بسمه تعالی

پروژه درس اقتصاد عمومی ۱

مقایسه عملکرد دو الگوریتم در پیش‌بینی قیمت مسکن

استاد درس: دکتر قادری

دستیار استاد: محمدامین واحدی‌نیا، حامد غلامی

دانشجو: بهنام خلیلی

نیمسال اول سال تحصیلی 01-1400

فهرست مطالب

۱. چکیده

این پروژه به مقایسه عملکرد دو الگوریتم در پیش‌بینی قیمت مسکن می‌پردازد. ابتدا فرایند و تکنولوژی‌ای را معرفی و تعریف می‌کنیم که این پیش‌بینی را برایمان ممکن و البته آسان می‌سازد و سپس دو الگوریتم را پیاده سازی کرده تا کامپیوتر قادر به پیش‌بینی قیمت شود. پیاده‌سازی این پروژه به کمک زبان برنامه‌نویسی python و به وسیله‌ی کتابخانه‌های مربوطه در محیط کدزنی انجام می‌شود. در نهایت نتایج آن دو را مقایسه می‌کنیم تا متوجه شویم کدام یک در مورد قیمت خانه‌ها دقیق‌تر عمل می‌کنند. این دو الگوریتم عبارتند از XGBoost و Neural Networks. بنگاهی که به آن پرداختیم، حدود ۳۶۰۰ مسکن انتخاب شده از شهر تهران می‌باشد که لیست ویژگی‌ها و قیمت آنها را از وب‌سایت Kaggle.com گرفته‌ایم. در انتها با کمک گرفتن از شاخص‌های معتبر این دو مدل که از دو الگوریتم متفاوت به دست آمده‌اند را مقایسه می‌کنیم.

کلمات کلیدی: یادگیری ماشین، داده، الگوریتم، پیش‌بینی، قیمت مسکن، دقت

۲. مقدمه

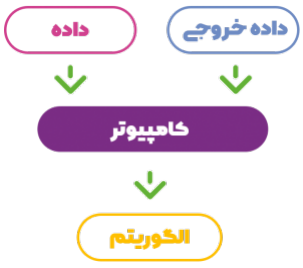
بیان مسئله

امروزه افراد زیادی نگران آینده‌ی خود می‌باشند و همواره در حال برنامه‌ریزی برای آن هستند. در خصوص امور اقتصادی یکی از موارد مطلوب جامعه، پیش‌بینی قیمت می‌باشد. شاید در گذشته این امر غیر ممکن به نظر می‌آمد ولی اکنون به کمک یادگیری ماشین می‌توان فراتر از این کار را نیز انجام داد. پیش‌بینی ممکن است برای ویژگی ثابت اما در گذر زمان باشد و یا ممکن است بخواهیم در یک زمان ثابت به پیش‌بینی یک مورد با ویژگی‌های متفاوت بپردازیم. با پیش‌بینی آینده می‌توان فهمید که بهترین فرصت برای خریدن، فروختن یا نگه داشتن دقیقا چه زمانی خواهد بود و اینگونه به سود می‌رسیم. اما اگر دقیقا در زمان حال بدانیم که الگوریتمی وجود دارد که قیمت را بر اساس هر ویژگی‌ای که موجود باشد برای ما تخمین می‌زند، می‌توانیم بهترین مورد را توسط الگوریتم انتخاب کنیم تا در کنار بهره‌مندی از ویژگی‌های دلخواهمان کمترین هزینه ممکن را پرداخت کنیم. در این پروژه به دو الگوریتم برای پیش‌بینی حالت دوم پرداخته شده و هدف این است تا بهینه‌ترین یا دقیق‌ترین الگوریتم برای این مورد مطالعه، انتخاب شود. در ادامه به تئوری‌های موجود در یادگیری ماشین می‌پردازیم و سپس پژوهش‌های گذشته را مطرح می‌کنیم.

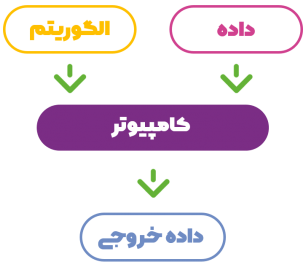
تعریف یادگیری ماشین

در کتاب‌ها تعاریف متنوعی از یادگیری ماشین وجود دارد. یکی از بهترین تعاریف را آقای آرتور سموئل در سال ۱۹۵۹ مطرح کرده است. به گفته ایشان، یادگیری ماشین حوزه‌ای است که بدون برنامه‌ریزی مستقیم، به کامپیوترها قابلیت یادگیری می‌دهد [۱]. از این تعریف میتوان دو برداشت مهم کرد. برداشت اول این است که به کمک یادگیری ماشین، کامپیوترها می‌توانند مباحث جدیدی یاد بگیرند و برداشت دوم به ما میگوید که یادگیری ماشین از برنامه‌ریزی مستقیم استفاده نمی‌کند. برای درک بهتر برنامه‌ریزی مستقیم از یک مثال استفاده می‌کنیم.

