

به نام خداوند

گزارش پروژه پایان ترم
اقتصادسنجی پیشرفته

علیرضا جمالی
alireza.jamalie@ut.ac.ir

* گزارش توضیح کدهای هر قسمت از پروژه، در پوشه مربوط به همان قسمت وجود دارد.
* تمامی کدهای تمرین ها و پروژه درس اقتصادسنجی به همراه تاریخچه آنها در [این مخزن گیت](#) قابل مشاهده است.

دانشکده مدیریت دانشگاه تهران
زمستان ۱۴۰۱

فهرست

3.....	پروژه ۱ - انتقال کد معاملات جفتی از متلب به پایتون
3	راهنمای فایل ها
3	داده ورودی
3	روش اجرا
4	خروجی
6.....	پروژه ۲ - مقایسه <i>VECM</i> با شبکه های عصبی
6	راهنمای فایل ها
6	داده ورودی
6	روش اجرا
7	محدودیت ها
7	خروجی
9.....	پروژه ۳ - آزمون خطی بودن ارزش های دیجیتال
9	راهنمای فایل ها
9	ورودی ها
9	روش اجرا
9	خروجی ها
11	پروژه ۴ - یافتن توزیع <i>Kernel Density</i> مدل <i>ECM</i>
11.....	راهنمای فایل ها
11.....	ورودی ها
11.....	روش اجرا
12.....	خروجی ها

پروژه ۱ - انتقال کد معاملات جفتی از متلب به پایتون

در این پروژه، کد mohPairesStocks.m به زبان پایتون نوشته می‌شود.

راهنمای فایل‌ها

توضیح	درس فایل
گزارش توضیح کدها به زبان انگلیسی	./01_pair_trading/02_paired_all.pdf
فایل ژوپیتر نوت‌بوک نهایی همراه با توضیحات	./01_pair_trading/02_paired_all.ipynb
فایل پایتون نهایی همراه با توضیحات	./01_pair_trading/02_paired_all.py
پوشه حاوی نمودارهای سهام هم‌انباشته	./01_pair_trading/plots/{inex_name}
فایل اکسل لیست سهام هم‌انباشته	./01_pair_trading/pairs_2023-01-15.xlsx

داده ورودی

- ۳۰ سهم موجود در شاخص Dow Jones (منبع: یاهوفایننس)
- ۴۰ سهم موجود در شاخص Dax (منبع: یاهوفایننس)
- ۴۰ سهم موجود در شاخص CAC 40 (منبع: یاهوفایننس)
- ۵۰ سهم موجود در شاخص پنجاه شرکت بزرگ بورس تهران (منبع: TSE)

- گام سری‌های قیمتی روزانه است.
- بازه زمانی برای سری‌های قیمتی، ۵۲۱ روز است که ۱۲۶ روز آخر به عنوان بازه آزمون انتخاب شده است.
- از لگاریتم قیمت‌ها برای محاسبات مربوط به تست‌های هم‌انباشتگی و رسم نمودارها استفاده می‌شود.

روش اجرا

به ازای هر جفت سهم موجود در هر شاخص، آزمون هم‌انباشتگی یوهانسون اجرا می‌شود. در صورتی که دو سهم هم‌انباشته باشند، الگوریتم معاملات جفتی اجرا می‌شود و پس از محاسبه بازده این الگوریتم، نمودار مربوطه رسم و در نهایت ذخیره می‌شود.

- تعداد تست‌های هم‌انباشتگی انجام شده (مجموع تمام ترکیب‌های دوتایی سهام موجود در هر شاخص):

$$\binom{30}{2} + \binom{40}{2} + \binom{40}{2} + \binom{50}{2} = 435 + 780 + 780 + 1225 = 3220$$

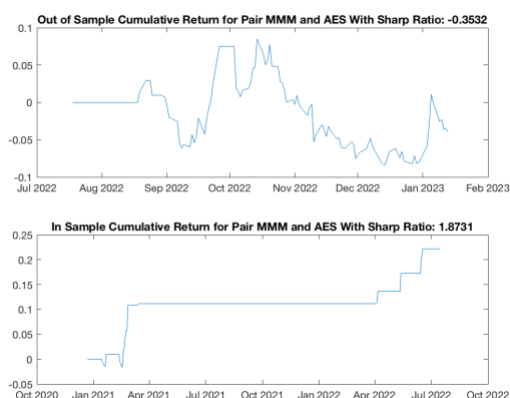
- برخی از توابع و کلاس‌های استفاده شده مهم:

