

شناخت،  
تحليل  
و پیش بینی بیت کوین



# بخش اول : شناخت

- ارز دیجیتال
  - مزایا و معایب آن
- بلاکچین
  - امنیت
  - نکات تکمیلی
  - بیت کوین و بلاکچین
  - ابزار هویت
- بیت کوین
  - استخراج بیت کوین
  - ارزش بیت کوین
  - پدیده نصف شدن بیت کوین
- صرافی و کیف پول

## بخش دوم: تحلیل

- میانگین متحرک و انواع آن
- ستون مد
- نقاط یا کراس اوور
- رگرسیون و انواع آن
- بررسی روش SVR و هسته‌های متفاوت



- ارز دیجیتال نوعی ارز است که فقط به صورت دیجیتال یا الکترونیکی در دسترس است و نه به صورت فیزیکی.
- به آن پول دیجیتال، پول الکترونیکی، ارز الکترونیکی یا پول نقد اینترنتی نیز گفته می‌شود.
- از آنجا انواع مختلفی از آن وجود دارند، ارزهای دیجیتال را می‌توان یک ابر مجموعه از ارزهای مجازی و رمز ارزها دانست.
- ارز رمزنگاری شده (رمزارز) نوع دیگری از ارز دیجیتال است که از رمزنگاری برای ایمن سازی و تأیید معاملات و مدیریت و کنترل ایجاد واحدهای جدید ارزی استفاده می‌کند.

- رمز ارزها این امتیاز را دارند که انتقال وجه مستقیماً بین دو طرف، بدون نیاز به شخص ثالث قابل اعتماد مانند بانک یا شرکت کارت اعتباری، اتفاق می‌افتد.
- امنیت این انتقال با استفاده از کلیدهای عمومی و کلیدهای خصوصی و اشکال مختلف سیستم‌های تشویقی، مانند اثبات کار (proof of work) یا اثبات سهام (proof of stake)، تأمین می‌شود.
- رای مثال : برای یک بلاک چین عمومی، تصمیم برای افزودن معامله به زنجیره با همفکری اتخاذ می‌شود. این بدان معنی است که اکثر "گره" ها (یا رایانه‌های موجود در شبکه) باید با معتبر بودن معامله موافقت کنند. افرادی که رایانه‌های موجود در شبکه را دارند انگیزه دارند تا معاملات را از طریق پاداش تأیید کنند. این فرایند به عنوان "اثبات کار" شناخته می‌شود.
- در سیستم های رمزنگاری مدرن، "کیف پول" – که جلوتر در خصوص آن بیشتر خواهیم دانست- یا آدرس حساب یک کاربر دارای یک کلید عمومی است، در حالی که کلید خصوصی فقط برای مالک شناخته شده است



- ماهیت نیمه ناشناس معاملات رمز ارزها، آنها را برای فعالیتهای غیرقانونی مانند پولشویی و فرار مالیاتی به خوبی مناسب می‌کند.
- با این حال، مدافعان رمز ارزها اغلب ناشناس بودن را بسیار با ارزش ارزیابی می‌کنند، و مزایای حفظ حریم خصوصی مانند حمایت از افشاگران یا فعالانی را که تحت دولت‌های سرکوبگر زندگی می‌کنند ذکر می‌کنند.

- بلاکچین یک دفتر مشترک و غیرقابل تغییر است که روند ثبت معاملات و ردیابی دارایی‌ها را در یک شبکه تجاری تسهیل می‌کند
- یک دارایی می‌تواند ملموس (یک خانه ، ماشین ، پول نقد ، زمین) یا نامشهود (مالکیت معنوی، حق ثبت اختراع، حق چاپ، مارک تجاری) باشد
- تقریباً هر چیزی که دارای ارزش باشد می‌تواند از طریق شبکه بلاکچین ردیابی و معامله شود

- تجارت بر اساس اطلاعات انجام می‌شود. هرچه سریعتر دریافت شده و دقیق‌تر باشد، بهتر است
- بلاکچین برای ارائه آن اطلاعات ایده‌آل است چراکه اطلاعات لحظه‌ای، اشتراکی و کاملاً شفاف ذخیره شده روی یک دفتر تغییرناپذیر را فراهم می‌کند که فقط اعضای مجاز شبکه می‌توانند به آن دسترسی داشته باشند.
- نام بلاکچین (زنجیره بلوک‌ها) اتفاقی انتخاب نشده است:
- دفتر دیجیتال اغلب به عنوان "زنجیره‌ای" توصیف می‌شود که از "بلوک" های جداگانه داده تشکیل شده است
- وقتی داده‌های تازه به صورت دوره‌ای به شبکه اضافه می‌شوند، یک "بلوک" جدید ایجاد می‌شود و به "زنجیره" متصل می‌شود
- چگونگی ایجاد این بلاک‌های جدید کلید امنیت بالای بلاکچین است



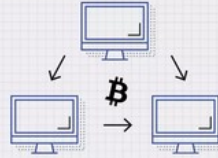
- قبل از افزودن بلوک جدید به دفتر، اکثر گره‌ها باید قانونی بودن داده‌های جدید را تایید کنند
- در خصوص رمز ارزها، ممکن است اطمینان حاصل شود که معاملات جدید در یک بلاک جعلی نباشد
- پس از توافق، بلوک به زنجیره اضافه می‌شود و معاملات مربوطه در دفتری توزیع شده ثبت می‌شوند، بلوک‌ها به طور ایمن به یکدیگر متصل می‌شوند و از ابتدای دفتر تا به حال یک زنجیره دیجیتالی امن تشکیل می‌دهند
- معاملات معمولاً با استفاده از رمزنگاری ایمن می‌شوند، به این معنی که گره‌ها برای پردازش یک معامله نیاز به حل معادلات پیچیده ریاضی دارند.
- به عنوان پاداش تلاش گره‌های شبکه در تأیید تغییرات در داده‌های به اشتراک گذاشته شده، گره‌ها معمولاً مقادیر جدیدی از واحد پول بلاکچین را پاداش می‌گیرند

## نکات تکمیلی بلاکچین

- بلاکچین نوع خاصی از پایگاه داده است.
- تفاوت آن با یک پایگاه داده معمولی در نحوه ذخیره اطلاعات است. بلاکچین داده‌ها را در بلوکی ذخیره می‌کند که سپس به هم زنجیر می‌شوند.
- با ورود اطلاعات جدید، آنها وارد یک بلوک جدید می‌شوند. هنگامی که بلوک با داده پر شد، به بلوک قبلی زنجیر شده، که باعث می‌شود داده‌ها به ترتیب زمانی با هم زنجیر شوند.
- انواع مختلف اطلاعات را می‌توان در زنجیره بلوکی ذخیره کرد اما بیشترین استفاده تاکنون به عنوان دفتر معاملات بوده است.
- در مورد بیت کوین، از بلاکچین به روشی غیر متمرکز استفاده می‌شود به طوری که هیچ شخص یا گروه واحدی روی آن کنترل ندارد - بلکه همه کاربران به طور جمعی کنترل خواهند داشت.
- بلاکچین‌های غیر متمرکز غیر قابل تغییر هستند، به این معنی که داده‌های وارد شده غیر قابل برگشت هستند. در مورد بیت کوین، این بدان معناست که معاملات به طور دائمی ثبت و برای همه قابل مشاهده هستند.



