



دانشکده مهندسی کامپیوتر

درس الگوریتم‌های معاملاتی

پروژه پایانی

مدرس دکتر رضا انتظاری ملکی

طراحان علی صمدی فرد، مهدی قضاوی

تاریخ انتشار ۱۴۰۴/۱۰/۱۳

تاریخ تحویل ۱۴۰۴/۱۱/۶

❖ در رابطه با پروژه

- این پروژه تلفیقی از مطالب ارائه شده در فصل‌های Strategy Backtesting، Strategy Development و Portfolio Management است. از دانشجویان انتظار می‌رود با ترکیب مفاهیم این فصل‌ها یک سیستم جامع طراحی و ارزیابی کنند.
- پروژه گروهی بوده و هر گروه می‌تواند شامل ۲ نفر باشد. در پایان، ارائه شفاهی طی روز مقرر انجام خواهد شد.
- داده‌های موردنیاز را از منابع آزاد همچون CoinMarketCap برای انتخاب رمزارز و از Yahoo Finance برای دریافت داده OHLC استخراج کنید.
- به هیچ وجه از دیگران **کپی** نکنید. در صورت مشاهده تقلب و کپی، نمره هر دو طرف **صفر** در نظر گرفته می‌شود.
- مهلت ارسال پروژه تا پایان وقت ۶ بهمن ماه می‌باشد.
- ارائه‌های پروژه در تاریخ ۷ بهمن ماه به صورت مجازی برگزار خواهند شد.
- فایل پاسخ خود را به صورت **FP_StudentID1_StudentID2.zip** نام‌گذاری کرده و در کوئرا ارسال کنید.
- پروژه را با زبان برنامه نویسی پایتون و در قالب **Jupyter Notebook** پیاده‌سازی کنید.
- قبل از ارسال، فایل خود را حتماً یک بار کامل اجرا کنید.
- توجه کنید که نوشتن گزارش برای این پروژه **الزامی** می‌باشد. شما باید در این گزارش، کد خود به همراه توضیحات خواسته شده در هربخش پروژه را تشریح کنید. همچنین از مرتب بودن گزارش خود اطمینان حاصل کنید.

هدف

هدف این پروژه، طراحی و ارزیابی سه ربات معاملاتی بر بستر پنج رمزارز منتخب و سپس بهینه‌سازی پرتفوی حاصل با استفاده از مدل میانگین-واریانس (Mean-Variance Optimization یا MVO) است. دانشجویان باید نشان دهند که چگونه از آزمون‌های آماری، اندیکاتورهای و روش‌های بهینه‌سازی برای ساخت یک استراتژی قابل‌اتکا بهره می‌برند و در نهایت با در نظر گرفتن ریسک و بازده، وزن‌های بهینه پرتفوی و عملکرد آن را ارائه دهند.

تعاریف

- **استراتژی میانگین برگشتی (Mean-reversion Strategy):** این استراتژی به دنبال کسب سود از انحراف‌های موقتی قیمت از ارزش واقعی است؛ فرض می‌کند که قیمت‌ها پس از شوک‌های کوتاه‌مدت به سمت میانگین تاریخی خود بازمی‌گردند.
- **استراتژی مارتینگل (Martingale Strategy):** رویکردی جهت‌دار که پس از هر ضرر حجم معامله را افزایش می‌دهد و فرض می‌کند که در نهایت یک معامله سودآور تمام زیان‌های قبلی را جبران می‌کند؛
- **ربات شبکه‌ای (Grid Bot):** رباتی خودکار که با قرار دادن سفارش‌های خرید و فروش با فاصله ثابت در یک محدوده قیمتی تلاش می‌کند از نوسانات بازار سود ببرد. اگر بازار روند شدید یک‌طرفه داشته باشد، تمام سرمایه به معاملات زیان‌ده اختصاص می‌یابد و زیان قابل توجهی رخ می‌دهد.
- **بهینه‌سازی میانگین-واریانس (MVO):** چارچوبی کلاسیک که به دنبال یافتن وزنی برای دارایی‌هاست تا برای مقدار مشخصی از بازده مورد انتظار، واریانس (ریسک) پرتفو حداقل شود. فرض می‌شود که تنها میانگین و واریانس بازده‌ها اهمیت دارند و کوواریانس بین دارایی‌ها قابل‌تخمین است. این روش پایه بسیاری از مدل‌های بهینه‌سازی مدرن است و برای پروژه حاضر به دلیل سادگی و آموزش‌پذیری مناسب انتخاب شده است.

