

به نام خدا

هوش مصنوعی:

پروژک اول:

ابتدا با استفاده از yfinance API دیتا مورد نظر خود که تاریخچه بیت کوین در طی 14 سال اخیر است را دانلود کرده و در یک دیتا فریم ذخیره کرده.

	Open	High	Low	Close	Adj Close	Volume
Date						
2014-09-17	465.864014	468.174011	452.421997	457.334015	457.334015	21056800
2014-09-18	456.859985	456.859985	413.104004	424.440002	424.440002	34483200
2014-09-19	424.102997	427.834991	384.532013	394.795990	394.795990	37919700
2014-09-20	394.673004	423.295990	389.882996	408.903992	408.903992	36863600
2014-09-21	408.084991	412.425995	393.181000	398.821014	398.821014	26580100
...
2022-05-20	30311.119141	30664.976562	28793.605469	29200.740234	29200.740234	30749382605
2022-05-21	29199.859375	29588.869141	29027.394531	29432.226562	29432.226562	17274840442
2022-05-22	29432.472656	30425.861328	29275.183594	30323.722656	30323.722656	21631532270
2022-05-23	30309.396484	30590.585938	28975.560547	29098.910156	29098.910156	31483454557
2022-05-24	29083.347656	29483.505859	28792.775391	29193.910156	29193.910156	29187108864

2807 rows × 6 columns

حال بعد از محاسبه نرخ بازدهی روزانه برای درک بهتر این نمودار آن را رسم میکنیم:

با توجه به هر دو نمودار میتوان فهمید که نرخ همبستگی بیت کوین با طلا و نفت به طور میانگین (متوسط) تقریباً صفر است و این نرخ در lag ها مختلف حول صفر نوسان می کند.

پروژک سوم:

برای این قسمت ابتدا با توجه به اینکه باید هر مقدار را در یک خوشه قرار دهیم دیتا را خوشه بندی کرده (مثل پروژک اول) و بر این اساس لیبل هر یک از مقادیر daily return را بدست می آوریم و بعد با توجه به اینکه برای پیشبینی هر مقدار به 4 مقدار قبلی نیاز داریم و لیبل های بدست آمده , دیتاست را مطلوب را می سازیم.

حال دیتاست را به 3 بخش train/validation/test با نسبت 0.8,0.1,0.1 تقسیم میکنم.

حال با توجه به به تعداد دیتا و تعداد ورودی و خروجی یک شبکه عصبی ساده به معماری زیر را می سازیم:

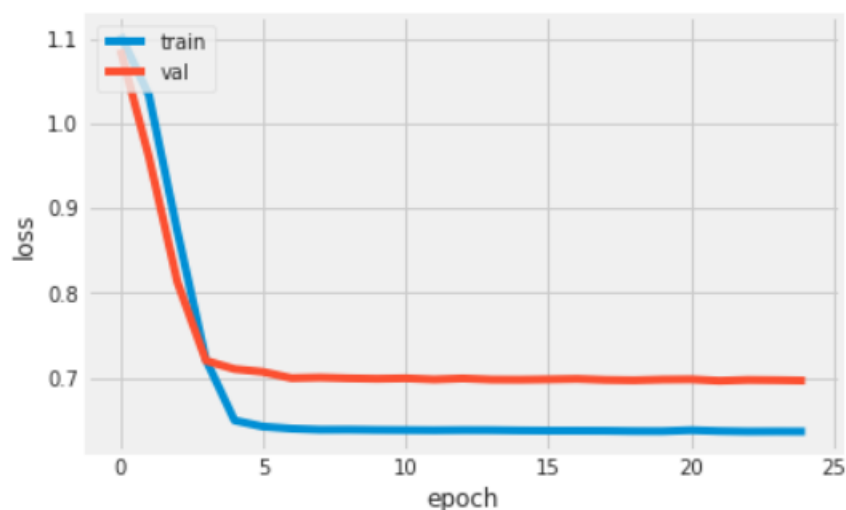
Model: "sequential_1"