فرض کنید یک ربات در آشپزخانه وجود دارد و ما می‌خواهیم به او بگوییم که فنجان را از کابینت برای ما بیاورد. در روش برنامه‌ریزی مستقیم با استفاده از دستورات رایج برنامه‌نویسی مانند if,else,while باید ربات را ابتدا به سمت کابینت هدایت کنیم و سپس به طور دقیق به کمک همان دستورات بگوییم که چگونه درب کابینت را باز کرده و فنجان را به گونه‌ای بردارد که نشکند. اما اگر رباتمان از یادگیری ماشین استفاده می‌کرد با دیدن کابینت مورد نظر، شروع به انتخاب الگوریتم‌های از پیش تعیین شده در یادگیری ماشین می‌کرد و به سمت کابینت می‌رفت. سپس به هنگام دیدن تصویر دستیگره‌ی کابینت یکی دیگر از الگوریتم‌ها را انتخاب کرده و درب کابینت را به بهترین روش باز می‌کرد. سپس به کمک انتخاب الگوریتم دیگری که همه‌ی آنها در یادگیری ماشین از قبل وجود داشته‌اند، فنجان را بدون اینکه بشکنند برایمان می‌آورد. در روش اول ما به رباتمان الگوریتم چگونگی انجام فعالیت‌ها و داده‌های آشپزخانه شامل ابعاد را دادیم اما در روش دوم ما فقط داده‌های آشپزخانه را دادیم و فنجان را از ربات خواستیم. برنامه‌ریزی مستقیم فنجان را به ما به عنوان خروجی داد اما یادگیری ماشین با الگوریتم‌های از پیش طراحی شده‌اش رباتمان را کنترل کرد و جایگاه خروجی با الگوریتمش متفاوت بود. در تصویر شماره ۱ نحوه‌ی عملکرد یادگیری ماشین و در تصویر شماره ۲ نحوه‌ی عملکرد برنامه‌ریزی مستقیم را مشاهده می‌کنید [۲].



۱. عملکرد یادگیری ماشین



۲. عملکرد برنامه‌ریزی مستقیم

دیدیم که کامپیوتری که از یادگیری ماشـــین اســـتفاده می‌کند، تمامی ورودی‌هایش داده‌ها هســـتند و ما از الگوریتم‌هایی که یادگیری ماشـین در اختیار آن کامپیوتر می‌گذارد به عنوان خروجی آسـتفاده می‌کنیم. پس کمی بیشـتر به این داده‌ها و الگوریتم‌ها بپردازیم.

منظور از داده، هر گونه اطلاعاتی است که به یکی از صورت‌های تصویر، صوت، متن، عدد و سیگنال ثبت شده‌اند. مفهومی در مورد داده‌ها وجود دارد که به آن برچســـب (label) می‌گویند. برخی داده‌ها برچســـب دارند و برخی فاقد آن می‌باشـــند و ما باید برچســـب آنها را تشـــخیص و به آن نســـبت دهیم. مفهوم برچســـب به ما نشـــان می‌دهد که هر داده به کدام گروه تعلق دارد و این تقسیم‌بندی گروه‌ها نیز بر اساس ویژگی‌های کل داده‌ها انجام می‌شود. در بسیاری از الگوریتم‌ها، هدف اصلی برچسب زدن به تمامی داده‌های موجود و یا پیش‌بینی ویژگی داده‌های جدید بر اساس برچسبی که دارند می‌باشد [۳].

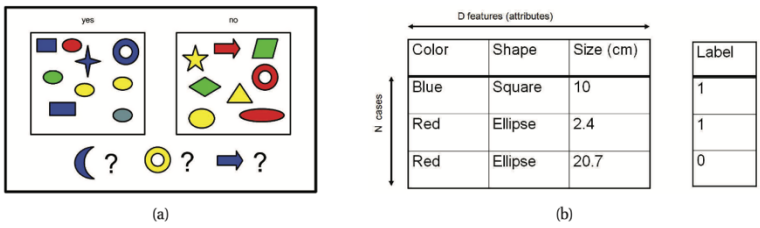
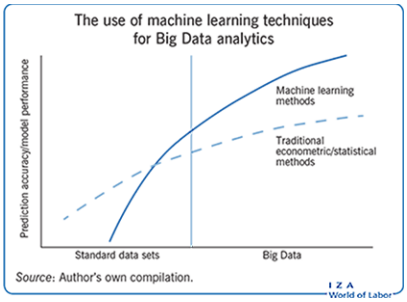


Figure 1.1 Left: Some labeled training examples of colored shapes, along with 3 unlabeled test cases. Right: Representing the training data as an $N \times D$ design matrix. Row i represents the feature vector $\mathbf{x}_i$. The last column is the label, $y_i \in \{0, 1\}$. Based on a figure by Leslie Kaelbling.

۳. نمایش مفهوم برچسب

ما در حال ورود به عصر کلان داده‌ها (big data) هستیم و در هر بخشی از زندگی می‌توان آنها را حس کرد. دقیقا مثل حجم اطلاعات زیادی که با هر بار گشتن در موتورهای جست‌وجو می‌بینیم (۱۰۰۰ میلیارد نتیجه) یا حجم ویدئوهایی که در سایت‌های مختلف بارگذاری شده‌اند. در واقع کلان داده به مجموعه‌ای از داده‌ها گفته می‌شود که بسیار بزرگتر و بیشتر از حد معمول است و دارای تکرارهای فراوان و همچنین گاهی دارای داده‌های شخصی میباشد. با توجه به کلان داده‌ها دیگر استفاده از روش‌های قدیمی برای مدیریت این مجموعه‌ها کاربردی نیست، اما استفاده از یادگیری ماشین این کار را برایمان آسانتر می‌کند. گفته شد که ورودی کامپیوتری که از یادگیری ماشـین اسـتفاده می‌کند، فقط داده می‌باشـد. طبق پژوهش‌ها میتوان دید که هر چه داده‌ها بیشـتر باشـد، کامپیوتر به شکل دقیقتری از الگوریتم‌ها بهره می‌برد [۴].



۴ . تفاوت استفاده‌ی کلان داده‌ها در یادگیری ماشین

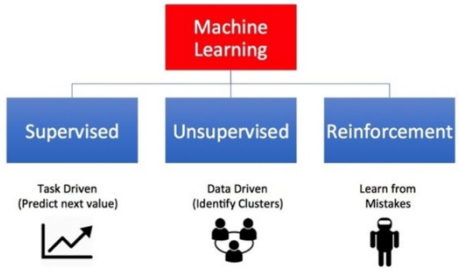
از الگوریتم‌های یادگیری ماشین میتوان به عنوان ابزار یا روشی یاد کرد که ما آنها را تعریف می‌کنیم تا روابط و الگوهای موجود در بین این داده‌ها، درک و استخراج شود و سپس از این روابط و الگوها استفاده کنیم. از اصلی‌ترین کاربردهای این الگوها و روابط

که توسط الگوریتم‌های یادگیری ماشین به دست می‌آیند، می‌توان به خودکارسازی (automation)، پیش‌بینی آینده (prediction) و تصمیم‌گیری غیر مطمئن (decision making under uncertainty) اشاره کرد که هم‌اکنون در بسیاری از صنایع و مناطق در حال پردازش و تأثیرگذاری مستقیم در روند زندگی انسان‌ها و بهبود آن می‌باشند.