توضیح	تابع یا کلاس
تابعی است که برای دریافت لیست سهام موجود در هر شاخص نوشته‌ام. برای سه شاخص خارجی، از پکیج PyTickerSymbols استفاده می‌شود و برای لیست سهام ایرانی، از وبسایت فیپیران مستقیماً خوانده می‌شود.	./01_pair_trading/helper_functions.stock_list
تابعی است که با دریافت لیستی از سهام، دیتافریمی از قیمت تعدیل‌شده آن‌ها را به عنوان خروجی می‌دهد.	./01_pair_trading/helper_functions.stock_prices
این تابع، تعداد وقفه مناسب را برای مدل vecm پیشنهاد می‌دهد.	statsmodels.tsa.vector_ar.vecm.select_order

statsmodels.tsa.vector_ar.vecm.select_coint_rank	این تابع، مرتبه هم‌انباشتگی را برای دو سری قیمتی محاسبه می‌کند. در درون این تابع، تست یوهانسون انجام می‌گیرد. از جمله موارد خروجی، آماره و مقادیر بحرانی تست یوهانسون است.
statsmodels.tsa.vector_ar.vecm.VECM	این کلاس با دریافت دو سری زمانی هم‌انباشته، یک آبجکت VECM به عنوان خروجی می‌دهد. با فراخوانی متد fit می‌توان پارامترهای مدل VECM را محاسبه کرد.

خروجی

خروجی کد پایتون بسیار به کد متلب شبیه است. برای مثال، در زیر نمودارهای نهایی دو سهم هم‌انباشته در خروجی متلب (چپ) و خروجی پایتون (راست) آورده شده است.



شکل 2 بازده الگوریتم معاملات جفتی محاسبه‌شده توسط کد متلب



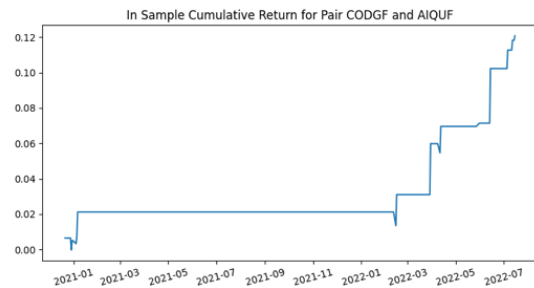
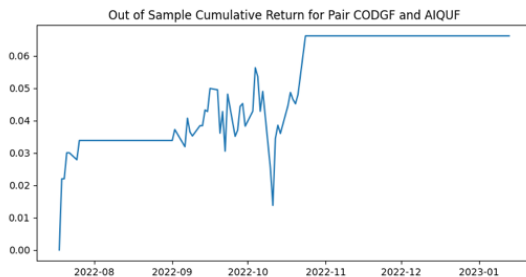
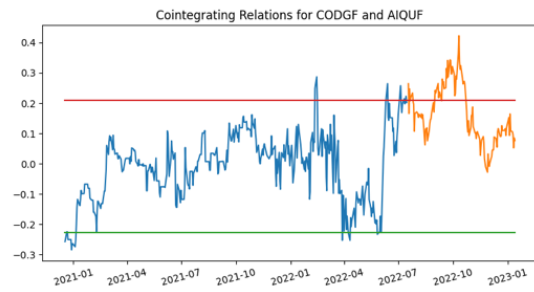
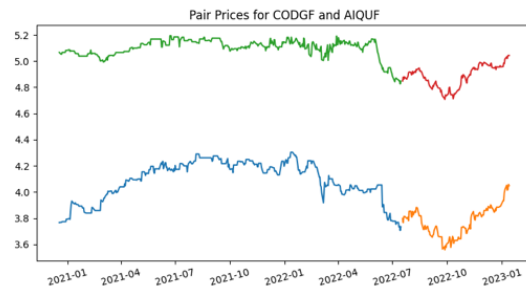
شکل 1 بازده الگوریتم معاملات جفتی محاسبه‌شده توسط کد پایتون

• تعداد جفت سهام هم‌انباشته:

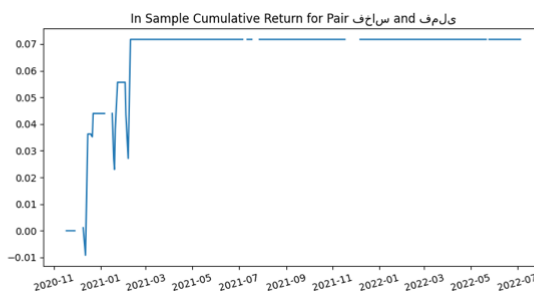
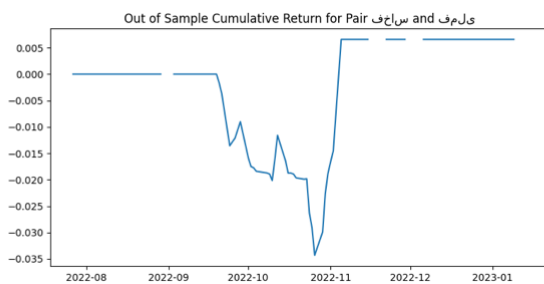
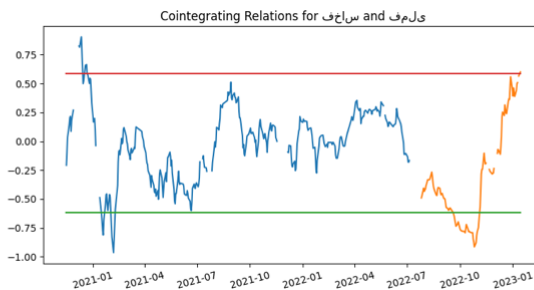
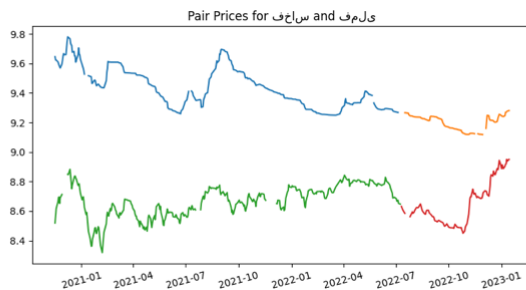
شاخص	تعداد
Dow Jones	۶۶
CAC 40	۱۴۵
Dax	۱۳۴
Teh50	۱۶۴
جمع	۵۰۹

