

SAP IBP Forecast Accuracy & Business Impact Simulation: Step-by-Step Summary

1. Project Goal and Strategic Context

The aim of this project was to simulate the logic and business value of SAP Integrated Business Planning (IBP) using Excel and Python, specifically to:

- Measure and improve forecast accuracy across SKUs
- Evaluate the financial and sustainability impact of demand planning
- Stress test business scenarios such as carbon tax, demand shock, and cash flow inefficiencies

This was done without access to a licensed SAP IBP environment, using Excel and open-source tools to replicate the logic and output of a demand planning simulation.

2. Dataset Design and Business Assumptions

We began by generating synthetic forecast vs actual data for 3 SKUs over a 24-month period:

- **SKUs:** SKU_A, SKU_B, SKU_C
- **Time Frame:** January 2023 – December 2024
- **Fields:** Month, SKU, Forecast, Actual

Each SKU was given different patterns:

- SKU_A: Moderate volatility and slight underforecasting.
- SKU_B: Consistent overforecasting
- SKU_C: High variance simulating difficult-to-predict demand

This setup allowed us to simulate a range of forecast errors and downstream operational risks.

3. Setting Up the Excel Engine

We created a workbook with the following structure:

- **ForecastData:** Contains raw forecast vs actual by SKU and month
- **KPI_Summary:** Calculates forecast KPIs: MAPE, Bias, Forecast Accuracy
- **Margin_Analysis:** Computes forecasted vs actual margins using unit price and cost
- **Carbon_Impact:** Simulates ESG exposure (kgCO2e per unit and revenue/kgCO2e)
- **Scenario_Compare:** Tracks before vs after values for accuracy and margin in stress tests
- **Forecast_to_Cash:** Compares forecasted revenue to actual realized revenue and cash

We used Excel formulas for all KPI calculations:

- **MAPE:** =AVERAGE(ABS(Forecast - Actual)/Actual)

- **Bias:** $=\text{SUM}(\text{Forecast} - \text{Actual})/n$
- **Forecast Accuracy:** $=1 - \text{MAPE}$

Each formula was carefully verified using Excel's fill-down feature, slicers for SKU filtering, and helper columns.

4. Python Setup (Optional)

To replicate the same logic programmatically:

- We set up a virtual environment using `python3 -m venv venv`
- Installed libraries: pandas, matplotlib, seaborn
- Wrote a Python script to import the CSV dataset and compute the same KPIs
- Used seaborn/matplotlib to generate plots for forecast accuracy and margin

This allowed us to validate Excel results and produce Python visualizations.

5. Stress Testing Business Scenarios

Scenario C: Carbon Tax

- A tax of \$0.05 per kgCO₂e was added to simulate policy pressure
- SKU_C had the best sustainability efficiency (revenue/kgCO₂e)
- Margin impact was analyzed before and after tax

Scenario D: Demand Shock

- Forecast accuracy KPIs were split into before/after shock
- The shock period was marked by sudden demand volatility
- Scenario_Compare sheet tracked accuracy shifts visually

Scenario F: Forecast-to-Cash Conversion

- Forecasted revenue was compared with actual cash received
- SKU_B showed discrepancies due to delayed realizations
- KPI: Forecast-to-Cash Efficiency = $\text{Cash Realized} / \text{Forecasted Revenue}$

6. Visualization and Dashboarding

All charts were built in Excel:

- Forecast Accuracy by SKU
- Forecast vs Actual Margin
- Revenue per kgCO₂e
- Scenario Before vs After Accuracy
- Total Inventory Holding Cost under Forecast Error

- Forecast vs Actual vs Cash Realized

Charts were exported as PNGs using right-click → Save as Picture.

7. GitHub Repository Structure

Created a clean, modular repository:

- /data: Input CSV files
- /excel_outputs: Final workbook
- /images: All chart PNGs
- README.md: Contains project overview, KPI logic, scenario descriptions

Uploaded via VS Code using `git add .`, `git commit`, and `git push`.

8. Executive Deck (PDF Storyboard)

Created an 8-slide PDF:

1. Title
2. Business Context
3. Forecast Accuracy
4. Margin Impact
5. ESG KPI (Revenue/kgCO₂e)
6. Scenario Tests (C, D, F)
7. KPI Summary Table
8. GitHub Access & Next Steps

Deck was exported from PowerPoint with GitHub link embedded.

9. Strategic Value and Outcome

- Demonstrated how Excel replicates IBP logic
- Highlighted ESG and financial metrics from forecasts
- Introduced simulation-based planning and scenario design
- Prepared artifacts for GitHub, resume, and stakeholder presentations