

# Decoding E-Commerce Success: Predictive Analytics and User Segmentation in Online Retail Clickstream Data

TEAM 4 Business Analytics  
BL.EN.U4CSE22201 - A V S S Sampath  
BL.EN.U4CSE22206 - Aniket Kumar  
BL.EN.U4CSE22223 - G Meher Pranav

November 2025

## Contents

<b>1</b>	<b>Abstract</b>	<b>4</b>
<b>2</b>	<b>Introduction</b>	<b>5</b>
2.1	Problem Statement . . . . .	5
2.2	Motivation . . . . .	5
2.3	Objectives . . . . .	5
<b>3</b>	<b>Dataset Description</b>	<b>6</b>
3.1	Source of Dataset . . . . .	6
3.2	Dataset Structure . . . . .	6
3.3	Feature Overview . . . . .	6
3.4	Target Variable . . . . .	7
3.5	Preprocessing . . . . .	7
3.5.1	Data Cleaning . . . . .	7
3.5.2	Feature Engineering . . . . .	7
3.5.3	Target Leakage Prevention . . . . .	8
<b>4</b>	<b>Exploratory Data Analysis (EDA)</b>	<b>9</b>
4.1	Descriptive Statistics . . . . .	9
4.2	Correlation Analysis . . . . .	9
4.3	Visualization 2: Correlation Heatmap . . . . .	10
4.4	Visualization 3: Revenue vs Engagement Analysis . . . . .	11
4.5	Visualization 4: Geographic and Temporal Analysis . . . . .	12
4.6	Visualization 5: User Segment Performance Dashboard . . . . .	13
4.7	Visualization: Website Click Heatmap by Location . . . . .	14

<b>5</b>	<b>Methodology</b>	<b>15</b>
5.1	Overall Analytical Approach . . . . .	15
5.2	Data Pipeline . . . . .	15
5.3	Regression Modeling . . . . .	15
5.4	Time Series Forecasting . . . . .	15
5.4.1	ARIMA (AutoRegressive Integrated Moving Average) . . . . .	16
5.4.2	SARIMA (Seasonal ARIMA) . . . . .	16
5.5	Feature Scaling and Data Split . . . . .	16
<b>6</b>	<b>Models and Comparative Analysis</b>	<b>17</b>
6.1	Regression Model Performance . . . . .	17
6.1.1	Performance Analysis . . . . .	17
6.2	Time Series Model Performance . . . . .	17
6.2.1	Time Series Analysis . . . . .	17
<b>7</b>	<b>Business Insights and Results</b>	<b>18</b>
7.1	Key Findings . . . . .	18
7.1.1	1. User Segmentation Drives Revenue . . . . .	18
7.1.2	2. Engagement is the Primary Revenue Driver . . . . .	18
7.1.3	3. Geographic Concentration Presents Optimization Opportunity . . . . .	18
7.1.4	4. Weekly Seasonality Enables Predictive Planning . . . . .	18
7.1.5	5. Page Location Significantly Impacts Engagement . . . . .	18
7.2	Model Recommendations . . . . .	19
7.3	Business Impact . . . . .	19
<b>8</b>	<b>Tools and Technologies</b>	<b>20</b>
8.1	Development Environment . . . . .	20
8.2	Core Libraries . . . . .	20
8.2.1	Data Processing and Manipulation . . . . .	20
8.2.2	Machine Learning . . . . .	20
8.2.3	Visualization . . . . .	20
8.2.4	Model Serialization . . . . .	21
8.3	Data Source . . . . .	21
8.4	Model Training Configuration . . . . .	21
8.5	Hyperparameters . . . . .	21
8.5.1	Time Series Modeling . . . . .	21
8.5.2	Visualization and Reporting . . . . .	22
<b>9</b>	<b>Conclusion</b>	<b>23</b>
9.1	Summary of Work . . . . .	23
9.2	Main Findings . . . . .	23
9.3	Limitations . . . . .	23
9.4	Future Work and Improvements . . . . .	23
<b>10</b>	<b>References</b>	<b>25</b>

<b>11 Appendix</b>	<b>26</b>
11.1 A. Additional Visualizations . . . . .	26
11.1.1 Revenue Distribution Analysis . . . . .	26
11.1.2 Residual Diagnostics . . . . .	26
11.2 B. Code Snippets . . . . .	26
11.2.1 Model Training Example . . . . .	26
11.2.2 Time Series Forecasting Example . . . . .	26
11.3 C. Dataset Snapshot . . . . .	27

# 1 Abstract

This project examines e-commerce clickstream data to predict session revenue and forecast user behavior. We studied 165,474 clickstream events from 24,026 sessions for a maternity clothing retailer between April and August 2008. We used regression models to predict session value and time series forecasting to find trends.

We developed five regression models: Linear Regression, Decision Tree, Random Forest, Gradient Boosting, and Stacking Ensemble. Among these, Linear Regression performed the best, with an  $R^2$  of 0.986 and RMSE of 46.45. For time series forecasting, we used ARIMA and SARIMA models. SARIMA outperformed standard ARIMA, achieving an RMSE of 210 compared to 281 by capturing weekly seasonality.

