پروژه داده کاوی

استاد راهنما: دکتر محمدرضا فقیهی حبیب آبادی

گرد آورنده: آرش سجادی ، شماره دانشجویی: ۴۰۰۴۲۲۰۹۶

دانشکده ریاضی دانشگاه شهید بهشتی

۳۰ خرداد ۱۴۰۱

چکیده:

یکی از مشکلاتی که در هنگام خرید و فروش خانه با آن مواجه هستیم، این است که مرجع خوبی برای قیمت‌گذاری خانه‌ها وجود ندارد. اغلب مشاورین املاک با توجه به در نظر گرفتن منفعت خود، اقدام به قیمت‌گذاری‌های جانب‌دارانه در پیشگاه خریدار و فروشنده می‌کنند. در این پروژه تلاش خواهم کرد یک سیستم پیشگویی منصافنه از قیمت خانه را در شهر تهران ارایه دهم. منبع اصلی من در کل این پروژه کتاب داده‌کاوی برای تحلیل خودکار کسب‌وکار: مفاهیم، فنون و کاربردهای R [۱] است. به‌علاوه منبع من برای تولید واژه‌نامه، پس از واژه‌نامه منبع اصلی مذکور، واژه‌نامه رسمی انجمن ریاضی ایران [۲] و انجمن آمار ایران [۳] است.

کلمات کلیدی: داده‌کاوی، پیشگویی قیمت مسکن

فهرست مطالب

مقدمه

داده[1]های من ثبت[2]های مشخصات ۳۴۷۹ خانه در شهر تهران است. این داده‌ها از وبسایت کگل [۴] گرفته شده است. شخصی که داده‌ها را بارگذاری کرده است مدعی شده است که این داده‌ها تحت نظر دانشگاه تربیت مدرس با تکیه بر داده‌های وبسایت‌های معاملات مسکن گرفته شده است. تاریخ ثبت داده‌ها همگی مربوط به آذر ماه سال ۱۴۰۰ می‌باشند. هر ثبت در این داده‌ها شامل ۸ ویژگی[3] است و هدف نهایی پروژه، پیشگویی قیمت مسکن در شهر تهران است.

۱ مقدمات و معرفی داده‌ها

هر ثبت در داده‌های این پروژه مشخص‌کننده ویژگی‌های یک خانه در شهر تهران است که برای فروش در بنگاه‌های معاملاتی آنلاین مکتوب گردیده. در ادامه به ویژگی‌های توصیفگر این ثبت‌ها اشاره خواهم کرد. اما قبل از هر چیز بایستی مشخص شود چرا این پروژه ارزش این را دارد که برای آن وقت صرف کنیم.

۱-۱ هدف داده کاوی

به گزارش باشگاه خبرنگاران جوان [۵] وجود مشاورین املاک در کشور نه تنها موجب کنترل قیمت‌ها نشده است بلکه علیرغم ادعای رئیس اتحادیه مشاوران املاک کشور، در موارد بسیاری به جای تنظیم بازار ملک به ساخت یک حباب مثبت یا منفی دامن زده است. از این رو وجود یک سیستم بی‌طرف برای قیمت‌گذاری منصفانه مسکن می‌تواند به این آشفتگی بازار مسکن سروسامان ببخشد و در صورتی که کار جدی‌تر با داده‌ی بیشتر و دقیق‌تر روی این قبیل پروژه‌ها صورت بگیرد، می‌توان انتظار داشت این التهاب در بازار مسکن تا حد خوبی کنترل شود. از این رو من تصمیم گرفتم به عنوان پروژه داده‌کاوی روی این موضوع تمرکز کنم.

۱-۲ تعریف هر متغیر

برای درک بهتر داده‌ها اولین قدم تعریف دقیق هر متغیر است. در این بخش هر ۸ ویژگی موجود در ماتریس داده‌ها تشریح خواهم کرد. به علاوه متغیرها از نظر پاسخ یا پیشگو[4] بودن نیز مشخص خواهند شد.

۱. **مساحت**[5]: متغیر اول داده‌ها، مشخص‌کننده مساحت داده‌ها به وسیله یک عدد صحیح[6] است. بدیهی است که مساحت، در حالت کلی یک عدد حقیقی[7] است. (البته بسیاری از فیزیک‌دانان معتقدند که فضا و زمان اساساً گسسته‌اند. پس شاید بهتر باشد که بگویم مساحت عددی گویاست.) در اینجا مساحت به نزدیک‌ترین عدد صحیح گرد شده است.

۲. **تعداد اتاق**: متغیر دوم، عددی صحیح است که تعداد اتاق‌های هر ثبت را مشخص می‌کند.

۳. **پارکینگ**: یک متغیر دودویی[8] نمایانگر تعلق داشتن و یا نداشتن پارکینگ به واحد مسکونی مورد نظر است. البته از نظر من بهتر بود تعداد پارکینگ‌های متعلق به یک واحد مسکونی با عددی صحیح مشخص می‌شد. ولیکن گردآونده داده‌ها به صورتی که گزارش کرده‌ام اقدام به گردآوری نمود است.

۴. **انباری**: یک متغیر دودویی نمایانگر تعلق داشتن و یا نداشتن انباری به واحد مسکونی مورد نظر است. البته مانند ویژگی قبلی، از نظر من بهتر بود مساحت انباری متعلق به یک واحد مسکونی توسط یک عدد صحیح یا حقیقی مشخص می‌شد. اما مجدداً گردآونده داده‌ها به صورتی که گزارش کرده‌ام، داده‌ها را گردآوری نموده‌اند.

۵. **آسانسور**: متغیری دودویی نمایانگر مجهز بودن و یا نبودن ساختمان شامل واحد مسکونی مورد نظر، به آسانسور است.

۶. **آدرس**: در هر ثبت آدرس واحد مسکونی توسط یک رشته[9] از حروف مشخص شده است. بدیهی است که این رشته‌ها توسط یک متغیر رسته‌ای[10] قابل بیان خواهند بود.

۷. **قیمت به تومان**: متغیر هفتم، یک عدد صحیح نمایانگر قیمت در نظر گرفته شده برای ملک مسکونی است. البته بهتر بود برای چنین پروژه‌ای یک کارشناس این قیمت‌گذاری را انجام می‌داد. چرا که این قیمت‌ها نمایانگر نظر مالکان واحدهای مسکونی است.

۸. **قیمت به دلار**: این متغیر به جهت ثبات بیشتر این ارز نسبت به قیمت خانه‌ها، صرفاً برای مقایسه قیمت واحدهای

[1] Data
[2] Record
[3] Feature
[4] Predictor

[5] Area
[6] Integer
[7] Real Number
[8] Binary
[9] String
[10] Categorical

```
 3rd Qu.: 120
 Max.   :3600
 NA's   :4
> summary(housePrice["Room"])
      Room
 Min.   :0.00
 1st Qu.:2.00
 Median :2.00
 Mean   :2.08
 3rd Qu.:2.00
 Max.   :5.00

> summary(housePrice["Parking"])
   Parking
 Min.   :0.0000
 1st Qu.:1.0000
 Median :1.0000
 Mean   :0.8479
 3rd Qu.:1.0000
 Max.   :1.0000

> summary(housePrice["Warehouse"])
  Warehouse
 Min.   :0.0000
 1st Qu.:1.0000
 Median :1.0000
 Mean   :0.9146
 3rd Qu.:1.0000
 Max.   :1.0000

> summary(housePrice["Elevator"])
   Elevator
 Min.   :0.0000
 1st Qu.:1.0000
 Median :1.0000
 Mean   :0.7873
 3rd Qu.:1.0000
 Max.   :1.0000

> summary(housePrice["Address"])
  Address
 Length:3479
 Class :character
 Mode  :character

> summary(housePrice["Price"])
    Price
 Min.   :3.600e+06
 1st Qu.:1.418e+09
 Median :2.900e+09
 Mean   :5.359e+09
 3rd Qu.:6.000e+09
 Max.   :9.240e+10
```

مسکونی در بازه‌های زمانی مختلف تهیه شده است. در واقع این ستون از ماتریس[11] داده‌ها اطلاعاتی به داده‌های ما اضافه نمی‌کنند چرا که نرخ دلار آمریکا، به مبلغ ثابت ۳۰ هزار تومان در زمان جمع‌آوری داده‌ها در نظر گرفته شده است. در نتیجه همبستگی[12] ۱ میان این ستون از ماتریس داده‌ها و ویژگی قبلی که قیمت واحد مسکونی به تومان بود، وجود دارد.

در این پروژه، ستون قیمت به دلار حذف خواهد شد همچنین ستون قیمت به تومان نقش متغیر پاسخ[13] را بازی خواهد کرد. (از این به بعد این متغیر را به‌اختصار قیمت می‌نامم) همچنین بقیه متغیرها پیشگو هستند.

۱-۳ نمونه یک ثبت

در این بخش می‌خواهم ثبت اول داده‌ها را به صورت عینی بررسی کنم. در جدل شماره ۱ می‌توانید سه ثبت اول داده‌ها را مشاهده کنید. در این بخش به توصیف اولین ثبت خواهم پرداخت.

Price(USD)	Price	Address	Elevator	Warehouse	Parking	Room	Area
$ ۶۱,۶۶۶/۶۷	۱,۸۵۰,۰۰۰,۰۰۰ T	Shahran	TRUE	TRUE	TRUE	۱	۶۳
$ ۶۱,۶۶۶/۶۷	۱,۸۵۰,۰۰۰,۰۰۰ T	Shahran	TRUE	TRUE	TRUE	۱	۶۰
$ ۱۸,۳۳۳,۳۳	۵۵۰,۰۰۰,۰۰۰ T	Pardis	TRUE	TRUE	TRUE	۲	۷۹

جدول ۱: سه ثبت اول داده‌ها

توصیف ثبت اول ما به این صورت است که یک خانه ۱ خواب با مساحت ۶۳ مترمربع، دارای پارکینگ، انباری و آسانسور در منطقه شهران به مبلغ ۱,۸۵۰,۰۰۰,۰۰۰ تومان معادل ۶۱,۶۶۶/۶۷ دلار آمریکا (در آذر ۱۴۰۰ با حساب هر دلار معادل ۳۰ هزار تومان) در یک سامانه آنلاین آگهی املاک برای فروش عرضه شده است.

۱-۴ خلاصه وضعیت ویژگی ها

در هنگام شروع کار با داده‌ها خوب است که خلاصه[14]ای از شاخص‌های مرکزی مهم آنها بدانیم. من کدها به همراه نتایج حاصل شده از آنها را که با مترجم[15] زبان برنامه‌نویسی R اجرا شده است را در ادامه قرار خواهم داد.

```
> summary(housePrice["Area"])
      Area
 Min.   :  30
 1st Qu.:  69
 Median :  90
 Mean   : 108
```

[11] Matrix
[12] Correlation
[13] Response Variable
[14] Summary
[15] Compiler

```
housePrice.df <- read.csv("E:/DS/term2/faghihi/housePrice.csv")
frequency_of_add=housePrice.df %>% count_('Address')
tmp=frequency_of_add[order(-frequency_of_add$n),] #sort frequency_of_add by "freq" column
other_samples=(tmp)[,1][70:193] #We selected all addresses with a frequency of less than 9.

for (i in 1:length(other_samples)){
    for(j in 1:dim(housePrice.df["Address"])[1]){
        if(housePrice.df[j,"Address"]==other_samples[i]) {
            housePrice.df[j,"Address"]="Other"
}}}

rm(frequency_of_add,tmp)
frequency_of_add=housePrice.df %>% count_('Address')
tmp=frequency_of_add[order(-frequency_of_add$n),]

for (i in 1:dim(tmp)[1]){
    for(j in 1:dim(housePrice.df["Address"])[1]){
        if(housePrice.df[j,"Address"]==tmp[i,1]) {
            housePrice.df[j,"Address"]=i
}}}
as.factor(housePrice.df$Address)
```

با اجرای کد فوق در محیط Rstudio آنچه من به دنبال آن هستم محقق می‌شود. ضمناً همان‌طور که در گزارش‌های پیشین دیدیم ۲۳ ثبت من فاقد آدرس بودند. با اجرای کد تا اینجا این ۲۳ ثبت در یک رسته قرار می‌گیرند. برچسب این دسته در ویژگی آدرس، "۳۸" می‌باشد.

۱-۵-۲ مقادیر گم‌شده

در حال حاضر، فقط ۴ ثبت ما شامل متغیر تهی[19] است و مقدار گم‌شده[20] محسوب می‌شود. مساحت این ۴ ثبت‌ها به درستی درج نشده است و با توجه به تعداد کمی که در مقابل کل داده‌ها دارند من تصمیم گرفتم که از آنها چشم‌پوشی کنم و یا به عبارتی، آنها را از ماتریس داده‌ها حذف کنم.

[19] Null
[20] Missing Value

```

> summary(housePrice["Price.USD."])
   Price.USD.
 Min.   :    120
 1st Qu.:  47275
 Median :  96667
 Mean   : 178634
 3rd Qu.: 200000
 Max.   :3080000
```

تابع describe در زبان برنامه‌نویسی R نیز می‌تواند اطلاعات خوبی را از داده‌ها در اختیار ما بگذارد. از این رو در مورد هر ویژگی، بخشی از گزارش خروجی این داده‌ها تحت تابع مذکور را در جدول ۲ قرار داده‌ام.

Variable	n	missing	distinct	Info	Sum	Mean	Gmd
Area	3475	4	239	1		108	89.60
Room	3479	0	6	804.0		08.2	7626.0
Parking	3479	0	2	387.0	2950	8479.0	2579.0
Warehouse	3479	0	2	234.0	3182	9146.0	1562.0
Elevator	3479	0	2	502.0	2739	7873.0	335.0
Address	3456	23	192				
Price	3479	0	934	1		359e+09.5	122e+09.6
Price.USD.	3479	0	932	1		178634	204055

جدول ۲: گزیده‌ای از خروجی تابع describe

۱-۵ پیش پردازش داده‌ها

پیش‌پردازش[16] داده به مراحلی گفته می‌شود که در آن داده‌ها برای داده‌کاوی آماده می‌شود. لازم به ذکر است که این مراحل جزء مهم‌ترین گام‌ها در داده‌کاوی هستند. در این بخش بایستی ماتریس داده‌ها را نهایی کنیم تا در بدو شروع فصل بعد، تصویری سازی[17] روی ماتریس نهایی داده‌ها صورت بگیرد.

۱-۵-۱ ویژگی آدرس

یکی از چالش‌هایی که در این پروژه پیش روی من است این است که آدرس‌ها به صورت رشته نوشته‌شده‌اند. آدرس برخی از ثبت‌ها فراوانی[18] کمی دارند و ممکن است برای ما مشکل‌ساز باشد. از این رو آدرس‌هایی که کمتر از ۱۰ فراوانی داشته باشند را در دسته سایر قرار می‌دهم. سپس به ترتیب فروانی (از فراوانی زیاد به کم) یک رشته عددی در چهارچوب یک متغیر رسته‌ای به هر آدرس در هر ثبت نسبت می‌دهیم. این‌گونه می‌توانم جزئیاتی که ممکن است در هنگام کار کردن با رشته‌های حروفی با آن مواجه شوم را از قلم بیندازم.

```
library(plyr)
```

[16] Preprocessing
[17] Visualization
[18] Frequency

```
> X=na.omit(housePrice.df)
> dim(X)
[1] 3475    8
```

۱-۵-۳ داده‌های دور افتاده

داده دورافتاده[۲۱] (داده پرت هم گفته می شود) داده هایی است که نسبت به سایر مشاهدات تفاوت قابل ملاحظه ای داشته باشد. [۶]

در داده‌های من یک داده وجود دارد که با وجود مساحت زیاد، قیمت نسبتاً پایینی دارد. این داده، در سطر ۲۱۶۹ ماتریس داده‌های من وجود دارد. این داده شرح دهنده یک خانه ۳۶۰۰ متری ۲ خواب، بدون پارکینگ انباری و آسانسور در محله شهریار [۲۲] به مبلغ ۹ میلیارد و ۷۲۰ میلیون تومان است. به عبارتی هر متر این خانه ۲ میلیون و ۷۰۰ هزار تومان قیمت‌گذاری شده است. همه این اطلاعات به ما نشان می‌دهد این ثبت مربوط به یک باغ است در شهریار که احتمالاً کاربری باغ دارد. اما پروژه ما مربوط به خانه‌های مسکونی در تهران است. لذا حذف این داده تحت عنوان داده دورافتاده خیلی غیر قابل انتظار نیست.

```
> X[2169,]
     Area Room Parking Warehouse Elevator
2172 3600    2       0         0        0

 Address    Price         Price.USD.
   58       9.72e+09      324000
```

با اجرای دستور پیش رو داده دورافتاده مذکور در ادامه حذف خواهد شد.

```
> X=X[-c(2169),]
```

۱-۵-۴ افراز داده‌ها

در این بخش می‌خواهم داده‌ها را به سه بخش مجموعه آموزشی[۲۳]، مجموعه اعتبار سنجی[۲۴] و مجموعه آزمون[۲۵] افراز[۲۶] کنیم. میدانیم که بیشترین سهم مربوط به مجموعه آموزشی است لذا ۶۰ درصد داده‌ها را به صورت تصادفی[۲۷] برای این مجموعه انتخاب می‌کنیم. ۳۰ درصد داده‌ها برای مجموعه اعتبار سنجی و ۱۰ درصد برای مجموعه آزمون به تصادف انتخاب می‌شود. اجرای کد پیشرو این فرایند را برای ما اجرا می‌کند.

