Amirkabir University of Technology
(Tehran Polytechnic)

پروژه دوم: شرح پیش بینی قیمت کریپتوکارنسی (بیتکوین)

استاد درس:
دکتر محمدپور
(adelm@aut.ac.ir)

تدریس یار:

المیرا طالبی
(elmiratelibianaraki@gmail.com)

دانشجو:

صبا صادقی - محمدرضا اردستانی
(ardestani.zm@gmail.com), (sabasedegi@gamil.com)

20, June, 2021

0) **Introduction and purpose (مقدمات و هدف)**

- 0.1) physiology of the market
- 0.2) Purposes of this project
- 0.3) Data set

Part 1) **Traditional Market analysis (روش های مرسوم سنتی برای تحلیل مارکت)**

- 1.1) Technical analysis
- 1.2) Fundamental analysis
- 1.3) Sentiment analysis

Part 2) **Using statistical methods (AR,MA, ARMA, ARIMA , SARIMAX) (روش های آماری)**

- 2.1) explaining our step-by-Step statistical approach
  - 2.1.1) about time series and stationarity components
  - 2.1.2) steps
- 2.2) AR method
- 2.3) MA method
- 2.4) ARMA method
- 2.5) ARIMA method
- 2.6) SARIMAX method

part 3) **Using Neural Networks methods (LSTM) (روش شبکه های عصبی)**

- 3.0) Explaining our step-by-Step NN approach
- 3.1) About LSTM
- 3.2) Results

part 4) **Conclusion and challenges (چالشها , نتیجه گیری)**

- 4.1) Conclusion
  - 4.1.1) Comparing the result with stats models
  - 4.1.2) Time and space complexity
- 4.2) Challenges
- 4.3) Future works

Part 5) **References (منابع و ماخذ)**

Part 6) **Appendix (متصلات)**

0) **Introduction and purpose (مقدمات و هدف)**

در ابتدا نگاهی کلی به اهداف و رویکرد پروژه خواهیم کرد و سپس روش های حل و داده کاوی انتخاب شده به تفصیل شرح داده خواهد شد.

0.1) physiology of the market

در این زیرشاخه به توضیحی درباره ماهیت (فیزولوژی) مارکت میپردازیم. این بازار ارزهای دیجیتال 24/7/365 است به این معنا که تعطیلی ندارد و در تمام ساعت روز هفته و سال مردم میتوانند مبادله انجام دهند. همچنین به دلیل وابسته بودن آن به عوامل مختلف (که تقریبا غیرقابل پیش بینی) هستند، ما seasonality و cycle ها از پیش مشخص (مانند داده های مسافران هواپیمایی) نداریم. همچنین دستکاری در این مارکت بسیار زیاد است و رسانه های خبری و influencer ها میتوانند روند پیش بینی شده توسط خیلی از کارشناسان را تغییر دهند و تقاضای خرید و یا فروش ایجاد کنند. مسئله ای که ما در داده هایی مانند داده های مسافران هواپیمایی به ندرت مواجه نمیشویم.

0.2) Purposes of this project

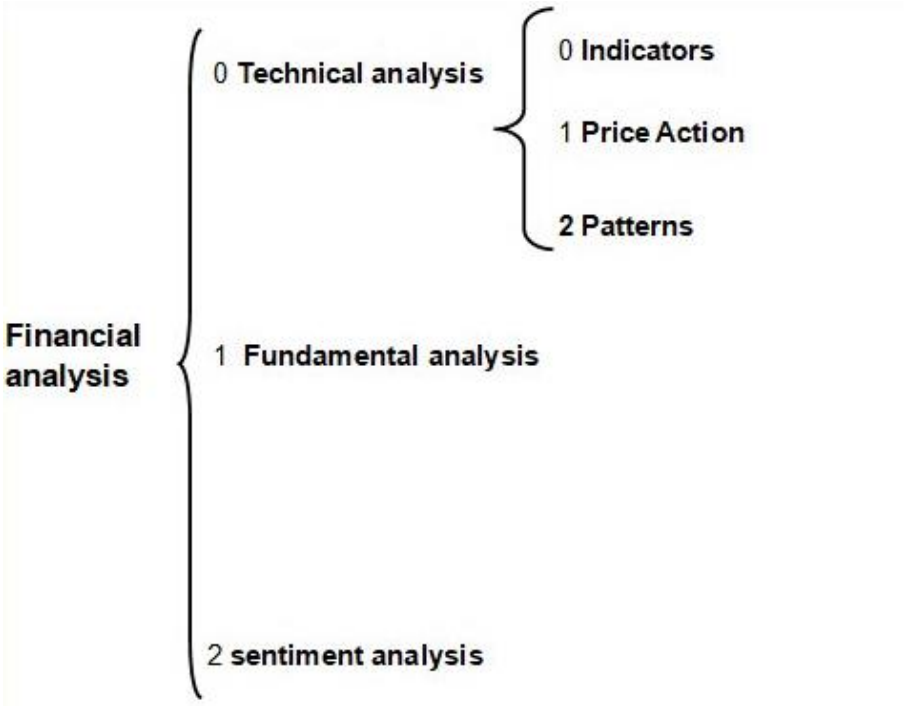
امروزه بخش زیادی از بازارهای مالی توسط بات های کامپیوتری اداره و خرید و فروش صورت میگیرد. برای نمونه در بازارهای مالی آمریکا 80 درصد معاملات اتوماتیک و بدون دخالت انسان صورت میگیرد. هدف ما در این پروژه این است که به اشخاصی که در این بازار قصد سرمایه گذاری را دارند با استفاده از سیستم پیش بینی کننده خود سیگنال بدهیم ، اما این کار برای عده ی محدودی خواهد بود.