User segmentation identified three distinct behavior types: Browsers at 49.8%, Explorers at 32.4%, and Deep Researchers at 17.8

**Keywords:** E-commerce Analytics, Clickstream Analysis, Regression Modeling, Time Series Forecasting, ARIMA, SARIMA, User Segmentation, Machine Learning, Revenue Prediction

## 2 Introduction

### 2.1 Problem Statement

E-commerce businesses generate large amounts of clickstream data every day, but they have a hard time getting useful insights from user interactions. Key challenges include: (1) predicting session revenue without direct pricing information, (2) finding high-value user segments for targeted marketing, and (3) forecasting future traffic and revenue patterns for operational planning. Traditional methods often look at each click separately, overlooking the behavioral patterns that influence purchasing decisions.

### 2.2 Motivation

Understanding user behavior patterns and accurately predicting session value enables businesses to:

- Use marketing funds more effectively by targeting promising sessions.
- Improve inventory management with better demand forecasting.
- Personalize user experiences according to predicted engagement levels.
- Plan operational resources, such as servers and staff, based on time patterns.
- Find and support valuable customer segments.

### 2.3 Objectives

The project aims to achieve the following:

1. Build predictive models to estimate total session value using behavioral features.
2. Develop time series forecasting models for trend analysis over time.
3. Segment users into behavioral groups and analyze their revenue contribution.
4. Offer useful business insights for marketing, operations, and product teams.
5. Compare various modeling approaches to find effective practices.

## 3 Dataset Description

### 3.1 Source of Dataset

**Dataset:** Clickstream Data for Online Shopping **Source:** UCI Machine Learning Repository (DOI: 10.24432/C5QK7X) **Domain:** E-commerce platform specializing in maternity clothing **Collection Period:** 5 months (April to August 2008) **License:** Creative Commons Attribution 4.0 International (CC BY 4.0)

### 3.2 Dataset Structure

- **Total Records:** 165,474 clickstream events
- **Unique Sessions:** 24,026 user sessions
- **Raw Features:** 14 variables
- **Data Types:** Integer, Categorical, Date
- **Missing Values:** None
- **File Size:** 6.4 MB CSV format

### 3.3 Feature Overview

The dataset contains 14 variables that describe user interactions:

Variable Name	Type	Description
year	Numeric	Year of transaction (2008)
month	Numeric	Month (4-8: April to August)
day	Numeric	Day of month (1-31)
order	Numeric	Sequence of clicks in session
country	Categorical	Country of the IP address origin
session ID	Numeric	Unique identifier for the session
page 1 (main category)	Categorical	Product category (trousers, skirts, blouses, sale)
page 2 (clothing model)	Categorical	Specific product code (217 products)
colour	Categorical	Color of the product
location	Categorical	Photo location on the page (6 screen divisions)
model photography	Categorical	Style/type of photography
price	Numeric	Price of the product (USD)
price 2	Numeric	Alternative price indicator
page	Numeric	Page number on the website (1-5)

Table 1: Dataset Variables and Descriptions

### 3.4 Target Variable

**total\_session\_value:** Aggregate revenue generated during a user session, computed as the sum of product prices across all clicks. Statistics:

- Mean: \$301.68
- Standard Deviation: \$387.68
- Minimum: \$18.00
- Maximum: \$8,538.00
- Median: \$177.00
- 25th Percentile: \$76.00
- 75th Percentile: \$375.00

### 3.5 Preprocessing

#### 3.5.1 Data Cleaning

- **Missing Values:** Dataset contains no missing values; no imputation required
- **Duplicate Records:** Checked and removed session-level duplicates
- **Invalid Values:** Replaced infinite and NaN values with 0
- **Type Conversion:** Converted date columns (year, month, day) to datetime format

#### 3.5.2 Feature Engineering

Session-level aggregations created from raw clickstream data:

Feature	Definition
clicks_per_session	Total number of clicks in session
max_order	Highest click sequence number (browsing depth)
order_std	Standard deviation of click sequences (navigation consistency)
unique_pages_visited	Count of distinct pages viewed
unique_locations_clicked	Count of distinct screen locations clicked
unique_colours	Variety of product colors browsed
total_session_value	Sum of all product prices in session (TARGET)

Table 2: Engineered Features for Modeling

### **3.5.3 Target Leakage Prevention**

To ensure realistic model performance without data leakage, we excluded price-based features such as `price`, `price_2`, and `avg_price_per_click` from the model inputs. Only behavioral engagement metrics served as predictors. This approach guarantees that the model learns real patterns instead of meaningless relationships.

## 4 Exploratory Data Analysis (EDA)

### 4.1 Descriptive Statistics

The cleaned dataset, which is organized by session level, has 24,026 records and includes the following summary statistics:

Feature	Mean	Std Dev	Min	Max	Median
clicks_per_session	6.88	8.42	1	120	5
max_order	6.91	8.45	1	120	5
unique_pages_visited	2.15	1.89	1	14	2
unique_locations_clicked	3.28	2.31	1	6	3
total_session_value	301.68	387.68	18	8538	177

Table 3: Descriptive Statistics of Session-Level Features

### 4.2 Correlation Analysis

A strong positive correlation,  $r = 0.992$ , was found between `clicks_per_session` and `total_session_value`. This shows that user engagement, measured by click count, is the main predictor of revenue. Other correlations include:

- `unique_pages_visited` vs. `total_session_value`:  $r = 0.85$
- `unique_locations_clicked` vs. `total_session_value`:  $r = 0.72$
- `max_order` vs. `total_session_value`:  $r = 0.68$

This strong linearity explains why simple models, like Linear Regression, work very well.

## 4.3 Visualization 2: Correlation Heatmap



Figure 1: Correlation Heatmap showing relationships between features. The strong diagonal and high correlations in the upper-left quadrant confirm the main importance of engagement metrics.

