دانشگاه پیام نور

پیاده سازی سری زمانی تک متغیره

با زبان پایتون

پروژه فناوری اطلاعات دوره کارشناسی

مهندسی کامپیوتر گرایش نرم‌افزار

ارائه‌شده به:

گروه علمی فناوری اطلاعات و ارتباطات

دانشکده فنی مهندسی

دانشگاه پیام نورمرکز تهران شمال

استاد راهنما:

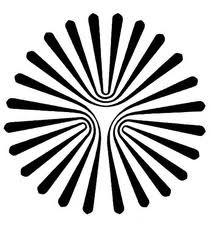
دکتر سیدعلی رضوی ابراهیمی

توسط:

فرزاد صحرائی

پاییز 1399



دانشگاه پیام نور

پیاده سازی سری زمانی تک متغیره

با زبان پایتون

پروژه فناوری اطلاعات دوره کارشناسی

مهندسی کامپیوتر گرایش نرم‌افزار

ارائه‌شده به:

گروه علمی فناوری اطلاعات و ارتباطات

دانشکده فنی مهندسی

دانشگاه پیام نورمرکز تهران شمال

استاد راهنما:

دکتر سیدعلی رضوی ابراهیمی

توسط:

فرزاد صحرائی

پاییز 1399

چکیده

امروزه با افزایش حجم اطلاعات و نیاز مبرم به تحلیل و پیشبینی روند اطلاعات در آینده جهت امور اقتصادی، اجتماعی و غیره، روش‌های تحلیل داده پا به عرصه وجود نهاده‌اند. یکی از این روش‌ها، تحلیل سری زمانی است. تمامی این مبحاث بسیار گسترده هستند و دانش بسیاری در زمینه اقتصاد سنجی، علوم کامپیوتر و سخت افزار در این عرصه وجود دارد. در این مستند به پیاده سازی سری زمانی تک متغیره می‌پردازیم.

در این پروژه ابتدا به تعاریف اولیه سری زمانی و در پایان به بحث پیاده سازی آن خواهیم پرداخت.

مطالب این پروژه در دو فصل تنظیم‌شده است.

فصل اول به مقدمه‌ای بر سری زمانی و تعاریف آن پرداخته‌شده است که باعث فهم بهتر فصل بعد می‌شود.

در فصل دوم به پیاده سازی سری زمانی در محیط پایتون و رابط گرافیکی آن با زبان C# خواهیم پرداخت.

فهرست

[**1** فصل اول 1](#_Toc75012251)

[**1-1** **مقدمه** 2](#_Toc75012252)

[**1-2** **سری زمانی** 2](#_Toc75012253)

[**1-3** **مولفه‌های سری زمانی** 3](#_Toc75012254)

[**1-4** **وضعیت ایستایی روند اطلاعات در سری زمانی** 6](#_Toc75012255)

[**1-5** **وضعیت ایستایی روند اطلاعات در سری زمانی** 7](#_Toc75012256)

[**1-5-1** **عملگر میانگین متحرک** 7](#_Toc75012257)

[**1-6** **تحلیل سری زمانی** 9](#_Toc75012258)

[**1-6-1** **روش پیش‌بینی میانگین متحرک** 9](#_Toc75012259)

[**1-6-2** **روش پیش‌بینی اتورگرسیو** 10](#_Toc75012260)

[**1-6-3** **روش پیش‌بینی میانگین متحرک-اتورگرسیو** 11](#_Toc75012261)

[**1** فصل دوم 12](#_Toc75012262)

[**2-1** **مقدمه** 13](#_Toc75012263)

[**2-2** **نصب نیازمندیهای تولید سامانه** 14](#_Toc75012264)

[**2-3** **مشخصات نرم افزاری پروژه** 15](#_Toc75012265)

[**2-4** **پیاده‌سازی سری زمانی با زبان پایتون** 15](#_Toc75012266)

[**2-5** **پیاده‌سازی رابط کاربری** 22](#_Toc75012267)

[**تقدیر** 26](#_Toc75012268)

# فصل اول

نگاهی به سری‌های زمانی

## **مقدمه**

سری‌زمانی مجموعه‌ای از نقاط داده به ترتیب زمانی فهرست شده است. معمولاً، یک سری زمانی یک توالی است که در نقاط متوالی با فاصله مساوی از زمان گرفته می‌شود. بنابراین توالی داده‌های زمان گسسته است. تجزیه و تحلیل سری‌های زمانی می‌تواند مفید باشد تا ببینید که دارایی‌ امنیتی یا متغیر اقتصادی معین در طول زمان چگونه تغییر می‌کند. به عنوان مثال از سری‌های زمانی برای محاسبه ارتفاعات جزر و مد اقیانوس‌ها، تعداد لکه‌های خورشید و مقدار بسته شدن روزانه میانگین‌ها متغیرهای اقتصادی مانند روند عملکرد یک سهم در بورس و یا نرخ رشد تورم و نقدینگی استفاده می‌شود.

سری های زمانی که به صورت نمودار خطی رسم می شوند، به طور کلی در آمار‌، پردازش سیگنال، تشخیص الگو، اقتصاد سنجی ، امور مالی ریاضی، پیش بینی هوا، پیش بینی زلزله، مهندسی کنترل، نجوم، مهندسی ارتباطات و عمدتا در هر حوزه ای از علمی و مهندسی که شامل اندازه گیری های زمانی است، استفاده می‌شوند.

در این فصل به تعریف سری سری زمانی به صورت کلی می‌پردازیم.

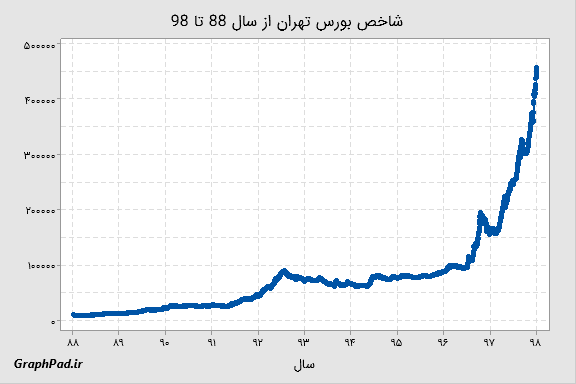
## **سری زمانی**

دنباله‌ای از داده‌ها که در یک محدود زمانی جمع‌آوری شده‌‌اند، یک سری زمانی را تشکیل می‌دهند. این داده‌ها تغییراتی که پدیده در طول زمان دچار شده را منعکس می‌کنند. بنابراین می‌توانیم این مقدارها را یک بردار وابسته به زمان بدانیم. در این حالت اگر X یک بردار باشد، سری زمانی را می‌توان به صورت زیر نشان داد؛ که در آن t، بیانگر زمان و X نیز یک متغیر تصادفی است.

X(t), t = 0,1,2,3,...

طبق این تعریف زمان t=0 نیز قابل تعریف است. این لحظه می‌تواند زمان تولد یک پدیده یا هنگامی باشد که اولین اطلاعات در آن لحظه ثبت شده است. به این ترتیب X(t) متغیر تصادفی X را در زمان t‌ نشان می‌دهد. مقدارهای مشاهده شده این متغیر تصادفی دارای ترتیبی هستند که زمان وقوع هر داده را نشان می‌دهند.

اگر متغیر تصادفی X، یک بعدی باشد،‌ یعنی از بین ویژگی‌های مختلف یک پدیده فقط از یکی ویژگی برای ایجاد مدل سری زمانی استفاده شود، مدل را «یک متغیره» (Univariate) می‌نامند. ولی اگر از چندین ویژگی برای ایجاد مدل سری زمانی استفاده شود، مدل سری زمانی را «چند متغیره» (Multivariate) می‌گویند.