بخش اول — انتخاب دارایی و جمع‌آوری داده

از میان ۵۰ رمزارز برتر بر اساس ارزش بازار، ۵ رمزارز را برای تحلیل انتخاب کنید. انتخاب باید براساس نقدینگی، حجم معاملات و تنوع بین کلاس‌های مختلف (مثلاً بیت‌کوین، اتریوم، یک آلت‌کوین مبتنی بر دیفای و ...) باشد.

داده‌های روزانه‌ی این رمزارزها را در بازه زمانی ۱ ژانویه ۲۰۲۴ تا ۱ ژانویه ۲۰۲۶ دریافت کنید. داده‌ها باید شامل قیمت باز، بسته، بیشینه، کمینه و حجم معاملات باشند.

برای هر دارایی، آزمون ADF را اجرا کنید تا سری‌های مانا یا غیرمانا را تشخیص دهید. نتیجه آزمون (p -value) را گزارش کنید و توضیح دهید کدام رمزارزها دارای روند یا نوسان پایدار هستند.

نمایه هرست (Hurst Exponent) و نیمه عمر هر سری قیمتی را محاسبه کنید و تفسیر کنید این شاخص‌ها چه اطلاعاتی در مورد رفتار روندی/بازگشتی رمزارزها می‌دهد. از کتابخانه‌های آماده برای محاسبه این دو شاخص استفاده نکنید؛ خودتان توابع لازم را بنویسید.

نمودار Candlestick و نمودار قیمتِ عادی شده هر رمزارز را رسم کنید تا بینش بصری نسبت به رفتار آنها پیدا کنید.

نکته مهم: این پنج رمزارز انتخاب شده در این بخش، در تمام بخش‌های بعدی پروژه (پیاده‌سازی ربات‌ها و بهینه‌سازی پرتفوی) مورد استفاده قرار خواهند گرفت. بنابراین در بخش‌های بعدی نیازی به انتخاب رمزارزهای جدید از میان ۵۰ رمزارز برتر نیست و تحلیل‌ها تنها بر همین پنج دارایی انجام می‌شود.

بخش دوم — پیاده‌سازی ربات‌های معاملاتی

در این بخش سه ربات متفاوت را پیاده‌سازی می‌کنید و عملکرد آن‌ها را روی همان پنج رمزارز انتخاب شده در بخش اول مورد بررسی قرار می‌دهید. در این بخش، بررسی می‌شود که هر ربات با این دارایی‌ها چگونه عمل می‌کند. در همه ربات‌ها فرض کنید کارمزد معاملات ناچیز ولی غیرصفر است و از یک حساب مجازی با موجودی اولیه مشخص (مثلاً ۱۰۰'۰۰۰ دلار) استفاده می‌کنید.

بخش دوم-۱: ربات میانگین‌برگشتی (Mean-Reversion Bot)

برای هر یک از ۵ رمزارز منتخب در بخش اول، در بازه زمانی ۱ ژانویه ۲۰۲۴ تا ۱ ژانویه ۲۰۲۶ (همان بازه جمع‌آوری داده بخش اول) آزمون ADF را در تایم‌فریم روزانه انجام دهید تا مانا بودن سری قیمت مشخص شود و در جدولی $p\text{-value}$ آن‌ها را گزارش کنید. اگر هیچ‌یک از سری‌ها مانا نبودند، این موضوع را صریحاً ذکر کنید.

