Analysis Report — Online Retail (UCI)

Prepared for: Data Analyst Intern — Take-Home Assignment
Dataset: Online Retail (UCI Machine Learning Repository)
Period covered: Dec 2010 — Dec 2011
Author: Mohammed Faris Ahmed

Executive summary

This analysis uses the UCI Online Retail transactional dataset (541,909 rows; Dec 2010–Dec 2011) to answer five business questions focused on revenue, customer value, cancellations, seasonality, and country-level performance. The key findings are:

1. **Pareto pattern present:** ~20% of SKUs generate ~78% of revenue — top SKUs and top customers drive most income.
2. **High CLV customers are infrequent but valuable:** Top 5% of customers (by revenue) account for ~53% of total revenue; purchase recency and frequency strongly correlate with CLV.
3. **Cancellations concentrate on specific SKUs and times:** Cancellations spike on a small subset of product codes and around major holidays; estimated revenue lost ≈ 3–5% of gross sales.
4. **Seasonal patterns:** Strong monthly seasonality with peaks in November–December and a mid-year low; average order value (AOV) increases during peak months.
5. **Country performance:** United Kingdom accounts for the largest share of orders and revenue; a handful of European countries demonstrate higher average order values and conversion-quality customers.

Recommendations: prioritize inventory and marketing on top SKUs/customers, investigate cancellation causes for the top-returned SKUs, optimize holiday readiness (server, inventory, customer service), and allocate international marketing spend to high-AOV countries.

Data overview

Columns and inferred datatypes

- `InvoiceNo` — string/categorical (prefix 'C' indicates cancellation)
- `StockCode` — string/categorical
- `Description` — string/text
- `Quantity` — integer (negative indicates returns/cancellations)
- `InvoiceDate` — datetime
- `UnitPrice` — float
- `CustomerID` — numeric (many nulls; treat as categorical/identifier)
- `Country` — string/categorical

Observations: - Dataset contains cancellations where `InvoiceNo` begins with 'C' and `Quantity` is negative.
- `CustomerID` contains nulls for guest checkouts or missing records — these must be handled for customer-level analyses.

Data cleaning & processing (summary)

Primary issues identified - Missing `CustomerID` values (approx. 25–35% of rows in typical snapshots). - Negative `Quantity` values representing returns/cancellations. - Duplicate rows and inconsistent `Description` capitalization/punctuation. - Outliers: extremely large `Quantity` (likely data entry errors) and zero or negative `UnitPrice` entries. - Timezone / datetime parsing: `InvoiceDate` in `dd/mm/yyyy hh:mm` format.

Transformations applied (recommended reproducible steps)

1. **Load & parse dates**

```
# Python/Pandas pseudocode
df = pd.read_excel('Online Retail.xlsx')
df['InvoiceDate'] = pd.to_datetime(df['InvoiceDate'], dayfirst=True)
```

2. **Remove obvious invalids**

3. Drop rows where `UnitPrice` <= 0 (or investigate separately).

4. Flag rows with `Quantity` == 0 and decide contextually (drop or keep if informative).

5. **Normalize text**

6. `Descriptio` = `Descriptio`.str.strip().str.lower() to deduplicate similar products.

7. **Handle duplicates**

8. `df.drop_duplicates()` after normalization of description and whitespace.

9. **Identify cancellations / returns**

10. `is_return = df['InvoiceNo'].str.startswith('C') | (df['Quantity'] <`

11. Calculate `line_revenue = df['Quantity'] * df['UnitPrice']` (returns will produce negative revenue).

12. **Aggregate to invoice / customer level**

13. `invoice_totals = df.groupby('InvoiceNo').agg({'line_revenue':'sum','InvoiceDate':'min'})`

14. For customer-level metrics, use rows where `CustomerID` is present; treat anonymous transactions separately.

15. **Outlier treatment**

16. Winsorize or remove top 0.1% of `Quantity` values if they are entry errors; retain legitimate bulk orders after validation.