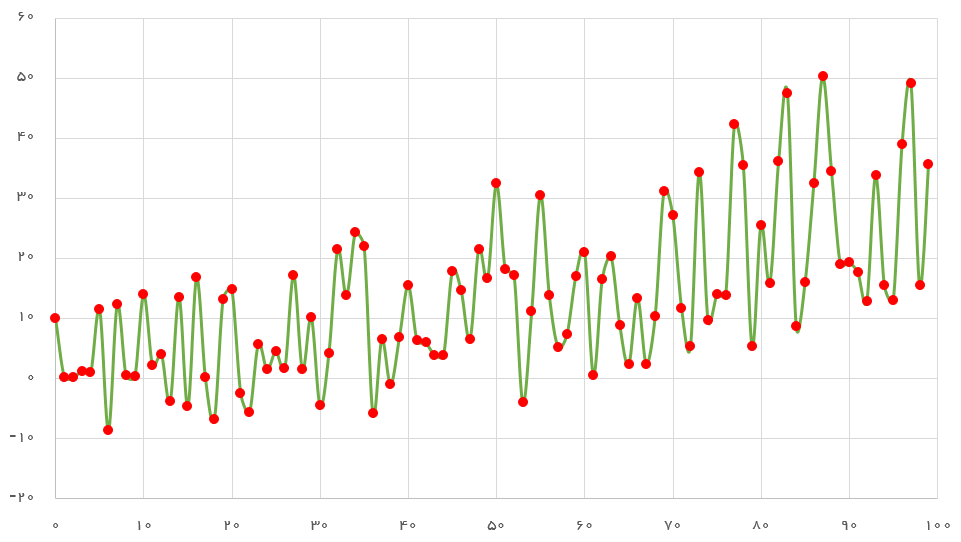
نمونه‌ای از سری زمانی تک متغیره

## **مولفه‌های سری زمانی**

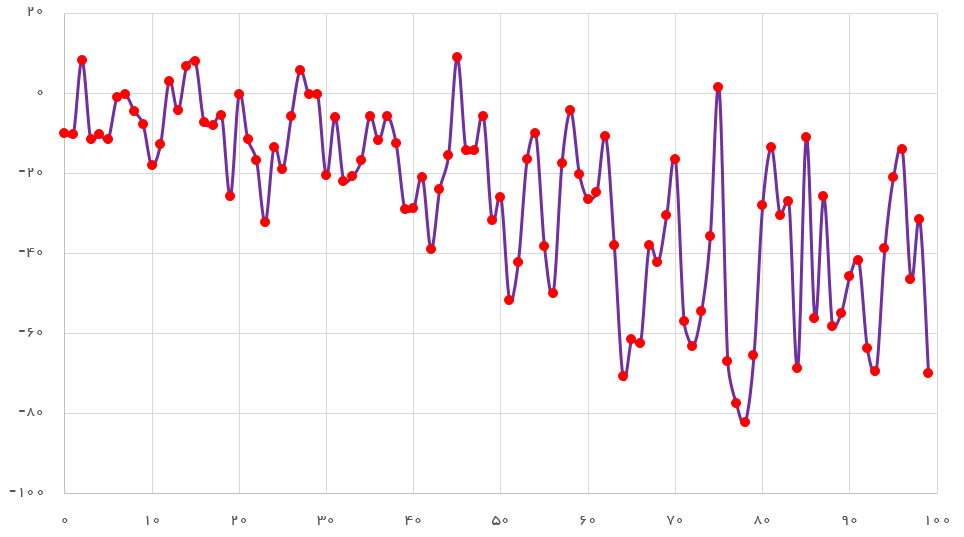
معمولا می‌توان الگوی رفتار یا مدل تغییرات یک سری زمانی را به چهار مولفه تفکیک کرد. این مولفه‌ها شامل «روند» (Trend)، «تناوب» (Cyclic)، «فصل» (Seasonal) و «تغییرات نامعمول» (Irregular) هستند. اگر نمودار مربوط به داده‌های سری زمانی را برحسب زمان ترسیم کنیم می‌توانیم این مولفه‌ها را تشخیص دهیم در نتیجه شناخت بهتری از داده‌های سری زمانی خواهیم داشت. در ادامه به معرفی و بررسی هر یک از این مولفه‌ها می‌پردازیم.

روند (Trend): تمایل سری زمانی به افزایش، کاهش یا حتی ثابت بودن، روند را تشکیل می‌دهد. در یک سری زمانی با روند افزایشی، انتظار داریم مقدارهای سری زمانی در زمان‌های t=1 و t=2 به صورت X(1) <= X(2) باشند. برای مثال روند برای سری زمانی مربوط به میزان جمعیت یا سرمایه در بازار بورس به صورت افزایشی، ولی روند برای میزان مرگ و میر با توجه به پیشرفت در امور پزشکی، کاهشی است.

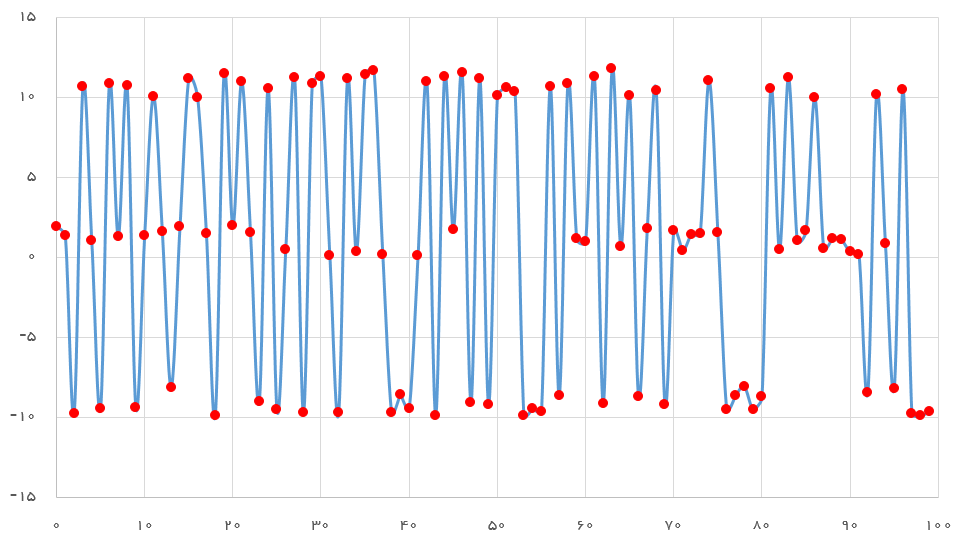
در تصویرهای زیر سه نمودار مربوط به سه سری زمانی در ۱۰۰ زمان مختلف با روندهای افزایشی، ثابت و کاهشی نشان داده شده است.



روند افزایشی

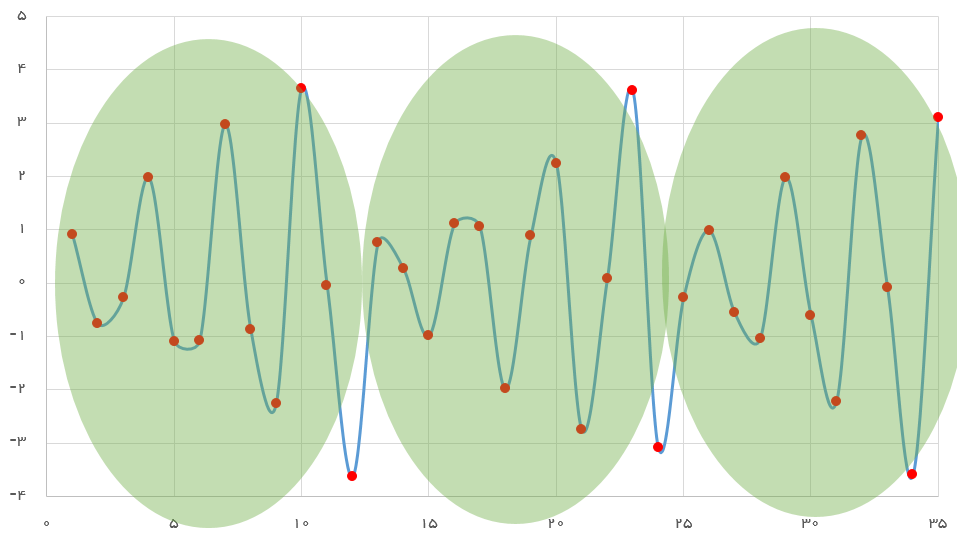


روند کاهشی



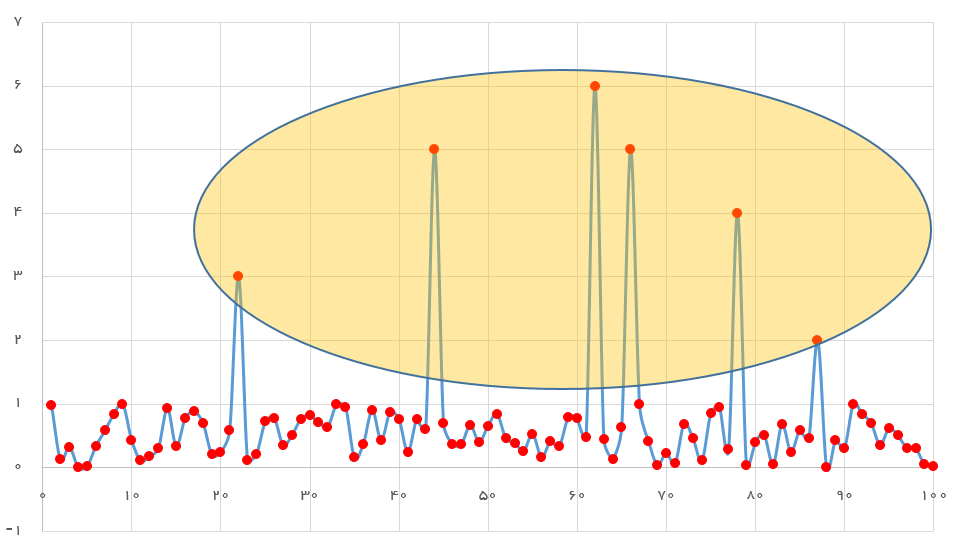
روند ثابت

تناوب (Cyclic): تغییرات یکسان و تکراری در مقاطع میان‌مدت، تناوب در سری زمانی نامیده می‌شود. معمولا این تناوب ممکن است هر دو سال یا بیشتر اتفاق بیافتد. برای مثال تناوب در کسب و کار دارای یک چرخه چهار مرحله‌ای است که باعث می‌شود داده‌های مربوط به کسب و کار در یک دوره تناوب 3 ساله تکرار شوند.



