



Institución Universitaria Digital de Antioquia IUDIGITAL

Faculta de Ingeniería y Ciencias Agropecuarias

Ingeniería de Software y datos

S30 - Proyecto Integrado V - Linea de Énfasis

Sistema de recolección de datos financieros E-Mini S&P 500 (ES = F)

**Omar Aleiser Alvarez Laverde
PREICA2501B020128**

Bucaramanga, Santander
11 Mayo 2025

Resumen

Este proyecto implementa un sistema automatizado para la recolección, procesamiento y análisis de datos históricos del indicador económico E-Mini S&P 500. Utilizando Python, se desarrolló una arquitectura modular que permite la extracción de datos mediante la API de Yahoo Finance, su enriquecimiento con indicadores técnicos, y la creación de modelos predictivos. El sistema se integra con GitHub Actions para actualizar los datos periódicamente, garantizando información actualizada para análisis financieros.

Introducción

En el contexto actual de los mercados financieros, el acceso a datos históricos precisos y actualizados es fundamental para la toma de decisiones informadas. El E-Mini S&P 500 es un contrato de futuros basado en el índice S&P 500, ampliamente utilizado como referencia para el mercado de valores estadounidense.

Este proyecto aborda la necesidad de automatizar la recolección y procesamiento de datos históricos del E-Mini S&P 500, implementando una solución que no solo extrae información de fuentes confiables, sino que también la enriquece con indicadores técnicos relevantes y desarrolla modelos predictivos para anticipar movimientos futuros del mercado.

La solución desarrollada se basa en principios de ingeniería de software modernos, incluyendo modularidad, mantenibilidad y automatización, permitiendo a analistas financieros y traders acceder a datos procesados y análisis predictivos de manera eficiente.

Objetivo general

Desarrollar un sistema automatizado para la recolección, procesamiento y análisis de datos históricos del indicador económico E-Mini S&P 500, que permita la generación de insights y predicciones para la toma de decisiones en el ámbito financiero.

Objetivo específicos

1. Implementar un módulo de recolección de datos que extraiga información histórica del E-Mini S&P 500 desde Yahoo Finance y la almacene en formatos estructurados (CSV y SQLite).
2. Desarrollar un módulo de enriquecimiento de datos que calcule indicadores técnicos relevantes como medias móviles, RSI, volatilidad y retornos acumulados.
3. Crear un módulo de modelado predictivo que utilice técnicas de aprendizaje automático para predecir movimientos futuros del E-Mini S&P 500.
4. Integrar el sistema con GitHub Actions para automatizar la actualización periódica de los datos y mantener la información actualizada.
5. Implementar un sistema de logging robusto que permita el seguimiento y depuración de los procesos automatizados.

Metodología

Arquitectura del Sistema

El sistema se desarrolló siguiendo una arquitectura modular con los siguientes componentes:

Módulo de Recolección (Collector): Responsable de extraer datos históricos del E-Mini S&P 500 utilizando la biblioteca `yfinance`, que proporciona acceso a la API de Yahoo Finance. Los datos se almacenan en formatos CSV y SQLite para facilitar su acceso y manipulación.

Módulo de Enriquecimiento (Enricher): Procesa los datos recolectados y calcula indicadores técnicos como:

- Medias móviles simples (SMA) de diferentes períodos
- Índice de Fuerza Relativa (RSI)
- Bandas de Bollinger
- MACD (Moving Average Convergence Divergence)
- Volatilidad histórica
- Retornos diarios y acumulados

Módulo de Modelado (Modeller): Implementa diversos algoritmos de aprendizaje automático para predecir movimientos futuros del precio, incluyendo:

- Regresión Lineal
- Regresión Ridge y Lasso

- Random Forest
- Gradient Boosting
- Support Vector Regression (SVR)

Módulo de Logging: Proporciona un sistema de registro detallado para monitorear la ejecución de los procesos y facilitar la depuración.