- ۱۶ درصد تست‌های هم‌انباشتگی این پروژه مثبت شدند. در تمرین آخر کلاسی، تست هم‌انباشتگی بر روی ۱۰۰۰ سری گام تصادفی با ۱۰۰ مشاهده انجام شد که فقط ۵ درصد این سری‌ها با هم‌انباشته بودند.
- لیست سهام هم‌انباشته هر شاخص در فایل اکسل 01_pair_trading/pairs_2023-01-15.xlsx در شیت‌های مجزا موجود است.
- به ازای هر جفت سهام هم‌انباشته، یک تصویر حاوی چهار نمودار در پوشه 01_pair_trading/plots/{index} ذخیره شده است.
- از آنجایی که تعداد تصاویر و نمودارها بسیار زیاد است (۵۰۹ تا)، در زیر، دو تصویر از میان آن‌ها برای نمونه انتخاب شده است.

● نمودار CODGF و AIQUF در شاخص CAC 40:



● نمودار فخاص و فملى در شاخص Teh50:



پروژه ۲ - مقایسه VECM با شبکه‌های عصبی

در این پروژه، خطای پیش‌بینی مدل VECM با مدل MPLRegression مقایسه می‌شود.

راهنمای فایل‌ها

توضیح	درس فایل
گزارش توضیح کد پیش‌بینی مدل VECM به زبان انگلیسی	./02_vecm_vs_neurals/01_vecm.pdf
گزارش توضیح کد پیش‌بینی مدل شبکه‌های عصبی به زبان انگلیسی	./02_vecm_vs_neurals/02_nn_multioutput.pdf
فایل ژوپیتِر نوت‌بوک کد پیش‌بینی مدل VECM همراه با توضیحات	./02_vecm_vs_neurals/01_vecm.ipynb
فایل ژوپیتِر نوت‌بوک کد پیش‌بینی مدل شبکه‌های عصبی همراه با توضیحات	./02_vecm_vs_neurals/02_nn_multioutput.ipynb
فایل پایتون کد پیش‌بینی مدل VECM همراه با توضیحات	./02_vecm_vs_neurals/01_vecm.py
فایل پایتون کد پیش‌بینی مدل شبکه‌های عصبی همراه با توضیحات	./02_vecm_vs_neurals/02_nn_multioutput.py
فایل اکسل خطاهای پیش‌بینی مدل VECM	./02_vecm_vs_neurals/vecm_mape.xlsx
فایل اکسل خطاهای پیش‌بینی مدل شبکه‌های عصبی	./02_vecm_vs_neurals/nn_mape.xlsx
پوشه حاوی نمودارهای پیش‌بینی مدل VECM	./02_vecm_vs_neurals/preds_vecm/{index_name}
پوشه حاوی نمودارهای پیش‌بینی مدل شبکه‌های عصبی	./02_vecm_vs_neurals/preds_nn/{index_name}

داده ورودی

- ۵۰۹ سهمی که در تمرین قبل هم‌انباشته تشخیص داده شدند.
- گام سری‌های قیمتی روزانه است.
- بازه زمانی برای سری‌های قیمتی، ۵۲۱ روز است که ۱۲۶ روز آخر به عنوان بازه آزمون انتخاب شده است.
- از لگاریتم قیمت‌ها برای محاسبات مربوط به تست‌های هم‌انباشتگی و رسم نمودارها استفاده می‌شود.

روش اجرا

ابتدا پیش‌بینی تک‌گام و چندگام مدل VECM برای همه جفت سهام هم‌انباشته صورت می‌گیرد (۵۰۹ جفت سهم) و خطای پیش‌بینی در فایل اکسل ذخیره می‌شود. سپس همین کار برای پیش‌بینی‌های مدل MLPRegressor نیز انجام می‌شود. در نهایت، با رسم هیستوگرام خطاهای هر دو مدل، به مشاهده تفاوت خطاها می‌پردازیم.