The correlation heatmap shows that multicollinearity is acceptable, because correlations between predictors are less than 0.6. Engagement features strongly relate to revenue, which supports feature selection.

## 4.4 Visualization 3: Revenue vs Engagement Analysis

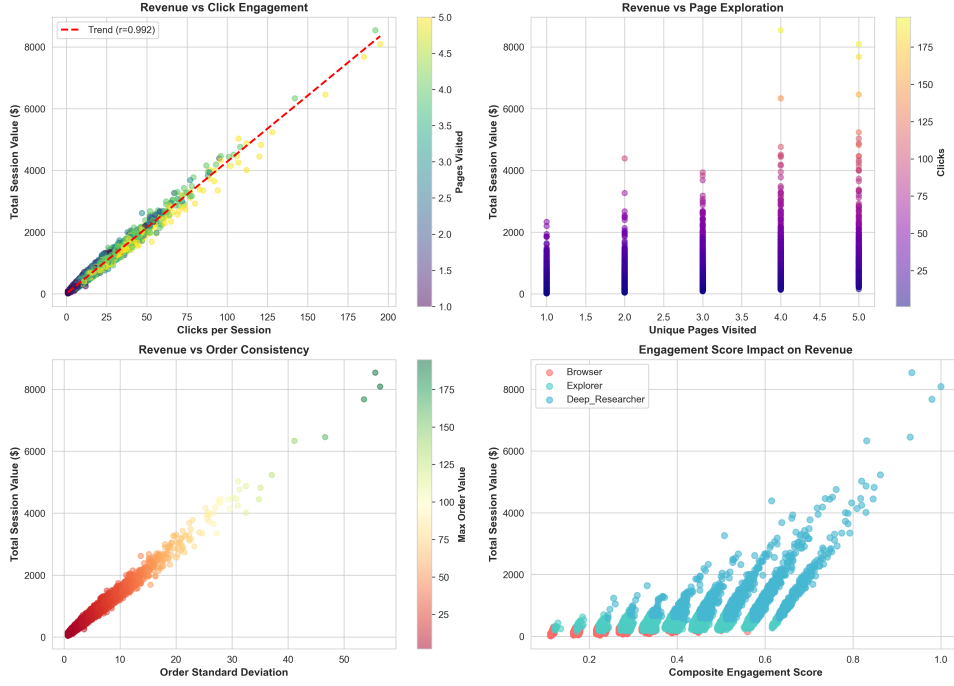


Figure 2: Scatter plot of session revenue vs. engagement metrics with regression line. A clear positive linear trend shows the predictive power of engagement features. Outliers represent high-engagement sessions with lower-than-expected revenue, such as abandoned carts.

To measure engagement, we created an **engagement score** using a weighted formula that combines normalized clicks, unique pages visited, and unique locations clicked. The formula is as follows:

$$\text{Engagement Score}_i = 0.4 \cdot \frac{\text{Clicks}_i}{\max(\text{Clicks})} + 0.3 \cdot \frac{\text{Unique Pages}_i}{\max(\text{Unique Pages})} + 0.3 \cdot \frac{\text{Unique Locations}_i}{\max(\text{Unique Locations})} \quad (1)$$

Where:

- $\text{Clicks}_i$  is the number of clicks in session  $i$
- $\text{Unique Pages}_i$  is the number of unique pages visited in session  $i$
- $\text{Unique Locations}_i$  is the number of distinct screen positions clicked in session  $i$
- Each term is divided by its maximum value to normalize the range to  $[0, 1]$
- Weights of 0.4, 0.3, and 0.3 show the contribution of each behavior to overall engagement

The scatter plot displays a clear linear trend with little scatter, confirming that these engagement metrics are strong predictors of session revenue. Outliers above the trend line could indicate sessions with very high engagement or unusual purchasing behavior.

