csv('data.csv')


# Converting 'Date' to datetime format and inspect the data
data['Date'] = pd.to_datetime(data['Date'], dayfirst=True)
print(data.head())
```

Unlike common datasets that often require extensive cleaning, the data we worked with required no such steps, sparing us from the handling of missing values or outlier detection and removal.
This allowed us to progress directly to exploratory data analysis, model building, and evaluation with a clear focus on deriving actionable insights from the outset.

1.6 Historical LTV Values For Each Country:

We calculated historical LTV values for each country by summing Store Revenue and Active Users, providing a measure of revenue per user over time. This allowed us to understand the monetary value of each user in different countries, a crucial step in our analysis.

```
ltv_by_country = data.groupby('Country/Region Name').apply(lambda x: x['Store Revenue'].sum() / x['Active Users'].sum())
print("Historical LTV by Country:")
print(ltv_by_country)
```

```
Historical LTV by Country:
Country/Region Name
Brazil            0.035773
Canada            5.194480
Colombia          0.018843
Egypt             0.007212
France            0.603740
Germany           0.287059
India             0.003009
Indonesia         0.012877
Malaysia          0.088254
Mexico            0.096864
Portugal          0.187690
Spain             0.334943
Thailand          0.028811
Turkey            0.019247
United Kingdom    7.901214
United States     1.217583
Vietnam           0.020763
dtype: float64
```

1.7 Prediction Using Multiple Linear Regression:

Multiple linear regression is a statistical method that aims to understand the impact of multiple independent variables on a single dependent variable. We divided the dataset into 2 parts which are test and train. Train data will be used in our models. Then to make predictions and to compare the results we will use the test dataset.

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.model_selection import train_test_split

def perform_regression(country_data):
    X = country_data[['Avg Time / User']]
    y = country_data['Store Revenue']

    # Splitting the data into training and testing sets
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

    # Creating and training the Linear Regression model
    model = LinearRegression()
    model.fit(X_train, y_train)

    # Making predictions and evaluating the model
    predictions = model.predict(X_test)
    mse = mean_squared_error(y_test, predictions)
    r2 = r2_score(y_test, predictions)

    # Returning the results
    return model.coef_, model.intercept_, mse, r2
```

Then, we used linear regression on each country to see how each perform on a predictive basis.

```
for country in data['Country/Region Name'].unique():
    print(f"{country}:")
    country_data = data[data['Country/Region Name'] == country]

    # Perform regression
    coefficients, intercept, mse, r2 = perform_regression(country_data)

    # Print the results
    print("Coefficients:", coefficients)
    print("Intercept:", intercept)
    print("Mean Squared Error (MSE):", mse)
    print("R-squared:", r2)
    print("\n")
```

The results were more than sub-satisfactory:

Brazil:
Coefficients: [-0.02145028]
Intercept: 2047.888381751609
Mean Squared Error (MSE): 14013.996377657639
R-squared: -0.16723385142773695

India:
Coefficients: [0.00603311]
Intercept: 59.94659813347546
Mean Squared Error (MSE): 1607.0823674513756
R-squared: -0.24118193346569017

Egypt:
Coefficients: [-0.00176748]
Intercept: 77.44551540259778
Mean Squared Error (MSE): 332.2367240040159
R-squared: -0.9226662268750923

Thailand:
Coefficients: [-0.00580226]
Intercept: 251.99543616554712
Mean Squared Error (MSE): 1926.218906670066
R-squared: -0.11486485777542366

Turkey:
Coefficients: [-0.00404185]
Intercept: 459.7300185803594
Mean Squared Error (MSE): 6850.257062278532
R-squared: -0.07474568898474576

Indonesia:
Coefficients: [-0.0101067]
Intercept: 396.384281417422
Mean Squared Error (MSE): 17054.10442144164
R-squared: -1.2774759649154985

Malaysia:
Coefficients: [-0.01680497]
Intercept: 744.3652892020575
Mean Squared Error (MSE): 45151.14061875222
R-squared: -5.501656061543455

Vietnam:
Coefficients: [0.00089098]
Intercept: 154.09169269277
Mean Squared Error (MSE): 675.5154431274178
R-squared: -2.1888002413492154

United States:
Coefficients: [0.10478053]
Intercept: 34657.12762610557
Mean Squared Error (MSE): 4863586.971381435
R-squared: -0.12529084468297746