روند تناوبی

تغییرات نامعمول (Irregular): این گونه تغییرات بر اثر عوامل تصادفی و غیرقابل پیش‌بینی ایجاد می‌شوند. برای مثال زلزله یا سیل در بررسی رشد جمعیت ممکن است اثرات بزرگی داشته باشد. این مولفه بعد از شناسایی توسط نمودار ترسیم شده از سری زمانی باید حذف شود. در غیر اینصورت نتایج حاصل از تحلیل سری زمانی ممکن است گمراه کننده باشند.



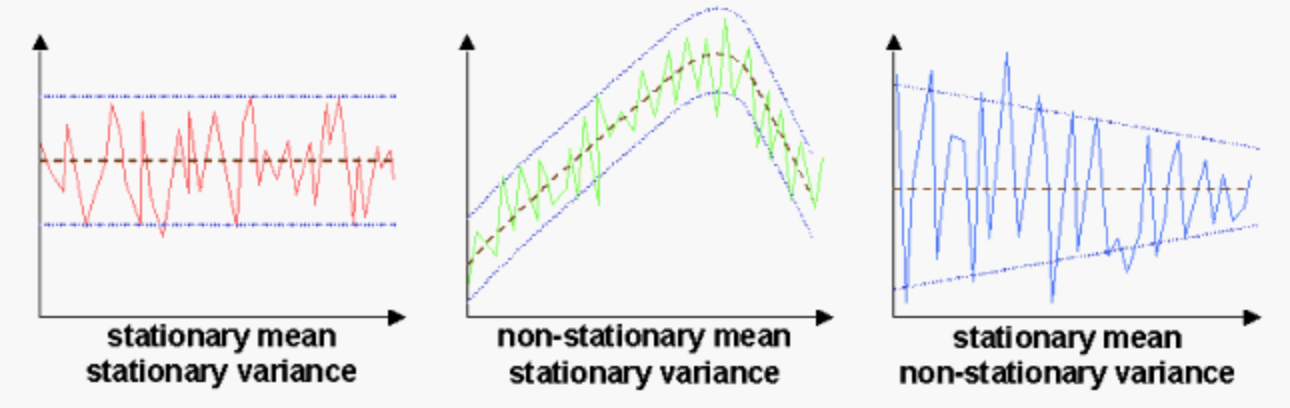
تغییرات نامعمول

به صورت کلی در روند اطلاعات در گذر زمان اطلاعات رفتارهایی تکرار شونده دارند و با تحلیل رفتار دیتا می‌توان به رفتار آن را درآینده پیشبینی کرد. سری زمانی مدل های تحلیل متفاوت و گسترده‌ای دارد. که در ادامه به آنها خواهیم پرداخت.

## **وضعیت ایستایی روند اطلاعات در سری زمانی**

در روند اطلاعات در سری زمانی جهت تحلیل تناوب دوره ای و فصلی اطلاعات می بایست ایستا باشد. به این معنا که روند اطلاعات میبایست به زمان وابسته باشد اما واریانس و میانگین آن وابسته به زمان نباشد.

همانطور که در تصاویر زیر مشاهده می شود، در حالتی که هم واریانس و هم میانگین ایستا هستند. روند اطلاعات هرچند که به زمان وابسته است ایستا خواهد بود. از روند ایستا امکان استخراج تناوب اطلاعات وجود دارد. در سایر روندهای نا ایستا ابتدا باید روند را ایستا کرد بعد به تحلیل دیتا پرداخت.



**میانگین ایستا وارینانس نا ایستا**

**میانگین نا ایستا وارینانس ایستا**

**میانگین ایستا وارینانس ایستا**

## **وضعیت ایستایی روند اطلاعات در سری زمانی**

از آنجایی که امکان پیش‌بینی برای سری‌های زمانی ناایستا (Non-stationary) به راحتی امکان پذیر نیست، بهتر است عواملی که باعث خارج شدن سری زمانی از حالت ایستایی هستند، حذف شوند. به این ترتیب باید مولفه‌های شناسایی شده در سری زمانی را حذف کنیم. به این کار «هموار سازی» (Smoothing) یا «صافی» (Filtering) می‌گویند. روش‌های مختلفی برای هموارسازی سری زمانی وجود دارد. عملگرهای میانگین متحرک، هموارسازی نمایی ساده، روش‌های تفاضل‌گیری و لگاریتم و… به حذف مولفه‌های سری زمانی کمک می‌کنند. در این نوشتار به بررسی عملگر میانگین متحرک پرداخته و نقش آن را در حذف مولفه روند بررسی و سپس سری زمانی ایستای تولید شده را، تحلیل می‌کنیم.

## **عملگر میانگین متحرک**

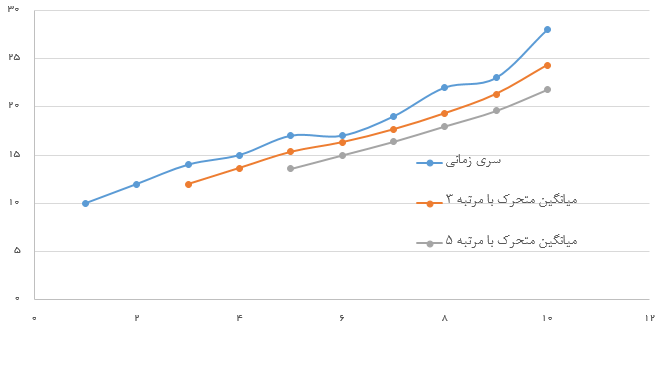
عملگر میانگین متحرک با توجه به درجه انتخابی برای میانگین‌گیری مثلا k، هر مقدار از سری زمانی را با میانگین k-1‌ مقدار قبلی و خودش جایگزین می‌کند. این کار به حذف روند و تشکیل الگوی ساده برای سری زمانی کمک بسیاری می‌کند. میانگین متحرک یک روش هموارسازی داده‌ها است. البته این عملگر، مولفه روند را برای سری زمانی برآورد می‌کند تا در زمان پیش‌بینی از آن استفاده شود.

واضح است که برای دو مقدار اول در سری زمانی، این کار امکان پذیر نیست زیرا نمی‌توان سه مقدار برای محاسبه میانگین در نظر گرفت. در نتیجه این گونه هموارسازی به کاهش مجموعه داده منجر می‌شود. هر چه درجه همواره سازی را بزرگتر انتخاب کنید، هموارسازی زودتر انجام خواهد شد ولی در عوض ممکن است خطای پیش‌بینی را افزایش دهد.

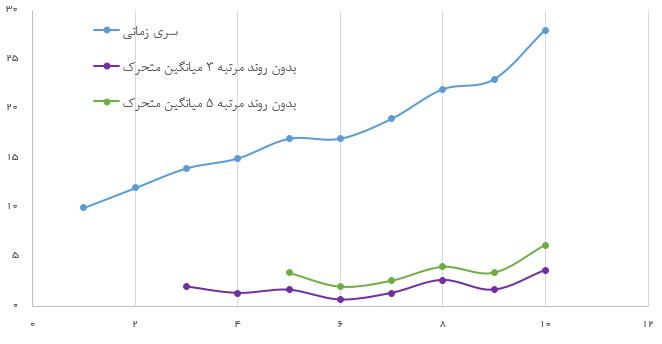
حال برای آنکه سری زمانی، خاصیت ایستایی پیدا کند و روند را از آن حذف کنیم، کافی است مقدارهای سری زمانی را از میانگین متحرک کم کنیم، سری زمانی جدید بدون روند خواهد بود. پس اگر Y(t) سری زمانی باشد و m(t) میانگین متحرک در زمان t در نظر گرفته شود، سری زمانی ایستای X(t)به صورت زیر مورد محاسبه قرار می‌گیرد.