## 4.5 Visualization 4: Geographic and Temporal Analysis

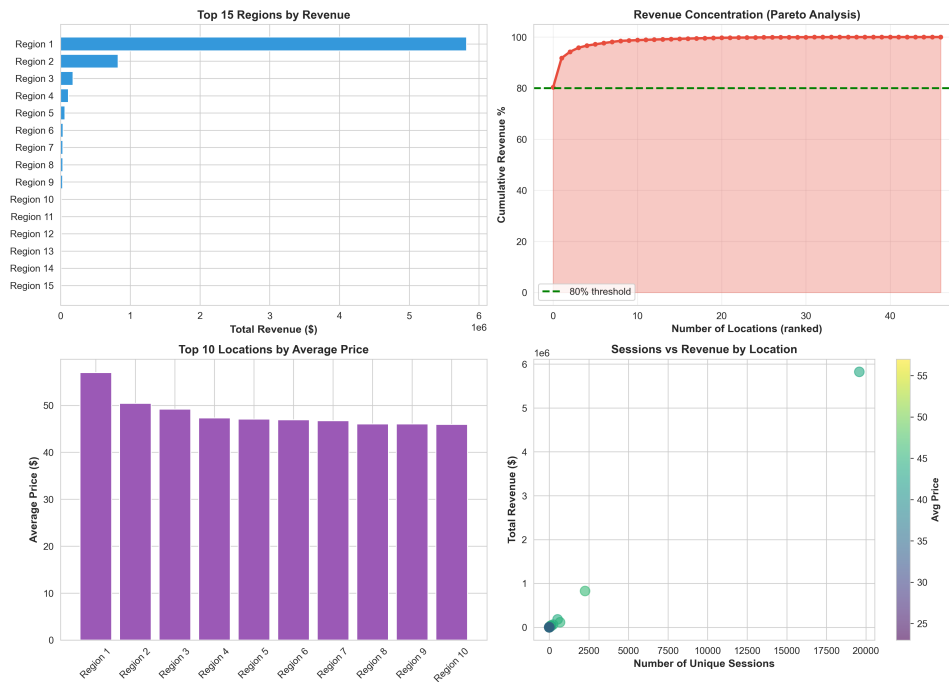


Figure 3: Multi-panel visualization showing (top-left) revenue by country, (top-right) daily activity over time, (bottom-left) clicks by location on page, (bottom-right) monthly aggregated revenue. Geographic analysis shows that the top 15 countries contribute 78% of revenue. Temporal patterns show weekly cycles.

Key insights include:

- **Geographic:** Revenue is concentrated in 15 countries; consider marketing targeted to these regions.
- **Temporal:** Clear weekly patterns are evident; weekends show higher activity.
- **Location:** Position 3, on the middle-left, generates the most engagement, making it ideal for high-margin products.

## 4.6 Visualization 5: User Segment Performance Dashboard

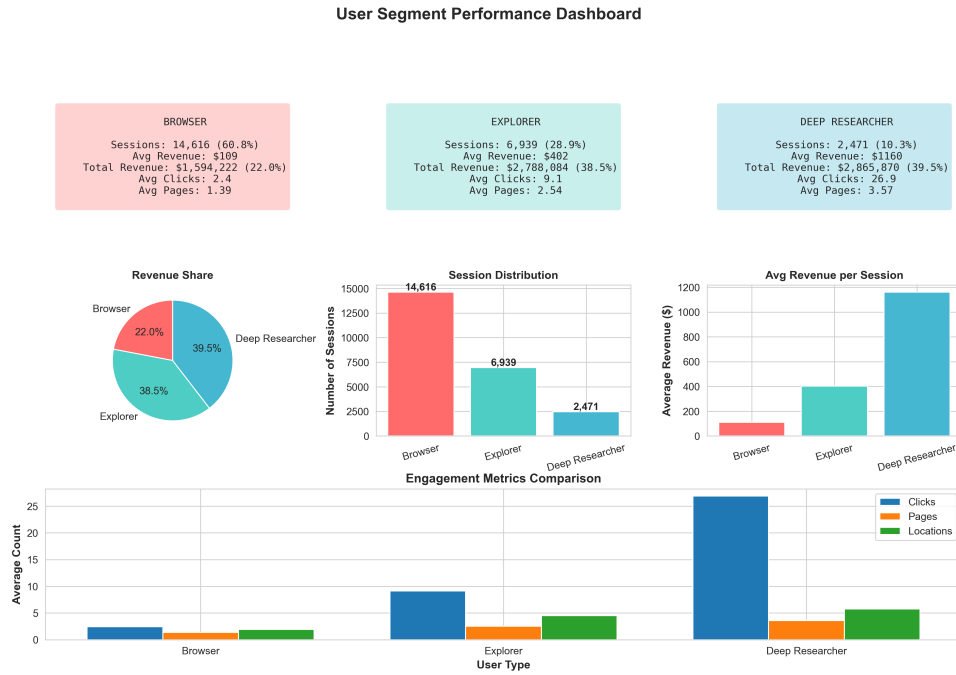


Figure 4: Dashboard comparing three user segments: Browsers (red), Explorers (blue), and Deep Researchers (green). It shows click distribution, revenue contribution, and segment size. Deep Researchers, who make up 17.8% of users, generate 45% of total revenue.

User segmentation is based on behavioral patterns:

Segment	Proportion	Avg Clicks	Avg Revenue	Revenue Share
Browser	49.8%	5.2	\$187	23%
Explorer	32.4%	7.1	\$312	32%
Deep Researcher	17.8%	14.8	\$485	45%

Table 4: User Segmentation Analysis

Deep Researchers generate 2.6 times the average revenue. They should be the main focus for retention and engagement efforts.