برای رمزارزهای مانا، نیمه عمر و نمایه هرست را محاسبه کنید؛ رمزارزها را بر حسب نیمه عمر به صورت صعودی مرتب کرده و نمودارهای کندل‌استیک و Hurst exponent آن‌ها را رسم کنید.

از میان سری‌های قیمتی نامانای همین پنج رمزارز، حداقل ۱۰ ترکیب سه‌تایی غیرمانا پیدا کنید که ترکیب خطی آن‌ها به کمک آزمون یوهانسن (Johansen) مانا می‌شود؛ اگر تعداد ترکیبات سه‌تایی به ۱۰ نرسید، ترکیبات چهارتایی و پنج‌تایی را نیز امتحان کنید تا نهایتاً ۱۰ ترکیب خطی مانا به دست آید. سپس از میان این سری‌های حاصل، ۵ سری با نیمه عمر کمتر را انتخاب کنید و برای هر کدام نمودار رمزارزهای اصلی و نمودار اسپرد (spread) را به صورت کندل‌استیک رسم کنید.

برای هر یک از ۵ سری منتخب، استراتژی زیر را پیاده‌سازی کنید:

۱. طول look-back را دو برابر نیمه‌عمر سری قرار دهید (اگر نیمه‌عمر کمتر از ۳۰ روز بود، اولین مضرب صحیح نیمه‌عمر که $30 \leq$ باشد انتخاب شود).
۲. EMA و انحراف‌معیار (STD) را با look-back محاسبه کنید و هر سه مقدار STD، EMA و half-life را چاپ نمایید.
۳. ورود به Short: وقتی قیمت high از EMA بیشتر از یک STD بالاتر بود، نیمی از سرمایه اختصاص‌یافته را در پایان کندل بفروشید؛ اگر قیمت به بیش از دو STD بالاتر رسید، نصف دیگر سرمایه را بفروشید.
۴. ورود به Long: وقتی قیمت low از EMA بیشتر از یک STD پایین‌تر بود، نیمی از سرمایه را بخرید؛ اگر به بیش از دو STD پایین‌تر رسید، نصف دیگر سرمایه را بخرید.
۵. خروج از پوزیشن: وقتی قیمت به EMA برگشت، تمام موقعیت‌های باز را ببندید.
۶. استراتژی را بر روی هر یک از ۵ اسپرد در همان بازه ۱ ژانویه ۲۰۲۴ تا ۱ ژانویه ۲۰۲۶ اجرا کنید تا داده‌های هر ربات در نهایت ابعاد یکسانی داشته باشد؛ حداکثر افت سرمایه (Max Drawdown)، نسبت شارپ، منحنی Equity و بازده کل را محاسبه و گزارش کنید.
۷. برای حذف بایاس نگاه به آینده، همانند تمرین HW4، بک‌تست را به‌صورت دوره‌های متوالی شش‌ماهه انجام دهید: بازه داده اصلی (۱ ژانویه ۲۰۲۴ تا ۱ ژانویه ۲۰۲۶) را به زیربازه‌های شش‌ماهه تقسیم کنید. در هر تکرار، ابتدا داده‌های شش‌ماهه گذشته را برای پیدا کردن سری‌های مانا و تعیین پارامترهای ربات استفاده کنید و سپس یک ماه آینده (ماه هفتم) را معامله کنید؛ پس از پایان هر ماه، پارامترها را با استفاده از داده‌های شش ماه اخیر به‌روز کرده و استراتژی را برای ماه بعد اجرا کنید تا انتهای بازه. این روش ضمانت می‌کند که همه ربات‌ها داده‌هایی با طول یکسان داشته باشند و بایاس نگاه به آینده کاهش یابد.
۸. نتایج حاصل از این ربات را در جدولی خلاصه کنید و به‌خصوص نشان دهید که کدام ترکیب‌ها سودآورتر بوده‌اند و چه وزن‌هایی ممکن است برای پرتفوی کلی پیشنهاد شود.