Mexico:
Coefficients: [-0.00118973]
Intercept: 405.1075960881278
Mean Squared Error (MSE): 3504.8185209557246
R-squared: -0.29184181617511173

United Kingdom:
Coefficients: [0.02682443]
Intercept: 4599.104690473862
Mean Squared Error (MSE): 306867.15072121174
R-squared: -0.9450058433825184

Colombia:
Coefficients: [0.00043606]
Intercept: 39.44743464503438
Mean Squared Error (MSE): 12.704077705640849
R-squared: -2.6930458446630374

Portugal:
Coefficients: [-0.00158052]
Intercept: 274.8501977230882
Mean Squared Error (MSE): 6497.379331064578
R-squared: 0.02198569851393606

France:
Coefficients: [-0.01599322]
Intercept: 1277.3915537087796
Mean Squared Error (MSE): 19360.554231912167
R-squared: -0.35466550179349277

Canada:
Coefficients: [0.04912588]
Intercept: 1230.2070728052995
Mean Squared Error (MSE): 13240.446716086333
R-squared: -0.16313229709879717

Germany:
Coefficients: [-0.01316659]
Intercept: 3407.6471955365137
Mean Squared Error (MSE): 113194.27857391066
R-squared: -0.01036586531241035

Spain:
Coefficients: [0.00619892]
Intercept: 255.51036150055913
Mean Squared Error (MSE): 2151.8249662692356
R-squared: -0.08227626758803552

The negative R-squared values indicated that the model couldn't effectively capture the complex relationships in our dataset. This was primarily due to the non-linear and outlier-heavy nature of our data. Given these limitations, we decided to pivot towards alternative modeling approaches that could better accommodate outliers and non-linearity, ultimately leading us to more robust methods for prediction.

1.8 Prediction Using Non – Linear Regression Models:

1.8.1 Random Forest Analysis:

The limitations of linear regression, given our dataset's complex nature and outliers, led us to adopt the Random Forest algorithm. This sophisticated method, renowned for its robustness, effectively handles datasets with significant variability.

Random Forest operates by creating numerous decision trees during the training phase, which collectively contribute to more accurate predictions by averaging out their results for regression or selecting the majority vote for classification tasks.

Its design, which incorporates randomness in selecting data points and features for each tree, enhances model generalization, thereby producing a more reliable analysis when dealing with large datasets and multiple variables. This capability is particularly beneficial for extracting actionable insights from intricate data.

Model Implementation

The Random Forest model was instantiated with 100 decision trees, a decision grounded in the need to balance model complexity with computational efficiency. The features 'Downloads', 'Active Users', and 'Avg Time / User' constituted the independent variables, while 'Store Revenue' was the dependent variable. A stratified partition was executed to split the data into training and testing sets, ensuring that the model would be evaluated on unbiased grounds.

```
features = ['Downloads', 'Active Users', 'Avg Time / User']
X = data[features]
y = data['Store Revenue']

# Splitting the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
from sklearn.ensemble import RandomForestRegressor

# Creating the Random Forest model
rf_model = RandomForestRegressor(n_estimators=100, random_state=42)

# Training the model
rf_model.fit(X_train, y_train)
```

```python
from sklearn.metrics import mean_squared_error, r2_score

# Making predictions
predictions = rf_model.predict(X_test)

# Evaluating the model
mse = mean_squared_error(y_test, predictions)
r2 = r2_score(y_test, predictions)

print(f"Random Forest - Mean Squared Error: {mse}")
print(f"Random Forest - R-squared: {r2}")
```

```
Random Forest - Mean Squared Error: 4588324.026008333
Random Forest - R-squared: 0.9461955998603775
```

