**گزارش پروژه** 

**استاد:** دکتر عبادی

**دانشجویان:**

آرین آقامحسنی، کیانا خسرویان،‌ علیرضا یزدان‌پناه

400103132، 99103869، 400103776

**بهار۱۴۰۳**

**معرفی داده و طرح مساله**

هدف اصلی این پروژه پیش‌بینی وضعیت تغییر قیمت بیت‌کوین در روز بعد است. برای این منظور از رگرسیون لجستیک استفاده کرده‌ایم. تغییرات قیمت بیت‌کوین به دو دسته‌ی "افزایش" (Change Status=1) و "کاهش" (Change Status=0) تقسیم شده‌اند.

در بازار سرمایه، پیش بینی دقیق قیمت، به ویژه برای بازه های زمانی کوتاه، به دلیل رفتار تصادفی آن(شبیه Random Walk) امری ناممکن است. حال اگر از الگوریتم های ویژه سری های زمانی نیز بهره نبریم، حتی تخمین آن نیز دشوار خواهد بود. بنابراین هدف خود را در این بخش به پیش بینی افزایش یا کاهش قیمت برای روز بعد خواهیم گذاشت.

##### **توضیح داده‌ها**

داده‌های مورد استفاده در این پروژه شامل ویژگی‌های مختلفی از قیمت بیت‌کوین در دوره‌های زمانی متفاوت است. این داده‌ها از منابع معتبر مالی استخراج شده‌اند و شامل ویژگی‌هایی مانند قیمت باز شدن، بسته شدن، بالا و پایین‌ترین قیمت در روز، حجم معاملات، و سایر شاخص‌های مالی است.

### **بخش: معرفی متغیرها**

#### **متغیرهای اولیه**

* **Name**: نام دارایی (در این پروژه فقط بیت‌کوین مد نظر است)
* **time Open**: زمان باز شدن بازار
* **time Close**: زمان بسته شدن بازار
* **time High**: زمان رسیدن به بالاترین قیمت
* **time Low**: زمان رسیدن به پایین‌ترین قیمت
* **open**: قیمت باز شدن
* **high**: بالاترین قیمت
* **low**: پایین‌ترین قیمت
* **close**: قیمت بسته شدن
* **volume**: حجم معاملات
* **market Cap**: ارزش بازار
* **Daily Price**: قیمت روزانه
* **Circulating Supply**: عرضه در گردش
* **Date**: تاریخ
* **Daily Change**: تغییر روزانه قیمت
* **Change Status**: وضعیت تغییر قیمت (افزایش یا کاهش)