بخش دوم-۲: ربات شبکه مارتینگل (Grid-Martingale Bot)

این ربات مطابق با تمرین سری پنجم پیاده‌سازی می‌شود:

۱. برای پنج رمزارز منتخب، بررسی کنید کدام‌یک در بازه ۱ ژانویه ۲۰۲۴ تا ۱ ژانویه ۲۰۲۶ در یک کانال صعودی یا نوسانی قوی قرار دارند (یعنی قیمت بالاتر از EMA200 باشد). در صورت عدم انطباق برخی دارایی‌ها با این شرط، همچنان آن‌ها را نگه دارید و در گزارش تحلیل کنید که عدم تطابق چه اثری بر عملکرد ربات دارد. این بازه زمانی با بازه سایر ربات‌ها یکسان است تا در نهایت داده‌های بازده ابعاد مشابهی داشته باشد.

۲. با استفاده از شاخص ATR، نوسانات اخیر بازار را برای تعیین فاصله بین خطوط شبکه (Grid Spacing) محاسبه کنید.
۳. تابعی برای تعیین سطوح شبکه بنویسید که بر اساس قیمت فعلی ۱۰ سطح خرید (Buy Orders) در زیر قیمت با فاصله‌های ثابت ایجاد کند.
۴. تابعی برای حجم تصاعدی (Martingale Sizing) بنویسید: حجم هر مرحله دو برابر مرحله قبلی باشد.
۵. **Average Price Calculator**: میانگین قیمت خرید کل دارایی‌های موجود در شبکه را در هر لحظه محاسبه کنید.
۶. قوانین ورود و مدیریت شبکه: ربات را طوری طراحی کنید که فقط در جهت صعودی وارد معامله شود؛ ابتدا در قیمت بازار پله صفر ایجاد کنید و سپس اگر قیمت پایین آمد و به خطوط پایین‌تر رسید، سفارش‌های خرید با حجم‌های مارتینگل فعال شوند.
۷. قوانین خروج و سودآوری: هنگامی که قیمت به $2/5\%$ بالاتر از میانگین کل شبکه رسید، تمام پوزیشن‌ها بسته و شبکه ریست شود؛ اگر قیمت از پایین‌ترین سطح (پله دهم) عبور کرد، همه پوزیشن‌ها با ضرر بسته و فعالیت ربات متوقف شود.
۸. این استراتژی را در بازه یکسان ۱ ژانویه ۲۰۲۴ تا ۱ ژانویه ۲۰۲۶ به صورت دوره‌های یک‌ماهه و برای تایم‌فریم‌های چهار ساعته، یک ساعته و روزانه بک‌تست کنید؛ ابتدا خروجی‌های هر ماه را جداگانه و سپس میانگین‌گیری کلی را گزارش کنید.
۹. بازده، نسبت شارپ و حداکثر افت سرمایه را برای هر رمزارز و هر تایم‌فریم محاسبه و مقایسه کنید. نقاط قوت و ضعف این ربات را تحلیل کنید و توضیح دهید چگونه یک روند شدید می‌تواند سرمایه را به پوزیشن‌های زیان‌ده اختصاص دهد.

بخش دوم-۳: ربات دنبال‌کننده روند با ADX (ADX Trend Bot)

این ربات بر مبنای تمرین سری سوم و اندیکاتور ADX طراحی می‌شود:

۱. برای هر یک از پنج رمزارز منتخب، مقادیر Directional Movement (+DM و -DM)، True Range (TR) و سپس Average True Range (ATR) را طبق فرمول‌های ارائه شده در تمرین محاسبه کنید. سپس Directional Indicators (+DI و -DI) و ADX را با دوره زمانی مناسب (مثلاً ۱۴ روز) به دست آورید.
۲. یک آستانه ثابت θ (معمولاً ۲۵) برای تشخیص بازار رونددار از بازار رنج انتخاب کنید. اگر ADX بزرگ‌تر از θ باشد، بازار رونددار در نظر گرفته می‌شود؛ در غیر این صورت بازار رنج است.

