

به نام خدا



پروژه نهایی درس تحلیل داده

موضوع:

تحلیل و پیش‌بینی قیمت ارز دیجیتال

اعضای گروه:

مهدی محمدی نسب ۸۱۰۱۰۰۵۶۵

کیهان رعیتی ۸۱۰۱۰۰۳۶۱

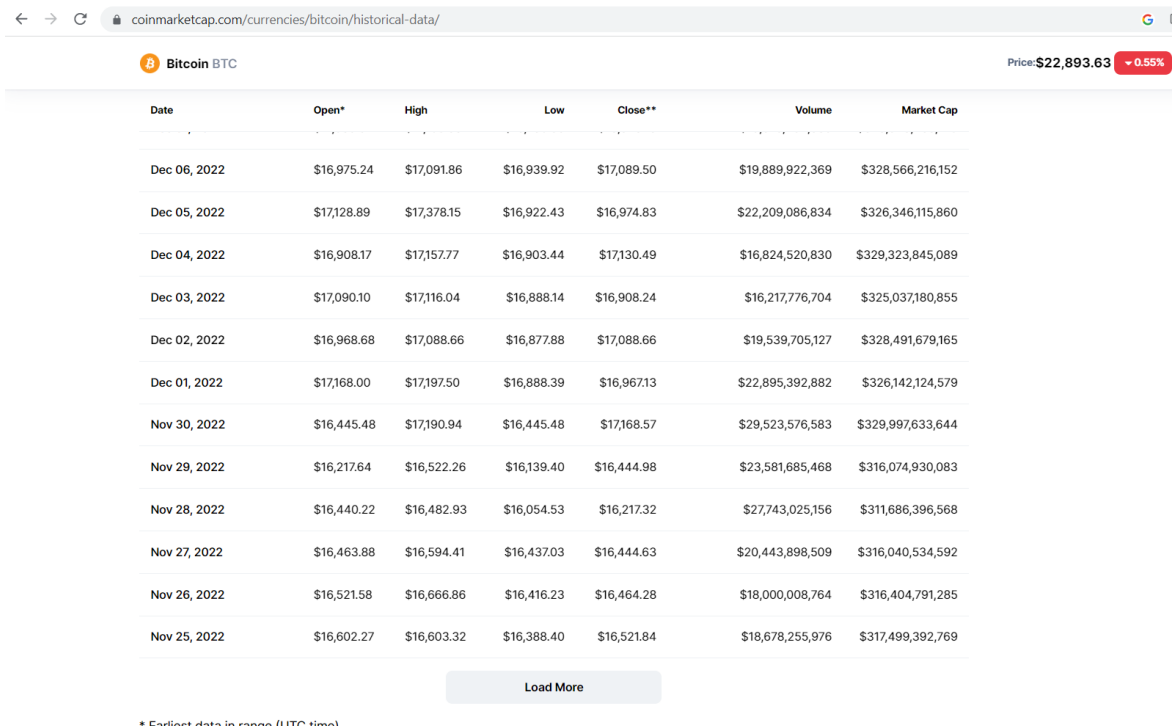
سارا رستمی ۸۱۰۱۰۰۳۵۵

فهرست

بخش اول: جمع‌آوری کردن و Crawl کردن	۳
بخش دوم: روش‌های تمیزسازی (پیش‌پردازش)	۴
بخش سوم: مصورسازی و تحلیل EDA	۶
بخش چهارم: انتخاب ویژگی و کاهش ابعاد	۱۳
بخش پنجم: روش‌های طبقه‌بندی	۱۴
مدل ۱: Logistic Regression	۱۵
مدل ۲: SVM	۱۵
مدل ۲: Decision Tree	۱۶
مدل ۳: KNN	۱۷
مدل ۴: BiLSTM	۱۸
مدل ۵: Transformer	۱۹
مدل ۶: CNN	۲۰
مقایسه مدل‌ها	۲۱
استفاده از ویژگی‌های random forest	۲۳
انتخاب و ensemble کردن مدل‌ها	۲۵

بخش اول: جمع‌آوری کردن و Crawl کردن

در این مرحله برای جمع‌آوری داده‌های ارز دیجیتال، از کتابخانه‌ی Selenium استفاده کردیم. قیمت روزانه‌ی ارزهای دیجیتال بیت‌کوین، اتریوم و تتر مربوط به سال‌های ۲۰۱۵ تا ماه جاری را از وبسایت <https://coinmarketcap.com> کراول کردیم. با استفاده از سلینیوم لینک مربوط به هر یک از ارزهای مذکور را در مرورگر باز کرده و دیتای موجود در جدول را جمع‌آوری کردیم همچنین برای لود کردن دیتاهای گذشته، برای کلیک کردن روی گزینه‌ی Load More وبسایت، thresholdی برابر با ۱۳۰ تعیین کرده تا دیتای جداول مربوط به سال‌های قبلی را هم جمع‌آوری کنیم (۱۳۰ بار کلیک بر روی گزینه). سپس داده‌های اتریوم و تتر را به صورت ستونی به داده‌های بیت‌کوین با دستور Join اضافه کردیم. در نهایت یک ستون label به داده‌ها اضافه کردیم. این ستون شامل مقادیر باینری ۰ و ۱ بوده و از روی ستون close (این ستون نشان‌دهنده‌ی قیمت close ارز بیت‌کوین در هر روز) ساخته می‌شود و نشان‌دهنده‌ی این است که قیمت بیت‌کوین نسبت به روز قبل کاهش (مقدار ۰) یا افزایش (مقدار ۱) داشته است. این داده‌ها را به عنوان داده‌ی خام در یک فایل csv ذخیره کردیم (بدون label ذخیره شده و پس از خواندن فایل ستون هدف اضافه خواهد شد). هدف ما پیش‌بینی ستون label می‌باشد. در واقع قصد داریم با داشتن دیتای یک روز، پیش‌بینی کنیم روند قیمت ارز دیجیتال بیت‌کوین در روز بعد صعودی یا نزولی است. دقت شود که داده‌ی مربوط به ارزهای تتر و اتریوم به عنوان فیچرهایی برای پیش‌بینی قیمت بیت‌کوین محسوب می‌شوند.



Date	Open*	High	Low	Close**	Volume	Market Cap
Dec 06, 2022	\$16,975.24	\$17,091.86	\$16,939.92	\$17,089.50	\$19,889,922,369	\$328,566,216,152
Dec 05, 2022	\$17,128.89	\$17,378.15	\$16,922.43	\$16,974.83	\$22,209,086,834	\$326,346,115,860
Dec 04, 2022	\$16,908.17	\$17,157.77	\$16,903.44	\$17,130.49	\$16,824,520,830	\$329,323,845,089
Dec 03, 2022	\$17,090.10	\$17,116.04	\$16,888.14	\$16,908.24	\$16,217,776,704	\$325,037,180,855
Dec 02, 2022	\$16,968.68	\$17,088.66	\$16,877.88	\$17,088.66	\$19,539,705,127	\$328,491,679,165
Dec 01, 2022	\$17,168.00	\$17,197.50	\$16,888.39	\$16,967.13	\$22,895,392,882	\$326,142,124,579
Nov 30, 2022	\$16,445.48	\$17,190.94	\$16,445.48	\$17,168.57	\$29,523,576,583	\$329,997,633,644
Nov 29, 2022	\$16,217.64	\$16,522.26	\$16,139.40	\$16,444.98	\$23,581,685,468	\$316,074,930,083
Nov 28, 2022	\$16,440.22	\$16,482.93	\$16,054.53	\$16,217.32	\$27,743,025,156	\$311,686,396,568
Nov 27, 2022	\$16,463.88	\$16,594.41	\$16,437.03	\$16,444.63	\$20,443,898,509	\$316,040,534,592
Nov 26, 2022	\$16,521.58	\$16,666.86	\$16,416.23	\$16,464.28	\$18,000,008,764	\$316,404,791,285
Nov 25, 2022	\$16,602.27	\$16,603.32	\$16,388.40	\$16,521.84	\$18,678,255,976	\$317,499,392,769

شکل ۱: داده‌های تاریخی بیت‌کوین در coinmarketcap.com

بخش دوم: روش‌های تمیزسازی (پیش‌پردازش)

۱. ابتدا تعداد nan های هر feature را نمایش می‌دهیم که نشان می‌دهد ما در داده‌های خود هیچ مقدار نامشخصی نداریم. قیمت های هر بخش با \$ مشخص شده بودند که ما آن را حذف کردیم. بعضی از مقادیر نیز شامل کاما بودند که آنها نیز لازم به پاکسازی داشتند.

