گزارش تمرین شماره ۱

Boston house prices dataset

گردآوری : **علی شریفی**

شماره دانشجویی : **۹۵۲۲۲۰۴۶**

۱ مقدمه و مسئله :

این تمرین در راستای آشنایی با مفاهیم مدل های خطی همانند رگرسیون خطی داده شده بود . دیتاست مورد استفاده در این تمرین مربوط به داده های قیمت خانه و پارامتر هایی که ممکن است در قیمت خانه تاثیر گذار هستند در شهر بوستون ، آمریکا می باشد . این دیتاست شامل اطلاعات مربوط به ۵۰۶ خانه می باشد . هر خانه شامل ۱۴ رکورد است که این رکورد ها به ترتیب عبارتند از :

- CRIM :
  per capita crime rate by town
- ZN :
  proportion of residential land zoned for lots over 25,000 sq.ft
- INDUS:
  proportion of non-retail business acres per town
- CHAS :
  Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
- NOX:
  nitric oxides concentration (parts per 10 million)
- RM:
  average number of rooms per dwelling
- AGE :
  proportion of owner-occupied units built prior to 1940
- DIS:
  weighted distances to five Boston employment centres
- RAD:
  index of accessibility to radial high-ways
- TAX :
  full-value property-tax rate per $10,000
- PTRATIO :
  pupil-teacher ratio by town
- B :
  $1000 \times (B_k - 0.63)^2$ where B_k is the proportion of blacks by town
- LSTAT :
  % lower status of the population
- MEDV :
  Median value of owner-occupied homes in $1000's

رکورد MEDV همان رکورد هدف ما می باشد یعنی ما میخواهیم با ساخت بهترین مدل خطی ممکن با ۱۳ پارامتر دیگر تخمین دقیقی از رکورد MEDV داشته باشیم . توزیع متغیر هدف در دیتاست درشکل (۱) نمایش داده شده است .

۲ راه حل و ایده ها :

در ابتدا با حذف تاثیر نویز در داده ها ، داده ها را با استفاده از روش mean normalize ، نرمال میکنیم . حال به

سراغ محاسبه تابع هزینه [1] با استفاده از معادله زیر می رویم :

$$h = (w^T X + b) \qquad (۱)$$

که در معادله (۱) ، w ماتریس وزن ها برای هریک از پارامتر ها و b ، بایاس مسئله می باشد .
حال تابع هزینه را به صورت زیر محاسبه میکنیم و هدف کلی ما کاهش مقدار تابع هزینه است که در عمل مسئله مطرح شده را به یک مسئله بهنیه سازی تبدیل میکند .

$$J = -\frac{1}{m}\sum_{i=1}^{m}(y^{(i)} - h^{(i)})^2 \qquad (۲)$$

هم چنین دو فرمول زیر را هم خواهیم داشت :

$$\frac{\partial J}{\partial w} = \frac{-2}{m}X(y-h) \qquad (۳)$$

$$\frac{\partial J}{\partial b} = \frac{-2}{m}\sum_{i=1}^{m}(y^{(i)} - h^{(i)}) \qquad (۴)$$

حال با استفاده از روش گرادیان کاهشی خواهیم داشت :
در هر iteration ، مقادیر w و b را بروز رسانی میکنیم تا تابع هزینه به کمترین حالت خود دست یابد .

$$w = w - \alpha \times \frac{\partial J}{\partial w} \qquad (۵)$$

$$b = b - \alpha \times \frac{\partial J}{\partial b} \qquad (۶)$$

در فرمول (۵) و(۶) ، α ، ضریب یادگیری [2] می باشد . با توجه به همگرا بودن روش گرادیان کاهشی انتظار خواهیم داشت تا حد امکان به نقطه اکسترمم تابع هزینه نزدیک شویم .
برای حل مسئله دو رویکرد متفاوت به کار گرفته میشود :

۱. رویکرد ساخت مدل با پارامتر های کمتر با استفاده از ۴۰۰ رکورد

۲. رویکرد ساخت مدل با تمامی پارامتر های مطرح شده با استفاده از ۴۰۰ رکورد

۳. رویکرد ساخت مدل با تمامی پارامتر های مطرح شده با استفاده از ۵۰۰ رکورد

در رویکرد شماره ۱ با محاسبه ، رابطه خطی بین متغیر بیان شده در مسئله ، تنها متغیر هایی را وارد مسئله میکنیم که رابطه خطی نسبتا قابل توجه ای را دارا می باشند . که انتظار داریم در اکثر مسائل این رویکرد نتایج ضعیف تری نسبت به رویکرد ۲ ارایه کند . همان طورکه در شکل (۲) مشاهده میشود دو متغیر LSTAT و RM به ترتیب با -0.74 و 0.69 بیشترین رابطه خطی را با متغیر هدف یعنی MEDV را دارا می باشند .

