

Propuesta: *Framework* de Trading Cuantitativo para el Mercado de Criptomonedas en la modalidad de futuros

Tabla de Contenidos

1. [Introducción](#)
 2. [Contexto](#)
 3. [Propuesta](#)
 1. [Obtención de Datos](#)
 2. [Observación y Transformación de Datos](#)
 3. [Modelación](#)
 4. [Análisis de Resultados](#)
 5. [Salida del Framework](#)
 4. [Conclusiones](#)
-

Introducción

Este documento presenta una propuesta integral para desarrollar un *framework* de trading cuantitativo para el mercado de criptomonedas, específicamente en la modalidad de futuros. La propuesta se realiza desde el punto de vista de la Ciencia de Datos, aprovechando el *machine learning* y el análisis de series temporales, con un enfoque robusto y basado en datos.

En primer lugar, el documento explica el problema y el contexto de este escenario complejo. Luego, define la propuesta, incluyendo soporte bibliográfico para recomendaciones específicas y análisis basado en casos para los detalles que dependen de los resultados de la implementación.

Contexto

Quantitative trading (trading cuantitativo) es un método que utiliza modelos matemáticos, computacionales y estadísticos para identificar oportunidades de trading. Es una estrategia que se enfoca en el análisis cuantitativo de vastos conjuntos de datos para generar perspectivas y decisiones de trading ^{^1}.

El **mercado de criptomonedas** consiste en monedas digitales o virtuales que utilizan criptografía para seguridad y operan en redes descentralizadas, típicamente basadas en tecnología *blockchain* ^{^2}.

Este mercado se define por varias características clave que lo diferencian de los mercados financieros tradicionales. Opera las 24 horas del día, los 7 días de la semana, sin cierres de mercado. También es conocido por su alta volatilidad, con fluctuaciones significativas de precios que ocurren en períodos de tiempo muy cortos, lo que presenta oportunidades únicas y riesgos para los *traders*. Además, el mercado proporciona una transparencia sin precedentes a través de datos *blockchain* disponibles públicamente, lo que permite el análisis de métricas *on-chain* ^{^2} ^{^3}.

En el volátil mercado de las criptomonedas, los algoritmos pueden procesar grandes cantidades de datos históricos y realizar análisis complejos para proporcionar proyecciones, lo que ofrece una ventaja significativa

en un entorno donde el tiempo es crítico.

Un **futures contract (contrato de futuros)** es un acuerdo legal estandarizado para comprar o vender un activo subyacente a un precio predeterminado en una fecha futura específica. En el contexto de las criptomonedas, el trading de *futures* permite especular sobre los movimientos de precios sin poseer directamente el activo. Un instrumento dominante es el *perpetual swap (swap perpetuo)*, un contrato de *futures* sin fecha de vencimiento ^{^4}. Su precio está vinculado al precio *spot* subyacente a través de un mecanismo de *funding rate* (tasa de financiación) periódica, donde las posiciones largas y cortas intercambian pagos en función de la diferencia de precio.

El trading de *futures* emplea mecanismos de margen sofisticados con consideraciones de riesgo importantes. Los requisitos de *initial margin* (margen inicial) suelen oscilar entre el 2% y el 50% (*leverage* de 50x a 2x), y los niveles de *maintenance margin* (margen de mantenimiento) activan advertencias de *liquidation*. La elección entre los modos de *cross-margin* y *isolated margin* representa una decisión crítica de gestión de riesgos.

El *isolated margin* asigna margen a posiciones individuales de forma independiente, lo que limita la pérdida al margen asignado y evita la contaminación entre posiciones, pero con una menor eficiencia de capital y un posible sub-utilización. Este modo se adapta a nuevas estrategias, posiciones de alto riesgo y enfoques experimentales. El *cross margin* utiliza un *pool* (fondo) compartido de margen en todas las posiciones, lo que ofrece una mayor eficiencia de capital y una mejor utilización del margen, pero conlleva un riesgo de contagio y la posibilidad de liquidaciones en cascada. Este modo funciona mejor para estrategias establecidas y para la gestión de riesgos a nivel de portafolio.