0.3) Data set

برای انجام تحلیل خود API Yahoo Finance استفاده شده است. شما با این API به انواع مختلف فرم هایی که داده ها ذخیره شده اند (دقیقه ای، ساعتی، روزانه، ماهانه و ...) دسترسی دارید. آدرس فایل های استفاده شده در بخش منابع آمده است و نیز شما میتوانید نمونه کد برای دسترسی گرفتن به APIرا در بخش Appendix (1) ملاحظه کنید.

Part 1) **Traditional Market analysis (روش های مرسوم سنتی برای تحلیل مارکت)**

برای انجام تحلیل، روش های سنتی وجود دارد که اکثر آن ها هنوز هم جزء بهترین روش های پیش بینی قیمت سهام میباشند.



در بازار کریپتوکارنسی ها یک شاخه دیگر هم به طور اختصاصی وجود دارد. در واقع در بخش Technical analysis یک شاخه دیگر به نام On-chain analysis نیز وجود دارد.علت آن این است که اکثر رمزارز ها decentralized هستند و همه به تمام اطلاعات روی شبکه دسترسی دارند و بررسی کردن این اطلاعات را On-chain analysis می نامیم.

در پروژه ای موازی این درس تمام سه شاخه بالا پیاده سازی شده و قابل دسترسی است.

Part 2) **Using statistical methods (ARIMA, EMA, SARIMAX)** **(روش های آماری)**

در این روش های آماری اتخاذ شده را بررسی کرده و در معیارهای بهینگی آن ها (AIC) آن ها مقایسه میشود. همچنین درباره متد گام به گامی که در پیش گرفته ایم توضیح داده خواهد شد.

2.1) Explaining out step-by-Step statistical approach

داده های سری زمانی به زمان وابسته اند و قابل تکرار نیستند و هدف ما ممکنه پیش بینی **گذشته** باشد اما معمولا آینده را پیش بینی میگویند.

روش ما روش پارامتری است به طور کلی اکثر روش های آماری پارامتری هستند، ینی توزیع داده ها و یا توزیع خطاهای خودمان را میدانیم.

برای این که مدل برازش بدیم چند مرحله باید طی شود:

1) مولفه های نا ایستایی در سری های زمانی را پیداکنیم. ترند ، سیزونالیتی ، نا ایستایی در واریانس و سیکل.

ناایستایی در واریانس ، زمان که زیاد میشود آن وقت داده ها از هم فاصله میگیرند و واریانسشون زیاد میشه (در یک قیف قرار میگیرند)

2) اولین کار رسم نسبت به زمان

3) با انجام کارهای یک ، ما یک سری زمانی ایستا به دست میآوریم ، بعد از آن میتوانیم ما مدل برازش بدهیم .

- مدل باکسو جنکینز نحوه ایستا سازی داده ها رو خیلی دقیق تر از بقیه بیان کرده اند.
- سوال چگونه میتوان سیزونالیتی را به دست آورد؟؟ زمان قدیم با رسم نمودار و دستی و سعی و خطا / امروزه با دستور decompose

4) یک کار مهم این است که یک سامری از داده ها بگیریم تا متوجه شویم که آیا داده گم شده داریم یا خیر / اگر داده گم شده داشته باشیم با روش هایی مثل جایگزینی با میانگین و یا kkn imputation and call man filter استفاده میکنیم.

5) در شکل مسافران هوایی داده ها توی قیف قرار میگیرند پس ما ناایستایی در واریانس رو داریم. سیزونالیتی مشخصا داریم ، ترند هم داریم.

6) اگر ناایستایی در واریانس داشته باشیم مدل ما از نوع ضربی خواهد بود .

7) میتوانیم تست ایستایی را هم انجام دهیم ، وقتی استفاده میکنیم که شک داشته باشیم.

اسم تست آن Dicky-fuller هست

8) راه دیگر برای تست stationary بودن تست کرولوموگرام هست. (Auto correlation function and Partial auto correlation function)

9) چه داده های ما ایستا باشند و چه نا ایستا ما باید مولفه های نا ایستایی را بشناسیم. / حالا وقتی ناایستایی داریم ان مولفه ها را حذف میکنیم و وقتی نداریم باید آن ها را بشناسیم که مدل چنتا پارامتر پاس بدهیم .

- ARIMA روی داده های اصلی فیت میشه و ARMA روی داده های ایستاشده و به اصطلاح رندوم.
- ARMA has only 2 parameters , autoregressive show number of Q (phi) and moving average part shows number of (theta)
- اما برای آریما ما باید میدانستیم که این داده ها ترند دارند و یا ندارند.
- پارامتر اول : مرتبه آوتو رگرسیو ، پارامتر دوم: ترند ، سومی مرتبه موویِنگ اوریج هست
- ARIMA: Auto Regressive Integrated with Moving Average
- AIC هرچه کمتر باشد بهتر است.

اگر میخواستیم نا ایستایی در واریانس رو از بین ببریم و داده ها توی قیف قرار نگیرند که بعد بیاییم و مدل جمعی برازش بدیم میتونستیم از داده ها **لگاریتم** بگیریم

مدل neural auto regressive ترکیبی از این مدل های آماری و شبکه های عصبی هست.