X ( t ) = Y ( t ) − m ( t )

به تصویر‌های زیر دقت کنید. در سمت راست نمودار سری زمانی به همراه مقدارهای حاصل از عملگرهای میانگین متحرک مرتبه ۳ و ۵ دیده می‌شود. همچنین در سمت راست سری زمانی ناایستا به رنگ آبی و سری‌های زمانی ایستای حاصل از عملگر میانگین متحرک مرتبه ۳ و ۵ به رنگ‌های بنفش و سبز نمایش داده شده‌اند. کاملا مشخص است که در سری‌های زمانی ایستای تولید شده،‌ مولفه روند وجود ندارد.



سری زمانی و نتیجه عملکرد میانگین متحرک 3 و 5



سری زمانی اصلی و سری زمانی ایستا براساس عملگر میانگین متحرک مرتبه ۳ و ۵

## **تحلیل سری زمانی**

بعد از شناسایی و حذف مولفه‌های اصلی سری زمانی، وقت آن رسیده که بتوانیم عمل پیش‌بینی را انجام دهیم، یعنی مدل ریاضی برای ارتباط بین مقدارهای سری زمانی را پیدا کنیم. در اینجا به دو مدل پیش‌بینی سری زمانی به نام «میانگین متحرک» (Moving Average) و «اوتورگرسیو» (Autoregressive) می‌پردازیم.

## **روش پیش‌بینی میانگین متحرک**

برای پیش‌بینی سری زمانی ایستا که روند از آن خارج شده است، می‌توان از مدل میانگین متحرک استفاده کرد. در این مدل سری زمانی X(t) به صورت زیر نوشته می‌شود:

X ( t ) = θ0 + Z ( t ) + θ1 Z ( t − 1 ) + … + θq Z ( t − q )

که در آن Z(t) همان خطای تصادفی با میانگین صفر و واریانس ثابت است. به این مدل، میانگین متحرک مرتبهq‌ گفته و به صورت MA(q) نمایش داده می‌شود. در این مدل مقدار سری زمانی در زمان t یک ترکیب خطی از خطاها تا زمان t است.

با توجه به تعریف مدل و صفر بودن میانگین Z(t)، مشخص است که میانگین سری زمانی ایستا برابر است با θ0 برای محاسبات راحت‌تر در مدل، می‌توان این میانگین را از سری زمانی ایستا کم کرد تا مدل به صورت زیر نوشته شود، که در آن θ0 و θ1 پارامترهای مدل هستند.

X ( t ) = Z ( t ) + θ1 Z ( t − 1 ) + … + θq Z ( t − q )

استفاده از این مدل باید با در نظر گرفتن یک شرط صورت پذیرد. شرطی که در این مدل باید رعایت شود، آن است که می‌بایست قدرمطلق ضرایب مدل کوچکتر از یک باشند.

مدل میانگین متحرک درجه ۱ به صورت زیر نوشته می‌شود:

X ( t ) = Z ( t ) + θ1 Z ( t − 1 )

اگر میزان خطا را برای زمان t به صورت X(t)−Z(t) نشان دهیم، خواهیم داشت.

X ( t ) − Z ( t ) = θ 1 Z ( t − 1 )

مجموع مربعات خطا در این مدل به صورت زیر است:

MSE = ∑ ( X ( t ) − Z ( t ) ) 2 = ∑ ( θ1 Z ( t − 1 ) ) 2

باید توجه کرد که در زمان صفر نمی‌توان خطا را اندازه گرفت در نتیجه محاسبه خطا و مقدار سری زمانی از زمان یک محاسبه شده که میزان خطا بعد از برآورد پارامتر محاسبه شده است.

## **روش پیش‌بینی اتورگرسیو**

اگر مقدارهای سری زمانی ایستا به صورتی باشند که به مقدارهای قبلی خود بستگی داشته باشند، از مدل اتورگرسیو استفاده می‌شود. در این حالت p را تعداد مشاهدات گذشته در نظر می‌گیریم که برای پیش‌بینی یک مقدار در نظر گرفته می‌شود. بنابراین می‌توانیم مدل اتورگرسیو را برای سری زمانی ایستا X(t) به صورت زیر بنویسیم:

X ( t ) = a 0 + a 1 X ( t − 1 ) + a 2 X ( t − 2 ) + … + a p X ( t − p ) + Z ( t )

در این رابطه a0,a1,a2,…,ap پارامترهای مدل اتورگرسیو و Z(t) خطای تصادفی در نظر گرفته می‌شود. چنین مدلی را به صورت AR(p) نشان می‌دهند و p را مرتبه مدل می‌نامند. انتخاب مقدار p نیاز به بررسی سری زمانی و میزان همبستگی مقدارهای سری زمانی به یکدیگر دارد.

از آنجایی که این مدل به مانند یک مدل رگرسیون نوشته شده، به آن مدل اتورگرسیو می‌گویند، با توجه به اینکه برای X(t) رگرسیون روی مقدارهای گذشته ایجاد شده، شرط استقلال متغیرهای توصیفی حداقل برای تعداد دسته‌های کوچکتر از p وجود نخواهد داشت. همچنین مقدار حال حاضر سری زمانی فقط به p مقدار قبلی وابسته بوده و به قبل از آن ارتباطی ندارد.

معمولا برای بررسی سری زمانی ایستا از مدل اتورگرسیو مرتبه اول یا دوم استفاده می‌شود. مدل اتورگرسیو مرتبه اول به صورت زیر است:

X ( t ) = a 0 + a 1 X ( t − 1 ) + Z ( t )

حال برای بدست آوردن ضریب a1 و a0 کافی است که رگرسیون حاصل از مقدارهای X(t) را روی X(t−1)‌ بدست آوریم. توجه داریم که در این حالت، روی سری زمانی ایستا محاسبات باید انجام بپذیرد.

## **روش پیش‌بینی میانگین متحرک-اتورگرسیو**

با توجه به خصوصیات هر یک از روش‌های میانگین متحرک و اتورگرسیو، امکان ترکیب این دو روش نیز وجود دارد. چنین مدلی به نام میانگین متحرک-اتورگرسیو معروف است و در چنین حالتی مدل را به صورتARMA(p,q) نشان می‌دهند. رابطه بین مقدارهای سری زمانی ایستا در این مدل به صورت زیر قابل مشاهده است:

X(t)=a0+a1X(t−1)+…,apX(t−p)+Z(t)+θ1Z(t−1)+…+θqZ(t−q)

با توجه به این موضوع مدل ARMA(1,1) به صورت زیر در خواهد آمد:

X(t)=a0+a1X(t−1)+Z(t)+θ1Z(t−1)

در این بخش با کلیات سری زمانی آشنا شدیم در بخش بعد به تولید آن با استفاده از امکانات زبان پایتون می‌پردازیم.

# فصل دوم

**پیاده سازی سری زمانی تک متغیره با زبان پایتون**

## **مقدمه**

در این پروژه، قصد داریم با زبان برنامه نویسی پایتون سری زمانی تک متغیره را پیاده سازی نماییم. سری زمانی از جمله روش‌های تحلیل داده است که به محققان امکان بررسی روند و رفتار یک شاخص در گذر زمان را می‌دهد. در ضمن با استفاده از آن می‌توان پیشبینی کرد که احتمالا رفتار آن شاخص در آینده به چه صورت است. مثلا می‌توان رفتار شاخص نقدینگی، طلا، ارز و غیره را پیشبنی نمود. اما در دنیا واقعی همانطور که مستحضر هستند، شاخص‌های مختلف بر یکدیگر اثرگذار هستند. لذا پیشبینی‌های سری زمانی تک متغیره دقیق نیستند. اما تحلیل خوبی از رفتار شاخص را در اختیار محققان قرار می‌دهند، که می‌تواند به آنها در سایر مدل پیشبینی مانند VAR، رگرسیون و شبکه عصبی کمک شایانی بنماید.

در این پروژه ابتدا روال نصب نیازمندیهای تولید سری زمانی را توضیح می‌دهیم. سپس به توضیح امکانات سامانه تولید شده خواهیم پرداخت.