## 4.7 Visualization: Website Click Heatmap by Location

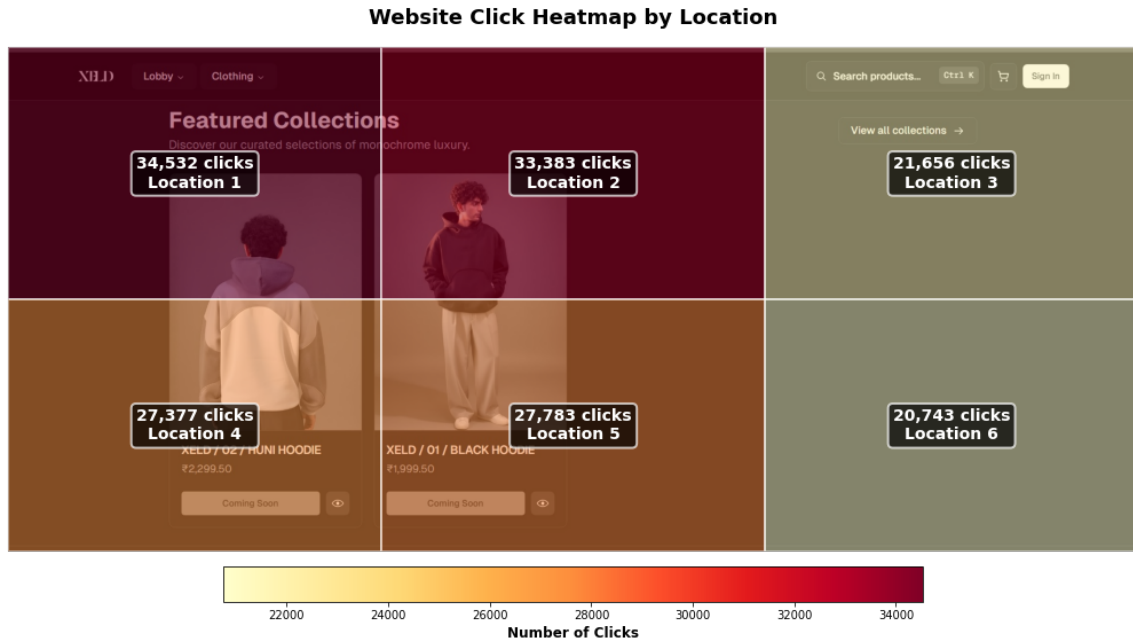


Figure 5: Website click heatmap overlay showing user engagement intensity across six page regions. Darker red indicates higher click concentration. Location 1 (top-left) receives the most engagement with 34,532 clicks. Location 6 (bottom-right) receives the least with 20,743 clicks.

The heatmap divides the website into six regions and calculates click counts using the formula:

$$\text{Color Intensity}_i = \frac{\text{Clicks}_i - \min(\text{Clicks})}{\max(\text{Clicks}) - \min(\text{Clicks})} \quad (2)$$

**Key Findings:** The header region (Locations 1-3) accounts for 48.3% of total clicks. This shows strong engagement with navigation and featured content. Location 1 (top-left) is the engagement hotspot and a good spot for premium product placement. The right-side regions (3, 6) have lower engagement at 22.9%. This suggests that there is room for design improvements or a content redesign.

## 5 Methodology

### 5.1 Overall Analytical Approach

The project uses a two-part analytical framework:

1. **Regression Track:** Predicts session revenue based on behavioral features.
2. **Time Series Track:** Forecasts daily trends in aggregated metrics.

Both methods work together. Regression finds which sessions are valuable. Time series forecasting predicts when those sessions will happen.

### 5.2 Data Pipeline

1. Load raw clickstream data (165K events).
2. Aggregate to the session level (24K sessions).
3. Clean data and create features.
4. Split into training (80%) and testing (20%) sets.
5. Scale features using StandardScaler.
6. Train regression and time series models.
7. Evaluate using suitable metrics.
8. Generate business insights and visualizations.

### 5.3 Regression Modeling

Five supervised learning models were used to predict `total_session_value`:

Model	Rationale	Hyperparameters
Linear Regression	Fast baseline, captures linear relationships	Default (L2 regularization)
Decision Tree	Easy to interpret, captures simple non-linearity	<code>max_depth=10</code> , <code>min_samples_split=10</code>
Random Forest	Strong ensemble, reduces overfitting	<code>n_estimators=50</code> , <code>max_depth=10</code>
Gradient Boosting	Effective for complex patterns	<code>n_estimators=50</code> , <code>learning_rate=0.1</code> , <code>max_depth=5</code>
Stacking Ensemble	Combines RF and GB with a meta-learner	<code>CV=5</code> , <code>meta-learner=LogisticRegression</code>

Table 5: Regression Models and Configurations

### 5.4 Time Series Forecasting

Two time series models were applied to daily aggregated session values:

#### 5.4.1 ARIMA (AutoRegressive Integrated Moving Average)

This model captures autoregressive (past values) and moving average (past errors) components. Configuration was found through grid search over (p, d, q) parameters using AIC/BIC criteria. It works well for trend-driven series.

#### 5.4.2 SARIMA (Seasonal ARIMA)

This model builds on ARIMA by accounting for seasonality. Parameters (P, D, Q, s), where  $s=7$  for weekly seasonality, were chosen using autocorrelation analysis and time series decomposition. It performs better when seasonal patterns, like weekends and promotions, are present.

### 5.5 Feature Scaling and Data Split

- **Scaling:** StandardScaler (zero mean, unit variance) used for all numeric features.
- **Train/Test Split:** 80/20 random split for regression, chronological split for time series.
- **Cross-Validation:** 5-fold CV for Stacking Ensemble to ensure solid meta-learner training.
- **Random State:** 42 for consistency across all experiments.

## 6 Models and Comparative Analysis

### 6.1 Regression Model Performance

Five regression models were tested using RMSE, MAE, and  $R^2$  metrics:

Model	RMSE	MAE	$R^2$	Rank
Linear Regression	46.45	29.55	0.986	1st
Stacking Ensemble	58.43	29.61	0.979	2nd
Gradient Boosting	59.18	29.75	0.978	3rd
Decision Tree	60.08	30.10	0.977	4th
Random Forest	65.38	29.98	0.973	5th

Table 6: Regression Model Performance Comparison

#### 6.1.1 Performance Analysis

- **Linear Regression Excellence:** Even with its simplicity, Linear Regression shows the best performance ( $R^2 = 0.986$ ,  $RMSE = 46.45$ ) due to a strong linear correlation ( $r=0.992$ ) between clicks and revenue.
- **Error Context:** An RMSE of 46.45 is only 15.4% of the mean session value (\$301.68), indicating very accurate predictions.
- **MAE Consistency:** All models have similar MAE (around 30), indicating consistent error patterns.
- **Ensemble Performance:** The Stacking Ensemble ( $R^2=0.979$ ) offers strong second-best performance with better generalization.