[21] Outlier

[۲۲] متغیر غیر عددی ۵۸ برای مشخص کردن خانه‌هایی که ویژگی آدرس آنها "شهریار" است انتخاب شده است.

[23] Training Set

[24] Validation Set

[25] Test Set

[26] Partition

[27] Random

```
set.seed(1)
train.rows<-sample(rownames(X),dim(X)[1]*
    0.6)
valid.rows<-sample(setdiff(rownames(X),train
    .rows),dim(X)[1]*0.3)
test.rows<-setdiff(rownames(X),union(train.
    rows,valid.rows))
training_set<-X[train.rows,]
validation_set<-X[valid.rows,]
test_set<-X[test.rows,]
```

در انتها برای گزارش از ماتریس نهایی داده‌ها گزارش زیر را مشاهده بفرمایید.

```
> dim(training_set)
[1] 2080    7
> dim(validation_set)
[1] 1040    7
> dim(test_set)
[1] 354   7
```

۱-۵-۵ نرمالیده کردن متغیرها

در اینجا از روش نرمالیده[۲۸] کردن داده ها با استفاده از Standard Scaler خواهیم پرداخت [۷] با اجرای کد پیش رو ویژگی‌های مساحت، تعداد اتاق و قیمت نرمالیده خواهند شد. ضمناً لازم به توضیح است که متغیرهای پارکینگ، انباری و آسانسور دودویی هستند و نیازی به نرمالیده شدن ندارند. همچنین متغیر آدرس یک متغیر رسته‌ای است و قابل نرمالیده شدن نیست.

این مرحله حتماً بایستی بعد از افراز داده‌ها صورت بگیرد که اطلاعات مجموعه تست از طریق نرمالیده شدن به داخل مجموعه آموزشی اصطلاحاً نشت نکند و به تعبیر علمی، نشت داده[۲۹] رخ ندهد.

```
X_standard_scaler.train<-training_set
X_standard_scaler.valid<-validation_set
X_standard_scaler.test<-test_set
#- - - - - - - - - - -
for(i in c(1,2,7)){
    X_standard_scaler.train[,i]=scale(
    training_set[i])
}
#- - - - - - - - - - -
for(j in c(1,2,7)){
    for (i in 1:dim(validation_set)[1]){
        X_standard_scaler.valid[i,j]=(
    validation_set[i,j]-mean(training_set[,j
    ]))/sd(training_set[,j])
```

[28] Normalized

[29] Data Leakage

```
    }
}
#- - - - - - - - - - -
for(j in c(1,2,7)){
    for (i in 1:dim(test_set)[1]){
        X_standard_scaler.test[i,j]=(test_
    set[i,j]-mean(training_set[,j]))/sd(
    training_set[,j])
    }
}
```

۱-۵-۶ کدگذاری متغیرهای رسته‌ای

با توجه به اینکه تعداد رسته‌های متغیر رسته‌ای من در ویژگی رسته‌ای آدرس عدد ۷۰ است، استفاده از روش‌های مبتنی بر Dummy encoding و One hot encoding منجر به افزایش چشمگیر تعداد ویژگی‌ها می‌شوند. لذا بهتر است در این مورد از روش مرسوم Target encoding بر اساس میانه‌ی متغیر پاسخ استفاده کنم. [۸] این مرحله نیز مانند بخش قبل حتماً بایستگی پس از بخش افراز داده‌ها انجام شود. دلیل این امر هم مانند آنچه در بخش قبل تشریح کرده‌ام است.

```
target_encod <- build_target_encoding( X_
    standard_scaler.train, cols_to_encode =
    "Address",target_col = "Price",
    functions = c("median"))
X_standard_scaler.train=target_encode( X_
    standard_scaler.train, target_encoding =
     target_encod)[,c(2,3,4,5,6,8,7)]
X_standard_scaler.valid=target_encode( X_
    standard_scaler.valid, target_encoding =
     target_encod)[,c(2,3,4,5,6,8,7)]
X_standard_scaler.test=target_encode( X_
    standard_scaler.test, target_encoding =
    target_encod)[,c(2,3,4,5,6,8,7)]
```

با توجه به اینکه در فصل بعدی ممکن است مقادیر عددی منسوب به متغیرهای رسته‌ای من تغییر کند فعلاً در این بخش از واردکردن این مقادیر در ماتریس اصلی داده‌ها خودداری می‌کنم. در ادامه بعد از نهایی شدن این مقادیر با داده‌های اصلی آنها را جایگزین خواهم کرد.

ضمناً دلیل استفاده من از شاخص مرکزی میانه، شکل ۱۳ است. این شکل به من نشان می‌دهد که میانگین پارامتر خوبی برای کدگذاری این متغیرهای رسته‌ای نخواهد بود. چراکه نمودارهای جعبه‌ای در اکثر موارد نامتقارن هستند. لذا میانگین معیار مناسبی نمی‌تواند باشد.

۲ تصویری سازی و اکتشاف داده‌ها

در فصل تصویری سازی احتیاج داریم که فرایند تصویری سازی روی کل داده‌های ما صورت بگیرد. از این رو در این فصل بیشتر با شیءهای X و X_scaled [۳۰] سر و کار خواهیم داشت.

۲-۱ بافت‌نگار ویژگی‌ها

در این بخش ویژگی‌هایی که قابلیت دارند از آنها بافت‌نگار[۳۱] رسم کنیم را مورد تحلیل و بررسی قرار می‌دهم. این فرایند به من کمک می‌کند تا در مورد توزیع هر یک از ویژگی‌ها اطلاعات خوبی را به دست آورم.

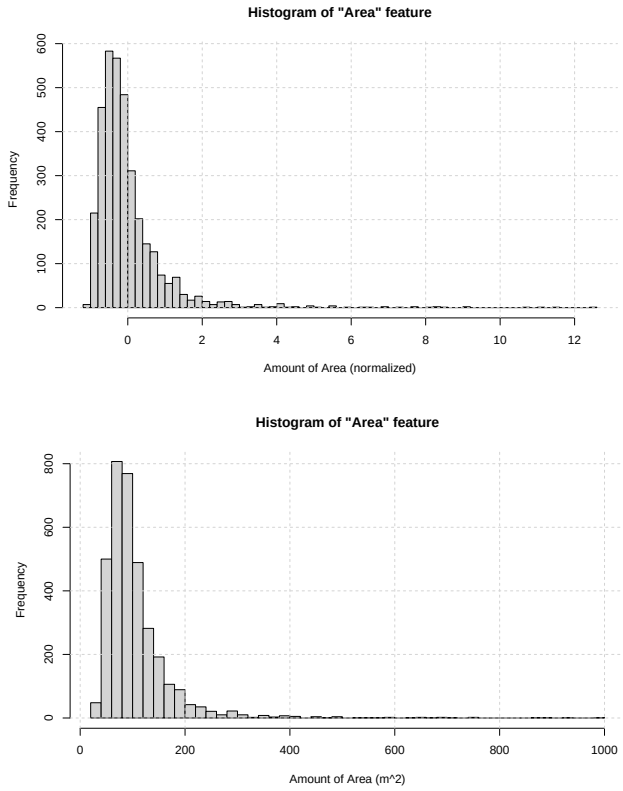
از آنجایی که بسیاری از ویژگی‌های ما مثل ویژگی مساحت نرمالیده شده‌اند، ممکن است تجسم خوبی از مقیاس آن‌ها نداشته باشیم. از این رو من در چنین مواردی مانند شکل ۱، دو بافت‌نگار ترسیم می‌کنم که به صورت عینی‌تر توزیع ویژگی مربوطه را به تصویر بکشم.

در خصوص بازه‌های بافت‌نگار ویژگی‌ها، حالت‌های مختلفی برای تعیین بازه‌های دسته‌بندی وجود دارد. هرچه طول بازه‌ها بیشتر باشد، نویز ناشی از نمونه‌گیری تصادفی را کم‌تر به نمایش می‌گذارد. از طرف دیگر هرچه طول بازه‌ها کمتر باشد، تخمین بهتری از توزیع می‌توان پیدا کرد. [۹]

من در این پروژه تلاش کرده‌ام که طول بازه‌های دسته‌بندی را برای هر ویژگی به گونه‌ای تعیین شوند که بهترین ارتباط بصری را با مخاطب برقرار کنند. البته طول و تعداد بازه‌های انتخاب شده لزوماً بهترین نیستند و قضاوت این موضوع به عهده استاد محترم درس خواهد بود.

شکل ۱: همانطور که در گزارش اولیه فصل اول دیدیم، میانگین [۳۲] ویژگی مساحت ۱۰۸ متر و میانه آن ۹۰ متر بوده است. از آنجا که در بافت‌نگارها میانه[۳۳] به شکل مشهودتری دیده می‌شود، باید بگویم که کاملاً واضح است که می‌توان خانه‌های حدود ۹۰ متر را خانه‌ای با مساحت عادی در نظر گرفت. نکته دوم اینکه در نمودار نرمالیده شده، "صفر" همان میانگین داده‌های نرمالیده است. کاملاً تأثیر داده‌های غیرعادی بزرگ (و نه داده‌های دورافتاده) را در فاصله دادن میانگین از میانه می‌توانیم مشاهده کنیم. ضمناً با اجرای یک کد کوتاه که در ادامه خواهم آورد متوجه خواهیم شد که بیش از ۵.۶۴ درصد خانه‌هایی که روی آنها مطالعه می‌کنم مساحتی بین ۶۰ تا ۱۲۵ مترمربع دارند. این حقیقت نیز از روی نمودار بافت‌نگار قابل مشاهده و صحه سنجی است.

```
> j=0
```



شکل ۱: بافت‌نگار ویژگی مساحت

```
> for (i in 1:3474){
+     if(X[i,1]<=125 & X[i,1]>=60){j=j+1}
+ }
> print(100*j/3474)
[1] 64.50777
```

شکل ۲: همان‌طور که از شکل ۲ پیداست، به نظر می‌رسد در مورد تعداد اتاق‌ها با یک توزیع[۳۴] تقریباً شبیه به توزیع نرمال [۳۵] سروکار داریم. بیشترین فراوانی مربوط به خانه‌های دو خواب و کمترین فراوانی برای خانه‌های بدون اتاق‌خواب است. ضمناً خانه‌های سه خواب فراوانی بیشتری نسبت به خانه‌های یک خواب دارند.

در ادامه بافت‌نگار ویژگی‌های پارکینگ، انباری و آسانسور را خواهید دید که چون متغیرهای دودویی داشتند از نرمالیده کردن آنها صرف‌نظر کردم. از این رو نمودار بافت‌نگار آنها را به صورت خام ارائه خواهم کرد.

شکل ۳: بیش از ۸۴ درصد خانه‌هایی که مورد مطالعه قرار دادم دارای دست‌کم یک پارکینگ هستند. این حقیقت را می‌توانستیم از خروجی تابع summary در فصل اول نیز دریابیم.

توجه داشته باشید آمارهایی که ارائه می‌کنم فقط مربوط به

[۳۰] متغیرهای عددی غیر دودویی X در X_scaled نرمالیده شده‌اند. این ماتریس از داده‌ها صرفاً برای تصویری سازی ایجاد شده است.

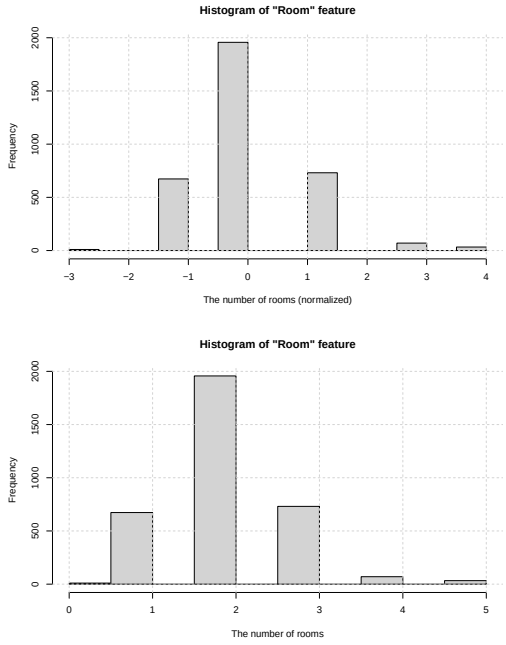
[31] Histogram

[32] Mean

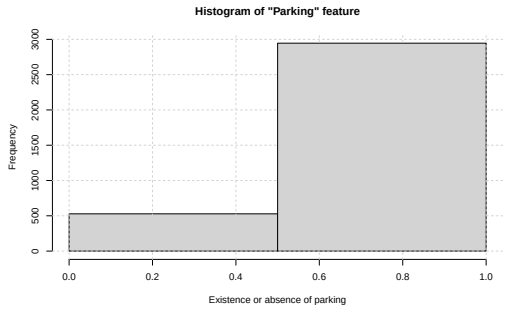
[33] Median

[34] Distribution

[35] Normal Distribution



شکل ۲: بافت‌نگار ویژگی تعداد اتاق

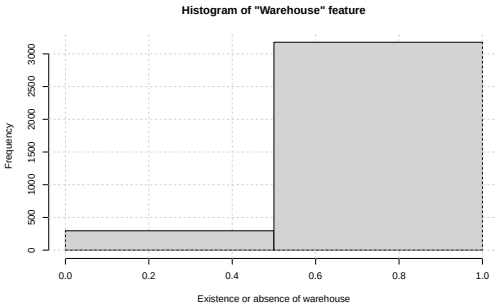


شکل ۳: بافت‌نگار ویژگی پارکینگ

داده‌های این پروژه است. ممکن است برداشت‌های ما از داده‌های این پروژه با آنچه در سطح استان تهران می‌بینیم متفاوت باشد. به‌عنوان مثال بسیاری از خانه‌هایی که قصد فروش ندارند، به‌هیچ‌وجه در این مجموعه داده‌ها نمی‌توانستند قرار بگیرند. پس نمی‌توان ادعا کرد که مثلاً بیش از ۸۴ درصد خانه‌های استان تهران دارای پارکینگ هستند.

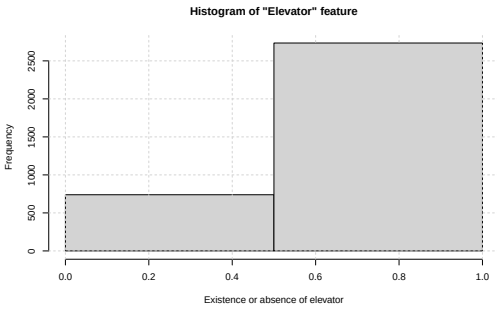
شکل ۴: در مورد ویژگی انباری نیز باید بگوییم مشابه ویژگی پارکینگ، یک نامتعادل[۳۶] بودنی را در داده‌ها مشاهده می‌کنم. ضمناً بیش از ۹۱ درصد خانه‌هایی که ما آنها را در این پروژه بررسی می‌کنیم دارای انباری هستند. (در مورد مساحت انباری هیچ اطاعاتی در دست نیست.)

[36] Imbalance



شکل ۴: بافت‌نگار ویژگی انباری

یکی از انتقادات من به این مجموعه داده‌ها این است که ویژگی‌ها به صورت مدبرانه انتخاب نشده‌اند. مثلاً در مورد ویژگی پارکینگ شاید بهتر بود تعداد پارکینگ به جای وجود یا عدم وجود آن ثبت می‌شد. همچنین شاید سال ساخت واحد مسکونی می‌توانست یک ویژگی بسیار مؤثر باشد. به‌هرحال من روی این داده‌ها کار می‌کنم و نمی‌توانم تغییری در آن ایجاد کنم ولیکن هر چند بسیار به داده‌ها انتقاد دارم امیدوارم با این ویژگی‌ها بتوانم نتیجه نسبتاً خوبی بگیرم.

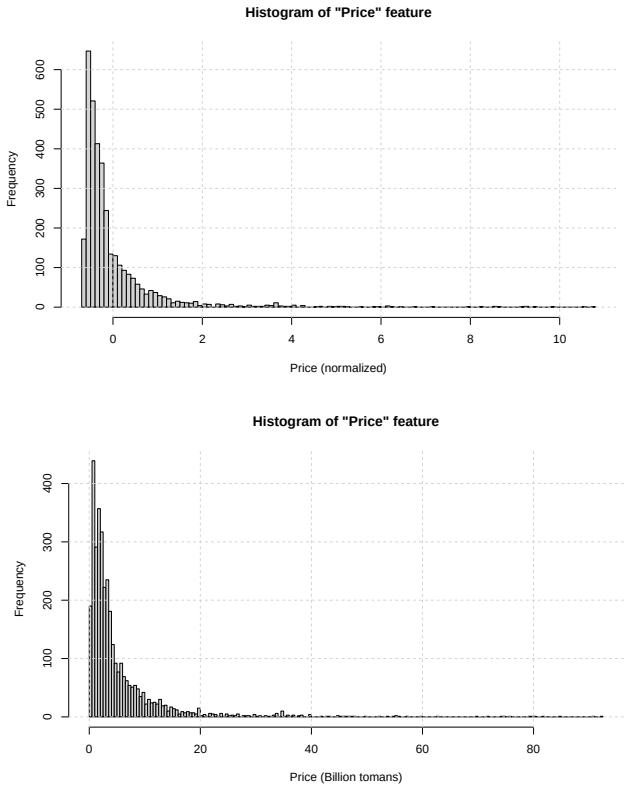


شکل ۵: بافت‌نگار ویژگی آسانسور

شکل ۵: در شکل ۵ آخرین بافت‌نگارِ دودویی را در مورد ویژگی آسانسور می‌بینیم که نسبت به دو ویژگی دودویی دیگر کمتر نامتعادل است. لازم به ذکر است که کمتر از ۸۰ درصد ثبت‌های ما دارای آسانسور هستند.