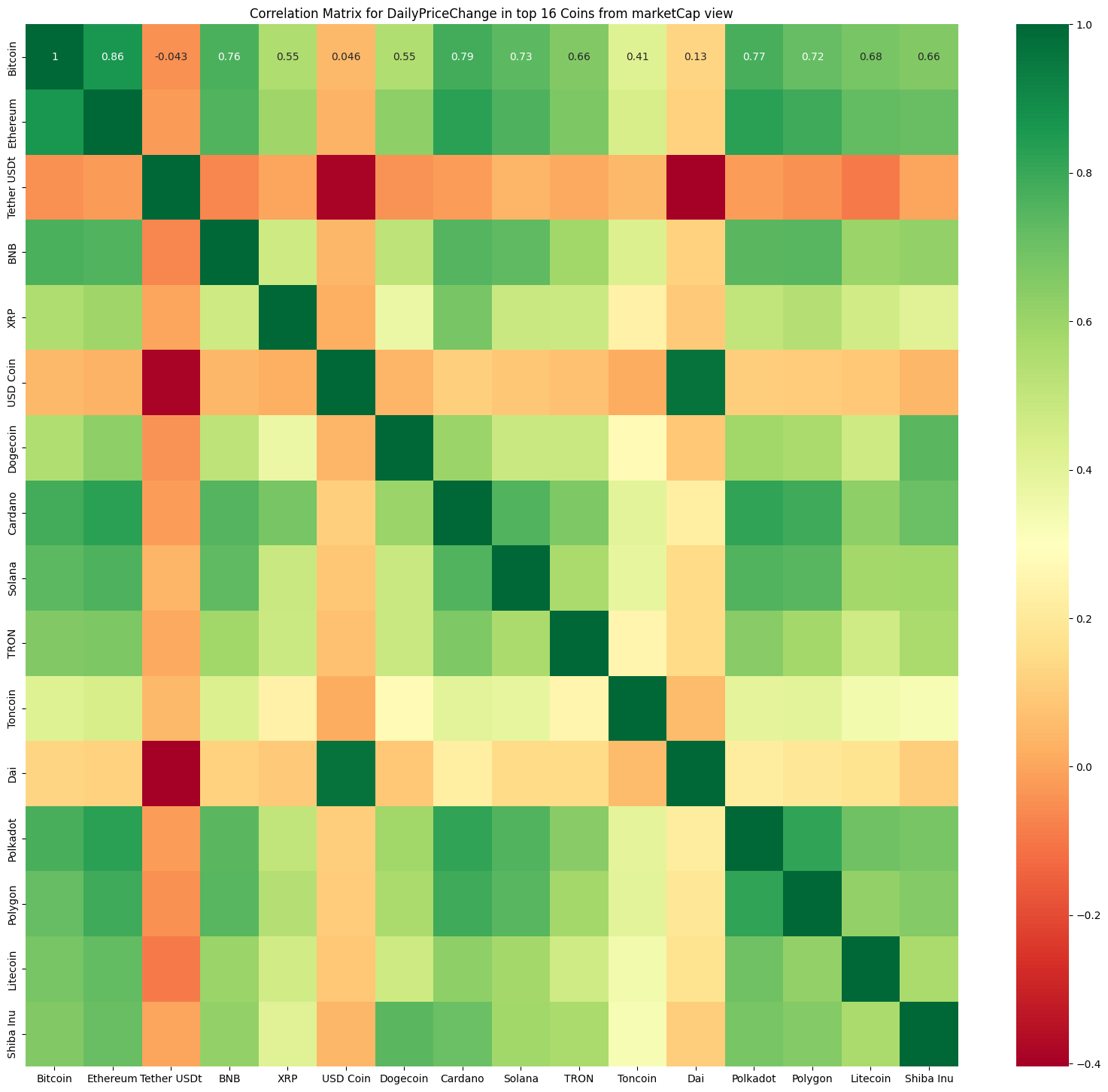
#### **متغیرهای ایجاد شده در مرحله‌ی مهندسی ویژگی‌ها**

* **Open\_hour**: ساعت باز شدن بازار
* **Close\_hour**: ساعت بسته شدن بازار
* **High\_hour**: ساعت رسیدن به بالاترین قیمت
* **Low\_hour**: ساعت رسیدن به پایین‌ترین قیمت
* **DailyReturn**: بازده روزانه محاسبه شده از تغییرات قیمت بسته شدن نسبت به روز قبل
* **HighLowSpread**: تفاوت بین بالاترین و پایین‌ترین قیمت
* **OpenCloseSpread**: تفاوت بین قیمت باز شدن و بسته شدن
* **Volatility**: نوسانات قیمت محاسبه شده با استفاده از انحراف معیار بازده روزانه در طول یک دوره‌ی هفت روزه
* **SMA\_7**: میانگین متحرک ساده ۷ روزه
* **SMA\_30**: میانگین متحرک ساده ۳۰ روزه
* **VolumeChange**: تغییرات حجم معاملات نسبت به روز قبل
* **MarketCapChange**: تغییرات ارزش بازار نسبت به روز قبل
* **RSI**: شاخص قدرت نسبی محاسبه شده بر اساس تغییرات قیمت بسته شدن در یک دوره‌ی ۱۴ روزه
* **EMA\_7**: میانگین متحرک نمایی ۷ روزه
* **EMA\_30**: میانگین متحرک نمایی ۳۰ روزه
* **Bollinger\_Mid**: میانگین متحرک ۲۰ روزه برای باند بولینگر
* **Bollinger\_Upper**: باند بالایی بولینگر (میانگین متحرک ۲۰ روزه به علاوه دو برابر انحراف معیار)
* **Bollinger\_Lower**: باند پایینی بولینگر (میانگین متحرک ۲۰ روزه منهای دو برابر انحراف معیار)

**آمار**

## **آمار توصیفی:**

پس از خواندن داده و بررسی‌های آماری پایه و اولیه مانند بررسی مقادیر تهی، ماتریس همبستگی بین ۱۶ coin که ارزش‌بازار(market cap) بیشتری را دارند را مورد بررسی قرار می‌دهیم تا ارتباطاتشان آشکار شود.