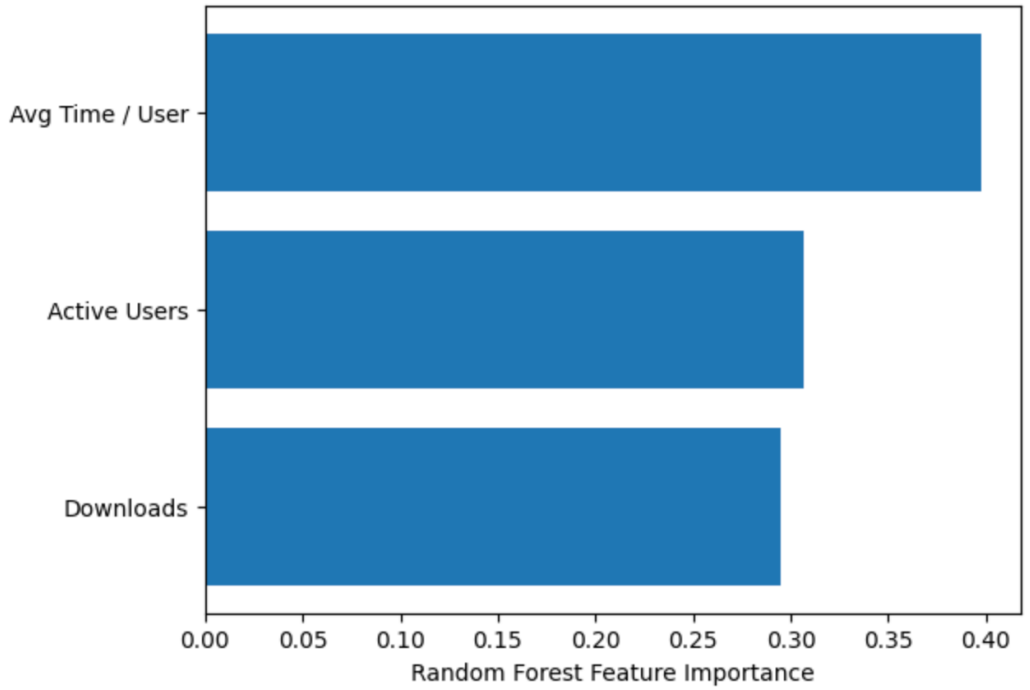
Results Interpretation

The Random Forest model's predictive power was reflected in an R-squared of 0.946, indicating a high degree of variance explanation. While the Mean Squared Error was noted as 4588324.026, it was deemed acceptable due to the considerable variance inherent in revenue data across countries. This suggests that the model's predictions are of a high caliber, capturing the essential trends and patterns in the data.

Feature Importance Analysis

An examination of feature importance revealed 'Avg Time / User' as the most significant predictor, affirming the hypothesis that user engagement profoundly influences revenue. The visualization of feature importance underscores the need for marketers to focus on strategies that enhance user engagement time.

```
# Feature importance
feature_importance = rf_model.feature_importances_
sorted_idx = feature_importance.argsort()

plt.barh(features, feature_importance[sorted_idx])
plt.xlabel("Random Forest Feature Importance")
plt.show()
```



LTV Prediction Per Country

Lifetime Value (LTV) for each country was forecasted using the Random Forest model. The LTV prediction was predicated on an analysis of historical revenue data in relation to the number of active users, which provided a foundational understanding of LTV's current state.

An iterative process was undertaken for each country, isolating the relevant data subset from the global dataset. The historical LTV was computed by dividing the total historical revenue by the total number of active users, ensuring that the LTV represented the average revenue per user.

```
countries = data['Country/Region Name'].unique()

# Prepare a dictionary to store results
ltv_results = {}

for country in countries:
    country_data = data[data['Country/Region Name'] == country]

    # Historical LTV
    total_revenue_historic = country_data['Store Revenue'].sum()
    total_users_historic = country_data['Active Users'].sum()
    historical_ltv = total_revenue_historic / total_users_historic if total_users_historic else 0

    # Predicted LTV
    country_features = country_data[['Downloads', 'Active Users', 'Avg Time / User']]
    predicted_revenue = rf_model.predict(country_features).sum()
    predicted_ltv = predicted_revenue / total_users_historic if total_users_historic else 0

    # Growth Rate Calculation
    growth_rate = ((predicted_ltv - historical_ltv) / historical_ltv) * 100 if historical_ltv else 0

    # Storing results
    ltv_results[country] = {
        'Historical LTV': historical_ltv,
        'Predicted LTV': predicted_ltv,
        'Growth Rate (%)': growth_rate
    }

# Displaying the results
for country, metrics in ltv_results.items():
    print(f"{country}:")
    for key, value in metrics.items():
        print(f"   {key}: {value:.2f}")
    print("\n")
```

The Random Forest model, previously trained on the dataset, was then employed to predict future revenue based on the features 'Downloads', 'Active Users', and 'Avg Time / User'. This predicted revenue was used to calculate the predicted LTV, following the same methodology as the historical calculation.