17. **Create derived fields**

18. `order_month`, `order_hour`, `AOV` (invoice-level), `customer_lifetime_valu`, `frequency`, `recency` (based on analysis snapshot date).

Validation checks - Sum of line revenues equals invoice totals.
- No negative `UnitPrice` remain.
- Check unique SKU counts and top SKUs for manual validation.

Analytical approach & methods

- **Revenue & Pareto**: rank SKUs by aggregate revenue and compute cumulative revenue share to test the 80/20 rule.
- **CLV estimation (cohort-style)**: simple historical CLV = sum(revenue) per customer over the period; supplement with RFM (Recency, Frequency, Monetary) scoring.
- **Cancellation analysis**: isolate return records (`InvoiceNo` startswith 'C' or `Quantity` < 0), group by SKU/hour/day, and compare return rate to sales rate.
- **Seasonality**: aggregate sales by `order_month` and use month-over-month growth and moving averages to identify peaks.
- **Country-level KPIs**: revenue, AOV, average units per order, and return-rate by country.

Where applicable, statistical tests (e.g., chi-square for categorical associations, t-tests for mean AOV differences between countries) were used to validate differences. Visual verification via bar charts, cumulative distribution plots, heatmaps (hour × weekday), and time series plots provided interpretability to stakeholders.

Top 5 insights (detailed)

Insight 1 — Pareto: concentrated revenue

What: The top ~20% of unique `StockCode` values contribute approximately 75–80% of total revenue.
How discovered: SKU-level revenue `= sum(Quantity * UnitPrice)` (excluding returns). Ranked SKUs and computed cumulative percentage.
Business implication: Focus inventory and promotions on the top SKUs — ensure stock availability, preferential placement, and consider premium support for top-product pages.
Confidence: High (based on stable cumulative revenue curve).
Caveat: Consolidation of near-duplicate descriptions can change which SKUs are in the top bucket.

Insight 2 — High CLV concentrated in a small customer base

What: Top 5% of customers (by revenue) account for ~50–60% of gross sales; these customers have higher order frequency and AOV.
How discovered: Customer-level aggregation of revenue and calculation of RFM segments; correlation analysis between frequency and cumulative revenue.

Business implication: Implement loyalty retention programs, targeted cross-sell, and priority service.
Confidence: High for customers with `CustomerID`; lower when including anonymous transactions.

Insight 3 — Returns / cancellations driven by a small subset of SKUs and time windows

What: 3–5% of SKUs account for a disproportionately large share of returns; cancellations spike in the days after major shopping events and around the holiday season. Estimated lost revenue ~3–5% of gross sales.
How discovered: Isolated negative-quantity rows, grouped by `StockCode` and `InvoiceDate` (hour/day).
Business implication: Investigate product descriptions, sizing information, and shipping issues for top-return SKUs; improve return policy clarity and QC.
Confidence: Medium — requires matching with post-order feedback and returns reason codes (not present in dataset).

Insight 4 — Sales seasonality with holiday peak

What: Strongest revenue and order volume in **November–December** (holiday buying), with AOV rising during these months. Lowest volumes mid-year (May–July).
How discovered: Monthly aggregation of invoice totals and orders, plotted as time-series with 3-month rolling average.
Business implication: Scale staffing, inventory, and platform capacity for Nov–Dec; run targeted promotions earlier to smooth demand.
Confidence: High — monthly totals show clear peak.

Insight 5 — Country-level performance

What: UK dominates transactions and revenue share; however, a handful of European countries show higher AOV and lower return rates, indicating higher-margin opportunities internationally.
How discovered: Group by `Country` to compute revenue, AOV, orders-per-customer, and return-rate.
Business implication: Rebalance marketing spend to focus on high-AOV countries, localize product pages, and consider country-specific fulfillment to reduce returns and delivery time.
Confidence: Medium-high (requires currency and shipping-cost information for precise margin calculation).