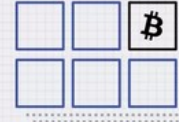
A new transaction is entered.



The transaction is then transmitted to a network of peer-to-peer computers scattered across the world.



This network of computers then solves equations to confirm the validity of the transaction.



Once confirmed to be legitimate transactions, they are clustered together into blocks.



These blocks are then chained together creating a long history of all transactions that are permanent.



The transaction is complete.

## بیت کوین و بلاکچین

- هدف بلاکچین این است که اطلاعات دیجیتالی ضبط و توزیع شود، اما ویرایش نشود
- فناوری بلاکچین برای اولین بار در سال 1991 توسط استوارت هابر و دبلیو اسکات استورنتتا، دو محقق که می خواستند سیستمی را اجرا کنند که در آن نمی توان مهر زمان سند را دستکاری کرد، ترسیم شد
- تقریباً دو دهه بعد، با راه اندازی بیت کوین در ژانویه 2009، بلاکچین اولین کاربرد واقعی خود را پیاده سازی کرد.
- پروتکل بیت کوین بر روی یک بلاکچین ساخته شده است
- بیت کوین از بلاکچین به عنوان ابزاری برای ثبت شفاف دفتر حساب استفاده می کند، ا
- ما از نظر تئوری بلاکچین می تواند برای ضبط غیر قابل تغییر هر تعداد داده استفاده شود.

## احراز هویت

- بلاکچین اصلی برای کار بدون مرجع مرکزی (به عنوان مثال بدون هیچ بانک یا تنظیم کننده‌ای که کنترل می‌کند که چه کسی تراکنش کند) طراحی شده است، اما معاملات هنوز باید احراز هویت شوند.
- این کار با استفاده از کلیدهای رمزنگاری انجام می‌شود، یک رشته داده (مانند رمز عبور) که کاربر را شناسایی می‌کند و به "حساب" یا "کیف پول" وی در سیستم امکان دسترسی می‌دهد.
- هر کاربر دارای کلید خصوصی خود و یک کلید عمومی است که همه می‌توانند آن را مشاهده کنند. استفاده از هر دوی آنها یک هویت دیجیتال امن ایجاد می‌کند تا کاربر را از طریق امضای دیجیتال احراز هویت کند و معامله ای را که می‌خواهد انجام دهد هموار سازد.
- همینطور که پیشتر گفته شد: هنگامی که معامله بین کاربران توافق شد، قبل از اینکه به یک بخش از زنجیره اضافه شود، باید تأیید یا مجاز شود.



- فرض بر این است که Satoshi Nakamoto یک اسم مستعار برای شخص یا افرادی است که پروتکل اصلی بیت کوین را در سال 2008 طراحی کرده و این شبکه را در سال 2009 راه اندازی کرده است.
- توسعه دهندگان بیت کوین مقدار آن را به 21 میلیون محدود کردند.
- هر بیت کوین را می توان به قطعات کوچکتر تقسیم کرد که کوچکترین کسر آن یکصد میلیونوم بیت کوین به نام "Satoshi" " - بنیانگذار ناکاموتو- است

# استخراج بیت کوین

- فرآیند استخراج شامل رایانه‌ها برای حل یک مسئله ریاضی بسیار چالش برانگیز است که به تدریج و با گذشت زمان دشوارتر می‌شود. هر بار که مسئله‌ای حل می‌شود، یک بلاک از بیت کوین پردازش می‌شود و استخراج کننده، بیت کوین جدید دریافت می‌کند.
- هر کاربر یک آدرس بیت کوین را برای دریافت بیت کوین‌های استخراج شده خود ایجاد می‌کند.
- مانند صندوق پستی مجازی با رشته‌ای از 27-34 عدد و حروف اما با این تفاوت که برخلاف صندوق پستی، هویت کاربر به آن پیوست نمی‌شود.
- برای اینکه ماینرهای بیت کوین در واقع با تایید معاملات، بیت کوین کسب کنند، دو اتفاق باید رخ دهد.
- اول، آنها باید معاملات به ارزش یک مگابایت را تایید کنند، که بسته به مقدار ذخیره داده‌های هر تراکنش، از نظر تئوری می‌تواند به اندازه یک تراکنش کوچک باشد اما اغلب چندین هزار است.
- دوم، برای افزودن مجموعه‌ای از معاملات به بلاکچین، ماینرها باید یک مسئله ریاضی محاسباتی پیچیده را که "اثبات کار" نیز نامیده می‌شود، حل کنند. آنچه آنها در واقع انجام می‌دهند تلاش برای دستیابی به یک عدد هگزادسیمال 64 رقمی است که "هش" نامیده می‌شود.
- سطح دشواری آخرین بلوک تا آگوست 2020 بیش از 16 تریلیون است. به این معنی که احتمال تولید هش توسط یک رایانه 1 از 16 تریلیون است.



هیچ کس مالک شبکه بیت کوین نیست، درست مانند اینکه هیچ کس صاحب فناوری ایمیل نیست. بیت کوین توسط همه کاربران بیت کوین در سراسر جهان کنترل می‌شود. در حالی که توسعه‌دهندگان در حال بهبود نرم‌افزار هستند، اما نمی‌توانند پروتکل بیت کوین را تغییر دهند زیرا همه کاربران در انتخاب نوع نرم‌افزار و نسخه خود آزاد هستند. برای اینکه سازگار با یکدیگر باشند، همه کاربران باید از نرم‌افزاری مطابق با همان قوانین استفاده کنند. بیت کوین فقط با یک توافق کامل بین همه کاربران می‌تواند به درستی کار کند. بنابراین، همه کاربران و توسعه‌دهندگان انگیزه قوی برای محافظت از این اتفاق نظر دارند.

## ارزش بیت کوین

- ارزها به این دلیل دارای ارزش هستند که می‌توانند به عنوان محل ذخیره ارزش و واحد مبادله مورد استفاده قرار گیرند
- ارزهای موفق دارای شش ویژگی اساسی هستند - کمبود، تقسیم پذیری، سودمندی، قابلیت حمل، دوام و جعل تقلبی.
- رمز ارز بیت کوین دارای ارزش است زیرا در خصوص این شش ویژگی بسیار خوب پاسخ داده است، اگرچه بزرگترین مسئله آن جایگاه آن به عنوان یک واحد مبادله است زیرا بیشتر مشاغل هنوز آن را به عنوان پرداخت قبول نکرده اند
- برخلاف سرمایه‌گذاری در ارزهای سنتی، بیت کوین توسط بانک مرکزی صادر نمی‌شود یا توسط دولت پشتیبانی نمی‌شود. بنابراین، سیاست پولی، نرخ تورم و اندازه‌گیری رشد اقتصادی که به طور معمول بر ارزش ارز تاثیر می‌گذارد، در مورد بیت کوین اعمال نمی‌شود.