Tecnologías Utilizadas

- **Python:** Lenguaje de programación principal
- **pandas:** Manipulación y análisis de datos
- **yfinance:** Acceso a datos históricos de Yahoo Finance
- **scikit-learn:** Implementación de modelos de aprendizaje automático
- **matplotlib/seaborn/plotly:** Visualización de datos
- **SQLite:** Almacenamiento estructurado de datos
- **GitHub Actions:** Automatización de procesos
- **joblib:** Serialización de modelos

Parte final

Dashboard para E-Mini S&P 500 (ES=F) con Modelo Ridge

Resumen

Este documento presenta el desarrollo de un dashboard analítico para el futuro E-Mini S&P 500 (ES=F), implementando un modelo de Regresión Ridge para la predicción de precios. La herramienta integra:

- Visualización interactiva de datos históricos.
- Indicadores técnicos (RSI, medias móviles, volatilidad).
- Correlaciones de mercado.
- Predicciones con intervalos de confianza.

El sistema está desarrollado en Python, utilizando bibliotecas como Streamlit para la interfaz y Plotly para gráficos interactivos.

Introducción

El E-Mini S&P 500 (ES=F) es uno de los contratos de futuros más negociados a nivel global, sirviendo como referencia clave para inversores institucionales y traders.

Este proyecto surge de la necesidad de:

- Automatizar el análisis técnico en tiempo real.
- Optimizar la toma de decisiones mediante modelos predictivos.
- Centralizar la información en una interfaz accesible.

El dashboard desarrollado permite monitorear el comportamiento del mercado, identificar tendencias y evaluar predicciones basadas en aprendizaje automático.

Justificación

Necesidad del Proyecto

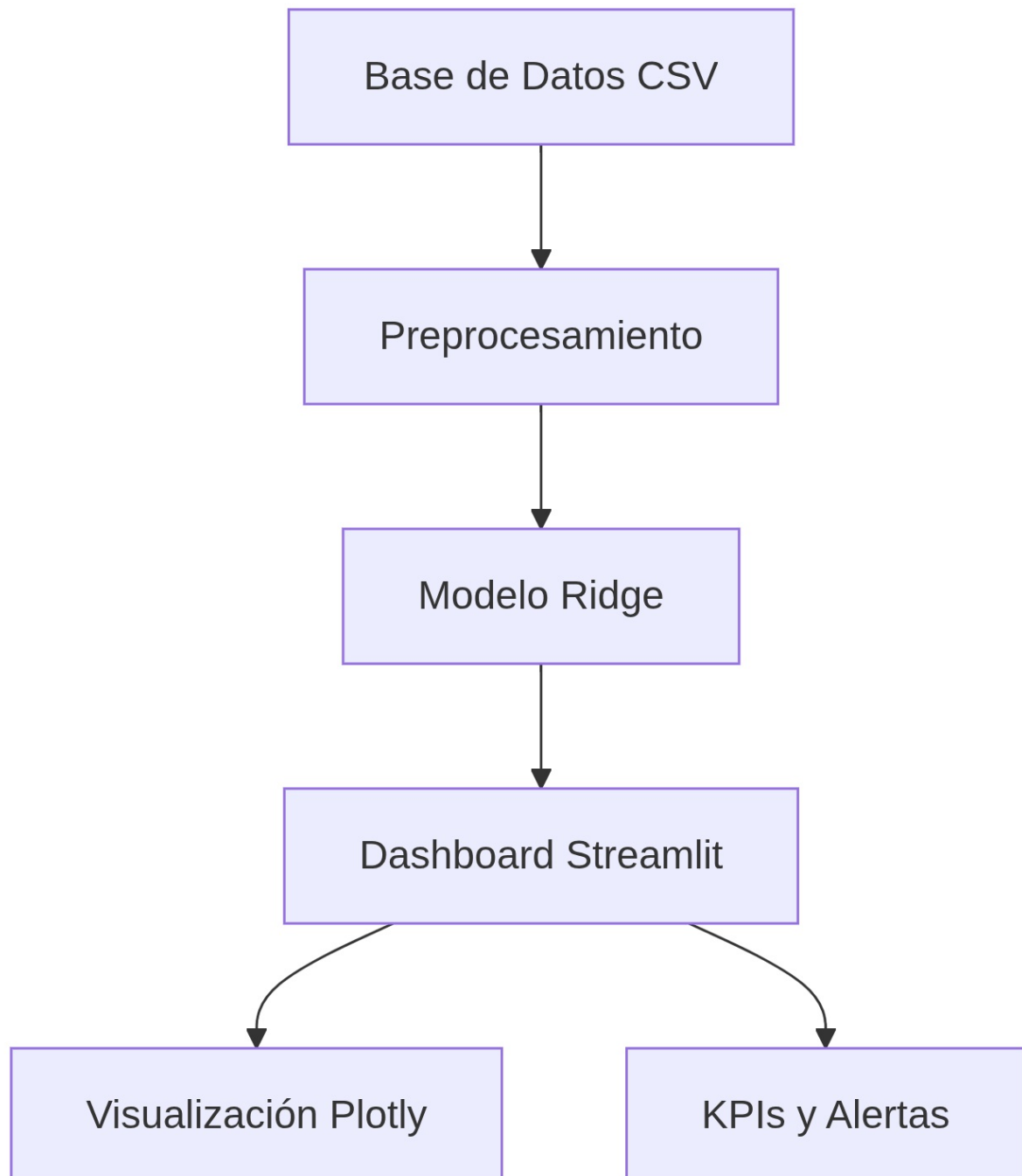
Los traders requieren herramientas que combinen:

- ✓ Visualización intuitiva de datos financieros.

- ✓ Indicadores técnicos avanzados.
- ✓ Modelos predictivos confiables.

Arquitectura del Sistema

Diagrama de Componentes



Tecnologías Utilizadas

Componente	Tecnología/Biblioteca	Función
Interfaz gráfica	Streamlit	Dashboard web interactivo
Visualización	Plotly	Gráficos dinámicos
Análisis técnico	Pandas/Numpy	Cálculo de indicadores
Modelo predictivo	Scikit-learn (Ridge)	Predicción de precios
Gestión de datos	CSV/SQLite	Almacenamiento y consulta

Flujo de Datos

Carga: Datos históricos desde archivos CSV.

Procesamiento:

- Cálculo de returns, volatilidad y RSI.
- Normalización de features para el modelo.
- Predicción: Generación de pronósticos con intervalos de confianza.
- Visualización: Renderizado en tiempo real en el navegador.

Conclusión

Este proyecto demuestra cómo la aplicación de técnicas de programación modernas y principios de ingeniería de software puede crear soluciones robustas para el análisis de datos financieros, proporcionando herramientas valiosas para la toma de decisiones informadas en el ámbito de las inversiones.

Link acceso al github:
https://github.com/oalvarez1990/PROJECT_INTEGER_V_2_05

Link acceso al video
<https://drive.google.com/file/d/1ehuC8n7Sh41hAkhMWan06OQqmM8ewLlr/view?usp=sharing>

Link acceso al despliegue: <https://projectv.streamlit.app/>

Bibliografía

- Yahoo Finance API Documentation. (2023). *Yahoo Finance API*. <https://finance.yahoo.com/>
- McKinney, W. (2022). *Python for Data Analysis: Data Wrangling with pandas, NumPy, and Jupyter* (3rd ed.). O'Reilly Media.
- Géron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (3rd ed.). O'Reilly Media.
- Murphy, J. J. (2022). *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications* (Revised ed.). New York Institute of Finance.
- GitHub Actions Documentation. (2023). *GitHub Actions*. <https://docs.github.com/en/actions>
- Scikit-learn: Machine Learning in Python. (2023). *Scikit-learn Documentation*. <https://scikit-learn.org/stable/documentation.html>