از اندیکاتورهای گوناگون موجود، می‌توان به صورت منتخب چندین اندیکاتور کاربردی و مهم را به عنوان ویژگی به داده‌ها اضافه نمود. این اندیکاتورها، به این دلیل که برای محاسبه شدن از چند کندل و اطلاعات قبل از خود استفاده می‌کنند (به عنوان مثال اندیکاتور moving average با دوره ۵۰، از قیمت ۵۰ روز قبل استفاده می‌کند)؛ بنابراین در ابتدای داده، مقادیر NaN برای اینگونه اندیکاتورها به وجود می‌آیند. برای حذف این مقادیر، ستون‌هایی که ویژگی‌های NaN دارند، حذف می‌شوند.

اندیکاتورها به این دلیل اضافه می‌شوند که در پیدا نمودن یک‌سری از الگوهای پنهان به مدل‌های یادگیری کمک نمایند.

همه ویژگی‌های موجود عبارت‌اند از:

- open_x, high_x, low_x, close_x, volume_x و market cap_x: قیمت OHLC، حجم و ظرفیت ارز اتریوم
- open_y, high_y, low_y, close_y, volume_y و market cap_y: قیمت OHLC و حجم و ظرفیت ارز تتر
- open, high, low, close, volume و market cap: قیمت OHLC، حجم و ظرفیت ارز بیت‌کوین
- ema_8: اندیکاتور exponential moving average با دوره زمانی ۸. این اندیکاتور

$$EMA_{Today} = \left(Value_{Today} * \left(\frac{Smoothing}{1 + Days} \right) \right) + EMA_{Yesterday} * \left(1 - \left(\frac{Smoothing}{1 + Days} \right) \right)$$

where:

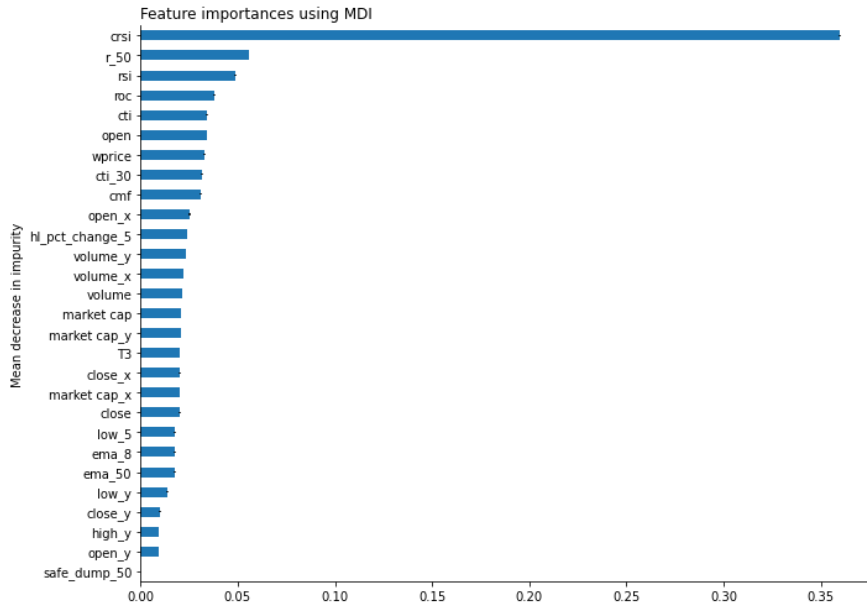
EMA = Exponential moving average

- ema_50: اندیکاتور exponential moving average با دوره زمانی ۵۰
- cti: اندیکاتور correlation trend indicator با دوره زمانی ۱۰ که قدرت و جهت کنونی یک ترند را نشان می‌دهد.
- cti_30: اندیکاتور cti با دوره زمانی ۳۰

- roc: اندیکاتور rate-of-change با دوره زمانی ۹، که درواقع گشتاور نوسان ساز مطلق است.
 - rsi: اندیکاتور Relative Strength Index با دوره زمانی ۱۴ که اندازه و سرعت بازار را نشان می دهد.
 - crsi: combined RSI که ترکیبی از اندیکاتورهای RSI و ROC است
 - r_50: اندیکاتور Williams_r با دوره زمانی ۵۰ که یک نوع اندیکاتور گشتاور است و برای نشان دادن oversold و overbought استفاده می شود.
 - hl_pct_change_5: بیشترین درصد تغییرات غلتان با دوره زمانی ۵
 - cmf: اندیکاتور chaikin money flow با دوره زمانی ۲۰ که یک نوع اندیکاتور میانگین حجم وزن دار است.
 - T3: به صورت زنجیره ای برروی قیمت نهایی EMA با دوره زمانی ۵ را اعمال نموده و سپس یک میانگین وزن دار را محاسبه می نماید.
 - low_5: کمترین قیمت از ۵ قیمت اخیر (low) را در هر لحظه محاسبه می کند.
 - wprice: این اندیکاتور با توجه به قیمت های high، low و close و همچنین volume، یک میانگین وزن دار بر حسب حجم را محاسبه می کند.
۲. کلاس های این مسئله، طبق تعریف در نظر گرفته شده برای هدف، متوازن است.

بخش سوم: مصورسازی و تحلیل EDA

۱. نتیجه random forest به صورت زیر است:

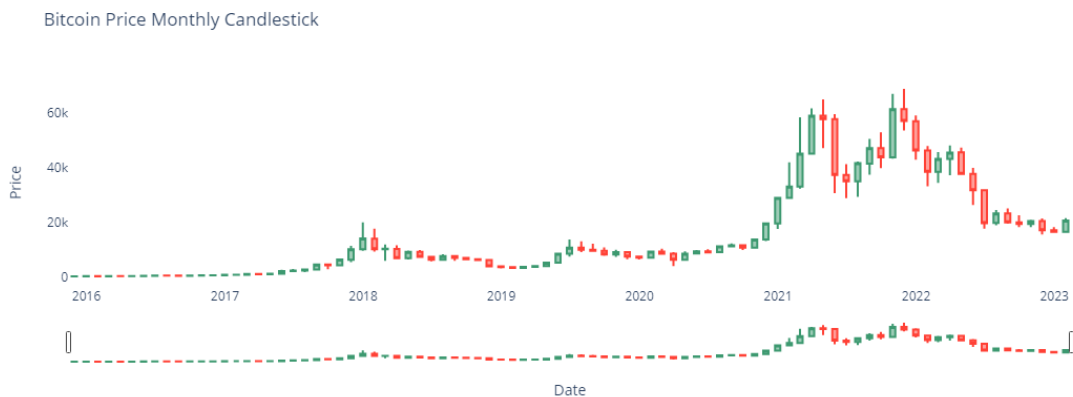


شکل ۲: انتخاب ویژگی‌ها با random forest و امتیاز MDI

اهمیت این ویژگی‌ها، با امتیاز MDI با توجه به label محاسبه می‌شود.

طبق نمودار بالا، ویژگی crsi با اختلاف از سایر ویژگی‌ها مهم‌تر بوده و بیشترین تأثیر برای طبقه‌بندی توسط مدل random forest را دارد؛ زیرا اطلاعات غنی و مهمی را از قیمت‌ها نظیر سرعت و اندازه بازار در اختیار دارد.

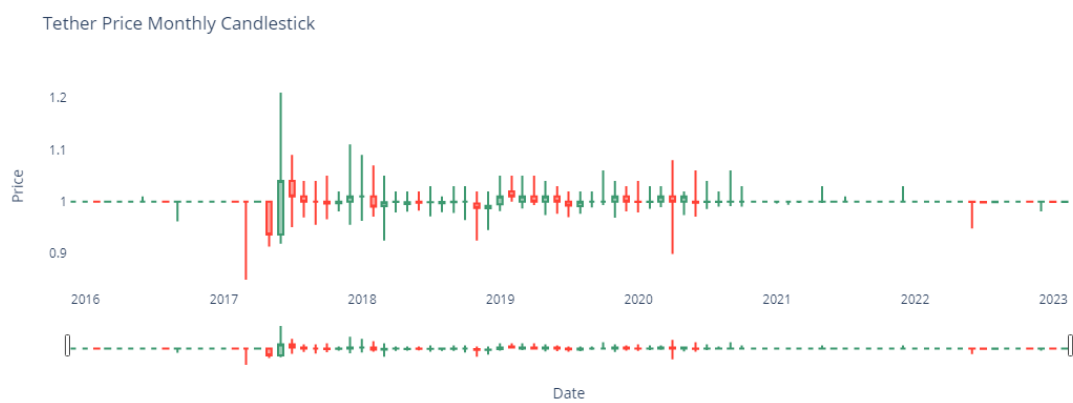
۲. ابتدا نمودارهای کندل‌استیک مربوط به هریک نشان داده می‌شوند:



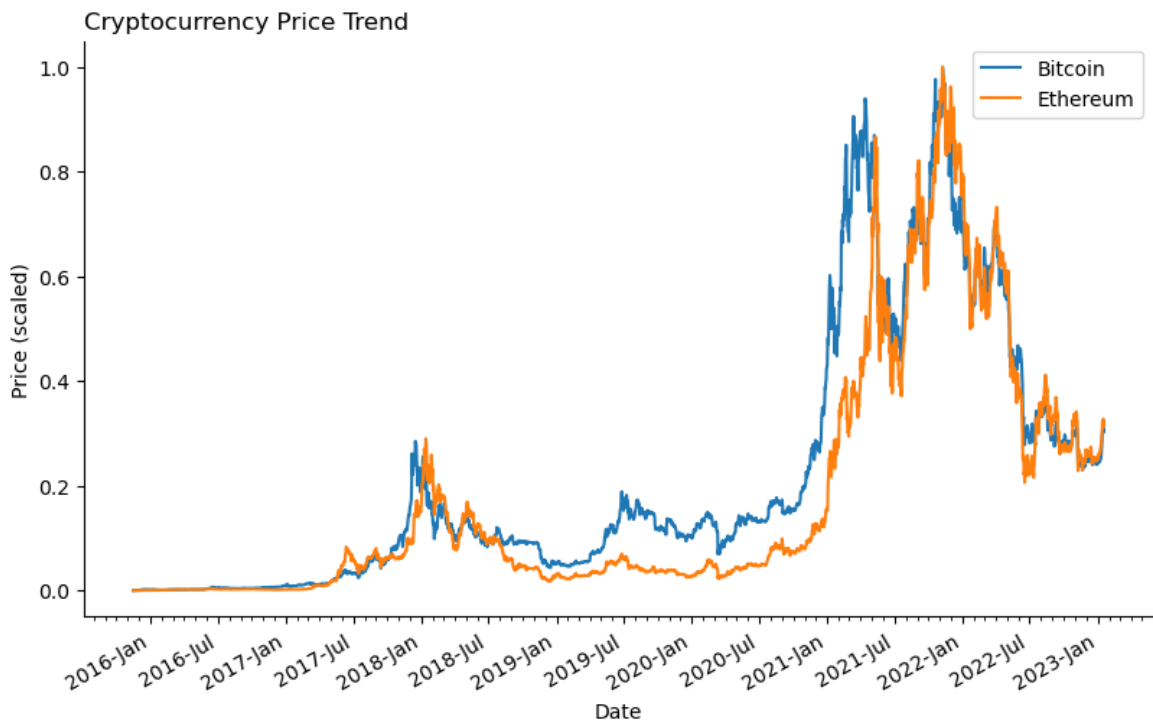
شکل ۳: نمودار candlestick ماهانه بیت‌کوین از سال ۲۰۱۵



شکل ۴: نمودار candlestick ماهانه اتریوم از سال ۲۰۱۵

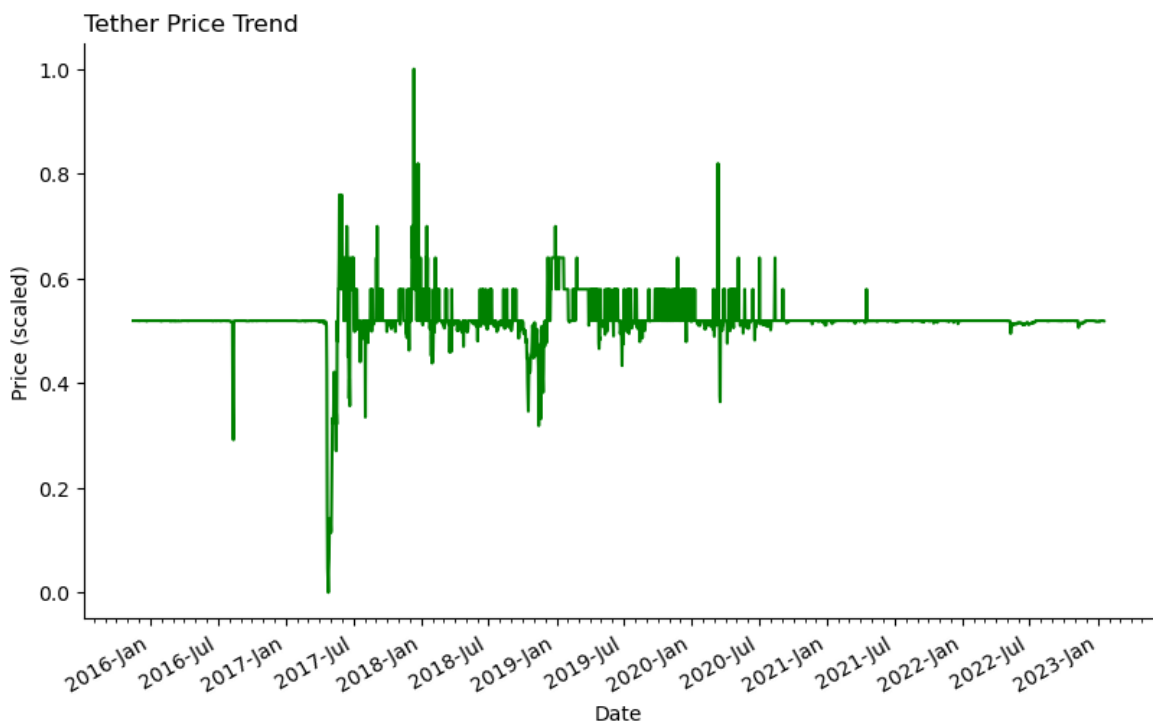


شکل ۵: نمودار candlestick ماهانه تتر



شکل ۶: نمودار مقایسه روند قیمت نهایی روزانه بیت‌کوین و اتریوم

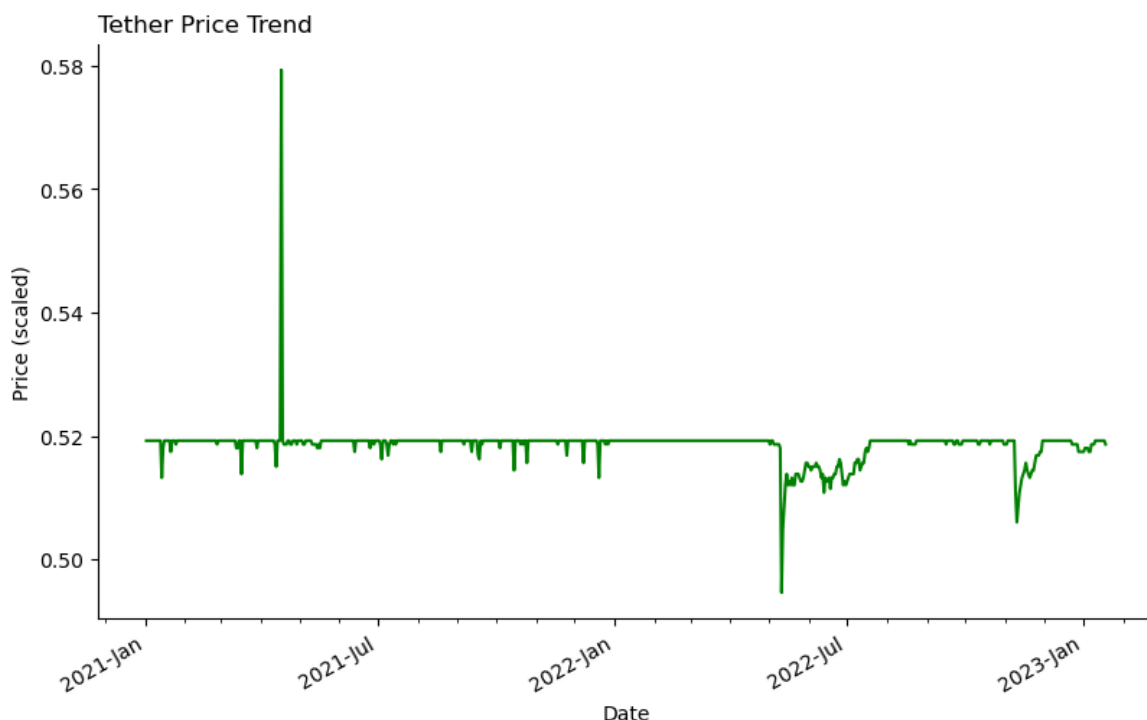
همانطور که در روند قیمت نهایی اتریوم و بیت‌کوین مشاهده می‌شود، این دو روندی بسیار مشابه داشته و بعضی مواقع اتریوم با کمی تأخیر نسبت به بیت‌کوین همان روند را تکرار می‌کند.