2.2) AR method

واژه AR به معنای Auto Regressive میباشد. این متد بدین صورت کار میکند که شما مشاهدات پله های قبلی خود را به مدل رگرسیونی خود feed میکنید و به دین صورت مقدار های آینده را پیش بینی میکنید.

بعد از طی تمام مراحل پیش پردازشی که در بخش 2.1 ذکر کردیم، مدل خود را fit میکنیم و خروجی به صورت زیر خواهد بود:

```
==============================================================================
Dep. Variable:                D.Close   No. Observations:                   80
Model:                 ARIMA(2, 1, 0)   Log Likelihood                10.533
Method:                       css-mle   S.D. of innovations              0.212
Date:                Wed, 16 Jun 2021   AIC                            -13.067
Time:                        14:23:31   BIC                             -3.539
Sample:                    11-01-2014   HQIC                            -9.247
                         - 06-01-2021
==============================================================================
                    coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const             0.0595      0.028      2.106      0.038       0.004       0.115
ar.L1.D.Close     0.1322      0.111      1.188      0.238      -0.086       0.350
ar.L2.D.Close     0.0307      0.115      0.266      0.791      -0.195       0.257
                                    Roots
=============================================================================
                  Real          Imaginary           Modulus         Frequency
-----------------------------------------------------------------------------
AR.1            3.9482           +0.0000j            3.9482            0.0000
AR.2           -8.2570           +0.0000j            8.2570            0.5000
-----------------------------------------------------------------------------
```

در این مدل AIC عدد منفی 13 شده است که هر چه قدر این عدد کمتر باشد مدل بهینه تر میباشد.

2.3) MA method

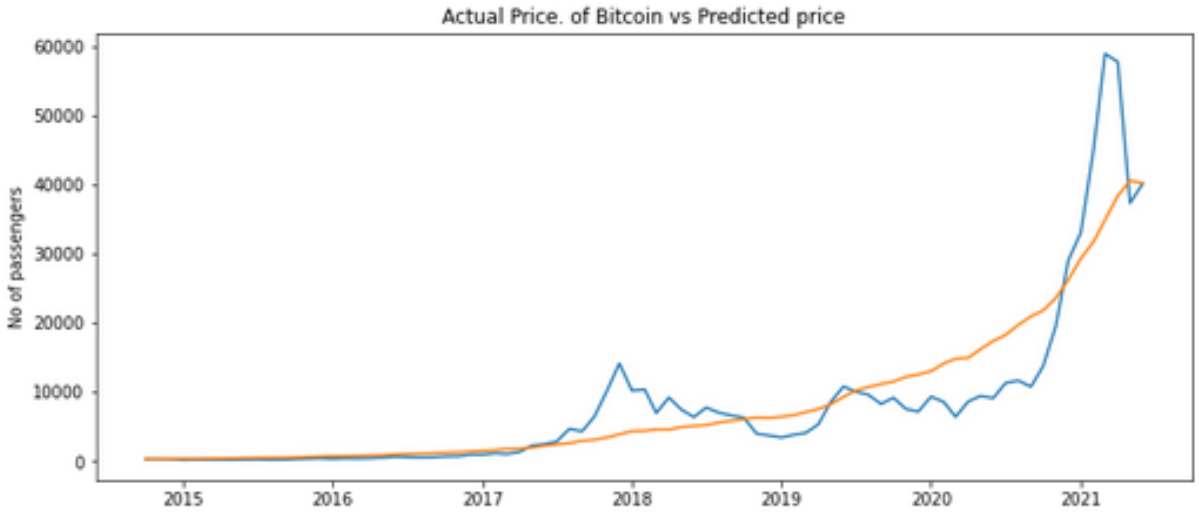
در این بخش درباره مدل MA (Moving Average) عملکرد آن در قالب Summary ارائه میشود. برای توضیحات الگوریتم Moving average باید اشاره کنم که از همان مدل ARIMA استفاده میشود ولی فقط پارامتر هایی پاس داده میشود که برای محسابه moving average نیاز است. بدین صورت از همان تابع ARIMA برای محاسبه اش استفاده میکنیم به طوری که پارامتر های فعال سازی بخش Auto regressive پاس داده نمیشود.

```
==============================================================================
Dep. Variable:                 D.Close   No. Observations:                   80
Model:                 ARIMA(0, 1, 2)   Log Likelihood                10.477
Method:                       css-mle   S.D. of innovations              0.212
Date:                Wed, 16 Jun 2021   AIC                            -12.954
Time:                        14:23:51   BIC                             -3.426
Sample:                    11-01-2014   HQIC                            -9.134
                         - 06-01-2021
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.0599      0.027      2.200      0.031       0.007       0.113
ma.L1.D.Close  0.1312      0.113      1.165      0.248      -0.090       0.352
ma.L2.D.Close  0.0172      0.108      0.160      0.873      -0.194       0.228
                                    Roots
=============================================================================
                  Real          Imaginary           Modulus         Frequency
-----------------------------------------------------------------------------
MA.1           -3.8140           -6.6022j            7.6247           -0.3334
MA.2           -3.8140           +6.6022j            7.6247            0.3334
-----------------------------------------------------------------------------
```

در این مدل AIC عدد منفی 13 شده است که هر چه قدر این عدد کمتر باشد مدل بهینه تر میباشد.