۳ ارزیابی نتیجه ها :

برای محاسبه دقت مدل ها از معادله زیر استفاده شده است :

$$100-(\frac{\sum_{i=1}^{n}|Y_{prediction-train} - Y_{train}|}{n})*100 \qquad (۷)$$

پس از ساخت مدل به رویکرد شماره (۱) به پارامتر های زیر برای مدل خطی خود رسیدیم .

```
2.7362403426066173
[-0.71722954  4.58938833]
```

که به ترتیب بایاس مسئله و ضرایب متغیر های LSTAT ، RM می باشند . مدل خطی ما به صورت زیر قابل نوشتن است :

$$\begin{aligned} &MEDV_{Predict} = \\ &2.7362403426066173+ \\ &(-0.71722954) \times LSTAT+ \\ &(4.58938833) \times RM \end{aligned} \qquad (۸)$$

که دقت این مدل بر روی داده های test و train به ترتیب برابر است با

```
train accuracy:
63.007451493317014 %
test accuracy:
66.28996975186952 %
```

حال به بررسی رویکرد ۲ یعنی استفاده از تمامی متغیر ها برای ساخت مدل می پردازیم و خواهیم داشت :
ضرایب متغیر ها :

[1] Cost Function

```
[[-0.02499207],
[ 0.07736133],
[-0.01024326],
[ 0.        ],
[-0.15992907],
[ 0.44881242],
[ 0.04889012],
[-0.29692406],
[ 0.        ],
[ 0.        ],
[-0.20748684],
[ 0.0726322 ],
[-0.49842745]]
```

بایاس :

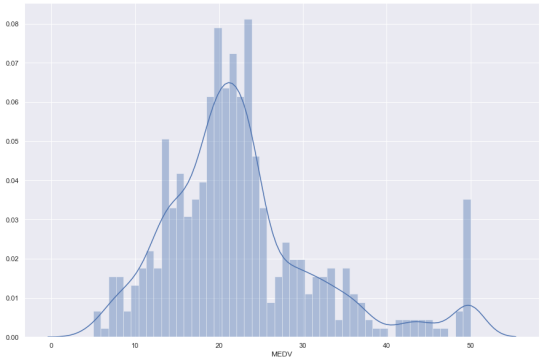
```
[0.00399818]
```

و طبق پیش بینی مشاهده خواهیم کرد این مدل ساخته شده دارای دقت بالاتری نسبت به مدل رویکرد شماره ۱ می باشد . دقت این مدل بر روی داده های test و train به ترتیب برابر است با

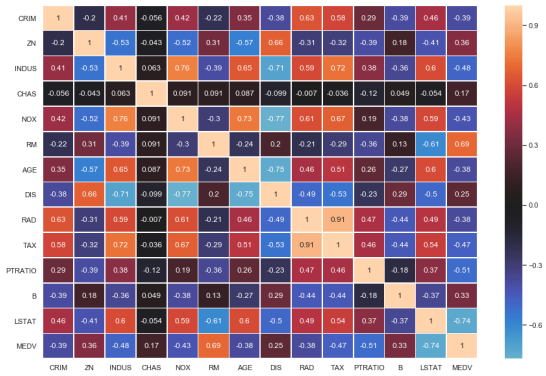
```
train accuracy:
92.4751187277718 %
test accuracy:
91.98427809498112 %
```

با افزایش متغیر ها از ۴۰۰ به ۵۰۰ در رویکرد شماره ۳ و با استفاده از تمامی متغیرها مشاهده میکنیم که مدل به دقت بهتری دست خواهد یافت و داریم :
ضرایب متغیر ها :

```
[[-0.02499207],
[ 0.07736133],
[-0.01024326],
[ 0.        ],
[-0.15992907],
[ 0.44881242],
[ 0.04889012],
[-0.29692406],
[ 0.        ],
[ 0.        ],
[-0.20748684],
[ 0.0726322 ],
[-0.49842745]]
```



شکل ۱: توزیع متغیر هدف در دیتاست



شکل ۲: ماتریس کورولیشن بین متغیرها

بایاس :

```
[7.89615721e-05]

train accuracy:
92.55582193832844 %
test accuracy:
96.00395275921642 %
```

۴ جمع بندی و نتیجه گیری

همان طور که مشاهده شد با افزایش متغیر ها در مدل خطی توانستیم به صورت قابل ملاحظه ای مدل خود را ارتقا دهیم .هم چنین نیز تاثیر افزایش تعداد رکورد های مورد استفاده در مدل نیز قابل توجه است . به دلیل over fit شدن مدل به دلایل مختلف از قبیل کم بودن داده ها شاید مدل کارآرایی خود را در مواجه با داده های جدید از دست دهد که برای جلوگیری از این کار توصیه میشود که از روش هایی از قبیل cross validation استفاده شود . هم چنین انتظار میرود که با افزایش تعداد داده ها، مدل به طور قابل توجه ای ارتقا یابد .