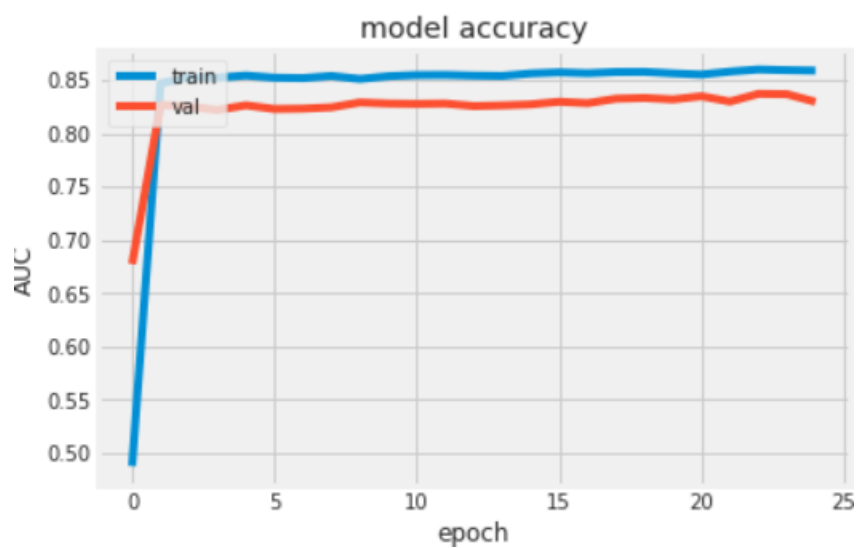
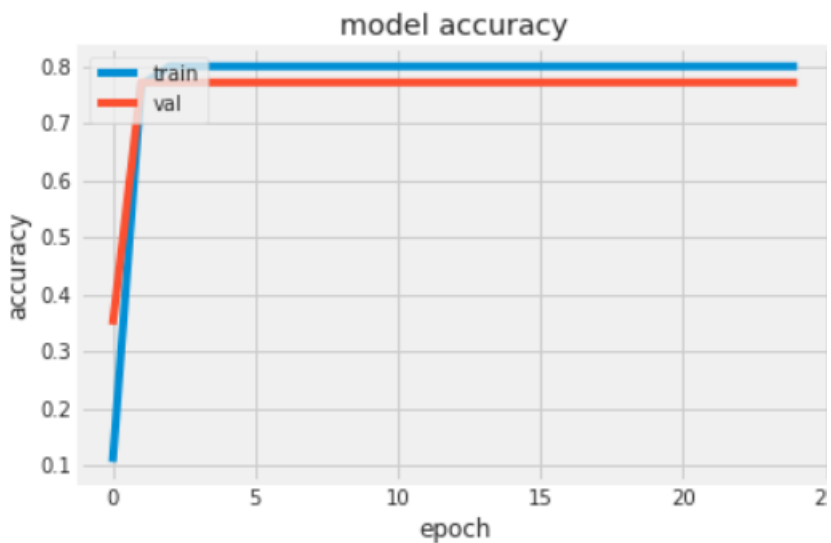
Layer (type)	Output Shape	Param #
dense_2 (Dense)	(None, 30)	150
dense_3 (Dense)	(None, 20)	620
dense_4 (Dense)	(None, 10)	210
dense_5 (Dense)	(None, 3)	33