2.5) ARIMA method

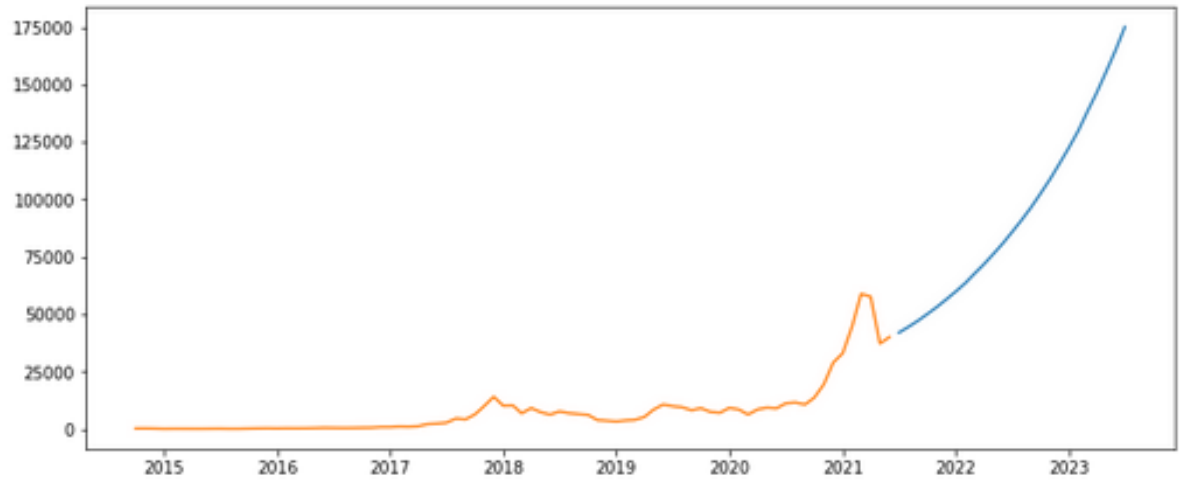
پیش بینی قیمت گذشته با استفاده از مدل ARIMA



پیش بینی قیمت 5سال آینده با استفاده از مدل ARIMA . همان طور که مشاهده میکنید مدل ما ترند را خوب تشخیص داده است ولی نتوانسته سیزونالیتی و سایکل ها را تشخیص دهد. (البته باید اشاره کنم که سیزن ها و سایکل ها منظم رخ نمیدهند و تشخیص آن ها فرایندی supervised ، در این مارکت ، هست .

XI. Plot showing Forcast for next ten years using ARIMA

```
[ ]  plt.plot(future_pred)
     plt.plot(z['actual'])
     plt.show()
```



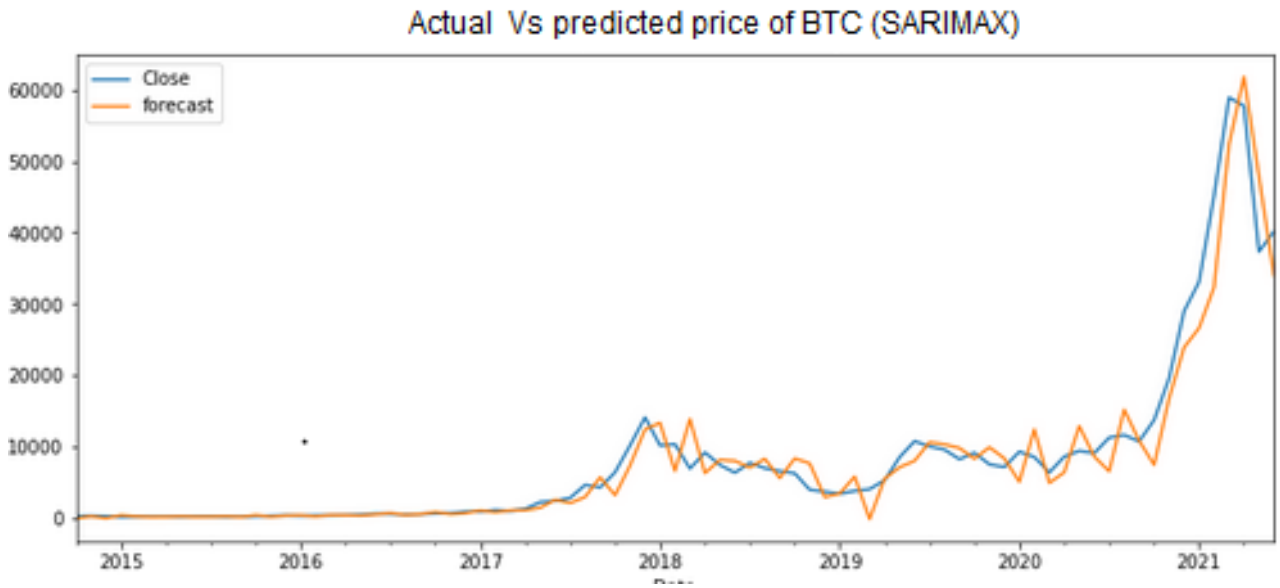
2.6) SARIMAX method

Dep. Variable:	Close	No. Observations:	81
Model:	SARIMAX(2, 1, 2)x(1, 1, 1, 12)	Log Likelihood	-650.118
Date:	Wed, 16 Jun 2021	AIC	1314.236
Time:	14:26:26	BIC	1329.773
Sample:	10-01-2014 - 06-01-2021	HQIC	1320.393
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	-0.1980	0.136	-1.457	0.145	-0.464	0.068
ar.L2	-1.0000	0.114	-8.789	0.000	-1.223	-0.777
ma.L1	0.6904	0.099	6.975	0.000	0.496	0.884
ma.L2	0.9200	0.103	8.935	0.000	0.718	1.122
ar.S.L12	-0.2424	0.681	-0.356	0.722	-1.578	1.093
ma.S.L12	-0.3878	0.666	-0.582	0.561	-1.694	0.918
sigma2	1.441e+07	1.12e-07	1.29e+14	0.000	1.44e+07	1.44e+07