## **نصب نیازمندیهای تولید سامانه**

زبان پایتون و ماژول‌های تولید شده برپایه آن به دلیل متن باز بودن دایما در حال تغییر هستند و این تغییرات موجب می‌گردد که در نسخه‌های متفاوت از این ماژول‌ها یکدیگر را پشتیبانی نکنند. لذا توصیه می‌گردد که یک محیط مناسب برای تولید نرم افزار خود محیا سازیم. برای تولید این محیط در این سامانه از نرم‌افزار Anaconda نسخه 3 استفاده می‌نماییم. این نرم‌افزار یک سیستم مدیریت محیط است که در آن می‌توان تمام ماژول‌های مورد نیاز فرآیند تولید را نصب و اطمینان حاصل کرد که با سایر امکانات موجود در سیستم ما تداخل نخواهد داشت. پس گام اول نصب نرم افزار Anaconda است.

پس از نصب نرم افزار Anaconda با دستور زیر یک محیط ایجاد می‌نماییم. این دستور نسخه پایتون مورد نیاز ما را نیز نصب خواهد کرد.

conda create -n myenv python=3.6

در این پروژه از ماژول StatsModels برای سری‌های زمانی استفاده می نماییم. برای نصب آن از دستور زیر استفاده می‌نماییم.

conda install -c conda-forge statsmodels

سایر نیازمندیهای تولید را که شامل امکانات ایجاد API و نیازمندیهای کار با دیتای پایتون است، را نصب می‌کنیم.

conda install -c anaconda pylint

conda install -c conda-forge matplotlib

conda install -c conda-forge tensorflow

conda install -c conda-forge scikit-learn

conda install -c anaconda flask

در گام بعدی برای اجرای پایتون در Microsoft Visual Studio از Nuget ماژول پایتون زیر را نصب می‌ نماییم.

Install-Package pythonnet -Version 2.3.0-py35-dotnet

پس از نصب موارد بالا، امکان شروع تولید را خواهیم داشت. در بخش بعدی مشخصات نرم افزاری سامانه تحلیل سری زمانی به صورت خلاصه ذکر خواهد شد.

## **مشخصات نرم افزاری پروژه**

این نرم افزار شامل دو بخش است که بخش تحلیل سری زمانی در محیط پایتون با استفاده از ماژول Statsmodels تولید شده است و قسمت رابط کاربری آن به زبان C# به صورت Web-Application تولید شده است.

در این سامانه از معماری ASP.NET استفاده شده است. در ضمن از امکانات دیگری مانند HighChart برای نمایش نمودارها، JQuery، CSS برای طراحی رابط کاربری استفاده شده است.

جهت ارتباط دو بخش سامانه از Web API استفاده شده است و برای انتقال اطلاعات از رابط کاربری به API و دریافت پاسخ از JSON و متد Post بهره برده است.

## **پیاده‌سازی سری زمانی با زبان پایتون**

در این بخش با استفاده از زبان پایتون و ماژول تولید شده StatsModel آن سری زمانی تک متغیره با متد‌های ARMA، ARIMA، SARIMA پیاده سازی می‌نماییم. این مدل‌ها که همگی از انواع سری زمانی تک متغیره هستند، در ادامه توضیح داده خواهند شد.

2-1-1 تعریف انواع مدل‌های سری زمانی

در ادامه هر یک از مدلهای سری زمانی تولید شده توضیح داده خواهد شد.

2-1-1-1 مدل سری زمانی ARMA

این مدل جهت تحلیل داده‌های مورد نظر کاربر، مدل‌های میانگین متحرک (MA) و اتوریگرسیو(AR) با هم ترکیب می‌نماید. در این ماژول ایستا بودن دیتا به دلیل ساختار مدل بسیار مهم است و بدون آن این مدل پاسخگو نخواهد بود.

2-1-1-2 مدل سری زمانی ARIMA

این مدل جهت تحلیل داده‌های مورد نظر کاربر، علاوه بر ترکیب مدل‌های میانگین متحرک (MA) و اتوریگرسیو(AR)، بسیاری از اطلاعات ناایستا را پشتیبانی می‌نماید. این مدل خود وظیفه ایستا کردن اطلاعات را با امکان تفاضل‌گیری به عهده می‌گیرد. اما تعداد دوره تفاضل باید به مدل اعلام گردد.

2-1-1-3 مدل سری زمانی SARIMA

این مدل مانند مدل ARIMA است، با این تفاوت که روند فصلی اطلاعات را نیز پشتیبانی می‌نماید.

2-1-2 وقفه ها (Lags)

برای میانگین متحرک و اتورگرسیو می بایست مشخص کرد که با چند دوره قبل باید تحلیل انجام شد. برای این کار از متدهای Partial Autocorrelation Function(pacf) و Autocorrelation Function(acf) استفاده می‌نماییم، که یک نمودار در اختیار کاربر قرار می‌دهد که از طریق آن با شمارش تعداد دوره‌هایی که از قسمت آبی نمودار خارج شده، تعداد Lag های میانگین متحرک و اتورگرسیو را تشخیص می دهد. Pacf جهت دروه‌های اتورگرسیو و acf برای دوره ‌های میانگین متحرک است. پیاده‌سازی و نمودار نمونه در زیر آمده است.

قسمت API:

@app.route('/Pacf', methods=["Post"])

def GetPlots():

try:

data = json.loads(json.dumps(request.json['Data']))

fileName = ts.PlotAcf(data)

return jsonify({"fileName": fileName})

except:

return jsonify({'PredictResult': '', 'forecast': '', 'Exception': GetException()})

پیاده سازی متد تولید نمودار مذکور:

def PlotAcf(data):

df = pd.DataFrame.from\_dict(data)

df.riders = df.riders.astype('float32')

fig = plt.figure(figsize=(12,8))

ax1 = fig.add\_subplot(211)

data = df['riders'].iloc[1:]

lag = df.shape[0] - 2

fig = sm.graphics.tsa.plot\_acf(data, lags=lag,alpha=.65, ax=ax1,zero=False)

ax2 = fig.add\_subplot(212)

lag = lag / 2

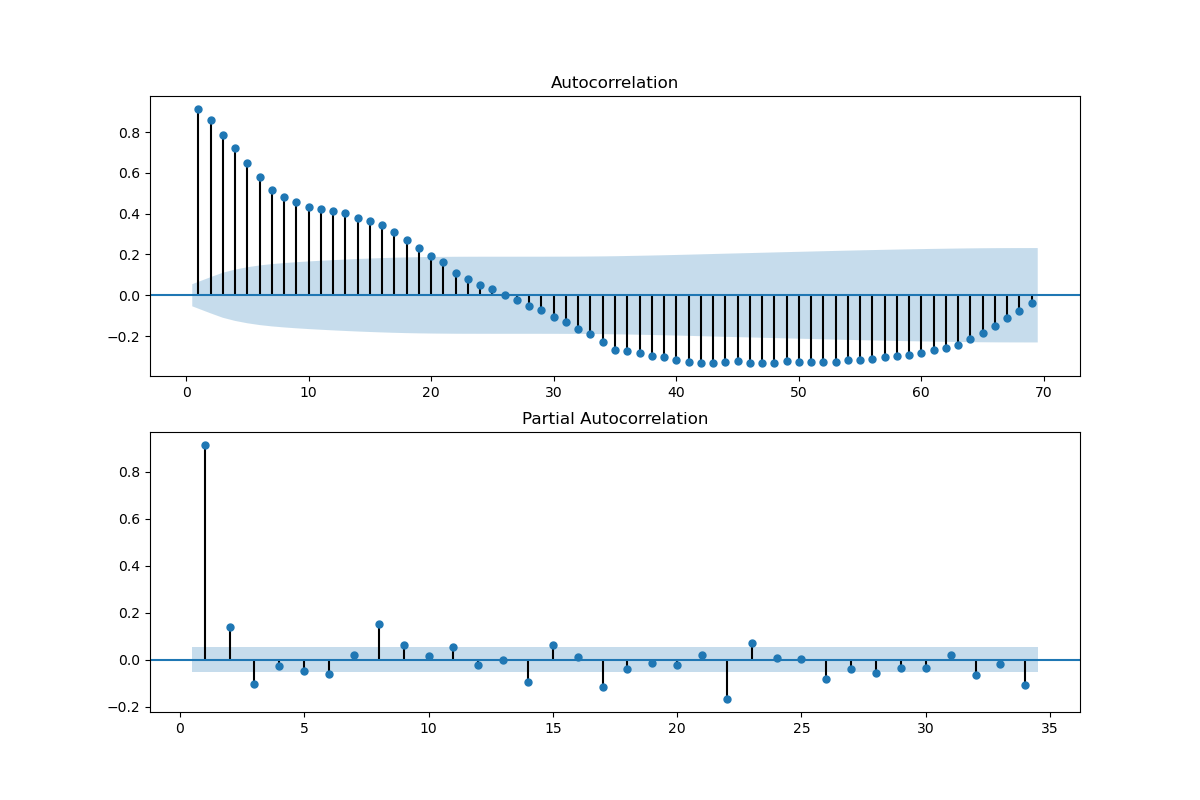
fig = sm.graphics.tsa.plot\_pacf(data, lags=lag,alpha=.65, ax=ax2,method='ywm',zero=False)

fileName = str(uuid.uuid4()) + ".png"

plt.savefig("PacfPlot/" + fileName)

return fileName

نمودار نمونه:



2-1-3 تجزیه و تحلیل روند اطلاعات (Partial Fraction Decomposition)

پس از یافتن وقفه‌ها (Lags)، نوبت به تحلیل روند اطلاعات است. به این معنا که روند عملکرد (Trend) و رفتار فصلی (Seasonal) و خطای رفتار فصلی (Resdual) شاخص مورد نظر را استخراج می‌کنیم. این اطلاعات به تشخیص وفقه‌های فصلی و میزان تفاضل مورد نیاز کمک می‌نمایید. در زیر پیاده سازی این متد آمده است و در نهایت نمودار‌های مورد نیاز روندها را به کاربر نمایش می‌دهد.