- برای پیش‌بینی‌ها از مدل‌های multioutput استفاده می‌شود.
- پیش‌بینی‌های چندگام، همگی دینامیک هستند.
- از شاخص MAPE برای مقایسه کارایی این دو مدل استفاده شده است. اصلی‌ترین دلیل این انتخاب، مستقل بودن از مقیاس این شاخص است. زیرا می‌خواهیم از طریق رسم هیستوگرام، خطای ۵۰۹ جفت سری زمانی را که مقادیر مختلفی دارند، با هم مقایسه کنیم و برای این کار، معیارمان باید مستقل از مقیاس باشد.
- ۵۰۹ جفت سهم هم‌انباشته از تمرین قبل داریم. به ازای هر جفت، چهار پیش‌بینی صورت می‌گیرد: تک‌گام سهم اول، چندگام سهم اول، تک‌گام سهم دوم، چندگام سهم دوم. بنابراین تعداد مقادیر خطای محاسبه شده باید $509 \times 4 = 2036$ باشد.

- برخی از توابع و کلاس‌های استفاده شده مهم:

توضیح	تابع یا کلاس
تابعی است که برای دریافت لیست سهام موجود در هر شاخص نوشته‌ام. برای سه شاخص خارجی، از پکیج PyTickerSymbols استفاده می‌شود و برای لیست سهام ایرانی، از وبسایت فیپیران مستقیماً خوانده می‌شود.	<code>./02_vecm_vs_neurals/helper_functions.stock_list</code>
تابعی است که با دریافت لیستی از سهام، دیتافریمی از قیمت تعدیل شده آن‌ها را به عنوان خروجی می‌دهد.	<code>./02_vecm_vs_neurals/helper_functions.stock_prices</code>
این کلاس با دریافت دو سری زمانی هم‌انباشته، یک آبجکت VECM به عنوان خروجی می‌دهد. با فراخوانی متد <code>fit</code> می‌توان پارامترهای مدل VECM را محاسبه کرد.	<code>statsmodels.tsa.vector_ar.vecm.VECM</code>
این تابع، مدل‌های پکیج <code>sklearn</code> را تبدیل به مدل‌های چندخروجی می‌کند.	<code>sklearn.multioutput.MultiOutputRegressor</code>
این تابع، از مدل MLP برای رگرسیون سری زمانی استفاده می‌کند.	<code>sklearn.neural_network.MLPRegressor</code>

محدودیت‌ها

- در بخشی از پیش‌بینی‌های تک‌گام، هنگامی که دیتای جدید به داده آموزش اضافه می‌شود، دیگر سری‌ها هم‌انباشته نیستند. خطای این جفت سهام، برای مدل VECM در نظر گرفته نشده‌اند.
- همچنین در برخی از پیش‌بینی‌های تک‌گام، هنگامی که داده جدید به داده آموزش اضافه می‌شود، ممکن است داده جدید مقدار `null` داشته باشد. مدل `MLPRegressor` این مشکل را خودبه‌خود حل می‌کند اما مدل VECM خطا می‌دهد. بنابراین خطای این جفت سهام، برای مدل VECM در نظر گرفته نشده‌اند.
- آموزش مدل `MLPRegressor` بسیار سخت بود. در واقع یافتن پارامترهای بهینه که در مجموع ۲۰۳۶ پیش‌بینی، بهترین نتیجه را ارائه دهد طاقت‌فرسا بود. خروجی‌های موجود این مدل هم‌چنان نیاز به بهبود دارند و راضی‌کننده نیستند.

خروجی

در مجموع ۲۰۳۶ پیش‌بینی انجام شده است (چندخروجی به صورت تک‌گام و چندگام). به ازای هر جفت سهام هم‌انباشته، یک نمودار ذخیره شده است. نمودارهای مربوط به پیش‌بینی‌های VECM در پوشه `./02_vecm_vs_neurals/preds_vecm` و نمودارهای مدل شبکه‌های عصبی در پوشه `./02_vecm_vs_neurals/preds_nn` ذخیره شده است.

هم‌چنین خطای MAPE پیش‌بینی‌ها در فایل اکسل `./02_vecm_vs_neurals/nn_mape.xlsx` ذخیره شده‌اند.

در زیر، دو نمونه از نمودارهای پیش‌بینی، یکی برای مدل VECM و دیگری برای مدل شبکه‌های عصبی آورده شده است.