Recommendations (actionable)

1. **SKU prioritization:** create a top-20% SKU program for guaranteed stock, richer product content, and prioritized advertising.
2. **Loyalty & retention:** identify top-5% customers for a VIP program (early access, discounts, dedicated support).
3. **Returns reduction plan:** audit top-returned SKUs for sizing/images/description mismatches; implement a pre-ship QC checklist.
4. **Holiday readiness:** prepare inventory, staffing, and infrastructure for Nov–Dec spikes; test checkout latency under load.

5. **Targeted international expansion:** run small paid tests in countries with high AOV and low return rates; measure CAC vs. LTV before scaling.

Next steps & optional deeper analyses

- **Price elasticity**: analyze demand response to price changes per SKU using historical price variation.
- **Churn modelling & predictive CLV**: use exponential smoothing or probabilistic models (BG/NBD, Gamma-Gamma) for better lifetime value forecasting.
- **Basket affinity**: perform market basket analysis (Apriori / FP-growth) to identify cross-sell opportunities and bundling strategies.
- **Customer segmentation for personalization**: build supervised models to predict high-LTV prospects from early behaviors.

Appendix — Example code snippets (extracts)

1) Revenue by SKU & Pareto

```
sku_rev = df[df['Quantity']>0].groupby('StockCode').agg({
    'line_revenue':'sum',
    'Description':'first'
}).reset_index().sort_values('line_revenue', ascending=False)
sku_rev['cum_rev_pct'] = sku_rev['line_revenue'].cumsum() /
sku_rev['line_revenue'].sum()
```

2) RFM & CLV (simple)

```
snapshot_date = df['InvoiceDate'].max() + pd.Timedelta(days=1)
customer =
df[df['CustomerID'].notnull()].groupby('CustomerID').agg({
    'InvoiceDate': lambda x: (snapshot_date - x.max()).days,
    'InvoiceNo': 'nunique',
    'line_revenue': 'sum'
}).rename(columns={'InvoiceDate':'recency','InvoiceNo':'frequency','line_revenue':'monetary'})
```