۳. قوانین ورود به پوزیشن بلند (Long): زمانی که ADX بالاتر از θ باشد و DI+ از پایین به بالا DI- را قطع کند، پوزیشن خرید به اندازه سرمایه اختصاص یافته به رمزارز باز کنید.
۴. قوانین ورود به پوزیشن کوتاه (Short): زمانی که ADX بالاتر از θ باشد و DI- از پایین به بالا DI+ را قطع کند، پوزیشن فروش باز کنید. اگر ADX به زیر θ سقوط کند، هیچ پوزیشن جدیدی باز نکنید و پوزیشن‌های قبلی را ببندید.
۵. برای کنترل ریسک، می‌توانید از Stop Loss (مثلاً ۲٪ زیر قیمت ورود برای پوزیشن‌های خرید و ۲٪ بالاتر برای پوزیشن‌های فروش) و Take Profit (مثلاً ۵٪ سود) استفاده کنید.
۶. این استراتژی را در بازه زمانی انتخاب شده (همان بازه بخش اول) برای هر رمزارز بک‌تست کنید و معیارهای عملکرد شامل بازده، نسبت شارپ، نسبت سورتینو و حداکثر افت سرمایه را گزارش دهید.
۷. نتایج را در قالب جدول و نمودارهای مناسب ارائه و تحلیل کنید. تأثیر انتخاب آستانه θ و دوره محاسبه ADX بر عملکرد را بررسی کنید و بهترین مقادیر را توضیح دهید.

بخش سوم — ارزیابی و مقایسه عملکرد ربات‌ها

۱. نتایج بک‌تست هر ربات را برای هر رمزارز در قالب جدول و نمودار جمع‌بندی کنید؛ این جداول باید شامل بازده متوسط، انحراف معیار بازده‌ها، نسبت شارپ، نسبت سورتینو، حداکثر افت سرمایه و طول متوسط معاملات باشد.
۲. ربات‌ها را از نظر پایداری عملکرد، ریسک به پاداش و حساسیت به پارامترها مقایسه کنید و توضیح دهید که کدام ربات در شرایط مختلف بازار (رونددار، نوسانی، بی‌رسم) بهتر عمل می‌کند.
۳. نقاط ضعف هر ربات را تحلیل کنید؛ مثلاً ربات مارتینگل چگونه ممکن است در روند شدید با زیان سنگین مواجه شود، یا ربات میانگین‌برگشتی در بازارهای بدون بازگشت کارایی نداشته باشد.
۴. با استفاده از نتایج این بخش، بازده مورد انتظار هر رمزارز را برای مرحله بعد (بهینه‌سازی پرتفوی) برآورد کنید.

بخش چهارم — بهینه‌سازی پرتفوی با مدل میانگین-واریانس (MVO)

۱. در این بخش، وظیفه اصلی شما یافتن وزن‌های بهینه‌ی پرتفوی برای همان پنج رمزارز انتخاب شده در بخش اول است. تمام داده‌های مورد استفاده در این بخش باید از نتایج بخش سوم (بک‌تست سه ربات روی همین دارایی‌ها و بازه زمانی مشترک) استخراج شود. خروجی وزن‌های نهایی باید به صورت جداگانه در گزارش و کد نمایش داده و مشخص شود.