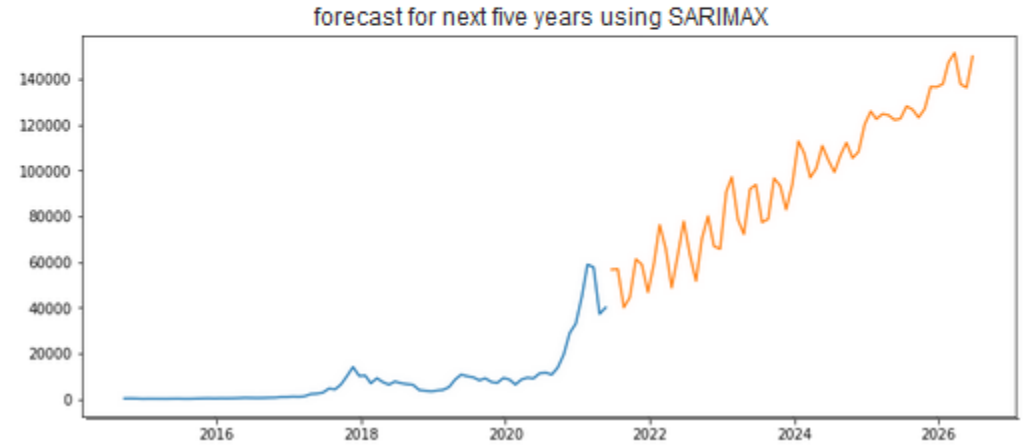
Ljung-Box (Q):	18.46	Jarque-Bera (JB):	30.27
Prob(Q):	1.00	Prob(JB):	0.00
Heteroskedasticity (H):	93.45	Skew:	0.32
Prob(H) (two-sided):	0.00	Kurtosis:	6.21

در این مدل AIC عدد 1314.2 شده است که هر چه قدر این عدد کمتر باشد مدل بهینه تر میباشد.



plot showing Forexast for next five years using SARIMAX

```
lt.plot(df1['Close'])
lt.plot(future_pred)
lt.title('Forecast for next five years using SARIMAX')
lt.ylabel('price of BTC')
lt.show()
```



part 3) **Using Neural Networks methods (LSTM) (روش شبکه های عصبی)**

در این روش شبکه های های عصبی حافظه دار ، که به طور گسترده در پیشبینی داده های سری زمانی نقش ایفا میکنند، استفاده میشود.

3.0) Explaining our step-by-Step NN approach

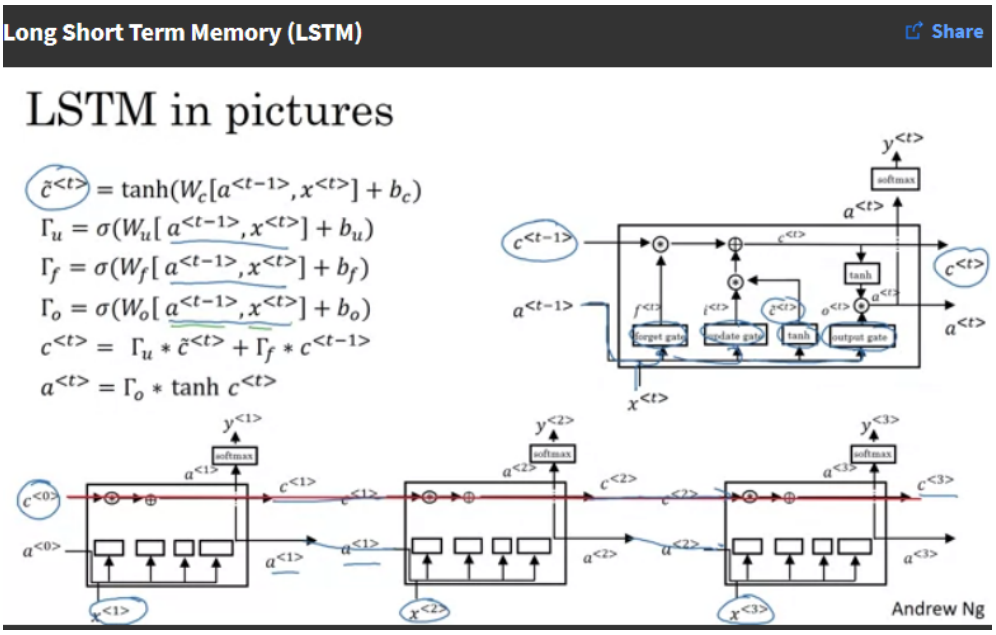
مراحل زیر را گام به گام در پروژه شبکه های عصبی خود دنبال کرده ایم:

- 1. We will collect the BTC stock data
- 2. Preprocessing the Data Train and Test
- 3. Create a Stacked LSTM model
- 4. Predict the test data and plot the output
- 5. Predict the future 30 days and plotting the output

در مرحله اول ما دیتا ها را ، مانند پروژه 3 که روش های آماری بررسی شد، از Yahoo API Finance میگیریم و در یک دیتاست پاندا ذخیره میکنیم.