# ارزش بیت کوین

قیمت‌های بیت کوین تحت تأثیر عوامل زیر است:

- عرضه بیت کوین و تقاضای بازار برای آن
- هزینه تولید بیت کوین از طریق فرآیند استخراج
- پاداش‌هایی که برای استخراج بیت کوین برای تایید معاملات بلاکچین صادر می‌شود
- تعداد رمز ارزهای رقیب
- صرافی‌هایی که در آن معاملات انجام می‌شود
- آیین نامه‌های مربوط به فروش آن
- حاکمیت داخلی آن



## پدیده نصف شدن بیت کوین

- نصف شدن بیت کوین رویدادی است که طی آن پاداش بلوک استخراج بیت کوین جدید به نصف کاهش می یابد
- استخراج کنندگان بیت کوین برای هر معامله ای که تایید می کنند 50٪ بیت کوین کمتری دریافت می کنند
- نصف شدن BTC هر 210,000 بلوک اتفاق می افتد ، که برابر است با نصف شدن تقریباً هر 4 سال یکبار.
- روند نصف شدن بیت کوین از تئوری اقتصادی رمز ارز پیروی می کند: از آنجا که بیت کوین مقدار محدودی دارد و به مرور زمان عرضه آن کاهش می یابد، قیمت بیت کوین می تواند با کاهش موجودی کلی "پایدار" و تورم پذیر شود - به همین دلیل نصف شدن بیت کوین وجود دارد.

## پدیده نصف شدن بیت کوین

همانطور که از جدول زیر مشاهده می کنید ، مقدار بیت کوین استخراج شده و پاداش بلوک در هر رویداد نصف شدن به نصف کاهش می یابد. تا سال 2032 ، بیش از 99٪ بیت کوین استخراج خواهد شد و تخمین زده می شود که تا سال 2140 طول بکشد تا 100٪ کل بیت کوین استخراج شود.

Halving	Date	Block	Block reward	Mined in period	% mined
BTC launch	3 January 2009	0	50	10,500,00	50
Halving 1	28 November 2012	210,000	25	5,250,000	75
Halving 2	9 July 2016	420,000	12.5	2,625,000	87.5
Halving 3	May 2020	630,000	6.25	1,312,500	93.75
Halving 4	Expected 2024	840,000	3.125	656,250	96.875
Halving 5	Expected 2028	1,050,000	1.5625	328,125	98.4375
Halving 6	Expected 2032	1,260,000	0.78125	164,062.5	99.21875

- کیف پول رمز ارز یک وسیله، رسانه فیزیکی، برنامه یا خدماتی است که کلیدهای عمومی و / یا خصوصی را برای معاملات رمز ارز ذخیره می کند
- تکلیف پول ها بر اساس نحوه نگهداری کلید خصوصی به دو دسته سرد (Cold) و داغ (Hot) تقسیم می شوند:
- کیف پول های سرد، که دارایی را بدون اتصال به اینترنت و به صورت آفلاین نگهداری می کنند و تنها در مواقع نیاز مثل انتقال بیت کوین، به اینترنت متصل می شوند. این نوع کیف پول ها به خاطر شیوه نگهداری کلید خصوصی، امنیت بیشتری دارند و عمده‌تاً برای سرمایه‌گذاری بلندمدت استفاده می شوند.
- کیف پول های داغ که در آنها دارایی در شرایطی نگهداری می شود که دسترسی به اینترنت وجود دارد. استفاده از یک کیف پول گرم لزوماً به این معنا نیست که کلید خصوصی در سرورهای اینترنتی نگهداری می شود، بلکه به متصل بودن دستگاه به اینترنت اشاره دارد.
- کیف پول های بیت کوین در طبقه بندی دیگری به دو نوع فیزیکی و نرم افزاری تقسیم می شوند. کیف پول فیزیکی خود شامل دو نوع کیف پول سخت افزاری و کیف پول کاغذی است. کیف پول های نرم افزاری نیز به سه دسته کیف پول های دسکتاپ، موبایل و تحت وب تقسیم می شوند

کیف پول‌های زیر در حال حاضر بهترین گزینه‌های موجود برای ذخیره بیت کوین هستند:

- کیف پول نرم‌افزاری Trust Wallet (موبایل)
- کیف پول سخت‌افزاری Ledger
- کیف پول سخت‌افزاری Trezor
- کیف پول نرم‌افزاری Mycelium (موبایل)
- کیف پول نرم‌افزاری BlueWallet (موبایل)
- کیف پول نرم‌افزاری Electrum (دسکتاپ و موبایل)
- کیف پول نرم‌افزاری Exodus (دسکتاپ و موبایل)
- کیف پول نرم‌افزاری Bitcoin Wallet (موبایل)
- کیف پول نرم‌افزاری Blockchain.com (تحت وب)
- کیف پول کاغذی بیت کوین با وب‌سایت Bitaddress.org (ذخیره‌سازی سرد)

- برای شروع کار، به یک صرافی رمز ارز نیاز دارید که بتوانید ارزهای دیجیتال مانند Ethereum ، Bitcoin و Dogecoin را خریداری و بفروشید.
- براساس یک نظرسنجی، سه صرافی برتر ارزهای رمزنگاری شده Huobi ، Binance و OKEX هستند
- همچنین: بهترین صرافی های رمز ارز در سال 2021:
- بهترین به طور کلی: Coinbase و Coinbase Pro
- بهترین برای مبتدیان: Cash App
- Altcoins: Binance برای
- بهترین صرافی غیرمتمرکز: Bisq



## بخش دوم: تحلیل روند بیت کوین روی داده ی یک ساعته

- میانگین متحرک و انواع آن
- ستون مد
- نقاط یا کراس اوور
- رگرسیون و انواع آن
- بررسی روش SVR و هسته های متفاوت





## میانگین متحرک Moving Average

- محاسبه‌ای است برای تجزیه و تحلیل نقاط داده با ایجاد یک سری میانگین از زیرمجموعه‌های مختلف مجموعه داده‌ی کامل
- هر چه مدت زمان میانگین متحرک بیشتر باشد ، تاخیر یا lag بیشتر خواهد بود
- ارقام میانگین متحرک 50 و 200 روزه سهام به طور گسترده توسط سرمایه گذاران و معامله گران دنبال می شود و به عنوان سیگنال ای مهم تجاری در نظر گرفته می شوند.

## میانگین متحرک ساده (SMA)

- ساده‌ترین شکل میانگین متحرک که به عنوان میانگین متحرک ساده شناخته می‌شود، با در نظر گرفتن میانگین حسابی مجموعه‌ای از مقادیر مشخص شده محاسبه می‌شود.
- $$\text{Simple Moving Average} = (A_1 + A_2 + \dots + A_n) / n$$

where:

$A_n$  = قیمت مورد نظر در زمان  $n$   
 $n$  = تعداد کل زمان‌ها

## میانگین متحرک نمایی (EMA)

- میانگین متحرک نمایی نوعی از میانگین متحرک است که در تلاش برای پاسخگویی بیشتر به اطلاعات جدید، به قیمت‌های اخیر وزن بیشتری می‌بخشد.
  - محاسبه ی EMA :
1. میانگین متحرک ساده را در یک بازه زمانی خاص محاسبه کنید
  2. ضریب توزین EMA که به عنوان "عامل هموارکننده" گفته می‌شود را محاسبه کنید، که به طور معمول از فرمول زیر پیروی می‌کند:
- $$[2 / ( \text{دوره زمانی انتخاب شده} + 1 )]$$