در ادامه برای بررسی نرمال بودن داده نمونه گیری انجام می‌دهیم که سکه‌های متفاوت را شامل می‌شود و روی هر کدام از این نمونه‌ها، بازه اطمینان ۹۸ درصدی تعریف می‌کنیم، که انتظار داریم میانگین نمونه‌ها نرمال باشند (با توجه به قضیه حد مرکزی و اینکه تعداد نمونه گیری ما نیز از ۳۰ بیشتر است).

دو تست Shapiro-Wilk test (که در واقع تست فرضی با فرض 0 نرمال بودن داده ها انجام داده و پی ولیو را برای بررسی برمیگرداند) و Kurtosis test (آزمون کورتوسیس برای بررسی تیزی یا پخ بودن توزیع داده‌ها نسبت به توزیع نرمال استفاده می‌شود. اگر مقدار کورتوسیس بزرگ‌تر از صفر باشد، نشان‌دهنده تیزی بیشتر توزیع و اگر کمتر از صفر باشد، نشان‌دهنده پخی بیشتر توزیع نسبت به توزیع نرمال است.) را روی داده‌ها برای نرمال بودن انجام می‌دهیم که در نتیجه آزمون‌هایمان متوجه غیر نرمال بودن داده می‌شویم.

برای رفع مشکل نرمال نبودن، دو ترنسفورمیشن Box-Cox transformation و Yeo johnson رو بر روی داده اعمال می‌کنیم و دوباره به بررسی نرمالیتی می پردازیم که تا حدی به بهبود نتایج مان کمک می‌کند.(توجه شود که ترنسفورمیشن Box-Cox نیاز دارد تمام داده هایمان مقادیر مثبت داشته باشند، و ترنسفورمیشن Yeo johnson رو دقیقا به همین دلیل وارد کردیم که نسبت به بقیه تاثیر بهتری داشت).

با توجه به threshold ها خیلی قابل قبول نیست ولی در دنیای واقعی تا حدی نرمال بودن را برای انجام t-test میپذیریم و در نهایت نرمال بودن این داده را که ۹۸ درصد داده داخل حدود اطمینان قرار دارند نتیجه می‌گیریم.

## **آزمون فرض:**

برای آزمون فرض اول دو موضوعیت را بررسی می‌کنیم که آیا volume و price در روز‌های هفته و روز‌های آخر هفته تفاوت میانگین قیمتی دارند.

در ابتدا داده روز‌های هفته و آخر هفته را جدا می‌کنیم و نرمال بودن آنها را چک می‌کنیم که هیچ‌کدام با توجه به شاخص هایمان به هیچ وجه نرمال نیستند(حتی با ترنسفورمیشن ها).

مجبور به استفاده از آزمون mann-whitney می‌شویم (که نیاز به نرمال بودن داده ها ندارد و همچنان صرفا وجود رابطه را چک میکند) و متوجه می‌شویم رابطه‌ای بین روز هفته یا آخر هفته بودن و تغییر قیمت و همچنین volume معامله وجود ندارد.

و در ادامه با استفاده از IQR تعدادی از داده حذف می‌کنیم (حذف داده های پرت باید با تعداد پایین بوده و باعث عوض شدن طبیعت داده نشود) و داده‌های پرت که حذف نمی‌کنیم همچنان نرمال نیستند و با وجود اعمال Yao-Johnson transformation همچنان نرمال نمی‌شوند، حال برای استفاده از box-cox transformation قدر مطلق تغییر قیمت را در نظر می‌گیریم، دوباره با غیر نرمالیتی مواجه می‌شویم و در نهایت با آزمون mann-whitney می‌توانیم بگوییم مقدار قدر مطلق تغییر قیمت با روز هفته رابطه دارد و در آخر هفته‌ها تغییرات قیمتی کمتری را شاهد هستیم.

و در آزمون فرض دوم بررسی می‌کنیم که اگر coin ها را به دو گروه تقسیم کنیم آیا دارای میانگین قیمت برابر هستند، با تکرار مراحل آزمون فرض قبل (تلاش برای نرمال کردن، اعمال ترنسفورمیشن ها، و استفاده از آزمون مربوط) به این نتیجه می‌رسیم که این میانگین در دو گروه ساختگی برابر است.

**رگرسیون**

برای بخش رگرسیون، روی یک کوین مشخص متمرکز خواهیم شد (بیت کوین)، در مرحله بعد به مهندسی ویژگی پرداختیم و یک سری از شاخص های بورس که می توانست در آینده به ما کمک کند را وارد داده ها کردیم.