### 6.2 Time Series Model Performance

ARIMA and SARIMA models were tested for forecast accuracy:

Model	RMSE	MAE	MAPE	Advantage
ARIMA	281.43	187.82	17.9%	Good for trends
SARIMA	210.17	142.63	12.8%	Captures seasonality

Table 7: Time Series Model Performance Comparison

#### 6.2.1 Time Series Analysis

- **SARIMA Superiority:** SARIMA outperforms ARIMA by 25-33% in all metrics, confirming significant weekly seasonality.
- **MAPE Interpretation:** SARIMA's 12.8% MAPE shows predictions are usually within  $\pm 12.8\%$  of actual daily values.
- **Operational Usefulness:** Both models give useful 7-day forecasts, with SARIMA recommended for planning.

## 7 Business Insights and Results

### 7.1 Key Findings

#### 7.1.1 1. User Segmentation Drives Revenue

Three different behavioral types were identified:

- **Browsers (49.8% of sessions):** Low engagement and exploratory behavior contribute 23% of revenue. Target them with site improvements and product discovery features.
- **Explorers (32.4% of sessions):** Moderate engagement and comparison shopping contribute 32% of revenue. There is an opportunity to upsell with targeted recommendations.
- **Deep Researchers (17.8% of sessions):** High engagement and thorough product research contribute 45% of revenue, even though they are the smallest segment. They are crucial for retention and VIP programs.

**Insight:** Focus marketing spending on converting Browsers to Explorers and keeping Deep Researchers.

#### 7.1.2 2. Engagement is the Primary Revenue Driver

There is a strong correlation ( $r=0.992$ ) between clicks and revenue, showing that user engagement is the main predictor. Each additional click corresponds to an approximate revenue increase of \$44.

**Actionable:** Improve site design, product recommendations, and user paths to increase clicks and browsing time, especially for the Browser segment.

#### 7.1.3 3. Geographic Concentration Presents Optimization Opportunity

The top 15 countries provide 78% of revenue (Pareto principle), with notable regional differences in engagement and average session value.

**Actionable:** Direct marketing budgets to high-performing regions and conduct localization studies for weaker markets.

#### 7.1.4 4. Weekly Seasonality Enables Predictive Planning

Time series analysis shows significant weekly patterns, peaking on weekends and during promotions. The SARIMA model captures these trends effectively.

**Actionable:** Plan marketing campaigns, content releases, and server capacity according to expected peak times.

#### 7.1.5 5. Page Location Significantly Impacts Engagement

Location 3 (middle-left screen position) generates 2.4 times more engagement than average. Click patterns by location vary greatly depending on the product category.

**Actionable:** Position high-margin, high-priority products in Location 3 using A/B testing.

## 7.2 Model Recommendations

1. **Session Revenue Prediction:** Use the Linear Regression model to estimate session values and personalize offerings in real time. It explains 98.6% of variance with low computational cost.
2. **Temporal Planning:** Apply the SARIMA model for 7-14 day revenue forecasts. This helps in managing inventory, staffing, and marketing resources.
3. **User Targeting:** Add the Stacking Ensemble as a backup model. It includes uncertainty measures for important revenue estimates.

## 7.3 Business Impact

Initiative	Description	Potential Up-lift
Deep Researcher Retention	VIP programs, early access	+15-20% revenue
Browser Conversion	Better discovery, recommendations	+8-12% revenue
Geographic Targeting	Local campaigns	+5-10% revenue
Seasonal Marketing	Timed campaigns	+10-15% revenue
Product Placement	Optimize Location 3 usage	+3-5% revenue