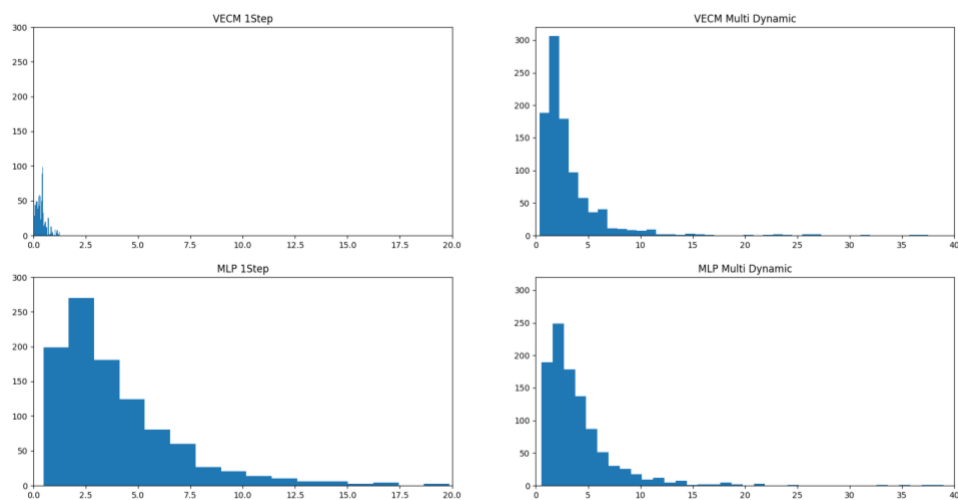


شکل ۱ نمودار پیش‌بینی مدل VECM برای دو سهم CAT و DOW



شکل ۲ نمودار پیش‌بینی مدل MLPRegressor برای دو سهم CAT و DOW

در زیر، نمودار هیستوگرام خطای MAPE برای همه جفت سهام هم‌انباشته، نمایش داده شده است. همان‌طور که مشخص است، مدل VECM هم در پیش‌بینی تک‌گام و هم پیش‌بینی چندگام، به‌طور میانگین خطای کم‌تری دارد.



شکل ۳ هیستوگرام خطای دو مدل VECM و MLP در دو نوع تک‌گام و چندگام

پروژه ۳ - آزمون خطی بودن ارزشهای دیجیتال

در این پروژه، تست خطی بودن برای سری‌های قیمتی ارزشهای دیجیتال در بازه‌های زمانی متفاوت انجام می‌شود.

راهنمای فایل‌ها

توضیح	درس فایل
گزارش توضیح کد	<code>./03_crypto_nIntest/01_crypto_nIntest.pdf</code>
فایل ژوپیتر نوت‌بوک کد همراه با توضیحات	<code>./03_crypto_nIntest/01_crypto_nIntest.ipynb</code>
فایل پایتون کد همراه با توضیحات	<code>./03_crypto_nIntest/01_crypto_nIntest.py</code>
پوشه حاوی نمودارها و جداول نهایی	<code>./03_crypto_nIntest/output</code>

ورودی‌ها

- قیمت‌های تاریخی ده ارز دیجیتال
- بازه زمانی، از روز اول معامله هر ارز تا به امروز است.
- گام زمانی سری‌ها، روزانه است.

روش اجرا

ابتدا سری‌های زمانی هر ارز دیجیتال، به تکه‌های یک‌ساله، دو‌ساله، سه‌ساله و پنج‌ساله تقسیم شدند. تست خطی بودن توسط پکیج `nIntest` بر روی هر کدام از این تکه‌ها اجرا شد. خروجی تابع `nIntestuniv` نتیجه چهار تست است. اگر حداقل دو تست، خطی بودن را رد کند، رای به غیرخطی بودن آن تکه از سری زمانی می‌دهیم.

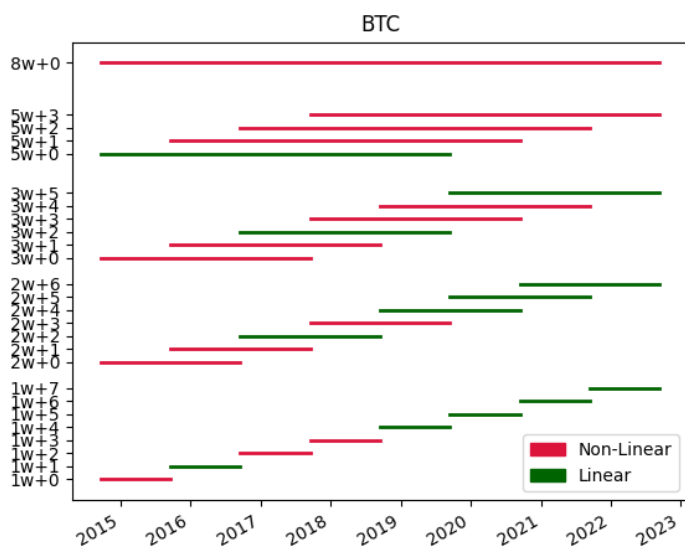
- برخی از توابع و کلاس‌های استفاده شده مهم:

توضیح	تابع یا کلاس
این تابع با دریافت یک سری زمانی، چهار تست خطی بودن را روی آن اجرا می‌کند و p -value هر تست را به عنوان خروجی می‌دهد.	<code>nIntest.nIntstuniv</code>