در مرحله بعد به بخش EDA میرسیم، در این بخش به تحلیل نمودار های مالی از روی داده ها پرداختیم تا ضمن آشنایی نهایی با داده، بتوانیم در مرحله پیش پردازش ارزش هر پیش بینی کننده را بسنجیم. در انتهای این بخش ماتریس کورولیشن ها رو نیز بررسی میکنیم که دوباره برای قسمت feature selection کمک خواهد کرد.

مراحل پیش پردازش داده ها :

1 - نرمال سازی (همانطور که قبل دیدیم داده ها نرمال نبوده و باید با اعمال ترنسفورمیشن ها آنها رو نرمال میکردیم که در ابتدا این بخش از فایل آمار دیتای نرمال شده رو آوردیم)

2 - استاندارد کردن (استانداردسازی داده‌ها به این منظور انجام می‌شود که همه ویژگی‌ها در یک مقیاس مشترک قرار گیرند و تاثیر مقیاس‌های مختلف بر مدل کاهش یابد. در این پروژه از StandardScaler از کتابخانه sklearn استفاده شده است.)

3 - لیبل دار کردن متغیرهای indicator

4 - بررسی Multicollinearity (با استفاده از ترم های VIF)

در این بخش نرخ (VIF (Variance Inflation Factor رو برای تمام فیچر های داده محاسبه میکنیم که هست :

| با توجه به مقادیر VIF (که بالاتر از 10 غیر قابل قبول است)،  باید مقدار خوبی از فیچر هارو حذف کنیم،  که این به این معناست که هیچ کدام از این فیچر ها نمیتونن پیش بینی کننده خوبی برای ما باشند. |  |
| --- | --- |

5 - اضافه کردن ترم های Interaction

با توجه به دنیای فاینانس، اینترکشن های بین

'OpenCloseSpread', 'HighLowSpread'

'VolumeChange', 'MarketCapChange'

'DailyChange', 'Volatility'

میتوانند در معادله رگرسیون کمک کننده باشند

تغییرات روزانه و نوسانات قیمت می‌توانند تأثیر زیادی بر وضعیت بازار و تصمیم‌گیری سرمایه‌گذاران داشته باشند. تعامل بین این دو ویژگی می‌تواند اطلاعات بیشتری در مورد رفتار قیمت در روز بعد فراهم کند.

حجم معاملات و ارزش بازار دو عامل مهم در تحلیل بازارهای مالی هستند. تغییرات در این دو ویژگی می‌تواند نشان‌دهنده تحرکات مهمی در بازار باشد. تعامل بین تغییرات حجم معاملات و تغییرات ارزش بازار می‌تواند به شناسایی الگوهای خاصی که بر وضعیت قیمت بیت‌کوین تأثیر می‌گذارد، کمک کند.

اسپردهای قیمت باز و بسته شدن و بالاترین و پایین‌ترین قیمت می‌توانند نشان‌دهنده نوسانات و تغییرات مهم در طول روز باشند. تعامل بین این دو ویژگی می‌تواند به شناسایی روندهای روزانه و الگوهای نوسانی که ممکن است بر وضعیت تغییر قیمت بیت‌کوین تأثیر بگذارد، کمک کند.

تکنیک دیگری که در لجستیک رگرشن می توان به کار برد، متود Over/Under Sampling هست که برای دیتا ست ها با کلاس هایی با تعداد بسیار متفاوت استفاده می شود.

اما با توجه به طبیعت مساله، نمی تواند کمکی بکند.

**مدل**

برای این بخش چند مدل داریم :

1 - رگرسیون لجستیک با Statsmodels :

در مدل اول از کتابخانه‌ی Statsmodels برای اجرای رگرسیون لجستیک استفاده شده است. متغیرهای ورودی شامل تمام ویژگی‌ها به جز ChangeStatus هستند که بعد از فیلتر های VIF هستند :

(DailyChange, High\_hour, Low\_hour, HighLowSpread, OpenCloseSpread, Volatility, VolumeChange, MarketCapChange, RSI)

و متغیر خروجی ChangeStatus است.

برای اجرای مدل با این داده ها به مشکل Singular Matrix برخوردیم.

