05_consumer_confidence_cleaning_eda

December 27, 2025

1 Consumer Confidence Cleaning and EDA

Data Source: Trading Economics (manually extracted)
Location: `data/interim/consumer_confidence_tradEconDec2025_manual.csv`
Purpose: Clean and validate consumer confidence data
Date: December 2025

1.1 Objectives

1. Clean and validate consumer confidence data
2. Validate against Travel Manitoba Q4 2024 & Q1 2024 infographics
3. Basic trend analysis
4. Prepare dataset for Power BI dashboard

1.2 Setup

[101]:
```
# Path setup
import sys
from pathlib import Path
project_root = Path.cwd().parent
sys.path.insert(0, str(project_root / 'scripts'))
from paths import raw, processed, interim
```

[102]:
```
# Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Plotting style
plt.style.use('seaborn-v0_8-darkgrid')
sns.set_palette('husl')
%matplotlib inline

# Display options
pd.set_option('display.max_columns', None)
pd.set_option('display.float_format', '{:.2f}'.format)
```

```
print('  Libraries loaded')
```

```
 Libraries loaded
```

1.3 Part 1: Data Loading & Cleaning

1.3.1 1.1 Load Raw Data

[103]:
```
csv_path = interim() / 'consumer_confidence_tradEconDec2025_manual.csv'

if not csv_path.exists():
    print(f'ERROR: File not found at {csv_path}')
else:
    print(f'  Found: {csv_path}')
    print(f'   Size: {csv_path.stat().st_size:,} bytes')
```

```
 Found: /Users/dpro/projects/travel_manitoba/data/interim/consumer_confidence_t
radEconDec2025_manual.csv
  Size: 322 bytes
```

[104]:
```
# Load CSV
df_raw = pd.read_csv(csv_path, encoding='utf-8-sig', dtype={'Month': str})

print('RAW DATA')
print('='*80)
print(f'Shape: {df_raw.shape}')
print(f'\nAll rows:')
df_raw
```

```
RAW DATA
================================================================================
Shape: (13, 3)

All rows:
```

[104]:

	Month	canada_consumer_confidence_points	\
0	Dec 2024	47.90	
1	Jan 2025	48.60	
2	Feb 2025	48.80	
3	Mar 2025	48.20	
4	Apr 2025	45.40	
5	May 2025	48.10	
6	Jun 2025	48.80	
7	Jul 2025	48.10	
8	Aug 2025	47.70	
9	Sept 2025	47.80	
10	Oct 2025	46.00	
11	Nov 2025	47.40	
12	Dec 2025	NaN	

```
    usa_consumer_confidence_points
0                            74.00
1                            71.10
2                            64.70
3                            57.00
4                            52.20
5                            52.20
6                            60.70
7                            61.70
8                            58.20
9                            55.10
10                           53.60
11                           51.00
12                           52.90
```

1.3.2 1.2 Clean Data

[105]:
```
# Clean data
df_cleaned = df_raw.copy()

# Ensure Month column stays as string
if 'Month' in df_cleaned.columns:
    df_cleaned['Month'] = df_cleaned['Month'].astype(str)

# Ensure numeric columns are float (skip Month column)
for col in df_cleaned.columns:
    if col != 'Month' and df_cleaned[col].dtype == 'object':
        df_cleaned[col] = pd.to_numeric(df_cleaned[col], errors='coerce')

print('  Cleaned data')
print(f'\nData types:')
print(df_cleaned.dtypes)
```

```
 Cleaned data

Data types:
Month                                object
canada_consumer_confidence_points    float64
usa_consumer_confidence_points       float64
dtype: object
```