شکل ۶: این نمودار حاوی اطلاعات متغیر پاسخ ماست. چیزی که توجه من را جلب کرد این بود که شباهت نسبی‌ای بین این نمودار و شکل ۱ وجود دارد که می‌تواند بیانگر یک همبستگی میان متغیر پیشگوی مساحت و پاسخ باشد. در کل می‌توانیم به‌وضوح ببینیم هر چه قیمت خانه‌ها بیشتر باشند، فراوانی آنها نیز کمتر است.

به‌علاوه در شکل ۶ دو نمودار مشاهده می‌کنیم. نمودار اول، بافت‌نگار داده‌های نرمالیده است و نمودار دوم بافت‌نگار قیمت



شکل ۶: بافت‌نگار متغیر پاسخ

خانه‌ها برحسب میلیارد تومان است.

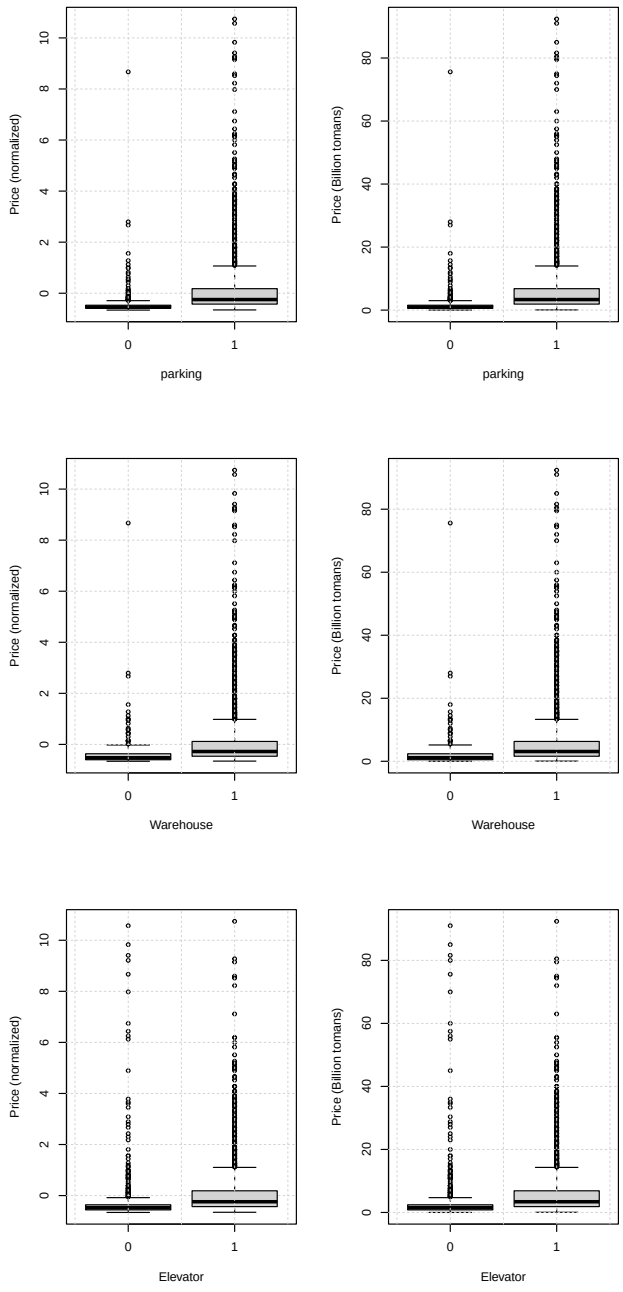
۲-۲ نمودار جعبه‌ای

در این بخش تلاش خواهم کرد از نمودار جعبه‌ای[۳۷] و نمودار جعبه‌ای پهلوبه‌پهلو[۳۸] برای نشان دادن توزیع ویژگی‌ها نسبت به هم خصوصاً نسبت به متغیر پاسخ، استفاده کنم.

شکل ۷: در ابتدا در شکل ۷ برای متغیر هایی که بیانگر داشتن یا نداشتن امکاناتی در منزل مسکونی است از نمودارهای جعبه‌ای پهلوبه‌پهلو استفاده می‌کنم. این کار به من کمک خواهد کرد تا ببینم تا چه حد به طور کلی وجود یا عدم وجود این امکانات از قبیل پارکینگ، انباری و آسانسور روی قیمت یک خانه مسکونی در شهر تهران (بر اساس داده‌های در اختیار ما) می‌تواند اثربخش باشد.

از آن‌جایی که داده‌ها نرمالیده شده‌اند، ممکن است نمودار داده‌های نرمالیده نتواند مقدار متغیر قیمت را به مخاطب انتقال دهد، لذا تصمیم گرفتم یک ستون به متغیر قیمت برحسب میلیارد تومان اختصاص دهم.

همان‌طور که از شکل ۷ مشخص است قیمت خانه‌هایی که دارای پارکینگ هستند به شکل مشهودی بالاتر است. این موضوع

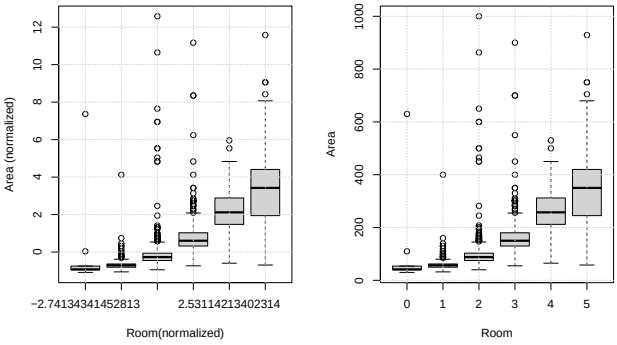


شکل ۷: نمودار جعبه‌ای پهلوبه‌پهلو متغیرهای دودویی - قیمت

در مورد انباری و آسانسور نیز صدق می‌کند. چیزی که توجه من را جلب کرد این بود که بر خلاف نمودارهای جعبه‌ای مربوط به انباری و پارکینگ، داده‌هایی که نرم‌افزار آنها را دور افتاده تشخیص داده است در نموداری که مربوط به ویژگی آسانسور است به شکل متوازن‌تری بین خانه‌های با آسانسور و بی آسانسور توزیع شده است. حدس من این است که این داده‌های دور افتاده می‌توانند خانه‌های ویلایی گران قیمتی باشند که نیاز به آسانسور ندارند. اما فارغ از این داده‌هایی که تعداد کمی دارند، عمده داده‌ها به ما می‌گویند داشتن هر سه این امکانات می‌توانند تأثیر افزایشی روی

[37] Boxplot
[38] Side-By-Side Boxplot

قیمت خانه‌ها داشته باشند.



شکل ۸: نمودار جعبه‌ای پهلوبه‌پهلو متغیر تعداد اتاق – مساحت

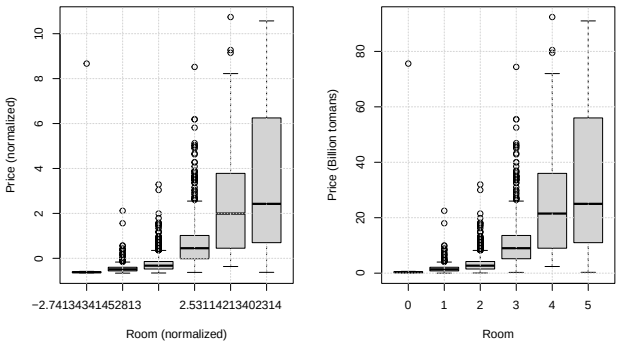
شکل ۸: به صورت کلی به من این نتیجه را می‌دهد که تعداد اتاق، با مساحت خانه‌ها رابطه مستقیم دارد. این حقیقت نشان دهنده همبستگی مثبت میان متغیر پیشگو تعداد اتاق و متغیر پیشگو مساحت است. (مقدار دقیق همبستگی در اجرای کد پیش رو قابل مشاهده است) خالی از لطف نیست که اشاره کنم، از آنجایی که حدس می‌زنم به احتمال بسیار قوی متغیر مساحت، با متغیر پاسخ، همبستگی قابل توجهی دارد؛ هر چه همبستگی میان متغیر مساحت و متغیر تعداد اتاق کمتر از ۱ باشد و در عین حال متغیر تعداد اتاق با متغیر پاسخ همبستگی نزدیک‌تری به ۱ داشته باشند، این دو ویژگی در پیشگویی کمک بیشتری به من خواهند کرد.

```
>cor(X[,1] , X[,2])
            Room
Area 0.6570648
```

همان‌طور که در شکل ۲، یا بافت نگار ویژگی تعداد اتاق قابل ملاحظه است، بیشترین فراوانی مربوط به خانه‌های دوخواب است. از این رو طبیعی است که داده‌های دور افتاده بیشتری را در شکل ۸ برای خانه‌های دوخواب و پس از آن برای خانه‌های سه‌خواب شاهد باشیم. (تصور من این است که بسیاری از این داده‌های دورافتاده مربوط به خانه‌های ویلایی است)

شکل ۹: اولین نکته‌ای که این شکل به من نشان می‌دهد این است که سه شاخص آماری، شامل میانه، چارک[۳۹] سوم و صدک[۴۰] صدم برای متغیر پاسخ یا همان قیمت خانه با افزایش تعداد اتاق خواب‌ها افزایش می‌یابد. نکته دیگری من متوجه آن هستم این است که داده‌های دورافتاده در خانه‌های دو و سه‌خواب بیشترین سهم را به خود اختصاص داده‌اند. یعنی خانه‌هایی که به نسبت

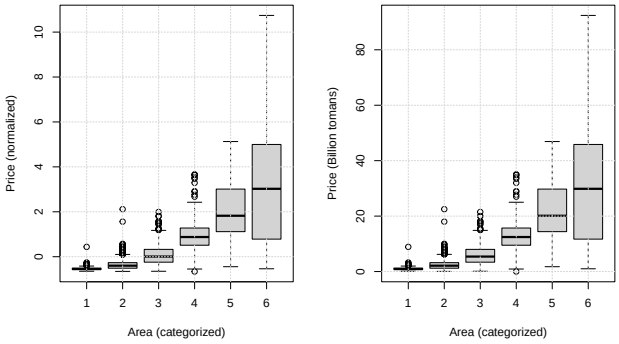
[39] Quartile
[40] Percentile



شکل ۹: نمودار جعبه‌ای پهلوبه‌پهلو متغیر تعداد اتاق – قیمت

تعداد اتاق‌هایی که در خود جای داده‌اند، خیلی گران هستند، غالباً دو و سه‌خواب هستند.

یک خانه حدوداً ۸۰ میلیارد تومانی داریم که هیچ اتاق خوابی ندارد. این ثبت را بررسی کردم و متوجه شدم این ثبت مربوط به یک ملک ۶۳۰ متری با قیمت ۷۵ میلیارد و ۶۰۰ میلیون تومان، واقع در منطقه تجریش (در دسته بندی ما در رسته ۱ یا همان سایر قرار دارد) این‌طور که مشخص است، احتمالاً این ثبت مربوط به یک قطعه زمین بایر در منطقه تجریش است. در مورد حذف آن در صورت لزوم در ادامه تصمیم خواهم گرفت.



شکل ۱۰: نمودار جعبه‌ای پهلوبه‌پهلو متغیر رسته‌ای شده مساحت – قیمت

شکل ۱۰: از آنجایی که میان شکل‌های ۸ و ۹ جای خالی نمودار جعبه‌ای پهلوبه‌پهلو مساحت – قیمت حس می‌شد و از طرفی مساحت متغیری پیوسته بود برای نمایش بهتر، در شکل ۱۰ مساحت خانه‌ها را به ۶ دسته مطابق جدول ۳ طبقه‌بندی کردم.

همان‌طور که پیش‌تر حدس زده بودم رابطه مستقیمی میان متغیر پاسخ و متغیر پیشگوی مساحت وجود دارد. یعنی به طور کلی خانه‌های بزرگ‌تر گران‌تر هستند. البته این حدس خیلی واضحی است اما برداشت ما از تحلیل علمی داده‌ها سندیت دارد، لذا تا

Area ≤ ۵۰	۵۰ < Area ≤ ۱۰۰	۱۰۰ < Area ≤ ۱۵۰	۱۵۰ < Area ≤ ۲۰۰	۲۰۰ < Area ≤ ۲۵۰	۲۰۰ < Area
۱	۲	۳	۴	۵	۶

جدول ۳: طبقه‌بندی ویژگی مساحت برای رسم شکل ۱۰

زمانی که داده‌ها به ما اطلاعاتی را ندهند هیچ سوگیری نخواهم داشت.

شکل ۱۱: نمودار جعبه‌ای قیمت هر مترمربع خانه در شهر تهران

شکل ۱۱: شکل ۱۰ این ایده را به من داد که پراکندگی[41] قیمت هر متر خانه را در تهران بررسی کنم. همچنین برای درک بهتر از جدول ۴ برای تفسیر بهتر شکل ۱۱ نیز بهره بردم. برداشت من این است که قیمت هر متر خانه‌های مورد بررسی ما به طور متوسط قیمتی بین ۳۰ تا ۴۰ میلیون تومان دارند. اما خانه‌هایی که قیمت آنها بسیار بالاتر از این اعداد هستند نیز با فراوانی کمتر نیز وجود دارند. از این رو می‌توان پیش‌بینی کرد که احتمالاً بافت نگار متناظر با شکل ۱۱، مانند شکل‌های ۱ و ۶ دارای چولگی مثبت[42] خواهد بود.

Min.	۱st Qu.	Median	Mean	۳rd Qu.	Max.
۰٫۰۲۲۵	۲۰٫۰۰	۳۴٫۶۱۵۴	۴۱٫۱۶۰۷	۵۴٫۸۸۶۱	۴۱۶٫۶۶۶۷

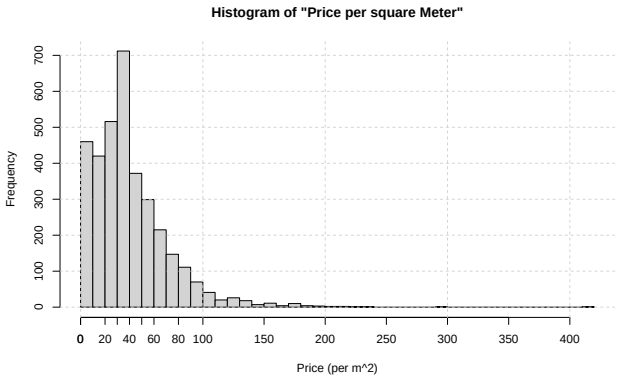
جدول ۴: جدول شاخص‌های آماری مرتبط با شکل ۱۱ بر حسب میلون تومان

تحلیل ما از جدول ۴ و شکل ۱۱ به ما نشان می‌داد که به طور متوسط خانه‌ها بین ۳۰ تا ۴۰ میلیون تومان (به ازای هر متر) قیمت دارند. اما بررسی بافت نگار توزیع قیمت هر متر خانه در تهران اطلاعات دیگری نیز به من می‌دهد. در نتیجه بر آن شدم تا برای مقایسه استثنائا در این قسمت مربوط به نمودارهای جعبه‌ای، یک بافت نگار را نیز بررسی کنم

من در این بافت نگار تعداد میله‌ها را حدود ۴۰ عدد تنظیم

[41] Dispersion

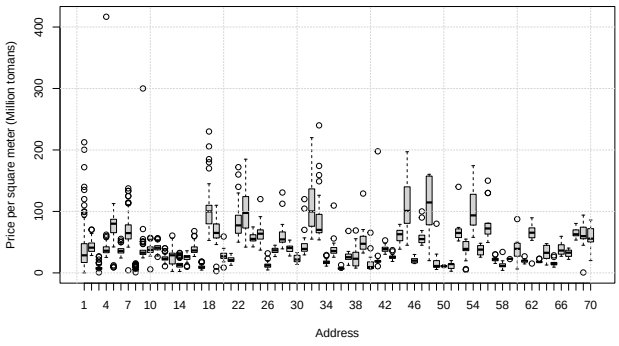
[42] Positive Skewness



شکل ۱۲: بافت‌نگار توزیع قیمت هر متر خانه در شهر تهران

کردم. اگر این عدد روی ۶۰ تنظیم می‌شد متوجه می‌شدیم که تقریباً تعداد خانه‌های ۱۰ تا ۲۰ میلیون تومان (به ازای واحد مساحت) با خانه‌های ۳۰ تا ۳۵ میلیون تومانی (به ازای واحد مساحت) برابر است. اما تعداد خانه‌های ۰ تا ۱۰ میلیون تومانی بسیار کمتر از خانه‌های ۳۵ تا ۴۰ میلیون تومانی است. لذا قضاوت اینکه قیمت یک متر خانه در تهران چقدر است کار سخت‌تری از اعلام یک عدد یا یک بازه قیمتی است.

آخرین نموداری که در بخش نمودارهای پهلوبه‌پهلو می‌خواهم بررسی کنم متشکل از ۷۰ نمودار جعبه‌ای است. این نمودار به قیمت هر متر خانه در مناطق مختلف تهران می‌پردازد.