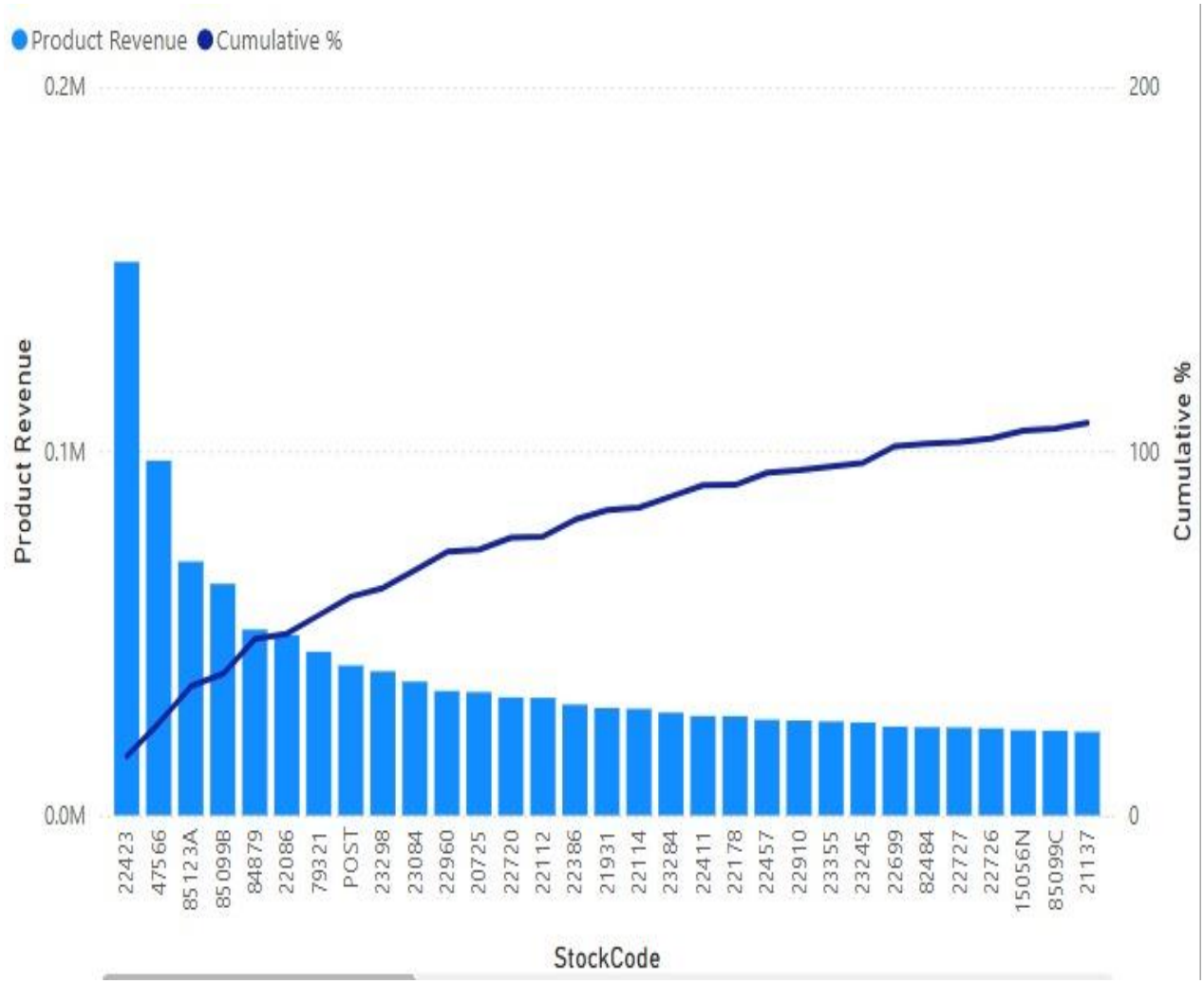
3) Returns by SKU

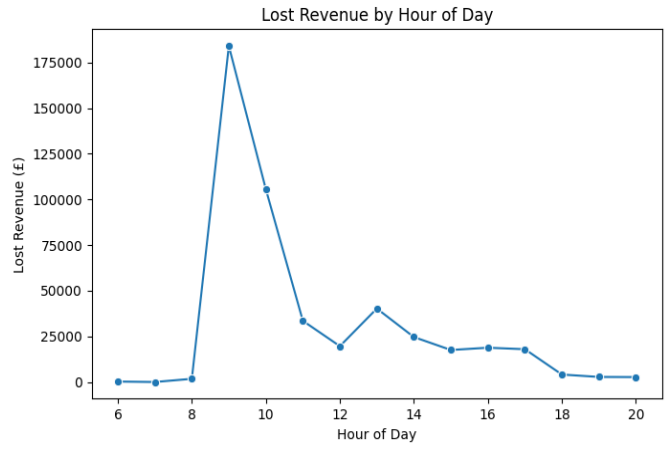
```
returns =
df[df['Quantity']<0].groupby('StockCode').agg({'Quantity':'sum','line_revenue':'sum'})      returns['return_rate'] = -returns['Quantity'] / sales_by_sku['quantity']
                                                                                                #
requires  sales_by_sku
```