3.1) About LSTM

برای فهم روشی Stacked LSTM از کورس Sequential models استفاده شده است ولی برای استفاده از آن از کتابخانه های tensorflow و Keras استفاده شده است.



عکس اقتباص شده از کورس Andrew NG , Sequence models

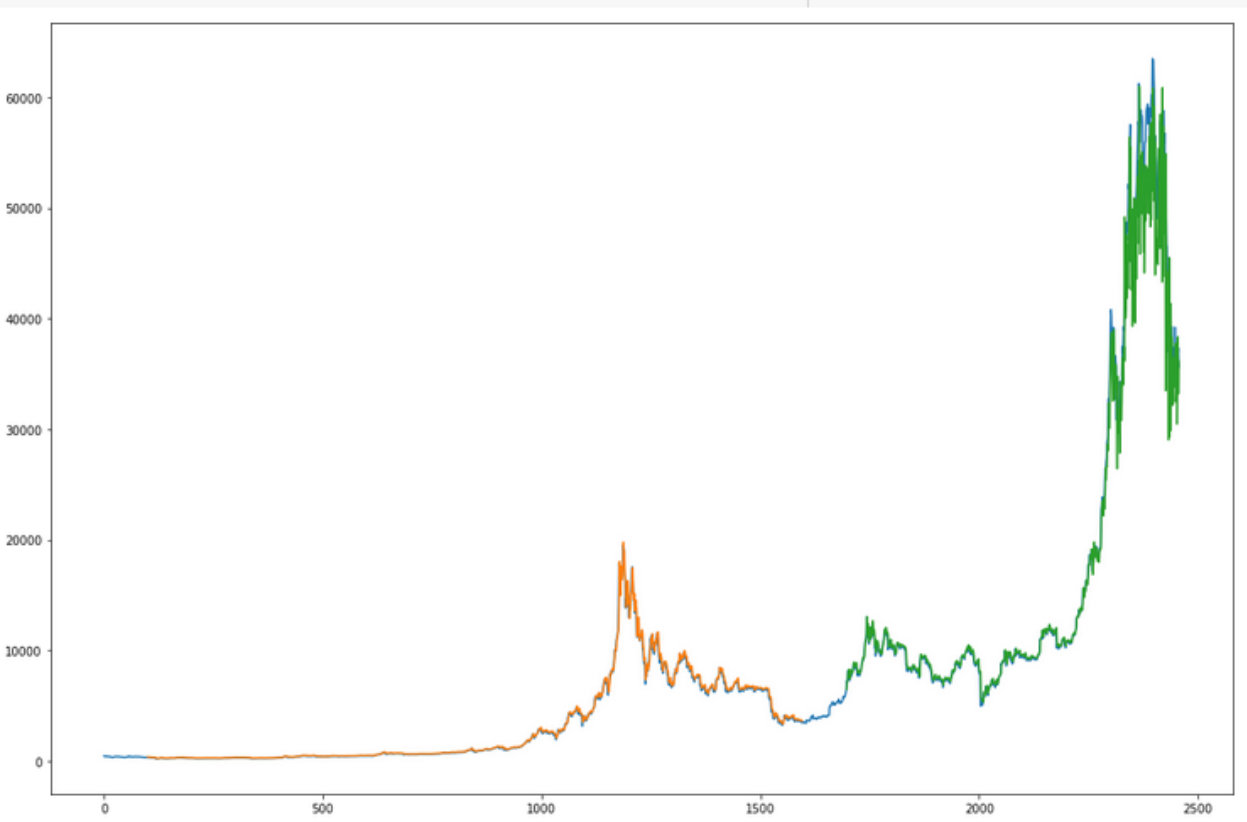
لازم به ذکر است در پیش بینی از Rolling window به سایز 100 روز استفاده شده است.

3.2) Results

بعد از 100 epoch مقدار loss-value به صورت زیر شده است:

```
24/24 [==============================] - 5s 192ms/step - loss: 2.6201e-05 - val_loss: 0.0024
Epoch 95/100
24/24 [==============================] - 5s 192ms/step - loss: 2.5361e-05 - val_loss: 0.0020
Epoch 96/100
24/24 [==============================] - 5s 191ms/step - loss: 3.2225e-05 - val_loss: 0.0021
Epoch 97/100
24/24 [==============================] - 5s 192ms/step - loss: 2.0951e-05 - val_loss: 0.0025
Epoch 98/100
24/24 [==============================] - 5s 194ms/step - loss: 2.1768e-05 - val_loss: 0.0023
Epoch 99/100
24/24 [==============================] - 5s 194ms/step - loss: 2.9429e-05 - val_loss: 0.0017
Epoch 100/100
24/24 [==============================] - 5s 192ms/step - loss: 3.5873e-05 - val_loss: 0.0017
<tensorflow.python.keras.callbacks.History at 0x7f9b73943210>
```