شکل ۱۳: نمودار جعبه‌ای پهلوبه‌پهلو متغیر رسته‌ای آدرس- قیمت هر مترمربع خانه

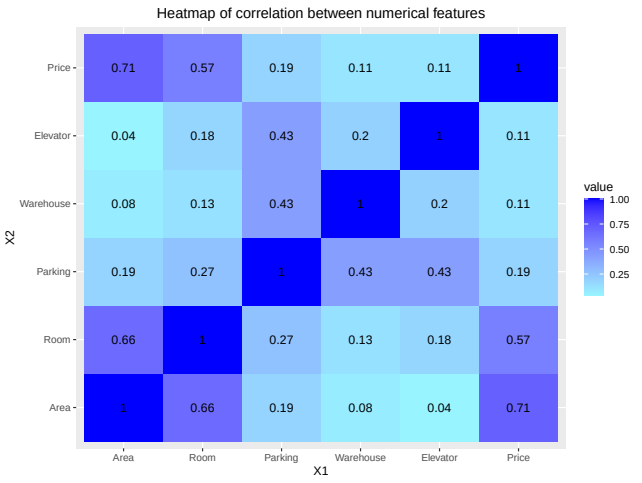
شکل ۱۳: همان‌طور که قبلاً هم اشاره کرده‌ام متغیر پیشگو رسته‌ای آدرس، با توجه به فراوانی آدرس‌ها نام‌گذاری شدند. رسته شماره ۱ نشان دهنده رسته سایر و باقی رسته‌ها بر اساس فراوانی هر یک متناظر با عددی از ۲ تا ۷۰ هستند. این نمودار به من نشان می‌دهد که اولاً بازه قیمتی در مناطقی که گران قیمت‌تر هستند بزرگ‌تر است. یعنی در مناطق گران قیمت‌تر، خانه‌ها قیمت دقیقی ندارند. لذا این نکته کلیدی را به من نشان می‌دهد که ضرورت

وجود یک سیستم پیش‌گویانه قیمت در این مناطق بیشتر از مناطق ارزان قیمت‌تر است. چرا که آمار به من یک آشفتگی در قیمت گذاری را نشان می‌دهد. ثانیاً بیشترین و کمترین تعداد خانه‌های برای فروش گذاشته شده، در مناطقی هستند که قیمت هر متر نزدیک به میانگین و میانه است. مناطق خیلی گران‌تر بر خلاف ادعای بنگاه‌های معاملاتی دارای بازار داغ‌تر خرید و فروش نیستند.

ضمناً با توجه به بررسی‌های من در خود داده‌ها منطقه ۵ داغ‌ترین بازار خرید و فروش املاک را دارد. پیش‌تر یک وب‌سایت خرید و فروش مسکن نیز این ادعا را مطرح کرده بود و دلایل خود را برای این امر ذکر کرده بود. [۱۰]

۳-۲ نمودار حرارتی

نمودار حرارتی[۴۳] یک نوع تصویرسازی داده‌ها[۴۴] است که در آن همبستگی بین ویژگی‌ها با یک رنگ نمایش داده‌می‌شود. ویژگی‌هایی که دارای همبستگی بالایی با یکدیگرهستند با رنگ‌های تیره‌تری نسبت به ویژگی‌هایی که دارای همبستگی کمتری نسبت به هم هستند به نمایش گذاشته می‌شوند. به طور معمول برای نمایش بصری اعداد یک بازه رنگی از روشن تا تیره (از کم به زیاد) در نظر گرفته می‌شود. البته این نمودارها همواره توسط یک ماتریس عددی قابل‌بیان هستند. اما نمودار حرارتی از نظر بصری قابلیت درک بیشتری دارد. [۱۱]



شکل ۱۴: نمودار حرارتی جدول همبستگی ویژگی های عددی

شکل ۱۴: این نمودار به ما نشان می‌دهد انباری و آسانسور به طور کلی تأثیر کمی روی قیمت خانه‌ها داشته‌اند. اما داشتن آن مزیت محسوب می‌شود. (این شهود را در بازار نیز شاهد هستیم.) اهمیت پارکینگ حدود دو برابر انباری و آسا سنسور گزارش‌شده است اما همچنان مهم‌ترین پارامتر نیستند. دو پارامتر مساحت و تعداد اتاق بیشترین همبستگی را با قیمت خانه دارند و البته

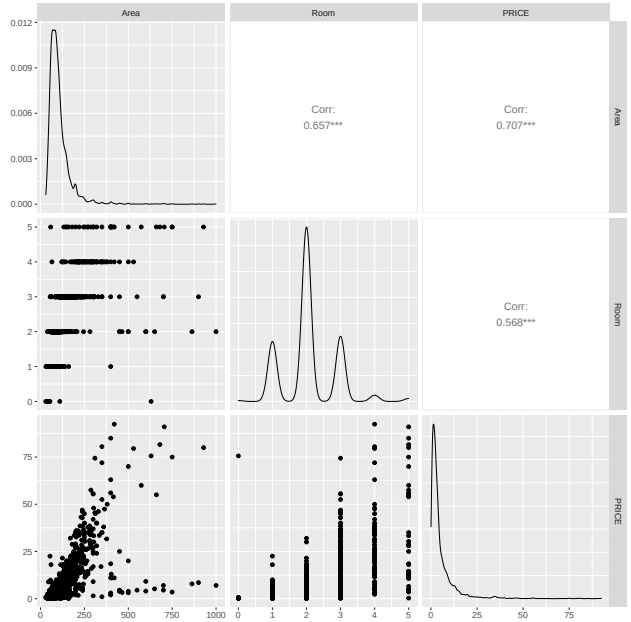
[43] Heatmap
[44] Data Visualization

ویژگی‌ای که در این تصویر قابل مشاهده نیست (چون یک مغیر رسته‌ای است) ویژگی آدرس هست. همان‌طور که در شکل ۱۳ قابل مشاهده است یک ویژگی بسیار اثرگذار روی قیمت خانه‌ها آدرس است که در ادامه گزارش به اهمیت آنها اشاره خواهم کرد.

	Area	Room	Parking	Warehouse	Elevator	Price
Area	۱	۰٫۶۵۷۱	۰٫۱۹۴۷	۰٫۰۷۶۲	۰٫۰۴۴۱۸	۰٫۷۰۶۷
Room	۰٫۶۵۷۱	۱	۰٫۲۷۴۹	۰٫۱۲۹۹	۰٫۱۸۱۷	۰٫۵۶۷۶
Parking	۰٫۱۹۴۷	۰٫۲۷۴۹	۱	۰٫۴۳۰۸	۰٫۴۳۰۴	۰٫۱۹۰۵
Warehouse	۰٫۰۷۶۲	۰٫۱۲۹۹	۰٫۴۳۰۸	۱	۰٫۲۰۱۶	۰٫۱۱۰۰
Elevator	۰٫۰۴۴۲	۰٫۱۸۱۷	۰٫۴۳۰۴	۰٫۲۰۱۶	۱	۰٫۱۱۱۷
Price	۰٫۷۰۶۷	۰٫۵۶۷۶	۰٫۱۹۰۵	۰٫۱۱۰۰	۰٫۱۱۱۷	۱

جدول ۵: جدول همبستگی داده‌ها

برای پیدا کردن دید بهتر روی سه ستون مهم در ماتریس داده‌ها شامل مساحت، تعداد اتاق و قیمت (برحسب میلیارد تومان) شکل پیش رو را که حاوی ماتریس نمودار پراکنش است را نیز ضمیمه نمودارهای قبل کرده‌ام.



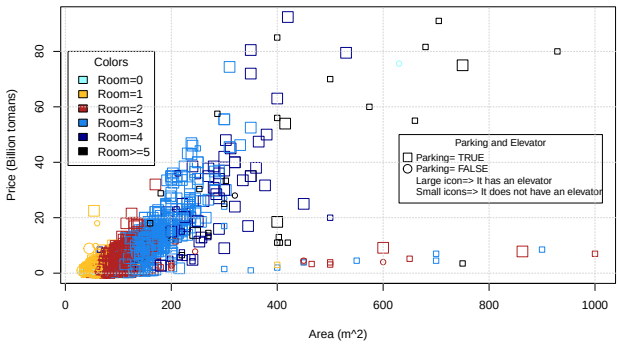
شکل ۱۵: ماتریس نمودار پراکنش برای قیمت و دو پیشگوی عددی مساحت و تعداد اتاق

۴-۲ تصویری‌سازی چند بُعدی

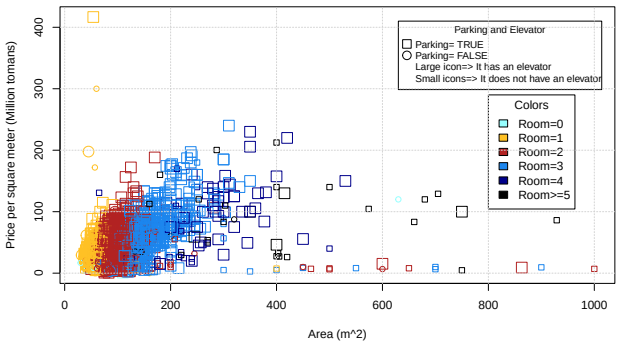
در این بخش تلاش می‌کنم با افزودن متغیرهای رسته‌ای از طریق کدگذاری رنگی و ... جنبه‌های بیشتر از دو بُعد را با استفاده از نمودار پراکنش[۴۵] داده‌ها به تصویر بکشم.

شکل ۱۶: این تصویر در واقع در شکل ۱۵ (پایین سمت چپ تصویر) یک‌بار رسم شده است اما در این تصویر شاهد

[45] Scatter Diagram



شکل ۱۶: نمودار چند بُعدی کدگذاری شده مساحت–قیمت



شکل ۱۷: نمودار چند بُعدی کدگذاری شده مساحت–قیمت هر متر

کدگذاری‌هایی هستیم که به لطف آنها می‌توان، هرچند به صورت شهودی متغیر پاسخ به همراه ۴ ویژگی دیگر را به صورت مجزا مورد مشاهده قرار داد.

ستون‌های عمودی و افقی را به ترتیب به متغیر پاسخ و ویژگی پیشگوی مساحت (که بالاترین نرخ همبستگی را با متغیر پاسخ داشت) اختصاص داده‌ام. رنگ هر داده نمایانگر تعداد اتاق خواب‌های خانه موردنظر است. همچنین ویژگی پارکینگ توسط نوع شکل هندسی متناظر با داده مشخص شده است. در انتها با اندازه اشکال هندسی رسم شده برای هر داده داشتن یا نداشتن آسانسور را در هر ثبت مشخص کرده‌ام.

اولین چیزی که من می‌بینم رنگ مربوط به خانه‌های یک تا سه خواب و بعد از آن خانه‌های ۴ خواب است. در مورد خانه‌های یک تا سه خواب، می‌توان گفت بیشینه‌ی قیمت بیشتری را (با صرف نظر از داده‌های دور افتاده) شاهد هستیم. به عبارت ساده‌تر عموماً هر چه تعداد خواب خانه بیشتر باشد، آن خانه می‌تواند گران‌تر باشد.

نکته دیگر این است که اولاً خانه‌های دارای پارکینگ بسیار بیشتر از خانه‌های بدون پارکینگ هستند و ثانیاً معمولاً خانه‌های بدون پارکینگ ارزان‌تر هستند. یادآوری می‌کنم که بیش از ۸۴ درصد خانه‌های مورد مطالعه من دارای آسانسور هستند.

همچنین با نگاهی اجمالی، واضح است که هم تعداد خانه‌های دارای آسانسور از خانه‌های بدون آسانسور بیشتر است و هم کمتر شاهد خانه‌های بدون آسانسور نسبتاً گران‌قیمت هستیم. اما باید اضافه کنم که خانه‌های بدون آسانسور به وضوح در زمره داده‌های دور از محل تجمع کلی داده ها دیده می‌شوند. احتمالاً به این دلیل است که خانه های ویلایی که عموماً خیلی گرانند، فاقد آسانسور هستند.

شکل ۱۷: توضیحات مشابه در این نمودار مانند نمودار ۱۶ نیز صادق است. لذا فقط به تفاوت‌ها خواهم پرداخت. در نمودار ۱۷ من به جای قیمت کلی خانه که برحسب میلیارد تومان به تصویر کشیده شده بود قیمت هر متر خانه را مورد بررسی قرار داده‌ام.

قیمت هر متر خانه به طور غیرمستقیم گویای متغیر رسته‌ای آدرس است. (این ادعا در شکل ۱۷ صحه سنجی گردیده است.) در این نمودار می‌توان دید خانه‌های دو خواب عموماً مساحت بیشتری از خانه‌های یک خواب دارند اما به ازای هر متر تفاوت قیمتی چندانی ندارند.

همچنین می‌توان دید محله‌های ارزان قیمت‌تر علاقه کمتری به خانه‌های ۳ خواب نشان می‌دهند.[۴۶] بر خلاف تصور عام، خانه‌های ۴ خواب هم در محلات گران قیمت و به همان نسبت در محلات ارزان قیمت دیده می‌شوند.

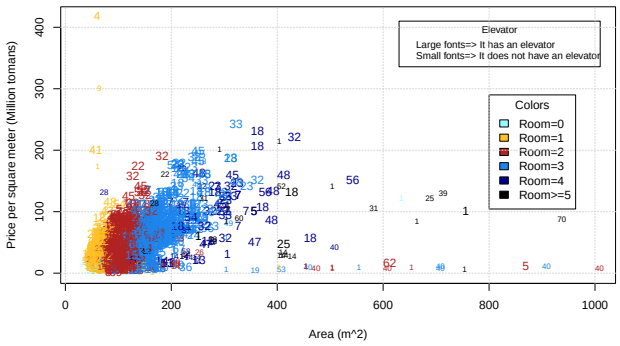
داده‌های دورافتاده اینجا در مورد خانه‌های یک خواب بسیار شگفت انگیزند. (داده‌های دور افتاده دیگر تا حدی قابل توجیه هستند.) اما این ۴ ثبت از خانه‌های یک خواب، نه تنها مساحت زیادی ندارند (در نتیجه یک خواب هستند) بلکه قیمت یک متر خانه در آنها به شکل قابل توجهی گران است. من حتی در مورد یکی از ثبت‌ها که نه آسانسور دارد و نه پارکینگ دارد و نه با توجه به مساحت می‌تواند یک خانه ویلایی خیلی بزرگی باشد، علت گران بودن را در موقعیت تجاری می‌بینم.

نکته دیگر اینکه خانه‌های ۵ خواب و یا بیشتر، معمولاً آسانسور ندارند. احتمالاً همان‌طور که چندین بار اشاره کرده‌ام ممکن است این ثبت‌ها مربوط به خانه‌های ویلایی باشند.

شکل ۱۸: از آنجایی که متوجه شدم شکل ۱۷ ارتباط معناداری بایستی با محله داشته باشد، سعی کرد رسته کدگذاری شده هر محله را به جای هر ثبت بنویسم. در این نمودار ویژگی پارکینگ دیده نمی‌شود. اما متغیر رسته‌ای آدرس جای اشکال هندسی شکل ۱۷ و ۱۶ را گرفت است.

من متوجه شدم که رسته ۱ که رسته "سایر" بود به شکل خیلی پراکنده‌ای در محلات مختلف شامل محلات خیلی گران‌قیمت و

[۴۶] به شکل ۱۹ رجوع کنید.

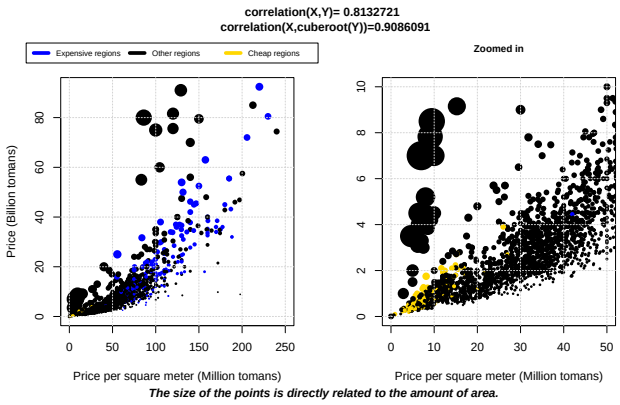


شکل ۱۸: نمودار چند بُعدی کدگذاری شده مساحت-قیمت هر متر با تمرکز روی ویژگی آدرس

خیلی ارزان‌قیمت توزیع‌شده است. که البته با تجه به نوع انتخاب دسته دور از انتظار هم نیست.

نکته دیگر این است که همان‌طور که انتظار می‌رفت بسیاری از رسته‌های مشابه در این نمودار خیلی نزدیک به هم هستند. متأسفانه برای ساختن محور عمودی در این نمودار از ضریبی از متغیر پاسخ استفاده شده. اگر چنین نبود و آدرس، متغیر پاسخ بود، الگوریتم KNN می‌توانست یک گزینه خوب برای پیشگویی آدرس باشد.

نکته دیگری که به صورت غیر رسمی عنوان می‌کنم این است که هر چه مساحت خانه‌ها بیشتر باشند، به طور کلی قیمت به ازای هر متر در همان محله کاهش می‌یابد. از آنجایی که مصداق عددی برای آن نیاوردم به صورت رسمی این نکته را تأیید نمی‌کنم. اما به شکل شهودی مصداق‌های زیادی را در شکل می‌بینم. (البته که خلاف این مصداق‌ها هم وجود دارند.)