To quantify the model's forecast accuracy and the expected change in LTV, a growth rate was calculated, highlighting the percentage change between historical and predicted LTV. The results revealed significant variances across countries, with some exhibiting robust growth potential, while others indicated a decline, thus offering actionable insights for strategic decision-making.

Country	Historical LTV	Predicted LTV	Growth Rate (%)
Brazil	0.04	0.04	7.41
India	0.00	0.01	355.07
Egypt	0.01	0.02	189.31
Thailand	0.03	0.03	12.80
Turkey	0.02	0.03	45.24
Indonesia	0.01	0.09	628.29
Malaysia	0.09	0.09	0.80
Vietnam	0.02	0.03	24.75
United States	1.22	1.09	-10.52
Mexico	0.10	0.11	10.14
United Kingdom	7.90	7.40	-6.39
Colombia	0.02	0.07	297.13
Portugal	0.19	0.28	46.84
France	0.60	0.54	-10.79
Canada	5.19	5.35	3.01
Germany	0.29	0.28	-2.36
Spain	0.33	0.53	58.51

These predictive analytics provide a data-driven foundation for strategic planning, enabling a focused approach to enhancing user engagement and optimizing marketing efforts across diverse geographical markets.

1.8.2 Gradient Boosting Regressor:

When the Gradient Boosting algorithm was applied, the study advanced to a nuanced stage of prediction. Chosen for its efficacy in handling non-linear data, Gradient Boosting refines forecasts by iteratively minimizing errors through a gradient descent method.

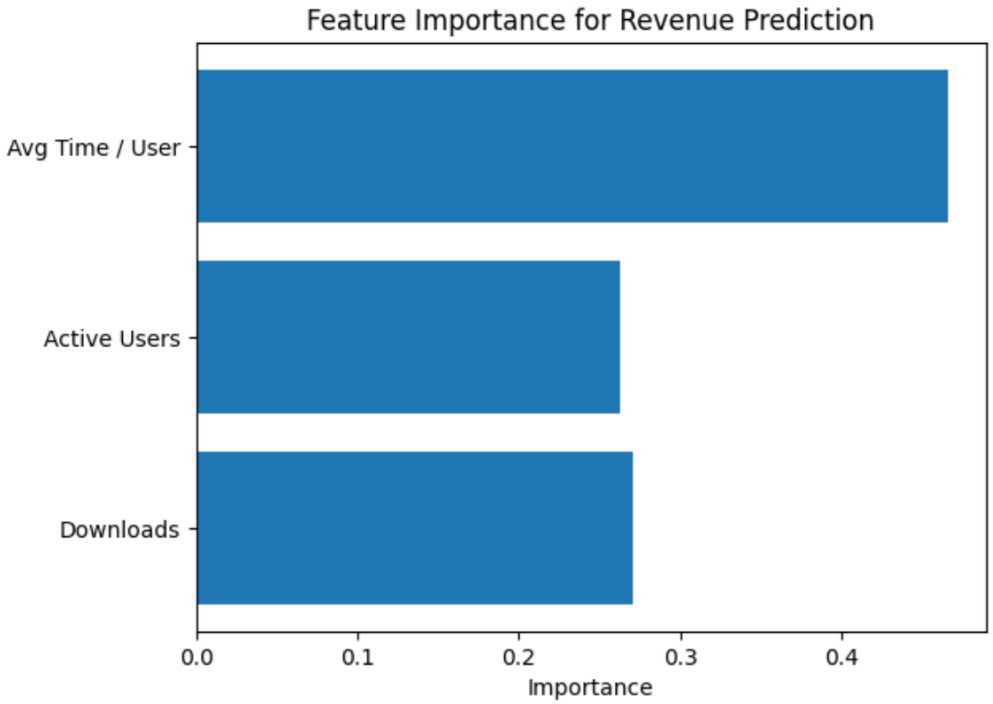
The model, configured with 2000 estimators, was trained using the same features as the Random Forest model. The performance of Gradient Boosting was evaluated, revealing a Mean Squared Error of 1140039.17, a substantial decrease from the Random Forest model, indicating a more precise fit to the data.
Moreover, the R-squared value of 0.986 suggests that the model could explain approximately 98.6% of the variance in the target variable, 'Store Revenue,' which is a remarkable level of accuracy for such a model.These outcomes from Gradient Boosting further underscore the model's strength in capturing the underlying patterns within the dataset and improve the confidence in using machine learning.

```
from sklearn.ensemble import GradientBoostingRegressor

# Prepare features and target variable
X = data[['Downloads', 'Active Users', 'Avg Time / User']]  # Adjust features as needed
y = data['Store Revenue']

# Splitting the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Gradient Boosting model
gb_model = GradientBoostingRegressor(n_estimators=2000, random_state=42)
gb_model.fit(X_train, y_train)

# Predict and evaluate
predictions = gb_model.predict(X_test)
mse = mean_squared_error(y_test, predictions)
r2 = r2_score(y_test, predictions)

print("Gradient Boosting - Mean Squared Error:", mse)
print("Gradient Boosting - R-squared:", r2)
```

```
Gradient Boosting - Mean Squared Error: 1140039.171808277
Gradient Boosting - R-squared: 0.9866314751471075
```