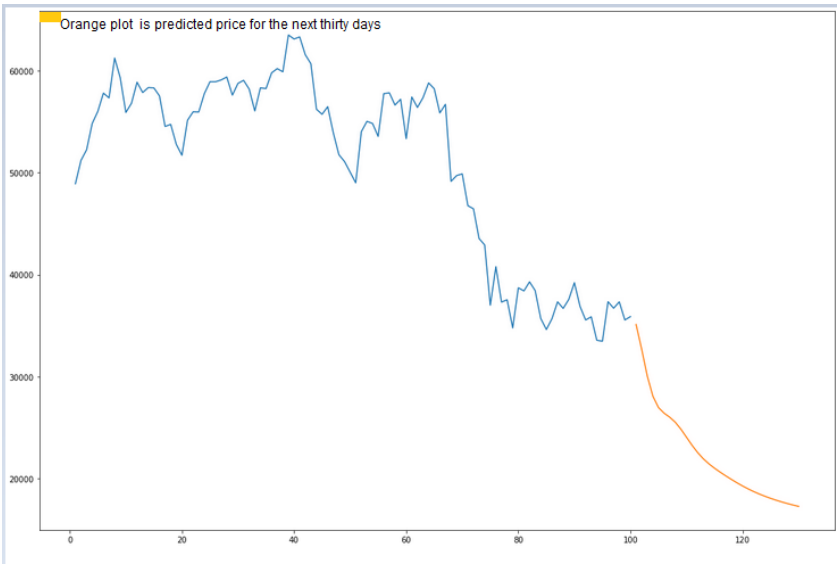
در شکل زیر نشان داده شده است که مدل ما چگونه داده های تست و آموزش را پیش بینی کرده است :



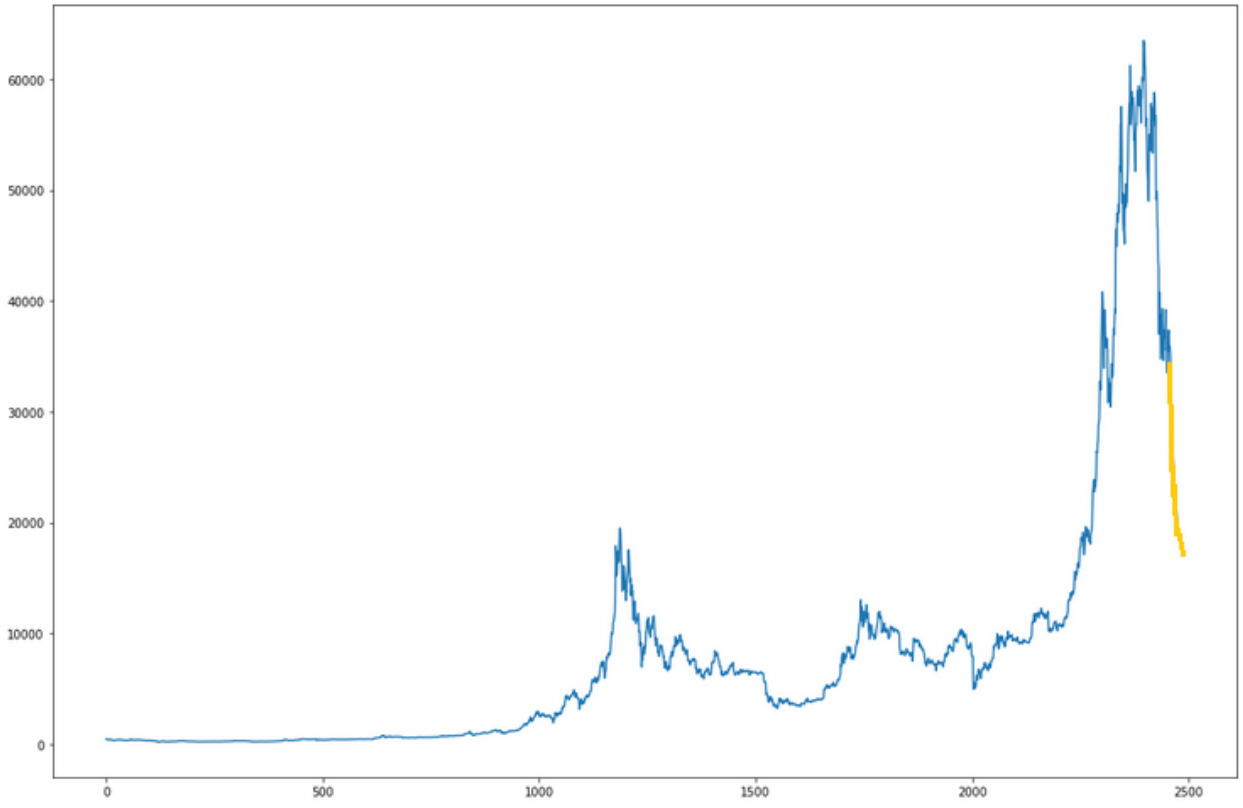
نمودار آبی رنگ نمودار قیمت اصلی BTC میباشد نمودار نارنجی رنگ مقدار پیش بینی شده روی داده های آموزش و نمودار سبز ، قیمت پیش بینی شده روی داده های تست میباشد که نشان میدهد که به خوبی جهت حرکت را capture کرده است.

پیش بینی قیمت برای 30 روز آینده را میتوانید در دو عکس پایین مشاهده کنیم.

پیش بینی قیمت برای 30 روز آینده (نمای نزدیک – چارت قیمتی برای 4 ماه اخیر) :



پیش بینی قیمت برای 30 روز آینده (نمودار شامل داده ها از ابتدای 2015 تا 2021-06):



part 4) **Conclusion and challenges (چالشها , نتیجه گیری)**

در این قسمت یک بار دیگر نتایجی که به دست آوردیم را در یک محل جمع آوری و جمع بندی کرده و نیز به چالش هایی که در طول پروژه به آن برخوردیم اشاره خواهیم کرد و نیز اهداف آینده پروژه ذکر میشوند.