انواع مختلف مسئله‌های یادگیری ماشین

هنگام بررسی مسائل یادگیری ماشین آنها را به ۳ دسته تقسیم می‌کنند تا از الگوریتم‌های مرتبط با همان نوع استفاده کنند. این ۳ دسته عبارتند از:

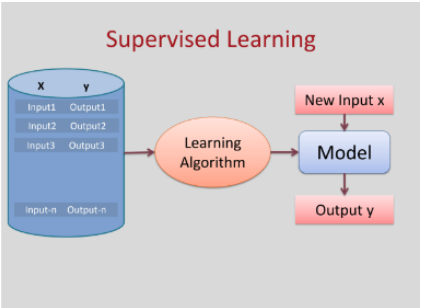
- یادگیری با نظارت (Supervised learning)
- یادگیری بدون نظارت (Unsupervised learning)
- یادگیری تقویتی (Reinforcement learning)



۵. انواع مختلف یادگیری ماشین

در یادگیری با نظارت تمامی داده‌ها برچسب دارند و جزئیات و ویژگی‌های آنها کاملا مشخص است. وظیفه‌ی یادگیری ماشین در مورد این داده‌ها این است که ابتدا به کمک الگوریتم‌های این دسته، داده‌های ورودی (x) را به داده‌های خروجی (y) ارتباط دهد (mapping) و سپس به کمک رابطه‌ای که به دست می‌آورد برای هر x جدیدی که به کامپیوتر می‌دهیم، y مرتبط به آن را کامپیوتر به ما بدهد. برای انجام این عملیات - همچنین الگوریتم‌های دو دسته‌ی دیگر - معمولا داده‌ها به دو بخش تقسیم می‌شوند: Train set و Test set. پیاده‌سازی الگوریتم با train set انجام می‌شود و روابط x و y به دست می‌آید. سپس سیستم رابطه‌ی به دست آمده را با داده‌های test set آزمایش میکند تا ببیند دقت این پیش‌بینی‌ها چه قدر است. اگر داده‌ها پیوسته یا به صورت عددی باشند از رگرسیون (regression) و اگر داده‌ها به شکل گسسته یا توصیفی باشند از کلاس‌بندی (classification) استفاده می‌شود.

در رگرسیون مجموعه‌ای از روش‌ها استفاده می‌شود تا رابطه‌ای به شکل $y = f(x)$ به دست آید (همان mapping) که بیشترین تطبیق را با نمودار داده‌های ما داشته باشد. حال به ازای هر ورودی x، ما از y پیش‌بینی خواهیم داشت. روش‌هایی که در کلاس‌بندی نیز استفاده می‌شوند دقیقا به دنبال دست یافتن به همین تابع می‌باشند، با این تفاوت که جنس y از داده‌های گسسته و توصیفی است. منظور از مدل (model) در مسائل یادگیری ماشین همان رابطه‌ای است که الگوریتم با کمک داده‌ها به دست آورده است. در واقع هدف اصلی کامپیوتر مدل کردن مسئله است که بتوان از این مدل برای پیش‌بینی استفاده کرد و با در دست داشتن x، y را تا حد مطلوبی تخمین بزنیم.



۶. عملکرد یادگیری با نظارت

در یادگیری بدون نظارت، داده‌ها هیچ‌گونه برچسبی ندارند و این وظیفه‌ی الگوریتم است که برخی از ویژگی‌های داده‌ها را به هم

مرتبط سازد و آنها را با حدس مناسبی در گروه‌های متفاوت قرار دهد و برچسب مربوطه را به آنها اضافه کند. در نوع آخر که یادگیری تقویتی است، ما کامپیوتر را که حامل الگوریتم‌ها می‌باشد در محیط قرار می‌دهیم تا شروع به فعالیت کند. سپس با تنبیه و تشویق، بعد از انجام هر فعالیت کامپیوتر یاد می‌گیرد که در محیط کدام انتخاب به خواسته‌ی ما نزدیک‌تر است. به عبارتی از اشتباهاتش می‌آموزد. پرداختن به یادگیری بدون نظارت و یادگیری تقویتی از اهداف این پروژه نیست و به آنها بیشتر پرداخته نخواهد نشد. مورد مطالعه (Case study) این پروژه تعداد محدودی از خانه‌های شهر تهران است که ویژگی‌های خانه‌ها x و قیمت خانه y می‌باشد. از آنجا که هدف ما پیش‌بینی قیمت است و همچنین داده‌های ما دارای برچسب می‌باشند، بنابراین از نوع اول روش‌های یادگیری ماشین که یادگیری با نظارت است استفاده می‌کنیم.

مرور ادبیات

در سال ۱۹۹۶ میلادی یک گروه سه نفره در مورد ارتباط بین قیمت خانه و ویژگی‌های آن تحقیق کردند و هدفشان این بود که ببینند آیا این ویژگی‌ها تأثیر بیشتری روی قیمت می‌گذارند یا مالک به طور مستقل قیمت را تعیین می‌کند. این تحقیق نشان داد که بر خلاف باور مردم آن زمان، ارتباط چشم‌گیری بین قیمت و شرایط خانه وجود دارد و مالک تأثیر نسبتا کمتری روی قیمت دارد [۵]. بین در مقاله‌اش که سال ۲۰۰۴ منتشر کرد برای اولین بار به پیش‌بینی قیمت مسکن پرداخت و هدفش این بود که نتیجه‌ی پیش‌بینی‌اش را با پیش‌بینی‌های موجود در روزنامه‌ها و اخبار مقایسه کند و این کار را به کمک یک الگوریتمی که از رگرسیون استفاده می‌کرد انجام داد. در نهایت توانست به این نظریه برسد که خروجی الگوریتمی که به کار برده است به اندازه‌ی پیش‌بینی‌های موجود در بازار قابل اعتماد است [۶]. بعد از این تحقیق، مطالعات بسیاری در این زمینه انجام شده و الگوریتم‌های متفاوتی استفاده و حتی ساخته شده‌اند تا محقق به هدف مورد نظرش در پیش‌بینی قیمت مسکن برسد.