Propuesta

1. Obtención de Datos

La estrategia de recolección de datos debe diferenciar entre las fases de modelación/entrenamiento y el uso en producción.

Para los datos a recolectar en la fase de modelación y entrenamiento, se requiere una representación integral de todos los escenarios con una representación balanceada entre diferentes criptomonedas, pares de trading y condiciones de mercado. Los datos históricos deben abarcar múltiples años (mínimo 3-5 años) para capturar patrones estacionales, ciclos de mercado y tendencias a largo plazo. Los datos multi-frecuencia deben estar correctamente etiquetados para permitir el entrenamiento en diferentes períodos de tiempo, y los datos deben incluir diversos regímenes de mercado con una representación apropiada.

Para el uso en producción en tiempo real, son esenciales los flujos continuos de datos, en el mismo formato que los datos de entrenamiento. Estos datos sirven tanto para predicciones como para el monitoreo continuo del rendimiento del modelo final, con sistemas automatizados de alerta para eventos especiales del mercado y degradación del rendimiento del modelo.

Detalles de implementación

Para la recolección de datos, el *pipeline* debe realizar una extracción regular de datos de los principales *exchanges* de criptomonedas como Binance, OKX y Bybit con frecuencia de extracción configurable y

programación adaptativa durante períodos de alta volatilidad. El *pipeline* debe soportar múltiples tipos de datos incluyendo *OHLCV*, datos de *order book*, *funding rates* y *open interest*.

Los datos deben contener meta-parámetros para un análisis mejorado más allá de las fluctuaciones de precios. Incluyendo identificadores de *exchange*, métricas de liquidez, clasificaciones de activos, niveles de capitalización de mercado, jurisdicción regulatoria, etc.

Para la infraestructura en tiempo real, son esenciales los clientes *WebSocket* robustos con lógica de *retry*, sistemas de colas de mensajes para manejar ráfagas de datos, procesamiento paralelo para flujos de datos de alta frecuencia, y monitoreo integral para la completitud y calidad de datos, con alertas automatizadas integradas en los sistemas de validación del modelo.

2. Observación y Transformación de Datos

Esta sección se enfoca en el procesamiento de datos durante la fase de entrenamiento y selección del modelo. Las transformaciones determinadas al final de la primera iteración de entrenamiento serán integradas al uso en tiempo real del *framework*.

Dado que el conjunto de datos es extenso y variado, es necesario utilizar herramientas automatizadas de observación de datos. Se requiere incluir análisis automático de continuidad para detectar brechas en los datos, detección estadística de valores atípicos usando *Z-score* y umbrales específicos del dominio, validación de consistencia lógica y monotonicidad temporal; para luego analizar estos casos manualmente.

Los datos necesitan ser limpiados, utilizando múltiples enfoques según los problemas detectados. Por ejemplo, eliminación de datos para corrupción irrecuperable, relleno hacia adelante (*forward-filling*) para interrupciones cortas, interpolación lineal para brechas medianas, imputación probabilística para datos multivariados faltantes complejos, y validación multi-fuente usando proveedores de datos alternativos para brechas críticas.

Luego los datos necesitan ser normalizados. En este tipo de datos las principales preocupaciones son la sincronización de *timestamps* entre todas las fuentes, normalización de divisas a referencias estables, escalado ajustado por volatilidad para características basadas en precios, y técnicas de escalado robustas para manejar distribuciones de cola pesada.

Si tenemos datos categóricos, deben ser codificados usando el método correcto según el tipo de característica; ya sea *Label Encoding*, *Ordinal Encoding*, *One-Hot Encoding*, u otros.

El análisis de *features* involucra análisis de correlación para identificar *features* redundantes, análisis de componentes principales para reducción de dimensionalidad en conjuntos con *features* altamente correlacionados, y otros, dependiendo de los datos.

División de datos

Después de que los datos son limpiados y transformados apropiadamente, es necesario determinar cómo dividir los datos para realizar el entrenamiento en varios modelos con variedad de hiper-parámetros, luego probar en las métricas elegidas y ajustar los parámetros, mientras se mantiene separado un grupo representativo de datos, nunca visto por ningún modelo, para realizar la validación y selección final del modelo.