4.1.1) Comparing the result with stats models

در یک قاب اگر بخواهیم همه پیش بینی ها را داشته باشیم و آن ها را مقایسه کنیم میتوان گفت مدل ARMA موفق به capture کردن ترند شده است ولی fluctuate های در قیمت را به خوبی منعکس نکرده است. و SARIMAX هم روند بلند مدت و سیزونالیتی را درست تشخصی داده است ولی AIC آن بالا است. روش LSTM هم پیش بینی های کوتاه مدت (20-30 روز آینده) به خوبی میتواند عمل کند و همینطور loss-value خیلی کمی دارد. مزیت دیگر آن این است که پارامتری نیست و میتواند مستقل از پارامتر های سیکل و ترند و سیزونالیتی مدل خود را با دقت بسیار بالا train کند.

4.1.2) Time and space complexity

یک مشاهده تجربی بر پروژه های پیاده سازی شده برای مدل های آماری و مدل شبکه های عصبی نشان داده که شبکه های عصبی زمان و فضای بسیار چشمگیری استفاده میکنند و علت آن هم این است که برای ساخت مدل تعداد بسیار زیادی weight and hyper parameters حدس زده میشود. در مدلی که ما اجرا کردیم تعداد epoch ها نیز 100 قرار دادیم و هم load زیادی برای الگوریتم ایجاد کرد .(آموزش مدل شبکه های عصبی به طور تجربی حدود 50 دقیقه طول کشید درصورتی که مدل های آماری کمتر از 6 دقیقه به طول انجامیدند.)

4.2) Challenges

- به علت پیچیده بودن داده های کریپتوکارنسی ها نسبت به داده های "مسافران هواپیمایی" زمان زیادی صرف پیدا کردن بهترین فرمت استفاده در پروژه (روزانه بودن یا ماهانه بودن time frame) صرف شد.
- زمان زیادی صرف پیدا کردن بهترین حالت تقسیم دیتا ست به دو بخش آموزش و تست در شبکه های عصبی گرفته شد.
- کتابخانه matplotlib مشکلی در نمایش دادن تعداد زیاد رکورد داده ها داشت که از یک لایبرری مناسب برای نمایش قیمت استاک ها استفاده شد که در Appendix (2) قابل مشاهده میباشد. البته در ورژن های نهایی پروژه مشکل کتاب خانه matplotlib هم حل شد.
- به علت جدید بودن هم روش های اماری و هم روش شبکه های عصبی زمان زیادی صرف یادگیری و فهم روند کلی آن ها و فهم چگونگی استفاده از کتابخانه های آماده برای پیاده سازی آن ها صرف شد.

4.3) Future works

هدف بعدی این رشته از پروژه ها و تحقیقات که در آینده دنباله خواهد گرفته شد این است که بتوان از یک تکنیک که معامله گر های حرفه ای از آن استفاده میکنند در پیاده سازی شبکه عصبی خود استفاده کرد . اسم این تکنیک Multiple time-frame analysis نام دارد.

مورد دیگر بررسی مدل Garch and ARCH هست و نتایج آن را با مدل های استفاده شده در پروژه 3 و 4 خود مقایسه کنیم.

Part 5) **References (منابع و ماخذ)**

[traditional trading پیاده سازی بخش]

https://radarcx.herokuapp.com/

[Auto Regressive explanation]

https://machinelearningmastery.com/autoregression-models-time-series-forecasting-python/

[Learning LSTM neural network method]

https://www.coursera.org/lecture/nlp-sequence-models/long-short-term-memory-lstm-KXoay

[I have used following material for implementing the 3'rd project]

https://www.youtube.com/watch?v=SoMzQUdeFjI

http://rstudio-pubs-static.s3.amazonaws.com/311446_08b00d63cc794e158b1f4763eb70d43a.html

https://www.youtube.com/watch?v=e8Yw4alG16Q

https://www.youtube.com/watch?v=F5cz6RGrqf8

[I have used following material for implementing the 4'th third project]

https://www.youtube.com/watch?v=Vfx1L2jh2Ng

https://www.youtube.com/watch?v=H6du_pfuznE

Part 6) **Appendix (متصلات)**

Appendix (#1): Yahoo Finance API code

```
!pip install yfinance
!pip  install ta
```

```
 import yfinance as yf
```

```
dt= yf.download(tickers='BTC-USD',period='100mo',interval='1mo')
```

```
dt
dt.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 82 entries, 2014-10-01 to 2021-06-12
Data columns (total 6 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   Open       82 non-null     float64
 1   High       82 non-null     float64
 2   Low        82 non-null     float64
 3   Close      82 non-null     float64
 4   Adj Close  82 non-null     float64
 5   Volume     82 non-null     int64  
dtypes: float64(5), int64(1)
memory usage: 4.5 KB
```

Appendix (#2): Better library for plotting charts

```
# Data vis
import plotly.graph_objs as go
# declare plot
fig = go.Figure()
```

```
fig.add_trace(go.Candlestick(x=dt.index,
                             open=dt['Close'],
                             high=dt['Close'],
                             low=dt['Close'],
                             close=dt['Close'],
                             name='market data'))

# Add title
fig.update_layout(title='bitcoin live share price',
                  yaxis_title='Bitcoin price (USD)')
```

```
# X axis
fig.update_xaxes(rangeslider_visible=True,
                 rangeselector=dict(
                 buttons=list([
                             dict(count=3, label="3D",step="day",stepmode
="backward"),
                             dict(count=30, label="30D",step="day",stepmo
de="backward"),
                             dict(step="all")
                 ])
                 ))
```