شکل ۷: روند قیمت نهایی تتر از سال ۲۰۱۵

اما روند تتر شباهتی با دو ارز دیگر ندارد؛ بلکه به صورت نوسانی بوده و معمولاً در یک بازه مشخص قرار داد.

با توجه به نمودار در اواسط نمودار در سال‌های ۲۰۱۷ تا ۲۰۲۱ نوسانات شدیدی نسبت به پس از آن وجود دارد. در شکل زیر روند تتر از ۲۰۲۱ به بعد مشاهده می‌شود. می‌توان رابطه‌ای به نسبت معکوس را بین نوسانات تتر و بیت‌کوین تا حدی از این سه نمودار مشاهده نمود.

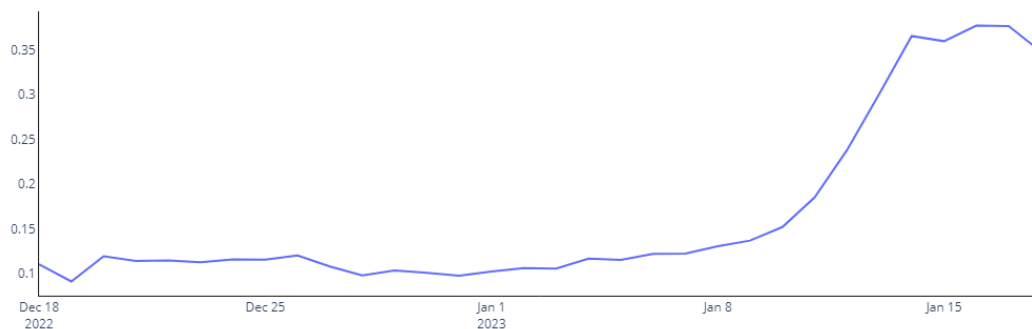


شکل ۸: روند قیمت تتر در سال اخیر

برای بررسی دقیق‌تر الگوهای تکرار شونده بین قیمت‌های نهایی بیت‌کوین و همچنین قیمت بین‌کوین با اتریوم از توابع mape و correlation برای محاسبه فاصله و شباهت دو سیگنال استفاده شده است.

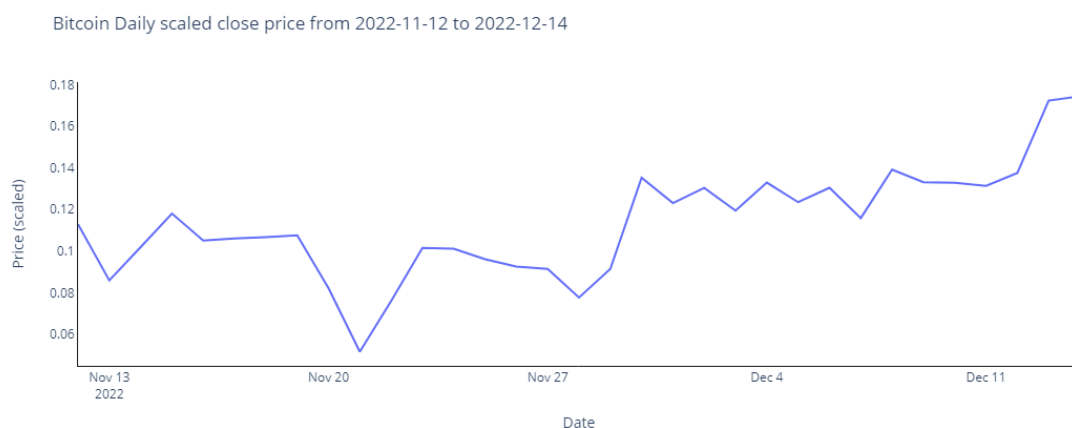
از آنجا که مهم‌ترین مسئله این است که آخرین وضعیت کنونی و الگوی آن چگونه و شبیه به چه الگو یا الگوهای تکرار شونده است، تا با استفاده از آن‌ها بتوان درباره شباهت و ادامه آن الگو بحث کرد؛ پس یک ماه آخر از داده را انتخاب می‌نماییم:

Bitcoin Daily scaled close price from 2022-12-18 to 2023-01-18



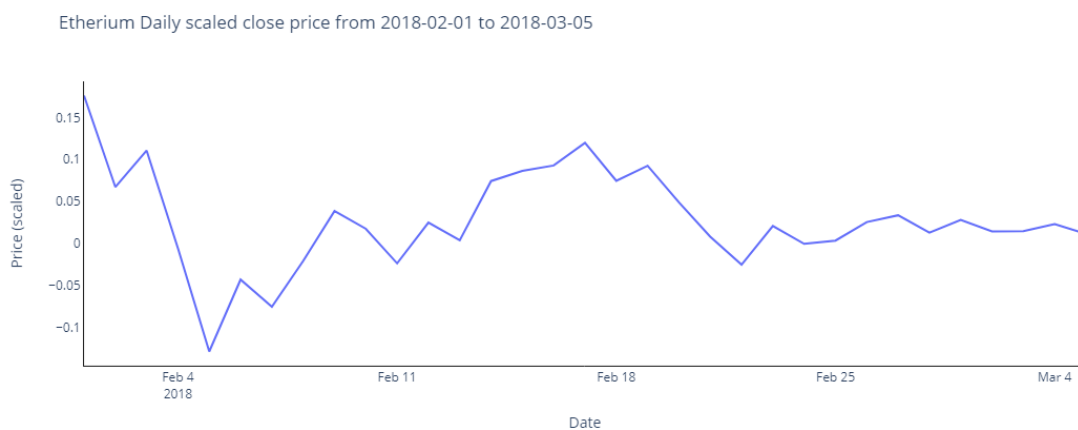
شکل ۹: قیمت روزانه بیت کوین در یک ماه اخیر

با استفاده از توابع mape و correlation سعی بر انتخاب نزدیک‌ترین الگو از گذشته برای این ماه می‌شود:
 بر اساس mape نزدیک‌ترین الگو از گذشته بیت کوین، الگوی زیر است:



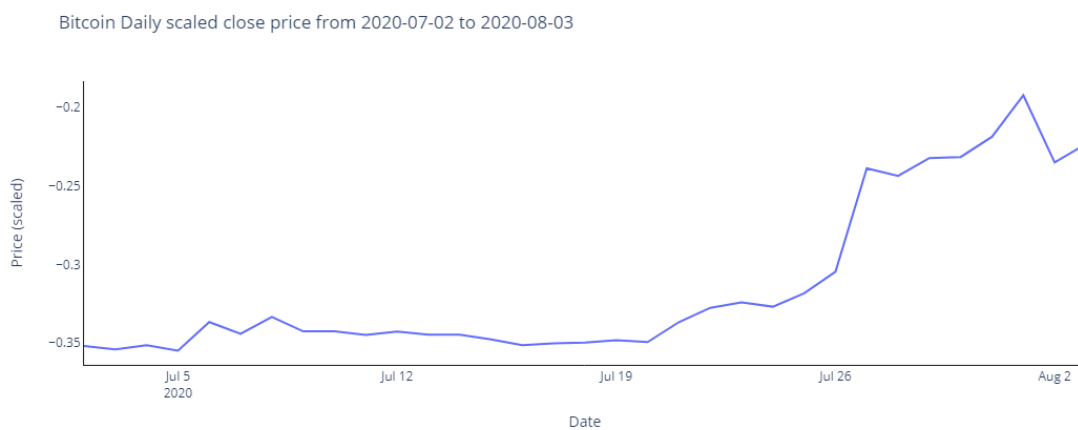
شکل ۱۰: الگوی مشابه پیدا شده از بیت کوین با یک ماه اخیر بیت کوین با تابع mape

و از گذشته اتریوم، الگوی زیر پیدا می‌شود:



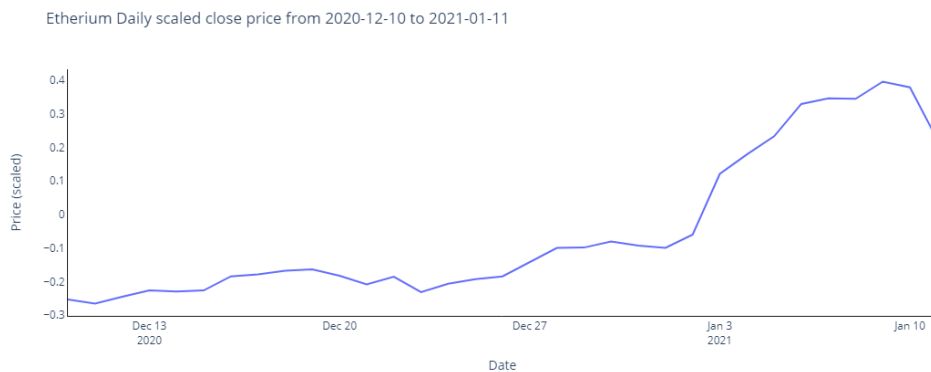
شکل ۱۱: الگوی مشابه پیدا شده از اتریوم با یک ماه اخیر بیت کوین با تابع **mape**

با تابع **correlation** از گذشته بیت کوین:



شکل ۱۲: الگوی مشابه پیدا شده از بیت کوین با یک ماه اخیر بیت کوین با تابع **correlation**

با تابع **correlation** و جستجو در گذشته اتریوم:



شکل ۱۳: الگوی مشابه پیدا شده از اتریوم با یک ماه اخیر بیت کوین با تابع **correlation**

همانطور که مشاهده می‌شود، با تابع correlation شباهت‌های بهتری پیدا می‌شوند. با پیدا نمودن این شباهت‌ها، می‌توان پیش‌بینی‌های مختلف انجام داده و همچنین الگوریتمی را که با خرید و فروش در آن الگو به سود ختم می‌شود بر روی الگوی کنونی اعمال نمود.

بخش چهارم: انتخاب ویژگی و کاهش ابعاد

به طور کلی در این پروژه، یک بار با روش kendall و یک بار با استفاده از روش انتخاب ویژگی MDI و random forest مدل‌ها را آموزش می‌دهیم. با توجه به دقت‌های به دست آمده، نحسی ابعاد منجر به دقت کمتری در مدل‌های حافظه‌دار شد.

در این قسمت با محاسبه ماتریس همبستگی بین ویژگی‌ها (رنک kendall)، ویژگی‌های با همبستگی بالای ۰.۹۵ کاندیدای حذف خواهند بود، که با توجه به اهمیت بعضی ویژگی‌ها مانند market cap و قیمت‌های نهایی در طبقه بندی، با اینکه همبستگی بالا برای آن‌ها به وجود آمده، اما فقط high_x, low_x, high و low حذف خواهند شد؛ زیرا با توجه به تست‌ها و مدل‌های مختلف آزمایش شده، اگر ویژگی‌های دیگر نیز حذف شوند عملکرد مدل‌ها ضعیف‌تر خواهد شد. در نهایت داده‌ها ۲۸ ویژگی خواهند داشت.

```
['high_x',  
 'low_x',  
 'close_x',  
 'market cap_x',  
 'high',  
 'low',  
 'close',  
 'market cap',  
 'ema_8',  
 'T3',  
 'low_5']
```

```
1 x.drop(['high_x' , 'low_x' , 'high' , 'low'], axis=1, inplace=True)
```

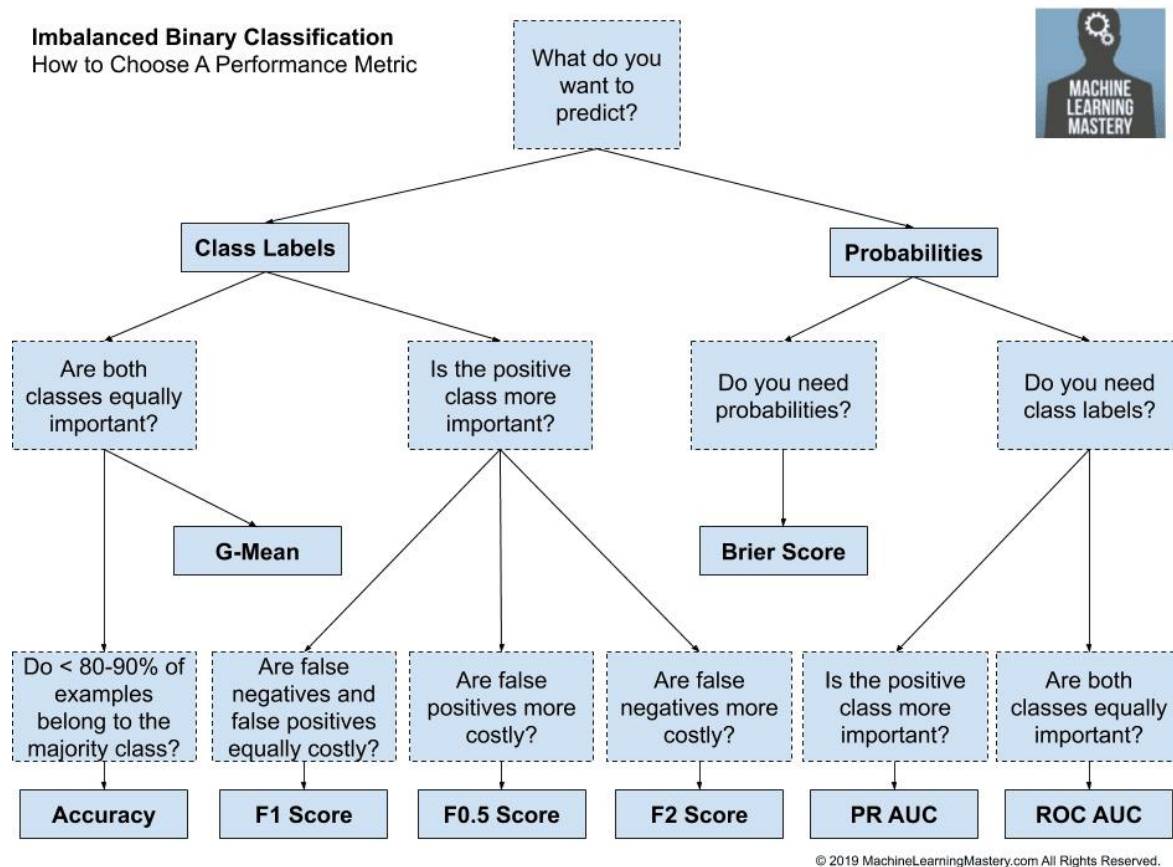
شکل ۱۴: حذف ویژگی‌ها به کمک correlation به صورت unsupervised

بخش پنجم: روش‌های طبقه‌بندی

مدل‌های استفاده‌شده را در ادامه نام‌برده و نتایج هریک را توضیح می‌دهیم.

لازم به ذکر است که ۹۰ درصد داده‌ها را به عنوان داده آموزشی و ۱۰ درصد را به عنوان داده تست انتخاب نموده‌ایم.

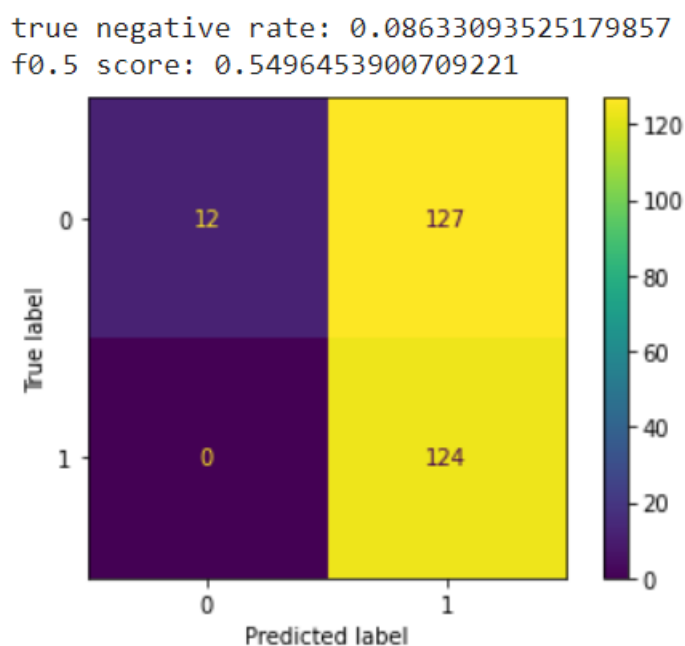
علاوه بر دقت، precision، recall و f1-score هر یک از مدل‌ها برای هر یک از کلاس‌ها نشان داده شده‌اند. با توجه به شکل زیر، مسئله طبقه‌بندی باینری و اهمیت بیشتر کلاس ۱، که در صورتی که درصد تشخیص آن بالاتر باشد، ضرر کمتری به همراه دارد، بنابراین از امتیاز f0.5-score استفاده می‌شود. همچنین معیار true negative rate (TNR) نیز اهمیت دارد؛ زیرا اگر در پیش‌بینی نزول‌ها درست عمل نشود می‌تواند منجر به ضرر شود. در نهایت precision و recall میانگین نیز مقایسه می‌شوند. معیار ROC AUC در نظر گرفته نشده است؛ زیرا کلاس‌ها تقریباً متعادل بوده و این عدد نزدیک به دقت خواهد بود.