در سال ۲۰۱۴ دو دانشجو اهل کره پروژه‌ای با داده‌های منطقه‌ی Fairfax در ایالت Virginia انجام دادند تا سه مورد از الگوریتم‌ها را با یکدیگر مقایسه کنند که دقیق‌ترین آنها ripper با ۷۴ درصد دقت و ضعیف‌ترینشان naïve bayes بود اما همین الگوریتم نیز نزدیک به ۷۰ درصد دقت پیش‌بینی داشت که باز هم از پیش‌بینی‌های شخصی در جامعه بهتر بود [۷]. این الگوریتم‌ها در زمان حال دیگر اعتبار ندارند، پس به روش‌های جدید برای مدل‌سازی در مطالعات آتی می‌پردازیم.

چانگچن و هایوو طی مقاله‌ای در سال ۲۰۱۸ در شهر Arlington ایالت Virginia به وسیله random forests یک مدل برای قیمت مسکن به دست آوردند. دیدگاه مقاله این بود که اگر از الگوریتم‌های مربوط به رگرسیون خطی استفاده کنند، ویژگی‌های غیر توصیفی و غیر خطی خانه‌ها مثل منطقه‌ای که در آن واقع شده یا آب‌وهوا در پیش‌بینی لحاظ نخواهد شد. پس به سراغ الگوریتم جنگل‌های تصادفی رفتند که از کلاس‌بندی استفاده می‌کند. در نهایت آن را با خروجی الگوریتم رگرسیون خطی به کمک دو شاخصه‌ی R-squre و RMSE مقایسه کردند (این دو شاخصه میزان خطا را با واحدی مختص به خودشان نشان می‌دهند که مورد اول هر چه بیشتر باشد و مورد دوم هر چه به صفر نزدیکتر باشد، مدل ما از دقت بیشتری برخوردار است. در بخش اعتبارسنجی در مورد آنها توضیح کاملی می‌آید). طبق انتظارشان مدل‌سازی بر اساس کلاس‌بندی جنگل‌های تصادفی از رگرسیون خطی عملکرد بهتری داشت. حتی با تغییر دادن تعداد ویژگی‌های مورد استفاده در مدلسازی نیز نتیجه تغییر نکرد [۸].

Features/ Methods	R-square	RMSE
Features: Year built, lot size		
Random forests	0.639135706989	367.054972867
Linear regression	0.344215758946	407.940568982
Features: Zip code, latitude, longitude, year built, lot size		
Random forests	0.68614045456	357.59049678
Linear regression	413.775211328	413.775211328
Features: latitude, longitude, year built, lot size		
Random forests	0.701680132695	352.892749026
Linear regression	0.407037969505	389.06342124
Features: Zip code, latitude, longitude, year built, lot size		
Random forests	0.701310346391	352.065553406
Linear regression	0.539887986037	381.280477804

۷. نمودار شاخص‌های مقایسه‌ی نتیجه‌ی دو مدلسازی انجام شده

قیمت مسکن و قدرت توان خرید مردم به شکل غیر مستقیم در وضعیت اقتصادی جامعه تأثیر می‌گذارد و به همین دلیل است که میزان مطالعات در این زمینه بسیار زیاد است و به شکل پیوسته نیز به آن اضافه می‌شود. مقاله‌ای که در مورد آن صحبت شد نزدیک‌ترین به هدف این پروژه می‌باشد. قدم اول انجام پیش‌بینی قیمت و قدم دوم مقایسه‌ی مدل‌سازی‌ها برای یافتن بهترین روش موجود در یادگیری ماشین برای انجام این کار است. در این پروژه به سراغ الگوریتم‌های رگرسیون می‌رویم و بر خلاف نتیجه‌ی مقاله‌ی بالا معتقدیم که رگرسیون به دلیل انجام محاسبات بیشتر، خروجی دقیق‌تری در اختیارمان می‌گذارد. تنها کافی است تا ویژگی‌های توصیفی را به یک سری مجموعه عددی خاص به صورت یک به یک نسبت دهیم. به عنوان مثال اگر y یکی از دو حالت بله یا خیر باشد، بله را عدد یک و خیر را عدد صفر در نظر می‌گیریم و در محاسبات استفاده می‌کنیم.

در ادامه دو الگوریتم مذکور را به کمک کتابخانه‌های معروف python، به کار می‌بریم تا بتوانیم مدل کامل و دقیقی از لیست ۳۶۰۰ تایی خانه‌های تهران [۹] به دست آوریم. سپس با مقایسه‌ی نتایج درمی‌یابیم که کدام روش عملکرد بهتری در پیش‌بینی قیمت مسکن خواهد داشت.

