Investment Analysis Platform

Description

The Investment Analysis Platform is a comprehensive Python-based web application for analyzing investment data, generating insights, and creating detailed reports. Built with Streamlit, this platform allows investors and financial analysts to explore ETF data, benchmark comparisons, and predictive trends through an intuitive interface.

Key Features

- **Portfolio Analytics**: Analyze portfolio composition, risk metrics, and performance statistics
- **Benchmark Comparison**: Compare investment performance against major market indices
- **Sector Analysis**: Visualize sector allocations and compare across different funds
- **Predictive Models**: Forecast future price movements using linear regression and ARIMA models
- **PDF Report Generation**: Create professional investment reports with key metrics and visualizations
- **Data Visualization**: Interactive charts for prices, returns, correlations, and performance metrics
- **Performance Metrics**: Calculate Sharpe ratios, drawdowns, alpha, beta, and other key financial metrics

Technical Details

- **Frontend**: Streamlit for interactive web interface
- **Data Analysis**: Pandas and NumPy for data processing
- **Visualization**: Plotly and Matplotlib for interactive charts
- **Statistical Models**: Statsmodels and Scikit-learn for predictive analytics
- **Reporting**: ReportLab for PDF generation
- **Deployment**: Compatible with Azure for cloud hosting

How It Works

The platform follows a modular architecture with specialized components:

1. **Data Processing**: Clean and transform financial data
2. **Portfolio Analysis**: Calculate risk-return metrics and portfolio statistics
3. **Visualization**: Generate interactive charts for data exploration
4. **Predictive Modeling**: Forecast future price movements
5. **Report Generation**: Create comprehensive PDF reports

Future Enhancements

- Dividend history analysis for total return calculations
- Portfolio optimization tools for ideal asset allocation
- Economic indicators integration for contextual analysis
- Expanded benchmarking capabilities
- User authentication and saved portfolio tracking

Deployment

The application is designed to be deployed on Azure services, including App Service, Container Instances, or Kubernetes Service depending on scalability needs.

Getting Started

To run the platform locally:

1. Clone this repository
2. Install required packages: pip install -r requirements.txt
3. Run the application: streamlit run app.py

Data Requirements

The platform works with these data formats:

- ETF daily price changes
- ETF sector allocations
- Benchmark data
- Price history data