شکل ۱۹: نمودار چند بُعدی کدگذاری شده قیمت هر متر- قیمت کل

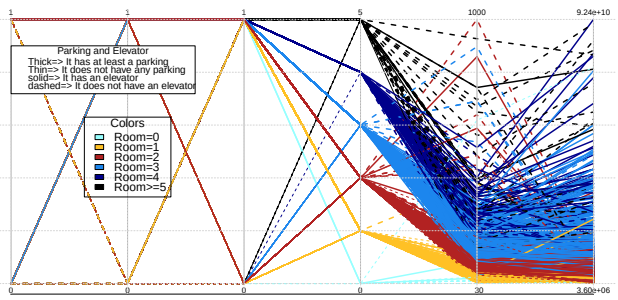
شکل ۱۹: پیش‌تر نتیجه‌گیری کرده بودم که مردم در مناطق ارزان قیمت علاقه کمتری به خانه‌های سه خواب نشان می‌دهند. در اینجا لازم است به دو مورد مهم اشاره کنم. این فرض در صورتی صحیح است که اولاً این نمونه‌ها یک نمونه خوب از جامعه باشند. یعنی به صورت تصادفی مناطق ارزان قیمت خالی از خانه‌های سه خواب تهیه نشده باشند. ثانیاً رابطه مستقیمی میان قیمت هر متر مناطق ارزان قیمت و گران قیمت وجود داشته باشد. با فرض اینکه این نمونه یک نمونه خوب باشد پیش می‌رویم.

در این شکل من نمودار قیمت هر متر-قیمت کل را با دو بزرگنمایی مختلف رسم کرده‌ام. نکته جالب اینجاست که نه تنها این دو پارامتر به هم وابسته هستند بلکه یک رابطه غیرخطی از جنس چندجمله‌ای درجه سه بین آنها برقرار است (رجوع کنید به بالانویس شکل ۱۹)

من در این نمودار چهار منطقه‌ای که میانه بالاتری در متغیر پاسخ، یا همان قیمت داشتند شامل مناطق الهیه، زعفرانیه، نیاوران و فرمانیه با رنگ آبی مشخص کرده‌ام. همچنین چهار منطقه ارزان قیمت شامل پرند، پردیس، پاکدشت و چیتگر شمالی نیز با رنگ طلایی مشخص شده‌اند. بقیه مناطق با رنگ مشکی در نمودار قابل مشاهده‌اند.

واضح است که داده‌های مناطق ارزان قیمت ما هم از نظر متغیر پاسخ یعنی قیمت کل، و هم از نظر قیمت هر متر به شکل قابل توجهی دارای مقادیر کمتری از خانه‌های گران قیمت هستند. لذا می‌توان به محکمی ادعا کرد که متغیر آدرس که نمایانگر مناطق مختلف است، ارتباط قابل توجهی با قیمت هر متر خانه دارد. لذا ادعای من در شکل ۱۷ ادعای غلطی نبوده است.

۲-۵ نمودار با محورهای موازی



شکل ۲۰: نمودار چند بُعدی کدگذاری شده با محور های موازی

شکل ۲۰: این نمودار به شکل دیگری حاوی اطلاعاتی است که نمودار ۱۶ در اختیار ما می‌گذاشت. من سعی کردم در این نمودار محورهای موازی، همان شکل ۱۶ را برای به تصویر کشیدن بهتر ابعاد بالاتر استفاده کنم.

۶-۲ کاهش بُعد

در یادگیری ماشین[47] و آمار کاهش بُعد[48] یا کاهش ابعاد روند کاهش تعداد متغیرهای تصادفی راهنماییده[49] [۱۲] از طریق به دست آوردن یک مجموعه از متغیرهای اصلی می‌باشد. کاهش ابعاد را می‌توان به انتخاب ویژگی و استخراج ویژگی تقسیم کرد. [۱۳]

۷-۲ تجزیه و تحلیل مؤلفه اصلی

تجزیه و تحلیل مؤلفه اصلی[50] فرآیند محاسبه مؤلفه‌های اصلی و استفاده از آنها برای انجام تغییر مختصات[51] بر روی داده‌ها است، به گونه‌ای حداکثر واریانس[52] داده‌ها حفظ شود. گاهی اوقات این الگوریتم فقط از چند مؤلفه اصلی استفاده می‌کند و بقیه را نادیده می‌گیرد. [۱۴]

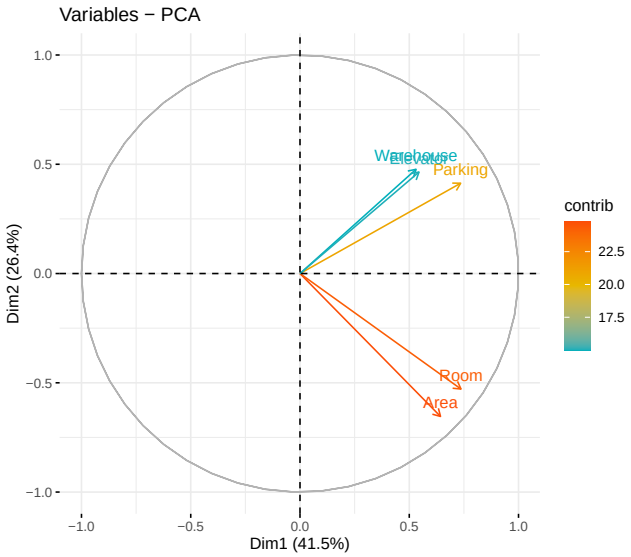
من در این قسمت صرفاً برای اینکه توضیح بدهم کاهش بُعد گزینه مناسبی برای من در این پروژه نیست گزارش‌های پیش‌رو را گرفتم. اما پیش از آن نیز با توجه به پایین بودن تعداد ویژگی‌ها کاملاً مشخص بود که قرار نیست کاهش بُعدی روی داده‌ها صورت بگیرد. ضمناً با توجه به شکل ۱۴ می‌توان گفت متغیرهای پیشگوی آسانسور و انباری همبستگی بالایی با هم ندارند. اما اگر ما از مؤلفه‌های اصلی در الگوریتم PCA کمک بگیریم، تصویر این دو متغیر پیشگو در راستای دو برداری که بیشترین سهم را در حفظ واریانس داده‌ها دارند، همبستگی قابل توجهی با هم خواهند داشت. این حقیقت را می‌توانید از طریق بررسی شکل ۲۱ برداشت کنید.

	PC۱	PC۲	PC۳	PC۴	PC۵
Standard deviation	۱٫۴۴۰۷	۱٫۱۴۹۴	۰٫۸۹۵۶	۰٫۶۸۷۷	۰٫۵۷۲۷۸
Proportion of Variance	۰٫۴۱۵۲	۰٫۲۶۴۲	۰٫۱۶۰۴	۰٫۰۹۴۶	۰٫۰۶۵۶۲
Cumulative Proportion	۰٫۴۱۵۲	۰٫۶۷۹۴	۰٫۸۳۹۸	۰٫۹۳۴۴	۱٫۰۰۰۰۰

جدول ۶: جدول اهمیت مؤلفه‌ها

من برای اینکه بدانم در این بخش باید به کاهش بُعد تن بدهم یا خیر، منابع مختلفی را بررسی کردم. یکی از پاسخ‌هایی که به این سؤال کلی که کاهش بُعد خوب است یا خیر داده شد به نظرم پخته‌تر از بقیه جواب‌ها آمد. من با ذکر نام نویسنده این مطلب را به زبان فارسی ترجمه و سپس از قول ایشان نقل می‌کنم.

> هیچ پاسخ کلی درست یا غلطی برای کاهش بُعد وجود ندارد. درستی یا غلط بودن کاملاً بستگی به



شکل ۲۱: نمودار تحلیل PCA

موقعیت دارد. اجازه دهید چند مثال را برای نشان دادن حالات مختلف بیان کنم. این مثال‌ها ممکن است جامع نباشند، اما ذهنیت خوبی از آنچه باید انجام شود به شما می‌دهند.

- شما در حال ساخت یک مدل رگرسیون با ۱۰ متغیر هستید، که برخی از ویژگی‌های آنها همبستگی ملایمی دارند. در این حالت نیازی به کاهش ابعاد نیست. از آنجایی که بین متغیرها همبستگی کمی وجود دارد، همه آنها اطلاعات جدیدی را در اختیار مدل می‌گذارند. شما باید همه متغیرها را در ترکیب نگه دارید و یک مدل بسازید.
- مجدداً در حال ساخت یک مدل رگرسیون هستید. این بار ۲۰ متغیر پیشگو دارید و برخی از این متغیرها همبستگی بالایی دارند. به عنوان مثال، به دو نوع مجموعه داده فکر کنید.

  داده‌های مخابراتی: تعداد تماس‌های مشتری در یک ماه و صورتحساب ماهانه دریافتی او.

  داده‌های بیمه: تعداد بیمه نامه‌ها و کل حق بیمه فروخته شده توسط یک نماینده یا شعبه.

  در این موارد، به دلیل اینکه تعداد متغیرهای پیشگو محدودی با همبستگی بسیار بالا دارید، می‌توانید فقط یکی از این متغیرها را به مدل خود اضافه کنید. با وارد کردن همه متغیرها، چیز

[47] Machine Learning
[48] Dimension Reduction
[49] Supervised
[50] Principal Component Analysis
[51] Coordinate
[52] Variance

جدیدی به مدل اضافه نمی‌کنید. حتی مقدار زیادی از این مقدار اضافی که اتفاقاً همبستگی بالایی هم دارند می‌تواند نویز باشد. شما هنوز هم می‌توانید از استفاده از هر تکنیک کاهش ابعادی رسمی در این مورد صرف‌نظر کنید (اما با وارد کردن تعداد محدودی از متغیرهای پیشگو به جای همه آنها ذاتاً کاهش بُعد را انجام داده‌اید).

- حال فرض کنید که شما ۵۰۰ متغیر پیشگو از این قبیل دارید که برخی از آنها با یکدیگر همبستگی دارند. به عنوان مثال، خروجی داده‌های سنسورها از یک گوشی هوشمند. با توجه به تعداد بالای ویژگی‌ها و همبستگی نسبی خیلی از آنها که با هم مرتبط هستند و اتفاقاً خیلی هم زیاد هستند، نمی‌توان به‌طور جداگانه فهمید که کدام‌یک به کدام‌یک مرتبط است. در این مورد، قطعاً باید از تکنیک‌های کاهش ابعاد استفاده کنید. شما ممکن است معنای واقعی این بردارها را بفهمید یا نه، اما همچنان می‌توانید با مشاهده نتیجه الگوریتم‌های کاهش بُعد تأثیر بسیاری از این ویژگی‌ها را درک کنید.

  در بسیاری از سناریوهایی مانند این، همچنین خواهید دید که بیش از ۹۰ درصد اطلاعات را با کمتر از ۱۵ تا ۲۰ درصد متغیرها حفظ خواهید کرد. از این رو، الگوریتم‌های کاهش بُعد می‌توانند کاربردهای بسیاری داشته باشند.

---

[۱۵] Kunal Jain from Gurgaon, India

با توجه به توضیحات مذکور من نیز از اعمال الگوریتم‌های کاهش بُعد بر روی داده‌های خود، خودداری می‌کنم.

۸-۲ داده کاهی

معمولاً در پروژه‌های داده‌کاوی خیلی کم پیش می‌آید که بخشی تحت عنوان داده کاهی[۵۳] داشته باشیم. اما دلیلی که من را به این نتیجه رساند تا در این پروژه این بخش را بگنجانم، شکل ۱۳ بود. شما در این شکل می‌بینید که داده‌هایی از محله‌های مختلف هستند، که به شکل غیرقابل توجیهی با قیمت میانگین منطقه فاصله دارند. وجود این داده‌ها ممکن است دلایل زیادی داشته باشد. از وجود نویز در داده‌ها گرفته تا شرایط خاص ملک، همه می‌توانند دلایل دورافتادگی این داده‌ها باشند. دلیل هر کدام از این‌ها که باشد، نباید اجازه دهیم این داده‌ها، مدل ما را فریب دهند. به خصوص که پروژه ما یک پیشگویی دودویی نیست و الگوریتم‌های رگرسیونی بسیار به این داده‌های دورافتاده حساس هستند.

به طور خلاصه با بررسی داده‌ها و شکل مذکور به این نتیجه رسیدم که ثبت‌های جدول ۷ را بررسی کنم. پس از بررسی به این نتیجه رسیدم که داده‌های جدول ۸ باید از مجموعه داده‌ها حذف شوند.

Area	Room	Parking	Warehouse	Elevator	Address	Price
۱۳۰	۲	۱	۱	۱	Dorous	$۱٫۶۹ \times ۱۰^{۱۰}$
۳۶۵	۴	۱	۱	۱	Dorous	$۴٫۷۴۵ \times ۱۰^{۱۰}$
۳۵۰	۴	۱	۱	۱	Niavaran	$۸٫۰۵ \times ۱۰^{۱۰}$
۳۵۰	۳	۱	۱	۰	Pasdaran	$۱٫۰ \times ۱۰^{۹}$
۱۴۵	۳	۱	۱	۱	Pasdaran	$۱٫۴۵ \times ۱۰^{۹}$
۷۰۵	۵	۱	۱	۰	Abazar	$۹٫۱ \times ۱۰^{۱۰}$
۸۳	۲	۱	۱	۱	Ozgol	$۵٫۵ \times ۱۰^{۷}$
۳۵۰	۴	۱	۱	۱	Niavaran	$۷٫۲ \times ۱۰^{۱۰}$
۳۰۰	۳	۱	۱	۰	Golestan	$۲٫۴ \times ۱۰^{۱۰}$
۳۱۰	۳	۱	۱	۱	Aqdasieh	$۷٫۴۴ \times ۱۰^{۱۰}$
۵۳۰	۴	۱	۱	۱	Dorous	$۷٫۹۵ \times ۱۰^{۱۰}$
۶۰	۱	۰	۰	۰	Shahr-e-Ziba	$۱٫۸ \times ۱۰^{۱۰}$
۳۲۰	۵	۰	۰	۰	Sattarkhan	$۲٫۸ \times ۱۰^{۱۰}$
۵۴	۱	۱	۱	۱	West Ferdows Boulevard	$۲٫۲۵ \times ۱۰^{۱۰}$
۴۵	۱	۰	۰	۱	Si Metri Ji	$۸٫۹ \times ۱۰^{۹}$

جدول ۷: جدول کاندیدای داده‌کاهی

در مورد هر یک از ثبت‌های جدول ۸ توضیح مختصری می‌دهم که توجیهم را برای حذف این داده‌ها بیان کنم.

- یک خانه ۳۵۰ متری سه خواب در منطقه پاسداران مجهز به انباری و پارکینگ فقط یک میلیارد تومان ثبت شده است که انتظار داریم این عدد حداقل در کمترین حالت ۱۰ برابر قیمت ثبت شده باشد.

- یک خانه ۱۴۵ متری سه خواب در منطقه پاسداران مجهز به هر سه امکانات پارکینگ، انباری و آسانسور که فقط ۱ میلیارد و ۴۵۰ میلیون تومان ثبت شده است. تخمین ذهنی من به‌هیچ‌وجه این داده را صحیح نمی‌داند.

- یک خانه ۸۳ متری ۲ خواب در منطقه ازگل یک خواب بدون انباری، آسانسور و پارکینگ که فقط ۵۵۰ میلیون تومان ثبت شده است. این خانه با توجه به مساحت نه چندان زیاد و عدم مجهز بودن به امکانات مذکور قیمت بالایی نمی‌تواند داشته باشد ولی ۵۵۰ میلیون تومان برای یک خانه در شمال تهران حتی در صورتی که خانه در طبقه منفی ۱ واقع‌شده باشد هم بسیار دور از ذهن است.

- خانه‌ای ۶۰ متری یک خواب بدون انباری، پارکینگ و آسانسور در منطقه شهر زیبا که یک منطقه گران‌قیمت

[53] Data Reduction

محسوب نمی‌شود، ۱۸ میلیارد تومان ارزش‌گذاری شده است. این خانه حتی اگر یک دهم این مقدار قمت گذاری می‌شد باز هم ارزش خرید بالایی نداشت.

- خانه‌ای ۵۴ متری یک خواب مجهز به انباری، پارکینگ و آسانسور در منطقه بلوار فردوس که مانند شهر زیبا یک منطقه گران‌قیمت محسوب نمی‌شود، بیش از ۲۲ میلیارد تومان (به عبارتی متری بیش از ۴۰۰ میلیون تومان) ارزش‌گذاری شده است. نیاز به توضیح نیست که این ثبت هم به‌احتمال قریب‌به‌یقین به‌اشتباه ثبت‌شده است.

- خانه‌ای ۴۵ متری یک خواب مجهز آسانسور بدون انباری و پارکینگ در منطقه سی متری جی که از مناطق بسیار ارزان‌قیمت تهران به شمار می‌رود، نزدیک به ۹ میلیارد تومان (به عبارتی متری حدوداً ۱۴۰ میلیون تومان) قیمت‌گذاری شده است. کافی است یادآور شوم که چارک سوم قیمت هر متر خانه در مناطق زعفرانیه، ولنجک و نیاوران از این عدد کمتر است.