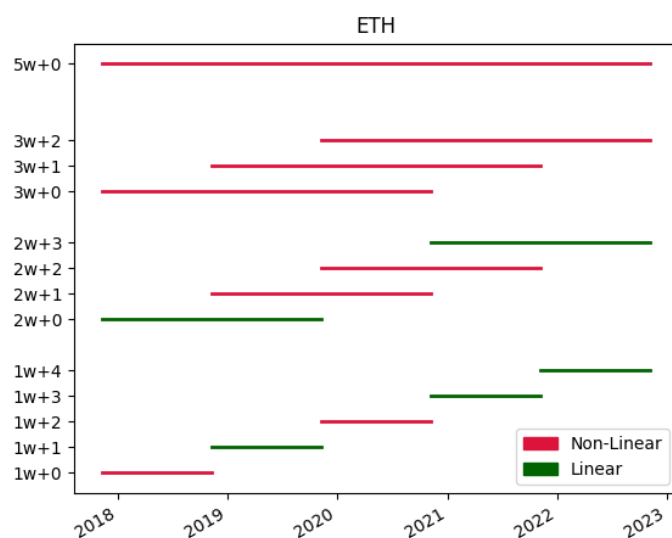
خروجی‌ها

به ازای هر کدام از ارزشهای دیجیتال، یک نمودار و یک جدول به صورت تصویر در پوشه `./03_crypto_nIntest/output` ذخیره شده‌اند. در زیر، دو نمونه از خروجی‌ها آورده شده است. می‌توان نتیجه گرفت که خطی/غیرخطی بودن، وابسته به زمان است.

در نمودارهای زیر، محور افقی زمان است و در محور عمودی، تکه‌های از سری زمانی در پنجره‌ها و زمان‌های مختلف است. در بازه‌هایی که سری زمانی غیرخطی است، از رنگ قرمز و بازه‌هایی که خطی است از رنگ سبز برای نمایش استفاده شده است. در زیر، نمودار بیت‌کوین و اتریوم به همراه جداول مربوطه برای نمونه آورده شده است.



شکل 6 نتیجه آزمون خطی بودن بیت‌کوین در تکه‌زمان‌های متفاوت



شکل 7 تست خطی بودن اتریوم در تکه‌زمان‌های متفاوت

symbol	from	to	window_size	offsetY	is_nln
BTC	2014-09-17	2015-09-16	1	0	True
BTC	2015-09-17	2016-09-15	1	1	False
BTC	2016-09-16	2017-09-15	1	2	True
BTC	2017-09-16	2018-09-15	1	3	True
BTC	2018-09-16	2019-09-15	1	4	False
BTC	2019-09-16	2020-09-14	1	5	False
BTC	2020-09-15	2021-09-14	1	6	False
BTC	2021-09-15	2022-09-14	1	7	False
BTC	2014-09-17	2016-09-15	2	0	True
BTC	2015-09-17	2017-09-15	2	1	True
BTC	2016-09-16	2018-09-15	2	2	False
BTC	2017-09-16	2019-09-15	2	3	True
BTC	2018-09-16	2020-09-14	2	4	False
BTC	2019-09-16	2021-09-14	2	5	False
BTC	2020-09-15	2022-09-14	2	6	False
BTC	2014-09-17	2017-09-15	3	0	True
BTC	2015-09-17	2018-09-15	3	1	True
BTC	2016-09-16	2019-09-15	3	2	False
BTC	2017-09-16	2020-09-14	3	3	True
BTC	2018-09-16	2021-09-14	3	4	True
BTC	2019-09-16	2022-09-14	3	5	False
BTC	2014-09-17	2019-09-15	5	0	False
BTC	2015-09-17	2020-09-14	5	1	True
BTC	2016-09-16	2021-09-14	5	2	True
BTC	2017-09-16	2022-09-14	5	3	True

شکل 5 نتایج تست خطی بودن بیت‌کوین (جدول معادل با نمودار بالا)

symbol	from	to	window_size	offsetY	is_nln
ETH	2017-11-09	2018-11-08	1	0	True
ETH	2018-11-09	2019-11-08	1	1	False
ETH	2019-11-09	2020-11-07	1	2	True
ETH	2020-11-08	2021-11-07	1	3	False
ETH	2021-11-08	2022-11-07	1	4	False
ETH	2017-11-09	2019-11-08	2	0	False
ETH	2018-11-09	2020-11-07	2	1	True
ETH	2019-11-09	2021-11-07	2	2	True
ETH	2020-11-08	2022-11-07	2	3	False
ETH	2017-11-09	2020-11-07	3	0	True
ETH	2018-11-09	2021-11-07	3	1	True
ETH	2019-11-09	2022-11-07	3	2	True
ETH	2017-11-09	2022-11-07	5	0	True

شکل 4 نتایج تست خطی بودن بیت‌کوین (جدول معادل با نمودار بالا)

پروژه ۴ - یافتن توزیع Kernel Density مدل ECM

در این پروژه، سعی می‌شود توزیع Kernel Density مدل ECM برای جفت سهام هم‌انباشته پروژه شماره ۱ تخمین زده شود.