شکل ۱۵: درخت انتخاب معیار مناسب

مدل ۱: Logistic Regression

از آنجا که ابعاد داده‌ها زیاد، تعداد داده‌ها به نسبت ابعاد کم بوده، و همچنین همبستگی بالا بین ویژگی‌ها وجود دارد، از این مدل که نسبت به بقیه مدل‌ها نیز ساده‌تر است، دقت بالا انتظار نمی‌رود. نتایج بدست آمده به صورت زیر است:



شکل ۱۶: ماتریس confusion، TNR و f0.5-score برای Logistic Regression

	precision	recall	f1-score	support
0	1.00	0.09	0.16	139
1	0.49	1.00	0.66	124
accuracy			0.52	263
macro avg	0.75	0.54	0.41	263
weighted avg	0.76	0.52	0.40	263

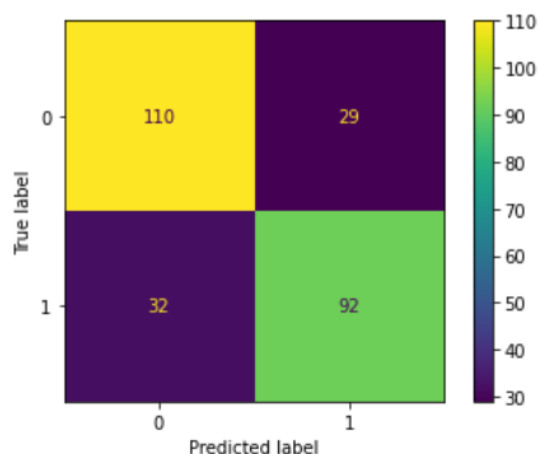
شکل ۱۷: گزارش طبقه بندی Logistic Regression

همانطور که مشاهده می‌شود دقت ۵۲ درصد بوده که دقت پایینی است؛ علاوه بر آن، f0.5-score نیز عدد پایینی حاصل شده است که پیش‌بینی اشتباه بالایی صعود، ریسک بالایی معامله را در پی دارد.

مدل ۲: SVM

مدل SVM برای داده‌های طبقه‌بندی باینری است. پس از بررسی انواع مختلف SVM، بهترین کرنل، sigmoid بدست آمده است. نتایج آن به صورت زیر است:

true negative rate: 0.7913669064748201
f0.5 score: 0.7565789473684211



شکل ۱۸: ماتریس confusion، TNR و f0.5-score برای SVM

	precision	recall	f1-score	support
0	0.77	0.79	0.78	139
1	0.76	0.74	0.75	124
accuracy			0.77	263
macro avg	0.77	0.77	0.77	263
weighted avg	0.77	0.77	0.77	263

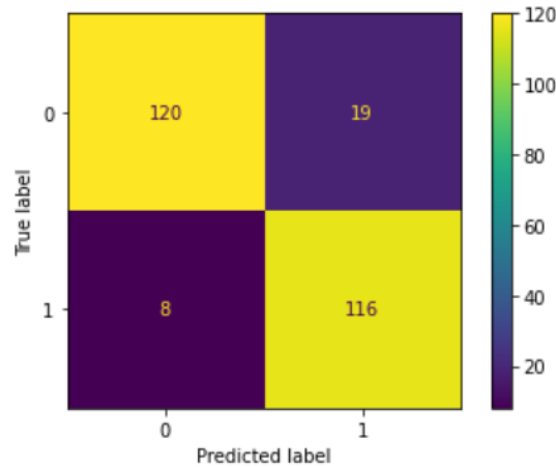
شکل ۱۹: گزارش طبقه‌بندی SVM

همانطور که مشاهده می‌شود، مقادیر true negative rate و f0.5-score به نسبت مدل قبل بهتر شده و قابل قبول هستند. مقادیر precision و recall متوسط نیز نزدیک به ۸۰ بوده که می‌تواند علاوه بر ضرر کم، امکان سود بالاتر را نیز بدهد.

مدل ۲: Decision Tree

نتایج این مدل به صورت زیر است:

true negative rate: 0.8633093525179856
f0.5 score: 0.8734939759036144



شکل ۲۰: ماتریس confusion، TNR و f0.5-score برای Decision Tree

	precision	recall	f1-score	support
0	0.94	0.88	0.91	139
1	0.87	0.94	0.90	124
accuracy			0.90	263
macro avg	0.91	0.91	0.90	263
weighted avg	0.91	0.90	0.91	263

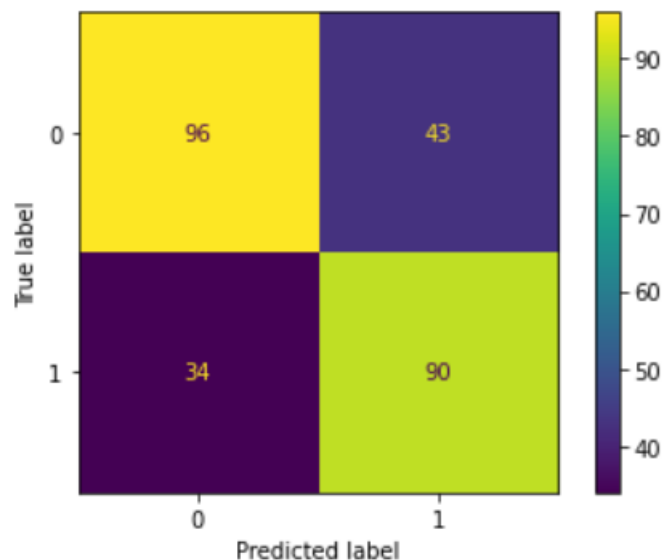
شکل ۲۱: گزارش طبقه‌بندی Decision Tree

این مدل برای طبقه‌بندی به خوبی آموزش داده شده و نتایج قابل توجهی داشته است که البته بیشتر ویژگی‌ها روی آن تأثیر داشته و برای مجموعه داده‌های بزرگ به خوبی عمل نخواهد نمود؛ زیرا بدست آوردن یک درخت تصمیم بهینه، یک مسئله np-complete است.

مدل ۳: KNN

مدل KNN یک مدل شناخته شده و قابل قبول برای داده‌های سری زمانی معرفی شده است. پس از بررسی همسایگی‌های مختلف در این مدل، تعداد همسایگی ۴ به یک مدل بهینه دست یافتیم:

true negative rate: 0.6906474820143885
f0.5 score: 0.6859756097560975



شکل ۲۲: ماتریس confusion, TNR و f0.5-score مدل KNN

	precision	recall	f1-score	support
0	0.74	0.69	0.71	139
1	0.68	0.73	0.70	124
accuracy			0.71	263
macro avg	0.71	0.71	0.71	263
weighted avg	0.71	0.71	0.71	263

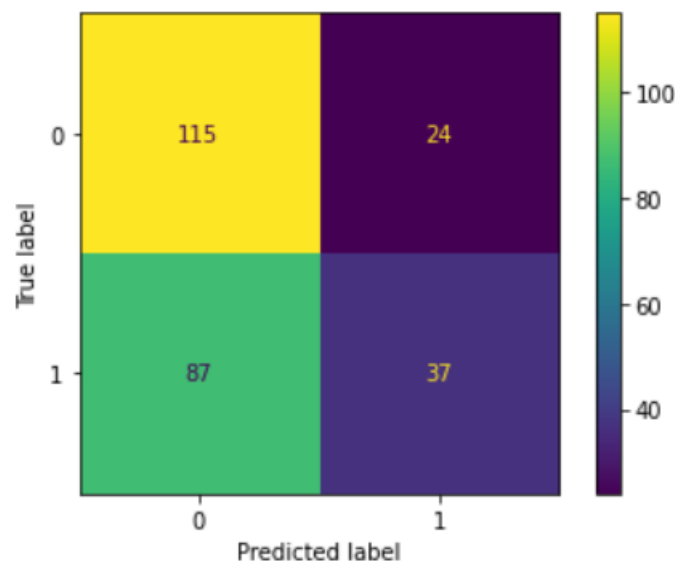
شکل ۲۳: گزارش طبقه‌بندی مدل KNN

همانطور که مشاهده می‌شود، این مدل بین decision tree و svm قرار می‌گیرد.