1.3.3 1.3 Data Quality Checks

[106]:
```
print('DATA QUALITY SUMMARY')
print('='*80)
print(f'Total rows: {len(df_cleaned)}')
print(f'Total columns: {len(df_cleaned.columns)}')
```

```
print(f'Date range: {df_cleaned["Month"].iloc[0]} to {df_cleaned["Month"].
 ↪iloc[-1]}')

print(f'\nColumns:')
print(df_cleaned.columns.tolist())

print(f'\nMonth values (first 5):')
print(df_cleaned['Month'].head().tolist())

print(f'\nNull values per column:')
null_summary = df_cleaned.isnull().sum()
if null_summary.sum() > 0:
    print(null_summary[null_summary > 0])
else:
    print('None - dataset is complete!')
```

```
DATA QUALITY SUMMARY
================================================================================
Total rows: 13
Total columns: 3
Date range: Dec 2024 to Dec 2025

Columns:
['Month', 'canada_consumer_confidence_points', 'usa_consumer_confidence_points']

Month values (first 5):
['Dec 2024', 'Jan 2025', 'Feb 2025', 'Mar 2025', 'Apr 2025']

Null values per column:
canada_consumer_confidence_points    1
dtype: int64
```

1.3.4 1.4 Validate Against Infographics

[107]:
```
print('VALIDATION AGAINST TRAVEL MANITOBA INFOGRAPHICS')
print('='*80)

# Q4 2024 / Dec 2024 - Jan 2025 period
# From infographic: Canada 48.6 points, USA 71.1 points
print('\nDec 2024 - Jan 2025 Period')
print('-'*60)

# Find most recent data
latest = df_cleaned.tail(2)
print(latest)

if 'canada_consumer_confidence_points' in df_cleaned.columns:
```

```
    canada_latest = df_cleaned['canada_consumer_confidence_points'].iloc[-1]
    print(f'\nCanada (latest): {canada_latest:.1f} points')
    print(f'Expected: ~48.6 points')

if 'usa_consumer_confidence_points' in df_cleaned.columns:
    usa_latest = df_cleaned['usa_consumer_confidence_points'].iloc[-1]
    print(f'\nUSA (latest): {usa_latest:.1f} points')
    print(f'Expected: ~71.1 points')

print('\nNote: Validate actual values match infographic')
```

```
VALIDATION AGAINST TRAVEL MANITOBA INFOGRAPHICS
================================================================================

Dec 2024 - Jan 2025 Period
------------------------------------------------------------
       Month  canada_consumer_confidence_points  \
11  Nov 2025                              47.40
12  Dec 2025                                NaN

    usa_consumer_confidence_points
11                           51.00
12                           52.90

Canada (latest): nan points
Expected: ~48.6 points

USA (latest): 52.9 points
Expected: ~71.1 points

Note: Validate actual values match infographic
```

1.4 Part 2: Basic EDA

1.4.1 2.1 Summary Statistics

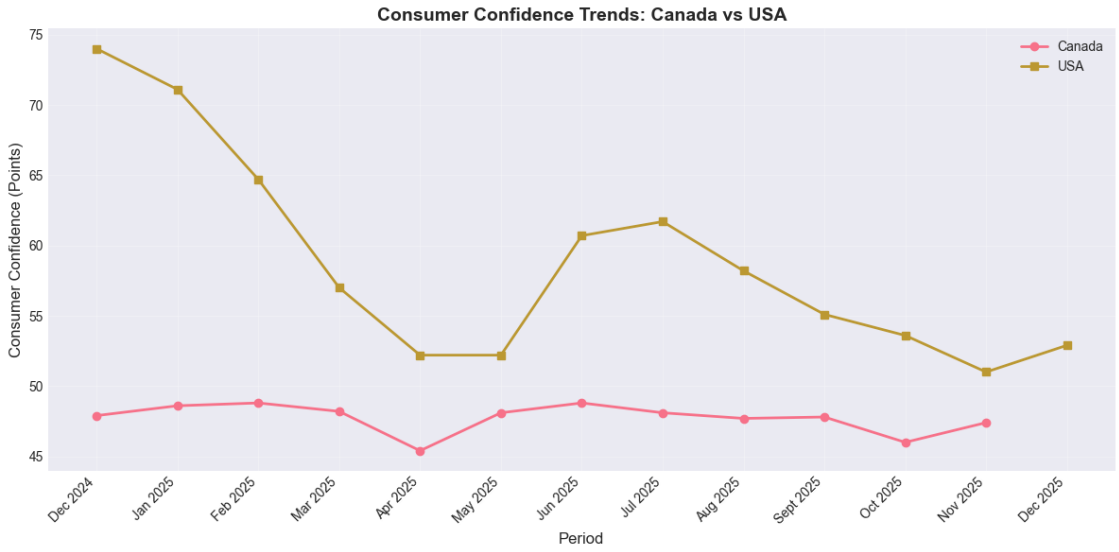
[108]:
```
print('SUMMARY STATISTICS')
print('='*80)
print(df_cleaned.describe())
```

```
SUMMARY STATISTICS
================================================================================
       canada_consumer_confidence_points  usa_consumer_confidence_points
count                              12.00                           13.00
mean                               47.73                           58.80
std                                 1.05                            7.40
min                                45.40                           51.00
25%                                47.62                           52.90
50%                                48.00                           57.00
```