۳. بدنه‌ی اصلی

دو اگوریتمی که قرار است در این پژوهش با یکدیگر مقایسه شوند و هر دو می‌توانند داده‌های پیوسته و گسسته را پوشش دهند، XGBoost و شبکه‌های عصبی (Neural networks) می‌باشند. در این بخش به جزئیات داده‌هایی که در اختیار داریم می‌پردازیم و سپس به کمک این دو الگوریتم مدلسازی‌ها را انجام می‌دهیم تا در ادامه آنها را با یکدیگر مقایسه کنیم. تمامی این کارها در به کمک زبان برنامه‌نویسی python و کتابخانه‌های موجود در آن انجام شده که عبارتند از:

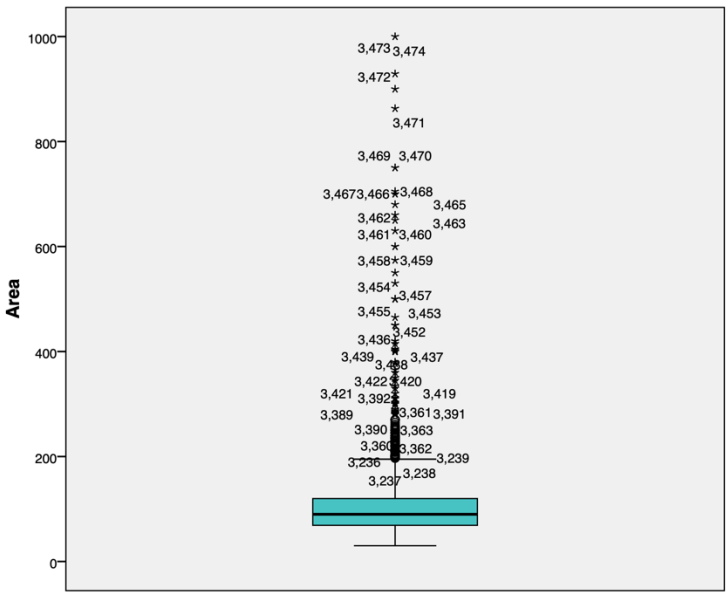
- Numpy: برای محاسبات ریاضی استفاده می‌شود.
- Pandas: برای وارد کردن آرایه‌ی داده‌ها و انجام فعالیت روی سطر و ستون‌های دیتاست استفاده می‌شود.
- Matplotlib: برای رسم نمودارها و جداول مربوط به داده‌ها و نتایج (برخی از نمدارها و جداول آورده شده در این پژوهش با استفاده از نرم‌افزار SPSS می‌باشد) مورد استفاده قرار می‌گیرد.
- Sklearn: به کمک این کتابخانه می‌توان از اکثر الگوریتم‌های یادگیری ماشین بهره برد و همچنین توابع مورد نیاز برای انجام مدلسازی و اعتبارسنجی را به سیستم اضافه کرد (بعضی از الگوریتم‌ها را باید به صورت جداگانه نصب و به برنامه اضافه کرد مانند XGBoost)

بنگاه مورد بررسی و داده‌ها

هدف این پژوهش قیمت مسکن بود بنابراین پس از جست‌وجو در اینترنت نهایتا به یک دیتاست در سایت Kaggle به آدرس https://www.kaggle.com/mokar2001/house-price-tehran-iran دست یافتیم که شامل ۳۴۷۴ مورد خانه می‌باشد. اطلاعات مربوط به این خانه‌ها در ۶ ویژگی تقسیم و نشان داده شده است و قیمت آنها نیز به ریال و دلار در آخر نشان داده شده که برای راحتی نمایش در نمودارها از قیمت دلاری آن (۱ دلار = ۳۰ هزار تومان) بهره می‌بریم.

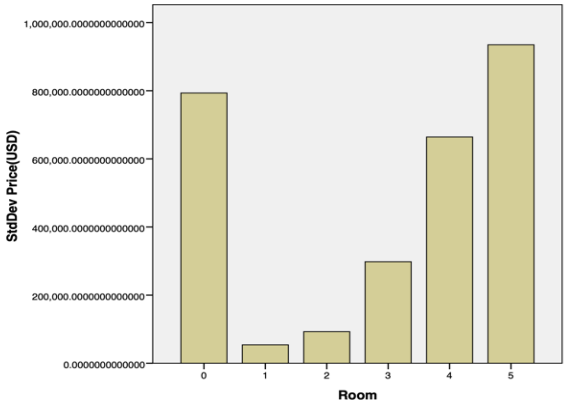
ویژگی‌های این دیتاست عبارت است از:

- مساحت (Area): با توجه به نمودار جعبه‌ای که از تقسیم‌بندی خانه‌ها به نسبت مساحت آنها به دست آمده این برداشت را می‌توان کرد که حدود ۴۶ درصد خانه‌ها مساحتی کمتر از ۱۰۰ مترمربع دارند و قریب به ۴۷ درصد آنها نیز مساحتشان بین ۱۰۰ تا ۲۰۰ مترمربع می‌باشند. ۲۳۸ خانه نیز مساحتی بیش از ۲۰۰ مترمربع دارند که توسط برنامه داده‌ی دورافتاده تلقی شده و در نمودار جعبه‌ای محاسبه نشدند.

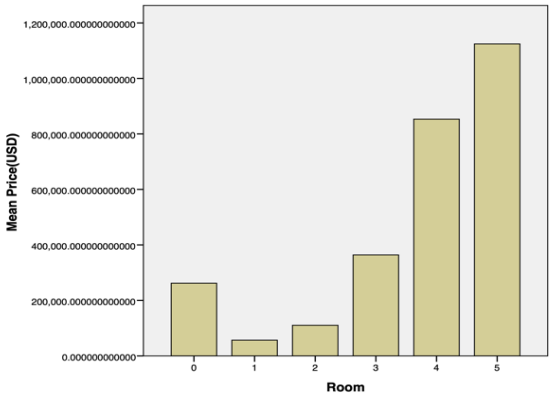


۸. نمودار جعبه‌ای Area

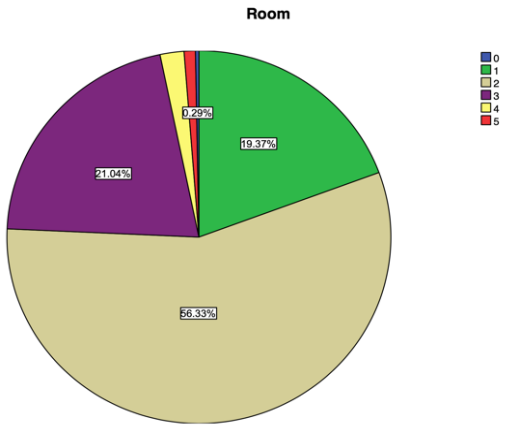
- تعداد اتاق‌ها (Room): بازه‌ی این ویژگی اعداد صحیح در [۰,۵] می‌باشد و طبق جداول زیر میتوان گفت که به جز مورد صفر این ویژگی با قیمت رابطه مستقیم دارد. علت عدم ناهماهنگی در داده‌ی صفر مقدار بسیار کمتر آن نسبت به دیگر اعداد است.