مدل ۴: BiLSTM

برای داده‌های سری زمانی، جهت پیدا نمودن الگوهای مناسب، می‌توان از مدل‌های شبکه عصبی حافظه دار استفاده نمود. البته در صورتی که تعداد ویژگی‌ها زیاد باشد، ممکن است به دقت بالایی نرسد، اما میزان overfitting یا underfitting آن نسبت به بقیه شبکه‌ها کمتر است. با توجه به دقت در شکل زیر، دقت ۵۸ بدست آمده که در آموزش نیز دقت آن حدود ۶۲ درصد بوده است.

true negative rate: 0.8273381294964028
f0.5 score: 0.5027173913043479



شکل ۲۴: ماتریس confusion، TNR و f0.5-score مدل BiLSTM

	precision	recall	f1-score	support
0	0.57	0.83	0.67	139
1	0.61	0.30	0.40	124
accuracy			0.58	263
macro avg	0.59	0.56	0.54	263
weighted avg	0.59	0.58	0.55	263

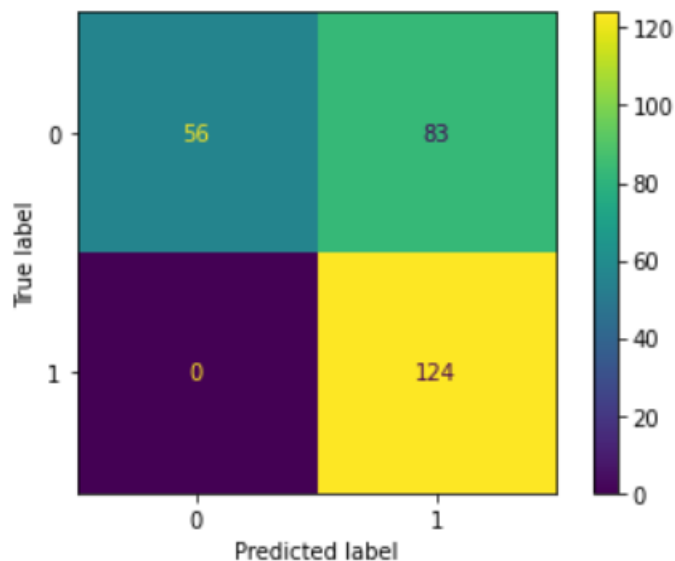
شکل ۲۵: گزارش طبقه‌بندی مدل BiLSTM

همانطور که مشاهده می‌شود، نکته مثبت این مدل، TNR بالا یعنی ۸۲ درصد است. اما با توجه به اهمیت بیشتر f0.5-score در ضرر، این عدد تقریباً با یک مدل ساده شانسی برابر شده است؛ پس قابل قبول نیست.

مدل ۵: Transformer

مدل‌های transformer که اخیراً مطرح شده‌اند، امروزه برای داده‌های سری زمانی نیز پیشنهاد و در مسائل مختلف استفاده می‌شوند. در این قسمت از مدل مناسب پیشنهاد شده برای سری زمانی از keras استفاده شده است که نتایج آن به صورت زیر است:

true negative rate: 0.4028776978417266
f0.5 score: 0.6512605042016807



شکل ۲۶: ماتریس confusion، TNR و f0.5-score برای مدل transformer

	precision	recall	f1-score	support
0	1.00	0.40	0.57	139
1	0.60	1.00	0.75	124
accuracy			0.68	263
macro avg	0.80	0.70	0.66	263
weighted avg	0.81	0.68	0.66	263

شکل ۲۷: گزارش طبقه‌بندی مدل transformer

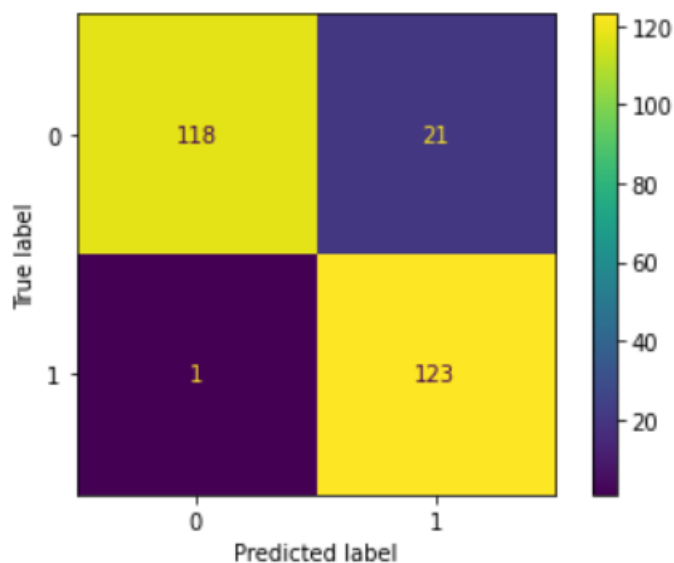
همانطور که مشاهده می‌شود، precision و recall بدست آمده تا حدی قابل قبول است. دقت آن نیز تا نزدیک به ۷۰ رسیده، و امتیاز f0.5 score آن حدود ۶۵ درصد است که نسبت به مدل BiLSTM بهتر بوده؛ اما همچنان از knn و svm کمتر است. این مدل به نمونه داده‌های بیشتر برای دقت بهتر نیاز دارد.

مدل ۶: CNN

مدل‌های CNN از مدل‌های قدرتمند در استخراج ویژگی‌ها و روابط مفید از داده‌ها هستند که می‌توانند نتایج خوبی را در مسائل طبقه‌بندی به همراه داشته باشند. از آنجا که داده‌ها سری زمانی هستند، از لایه‌های کانوولوشنی یک بعدی می‌توان استفاده نمود. نتایج بدست آمده به صورت زیر است:

true negative rate: 0.8489208633093526

f0.5 score: 0.8785714285714284



شکل ۲۸: ماتریس confusion, TNR و f0.5-score مدل CNN

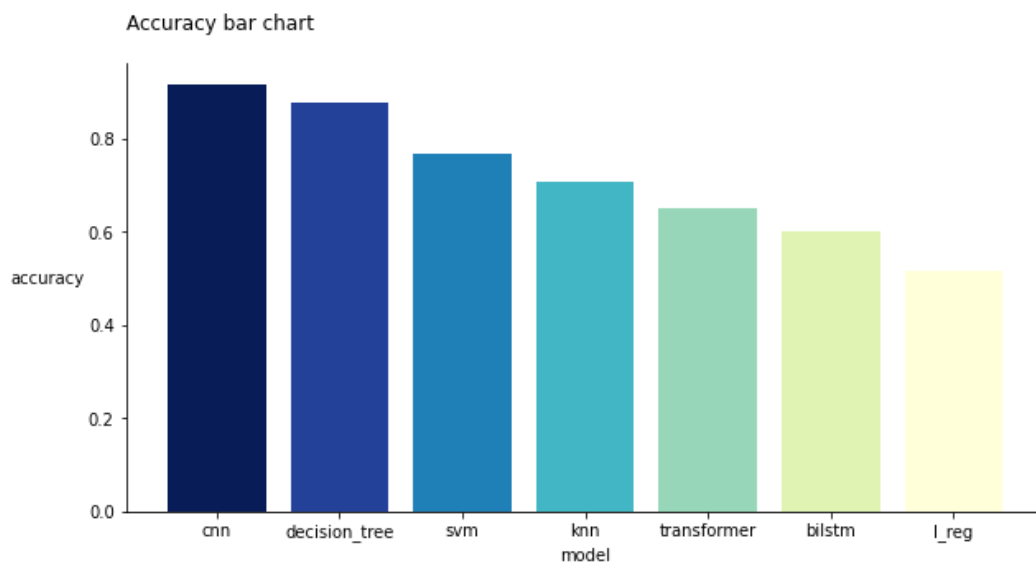
	precision	recall	f1-score	support
0	0.99	0.85	0.91	139
1	0.85	0.99	0.92	124
accuracy			0.92	263
macro avg	0.92	0.92	0.92	263
weighted avg	0.93	0.92	0.92	263

شکل ۲۹: گزارش طبقه‌بندی مدل CNN

همانطور که مشاهده می‌شود، به دقت ۹۲ درصد و f0.5-score بالا و نزدیک به ۹۰ درصد رسیده است. استفاده از این مدل ریسک ضرر کمی را به همراه خواهد داشت.

مقایسه مدل‌ها

در نمودار میله‌ای زیر، دقت همه مدل‌های استفاده شده مقایسه شده‌اند. نتیجه نمودار زیر نشان می‌دهد که مدل CNN با دقت ۹۲ درصد بهتر از همه عمل نموده و پس از آن مدل درخت تصمیم با دقت ۹۰ درصد بوده است. همچنین مدل رگرسیون ضعیف‌ترین عملکرد را داشته است.