75%	48.30	61.70
max	48.80	74.00

1.4.2 2.2 Consumer Confidence Trends

[109]:
```
# Simple line plot
fig, ax = plt.subplots(figsize=(12, 6))

if 'canada_consumer_confidence_points' in df_cleaned.columns:
    ax.plot(range(len(df_cleaned)),
            df_cleaned['canada_consumer_confidence_points'],
            marker='o', label='Canada', linewidth=2)

if 'usa_consumer_confidence_points' in df_cleaned.columns:
    ax.plot(range(len(df_cleaned)),
            df_cleaned['usa_consumer_confidence_points'],
            marker='s', label='USA', linewidth=2)

ax.set_xlabel('Period', fontsize=12)
ax.set_ylabel('Consumer Confidence (Points)', fontsize=12)
ax.set_title('Consumer Confidence Trends: Canada vs USA',
             fontsize=14, fontweight='bold')
ax.legend()
ax.grid(True, alpha=0.3)

# Set x-axis labels if Month column exists
if 'Month' in df_cleaned.columns:
    ax.set_xticks(range(len(df_cleaned)))
    ax.set_xticklabels(df_cleaned['Month'], rotation=45, ha='right')

plt.tight_layout()
plt.show()

print('\nKey Observations:')
print('- USA consumer confidence consistently higher than Canada')
print('- Track trends over time for dashboard')
```




```
Key Observations:
- USA consumer confidence consistently higher than Canada
- Track trends over time for dashboard
```

1.5 Part 3: Save Processed Data

[110]:
```
# Save cleaned data
output_path = processed() / 'consumer_confidence_clean.csv'
df_cleaned.to_csv(output_path, index=False)

print('  SAVED PROCESSED DATA')
print('='*80)
print(f'Location: {output_path}')
print(f'Size: {output_path.stat().st_size:,} bytes')
print(f'Shape: {df_cleaned.shape}')
print(f'\nReady for Power BI import!')
```

```
  SAVED PROCESSED DATA
================================================================================
Location: /Users/dpro/projects/travel_manitoba/data/processed/consumer_confidenc
e_clean.csv
Size: 315 bytes
Shape: (13, 3)

Ready for Power BI import!
```

1.6 Summary

1.6.1 Data Cleaning

- Loaded manual CSV from interim directory
- Cleaned numeric formatting
- Validated against infographic values
- Saved to `data/processed/consumer_confidence_clean.csv`

1.6.2 Key Findings

1. USA consumer confidence higher than Canada
2. Monthly tracking available for trend analysis

1.6.3 Next Steps

1. Import `consumer_confidence_clean.csv` into Power BI
2. Create visualizations matching Travel Manitoba style