

پروژک اول:

استخراج و تقسیم‌بندی داده:

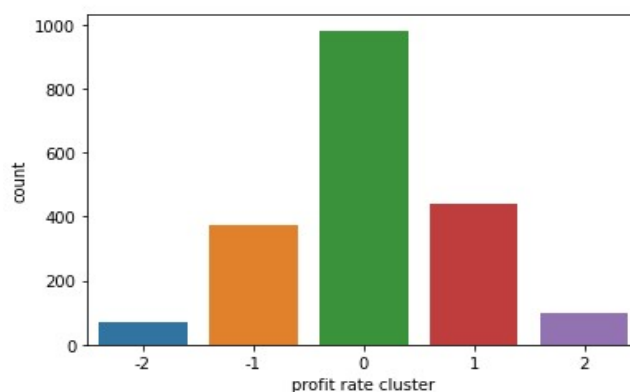
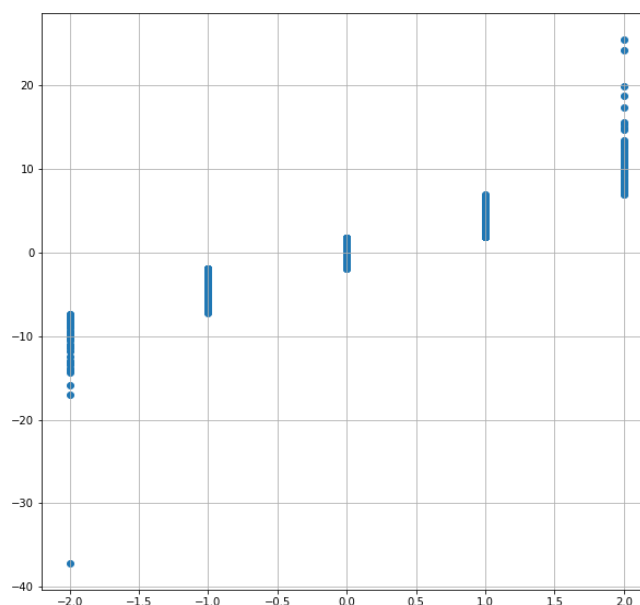
از کتابخانه `yfinance` برای گرفتن داده‌ها در بخش `AI` استفاده شده است. داده‌های بیت‌کوین به صورت روزانه شامل قیمت باز و بسته شدن، بالاترین و پایین‌ترین قیمت و حجم معاملات در بازه‌ی زمانی ۱ ژانویه ۲۰۱۷ تا ۲۲ می ۲۰۲۲ استخراج شدند. سپس نرخ بازدهی روزانه این کوین محاسبه و به دیتافریم اضافه گردید. در ادامه با استفاده از کتابخانه `sklearn` داده‌ها به صورت ۸۰ درصد برای آموزش و ۲۰ درصد برای آزمایش تقسیم‌بندی شده‌اند.

ساخت مدل:

برای این بخش از الگوریتم `kmeans` برای دسته‌بندی داده‌ها استفاده و تعداد دسته‌ها ۵ در نظر گرفته شد.

نتیجه:

در نهایت داده‌ها با نرخ بازدهی ۲- تا ۲ درصد در دسته بازدهی صفر، داده‌ها با نرخ بازدهی ۲ الی ۷ و ۲- الی ۷- درصد به ترتیب در دسته بازدهی ۱+ و ۱- و داده‌ها با نرخ بازدهی ۷+ درصد و بیشتر و ۷- درصد و کمتر به ترتیب در دسته بازدهی ۲+ و ۲- قرار گرفتند. در نهایت تعداد دسته‌ها به ۳ دسته کاهش داده شد و برای نرخ بازدهی ۲+ درصد و بیشتر عدد ۱+ و نرخ بازدهی ۲- درصد و کمتر عدد ۱- و ۲- تا ۲ درصد هم عدد ۰ در نظر گرفته شد.