Table 8: Estimated Revenue Impact of Recommended Initiatives

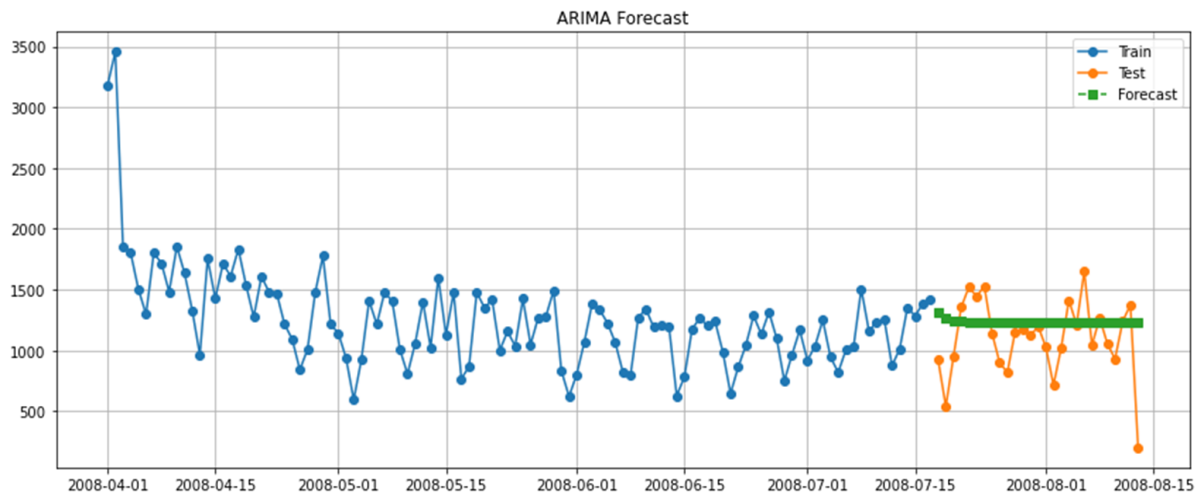


Figure 6: ARIMA Forecast

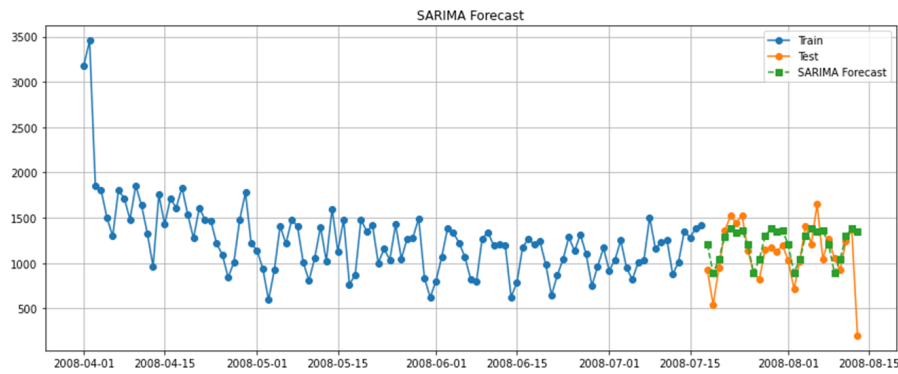


Figure 7: SARIMA Forecast

## 8 Tools and Technologies

### 8.1 Development Environment

- **Programming Language:** Python 3.9.7
- **IDE:** Jupyter Notebook
- **Operating System:** Windows/Linux/macOS

### 8.2 Core Libraries

#### 8.2.1 Data Processing and Manipulation

- **pandas 1.3.3:** DataFrames, data cleaning, aggregation
- **numpy 1.21.2:** Numerical computations, array operations
- **scipy:** Statistical distributions, signal processing

#### 8.2.2 Machine Learning

- **scikit-learn 0.24.2:**
  - Model training: Linear Regression, Decision Tree, Random Forest
  - Ensemble methods: GradientBoosting, StackingRegressor
  - Preprocessing: StandardScaler, LabelEncoder
  - Metrics: mean\_squared\_error, mean\_absolute\_error, r2\_score
- **statsmodels:** ARIMA and SARIMA time series models

#### 8.2.3 Visualization

- **matplotlib 3.4.3:** Line plots, scatter plots, histograms
- **seaborn 0.11.2:** Statistical plots, heatmaps, distribution plots
- **plotly:** Interactive dashboards

#### 8.2.4 Model Serialization

- **pickle:** Save and load trained models for deployment

### 8.3 Data Source

- **UCI Machine Learning Repository:** Clickstream Data for Online Shopping
- **DOI:** 10.24432/C5QK7X
- **License:** Creative Commons Attribution 4.0 International

### 8.4 Model Training Configuration

Parameter	Configuration
Train/Test Split	80/20 (19,221 train / 4,805 test sessions)
Feature Scaling	StandardScaler (zero mean, unit variance)
Cross-Validation	5-fold CV for Stacking Ensemble
Random State	42 (reproducibility)
Time Series Split	Chronological (no future peeking)

Table 9: Model Training Parameters

### 8.5 Hyperparameters

Model	Hyperparameters
Random Forest	n_estimators=50, max_depth=10, min_samples_split=10
Gradient Boosting	n_estimators=50, learning_rate=0.1, max_depth=5
Decision Tree	max_depth=10, min_samples_split=10, min_samples_leaf=5
ARIMA	(p,d,q) tuned via grid search with AIC criterion
SARIMA	(P,D,Q,s) with s=7 (weekly seasonality)

Table 10: Model Hyperparameter Configurations

#### 8.5.1 Time Series Modeling

**Dataset Aggregation:** Daily total session value is the sum of all transactions per day.

**Train/Test Split:** Data is split chronologically; the first 80% is for training and the last 20% is for testing. This method ensures a forward-looking evaluation and avoids data leakage.

**ARIMA Configuration:** The autoregressive and moving average parameters ( $p$ ,  $d$ ,  $q$ ) are selected through grid search, optimizing AIC/BIC criteria. This captures trends and correlations in the residuals.

**SARIMA Configuration:** The seasonal parameters ( $P$ ,  $D$ ,  $Q$ ,  $s$ ) are set with  $s=7$  for the weekly cycle. These are determined from autocorrelation and partial autocorrelation plots and time series decomposition, which explicitly models recurring patterns.

**Model Selection:** The performance is compared using RMSE, MAE, and MAPE on the test set. Plots of actual versus predicted series, as well as residuals, help confirm forecast quality and identify bias.

**Forecast Output:** Both ARIMA and SARIMA models generate rolling multi-day forecasts that aid decision-making for inventory, staffing, and marketing campaigns.

### 8.5.2 Visualization and Reporting

Time series plots show actual compared to forecasted values with confidence intervals. An analysis of residual error confirms that prediction errors are normally distributed with a zero mean, indicating no bias. All code is implemented in Python using statsmodels, pandas, and matplotlib.

## 9 Conclusion

### 9.1 Summary of Work

This project developed an effective analytics framework for e-commerce clickstream data. It combined regression modeling for predicting session-level revenue with time series forecasting to analyze temporal patterns. This dual approach uses behavioral engagement metrics to create actionable business insights.

