

Financial Planning & Analysis (FP&A) Forecasting Dashboard

1. Introduction

The FP&A Forecasting Dashboard is an interactive web application designed to provide insightful financial analysis using historical data and predictive modeling. The dashboard visualizes forecasts, actuals, and anomalies for key financial metrics like revenue, cost of goods sold (COGS), operating expenses, and net income. Built using Python and Streamlit, it integrates multiple components for forecasting, visualization, and business metric evaluation.

2. Objective

The main goal of this project is to enable better financial planning by:

- Forecasting key financial metrics for upcoming periods
- Comparing actual vs forecast values
- Measuring forecast accuracy using metrics
- Detecting anomalies in financial trends
- Providing a clean, interactive dashboard for business users

3. Technologies Used

- Python
- Streamlit (Dashboard UI)
- Matplotlib (Visualization)
- Pandas (Data manipulation)
- Custom modules for Forecasting, Metrics, Budget Comparison, and Anomaly Detection

4. Dataset Details

The project uses synthetically generated but realistic historical financial data to simulate a business's financial operations over time. The datasets include:

- Historical Revenue: Monthly revenue values for the past 60 months.
- Historical Costs (COGS): Monthly cost of goods sold values.
- Operating Expenses: Monthly general and administrative operating expenses.
- Forecast Period: Defined (e.g., 6 months) to generate forecasts using statistical models.
- Budget (Optional): Expected values set by finance teams to evaluate budget compliance.

Data is stored and loaded using Python dictionaries and Pandas Series. The format of data ensures flexibility and extensibility for plugging in real financial datasets later.

5. Forecasting Approach

Forecasting is done using statistical models (such as ARIMA or Prophet depending on setup). Separate functions are used to predict:

- Revenue
- Cost of Goods Sold
- Operating Expenses
- Net Income (calculated from above)

6. Metrics Implemented

- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
- Mean Absolute Error (MAE)
- Profit Margin Calculation
- Revenue Growth Rate

These metrics help assess both business performance and model prediction accuracy.

7. Dashboard Features

- Summary metrics at the top (Profit Margin, Growth)
- Accuracy statistics under expandable sections
- Clean plots comparing Actual vs Forecast for each metric
- Tabular view for side-by-side comparisons
- Anomaly detection alerts for sudden spikes or drops

8. Deployment

The app can be deployed locally using Streamlit CLI or hosted using platforms like Streamlit Community Cloud, Heroku, or AWS.

9. Future Improvements

- Integrate real-time financial APIs
- Add authentication and multi-user support
- Improve UI with Plotly or Altair for advanced visuals
- Extend anomaly detection with ML models