ماتریس تکین یا ماتریس تک‌بعدی، ماتریسی است که دترمینان آن برابر با صفر است. این به این معناست که این ماتریس معکوس‌پذیر نیست و بنابراین نمی‌توان برخی محاسبات را بر روی آن انجام داد، مانند یافتن معکوس ماتریس.

مشکل ماتریس تکین معمولا به دلایل زیر رخ می‌دهد :

1 - وجود Perfect Separation در داده ها

2 - وجود ستون های با واریانس 0

3 - وجود ستون های تکراری

که در مساله ما مشکل perfect separation وجود دارد، بعد از حل این مشکل و حذف متغیر هایی که باعث شدند، تعداد زیادی از ستون ها حذف میشوند.

تحلیل و تفسیر خروجی ها :

ماتریس درهم‌ریختگی (Confusion Matrix):

مدل 35 مورد از 38 کاهش قیمت را به درستی پیش‌بینی کرده (True Negative) و 3 مورد را به اشتباه افزایش پیش‌بینی کرده است (False Positive).

مدل 28 مورد از 29 افزایش قیمت را به درستی پیش‌بینی کرده (True Positive) و 1 مورد را به اشتباه کاهش پیش‌بینی کرده است (False Negative).

نمره ROC AUC:

مقدار 0.968 نشان‌دهنده‌ی عملکرد بسیار خوب مدل در تفکیک بین کلاس‌های 0 و 1 است. هرچه این مقدار به 1 نزدیک‌تر باشد، مدل عملکرد بهتری دارد.

نمره R2:

مقدار 0.756 نشان‌دهنده‌ی این است که مدل حدود 75.6% از واریانس داده‌ها را توضیح می‌دهد.

نمره‌ی اعتبارسنجی متقاطع:

میانگین نمره‌ی اعتبارسنجی متقاطع (دقت) برابر 0.937 است که نشان‌دهنده‌ی پایداری مدل در مجموعه‌های داده‌ی مختلف است.

2 - رگرسیون لجستیک (مدل 2 رگرسیون لجستیک بدون Interaction و مدل 3 با Interaction) :

در مدل‌های دوم و سوم از کتابخانه‌ی Scikit-learn برای اجرای رگرسیون لجستیک استفاده شده است. مدل دوم شامل ویژگی‌های پایه‌ای و مدل سوم شامل ویژگی‌های تعاملی است.

مدل دوم شامل ویژگی‌های پایه‌ای بود و مدل سوم شامل ویژگی‌های تعاملی. هر دو مدل عملکرد مشابهی نشان دادند و دقت بالایی داشتند. این نشان می‌دهد که اضافه کردن ویژگی‌های تعاملی تأثیر زیادی بر دقت کلی مدل نداشته است.

گزارش طبقه‌بندی:

Precision: دقت در پیش‌بینی افزایش (کلاس 1) برابر 94% و در پیش‌بینی کاهش (کلاس 0) برابر 96% است. این نشان می‌دهد که مدل‌ها در پیش‌بینی هر دو کلاس به خوبی عمل کرده‌اند.

Recall: فراخوانی برای کلاس 1 برابر 96% و برای کلاس 0 برابر 94% است. این بدان معناست که مدل‌ها توانسته‌اند اکثر تغییرات قیمت را به درستی شناسایی کنند.

F1-Score: میانگین وزنی precision و recall برای هر دو مدل برابر 95% است که نشان‌دهنده‌ی تعادل خوبی بین دقت و فراخوانی است.

ماتریس درهم‌ریختگی (Confusion Matrix):

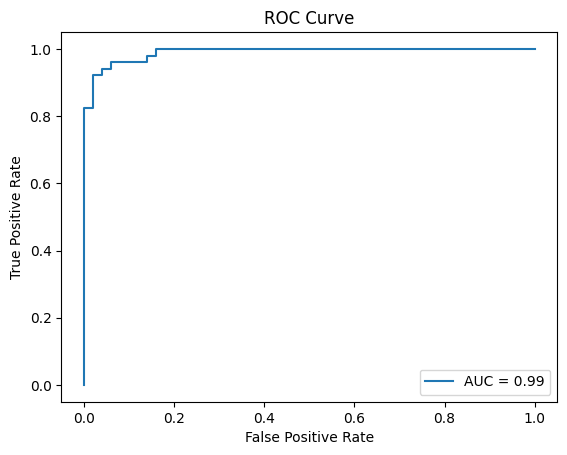
هر دو مدل 47 مورد از 50 کاهش قیمت را به درستی پیش‌بینی کرده (True Negative) و 3 مورد را به اشتباه افزایش پیش‌بینی کرده‌اند (False Positive).