## میانگین متحرک نمایی (EMA)

- فرمول نهایی محاسبه‌ی میانگین متحرک نمایی Exponential Moving Average به شکل زیر است :

- $$EMA_t = [V_t \times (s/1+d)] + EMA_y \times [1-(s/1+d)]$$

where:

$EMA_t$  = میانگین متحرک نمایی امروز

$V_t$  = ارزش امروز

$EMA_y$  = میانگین متحرک نمایی دیروز

$s$  = فاکتور هموارسازی

$d$  = تعداد روزها

- با فرمولی نوشتاری نیز به صورت زیر تعریف میشود :

- $$\text{Current EMA} = [\text{Closing Price} - \text{EMA (Previous Time Period)}] \times \text{Multiplier} + \text{EMA (Previous Time Period)}$$



## میانگین متحرک وزنی

- میانگین‌های متحرک وزنی وزنه‌ی سنگین‌تری را به نقاط داده فعلی نسبت می‌هند ، زیرا این نقاط مرتبط‌تر از نقاط داده در گذشته‌های دور هستند. مجموع وزن باید به 1 ( یا 100 درصد ) برسد.
- $$WMA = [Price_1 \times n + Price_2 \times (n-1) + \dots + Price_n] / [n \times (n+1) / 2]$$

## نکات تحلیلی میانگین‌ها

- از آنجا که یک میانگین متحرک نمایی EMA از یک ضریب وزنی نمایی استفاده می‌کند تا وزن بیشتری به قیمت‌های اخیر بدهد، برخی معتقدند که این یک شاخص بهتر از روند در مقایسه با WMA یا SMA است.
- برای داده‌های کوتاه مدت از EMA باید استفاده کرد، چون نسبت به تغییرات آنی سریع‌تر واکنش نشان می‌دهد.
- مناسب‌ترین میانگین متحرک برای تحلیل میان مدت، WMA است؛ چون سرعت واکنش به داده‌های کوتاه مدت از SMA سریع‌تر ولی از EMA کمتر است.
- برای تحلیل‌های بلند مدت هم به ترتیب اهمیت، استفاده از میانگین‌های SMA و در ادامه WMA می‌تواند مفید واقع شود.

- ستون مد، ستونی است که در هر خانه‌ی آن، داده‌ی پر تکرار ستون‌های قبلی ذخیره شده است.
- اکثر ستون‌های میانگین متحرک فاقد مد بودند؛ این به این معنی است که داده‌ی تکراری در ستون‌های میانگین وجود ندارد یعنی در روند تشکیل شده توسط میانگین‌های متحرک، به قله‌ها یا دره‌های یکسان نمی‌خوریم. پس میتوان این نتیجه را گرفت که :  
" میانگین‌های متحرک برای بررسی روندهای سایید مناسب نیستند. "



## جدول های فراوانی

- جداول فراوانی با طول طبقات 5، 10، 15 و 20 برای هر یک از قیمت ها و میانگین ها رسم شد و نکات زیر مورد نظر واقع شد :
- با توجه به فراوانی بالای دو دسته ی (6000-12000) و (12000-18000) به این نکته دست می یابیم که روند حرکتی بیت کوین از قیمت 6000 تا 18000 به صورت صعودی اما با نوسانی ملایم بوده چون در این دو بازه قیمت های نزدیک به هم را ثبت کرده است .
- در بازه های قیمتی 18000 تا 60000 روند نوسانی به مراتب بیشتری را طی کرده اما به دست آوردن الگوی این نوسان نمی تواند کار دشواری باشد؛ چون از قیمت 18000 تا 60000 در بازه های پشت سر هم 3000 تایی، دسته های جدول فراوانی ما دارای فراوانی های نزدیک به هم هستند.

## تقاطع یا Crossover

- کراس اوور نقطه‌ای از نمودار معاملاتی است که در آن قیمت یک اوراق بهادار و یک خط نشانگر فنی ( اندیکاتور) با یکدیگر تلاقی می‌کنند، یا زمانی که دو شاخص خود با یکدیگر برخورد می‌کنند. از کراس اوورها برای تخمین عملکرد یک ابزار مالی و پیش‌بینی تغییرات در روند مانند reversals یا breakouts استفاده می‌شود
- تقاطع میانگین 5 دوره‌ای از سمت پایین به سمت بالا با میانگین 15 دوره‌ای نشان‌دهنده یک breakout است. این امر همچنین نشان‌دهنده روند صعودی است که از پستی و بلندی‌های بالاتری ساخته می‌شود. مشاهده میانگین متحرک پنج دوره‌ای که از 15 دوره به سمت پایین متقاطع می‌شود، یک شکست یا breakdown را نشان می‌دهد. این نیز نشان‌دهنده روند نزولی است که متشکل از بالا و پایین‌های پایین‌تری است. (کف‌ها نسبت به کف قبلی خود پایین‌تر هستند).
- جای میانگین 5 دوره‌ای و میانگین 15 دوره‌ای، به ترتیب هر میانگین کوتاه مدت و بلند مدت (میان مدت به جای بلند مدت) دیگری هم می‌توان استفاده کرد.



	A	B	C	D	L	M	U	V
	openTime	open	MA for open - int: 5	open crossover with MA(5)	EMA for open - int: 5	open crossover with EMA(5)	WMA for open - int: 5	open crossover with WMA(5)
1								
2	1.59E+12	7266.11						
3	1.59E+12	7329.9						
4	1.59E+12	7367.39						
5	1.59E+12	7349.58			0			
6	1.59E+12	7311.39	7324.874		2441.6247		7332.2233	
7	1.59E+12	7322.08	7336.068		4073.1058		7331.292	
8	1.59E+12	7252.8	7320.648		5155.6199		7303.536	
9	1.59E+12	7278.63	7302.896		5871.3786		7289.53	
10	1.59E+12	7307.11	7294.402	cross up	6345.7197		7290.9347	cross up
11	1.59E+12	7377.26	7307.576		6666.3385		7318.554	
12	1.59E+12	7356.65	7314.49		6882.389		7334.912	
13	1.59E+12	7372.24	7338.378		7034.3853		7354.162	
14	1.59E+12	7420.83	7366.818		7145.1962		7381.646	
15	1.59E+12	7381.03	7381.602	cross down	7223.9981		7386.3833	cross down

مثالی از میانگین متحرک‌ها و تقاطع‌ها در فایل اکسل





## تحلیل و پیش‌بینی قیمت بیت کوین براساس روش‌های مختلف یادگیری ماشین



- از معروف‌ترین این تکنیک‌ها، رگرسیون نام دارد؛ که با بکارگیری آن می‌توان چگونگی رفتار یک متغیر (متغیر وابسته)، نسبت به یک یا چند متغیر دیگر (متغیر یا متغیرهای مستقل) را بدست آورد
- رگرسیون برای بررسی رفتار یک متغیر وابسته مورد استفاده قرار می‌گیرد؛ اما این تعریف بسیار کلی است. منظور از بررسی چگونگی رفتار یعنی نسبت دادن (فیت کردن) یک تابع ریاضی با عبارت مشخص به متغیر وابسته به گونه‌ای که اختلاف تک تک نقاط داده با این عبارت ریاضی، بسیار کم باشد
- در این کار، متغیر مستقل، زمان و متغیر وابسته، میانگین قیمت‌های OHLC داده‌ی بیت کوین است.
- واژه‌ی OHLC مخفف چهار کلمه‌ی High، Low، Open و Close است که به ترتیب به قیمت‌های یک کندل بیت کوین در هنگام باز شدن، بیشترین قیمت آن کندل، کمترین قیمت همان کندل و در آخر قیمت بسته شدن کندل مورد نظر اشاره دارد.