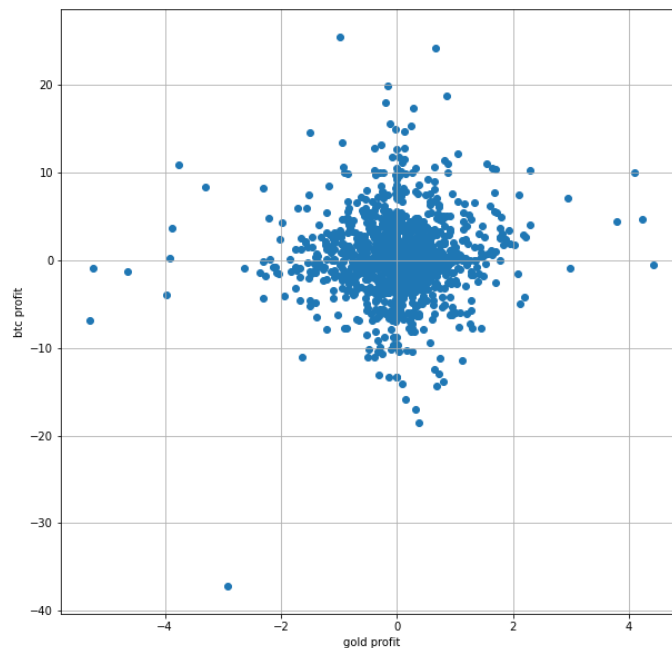


پروژک دوم:

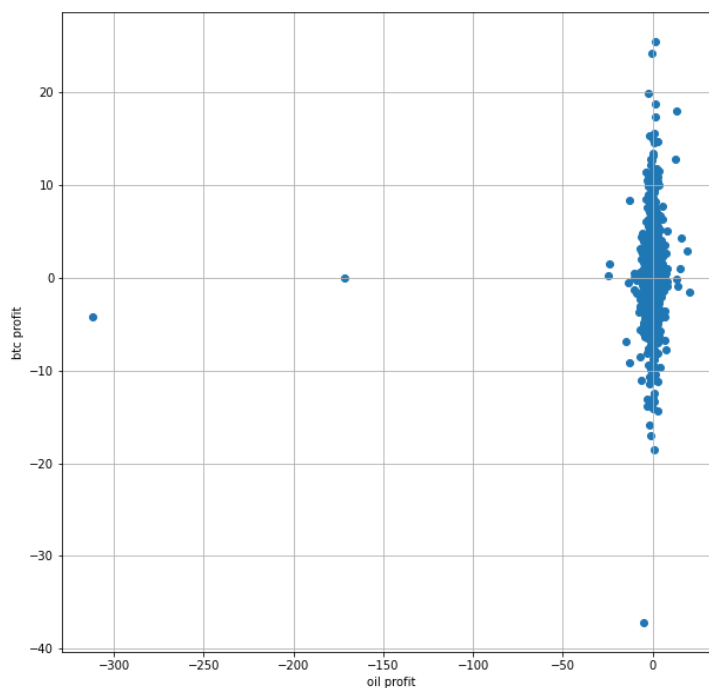
استخراج و تقسیم‌بندی داده:

داده‌های نفت و طلا به صورت روزانه شامل قیمت باز و بسته شدن، بالاترین و پایین‌ترین قیمت و حجم معاملات در بازه‌ی زمانی ۱ ژانویه ۲۰۱۷ تا ۲۲ می ۲۰۲۲ استخراج شدند. سپس نرخ بازدهی روزانه این دو محاسبه و به دیتافریم‌ها اضافه گردید. در ادامه نرخ بازدهی طلا، نفت و بیت کوین در یک دیتا فریم جداگانه ریخته شد. ضریب همبستگی طلا به بیت کوین و نفت به بیت کوین به صورت جداگانه محاسبه شد. همچنین نسبت این دو به بیت کوین به صورت جداگانه در دو نمودار به نمایش درآمده است.