۹. نمودار انحراف معیار Room

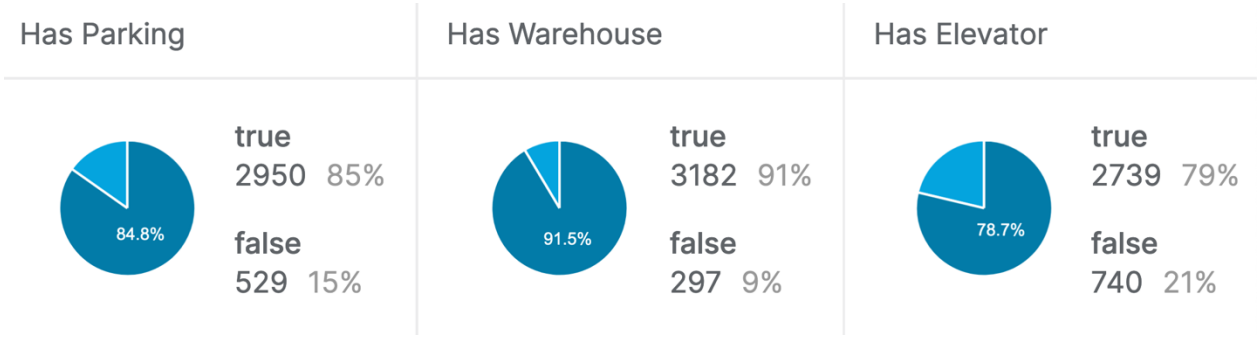


۱۰. نمودار میانگین Room



۱۱. نمودار دایره‌ای پراکندگی Room

- آسانسور (Elevator)
- پارکینگ (Parking)
- انباری (Warehouse)



۱۲. نمودارهای مربوط به نوع پراکندگی آسانسور، انباری و پارکینگ

- منطقه (Address): بیشترین تکرار منطقه مربوط به پردیس با ۱۴۶ مورد است و کمترین تکرار با فقط یک مورد می‌باشد که ۴۱ منطقه به صورت مشترک با این تعداد در دیتاست آمده‌اند.
- قیمت به دلار و تومان (Price, Price USD):

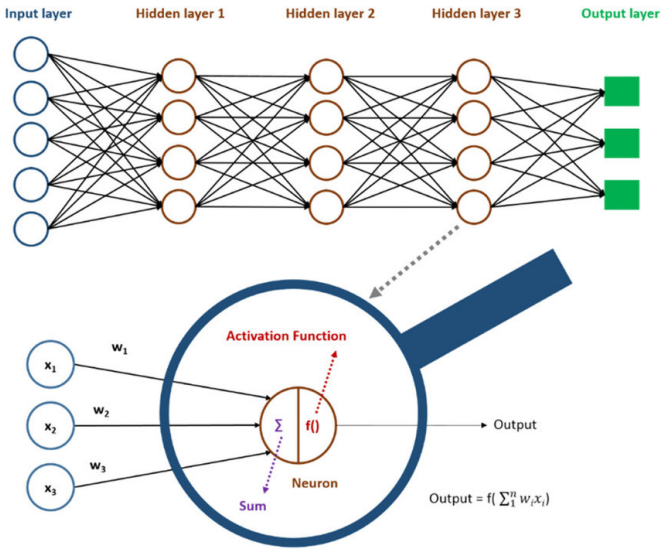
Price(USD)		
N	Valid	3474
	Missing	0
Mean		1.7850E5
Mode		6.6666E4
Std. Deviation		2.7009E5
Variance		7.295E10
Range		3.0798E6
Minimum		1.2000E2
Maximum		3.0800E6
Percentiles	25	4.7125E4
	50	9.6195E4
	75	2.0000E5

۱۳. جدول معیارهای پراکندگی قیمت خانه‌ها

مدلسازی

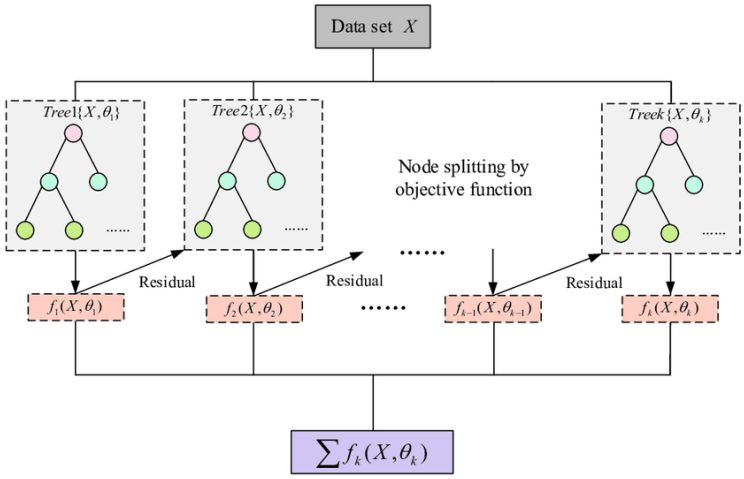
قبل از آغاز مدلسازی باید ذکر کرد که داده‌ها را به دو دسته‌ی trian_set و test_set تقسیم می‌کنیم و سپس مدل را تنها روی trian_set پیاده می‌کنیم تا الگوریم روابط مورد نظر را بیابد و سپس میزان دقت و خطا را به کمک بخش test_set به دست می‌آوریم. اینکه حجم داده‌های هر یک از این دو بخش چه مقدار باشد بر روی عملکرد مدل تأثیر می‌گذارد. بنابراین هنگام مقایسه، درصدهای متفاوتی از این تقسیم‌بندی را امتحان می‌کنیم.