## رگرسیون خطی

- در این نوع از رگرسیون ، یک تابع خطی به فرم  $y = wx + b$  را به نقاط داده خود نسبت می دهیم
- هدف از نسبت دادن خط یا هر نوع تابع دیگری به مجموعه نقاط این است که فاصله آن تابع نسبت به تک تک نقاط حداقل ممکن باشد. برای اندازه گیری این فاصله از مفهومی به نام تابع هزینه استفاده می کنیم.
- تابع هزینه مورد استفاده در این مساله ، تابع SSE یا Sum Square Error است که به معنی مجموع مربعات خطا است و فرمولی به شکل زیر دارد :

$$\sum_{i=1}^M (y_i - \hat{y}_i)^2 = \sum_{i=1}^M \left( y_i - \sum_{j=0}^p w_j \times x_{ij} \right)^2$$

- که در آن مقدار واقعی و مقدار تولید شده توسط تابع فیت شده ی ماست؛ که چون از رگرسیون خطی صحبت می کنیم مقدار برابر با خواهد بود.

## توضیح کد رگرسیون خطی

```
from sklearn.linear_model import LinearRegression
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats.stats import pearsonr
from scipy import stats
from google.colab import drive
drive.mount('/content/gdrive')
```

- کتابخانه‌های numpy و pandas برای دستکاری راحت داده ها
- Matplotlib برای تصویر سازی توابع و داده ها
- Scipy برای محاسبه‌ی معیار همبستگی پیرسون و حساب کردن مقدار PPF که در واقع معکوس تابع چگالی احتمالی مشخص می‌کند به ازای چه مقداری، مقدار احتمال در داده‌ای با توزیع نرمال مقدار مشخص شده به عنوان ورودی این تابع است
- کتابخانه‌ی google.colab و در کل دو خط آخر برای خواندن فایل داده ها از گوگل درایو است
- خط اول هم همانطور که از ظاهرش مشخص است به جهت استفاده از مدل رگرسیون خطی نوشته شده است.

## توضیح کد رگرسیون خطی

بخش بعدی برای ساختن متغیر مستقل است. در این جا متغیر مستقل ، ساعت است که با شماره به آن اشاره خواهیم کرد. پس ما نیاز داریم به تعداد داده‌های متغیر وابسته ، ساعت یا متغیر مستقل داشته باشیم. به همین دلیل به شکل زیر عمل می‌کنیم:

```
x_data = np.arange(0, len(y_data), 1)
x_data_composed = x_data.reshape(-1, 1)
```

خط دوم صرفاً برای تغییر بعد در متغیر مستقل است. به این صورت که متغیر مستقل را به صورت یک بردار ستونی به برنامه بشناسانیم.

- این بخش مربوط به خواندن تمام فایل اکسل یا همان my\_pandas\_file و همینطور ستون y\_data به عنوان متغیر وابسته است :

```
my_pandas_file = pd.read_excel('/content/gdrive/My Drive/BTC.xlsx')
y_data = my_pandas_file.get('mean')
y_data
```

```
0      7298.5025
1      7363.5200
2      7365.5975
3      7322.9950
4      7306.2575
```

```
...
8995   62049.8850
8996   62092.6850
8997   62257.4600
8998   62012.4200
8999   61648.9850
```

```
Name: mean, Length: 9000, dtype: float64
```

## توضیح کد رگرسیون خطی

```
correlation = pearsonr(x_data, y_data)
correlation
```

```
(0.8759949327183532, 0.0)
```

که نتیجه میگیریم متغیرهای مساله با یکدیگر رابطه مستقیم قوی دارند.

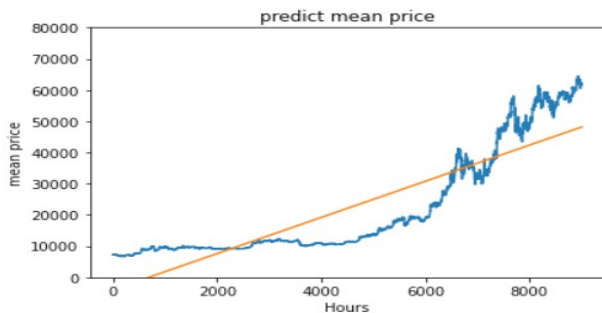
- پس از این بخش، حساب کردن نوع و قوت رابطه ی بین متغیر مستقل و وابسته را در پیش داریم که طبق گفته های بالا از معیار پیرسون برای حساب کردن این همبستگی استفاده می کنیم. دلیل این کار نیز این است که تکنیک رگرسیون برای متغیرهای مستقل و وابسته ای که دارای روابط معکوس یا مستقیم به نسبت قوی (قدر مطلق عدد بدست آمده از رابطه پیرسون از یک عدد مشخصی بزرگتر باشد) هستند به خوبی جواب می دهد و برای متغیرهایی که عدد پیرسون آن ها نزدیک به صفر در می آید؛ یعنی متغیرهایی که به یکدیگر مرتبط نیستند؛ جوابگو نخواهد بود. کد این بخش نیز به صورت روبه رو خواهد بود:



## توضیح کد رگرسیون خطی

```
lin_regression = LinearRegression().fit(x_data_composed, y_data)
model_line = lin_regression.predict(x_data_composed)
```

```
plt.plot(y_data)
plt.plot(model_line)
plt.xlabel('Hours')
plt.ylabel('mean price')
plt.title('predict mean price')
plt.ylim(0,80000)
```



- عکس اول از بالا در مورد فیت کردن مدل رگرسیون خطی روی متغیرهای مستقل و وابسته است. در خط اول یک رگرسیون خطی با قرار دادن `X_data_composed` به جای متغیر مستقل و `y_data` در نقش متغیر وابسته می‌سازیم و آن را متناسب با متغیرهای خود آموزش می‌دهیم (train) در خط دوم نیز بار دیگر متغیر مستقل خود را به مدل رگرسیونی آموزش دیده شده می‌دهیم تا به ازای هر مقدار متغیر مستقل، یک مقدار برای متغیرهای وابسته تخمین بزند.

- در ادامه به تصویر سازی داده و رگرسیون خطی فیت شده روی آن می‌رسیم (عکس دوم)

- خروجی کد تصویر سازی در عکس سوم به خوبی قابل مشاهده است.