۲. بازده مورد انتظار هر یک از پنج رمزارز منتخب را بر اساس نتایج بک تست هر سه ربات (mean-reversion, مارتنینگل و ADX) در بخش سوم محاسبه کنید؛ می‌توانید از بازده ماهانه یا بازده کل استفاده کنید و در گزارش توضیح دهید چرا این معیار را انتخاب کرده‌اید. توجه کنید که همه بازده‌ها باید بر پایه همان بازه زمانی مشترک (۱ ژانویه ۲۰۲۴ تا ۱ ژانویه ۲۰۲۶) باشند تا در محاسبات پرتفوی ابعاد یکسانی داشته باشند.
۳. ماتریس کوواریانس بازده‌های ماهانه همین پنج رمزارز را بر اساس همان سری بازده‌هایی که در گام قبل محاسبه کرده‌اید تخمین بزنید. برای ساده‌سازی می‌توانید از دوره زمانی یک‌ساله یا تمام بازه مطالعه استفاده کنید، اما این بازه باید با بازه تحلیل ربات‌ها هماهنگ باشد.
۴. مسئله بهینه‌سازی میانگین-واریانس را به فرم زیر بنویسید: کمینه کردن واریانس پرتفو ($\sigma_p^2 = w^T \Sigma w$) با فرض اینکه مجموع وزن‌ها برابر ۱ است ($1^T w = 1$) و هیچ وزن منفی (فروش استقراضی) مجاز نیست، همچنین بازده مورد انتظار پرتفو ($\mu_p = \mu^T w$) باید حداقل برابر با مقدار هدف تعیین‌شده توسط تیم شما باشد. این مسئله هم‌ارز با بیشینه کردن بازده برای سطح مشخصی از ریسک است.
۵. برای حل این مسئله، برای سادگی کار، اجازه دارید علاوه بر پیاده‌سازی دستی فرمول‌ها، از کتابخانه‌های آماده پایتون مانند cvxpy یا PyPortfolioOpt نیز استفاده کنید. این کتابخانه‌ها امکان حل مسائل کمینه‌سازی واریانس، بیشینه کردن نسبت شارپ یا یافتن پرتفو با بازده هدف را فراهم می‌کنند. با این حال توصیه می‌شود ابتدا مفهوم مسئله و ساختار تابع هدف و قیود را درک کنید.
۶. در صورتی که می‌خواهید خودتان فرمول‌های بهینه‌سازی میانگین-واریانس را بر اساس ماتریس کوواریانس Σ ، بردار بازده‌های مورد انتظار μ و بردار یک‌ها $\mathbf{1}$ پیاده‌سازی کنید، می‌توانید از کتابخانه‌های پایه‌ای پایتون مثل numpy.linalg برای معکوس کردن ماتریس‌ها و حل دستگاه‌های خطی بهره ببرید، اما باید مراحل زیر را انجام دهید:
۷. برای یافتن پرتفوی کمینه واریانس بدون محدودیت بازده، از [فرمول](#) $w_{GMV} = \frac{\Sigma^{-1} \mathbf{1}}{\mathbf{1}^T \Sigma^{-1} \mathbf{1}}$ استفاده کنید. این وزن‌ها کمینه‌ترین واریانس را بین تمام پرتفوهایی دارند که وزن‌ها جمعاً ۱ است.
۸. برای یافتن پرتفوی با بازده هدف $\mu_{p,0}$ ، مسئله کمینه‌سازی واریانس را با دو قید $1^T w = 1$ و $\mu^T w = \mu_{p,0}$ حل کنید. از طریق روش ضرب لاگرانژ، سیستم خطی
$$\begin{bmatrix} 2\Sigma & \mu & \mathbf{1} \\ \mu^T & 0 & 0 \\ \mathbf{1}^T & 0 & 0 \end{bmatrix} \begin{bmatrix} w \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 0 \\ \mu_{p,0} \\ 1 \end{bmatrix}$$
 به دست می‌آید که با معکوس کردن ماتریس سمت چپ، وزن‌ها w به دست می‌آیند. این روش را در کد خود پیاده‌سازی کنید.
۹. می‌توانید برای [پرتفوی بیشینه نسبت شارپ \(تنگنایی\)](#) نیز از فرمول $w_{tan} = \frac{\Sigma^{-1}(\mu - r_f \mathbf{1})}{\mathbf{1}^T \Sigma^{-1}(\mu - r_f \mathbf{1})}$ استفاده کنید که نرخ بدون ریسک r_f را در نظر می‌گیرد.