راهنمای فایل‌ها

توضیح	درس فایل
گزارش توضیح کد	./04_kernel_dist/04_dist.pdf
فایل ژوپیتر نوت‌بوک کد همراه با توضیحات	./04_kernel_dist/04_dist.ipynb
فایل پایتون کد همراه با توضیحات	./04_kernel_dist/04_dist.py
فایل اکسل حاوی نتایج تست‌ها	./04_kernel_dist/ecm_dists.xlsx
پوشه حاوی نمودارها به تفکیک شاخص	./04_kernel_dist/plots/{index_name}

ورودی‌ها

- ۵۰۹ جفت سهم هم‌انباشته از تمرین اول.
- بازه زمانی در این تمرین (برخلاف تمرین‌های گذشته)، ۷۲۰ روز در نظر گرفته شده است. زیرا برای انجام آزمون‌های فرض، بهتر است که تعداد داده‌ها بیشتر باشد.
- بنابراین نیاز است که پیش از محاسبه مدل ECM، دوباره آزمون هم‌انباشتگی اجرا شود تا مشخص شود که آیا این سهام در این بازه زمانی نیز هم‌انباشته هستند یا نه.

روش اجرا

به ازای هر جفت سهم هم‌انباشته، چهار تست نرمال بودن شامل: `jarque_bera`، `anderson`، `cramervonmises` و `lilliefors` انجام می‌گیرد. نتایج این چهار تست در نهایت در فایل اکسل ذخیره می‌شود. هم‌چنین تست `Kolmogorov-Smirnov` در این تمرین به گونه‌ای پیاده‌سازی شده که بتواند علاوه بر نرمال بودن، توزیع‌های دیگر را شامل: `lognorm`، `chi2`، `t`، `beta`، `gamma`، `weibull_min` آزمون کند. توضیحات مربوط به کدها، در گزارش کدها آورده شده است.

مراحل اجرای آزمون کولموگروف-اسمیرنوف به این شکل است:

- با توجه به `kernel density` مدل ECM، بهترین پارامترهای هر توزیع را می‌یابیم.
- به کمک تابع `kstest` در پکیج `scipy` تست می‌کنیم که آیا سری زمانی، با هر کدام از توزیع‌های ذکر شده در بالا (با پارامترهایی به دست آمده از مرحله قبل) چقدر شباهت دارد. فرض صفر این آزمون، شباهت این دو توزیع به هم است. اگر فرض صفر رد شود، می‌توان نتیجه گرفت که دو توزیع به هم شباهت ندارند. در این تمرین، هر جا فرض صفر رد نشود، نتیجه گرفته شده است که دو توزیع همانند هم هستند (که زیاد دقیق نیست اما برای رسم نمودارها و خروجی اکسل، این فرض مورد نیاز است).
- اگر توزیع `kernel density` با هر کدام از توزیع‌های ذکر شده شباهت داشته باشد، علاوه بر ذخیره کردن نام آن توزیع در فایل اکسل، نمودار `kernel density` به همراه آن توزیع رسم و ذخیره می‌شود.

- برخی از توابع و کلاس‌های استفاده شده مهم:

توضیح	تابع یا کلاس
چهار تست نرمال بودن توزیع از پکیج <code>scipy</code> . فرض صفر همه این تست‌ها، نرمال بودن توزیع دیتای ورودی به این توابع است.	<code>scipy.stats.jarque_bera</code> <code>scipy.stats.anderson</code> <code>scipy.stats.cramervonmises</code> <code>scipy.stats.lilliefors</code>