هر دو مدل 49 مورد از 51 افزایش قیمت را به درستی پیش‌بینی کرده (True Positive) و 2 مورد را به اشتباه کاهش پیش‌بینی کرده‌اند (False Negative).

نمودار ROC:

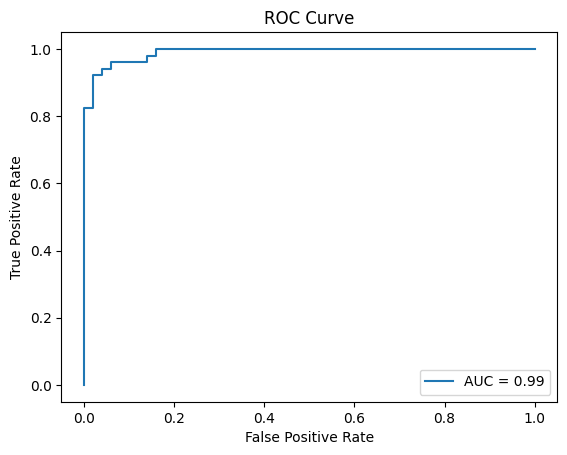
هر دو مدل دارای عملکرد مشابهی در نمودار ROC هستند که نشان‌دهنده‌ی قابلیت تفکیک خوب بین کلاس‌های 0 و 1 است.

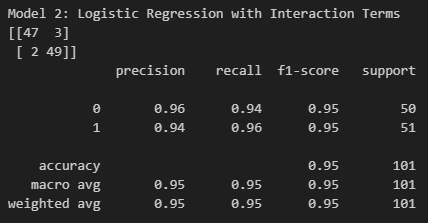
برای مدل 2 :





برای مدل 3 :





انتخاب مدل نهایی (Model Selection)

با توجه به نتایج به دست آمده از سه مدل مختلف، تحلیل و تفسیر معیارهای مختلف انجام شده است. معیارهای اصلی برای انتخاب مدل نهایی عبارتند از:

دقت (Accuracy):

مدل ۱: 94%

مدل ۲: 95%

مدل ۳: 95%

گزارش طبقه‌بندی:

مدل ۱: دقت بالای 90% برای هر دو کلاس، با فراخوانی بالای 92% برای کلاس 0 و 97% برای کلاس 1.

مدل ۲ و ۳: دقت بالای 94% برای هر دو کلاس، با فراخوانی بالای 94% برای هر دو کلاس.

نمره ROC AUC:

مدل ۱: 0.968

مدل ۲ و ۳: مشابه

نمره R2:

مدل ۱: 0.756

نمره‌ی اعتبارسنجی متقاطع:

مدل ۱: 0.937

با توجه به این معیارها، مدل‌های ۲ و ۳ دقت و عملکرد مشابهی دارند و هر دو عملکرد بسیار خوبی در پیش‌بینی تغییرات قیمت بیت‌کوین نشان داده‌اند.

نتیجه‌گیری

با در نظر گرفتن نتایج مدل‌ها و معیارهای مختلف، مدل‌های ۲ و ۳ به دلیل دقت بالاتر و عدم وجود مشکلات ماتریس تکین، گزینه‌های بهتری برای پیش‌بینی تغییرات قیمت بیت‌کوین هستند. هرچند مدل ۱ نیز عملکرد خوبی داشته است، اما به دلیل مشکلات ماتریس تکین و پیچیدگی بیشتر، مدل‌های ۲ و ۳ ترجیح داده می‌شوند.

در نهایت، مدل ۳ با ویژگی‌های تعاملی انتخاب می‌شود زیرا علاوه بر دقت بالا، توانایی بیشتری در کشف روابط پیچیده بین ویژگی‌ها دارد و می‌تواند به بهبود پیش‌بینی‌ها کمک کند.