۱۰. وزن‌های محاسبه‌شده برای هر روش (پرتفوی کمینه واریانس، پرتفوی با بازده هدف و پرتفو با بیشینه نسبت شارپ) را در جدول و نمودار نمایش دهید و تحلیل کنید که چرا برخی دارایی‌ها وزن بیشتری دارند و برخی کمتر. همچنین اگر از کتابخانه‌های آماده استفاده کرده‌اید، کد و توضیح نحوه استفاده از توابع آن را ارائه دهید تا مشخص باشد چگونه وزن‌ها محاسبه شده‌اند.

۱۱. عملکرد پرتفو را با استفاده از شاخص‌هایی مانند بازده میانگین، انحراف معیار، نسبت شارپ، حداکثر افت سرمایه و نسبت بازده به ریسک ارزیابی کنید. علاوه بر روش‌های بهینه‌سازی مذکور، پرتفوهایی مساوی وزن و پرتفو با بیشترین بازده (حداکثر بازده بدون محدودیت ریسک) را نیز محاسبه و با پرتفوهایی بهینه مقایسه کنید.

۱۲. تحلیل حساسیت انجام دهید: نشان دهید که اگر بازده‌های مورد انتظار یا ماتریس کوواریانس کمی تغییر کنند، وزن‌های بهینه چگونه تغییر می‌کنند. این تحلیل به شما کمک می‌کند درک کنید که نتایج چقدر به داده‌های ورودی حساس‌اند.

۱۳. در پایان این بخش، وزن‌های بهینه‌ی نهایی و عملکرد پرتفو را به صورت واضح به عنوان خروجی اصلی پروژه گزارش دهید.

بخش پنجم — مدیریت ریسک و حذف بایاس نگاه به آینده

۱. برای جلوگیری از بایاس نگاه به آینده (look-ahead bias)، در هر بخش بک‌تست از رویکرد پیمایشی استفاده کنید: داده‌های گذشته (مثلاً شش ماه) را برای انتخاب پارامترهای ربات‌ها و تخمین بازده و کوواریانس استفاده کنید و سپس یک ماه بعدی را معامله کنید. پس از پایان هر ماه، پارامترها و تخمین‌ها را با استفاده از آخرین داده‌ها به‌روز کنید و این روند را تا پایان بازه مورد مطالعه ادامه دهید.

۲. اگر در پایان ماه موقعیت بازی باقی نماند، فوراً به‌روزرسانی پارامترها را انجام دهید؛ در غیر این صورت اجازه دهید استراتژی موقعیت را ببندد و سپس بازآرایی کنید.

۳. نتایج این فرآیند را به صورت جدول زمانی ارائه کنید تا نشان دهید چگونه وزن‌های پرتفوی و عملکرد آن در طول زمان تغییر می‌کند.

نحوه تحویل و نکات پایانی

- پروژه باید با زبان پایتون و در قالب Jupyter Notebook پیاده‌سازی شود. پیش از ارسال، نوت‌بوک را به طور کامل اجرا کنید.
- فایل نوت‌بوک و گزارش پروژه را در قالب یک فایل zip ارسال کنید. گزارش باید شامل توضیح نحوه پیاده‌سازی، تحلیل نتایج، جداول و نمودارها، پاسخ به سؤالات و تحلیل‌های خواسته‌شده باشد.
- نام اعضای گروه و شماره دانشجویی را در ابتدای گزارش ذکر کنید.

با آرزوی موفقیت