Area	Room	Parking	Warehouse	Elevator	Address	Price
۳۵۰	۳	۱	۱	۰	Pasdaran	۱×۱۰^{۹}
۱۴۵	۳	۱	۱	۱	Pasdaran	$۱٫۴۵ \times ۱۰^{۹}$
۸۳	۲	۱	۱	۱	Ozgol	$۵٫۵ \times ۱۰^{۷}$
۶۰	۱	۰	۰	۰	Shahr-e-Ziba	$۱٫۸ \times ۱۰^{۱۰}$
۵۴	۱	۱	۱	۱	West Ferdows Boulevard	$۲٫۲۵ \times ۱۰^{۱۰}$
۴۵	۱	۰	۰	۱	Si Metri Ji	$۸٫۹ \times ۱۰^{۹}$

جدول ۸: جدول نهایی داده‌هایی که در بخش داده‌کاهی بایستی حذف شوند

۹-۲ اعمال تغییرات نتیجه‌گیری شده

با اجرای کد پیش رو داده‌های مذکور از ماتریس اصلی داده‌های من پاک می‌شوند.

```
#deleting outliers
X<-X[-c(1442,1516,2767,3128,3390,3416),]
```

واضح است که کدگذاری من روی متغیرهای رسته‌ای و همچنین افراز داده‌های من باید با توجه به همین ماتریس جدید داده‌ها به‌روزرسانی شوند.

```
#Data splitting
set.seed(1)
train.rows<-sample(rownames(X_add),dim(X)[1]
    *0.6)
valid.rows<-sample(setdiff(rownames(X_add),
    train.rows),dim(X)[1]*0.3)
test.rows<-setdiff(rownames(X_add),union(
    train.rows,valid.rows))
training_set<-X_add[train.rows,]
validation_set<-X_add[valid.rows,]
test_set<-X_add[test.rows,]


#Data standardization
X_standard_scaler.train<-training_set
X_standard_scaler.valid<-validation_set
X_standard_scaler.test<-test_set
#- - - - - - - - - - - - - - - - -
for(i in c(1,2,7)){
    X_standard_scaler.train[,i]=scale(
    training_set[i])
}
#- - - - - - - - - - - - - - - - -
for(j in c(1,2,7)){
    for (i in 1:dim(validation_set)[1]){
        X_standard_scaler.valid[i,j]=(
    validation_set[i,j]-mean(training_set[,j
    ]))/sd(training_set[,j])
    }
}
#- - - - - - - - - - - - - - - - -
for(j in c(1,2,7)){
    for (i in 1:dim(test_set)[1]){
        X_standard_scaler.test[i,j]=(test_
    set[i,j]-mean(training_set[,j]))/sd(
    training_set[,j])
    }
}

#target_encoding
target_encod <- build_target_encoding( X_
    standard_scaler.train, cols_to_encode =
    "Address",target_col = "Price",
    functions = c("median"))
X_standard_scaler.train=target_encode( X_
    standard_scaler.train, target_encoding =
     target_encod)[,c(2,3,4,5,6,8,7)]
X_standard_scaler.valid=target_encode( X_
    standard_scaler.valid, target_encoding =
     target_encod)[,c(2,3,4,5,6,8,7)]
X_standard_scaler.test=target_encode( X_
    standard_scaler.test, target_encoding =
    target_encod)[,c(2,3,4,5,6,8,7)]

#For convenience
train.data<-X_standard_scaler.train
valid.data<-X_standard_scaler.valid
test.data<-X_standard_scaler.test
```

۳ رگرسیون خطی چندگانه

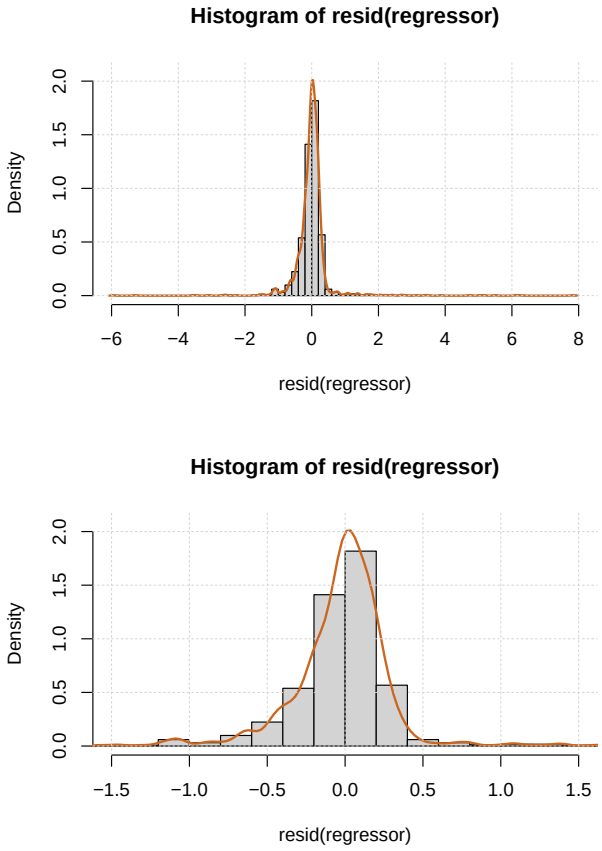
اکنون آماده هستیم تا اولین مدل پیش‌گویانه را روی داده‌های مجموعه آموزشی اعمال کنیم. همان‌طور که نام فصل گویاست از مدل رگرسیون خطی چندگانه[۵۴] برای پیشگویی استفاده می‌کنیم. دلیل این‌که این مدل را قبل از مدل‌های دیگر استفاده کرده‌ام این است این مدل در انتخاب ویژگی‌ها به من کمک بزرگی می‌کند. در ادامه خواهم گفت که کدام ویژگی‌ها اهمیت بیشتری دارند.

```
regressor=lm(formula=Price~.,data=train.data
    )
#summary(regressor)
```

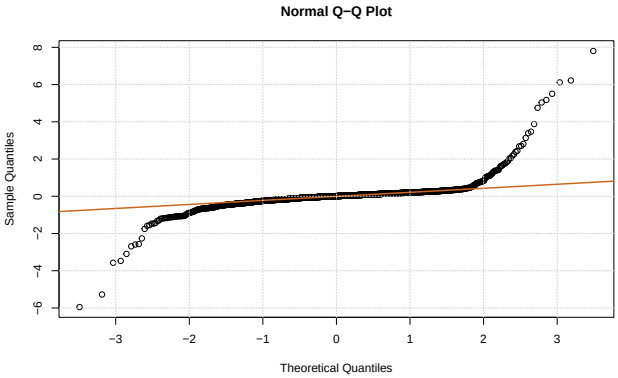
با اجرای دستور فوق مدل رگرسیون[۵۵] من اجرا می‌شود. یک سؤال مهم اینجاست که آیا مانده[۵۶]های ما از توزیع نرمال پیروی می‌کنند؟ اگر مانده‌های من از توزیع نرمال پیروی نکنند، مدل رگرسیونی دچار اشکال می‌شود. لذا اول این نکته را به کمک دستورهای پیشرو بررسی می‌کنم.

```
par(mfcol=c(2,1))
hist(resid(regressor),,breaks=80,prob = TRUE
    ,ylim=c(0,2.1))
lines(density(resid(regressor)),lwd = 2,col
    = "chocolate3")
hist(resid(regressor),,breaks=80,prob = TRUE
    ,ylim=c(0,2.1),xlim=c(-1.5,1.5))
lines(density(resid(regressor)),lwd = 2,col
    = "chocolate3")

par(mfcol=c(1,1))
qqnorm(resid(regressor))
qqline(resid(regressor),lwd = 2,col = "
    chocolate3")
```



شکل ۲۲: توزیع مانده‌های مدل رگرسیونی روی متغیر پاسخ



شکل ۲۳: نمودار چندک چندک مدل رگرسیونی روی متغیر پاسخ

شکل ۲۲ نمایانگر این است که توزیع مانده‌های مدنظر ما تقریباً نرمال است اما این حقیقت را نیز می‌توان از تحلیل نمودار چندک چندک[۵۷] در شکل ۲۳ نیز دریافت.

اکنون می‌توانیم با خاطری آسوده‌تر مدل رگرسیونی خود را روی داده‌ها اعمال کنیم. اما قبل از هر چیز دوست دارم به اهمیت بررسی این فرض توزیع نرمال مانده‌ها اشاره کنم. اساتید بزرگوار آقایان Donald A. Pierce و Daniel W. Schafer، از اعضای هیئت‌علمی گروه آمار دانشگاه ایالتی اورگان آمریکا در مقاله‌ای [۱۶] مفصل به اهمیت این موضوع می‌پردازند و خاطرنشان می‌شوند که بسیاری از پروژه‌هایی که از مدل رگرسیونی چندگانه استفاده می‌کنند این فرض را در نظر نمی‌گیرند و به همین دلیل نتایج این مدل‌ها قابل‌اعتماد نیستند.

قبل از هر چیز از تابع `summary` که پیش‌تر آن را به صورت توضیح[۵۸] نوشته بودیم خروجی می‌گیریم تا مهم‌ترین ویژگی‌ها را مشخص کنیم.

[54] Multiple Linear Regression
[55] Regression
[56] Residual
[57] QQ Plot
[58] Comment

```
> summary(regressor)

Call:
lm(formula = Price ~ ., data = train.data)

Residuals:
     Min       1Q   Median       3Q      Max
-5.9518 -0.1515   0.0086   0.1410   7.8056

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)               0.09112    0.04831   1.886  0.05943 .
Area                      0.49521    0.01812  27.334  < 2e-16 ***
Room                      0.04974    0.01775   2.802  0.00513 **
Parking                  -0.06978    0.04294  -1.625  0.10429
Warehouse                 0.10966    0.05080   2.159  0.03099 *
Elevator                 -0.04914    0.03451  -1.424  0.15466
Price_median_by_Address   0.76646    0.02552  30.030  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘’. 0.1 ‘ ’ 1

Residual standard error: 0.5729 on 2073 degrees of freedom
Multiple R-squared:  0.6728,  Adjusted R-squared:  0.6718
F-statistic: 710.3 on 6 and 2073 DF,  p-value: < 2.2e-16
```

با بررسی پی-مقدار[۵۹] ها متوجه می‌شویم ویژگی‌های مساحت و آدرس بیشترین اهمیت را دارا هستند. بعد از آن ویژگی تعداد اتاق و پس از آن ویژگی داشتن یا نداشتن انباری از نظر مدل رگرسیونی من مهم تلقی شده‌اند. اما آیا درست است که بقیه ویژگی‌ها را حذف کنم؟ قطعاً نه! من برای پاسخ به این سؤال که بهترین نتیجه را با وارد کردن کدام پیشگوها به مدل می‌گیرم، به داده‌های اعتبار سنجی نیاز دارم. لازم به ذکر است مدل رگرسیونی به طور کلی نیاز به داده‌های مجموعه تست ندارد. اما اینجا چون از داده‌های مجموعه اعتبار سنجی برای بهبود یادگیری مدل استفاده می‌کنم من در انتهای پروژه در صورتی که این مدل بالاترین دقت را داشته باشد، مدل رگرسیونی نهایی را به کمک مجموعه تست ارزش‌گذاری خواهم کرد.

اکنون با ارزیابی مدل رگرسیونی روی داده های مجموعه اعتبار سنجی آغاز میکنیم. ضمنا مبنای ما برای ارزیابی عملکرد در این پروژه متر میانگین توان دوم خطاها[۶۰] یا همان MSE است.

```
> y_pred=predict(regressor,newdata=valid.
    data)
> #data.frame(TruePrice=valid.data$Price ,
    Predicted_price=y_pred)
> MSE(valid.data$Price,y_pred)
[1] 0.4927284
```

دریافتم که میانگین توان دوم خطاها در تلاش اول من برابر با ۰٫۴۹۲۷۲۸۴ بود اکنون می‌خواهم با استفاده از روش‌های پیشرو[۶۱]، پسرو[۶۲] و گام‌به‌گام[۶۳] اقدام به بررسی مجدد خروجی مدل رگرسیونی با ویژگی‌های مختلف کنم. با توجه به اینکه مساحت یک ویژگی مهم تلقی شده بود از این ویژگی شروع می‌کنم. کد پیش رو حالت‌های مختلف را بررسی می‌کند. من از قرار دادن خروجی کد به علت طولانی شدن صرف‌نظر می‌کنم اما خلاصه تفسیر آن را قرار خواهم داد.

```
#variable selection
base_model=lm(Price~Area,data=train.data)
step_model=step(base_model,scope=list(upper=
    regressor , lower= ~1), direction="
    forward",trace=T)
print(step_model)
forward_pred=predict(step_model,newdata=
    valid.data)
#data.frame(TruePrice=valid.data$Price ,
    Predicted_price=forward_pred)
MSE(valid.data$Price,forward_pred)
#- - - - - - - - - - - - - - - - - - - - - -
base_model=lm(Price~Area,data=train.data)
step_model=step(base_model,scope=list(upper=
    regressor , lower= ~1), direction="
    backward",trace=T)
print(step_model)
backward_pred=predict(step_model,newdata=
    valid.data)
#data.frame(TruePrice=valid.data$Price ,
    Predicted_price=backward_pred)
MSE(valid.data$Price,backward_pred)
#- - - - - - - - - - - - - - - - - - - - - -
base_model=lm(Price~Area,data=train.data)
step_model=step(base_model,scope=list(upper=
    regressor , lower= ~1), direction="both"
    ,trace=T)
print(step_model)
stepwise_pred=predict(step_model,newdata=
    valid.data)
#data.frame(TruePrice=valid.data$Price ,
    Predicted_price=stepwise_pred)
MSE(valid.data$Price,stepwise_pred)
```

خروجی‌های بازبرازشی که از روش‌های forward و stepwise به‌دست آوردیم نتیجه‌ای مانند رگرسیون اولیه را روی داده‌های مجموعه اعتبار سنجی به من داده‌اند. توجه شما را به گزارش پیشرو جلب می‌نمایم.

```
> print(step_model)

Call:
lm(formula = Price ~ Area + Price_median_by_
    Address + Room +
    Elevator + Warehouse + Parking, data = train.
    data)

```

[59] P-Value
[60] Mean Squared Error
[61] Forward
[62] Backward
[63] Stepwise

```
Coefficients:
(Intercept)  Area  Price_median_by_Address  Room
0.09112    0.49521           0.76646      0.04974
Elevator    Warehouse    Parking
 -0.04914     0.10966    -0.06978
```

تفسیر من این است که انتخاب ویژگی رگرسیون خطی چندگانه به من می‌گوید همه متغیرهای من بامعنی هستند و برای پیشگویی مدل استفاده می‌شوند اما چیزی که برخلاف شهود من است این است که ویژگی‌های داشتن پارکینگ و آسانسور ضریب منفی دارد. یعنی مدل من خانه‌های دارای پارکینگ و آسانسور را از خانه مشابه بدون این امکانات ارزان‌تر پیشگویی می‌کند.

به‌هرروی عدد نهایی برای میانگین توان دوم خطاها در این روش روی مجموعه اعتبار سنجی ۰٫۴۹۲۷۲۸۴ است.

شایان ذکر است که همین شاخص وقتی‌که از متغیّر ظاهری[۶۴] استفاده می‌کردم، برابر ۰٫۶۳۰۰۷۵۳ و اگر داده‌ها را نرمال‌سازی نیز نمی‌کردم آماره‌های میانگین توان دوم خطاها و میانگین قدرمطلق خطاها[۶۵] به ترتیب $۱۰^{۱۹} \times ۳٫۸۸۶۰۴۲$ و ۲۹۰۰۴۹۸۹۸۷ می‌شدند.

[64] Dummy Variable

[65] Mean Absolute Error

۴ مدل درخت تصمیم

الگوریتم درخت تصمیم[۶۶]، یا درخت رگرسیون[۶۷] (در اینجا چون روی پیشگویی کار می‌کنیم درخت رگرسیون مفهوم دارد) دارای فرا پارامتر[۶۸]هایی است که احتیاج به تنظیم[۶۹] فرا پارامترها یا بهینه‌سازی فرا پارامتر[۷۰] دارند. یک راه ساده جستجوی شبکه‌ای[۷۱] است. در این روش با جایگزین کردن بخشی از فرا پارامترها تلاش می‌کنیم فرا پارامترهایی که بیشترین دقت (کمترین خطا) را روی مجموعه اعتبار سنجی کسب کرده‌اند را پیدا کنیم. سپس مدل را به عنوان مدل نهایی در این بخش (و نه در کل پروژه) معرفی کنیم.

```
#Define hyperparameters for grid search
minsplit<-seq(1,30,1)
maxdepth<-seq(1,30,1)
cp_float<-seq(0,30,1) #This variable must be a float number. Due to the limitations of the expand.grid function, I will modify this parameter later.
hyperparam_grid_dt_model<-expand.grid(minsplit=minsplit,maxdepth=maxdepth,cp=cp_float)
num_models<-nrow(hyperparam_grid_dt_model)
dt_model_grid_search<-list()
#Apply all models
for(i in 1:num_models){
    minsplit<-hyperparam_grid_dt_model$minsplit[i]
    maxdepth<-hyperparam_grid_dt_model$maxdepth[i]
    cp_float<-1/((sqrt(2))^(hyperparam_grid_dt_model$cp[i]))
    dt_model_grid_search[[i]]<-rpart(formula=Price~., data=train.data, method="anova" ,control=rpart.control(maxdepth=maxdepth,minsplit=minsplit,cp =cp_float))
}
#compute the loss (MSE based)
MSE_val_dt<-c()
for(i in 1:num_models){
    MSE_val_dt[i]<-MSE(predict(dt_model_grid_search[[i]], newdata =valid.data),valid.data[,7])
}
#identify the best model
best_dt_model<-dt_model_grid_search[[which.min(MSE_val_dt)]]
```

تا این مرحله بهترین مدل را روی داده‌های اعتبارسنجی پیدا کردیم. ابتدا یک گزارش بگیریم تا فرا پارامترهای منتخب مشخص شوند.

```
> best_dt_model$control
$minsplit
[1] 12
$minbucket
[1] 4
$cp
[1] 0.000345267
$maxcompete
[1] 4
$maxsurrogate
[1] 5
$usesurrogate
[1] 2
$surrogatestyle
[1] 0
$maxdepth
[1] 5
$xval
[1] 10
```

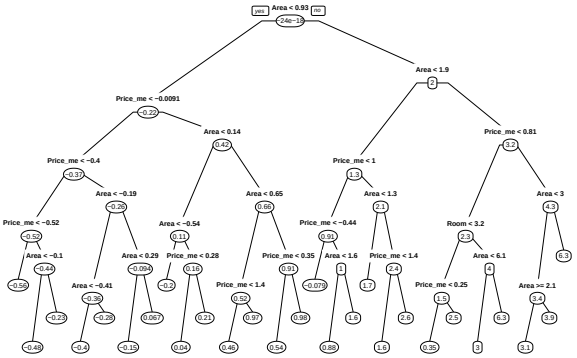
حال وقت آن رسیده است که خطای مدل نهایی این قسمت را روی مجموعه اعتبار سنجی بسنجم.

```
>y_pred=predict(best_dt_model,newdata=valid.data)
>MSE(valid.data$Price,y_pred)
[1] 0.2755766
```

لذا خطای ما روی مجموعه اعتبار سنجی برابر ۰٫۲۷۵۵۷۶۶ شد. اما از آنجایی که از درخت رگرسیون می‌توان تعدادی نمودار رسم کرد که به کمک آنها به درک درست‌تری از ویژگی‌ها برسیم، این فصل را در این نقطه تمام نمی‌کنم و قبل از آن این نمودارها را بررسی خواهم کرد.