scipy.stats.kstest

تست کولموگروف-اسمیرنوف برای
شباهت دو توزیع

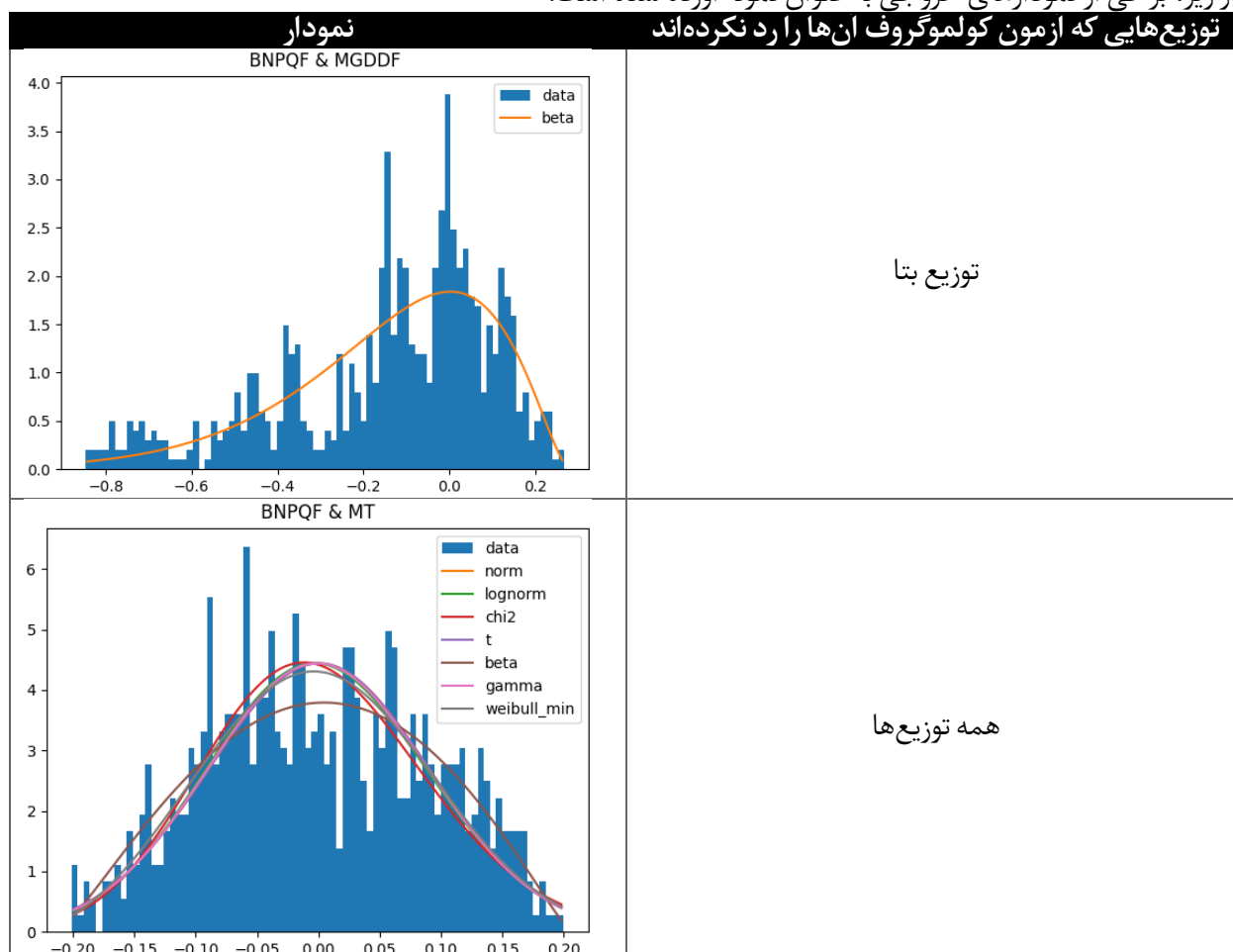
خروجی‌ها

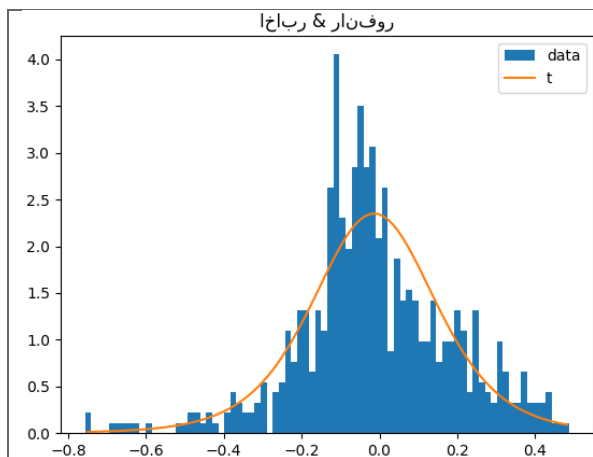
تعداد ۳۵۰ نمودار در پوشه `04_kernel_dist/plots` ذخیره شده‌اند. این نمودارها حاوی kernel density مدل ECM و توزیع‌های فیت‌شده بر آن هستند. هم‌چنین فایل اکسل `04_kernel_dist/ecm_dist.xlsx` حاوی نتایج همه تست‌های انجام شده است.

- از مجموع ۳۵۰ جفت سهم، ۱۶۶ تای آن‌ها از هیچ کدام از توزیع‌های بالا پیروی نمی‌کنند.
- در جدول زیر، تعداد جفت سهامی که هر کدام از توزیع‌های ذکرشده بر kernel density آن‌ها به خوبی برازش شده‌اند، آورده شده است: (دقت شود که به ازای هر جفت سهم، ممکن است چند توزیع برازش شده باشد).

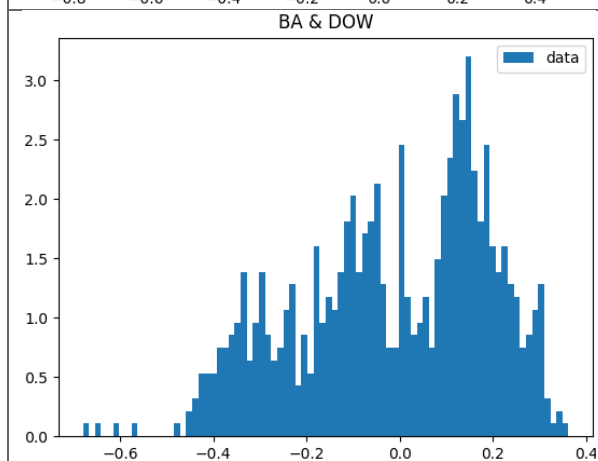
normal	lognorma	chi2	t	beta	gamma	weibull_min
۳۷	۵۰	۲۱	۱۳۵	۸۰	۴۰	۸۵

در زیر، برخی از نمودارهای خروجی به عنوان نمونه آورده شده است.





توزیع تی-استیودنت
(سهام اخبار و رانفور)



هیچ کدام از توزیع ها