Portfolio Account Manager v2

By: Kirill Panov, Rachel Donham, Isaiah Tensae, Olena Shemedyuk

Summary

Project 1 Summary:

1. Log In / Sign up
2. Email Verification
   a. API Generates a 6 digit code
   b. 6 digit code is emailed to the email associated with the user
   c. User inputs code into terminal
   d. Verifies their account
3. Portfolio Management (Crypto & Stock)
   a. Add a portfolio
   b. Edit a portfolio
   c. Remove A portfolio
4. Asset Management
   a. Add Asset
   b. Add Existing asset
5. Portfolio analyzer (only visible after email verification)

Project 2 Summary:

- Log In / Sign up
- Email Verification
  - API Generates a 6 digit code
  - 6 digit code is emailed to the email associated with the user
  - User inputs code into terminal
  - Verifies their account
- Portfolio Management (Crypto & Stock)
  - Add a portfolio
  - Edit a portfolio
  - Remove A portfolio
- Asset Management
  - Add Asset
  - Add Existing asset
  - Asset Performance Analyzer
    - Predictions based on technical indicators and ML models
    - Visualizations (based on technical indicators)
- Portfolio analyzer (only visible after email verification)

Goal

- Build a model that returns a recommendation on whether to trade long or short using a number of technical indicators and Binary Classifier.
- Build asset prediction and analysis functionalities in the current Portfolio Account Manager Analyzer Tool.
- Visualize used technical indicators for the whole analyzed period.

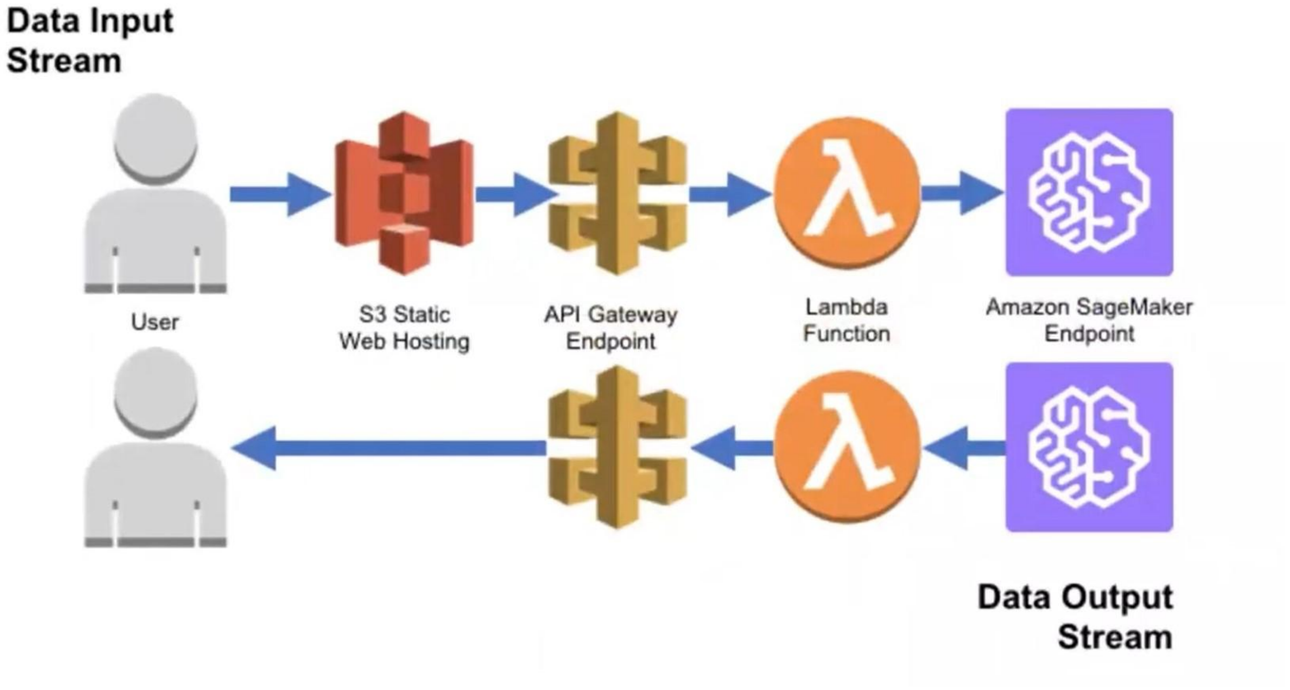
Machine Learning Model

- New feature provides the user with Simple and Professional modes to analyze the current asset in the portfolio
- Simple mode uses default predictors, default ML model and default rolling windows to calculate technical indicators
- Professional mode, that is available for verified users, provides the user with the choice of predictors (technical indicators) and models to build and fit different combinations of the modeles and analyze outcomes
- The current version provides a number of technical indicators that are used by the model to predict the target: SMA, EMA, Average Directional Index (ADX), Average True Range (ATR), Relative Strength Index (RSI), MACD, Rate of Change(ROC)
- Indicators are based on two rolling windows - fast and long. Configurable in Professional Mode
- The target is the possible direction of movement of the asset (Stock or Crypto) X days later, where X is configurable for the Professional mode.
- The current version supports RandomForestClassifier and KNeighborClassifier
    - Both are a commonly used machine learning models for classification problems.
    - The feature is flexible and allows to add more models, predictors(indicators) and their combinations in the future
    - Each model has number of parameters that can be tuned and configured using the GridSearchCV module which works with different specified ranges of these parameter values and gives the best combination (estimator) that would have the lowest error in cross-validation (TimeSeries cross-validator).