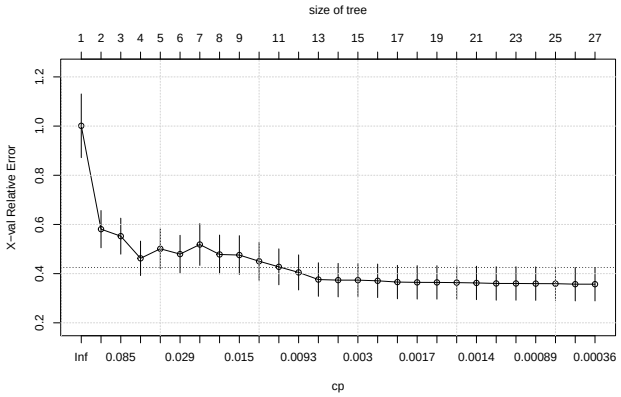
با توجه به شکل ۲۴ می‌توان دریافت که مهم‌ترین ویژگی‌ها مانند آنچه در فصل قبل مشاهده کردیم، ویژگی‌های آدرس و مساحت هستند. اما در برخی مواقع، الگوریتم، برای تشخیص بهتر از ویژگی تعداد اتاق نیز بهره برده است. این نشان می‌دهد ویژگی تعداد اتاق از نظر این مدل اهمیت بیشتری نسبت به سایر ویژگی‌ها داشته است. این میزان اهمیت دادن به داده‌ها را نیز در اجرای الگوریتم رگرسیون خطی چندگانه نیز مشاهده کردیم. اما در آن فصل مدل تشخیص داده بود برای خطای کمتر به همه ویژگی‌ها نیاز دارد. در حالی که در این مدل مشاهده کردیم که چنین نیست.

[66] Decision Tree
[67] Regression Tree
[68] Hyperparameter
[69] Tuning
[70] Hyperparameter Optimization
[71] Grid Search



شکل ۲۴: نمودار درخت رگرسیون

تا اینجای کار الگوریتم درخت رگرسیون بهترین نتیجه (کم‌ترین میانگین مربعات خطاها) را بر اساس داده‌های مجموعه اعتبار سنجی به من داده است. به نظر من ضعف این روش با توجه به ماهیت الگوریتم این است که در پیشگویی قیمت داده‌های دور افتاده از نظر متغیر پاسخ، نباید خوب عمل کند.



شکل ۲۵: نمودار خطا- complexity parameter

شکل ۲۵ نیز تأیید کننده انتخاب پیچیدگی پارامتر[۷۲] برای مدل درخت رگرسیونی است.

[72] Complexity Parameter

۵ مدل k نزدیک‌ترین همسایه

سومین الگوریتمی که قرار است روی داده‌ها اعمال کنم الگوریتم K نزدیکترین همسایه[۷۳] است. چالش مهم من در این الگوریتم محاسبه K مناسب است. لذا بایستی از داده‌های مجموعه اعتبار سنجی، اقدام به بهینه‌سازی فرا پارامتر نماییم.

لازم به ذکر است که من بایستی اجتماع ویژگی‌هایی که الگوریتم‌های درخت رگرسیون و رگرسیون خطی چندگانه آنها را با اهمیت تلقی کرده‌اند را در این الگوریتم وارد می‌کردم. اما من با بررسی همه حالت‌های مختلف به این نتیجه رسیدم که با وارد کردن ویژگی‌هایی که درخت رگرسیون آنها را مهم تلقی کرده بود، بالاترین بازدهی را خواهیم داشت. لذا از ارائه کدهای مربوط به اجتماع ویژگی‌ها، خودداری می‌کنم. اما به طور کلی باید بگویم میانگین توان دوم خطاها در حالتی که همه ویژگی‌ها را وارد مدل کنیم در بهترین حالت K مناسب، برابر ۰٫۵۲۰۱۵۴۲ خواهد شد که با اختلاف بالاترین خطای به‌دست آمده در این پروژه است.

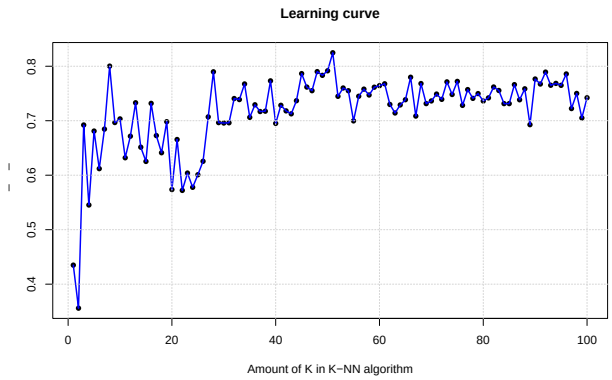
```
MSE_val_knn<-c()
for(i in 1:100){
    y_pred = knn(train = train.data[,c
    (1,2,6)],
         test =  valid.data[,c(1,2,6)],
         cl = train.data[, 7],
         k = i,
         prob = FALSE)
    MSE_val_knn[i]=MSE(as.numeric(as.matrix(
    y_pred)),valid.data[,7])
}
```

اکنون میانگین توان دوم خطاها همه حالت‌های ممکن در MSE_val_knn ذخیره شده است. حال برای به دست آمدن نتایج خواهیم داشت:

```
> which.min(MSE_val_knn)
[1] 2
> MSE_val_knn[which.min(MSE_val_knn)]
[1] 0.3559102
```

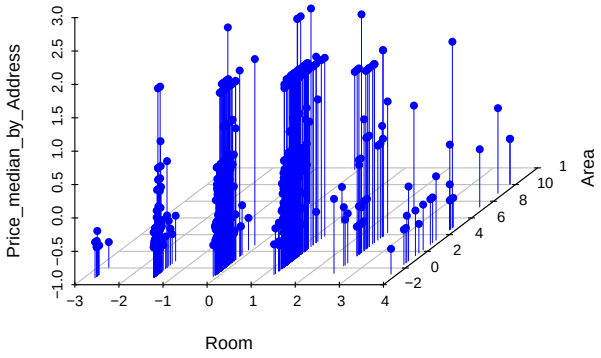
به عبارتی من با قراردادن $K = ۲$ بهترین نتیجه را از این مدل می‌گیرم. میانگین توان دوم خطاها در مدل نهایی من برابر ۰٫۳۵۵۹۱۰۲ است. این خطا از مدل رگرسیون خطی چندگانه نتیجه بهتری است اما همچنان بهترین نتیجه مربوط به درخت رگرسیون است.

میانگین توان دوم خطاها با مقادیر مختلف K به صورتی است که در شکل ۲۶ نمایش داده شده است.



شکل ۲۶: منحنی یادگیری

برای درک بهتر از داده‌هایی که ۲ نزدیک‌ترین همسایه روی آنها محاسبه می‌شوند می‌توان به نمودار ۲۷ مراجعه کرد. این نمودار تصویر واضحی از آنچه در این الگوریتم رخ می‌دهد به ما نشان می‌دهد. ضمناً توجه داشته باشید که مقیاس همه محورها برای نماش بهتر نمودار یکسان نیست.



شکل ۲۷: نمودار پراکنش سه بُعدی

[73] K-Nearest Neighbor

۶ مدل شبکه عصبی

تلاش اولیه من این بود که بتوانم یک مدل یادگیری عمیق[۷۴] روی داده‌ها در زبان برنامه‌نویسی R ایجاد کنم. ولی به علت عدم تسلط کافی من روی این زبان برنامه‌نویسی این مهم محقق نشد. مشکل اساسی من در اعمال مدل ساده شبکه عصبی[۷۵] روی داده‌ها این بود که اولاً در تعریف تابع فعال ساز[۷۶] محدودیت زیادی داشتم ثانیاً توان دسترسی مناسبی به الگوریتم بهینه سازی نداشتم. به هر حال با توجه به نکات گفته شده ابزار مناسبی برای استفاده حداکثری از توان شبکه عصبی برای من میسر نبود.

با این حال من مدل شبکه عصبی را روی همین داده‌ها با زبان برنامه‌نویسی پایتون پایتون[۷۷] اعمال کردم و نتیجه به طور شگفت انگیزی از آنچه در پیش‌رو خواهید دید، بهتر بود.

ضمناً باید اعلام کنم چون داده‌ها رگرسیونی هستند و نیاز به تابع فعال‌ساز دو دویی ندارند نیازی به تجانس داده‌ها در بازه ۰ و ۱ نیست. (توجه داشته باشید که داده‌ها از قبل یکبار نرمالیده شده‌اند.) [۱۷]

```
#Model training based on random hyper
    parameters
MSE_val_nn<-c()
HL<-list()
LR<-c()
activation_function<-c('logistic','tanh')
algorithm<-c('backprop','rprop+', 'rprop-','
    sag','slr')
AF<-c()
AL<-c()
for(i in 1:300){
    print(i)
    LR[i]<-runif(1,0,1)
    AF[i]<-floor(runif(1,1,3))
    AL[i]<-floor(runif(1,1,6))
    hidden_layer_num<-floor(runif(1,1,6))
    hidden_layer<-floor(runif(hidden_layer_
    num,3,16))
    hidden_layer<-hidden_layer[order(hidden_
    layer,decreasing = TRUE)]
    nn<-neuralnet(Price~. ,data=train.data,
                hidden = hidden_layer,
                linear.output = F,
                lifesign = 'full',
                rep=1,
                threshold=0.05,
                learningrate=LR[i],
                startweights="NULL",
                algorithm=algorithm[AL[i]],
                act.fct=activation_function[
    AF[i]],
                stepmax=10^5)
    y_pred_nn<-compute(nn, valid.data[,1:6])
    MSE_val_nn[i]<-MSE(y_pred_nn$net.result,
    valid.data[,7])
    HL[i]<-hidden_layer
}
```

از آنجایی که شبکه عصبی به راحتی دچار بیش برازش می‌شود من ۳۰۰ بار الگوریتم را با بردارهایی که نمایانگر لایه پنهان[۷۸]های مختلف و تصادفی (چه از نظر تعداد گره‌ها چه از نظر تعداد لایه‌ها) هستند اجرا کردم. همچنین از دیگر فراپارامترهای مختلف اعم از توابع فعال‌ساز و نرخ یادگیری[۷۹]های تصادفی و ... در هر حلقه استفاده کرده‌ام. در کمترین خطایی که روی مجموعه اعتبار سنجی حاصل شد را به عنوان مدل نهایی در نظر گرفتم.

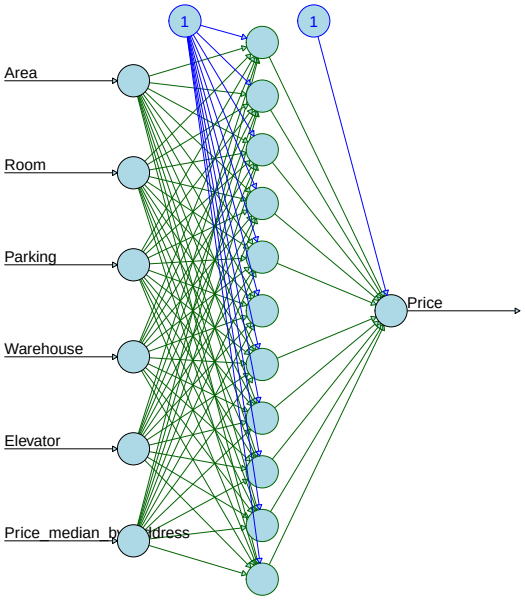
```
> best_i<-where.min[MSE_val_nn]
>HL[best_i] #To display the best hiddenlayer
     vector
[[1]]
[1] 11
>activation_function[AF[best_i]] #To display
     the best Activation Function
[1] "tanh"
>algorithm[AL[best_i]] #To display the best
    Algorithm
[1] "rprop+"
>LR[best_i] #To display the best Learning
    Rate
[1] 0.7052665
>MSE_val_nn[best_i]
[1] 0.695432
```

در نتیجه، میانگین توان دوم خطاهای بهترین مدل شبکه عصبی روی مجموعه اعتبار سنجی ۰٫۶۹۵۴۳۲ است. البته با توجه به تعداد کم داده‌ها نباید انتظار زیادی از این الگوریتم داشت. چرا که توان پیشگویی شبکه‌های عصبی روی داده‌های زیاد است.

به طور کلی یکی از ضعف‌های شبکه‌های عصبی این است که هنگامی که تعداد داده‌ها زیاد نیست نمی‌توانند به خوبی پیچیدگی داده‌ها را درک کنند.

نکته دیگر این است که در فصل شبکه‌های عصبی مرجع اصلی این پروژه نیز گفته شده بود معمولاً یک لایه شبکه عصبی برای درک پیچیدگی داده‌ها مناسب است. من ادعا نمی‌کنم شبکه عصبی نوشته شده پیچیدگی داده‌ها را درک کرده است. چرا که شاهد

[74] Deep Learning
[75] Neural Network
[76] Activation Function
[77] Python
[78] Hidden Layer
[79] Learning Rate

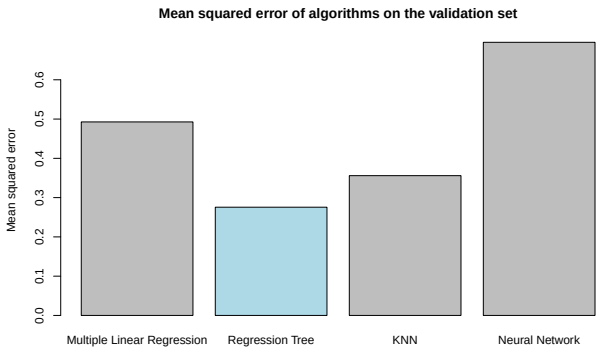


شکل ۲۸: دیاگرام شبکه عصبی منتخب

دقت مناسبی از مدل نهایی شبکه عصبی نیستیم. اما با توجه به جستجوی تصادفی انجام شده برای من جالب و حقیقتاً شگفت انگیز بود که شبکه عصبی یک لایه‌ای از بین ۳۰۰ مدل مختلف شبکه عصبی به عنوان بهترین مدل انتخاب شده بود. به شخصه به هیچ وجه این پیش‌بینی را نمی‌کردم.

انتخاب مدل نهایی

"یک تصویر به هزار کلمه می‌ارزد" این جمله برگرفته از مرجع اصلی است. من از یک نمودار میله‌ای[80] برای تصویری سازی و مقایسه میزان میانگین توان دوم خطاهای مدل‌های منتخب در هر فصل پروژه کمک گرفتم. به‌وضوح بهترین دقت مربوط به الگوریتم درخت رگرسیون بوده است.



شکل ۲۹: مقایسه میانگین توان دوم خطاهای مدل‌ها

حال وقت آن رسیده است که دقت مدل متنخب نهایی را روی مجموعه تست بررسی کنیم. مبنای ما در این پروژه میانگین توان دوم خطاها است اما میانگین قدرمطلق خطاها[81] را نیز برای تجسم بهتر مخاطب محاسبه کرده‌ام.

```
> MSE(predict(best_dt_model, newdata =test.
    data),test.data[,7])
[1] 0.1228406
> MAE(predict(best_dt_model, newdata =test.
    data),test.data[,7])
[1] 0.1797792
```

نتیجه از تصور شخصی من بهتر بود. میانگین توان دوم خطاهای ما روی مجموعه تست حتی از مجموعه اعتبار سنجی هم بهتر بود. در واقع میانگین توان دوم خطاها در مجموعه تست نهایی کمتر از نصف مجموعه اعتبار سنجی شد.

لذا خطای نهایی مدل پیاده‌سازی شده (بر اساس تابع خطای انتخاب شده) برابر ۰٫۱۷۹۷۷۹۲ است.

پیاده‌سازی روی داده‌های واقعی خارج از مجموعه داده‌ها

تا اینجا پروژه ما به سرانجام رسیده است اما من دوست دارم دو عدد ثبت آزمایشی را که خارج از مجموعه داده‌هاست را به کمک مدل منتخب قیمت‌گذاری کنم. دو ثبت من به شرح زیر است

۱. خانه‌ای ۷۵ متری ۲ سال ساخت در منطقه نیاوران دارای ۲ خواب، مجهز به پارکینگ انباری و آسانسور که در تاریخ مذکور حدود ۷ میلیارد و ۵۰۰ میلیون تومان قیمت‌گذاری شده بود.