مدل نهایی انتخاب شده: رگرسیون لجستیک با ویژگی‌های تعاملی (مدل ۳)

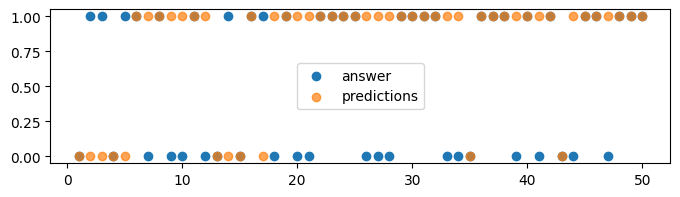
نتایجی که گرفتیم در واقع براساس high و low hour هست که این از انتظارات ما (خودم به شخصه فکر نمیکردم این 2 فیچر زیاد اهمیتی داشته باشند) را کمی تغییر داد :) (البته که وجود لیکیج هم ممکن هست)

می توان تا حدی این موضوع رو نداشتن پیچیدگی لازم مدل های رگرسیون نسبت داد.

در ادامه تحقیقی بیشتر با استفاده از متود های پیشرفته ماشین لرنینگ و دیپ لرنینگ (از جله LSTM و …)

**مدل های اضافه**

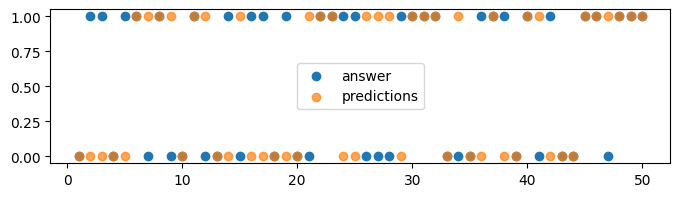
1 - مدل Random Forest Classifier



F1 Score : 0.69

ROC Score : 0.55

2 - مدل K Neighbor Classifier :



F1 Score : 0.55

ROC Score : 0.52

از بین این 2 مدل، رندوم فارست بهترین مدل ماست پس اگر این مدل رو روی داده تست پیاده کنیم :

F1 Score : 0.73

ROC Score : 0.6

و خب دلیل این اختلاف در معیار های سنجش این است که دیتای تست در زمان (و در نتیجه ترند) دیگری از بازار بوده، بنابراین توزیع مقادیر 0 و 1 متفاوت خواهد بود.

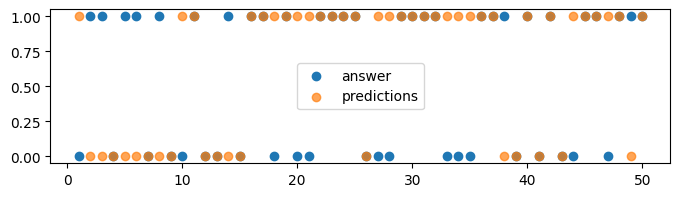
حال میتوان سیستم Back Testing رو هم امتحان کرد

بنابراین، اگر از بازه‌ی 30 روزه به عنوان روند استفاده کنیم، می‌توانیم روند بعدی را با دقت حدود 74٪ برای هر دو کلاس (1 و 0) پیش‌بینی کنیم. اما اگر سعی کنیم این کار را در میان روندها و به صورت گام به گام مانند مدل بک‌تستینگ خود انجام دهیم، شاهد کاهش دقت خواهیم بود زیرا مدل قادر به شناسایی موقعیت خود در روند مربوطه نیست.

بنابراین، دقت سیستم Back testing ما برای هر دو کلاس (1 و 0) حدود 67٪ خواهد بود.

با توجه به اینکه رندوم فورست بهترین مدل ما بود، تلاش میکنیم تا عملکردش رو بهتر کنیم

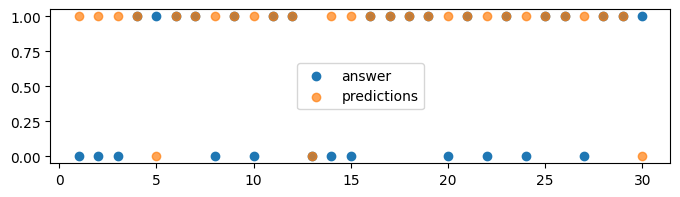
3 - مدل AdaBoostClassifier :



F1 Score : 0.66

ROC Score : 0.58

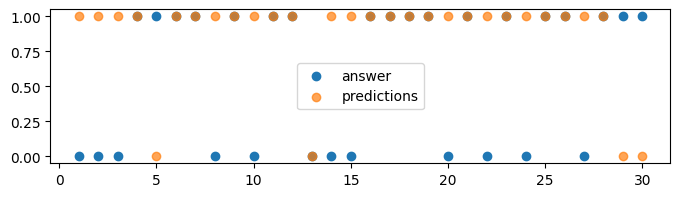
بعد از اعمال روی داده های تست خواهیم داشت :



F1 Score : 0.71

ROC Score : 0.48

و بعد از سیستم بک تستینگ :



F1 Score : 0.68

ROC Score : 0.45

همان‌طور که مشاهده می‌کنید، مدل AdaBoost با بک‌تستینگ سازگاری بیشتری دارد که این امر معیار بهتری برای امتیاز F1 ما است، زیرا توزیع بهتری از کلاس‌های 1 و 0 دارد.