=====
Total params: 1,013
Trainable params: 1,013
Non-trainable params: 0

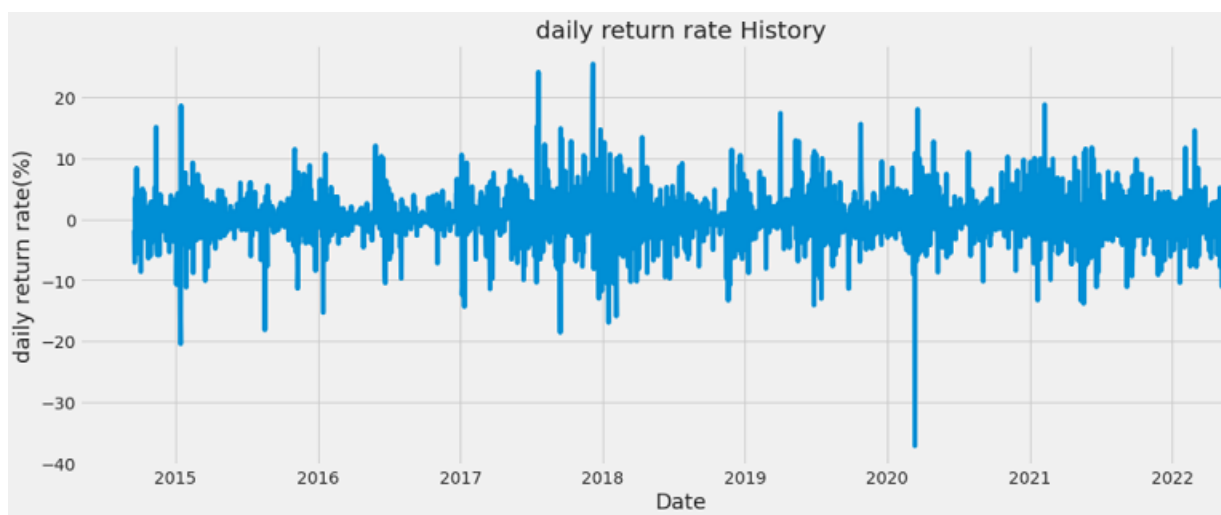
برای آموزش مدل از تکنیک earlystopping استفاده شده است.

حال نمودار های history را رسم میکنم:



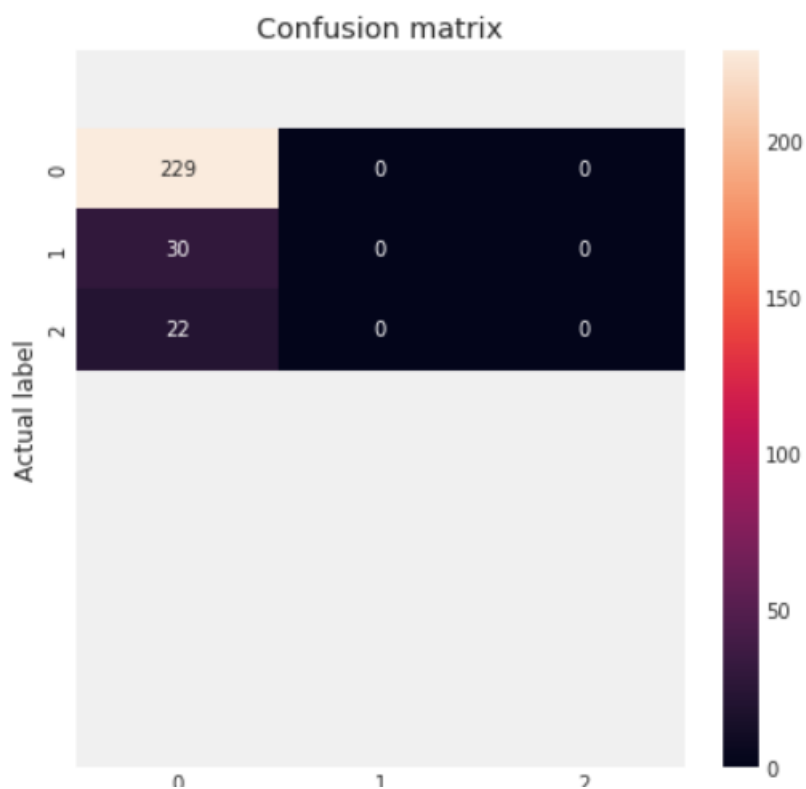


گزارش متریک ها بر روی دیتا تست (test part):



acc of test part is :0.8149466192170819
f1 of test part is :0.8149466192170819
precision of test part is :0.8149466192170819
recall of test part is :0.814946619217081

حال Confusion Matrix را برای داده های تست رسم میکنم:



به واضحی مشاهده میشود که مدل عملکرد خوبی نداشته است و همه داده های تست را از نوع کلاس یک پیشبینی شده است و مدل کند handle را نتوانسته دیتای imbalance