Los métodos estándar de validación cruzada, particularmente los enfoques K-Fold, no son apropiados para series temporales financieras. Es necesario tener en cuenta las dependencias temporales inherentes, para evitar que información futura influya en predicciones pasadas.

La estrategia propuesta es realizar una estrategia *walk-forward*. Definiendo una ventana de entrenamiento grande que abarque el 80-90% inicial de los datos cronológicamente. Luego una ventana de validación que contenga la última porción de la serie temporal que debe mantenerse separada del conjunto de entrenamiento hasta el final del proceso de modelación. La ventana grande de entrenamiento puede entonces dividirse en ventanas deslizantes de segmentos de train-test durante el entrenamiento de los modelos.

3. Modelación

Modelos Base y Candidatos

Es recomendable tener un modelo de "baseline" como referencia, para este *framework* se puede utilizar un modelo ARIMA ^{[^5](#)} como comparación.

Los principales modelos candidatos basados en la revisión de literatura incluyen *Gradient Boosting Machines* como *LightGBM* y *XGBoost*, variantes de redes *LSTM* para captura de dependencia temporal, híbridos *CNN-LSTM* para reconocimiento de patrones multi-escala, y más recientemente redes *Transformer* para modelado de dependencias ^{[^6](#) [^7](#)}.

Investigaciones recientes indican que las arquitecturas *Conv-LSTM*, *LSTM* bidireccionales, y *XGBoost Regressor* adecuadamente regularizados típicamente demuestran un rendimiento sólido en tareas de predicción de criptomonedas.

Métricas de Evaluación

Las métricas más frecuentes en la literatura, para el modelo propuesto, durante el entrenamiento son el Error Absoluto Medio (*MAE*), Error Cuadrático Medio (*MSE*), y R-cuadrado (*R²*).

Sin embargo, en la etapa de validación se deben incluir métricas más específicas al contexto. Como el *Sharpe Ratio* y *Sortino Ratio*, análisis de *drawdown* incluyendo *maximum drawdown* y período de recuperación, medidas de rentabilidad como *profit factor* y tasa de aciertos, y métricas de eficiencia de estrategia como la relación ganancia-dolor.

Las métricas específicas para criptomonedas incluyen el impacto del *funding rate* en el rendimiento de la estrategia, exposición al riesgo de *liquidation* y eficiencia del margen, rendimiento según diferentes condiciones de mercado, y viabilidad de costos de transacción después de una contabilidad realista.

Entrenamiento de validación cruzada y ajuste de hiper-parámetros

Se propone utilizar un enfoque de ventana deslizante, basado en una ventana de entrenamiento de tamaño fijo que avanza con cada paso; esto reduce los datos utilizados en el entrenamiento y descarta datos históricos potencialmente valiosos, pero obliga al modelo a adaptarse a las condiciones recientes del mercado.

El tamaño de la ventana deslizante puede ser un hiper-parámetro a ajustar, variando de 1 mes a 3 o 6 meses, dependiendo del conjunto de datos y las restricciones de los modelos.

Si para algunos modelos la combinación de hiper-parámetros posibles/relevantes es demasiado grande para realizar una ronda de entrenamiento para cada combinación, entonces basado en una revisión de literatura se debe determinar un subconjunto, utilizando valores recomendados que hayan tenido buenos resultados en las métricas de interés en este *framework*, y utilizando un conjunto de datos similar.

Selección de Modelos para Producción

El *framework* de evaluación multi-criterio clasifica principalmente por métricas financieras fuera de muestra como el *Sharpe ratio* y *maximum drawdown*, con una evaluación del rendimiento estadístico, una evaluación de la viabilidad operativa, y un análisis de la interpretabilidad y comprensión de modos de fallo.

Las **pruebas de robustez** involucran consistencia de rendimiento a través de múltiples períodos de validación, análisis de sensibilidad a variaciones de hiper-parámetros, pruebas de estrés bajo condiciones extremas de mercado, y análisis de capacidad con tamaños de posición crecientes.