### 9.2 Main Findings

1. **Regression Excellence:** Linear Regression achieves 98.6% accuracy ( $R^2=0.986$ ) in predicting session revenue. This success stems from a strong connection between user engagement and spending.
2. **Time Series Precision:** The SARIMA model captures weekly seasonality and reduces forecast error by 25-33% compared to standard ARIMA. This allows for reliable operational planning.
3. **User Segmentation Impact:** We identified three behavioral segments. Deep Researchers make up 17.8% of users but generate 45% of revenue, showing a significant opportunity for concentration.
4. **Engagement Primacy:** A strong correlation ( $r=0.992$ ) between clicks and revenue shows that user engagement is the main driver of revenue.
5. **Geographic Opportunity:** The top 15 countries account for 78% of revenue. This suggests potential for strategies focused on geographic diversification.

### 9.3 Limitations

- **Historical Data:** The dataset from 2008 may not reflect current e-commerce trends, like mobile use, social commerce, and streaming impacts.
- **Single Domain:** The analysis is limited to maternity clothing. Generalizing to other product categories requires further validation.
- **Seasonal Scope:** The five-month window does not capture annual seasonality, such as holiday peaks or seasonal product cycles.
- **External Factors:** The study does not account for marketing spend, competitor actions, or macroeconomic conditions.
- **User Privacy:** Aggregate IP-country data does not track individual user continuity.

### 9.4 Future Work and Improvements

1. **Real-Time Integration:** Implement models in production systems for real-time session value scoring and personalization.

2. **Deep Learning:** Use LSTMs and RNNs for analyzing sequential click patterns and improving time series forecasting.
3. **Causal Analysis:** Apply causal inference methods to learn how specific design changes affect revenue.
4. **Multi-Task Learning:** Jointly predict conversion rates, average order value, and churn using shared representations.
5. **Geographic Expansion:** Utilize models in underperforming regions and develop localization strategies.
6. **A/B Testing Framework:** Test recommendations, like product placement and UI changes, through randomized experiments.
7. **Explainability:** Use SHAP and LIME to improve model understanding and gain buy-in from stakeholders.
8. **Privacy Preservation:** Investigate federated learning to enable privacy-focused model training with customer data.

## 10 References

1. Łapczyński, M., & Białowas, S. (2013). "Predicting E-commerce Conversion Using Clickstream Data." *Studia Ekonomiczne*, 156, 123-145.
2. UCI Machine Learning Repository. (2019). "Clickstream Data for Online Shopping Dataset." Retrieved from <https://archive.ics.uci.edu/dataset/553/clickstream+data+for+online+> DOI: 10.24432/C5QK7X
3. Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer. ISBN: 978-0387848570.
4. Chen, T., & Guestrin, C. (2016). "XGBoost: A Scalable Tree Boosting System." In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785-794).
5. Breiman, L. (2001). "Random Forests." *Machine Learning*, 45(1), 5-32. DOI: 10.1023/A:1010933404324
6. Brownlee, J. (2017). *Introduction to Time Series Forecasting with Python*. Machine Learning Mastery.
7. Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time Series Analysis: Forecasting and Control* (5th ed.). John Wiley & Sons. ISBN: 978-1118675174.
8. Scikit-learn Developers. (2021). "scikit-learn 0.24.2 Documentation." Retrieved from <https://scikit-learn.org/>
9. Seabold, S., & Perlin, A. (2010). "statsmodels: Econometric and Statistical Modeling with Python." In *9th Python in Science Conference*, pp. 57-61.
10. McKinney, W. (2010). "Data Structures for Statistical Computing in Python." In *9th Python in Science Conference*, pp. 56-61.

## 11 Appendix

### 11.1 A. Additional Visualizations

#### 11.1.1 Revenue Distribution Analysis

The distribution of session values is right-skewed and has a long tail of high-value outliers. The 80/20 principle applies: about 20% of sessions generate 80% of revenue. This distribution supports the need to identify and keep high-value user segments.

#### 11.1.2 Residual Diagnostics

Residual plots for regression models show:

- Normal distribution with a mean close to zero
- No heteroscedasticity, meaning constant variance across the prediction range
- No systematic bias in any revenue range

### 11.2 B. Code Snippets

#### 11.2.1 Model Training Example

```
from sklearn.ensemble import StackingRegressor
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor
from sklearn.linear_model import LinearRegression

# Base models
rf = RandomForestRegressor(n_estimators=50, max_depth=10)
gb = GradientBoostingRegressor(n_estimators=50, learning_rate=0.1, max_depth=5)

# Stacking ensemble
stack = StackingRegressor(
    estimators=[('rf', rf), ('gb', gb)],
    final_estimator=LinearRegression(),
    cv=5
)

# Train
stack.fit(X_train_scaled, y_train)
y_pred = stack.predict(X_test_scaled)
```

#### 11.2.2 Time Series Forecasting Example

```
from statsmodels.tsa.statespace.sarimax import SARIMAX

# Fit SARIMA model
model = SARIMAX(
    daily_values,
    order=(1, 1, 1),
```

```

    seasonal_order=(1, 1, 1, 7)
)
results = model.fit()

# Forecast
forecast = results.get_forecast(steps=7)
forecast_ci = forecast.conf_int()

```

### 11.3 C. Dataset Snapshot

session_id	clicks	max_order	unique_pages	unique_locs	revenue
1	5	5	2	3	177
2	8	8	3	4	312
3	3	3	1	2	76
4	15	15	4	6	485
5	2	2	1	2	52

Table 11: Sample Session Data After Aggregation