## توضیح کد رگرسیون خطی

```
def get_prediction_interval(prediction, y_test, test_predictions, pi=.95):
```

```
...
```

```
Get a prediction interval for a linear regression.
```

```
INPUTS:
```

```
- Single prediction,  
- y_test  
- All test set predictions,  
- Prediction interval threshold (default = .95)
```

```
OUTPUT:
```

```
- Prediction interval for single prediction  
...
```

```
#get standard deviation of y_test
```

```
sum_errs = np.sum((y_test - test_predictions)**2)  
stdev = np.sqrt(1 / (len(y_test) - 2) * sum_errs)
```

```
#get interval from standard deviation
```

```
one_minus_pi = 1 - pi  
ppf_lookup = 1 - (one_minus_pi / 2)
```

```
z_score = stats.norm.ppf(ppf_lookup)
```

```
interval = stdev * z_score
```

```
#generate prediction interval lower and upper bound x_data.composed
```

```
lower, upper = prediction - interval, prediction + interval
```

```
return lower, prediction, upper
```

- برای کشیدن کانال رگرسیون از روش معروف ضرب انحراف استاندارد در مقدار تابع **ppf** استفاده می‌کنیم. کد این بخش، دو آرایه‌ی مقدار واقعی و مقادیر تخمین زده شده و همینطور یک نقطه‌ی تخمین زده شده‌ی خاص و در آخر مقدار درصد پوشش کانال ( $\pi$ ) را می‌گیرد و با حساب کردن انحراف استاندارد مقادیر تخمین زده شده نسبت به مقادیر واقعی و همینطور محاسبه عددی که باعث می‌شود تا 95 درصد از داده‌های توزیع نرمال با محوریت فاصله از میانگین، پوشش داده شوند، و در آخر ضرب این دو مقدار، دامنه‌ی کانال را تعیین می‌کند. در آخر این مقدار مشخص را یک بار با نقطه‌ی تخمین زده شده خاص جمع و یکبار از آن کم می‌کنیم تا بالا و پایین کانال مشخص شود.

## توضیح کد رگرسیون خطی

```
## Plot and save confidence interval of linear regression - 95% x_data.composed

lower_vet = []
upper_vet = []

for i in model_line:
    lower, prediction, upper = get_prediction_interval(i, y_data, model_line)
    lower_vet.append(lower)
    upper_vet.append(upper)

plt.fill_between(np.arange(0, len(y_data), 1), upper_vet, lower_vet, color='b', label='Confidence Interval')

plt.plot(np.arange(0, len(y_data), 1), y_data, color='orange', label='Real data')

plt.plot(model_line, 'k', label='polynomial-SVR')

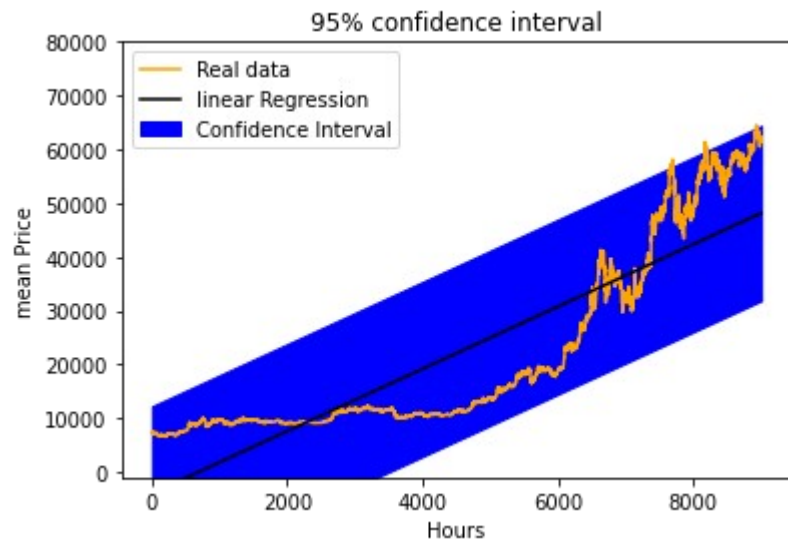
plt.xlabel('Hours')
plt.ylabel('mean Price')

plt.title('95% confidence interval')

plt.legend()
plt.ylim(-1000, 80000)
plt.show()
```

- در نهایت به بخش ترسیم داده و مدل رگرسیونی، این بار به همراه کانال رگرسیونی محاسبه شده براساس تک تک نقاط، می‌رسیم. در این بخش نیز به ازای تک تک نقاط، حدود پایین و بالای بازه ی کفایت را حساب می‌کنیم و هر کدام را در آرایه های در نظر گرفته برای هر کدام از این مقادیر، به ترتیب نقاط، ذخیره می‌کنیم. سپس با استفاده از امکانات کتابخانه matplotlib به ترسیم موارد مورد نظر می‌پردازیم. کد این قسمت و در ادامه خروجی کار در حالتی که رگرسیون ما خطی است قابل مشاهده خواهد بود:

## توضیح کد رگرسیون خطی



- خروجی تصویر سازی رگرسیون و کانال بدست آمده به شکل روبه رو است.

- مشاهده می کنیم که داده ی مورد نظرمون هم به خوبی با رگرسیون خطی تخمین زده شده و هم بازه ی کفایت به خوبی نمودار داده را با حداقل طول مورد نیاز در خود جای داده است. پس برای مساله ی ما ، رگرسیون خطی جواب می دهد.

## رگرسیون ریدج (Ridge)

- در رگرسیون Ridge ، عبارتی به اسم عبارت جریمه به تابع هزینه اضافه می‌شود که برابر با مربع اندازه بردار وزن  $w$  است.

$$\sum_{i=1}^M (y_i - \hat{y}_i)^2 = \sum_{i=1}^M \left( y_i - \sum_{j=0}^p w_j \times x_{ij} \right)^2 + \lambda \sum_{j=0}^p w_j^2$$

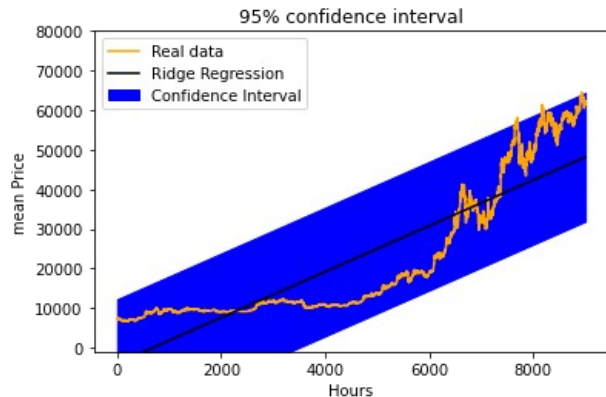
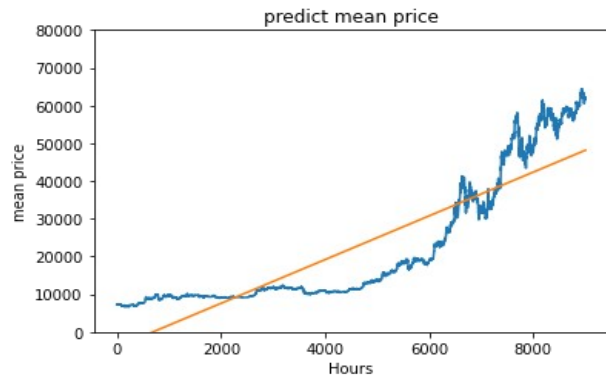
- هدف ما همواره کمینه کردن مقدار تابع هزینه است، به همین دلیل عبارت بالا در واقع معادله‌ی لاگرانژین نوشته شده برای مسئله کمینه کردن مقدار تابع هدف (در این جا، هزینه) تحت قید زیر است:

$$\text{For some } c > 0, \sum_{j=0}^p w_j^2 < c$$

- تابع هزینه بدست آمده به این معناست که باید همواره حواسمان به بزرگی مقادیر  $w$  باشد. در واقع عبارت جریمه مقادیر وزن را منظم سازی می‌کند به گونه‌ای که اگر مقادیر بردار وزن بیش از حد بزرگ باشد تابع هزینه جریمه خواهد شد. به عبارت دیگر؛ رگرسیون ریدج، مقادیر وزن را کوچک می‌کند تا از پیچیدگی مدل بکاهد. به این نوع منظم سازی ( منظم سازی با استفاده از نرم 2 ) منظم سازی نرم 2 می‌گویند.