ابتدا به سراغ شبکه‌های عصبی می‌رویم. این الگوریتم در یک یا چند لایه ضریبی برای هر یک از ویژگی‌ها و همچنین عدد ثابتی در نظر می‌گیرد تا به تابعی دست یابد که خروجی را پیش‌بینی کند. هر چه تعداد لایه‌ها بیشتر باشد، ضرایب نسبت داده شده بیشتر خواهد بود و تابع به دست آمده دقیق‌تر خواهد بود.



۱۴. توصیف نحوه‌ی عملکرد الگوریتم شبکه‌های عصبی

الگوریتم بعدی یعنی xgboost در بین جوامع یادگیری ماشین از محبوبیت بالاتری برخوردار است زیرا سرعت و عملکرد بهتری نسبت به الگوریتم‌های قبلی دارد. این الگوریتم تمامی داده‌های ورودی را با توجه به ویژگی‌هایی که در دیتاست وجود دارد را به یک درخت دودویی (binary trees) تقسیم می‌کند و سپس یک تابع برای برای شاخه‌ای مشخص از آن درخت تعریف می‌کند که داده‌های داخل انتهای آن شاخه را با تقریب بسیار خوبی پیش‌بینی کند. سپس تقریب‌های نامناسب موجود در درخت به نسبت تابع به دست آمده را به داخل درخت دیگری می‌برد و تابع جدیدی برای این فواصل ایجاد شده به دست می‌آورد تا فاصله‌ی پیش‌بینی از مقدار حقیقی به حداقل برسد. این روند ادامه پیدا می‌کند و درخت‌های شمارای بسیاری ایجاد می‌شود تا در نهایت با ترکیب توابع به دست آمده دیگر بتوانیم هر ورودی جدیدی را به یک خروجی مطلوب پیوست دهیم.



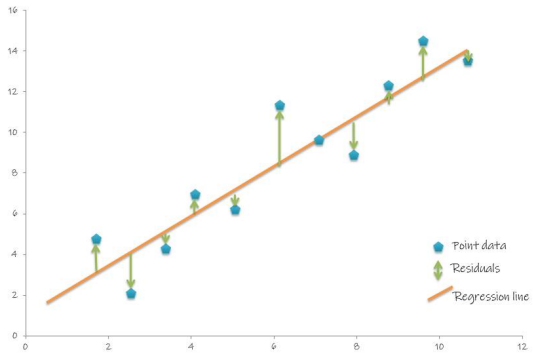
۱۵. توصیف نحوه‌ی عملکرد الگوریتم xgboost

با استفاده از کتابخانه‌هایی که در بخش قبل معرفی شد، در محیط برنامه‌نویسی python الگوریتم‌ها را بر روی دیتاستی که در اختیار داشتیم پیاده (fit) کردیم. فایل کدهایی که وظیفه‌ی این مدلسازی را به عهده داشتند در لینک زیر قرار دارد:

https://github.com/behnamkhalili/economyProject-ML

۳. بررسی میزان خطا

روابطی که به کمک مدل به دست آمده جدا از اینکه مدل خطی یا غیر خطی در ابعاد متفاوت باشند، با مقدار حقیقی معمولا تفاوت دارند. ساده‌ترین تفاوت را میتوان در مثال پایین دید:



۱۶. مفهوم خطای پیش‌بینی توسط تابع در فضای دوبعدی

برای اینکه بتوانیم بین دو مدل تشخیص دهیم که کدامیک دقیق‌تر و بهینه‌تر عمل می‌کند باید قادر باشیم این فواصل (residuals) را در کنار هم بررسی کنیم.

ابزار مقایسه

چهار شاخصه‌ی معروف برای مقایسه عملکرد مدل‌های پیش‌بینی وجود دارد:

- Mean absolute error (MAE):

$$\mathrm{MAE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} |y_i - \hat{y}_i| .$$

این شاخصه به طور مختصر به ما نشان می‌دهد که از نظر مقدار قیمتی، بازه‌ی خطای مدل ما چه قدر است.

- Mean absolute percentage error (MAPE):

$$MAPE = \frac{100\%}{n} \Sigma \left| \frac{y - \widehat{y}}{y} \right|$$

هنگامی که درباره‌ی میزان خطا یا دقت در جمعی صحبت شود، ناخداگاه مفهوم بازه‌ی صفر تا صد درصد در ذهن همه شکل می‌گیرد که به کمک این فرمول می‌توان درصد دقت خروجی‌ها را برای درک بهتر عملکرد مدل ارائه داد.

- Root mean square error (RMSE):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

این شاخص بسیار شبیه به فرمول انحراف معیار می‌باشد و دقیقا همین مفهوم را نیز در مورد خطای مدل به ما نشان می‌دهد. مقدار عددی مطلوب آن از صفر تا عددی نزدیک به میانگین قیمت کل خانه‌ها می‌باشد:

$$\mathrm{MAE} \le \mathrm{RMSE} \le n^{1/2} \cdot \mathrm{MAE}.$$

از آنجایی که هدف ما اینجا تنها مقایسه است، با توجه به فرمول می‌توان دریافت که RMSE هر چه کمتر باشد مدل ما بهینه‌تر پیش‌بینی کرده و خروجی به مقدار حقیقی نزدیک‌تر است.

- R-square:

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

Where,
$\hat{y}$ – *predicted value of y*
$\bar{y}$ – *mean value of y*

خروجی این فرمول هنگامی که مدل ما به سمت دقیق عمل کردن برود، به عدد یک میل میکند و هرچه مدل ما خروجی پرت‌تر در اختیارمان بگذارد از عدد یک دور می‌شود. بنابر این نزدیک‌ترین پاسخ به عدد یک مطلوب ما در هنگام مقایسه می‌باشد.