قسمت API:

@app.route('/decompose', methods=["Post"])

def GetDecompose():

try:

data = json.loads(json.dumps(request.json['Data']))

decomposition = ts.Decompose(data)

result = {"trend": dict(zip(decomposition.trend.index.format(), decomposition.trend.fillna('null'))), "resid": dict(zip(decomposition.resid.index.format(), decomposition.resid.fillna('null'))), "seasonal": dict(zip(decomposition.seasonal.index.format(), decomposition.seasonal.fillna('null'))), 'Exception': ''}

return jsonify(result)

except:

return jsonify({'PredictResult': '', 'forecast': '', 'Exception': GetException()})

پیاده سازی متد تجزیه:

def Decompose(data):

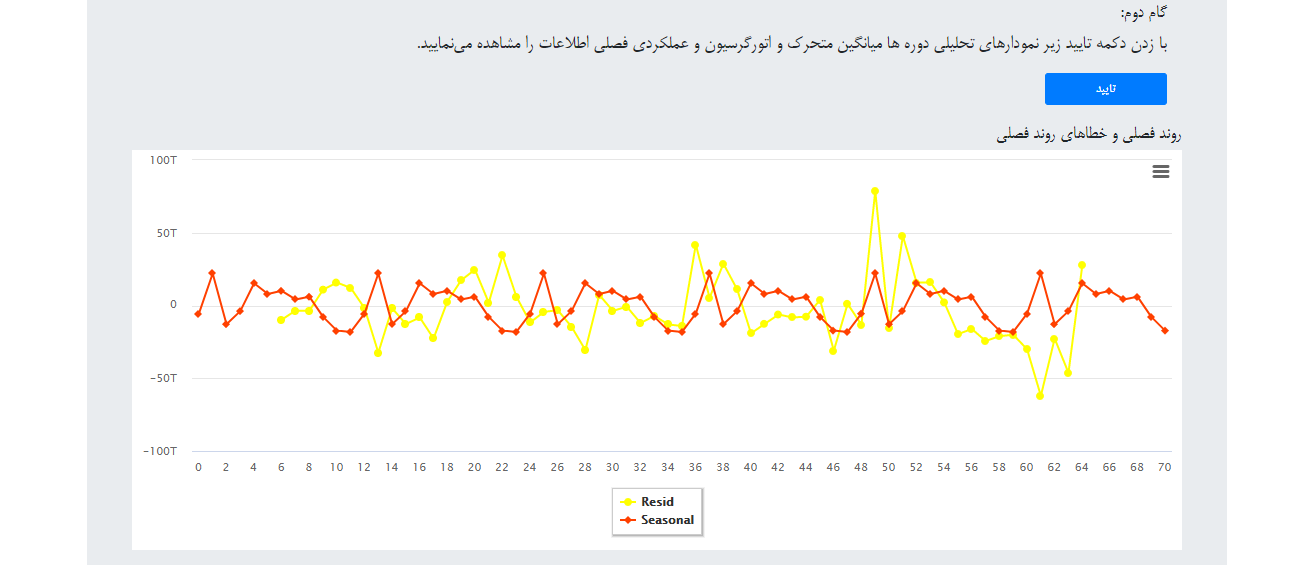
df = pd.DataFrame.from\_dict(data)

df.riders = df.riders.astype('float32')

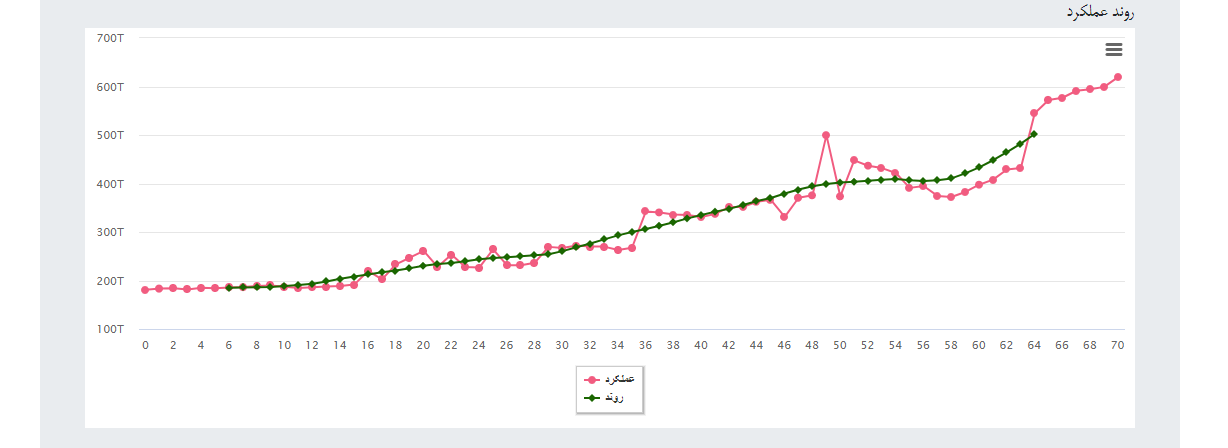
decomposition = seasonal\_decompose(df.riders, freq=12)

return decomposition

نمودار روند فصلی:



نمودار روند عملکرد:



2-1-4 پیشبینی (Forcasting)

در این مرحله امکان پیشبینی براساس سری زمانی را تولید می‌نماییم. در این بخش با توجه به مدل پیشبینی مورد نظر کاربر و تعداد وقفه‌های انتخاب شده توسط ایشان، پیشبینی انجام و نتیجه به صورت نمودار در اختیار کاربر قرار می‌گیرد. در این بخش برای تمامی مدل‌ها یک API تعریف شده است، اما هر مدل مجزا تولید گردیده است. در ادامه نحوه پیاده سازی آمده است.

2-1-4-1 پیاده سازی API

این وب سرویس هر سه مدل پیشبینی را دریافت و متد مربوطه را اجرا و نتیجه را به صورت JSON به سامانه ارائه می‌نماید. کد این وب سرویس در زیر آمده است.

@app.route('/ForeCast', methods=["Post"])

def ForeCast():

try:

perdictcount, forcastSteps, p, q, d, s, cType = GetParameters(request)

data = json.loads(json.dumps(request.json['Data']))

PredictResult,forecast,aic,hqic = ts.main(data, forcastSteps, perdictcount, p, q, d, s, cType)

result = {'PredictResult': PredictResult,

'forecast': forecast,

'aic': aic,

'hqic':hqic,

'Exception': ''}

return jsonify(result)

except:

return jsonify({'PredictResult': '', 'forecast': '', 'Exception': GetException()})

2-1-4-2 پیاده سازی ARMA

پیاده سازی ARMA در زیر آمده است.

def Arma(df):

mod = ARMA(df.riders, order=(\_p,\_q))

results = mod.fit()

aic = results.aic

hqic = results.hqic

bound = df.shape[0] - 1

lowerbond = bound - \_perdictcount

predict = results.predict(start = lowerbond + 1, end = bound, dynamic = False)

forecats = results.forecast(\_forcastSteps)

counter = 0

data = {}

for k in forecats[0]:

counter +=1

data[counter] =k

return predict, data,aic,hqic

2-1-4-3 پیاده سازی ARIMA

پیاده سازی ARIMA در زیر آمده است.

def Arima(df):

#try:

mod = ARIMA(df.riders, order=(\_p, \_d, \_q))

results = mod.fit()

aic = results.aic

hqic = results.hqic

bound = df.shape[0] - 1

lowerbond = bound - \_perdictcount

predict = results.predict(start = lowerbond + 1, end = bound, dynamic = False)

forecats = results.forecast(\_forcastSteps)

data = {}

counter =0

for k in forecats[0]:

counter +=1

data[counter] =k

return predict, data,aic,hqic

2-1-4-3 پیاده سازی SARIMA

پیاده سازی SARIMA در زیر آمده است.