## کد رگرسیون ریدج (Ridge)



- کد نسبت به قبل دچار دو تغییر خواهد شد. تغییر اول در استفاده از کتابخانه ی مخصوص Ridge به جای کتابخانه ی LinearRegression و تغییر دوم به هنگام فیت کردن مدل رگرسیون ریدج روی داده ها که به جای ساختن یک شی از کلاس کتابخانه Regression Linear ، باید یک شی از کلاس Ridge بسازیم.

- بخش اول :

```
from sklearn.linear_model import Ridge
```

- بخش دوم :

```
ridge = Ridge().fit(x_data_composed, y_data)
model_line = ridge.predict(x_data_composed)
```

- در نهایت خروجی ها به شکل رو به رو در خواهند آمد :



## رگرسیون لاسو (Lasso)

- مفهوم این رگرسیون همانند رگرسیون Ridge است؛ با این تفاوت که برای منظم سازی مقادیر بردار وزن از روش منظم سازی نرم-1 استفاده می کند. در واقع تابع هزینه به شکل زیر خواهد بود:

$$\sum_{i=1}^M (y_i - \hat{y}_i)^2 = \sum_{i=1}^M \left( y_i - \sum_{j=0}^p w_j \times x_{ij} \right)^2 + \lambda \sum_{j=0}^p |w_j|$$

- نکته ی مهم در مورد مزیت این روش منظم سازی نسبت به روش قبل است. در این روش ممکن است تعدادی از مقادیر بردار وزن صفر شود. به همین دلیل رگرسیون لاسو علاوه بر جلوگیری از بیش برآزش و همینطور پیچیدگی زیاد مدل، در انتخاب ویژگی هم می تواند به داد ما برسد.

# کد رگرسیون لاسو (Lasso)

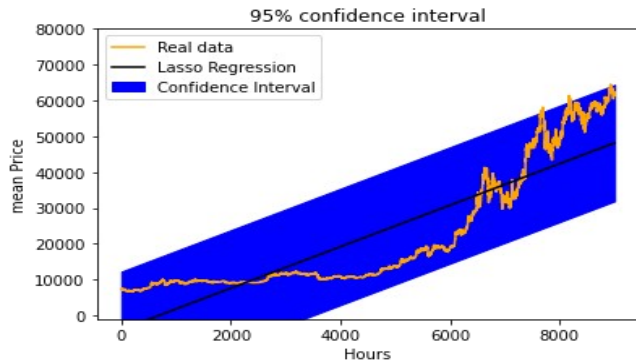
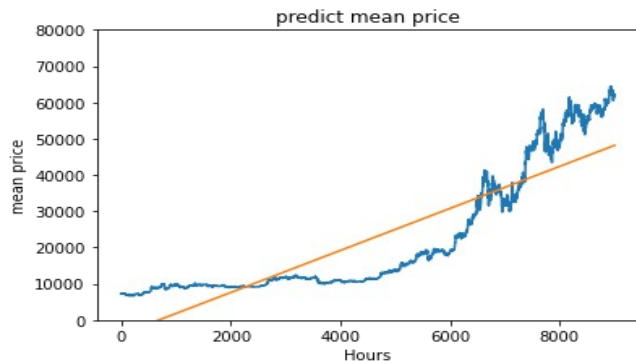
- تغییرات کد و خروجی نهایی این بخش نیز مشابه بخش قبل اما با تفاوت هایی جزئی به شکل زیر خواهد بود:

- بخش اول:

```
from sklearn.linear_model import Lasso
```

- بخش دوم:

```
lasso = Lasso().fit(x_data_composed, y_data)  
model_line = lasso.predict(x_data_composed)
```

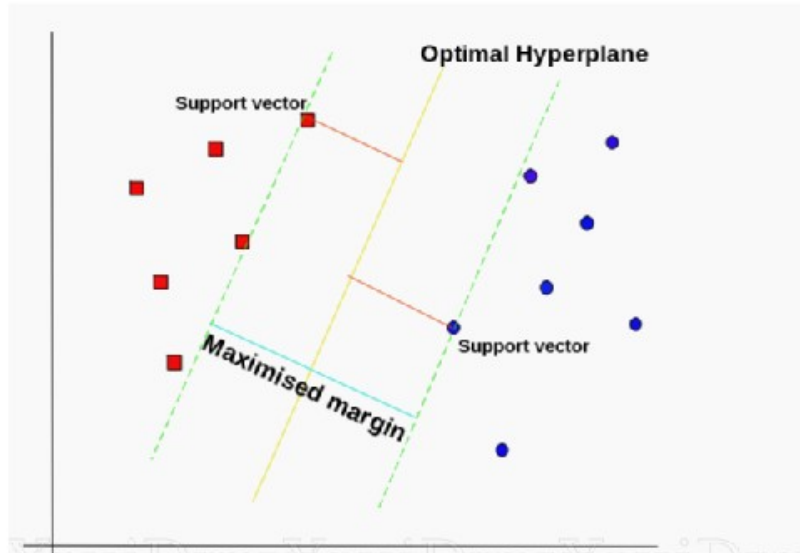


## رگرسیون بردار پشتیبان

- الگوریتم ماشین بردار پشتیبان SVM از تکنیک های معروف در حوزه یادگیری ماشین است که هم در مسائل کلاس بندی و هم در مسائل رگرسیون استفاده می شود
- مهم ترین تفاوت این روش با روش های معمول رگرسیون، این است که به جای کمینه کردن اختلاف بین مقدار واقعی و مقدار تخمین زده شده ( کمینه کردن تابع هزینه )، تلاش می کند تا بهترین نمودار ممکن را در محدوده ی بین نمودار و حدود تعیین شده بر روی داده، فیت بکند.
- در روش SVR نیز از هسته های روش SVM برای فیت کردن و نسبت دادن نمودار استفاده می شود

## هسته های روش SVR

- هسته ی خطی :