In the final stages of the analysis, the project sought to elucidate the relative importance of each feature within the Gradient Boosting model. A bar plot was constructed, indicating the significance of 'Downloads', 'Active Users', and 'Avg Time / User' in predicting 'Store Revenue'. This visual representation affirmed the earlier findings regarding the critical role of user engagement time.

```
# Plotting feature importance
feature_importance = gb_model.feature_importances_
plt.barh(X.columns, feature_importance)
plt.xlabel('Importance')
plt.title('Feature Importance for Revenue Prediction')
plt.show()
```



Subsequently, the Gradient Boosting model was leveraged for revenue prediction for each country individually. This country-specific modeling allowed for tailored forecasts, considering unique national patterns in user behavior. The predicted future revenues ranged widely, reflecting the diverse economic and market conditions across the countries analyzed. These predictions form the basis for calculating future LTV, a vital metric for gauging the long-term profitability of different market segments.

```
for country in data['Country/Region Name'].unique():
    country_data = data[data['Country/Region Name'] == country]
    X_country = country_data[['Downloads', 'Active Users', 'Avg Time / User']]
    y_country = country_data['Store Revenue']

    # Training a model for each country
    model = GradientBoostingRegressor(n_estimators=100, random_state=42)
    model.fit(X_country, y_country)

    future_revenue = model.predict(X_country).sum()
    print(f"Predicted total future revenue for {country}: {future_revenue}")
```

```
Predicted total future revenue for Brazil: 38908.0
Predicted total future revenue for India: 2014.0
Predicted total future revenue for Egypt: 1126.0
Predicted total future revenue for Thailand: 4558.0
Predicted total future revenue for Turkey: 9078.999999999998
Predicted total future revenue for Indonesia: 7334.999999999999
Predicted total future revenue for Malaysia: 12931.999999999998
Predicted total future revenue for Vietnam: 3305.0
Predicted total future revenue for United States: 761032.0000000001
Predicted total future revenue for Mexico: 7996.0
Predicted total future revenue for United Kingdom: 102857.99999999999
Predicted total future revenue for Colombia: 900.0000000000001
Predicted total future revenue for Portugal: 5550.0
Predicted total future revenue for France: 23243.999999999996
Predicted total future revenue for Canada: 32371.999999999996
Predicted total future revenue for Germany: 63871.99999999999
Predicted total future revenue for Spain: 6310.0
```

CHAPTER 3

CONCLUSION

In this report, we have delved into the intricate process of estimating the Lifetime Value (LTV) of customers from diverse geographical markets, harnessing the capabilities of both established and cutting-edge machine learning techniques. The investigation began with linear regression models which laid the groundwork by identifying key factors influencing revenue. However, it was through the application of sophisticated algorithms, specifically Random Forest and Gradient Boosting, that we gleaned more profound insights. These advanced models demonstrated their superiority in handling complex, non-linear data and unveiled the pivotal influence of user engagement time on revenue generation.

Our methodical feature importance analysis has showed the significant drivers behind revenue, leading to tailored, country-specific revenue forecasts and LTV predictions. The results have provided strategic directives for targeted marketing efforts and informed allocation of resources, illustrating the transformative impact machine learning holds within the sphere of predictive analytics.

By laying out a structured framework for the ongoing refinement of predictive models, this study contributes to the dynamic field of customer value estimation.

It suggests a path forward for future investigations to build upon the models and findings presented, adapting and evolving with the ever-changing patterns of customer behavior and market conditions. This venture into the predictive analytics landscape not only enhances our current understanding but also charts a course for future innovations in strategic business analysis.

REFERENCES

[1] Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32.

[2] Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189-1232.

[3] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning. *Springer*.