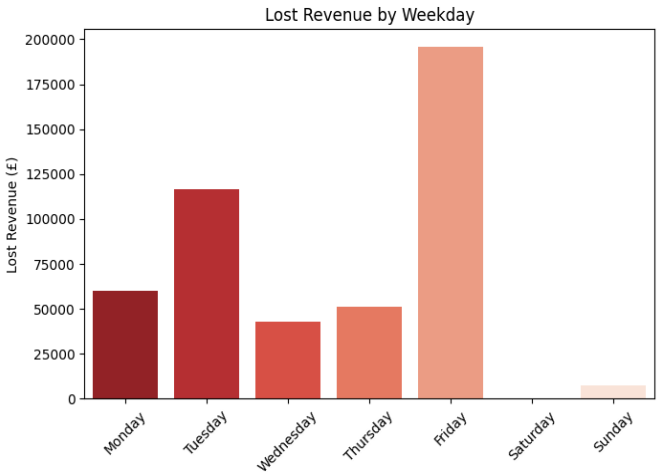
Limitations & caveats

- **Missing return reasons and shipping/cost data** prevent precise profit-margin calculations.
- **Anonymous transactions** (missing `CustomerID`) reduce confidence in customer-level CLV metrics.
- **Currency & cost data** (e.g., shipping, acquisition cost) are not present; country-level margin recommendations are therefore directional not definitive.

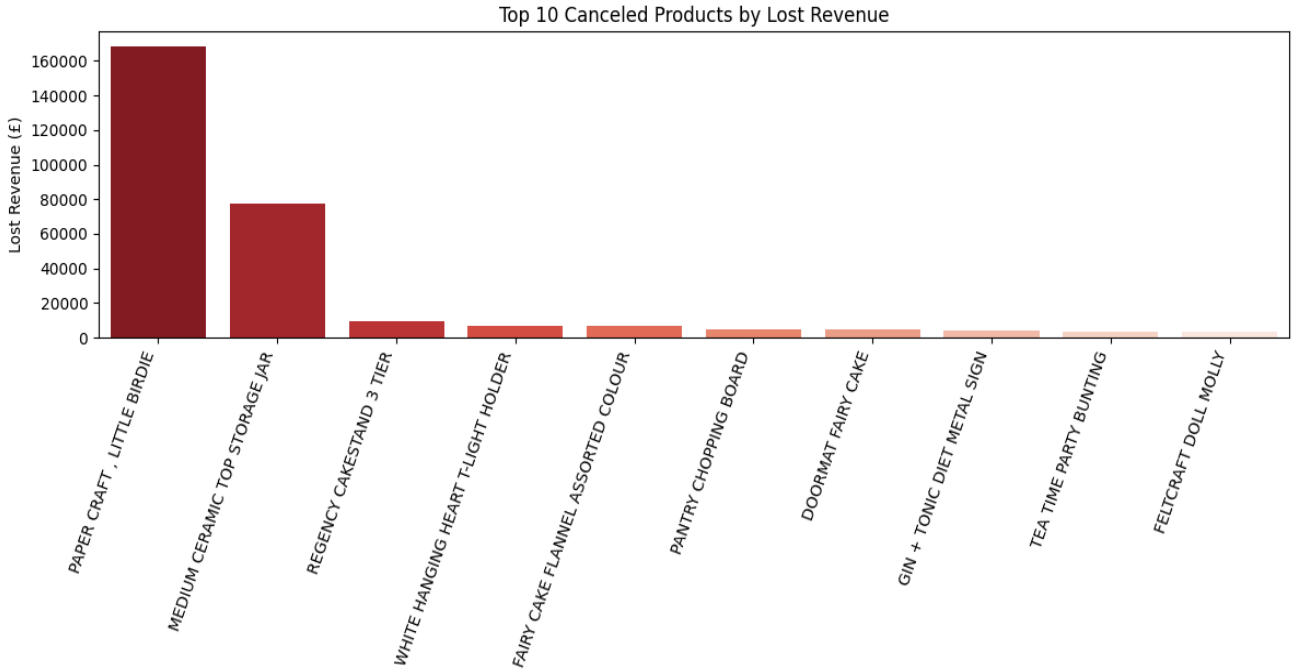
Visualizations:

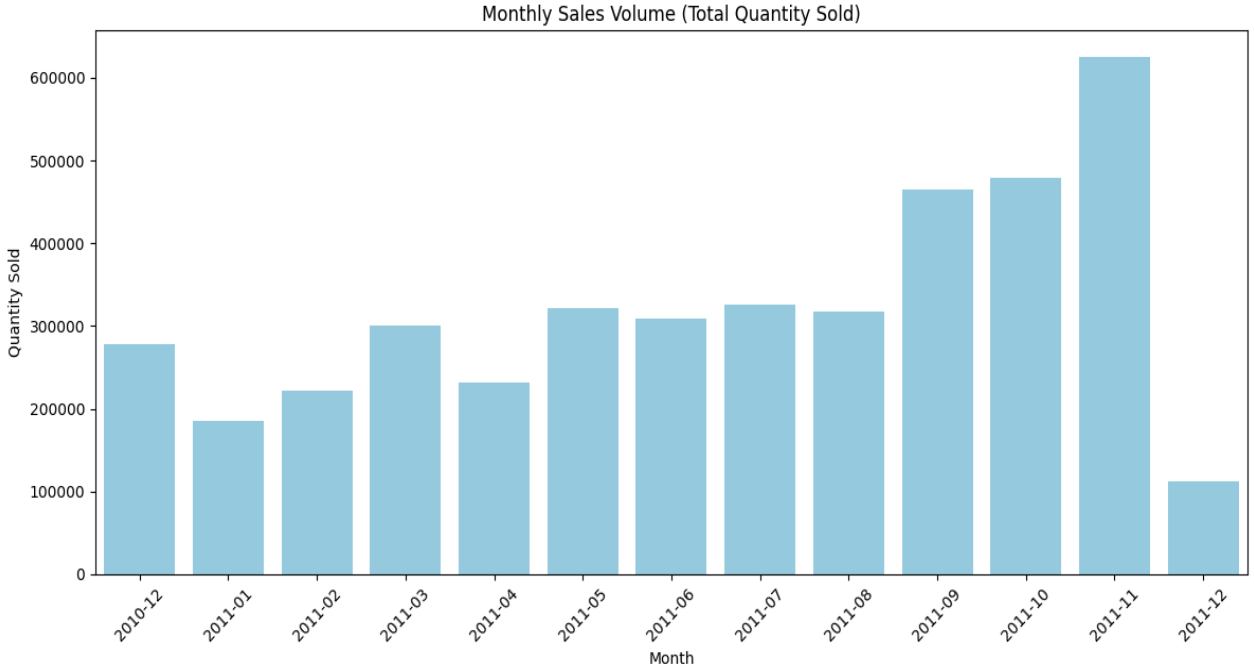






StockCode	Description	Sum of Revenue
22423	REGENCY CAKESTAND 3 TIER	151,934.90
47566	PARTY BUNTING	97,413.88
85123A	WHITE HANGING HEART T-LIGHT HOLDER	69,688.70
85099B	JUMBO BAG RED RETROSPOT	63,579.04
84879	ASSORTED COLOUR BIRD ORNAMENT	51,008.82
22086	PAPER CHAIN KIT 50'S CHRISTMAS	49,511.94
79321	CHILLI LIGHTS	44,924.42
POST	POSTAGE	41,140.90
23298	SPOTTY BUNTING	39,557.27
23084	RABBIT NIGHT LIGHT	36,729.16
23298	BUNTING , SPOTTY	634.70
85123A	CREAM HANGING HEART T-LIGHT HOLDER	178.51
Total		**646,302.24**




Monthly Sales Volume (Total Quantity Sold)
Quantity Sold
0
100000
200000
300000
400000
500000
600000
2010-12
2011-01
2011-02
2011-03
2011-04
2011-05
2011-06
2011-07
2011-08
2011-09
2011-10
2011-11
2011-12
Month