اعتبارسنجی دو مدل

در بخش مدلسازی گفته شد که داده‌ها به دو بخش trian_set و test_set تقسیم شده‌اند و تفاوت حجم این تقسیم‌بندی بر روی عملکرد مدل و اعتبارسنجی آن تأثیر می‌گذارد. برای رسیدن به نتیجه‌ی مطمئن این مقایسه را در درصدهای مختلف از حجم test_set انجام خواهیم داد. با اجرای کدهای نوشته شده و جمع‌آوری خروجی‌ها به اعداد جدول زیر دست یافتیم و آنها را با یکدیگر مقایسه کردیم:

	R-square	R-square	RMSE	RMSE	MAPE	MAPE	MAE	MAE	(metrics)
	XGBoost	Neural	XGB	NN	XGB	NN	XGB	NN	(models)
10%	0.194804	0.57	342098.59	248289	2.97	6.66	127603.05	112797	
15%	0.223357	0.59	318623.12	228955	2.23	3.86	123406.61	104859	
20%	0.250233	0.63	287222.92	201594	1.87	3.15	111871.35	90614	
25%	0.264731	0.48	282965.13	235841	1.64	4.53	111339.57	121232	
30%	0.282104	0.52	264185.43	215229	1.51	4.25	107687.72	106615	
35%	0.277254	0.55	261485.43	205186	1.4	3.67	106989.09	96074	
40%	0.284342	0.53	255134.21	206479	1.31	3.46	104431.96	96405	
(test_set lot)									

۱۷. جدول نهایی مقایسه دو مدل

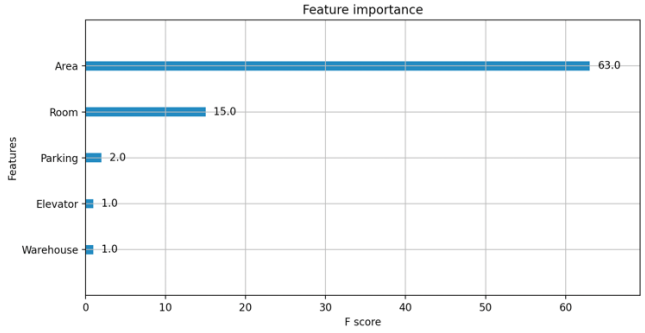
۴. نتیجه‌گیری

در این پژوهش پس از توضیحات مربوطه به موضوع، توانستیم قیمت مسکن در مناطق مختلف شهر تهران را با توجه برخی ویژگی‌هایشان در زمان حال پیش‌بینی کنیم و این کار با یادگیری ماشین تحقق یافت. هدف این بود که ببینیم بین دو الگوریتم بیان شده، خروجی کدامیک دقیق‌تر است با توجه به اینکه برخی از داده‌ها توصیفی و گسسته و برخی عددی و پیوسته بود.

با توجه به اعداد جدول ۱۷ و مقایسه‌ای که انجام شده می‌توان دریافت که مدل طراحی شده مبتنی بر شبکه‌های عصبی دقیق‌تر از XGBoost در ۳ شاخصه عمل کرده و تنها در شاخصه‌ی میانگین درصدی از مدل XGBoost ضعیف‌تر ارزیابی شده است. این برداشت را از این موضوع می‌توان کرد که تحلیل میزان خطا تنها با استفاده از درصد در برخی دیتاست‌ها مناسب نمی‌باشد و حتما باید دیگر شاخص ها را نیز بررسی کرد. همچنین بیشترین اختلاف نتایج نیز در test_size = 20% رخ داده که نشان می‌دهد بهترین تقسیم‌بندی حجمی برای این مورد مطالعاتی بوده است.

یکی دیگر از مفاهیمی که از این مدلسازی می‌توان استخراج کرد، میزان تأثیرگذاری هر یک از ویژگی‌ها بر روی خروجی نهایی است. با توجه به اینکه در جامعه‌ی ایران خانه با واحد میلیون تومان به ازای هر مترمربع خریدوفروش می‌شود، انتظار می‌رود که بیشترین تأثیر را ویژگی مساحت بر روی قیمت خروجی بگذارد. در این دو مدلسازی از ویژگی منطقه استفاده نشد. زیرا در کنار اینکه تنوع آن بسیار زیاد بوده، تعداد خانه‌های منطقه‌های چشمگیری عددی کمتر از ۱۰ می‌باشد و این موضوع به تابعی که روابط را شکل

می‌داد ضرر می‌رساند.



۱۸. نمودار میزان تأثیر هر یک از ویژگی‌ها

از نظر سرعت محاسبات، الگوریتم xgboost فوق‌العاده سریعتر نتیجه را اعلام کرد که این موضوع نشان می‌دهد با اینکه دقت کمتری داشت اما هنگامیکه حجم داده‌ها بسیار بیشتر از حد معقول باشد (کلان داده) باید به این الگوریتم بیشتر از دیگری بها داد.

۵. مراجع و ضمائم

[۱] : Some Studies in Machine Learning Using the Game of Checkers - A. L. Samuel - IBM JOURNAL' JULY ۱۹۵۹

[۲] : https://howsam.org/what-is-machine-learning/

[۳] : Machine Learning, A Probabilistic Perspective - Kevin P. Murphy

[۴] : https://wol.iza.org/articles/big-data-in-economics/long

[۵] : Adair, A., Berry, J., & McGreal, W. (۱۹۹۶). Hedonic modeling, housing submarkets and residential valuation. Journal of Property Research

[۶] : Bin, O. (۲۰۰۴). A prediction comparison of housing sales prices by parametric versus semi-parametric regressions. Journal of Housing Economics

[۷] : Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data

[۸] : A new machine learning approach to house price estimation - http://dx.doi.org/۱۰.۲۰۸۵۲/ntmsci.۲۰۱۸.۳۲۷

[۹] : https://www.kaggle.com/mokar۲۰۰۱/house-price-tehran-iran