شکل ۳۰: نمودار میله‌ای مقایسه دقت مدل‌ها با ۲۸ ویژگی

جدول زیر، یک نمای کلی از معیارهای مهم مدل‌ها و مقایسه آن‌هاست:

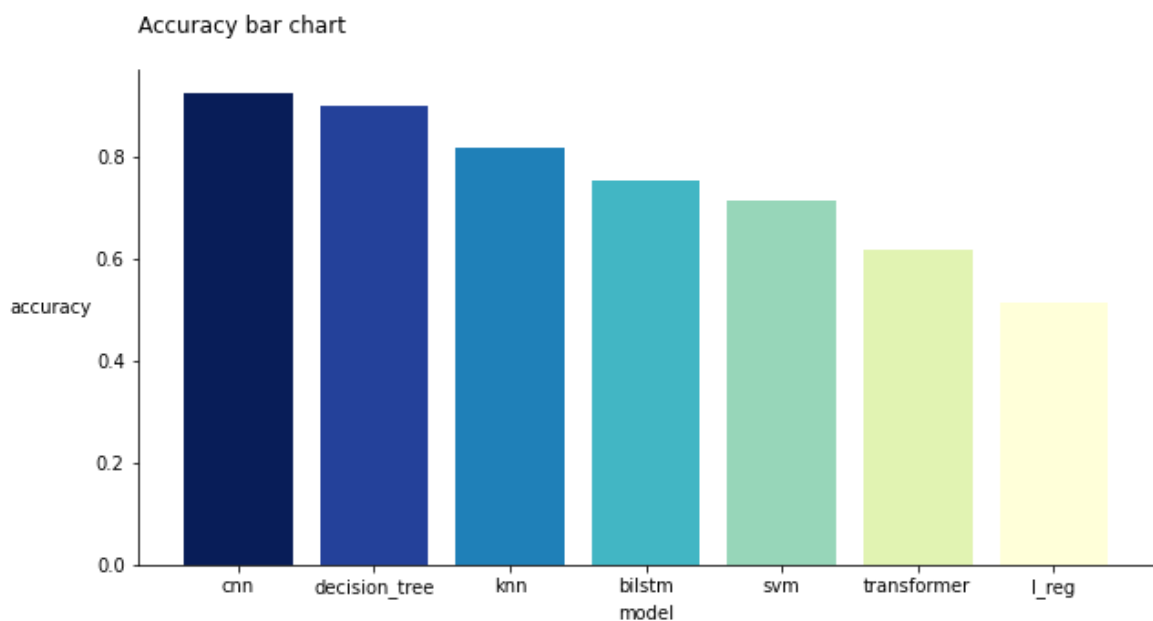
جدول ۱: مقایسه معیارهای مهم مدل‌ها و مقدار تحمل‌پذیری خطا

Metric Model	Precision	F0.5-Score	TNR	Accuracy	tol
Logistic Regression	0.76	0.54	0.09	0.52	
SVM	0.77	0.75	0.79	0.77	
Decision Tree	0.91	0.86	0.85	0.88	± 0.02
KNN	0.71	0.68	0.69	0.71	
Transformer	0.59	0.62	0.37	0.65	± 0.03
Bi-LSTM	0.81	0.48	0.79	0.57	± 0.03
CNN	0.93	0.86	0.85	0.90	± 0.03

این جدول نشان می‌دهد، با اینکه precision متوسط مدل‌ها بالا هستند، اما این معیار به تنهایی کافی نیست و f0.5-score و TNR نیز باید در نظر گرفته شوند.

استفاده از ویژگی‌های random forest

در این قسمت، از ۹ یا ۱۰ ویژگی مهم بدست آمده توسط random forest، استفاده نموده و مدل‌ها را مجدداً آموزش داده و مقایسه می‌کنیم.



شکل ۳۱: نمودار میله‌ای مقایسه دقت مدل‌ها با ویژگی‌های بدست آمده از random forest

همانطور که مشاهده می‌شود، CNN و درخت تصمیم در همان رتبه‌ها باقی مانده‌اند. مدل KNN با دقت ۸۲ درصد به رتبه سوم رسیده است، پس از آن مدل BiLSTM با دقت ۷۵ درصد در رتبه چهارم قرار دارد. مدل SVM عملکرد ضعیف‌تری نسبت به قبل، با ویژگی‌های بیشتر، داشته است. همچنان مدل رگرسیون از همه ضعیف‌تر با دقت ۵۱ درصد عمل نموده است.

جدول زیر مقایسه معیارهای مهم مدل‌ها را نشان می‌دهد:

جدول ۲: مقایسه معیارهای مهم مدل‌ها و مقدار تحمل‌پذیری خطا با ویژگی‌های بدست آمده از random forest

Metric Model	Precision	F0.5-Score	TNR	Accuracy	tol
Logistic Regression	0.76	0.54	0.07	0.51	
SVM	0.72	0.69	0.71	0.71	
Decision Tree	0.90	0.86	0.85	0.88	± 0.02
KNN	0.84	0.77	0.70	0.82	
Transformer	0.79	0.57	0.26	0.60	± 0.03
Bi-LSTM	0.76	0.72	0.71	0.77	± 0.03
CNN	0.93	0.88	0.87	0.91	± 0.03

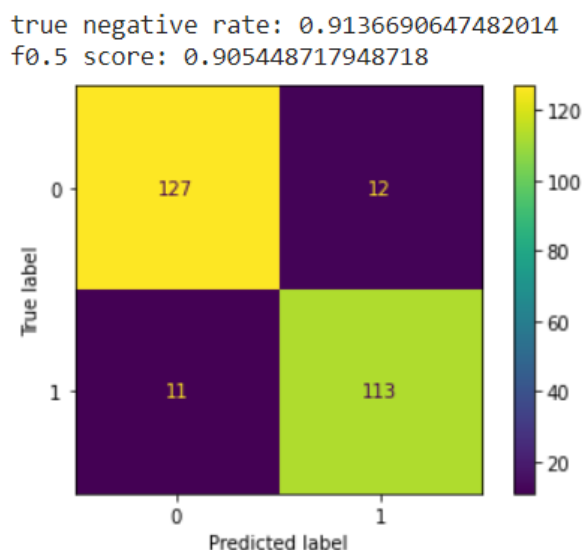
جدول بالا نشان می‌دهد که در سری زمانی، KNN و BiLSTM با ویژگی‌های کمتر و تأثیر بیشتر بهتر عمل می‌کنند؛ در حالی که SVM به تعداد ویژگی‌های بیشتری نیاز دارد. همچنین مدل transformer نیز به ویژگی‌های بیشتری نیاز دارد؛ البته برای این مدل در نمونه داده‌های بیشتر می‌توان دقیق‌تر نظر داد.

انتخاب و ensemble کردن مدل‌ها

در این قسمت، مدل‌های برتر انتخاب شده، و با استفاده از تکنیک voting، یک جواب نهایی بدست خواهد آمد. البته مدل Decision Tree در نظر گرفته نمی‌شود؛ زیرا در تعداد نمونه داده‌های بالا و داده‌های حجیم، بهینه عمل نمی‌کند و صرفاً جهت این است که نشان دهد در حدود چه دقتی می‌توان مدل‌های دیگر را آموزش داد.

- آموزش مدل SVM با ۲۸ ویژگی
- آموزش مدل KNN با ۹ یا ۱۰ ویژگی حاصل از random forest
- آموزش مدل CNN با ۹ یا ۱۰ ویژگی حاصل از random forest
- آموزش مدل BiLSTM با ۹ یا ۱۰ ویژگی حاصل از random forest

نتیجه بدست آمده از این رأی‌گیری به صورت زیر است:



شکل ۳۲: ماتریس confusion، TNR و f0.5-score مدل نهایی ensemble

	precision	recall	f1-score	support
0	0.92	0.91	0.92	139
1	0.90	0.91	0.91	124
accuracy			0.91	263
macro avg	0.91	0.91	0.91	263
weighted avg	0.91	0.91	0.91	263

شکل ۳۳: گزارش طبقه‌بندی مدل نهایی ensemble

مشاهدات و نتایج نشان دهنده عملکرد بهتر مدل رأی گیری است. نتیجه نهایی با ۵ بار اجرا به صورت زیر است:

جدول ۳: گزارش معیارهای مهم مدل نهایی

Metric Model	Precision	F0.5-Score	TNR	Accuracy	tol
Ensemble	0.91	0.89	0.89	0.89	± 0.03