def Sarima(df):

mod = sm.tsa.SARIMAX(df.riders, order=(\_p, \_d, \_q), seasonal\_order=(1, 1, 0, \_s), enforce\_stationarity=False)

results = mod.fit()

aic = results.aic

hqic = results.hqic

bound = df.shape[0] - 1

lowerbond = bound - \_perdictcount

predict = results.predict(start = lowerbond + 1, end = bound, dynamic = False)

forecats = results.forecast(\_forcastSteps)

data = {}

counter =0

for k in forecats:

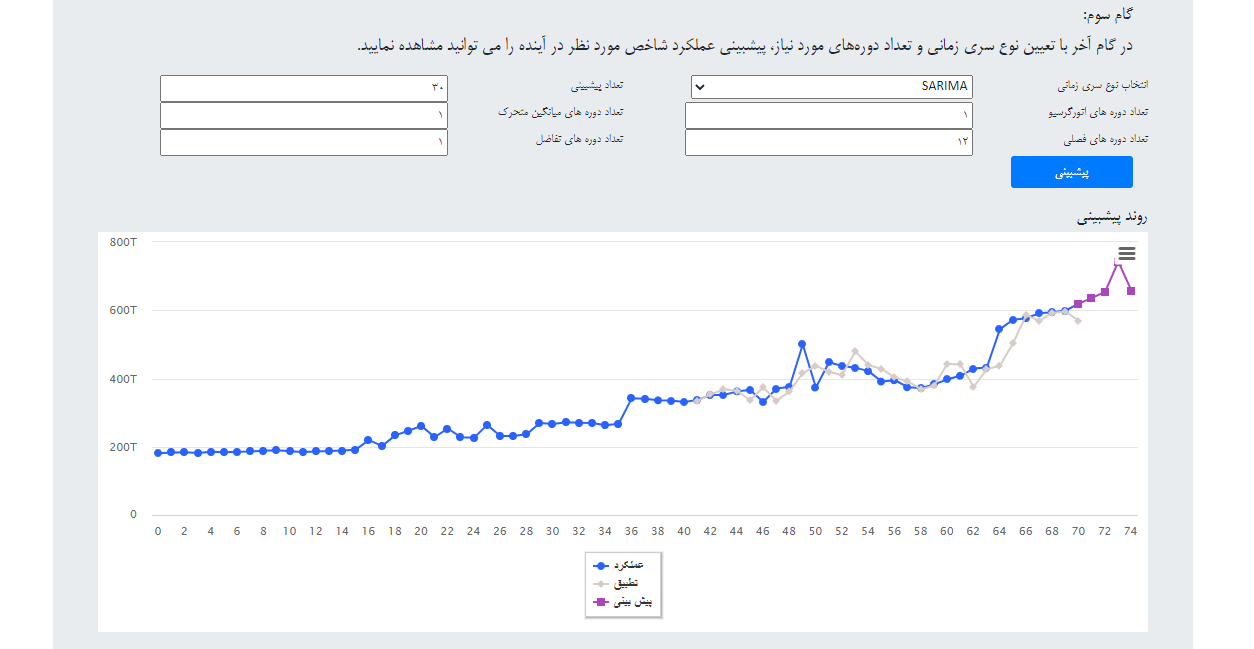
counter +=1

data[counter] =k

return predict, data,aic,hqic

2-1-4-3 نمودار پیشبینی

در این بخش نتیجه پیشبینی به صورت نمودار به کاربر نمایش داده می‌شود. در این نمودار قسمت سبز رنگ روند عملکرد شاخص‌ است. قسمت طوسی رنگ فرآیند تطبیق روند در پردازش سری زمانی است. و در نهایت در قسمت ارغوانی پیشبینی شاخص قابل مشاهده است.



## **پیاده‌سازی رابط کاربری**

این بخش با استفاده از زبان C# و در محیط ویژال استودیو با .NET Framework 4.7.2 پیاده سازی شده است. البته در یک کنترل نیز از زبان Visual Basic نیز بهره بداری گردیده است. در ضمن از امکانات دیگری مانند HTML, CSS, JQUERY نیز برای تولید رابط گرافیکی کاربر استفاده شده است. این برنامه شامل مدل‌های مختلف برای انتقال داده، سرویس‌های ارتباط با پایتون، فرم ارتباط با کاربر و کنترل‌های مورد نیاز است که موجب می‌گردد کاربر تعامل مناسب با این سامانه داشته باشد.

2-2-1 رابط گرافیکی

رابط گرافیکی که شامل قسمت‌های مختلف است، امکان ورود اطلاعات، مشاهده نمودارهای روند، تعیین مدل پیشبینی و در نهایت نمودار پیشبینی است که در ادامه به آنها خواهیم پرداخت.