در نهایت می‌توانیم بگوییم که مدل RandomForest ما در پیدا کردن کامل روند با امتیاز F1 حدود 74٪ بهتر عمل می‌کند و همچنین امتیاز F1 در بک‌تستینگ آن نیز حدود 67٪ است که نتیجه خوبی است.

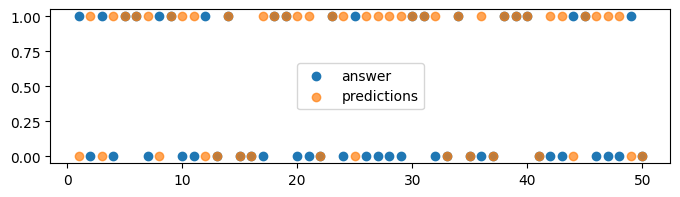
همچنین مدل AdaBoost ما در بک‌تستینگ کمی بهتر عمل کرده و امتیاز F1 بیش از 68٪ دارد.

سوالی که من به شخصه همواره برای من سوال بوده همیشه

سؤال جدید این است: «آیا بیت‌کوین نسبت به اخبار جهانی و نظرات مردم حساس است؟»

حالا ما سعی خواهیم کرد از تحلیل احساسات بر روی صفحه ویکی‌پدیای بیت‌کوین برای پاسخ به این سؤال استفاده کنیم.

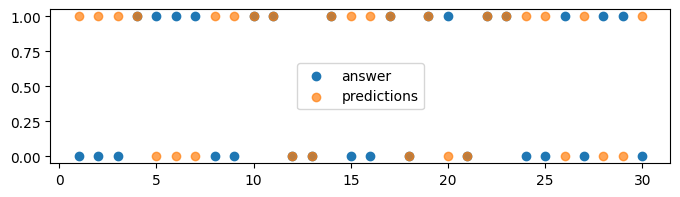
بعد از انجام دادن پری پروسسینگ ها و پر کردن داده های خالی، حال از مدل AdaBoost با سیستم Backtesting آن استفاده میکنیم :



F1 Score : 0.51

ROC Score : 0.49

بعد از بک تست :



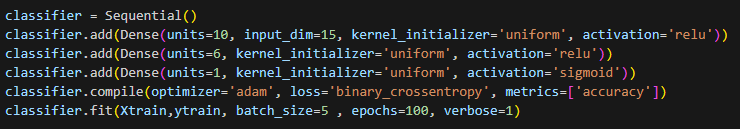
F1 Score : 0.47

ROC Score : 0.4

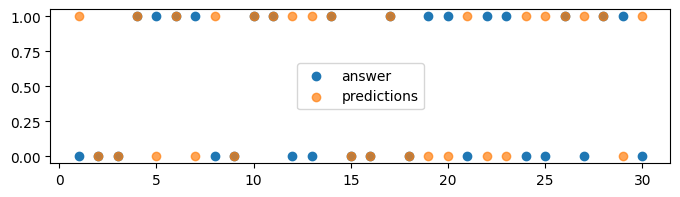
همان‌طور که مشاهده می‌کنید، بیت‌کوین و منبع خبری که انتخاب کرده‌ایم اشتراک زیادی ندارند و بنابراین نتیجه ضعیف‌تر از حتی توزیع تصادفی کلاس‌های 1 و 0 است!

میتوان برای تحلیل احساسات از مدل های قوی تری هم استفاده کرد :

مدل جدید یک شبکه عصبی با این گزارش است :



و برای این مدل داریم :



F1 Score : 0.5

ROC Score : 0.48

حتی تلاش با استفاده از شبکه‌های عصبی نیز کمک زیادی به ما نکرد، بنابراین به این نتیجه می‌رسیم که منبع ما (بازبینی‌های ویکی‌پدیا) ارتباط مناسبی با قیمت بیت‌کوین ندارد.