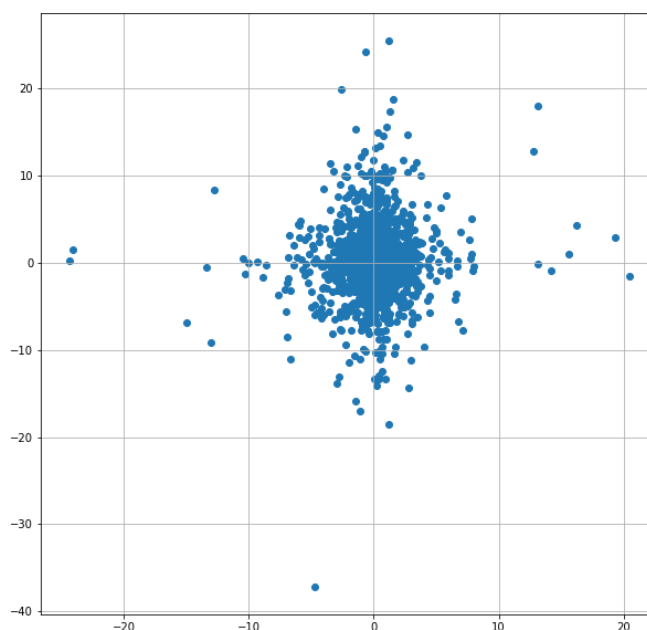
نسبت طلا به بیت کوین:



نسبت نفت به بیت کوین:



با حذف داده‌های پرت به تصویر زیر می‌رسیم:



نتیجه:

با توجه به شکل نمودارها و همچنین ضریب همبستگی طلا و نفت به بیت‌کوین که به ترتیب برابر ۰/۰۵ و ۰/۰۴ که نزدیک به صفر می‌باشد، می‌توان نتیجه گرفت که تغییرات این دو تأثیری بر قیمت بیت‌کوین ندارند. همچنین با اعمال لگ زمانی مثبت و منفی تغییری در این نتایج حاصل نگردید.

پروژک سوم:

نرمالسازی و تقسیم‌بندی داده:

از داده‌های استخراج شده در پروژک اول در این قسمت استفاده گردید. داده‌ها با استفاده از الگوریتم minmax جهت نرمالسازی به بازه صفر تا ۱ برده شدند و به صورت ۸۰ درصد برای آموزش و ۲۰ درصد برای آزمایش تقسیم بندی شدند. به این معنا که ابتدا ۲۰ درصد انتهایی داده‌ها به عنوان مجموعه آزمایش جدا شده و باقی داده‌ها وارد مدل می‌شوند.

مدل‌سازی:

به دلیل بالانس نبودن داده‌ها و تعداد زیاد دسته صفر از الگوریتم RandomOverSampler کتابخانه imblearn برای بالانس کردن داده‌ها استفاده گردید. سپس پارامترهای ماشین تنظیم شدند و در ادامه مقادیر مختلفی برای تعداد روز متوالی گذشته امتحان گردید که به نظر تعداد ۷ روز گذشته متوالی برای این مدل می‌تواند کارا باشد.

در آخر میزان precision، recall و f1-score مدل برای مجموعه‌های آموزش و آزمایش قابل مشاهده است. برای مجموعه train:

	precision	recall	f1-score	support
-1.0	0.32	0.34	0.33	336
0.0	0.62	0.66	0.64	804
1.0	0.35	0.30	0.32	434
accuracy			0.49	1574
macro avg	0.43	0.43	0.43	1574
weighted avg	0.48	0.49	0.49	1574

برای مجموعه test :

	precision	recall	f1-score	support
-1.0	0.25	0.25	0.25	106
0.0	0.50	0.49	0.49	177
1.0	0.33	0.34	0.33	104
accuracy			0.38	387
macro avg	0.36	0.36	0.36	387
weighted avg	0.38	0.38	0.38	387