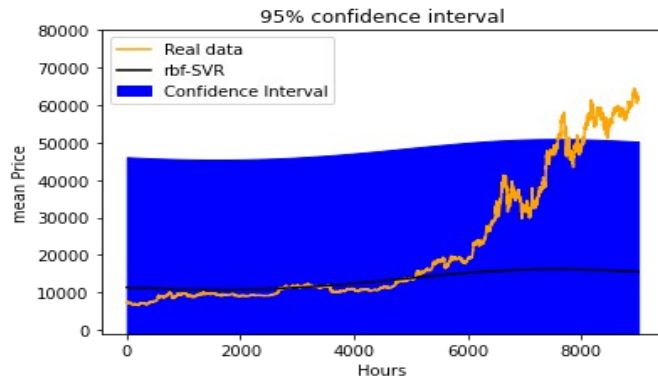
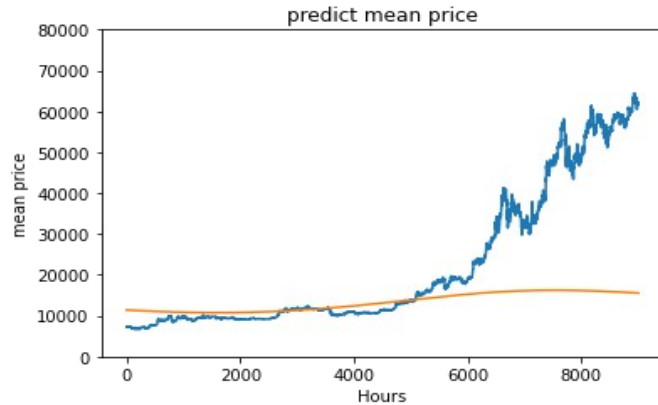


در حالت هسته ی خطی Linear این روش بسیار شبیه به رگرسیون خطی است. به همین دلیل می توان گفت که در این حالت هدف پیدا کردن بهترین خط ممکن برای ست کردن روی داده در فاصله ی خطوط محدوده یا خطوط پشتیبان است.

بردار های پشتیبان هم نقاطی از داده هستند که به خط زرد رنگ (خط فیت شده) نزدیک ترین اند. به همین خاطر خطوط پشتیبان و محدوده (خطوط سبز رنگ) هم از این نقاط می گذرند

اما نکته ی منفی این کرنل هنگام اجرا شدن در کد است که بسیار زمان بر است؛ به همین دلیل SVR بیشتر زمانی کاربرد دارد که پراکندگی داده ها به گونه ای باشد که نتوان برای تشخیص و تخمین رفتار داده ها به آنها خط نسبت داد

# هسته های روش SVR



## • هسته های RBF :

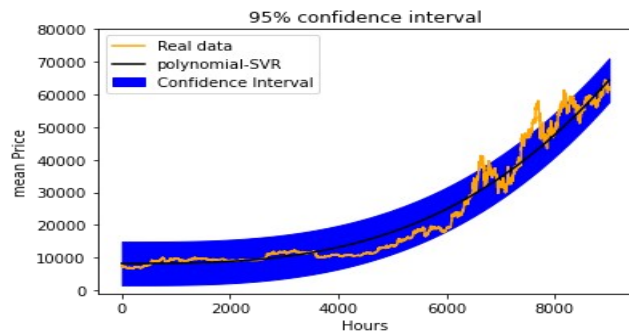
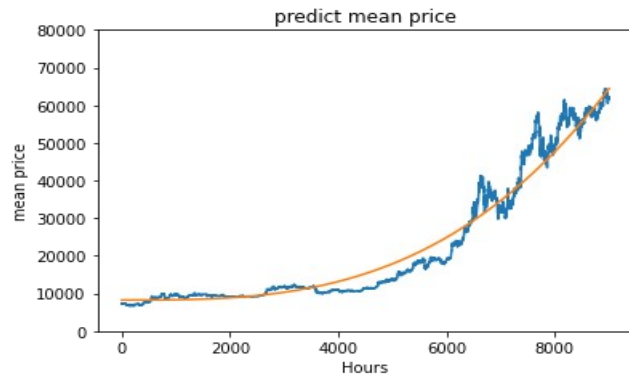
این هسته ها برای داده هایی که رگرسیون خطی مناسب آنها نیست کاربرد دارند. روش کار این هسته ها نیز به این شکل است که داده ها را به فضایی بزرگتر انتقال می دهند که در آن جا می توان رگرسیون خطی را روی داده ها اعمال کرد یا در کل به صورت خطی جداسازی کرد. تابعی که در این هسته استفاده می شود، توابع Radial Basis Function هستند.

## تغییرات کد در هسته RBF :

```
SV_regression =  
SVR(kernel='rbf').fit(x_data_composed,  
y_data)
```

```
model_line=  
SV_regression.predict(x_data_composed)
```

# هسته های روش SVR



- هسته ی Poly :

این هسته نیز همانطور که از اسمش مشخص است توابع چندجمله ای با درجه دلخواه ما را به داده ها نسبت می دهد.

- تغییرات کد polynomial :

```
SV_regression = SVR(kernel='poly', degree=3).fit(x_data_composed, y_data)
```

```
model_line = SV_regression.predict(x_data_composed)
```



### • تحلیل‌های تخصصی :

1. پس از انجام این آزمایش‌ها به این نتیجه می‌رسیم که برای تخمین قیمت میانگین بیت کوین نسبت به متغیر روز، تکنیک‌های رگرسیون خطی ساده، Ridge، lasso و همینطور روش SVR با کرنل چند جمله ای درجه 1، بسیار مناسب خواهند بود. در عوض در داده‌ی مورد نظر ما، کرنل rbf خروجی خوبی تولید نکرد.
2. با توجه به فراوانی بالای دو دسته‌ی (6000-12000) و (12000-18000) به این نکته دست می‌یابیم که روند حرکتی بیت کوین از قیمت 6000 تا 18000 به صورت صعودی اما با نوسانی ملایم بوده، چون در این دو بازه قیمت‌های نزدیک به هم را ثبت کرده است. در بازه‌های قیمتی 18000 تا 30000 روند نوسانی به مراتب بیشتری را طی کرده اما به دست آوردن الگوی این نوسان نمی‌تواند کار دشواری باشد؛ چون از قیمت 18000 تا 60000 در بازه‌های پشت سر هم 3000 تایی، دسته‌های جدول فراوانی ما دارای فراوانی‌های نزدیک به هم هستند.

- تحلیل‌های کلی در مورد بیت کوین :

1. میانگین‌های متحرک برای بررسی روندهای ساید مناسب نیستند

2. میانگین متحرک نمایی در هر بازه‌ای و در تقاطع با هر قیمت یا اندیکاتور دیگری ، در مقایسه با میانگین متحرک‌های دیگر در همان بازه و در تقاطع با همان قیمت یا اندیکاتور متقاطع با میانگین متحرک نمایی، نسبت به نوسان‌ها پاسخ‌های بیشتری می‌دهد و در نتیجه تعداد کراس اوورهای بیشتری را تولید می‌کند. در نتیجه میانگین متحرک نمایی برای معامله (حداقل روی بیت کوین) در بازه‌های زمانی کوتاه مدت ( یک دقیقه‌ای، پنج دقیقه‌ای ) مناسب است.

3. برخلاف میانگین نمایی، میانگین وزن دار - مخصوصا با پارامترهای بالا - نسبت به نوسان‌ها بسیار مقاوم است و کراس اوورها با تعداد‌های مناسب و به مراتب کمتری از حالت نمایی تولید می‌کند. پس میانگین متحرک وزن دار، برای معاملات در بازه‌های زمانی متوسط یا بلند مناسب‌تر است.