Estrategias de Ensemble y Combinaciones

Una estrategia que rara vez se menciona en la literatura es crear una arquitectura de *ensemble* dinámica. Esto permite utilizar modelos que sobresalen en diferentes circunstancias, usando varios modelos que se desempeñan de manera diferente en varios factores, de modo que podemos entrenar una capa que aprenda a determinar qué salida (qué modelo) es la mejor para cada entrada.

Las opciones de métodos de *ensemble* incluyen promedio ponderado estático basado en rendimiento histórico, ponderación dinámica usando meta-modelos simples, *model stacking* con predicciones fuera de pliegue como *features*, y promedio bayesiano de modelos para combinación probabilística.

4. Análisis de Resultados

Comparación de Rendimiento de Modelos: Los resultados de validación del *testing walk-forward* mostrarán el rendimiento de cada modelo en diferentes condiciones de mercado. Este análisis determinará qué modelos demuestran retornos ajustados al riesgo consistentes e identificará sus fortalezas específicas en varios regímenes de mercado.

El análisis revelará qué *features* impulsan el poder predictivo en diferentes modelos. Los indicadores técnicos como medidas de volatilidad y osciladores de momento pueden mostrar importancia variable según las condiciones del mercado, mientras que las métricas *on-chain* como flujos de *exchange* podrían demostrar valor predictivo durante las transiciones de régimen.

El análisis de rendimiento del *ensemble* determinará si la combinación de modelos proporciona beneficios sobre los enfoques individuales. Esto incluye evaluar si ciertos modelos se complementan entre sí al funcionar bien en diferentes entornos de mercado o si un solo modelo supera consistentemente a los demás.

Basado en estos resultados, se tomarán decisiones sobre la selección de modelos para el despliegue en producción. Si ciertos modelos muestran modos de fallo inesperados o si surgen nuevos patrones en la

validación, puede ser necesario realizar **rondas adicionales de entrenamiento** con conjuntos de *features* o arquitecturas de modelo ajustadas.

El resultado clave de este análisis es determinar la configuración óptima para el *trading* en vivo, ya sea que involucre un modelo único, un *ensemble* estático o un *ensemble* dinámico que se ajuste según la detección del régimen de mercado.

5. Salida del Framework

En la etapa final es momento de determinar el modelo seleccionado o la configuración del *ensemble* lista para el despliegue, incluyendo todos los parámetros necesarios y componentes de preprocesamiento.

El *framework* también debe determinar las reglas específicas para generar señales de *trading* basadas en las predicciones del modelo, incluyendo umbrales de confianza y parámetros de dimensionamiento de posición derivados del rendimiento de validación.

Como mencionamos anteriormente, se recomienda desarrollar un *framework* de monitoreo. Incluyendo puntos de referencia de rendimiento y umbrales de alerta para detectar la degradación del modelo, más disparadores de reentrenamiento automatizados basados en la metodología de validación.

Finalmente, el *framework* debe incluir la documentación, especificaciones completas para la integración en un entorno de *trading* en vivo, incluyendo requisitos de datos y protocolos de ejecución.

Conclusiones

Esta propuesta ha delineado un *framework* integral para desarrollar un sistema de *trading* cuantitativo específicamente diseñado para mercados de *futures* de criptomonedas.

El *framework* establece un *pipeline* completo desde la recolección de datos hasta el despliegue del modelo, con particular énfasis en técnicas de validación robustas adecuadas para series temporales financieras. La metodología de validación *walk-forward* y las métricas de evaluación multidimensionales proporcionan una base sólida para evaluar el rendimiento del modelo en condiciones de mercado realistas.

El enfoque de modelación propuesto ofrece flexibilidad a través de múltiples arquitecturas candidatas y estrategias de *ensemble*, permitiendo que el sistema se adapte a diferentes regímenes de mercado y capitalice diversos patrones predictivos.

Las mejoras futuras podrían explorar fuentes de datos adicionales, métodos de *ensemble* avanzados y algoritmos de detección de régimen más sofisticados. El diseño modular del *framework* propuesto permite el desarrollo y refinamiento continuo a medida que nuevas técnicas y datos estén disponibles.