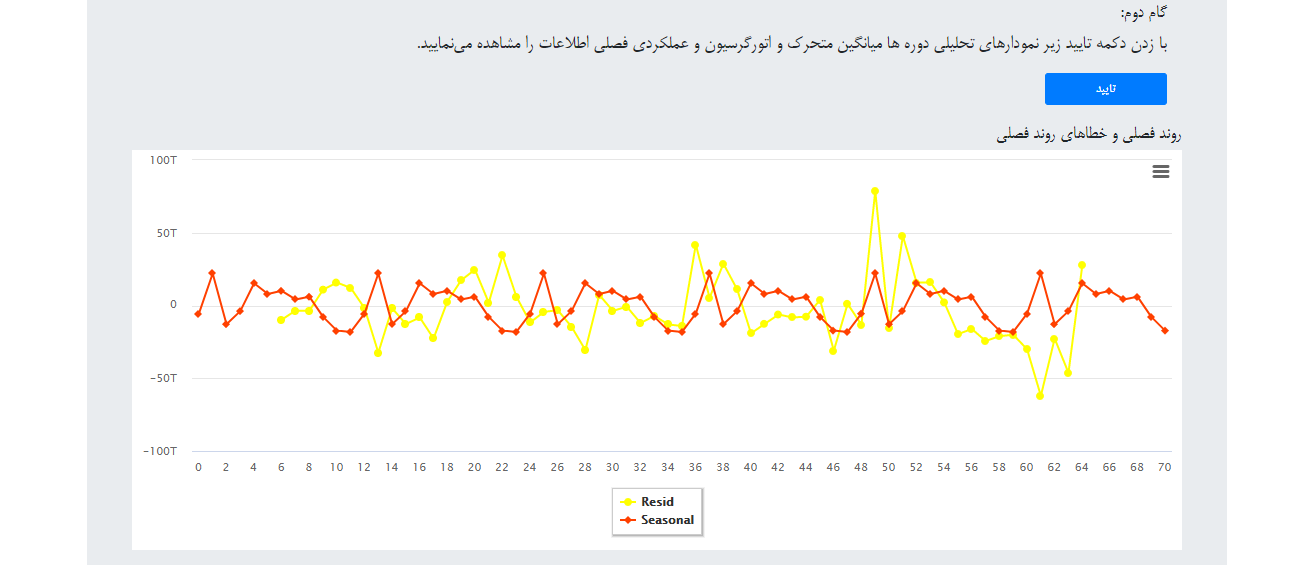
2-2-1-1 ورود اطلاعات عملکرد

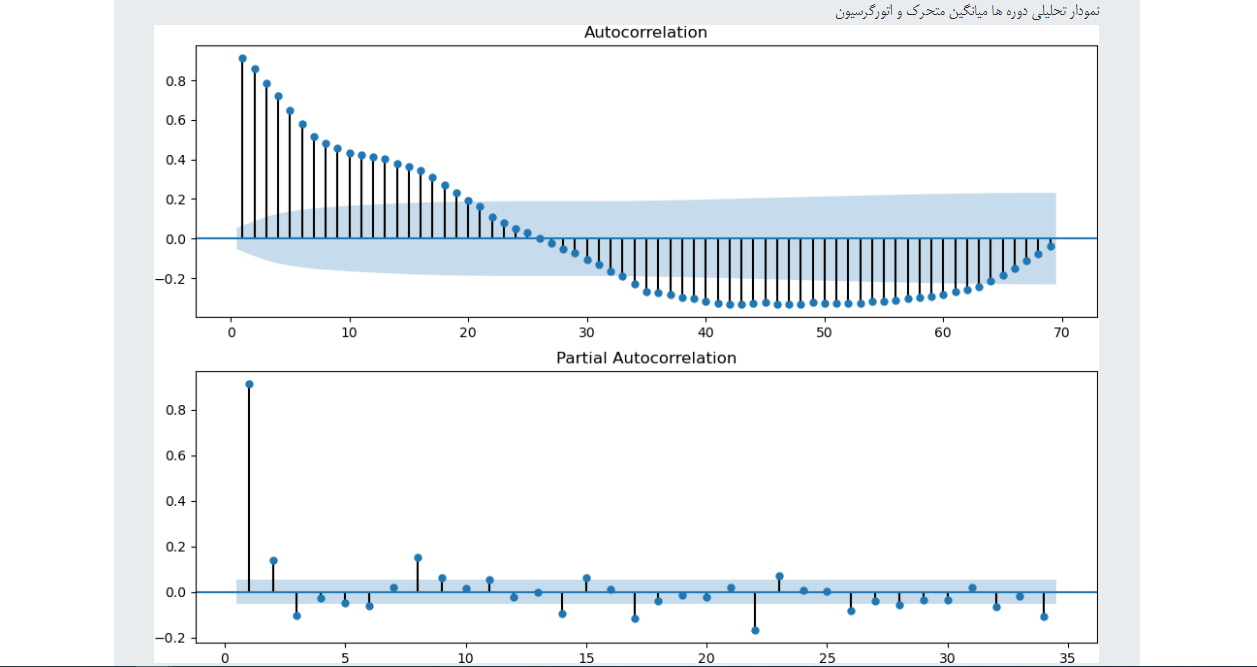
در این بخش کاربر امکان ورود اطلاعات روند عملکرد شاخص به صورت مقادیر جدا شده با کاما است. تصویر آن در زیر آمده است. پس از ورود اطلاعات کاربر با زدن دکمه تایید امکان تجزیه اطلاعات را خواهد داشت.



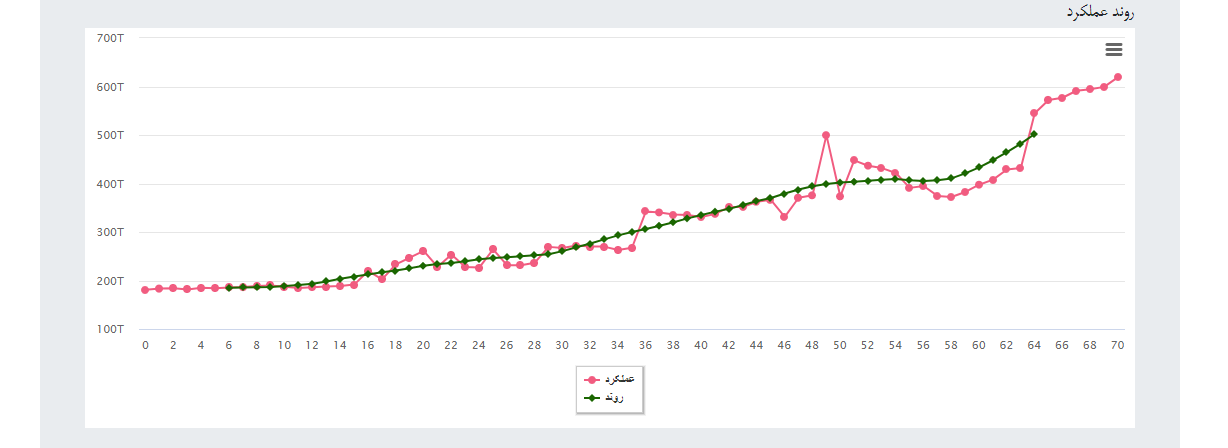
2-2-1-2 تجزیه و تحلیل اطلاعات

پس از ورود اطلاعات کاربر با زدن دکمه تایید امکان مشاهده نمودارهای روند عملکرد، فصلی و خطا را خواهد داشت و بر اساس آنها امکان تحلیل روند عملکرد شاخص برای ایشان فراهم خواهد گردید.



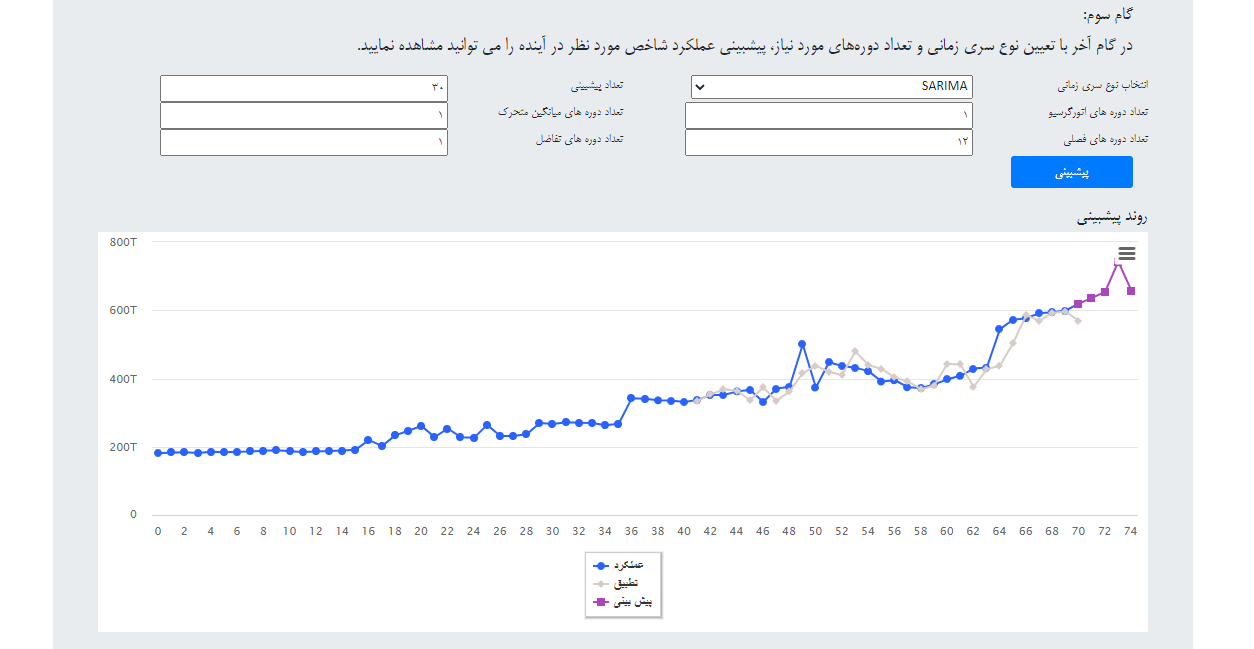


و در نهایت روند عملکرد شاخص نیز نمایش داده می شود. که در صفحه بعد نمودار آن نیز آمده است.



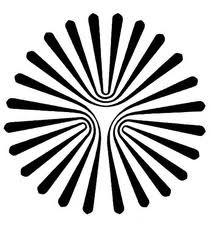
2-2-1-3 تعیین مدل‌ها و وقفه‌ها

در این قسمت کاربر سامانه مدل‌ها و وقفه‌های مورد نظر خود را برای میانگین متحرک، اتورگرسیو، وفقه‌های روند فصلی و تفاضلی تعیین می‌نماید و با زدن دکمه تایید به بخش پیشبینی خواهد رفت که در آنجا می‌تواند نمودار پیشبینی شاخص مورد نظر خود را مشاهده نماید.



## **تقدیر**

در پایان از زحمات استاد ارجمندم جناب آقای دکتر سیدعلی رضوی ابراهیمی کمال تشکر و امتنان را دارم و برای ایشان از درگاه خداوند سلامتی و بهروزی روز افزون آرزومندم.

Payam Noor University

**Developing time series by python**

A Project Report

Presented to:

Department of Information Technology and Communication

Faculty of Engineering

Payam Noor University of North Tehran

In Partial Fulfillment of the Requirement for the degree of

Bachelor of Science in

**Computer Engineering**

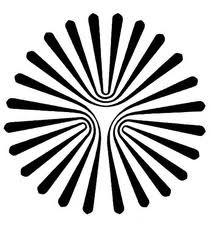
Advisor:

**DR. Seyed Ali Razavi Ebrahimi**

By:

**Farzad Sahraei**

Jan 2021

 Payam Noor University

**Developing time series by python**

A Project Report

Presented to:

Department of Information Technology and Communication

Faculty of Engineering

Payam Noor University of North Tehran

In Partial Fulfillment of the Requirement for the degree of

Bachelor of Science in

**Computer Engineering**

Advisor:

**DR. Seyed Ali Razavi Ebrahimi**

By:

**Farzad Sahraei**

Jan 2021