۲. خانه‌ای ۸۲ متری در منطقه شهرزیبا، با قدمت ۱۰ سال در تاریخ مذکور، دارای ۲ خواب، مجهز به پارکینگ انباری و آسانسور که ۲ میلیارد و ۸۰۰ میلیون تومان در تاریخ مذکور قیمت‌گذاری شده بود.

این مجموعه کوچک داده ها با توجه به نرمال‌سازی انجام شده روی داده های مجموعه آموزشی، نرمالیده شده و مدل نهایی روی آنها اعمال شد. مورد اول به مبلغ حدوداً ۷ میلیارد تومان، و خانه دوم به مبلغ حدوداً ۳ میلیارد تومان توسط مدل نهایی پیشگویی شد.

```
> MSE(predict(best_dt_model, newdata =kh),kh
    [,7])
[1] 0.0034291
```

همچنین خطای این مدل روی داده‌های جدید نسبتاً پایین بوده است.

فرایند انتخاب مدل

فرایند انتخاب مدل من بر اساس صفحات ۱۵۰ و ۱۵۱ کتاب Understanding Machine Learning from theory to algorithms نوشته شای شالو شوارتز و شای بن داوید بود. [۱۸]

با توجه به اینکه در آموزش دادن همه مدل‌ها از مجموعه اعتبارسنجی بهره بردیم، لذا اعتبار سنجی نهایی را روی مجموعه تست انجام دادم. خوشبختانه همان‌طور که مشخص است دچار بیش برازش نشدم. ضمناً این کتاب برای مواقعی که دقت ما روی مجموعه تست به شکل مشهودی کمتر از مجموعه اعتبار سنجی باشد راهکارهای خوب و مفصلی ارائه می‌دهد.

[80] Bar Chart
[81] Mean Absolute Error

جدول کدگذاری متغیر رسته‌ای

Label	Address String	Encoded by target
۱	Other	−۰٫۳۲۳۲۸۲۰۲
۲	Punak	−۰٫۲۹۵۵۸۸۱۷۸
۳	Pardis	−۰٫۵۸۵۲۷۱۷۷
۴	West Ferdows Boulevard	−۰٫۳۴۹۷۰۴۸۹۳
۵	Gheitarieh	۰٫۴۶۰۱۳۵۸۲۹
۶	Shahran	−۰٫۲۷۹۶۷۱۴۹۷
۷	Saadat Abad	۰٫۴۱۴۲۹۵۷۸۸
۸	Parand	−۰٫۶۱۹۶۵۱۸۰۱
۹	Shahr-e-Ziba	−۰٫۳۰۲۵۹۱۵۱۸
۱۰	Southern Janatabad	−۰٫۲۸۹۲۲۱۵۰۶
۱۱	Central Janatabad	−۰٫۲۲۲۳۷۱۴۴۶
۱۲	Jeyhoon	−۰٫۴۷۷۰۳۸۳۴
۱۳	Persian Gulf Martyrs Lake	−۰٫۲۵۹۲۹۸۱۴۶
۱۴	Andisheh	−۰٫۵۶۱۸۷۴۲۴۹
۱۵	Ostad Moein	−۰٫۴۶۶۲۱۴۹۹۷
۱۶	East Ferdows Boulevard	−۰٫۳۱۱۵۰۴۸۵۹
۱۷	Shahrake Qods	−۰٫۵۶۰۴۴۱۷۴۸
۱۸	Niavaran	۱٫۷۸۵۹۹۵۳۴۵
۱۹	Pasdaran	۰٫۵۹۲۵۶۲۶۱۴
۲۰	Pirouzi	−۰٫۴۴۵۲۰۴۹۷۸
۲۱	Salsabil	−۰٫۵۲۷۹۷۱۷۱۹
۲۲	Shahrake Gharb	۰٫۵۸۹۸۸۸۶۱۱
۲۳	Farmanieh	۱٫۶۷۷۷۶۱۹۱۵
۲۴	Heravi	۰٫۲۲۸۳۸۸۹۵۶
۲۵	Ekhtiarieh	۰٫۲۷۴۲۲۸۹۹۷
۲۶	Islamshahr	−۰٫۵۲۱۶۰۵۰۴۷
۲۷	Feiz Garden	−۰٫۲۷۰۱۲۱۴۸۹
۲۸	Yousef Abad	۰٫۱۱۵۰۶۲۱۸۸
۲۹	Northern Janatabad	−۰٫۲۱۶۰۰۴۷۷۴
۳۰	Qasr-od-Dasht	−۰٫۵۰۵۶۸۸۳۶۶
۳۱	North Program Organization	−۰٫۱۳۴۵۱۱۳۶۸
۳۲	Zaferanieh	۱٫۶۷۱۷۱۳۵۷۶
۳۳	Aqdasieh	۱٫۱۲۷۳۶۳۰۹۱
۳۴	Beryanak	−۰٫۵۴۰۷۰۵۰۶۴
۳۵	Narmak	−۰٫۲۷۹۶۷۱۴۹۷
۳۶	Pakdasht	−۰٫۵۵۷۲۵۸۴۱۲
۳۷	Azarbaijan	−۰٫۵۰۸۸۷۱۷۰۲
۳۸		−۰٫۳۰۸۳۲۱۵۲۳
۳۹	Abazar	−۰٫۰۹۵۰۳۷۹۹۹
۴۰	Damavand	−۰٫۴۵۷۹۳۸۳۲۳
۴۱	Si Metri Ji	−۰٫۴۶۴۳۰۴۹۹۵
۴۲	Southern Program Organization	−۰٫۳۳۳۱۵۱۵۴۵
۴۳	Tenant	−۰٫۴۹۹۳۲۱۶۹۳
۴۴	Marzdaran	۰٫۴۱۴۲۹۵۷۸۸
۴۵	Velenjak	۱٫۷۳۲۱۹۶۹۶۴
۴۶	Karoon	−۰٫۵۲۷۹۷۱۷۱۹
۴۷	Jordan	۰٫۶۶۱۶۴۱۰۰۹
۴۸	Elahieh	۰٫۹۶۵۰۱۲۹۴۶
۴۹	Golestan	−۰٫۵۳۷۵۲۱۷۲۷
۵۰	Kahrizak	−۰٫۵۵۹۰۰۹۲۴۷
۵۱	Northern Chitgar	−۰٫۵۵۶۶۲۱۷۴۵
۵۲	Mirdamad	۰٫۲۸۰۴۰۴۶۶۹
۵۳	Amirabad	−۰٫۱۲۰۵۰۴۶۸۹
۵۴	Kamranieh	۲٫۵۶۶۲۳۱۰۴۱
۵۵	Northren Jamalzadeh	−۰٫۳۳۶۹۷۱۵۴۹
۵۶	Dorous	۰٫۶۰۵۲۹۵۹۵۹
۵۷	Hashemi	−۰٫۵۴۵۷۹۸۴۰۲
۵۸	Shahryar	−۰٫۵۵۹۸۰۵۰۸۱
۵۹	Amirieh	−۰٫۴۴۰۶۸۴۶۴۱
۶۰	Sattarkhan	−۰٫۰۴۴۱۰۴۶۲۱
۶۱	Komeil	−۰٫۵۲۹۲۴۵۰۵۳
۶۲	Qalandari	۰٫۲۲۷۱۱۵۶۲۱
۶۳	Qazvin Imamzadeh Hassan	−۰٫۴۷۳۴۰۹۳۳۷
۶۴	Railway	−۰٫۳۰۳۸۶۴۸۵۲
۶۵	Rudhen	−۰٫۴۶۸۷۶۱۶۶۶
۶۶	West Pars	۰٫۱۷۹۳۶۵۵۷۹
۶۷	Air force	−۰٫۳۴۹۷۰۴۸۹۳
۶۸	Gholhak	۰٫۲۹۳۳۲۹۰۱۴
۶۹	Ozgol	−۰٫۰۳۱۳۷۱۲۷۶
۷۰	Zafar	۰٫۰۱۳۱۹۵۴۳

تقدیر و تشکر

بدین‌وسیله در وهله اول از استاد محترم درس، جناب آقای دکتر فقیهی بابت آموختن درس داده‌کاوی، تعریف پروژه و همه راهنمایی‌هایشان در طول پروژه تشکر و قدردانی می‌نمایم. باشد تا برداشت صحیحی از آنچه ایشان به ما آموختند داشته باشم.

همچنین از دوستان خوبم آقایان دکتر بهراد تقی بیگ‌لو و سهراب فریدی که من را در به سرانجام رساندن این پروژه راهنمایی کردند نهایت سپاسگزاری را دارم.

مراجع

[1] گالیت شمولی، پیترسی بروس، اینبال یاهاو، نیتین‌آر پاتل، کنت‌سی داده‌کاوی برای تحلیل خودکار کسب‌وکار: مفاهیم، فنون .لیختندال .و کاربردهای R

[2] I. M. Society, "واژه نامه ریاضی انجمن ریاضی ایران," September 2022.

[3] S. Research and T. Center, "واژه نامه و اصطلاحات انجمن آمار ایران," September 2022.

[4] M. Kariminejad., "House price," December 2021.

[5] M. Kabari, "مشاوران املاک، تنظیم گر بازار یا حباب ساز قیمت مسکن؟," September 2021.

[6] F. E. Grubbs, "Procedures for detecting outlying observations in samples," *Technometrics*, vol.11, no.1, pp.1–21, 1969.

[7] geetansh044., "How to normalize data in r?," December 2021.

[8] P. Cerda, G. Varoquaux, and B. Kégl, "Similarity encoding for learning with dirty categorical variables," *Machine Learning*, vol.107, no.8, pp.1477–1494, 2018.

[9] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. New York: Springer, fourth ed. , 2002. ISBN 0-387-95457-0.

[10] چرا بازار مسکن منطقه ۵ تهران داغ ترین بازار .وبسایت املاک‌باشی املاک است؟. www.amlakbashi.com, 1399.

[11] wikipedia., "Heatmap," December 2021.

[12] S. T. Roweis and L. K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science*, vol.290, no.5500, pp.2323–2326, 2000.

[13] P. Pudil and J. Novovičová, *Novel Methods for Feature Subset Selection with Respect to Problem Knowledge*, pp.101–116. Boston, MA: Springer US, 1998.

[14] wikipedia., "Principal component analysis," May 2022.

[15] K. Jain., "Linkedin profile," May 2020.

[16] D. A. Pierce and D. W. Schafer, "Residuals in generalized linear models," *Journal of the American Statistical Association*, vol.81, no.396, pp.977–986, 1986.

[17] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, *Efficient BackProp*, pp.9–48. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[18] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, 2014.

واژه‌نامه انگلیسی به فارسی

A

Activation Function . . . . . تابع فعال ساز

Area . . . . . مساحت

B

Backward . . . . . پسرو

Chart Bar . . . . . نمودار میله‌ای

Binary . . . . . دودویی

Boxplot . . . . . نمودار جعبه‌ای

C

Categorical . . . . . رسته‌ای

Comment . . . . . توضیح

Compiler . . . . . مترجم

Complexity Parameter . . . . . پیچیدگی پارامتر

Coordinate . . . . . مختصات

Correlation . . . . . همبستگی

D

Data . . . . . داده

Data Leakage . . . . . نشت داده

Data Reduction . . . . . داده کاهی

Data Visualization . . . . . تصویرسازی داده‌ها

Decision Tree . . . . . درخت تصمیم

Deep Learning . . . . . یادگیری عمیق

Dimension Reduction . . . . . کاهش بُعد

Dispersion . . . . . پراکندگی

Distribution . . . . . توزیع

Dummy Variable . . . . . متغیّر ظاهری

F

Feature . . . . . ویژگی

Forward . . . . . پیشرو

Frequency . . . . . فراوانی

G

Grid Search . . . . . جستجوی شبکه‌ای

H

Heatmap . . . . . نمودار حرارتی

Hidden Layer . . . . . لایه پنهان

Histogram . . . . . بافت‌نگار

Hyperparameter . . . . . فرا پارامتر

Hyperparameter Optimization . . . . . بهینه‌سازی فرا پارامتر

I

Imbalance . . . . . نامتعادل

Integer . . . . . عدد صحیح

K

K-Nearest Neighbor . . . . . K نزدیکترین همسایه

L

Learning Rate . . . . . نرخ یادگیری

M

Machine Learning . . . . . یادگیری ماشین

Matrix . . . . . ماتریس

Mean . . . . . میانگین

Mean Absolute Error . . . . . میانگین قدرمطلق خطاها

R

Random ..... تصادفی

Real Number ..... عدد حقیقی

Record ..... ثبت

Regression ..... رگرسیون

Regression Tree ..... درخت رگرسیون

Residual ..... مانده

Response Variable ..... متغیر پاسخ

S

Scatter Diagram ..... نمودار پراکنش

Side-By-Side Boxplot ..... نمودار جعبه‌ای پهلوبه‌پهلو

Stepwise ..... گام‌به‌گام

String ..... رشته

Summary ..... خلاصه

Supervised ..... راهنماییده

T

Test Set ..... مجموعه آزمون

Training Set ..... مجموعه آموزشی

Tuning ..... تنظیم

V

Validation Set ..... مجموعه اعتبار سنجی

Variance ..... واریانس

Visualization ..... تصویری سازی

Mean Squared Error ..... میانگین توان دوم خطاها

Median ..... میانه

Missing Value ..... مقدار گم‌شده

Multiple Linear Regression ..... رگرسیون خطی چندگانه

N

Neural Network ..... شبکه عصبی

Normal Distribution ..... توزیع نرمال

Normalized ..... نرمالیده

Null ..... تهی

O

Outlier ..... دورافتاده

P

Partition ..... افراز

Percentile ..... صدک

Positive Skewness ..... چولگی مثبت

Predictor ..... پیشگو

Preprocessing ..... پیش‌پردازش

Principal Component Analysis ... تجزیه و تحلیل مؤلفه اصلی

P-Value ..... پی-مقدار

Python ..... پایتون

Q

QQ Plot ..... نمودار چندک چندک

Quartile ..... چارک

واژه‌نامه فارسی به انگلیسی

افراز . . . . . Partition

بافت‌نگار . . . . . Histogram

بهینه‌سازی فرا پارامتر . . . . . Hyperparameter Optimization

تابع فعال ساز . . . . . Activation Function

تجزیه و تحلیل مؤلفه اصلی . . . Principal Component Analysis

تصادفی . . . . . Random

تصویرسازی داده‌ها . . . . . Data Visualization

تصویری سازی . . . . . Visualization

تنظیم . . . . . Tuning

تهی . . . . . Null

توزیع . . . . . Distribution

توزیع نرمال . . . . . Normal Distribution

توضیح . . . . . Comment

ثبت . . . . . Record

جستجوی شبکه‌ای . . . . . Grid Search

خلاصه . . . . . Summary

داده . . . . . Data

داده کاهی . . . . . Data Reduction

درخت تصمیم . . . . . Decision Tree

درخت رگرسیون . . . . . Regression Tree

دودویی . . . . . Binary

دورافتاده . . . . . Outlier

راهنماییده . . . . . Supervised

رسته‌ای . . . . . Categorical

رشته . . . . . String

رگرسیون . . . . . Regression

رگرسیون خطی چندگانه . . . . . Multiple Linear Regression

شبکه عصبی . . . . . Neural Network

صدک . . . . . Percentile

عدد حقیقی . . . . . Real Number

عدد صحیح . . . . . Integer

فرا پارامتر . . . . . Hyperparameter

فراوانی . . . . . Frequency

لایه پنهان . . . . . Hidden Layer

ماتریس . . . . . Matrix

مترجم . . . . . Compiler

متغیر پاسخ . . . . . Response Variable

متغیّر ظاهری . . . . . Dummy Variable

مجموعه آزمون . . . . . Test Set

مجموعه آموزشی . . . . . Training Set

مجموعه اعتبار سنجی . . . . . Validation Set

مختصات . . . . . Coordinate

مساحت . . . . . Area

مقدار گم‌شده . . . . . Missing Value

میانه . . . . . Median

میانگین . . . . . Mean

میانگین توان دوم خطاها . . . . . Mean Squared Error

میانگین قدرمطلق خطاها . . . . . Mean Absolute Error

نامتعادل . . . . . Imbalance

نرخ یادگیری . . . . . Learning Rate

نرمالیده . . . . . Normalized

نشت داده . . . . . Data Leakage

نمودار جعبه‌ای . . . . . Boxplot

نمودار جعبه‌ای پهلوبه‌پهلو . . . . . Side-By-Side Boxplot

نمودار حرارتی . . . . . Heatmap

نمودار پراکنش . . . . . Scatter Diagram

نمودار چندک چندک . . . . . QQ Plot

همبستگی . . . . . Correlation

واریانس . . . . . Variance

ویژگی . . . . . Feature

پایتون . . . . . Python

پراکندگی . . . . . Dispersion

پسرو . . . . . Backward

پیشرو . . . . . Forward

پیشگو . . . . . Predictor

پیش‌پردازش . . . . . Preprocessing

پی-مقدار . . . . . P-Value

پیچیدگی پارامتر . . . . . Complexity Parameter

یادگیری عمیق . . . . . Deep Learning

یادگیری ماشین . . . . . Machine Learning

K نزدیکترین همسایه . . . . . K-Nearest Neighbor

چارک . . . . . Quartile

چولگی مثبت . . . . . Positive Skewness

کاهش بُعد . . . . . Dimension Reduction