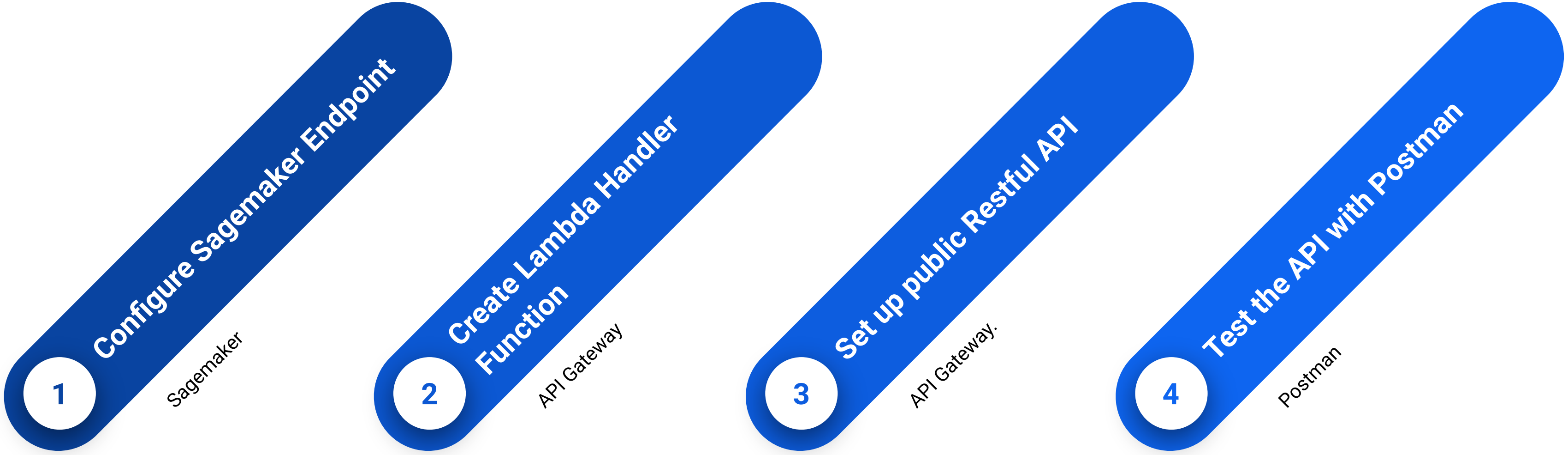
Machine Learning Model in Sagemaker

- Created a second “placeholder” machine learning model in Sagemaker
  - To achieve our final objective of a cloud-based, ML-powered REST API service that generates trade recommendations
  - This service will be called by the Portfolio Account Manager
  - The inputs will be an asset class (stocks/crypto) and a ticker
  - The output will be recommendation (0 or 1) and probability “score”
- Used the Sagemaker’s Linear Learner Algorithm
  - As a binary classifier
  - Initially built using the class exercise instructions for Module 15 with datafile from the Module 14 challenge
  - Some extensions from that lesson
    - Use a notebook instance
      - Github repository integration
    - Build a multi-model endpoint to respond to recommendation requests for different tickers

AWS API Architecture



Steps to deploy the ML Service API



1 Configure Sagemaker Endpoint

Sagemaker

2 Create Lambda Handler Function

API Gateway

3 Set up public Restful API

API Gateway.

4 Test the API with Postman

Postman

Data Collection, Exploration, & Cleanup

- Used the same data sources from project 1 (Portfolio Account Manager).
  - **Alpha Vantage API & Alpaca SDK** to get the current and historic asset prices for portfolio analysis
    - Using API on demand not storing any prices in database
- Dropping NaN values after calculating technical indicators (different types of moving averages using different rolling windows).

The Approach

Project Demo

Result

1. The Portfolio Account Manager tool is updated with new features that allows users to get more insight on their decision whether to purchase an asset along with deep analysis of the asset performance.
2. The Portfolio Account Manager is updated to produce a visualization of the outlook of the users current investments.

Next Steps

- Using more technical indicators and ML models to predict the assets performance
- Having the ability to directly buy and sell through the integration with real exchange.
- Apply new functionality not only for the particular asset but for the whole portfolio.
- Change data sources/API to collect more historical data (Alpaca and Alpha Vantage are limited by 1000 historical records). More data will allow us to better train the model and make better predictions.
- Generate entry/exit signals