

1. به جز فیچرهای نامبرده و اجاره تمام فیچرهای دیگر را دراپ می کنیم. **Preprocessing** را که شامل حذف کردن دیتای پرت اسکیل کردن و .. است انجام می دهیم. همچنین **heating** **type** که شامل مدل های زیادی است را ساده تر میکنیم تا با **get_dummies** ستون های زیادی به دیتافریم اضافه نشود. سپس آن را از حالت کتگوریکال خارج کرده و نومریکال می کنیم.

در ادامه باید مدل رگرسیون خطی با میانگین مربع خطاها را پیاده سازی کنیم. این مدل باید توابع فیت و پردیکت را داشته باشد. هر کدام را با توجه به الگوریتم ریاضی آن پیاده سازی میکنیم.

در آخر آنرا را با دیتایی که در دست داریم آموزش می دهیم. سپس **r2_score** و **mse** آنرا محاسبه می کنیم.

```
from sklearn import metrics
metrics.r2_score(y_test, y_pre_mse)

0.38759939728490267

[219] metrics.mean_squared_error(y_test, y_pre_mse)

156736.7871010926
```

درباره **r2_score**:

بهترین امتیاز ممکن 1.0 است و می تواند منفی باشد (زیرا مدل می تواند خودسرانه بدتر باشد). یک مدل ثابت که همیشه مقدار مورد انتظار **y** را پیش بینی می کند، بدون توجه به ویژگی های ورودی، امتیاز 0.0 را دریافت می کند.

2. برای مقایسه مدل پیاده سازی شده با **linear regression** از پکیج **sklearn** مدلی از آن میسازیم و آموزش میدهیم.

```
[224] metrics.r2_score(y_test, y_pre_lr)

0.4209749490045015

[225] metrics.mean_squared_error(y_test, y_pre_lr)

148194.7041556094
```

میبینیم مدلی که از **sklearn** ساختیم از مدل **linear regression** با **mse** بهتر آموزش داده شده است.

3. برای انجام این بخش نیاز به پیش پردازش های زیادی است که در پروژه قبلی انجام شده و از آنها استفاده میکنیم.



The image shows a Jupyter Notebook interface with a sidebar on the left containing icons for search, variables, files, expand/collapse, table of contents, and console. The main area displays two sections: 'a) Ridge' and 'b) Lasso'. Each section contains a code cell with Python code and a corresponding output cell showing the R-squared score.

```
▼ a) Ridge
```

```
[320] ridge = Ridge(alpha=1)
      ridge.fit(X_train, y_train)
      y_pre_ri = ridge.predict(X_test)
```

```
[321] metrics.r2_score(y_test, y_pre_ri)

0.5535044428178639
```

```
▼ b) Lasso
```

```
[322] lasso = Lasso(alpha=0.1)
      lasso.fit(X_train, y_train)
      y_pre_la = lasso.predict(X_test)
```

```
[323] metrics.r2_score(y_test, y_pre_la)

0.5537851596018275
```

مدل های ridge و lasso را نیز با استفاده از دیتای پیش پردازش شده آموزش می دهیم. میبینیم که lasso از بین ۴ مدل آموزش داده شده، بهتر عمل کرده است.