regression-pr-02

January 4, 2024

[65]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[66]:
```python
df=pd.read_csv('../08-Linear-Regression-Models/Advertising.csv')
```

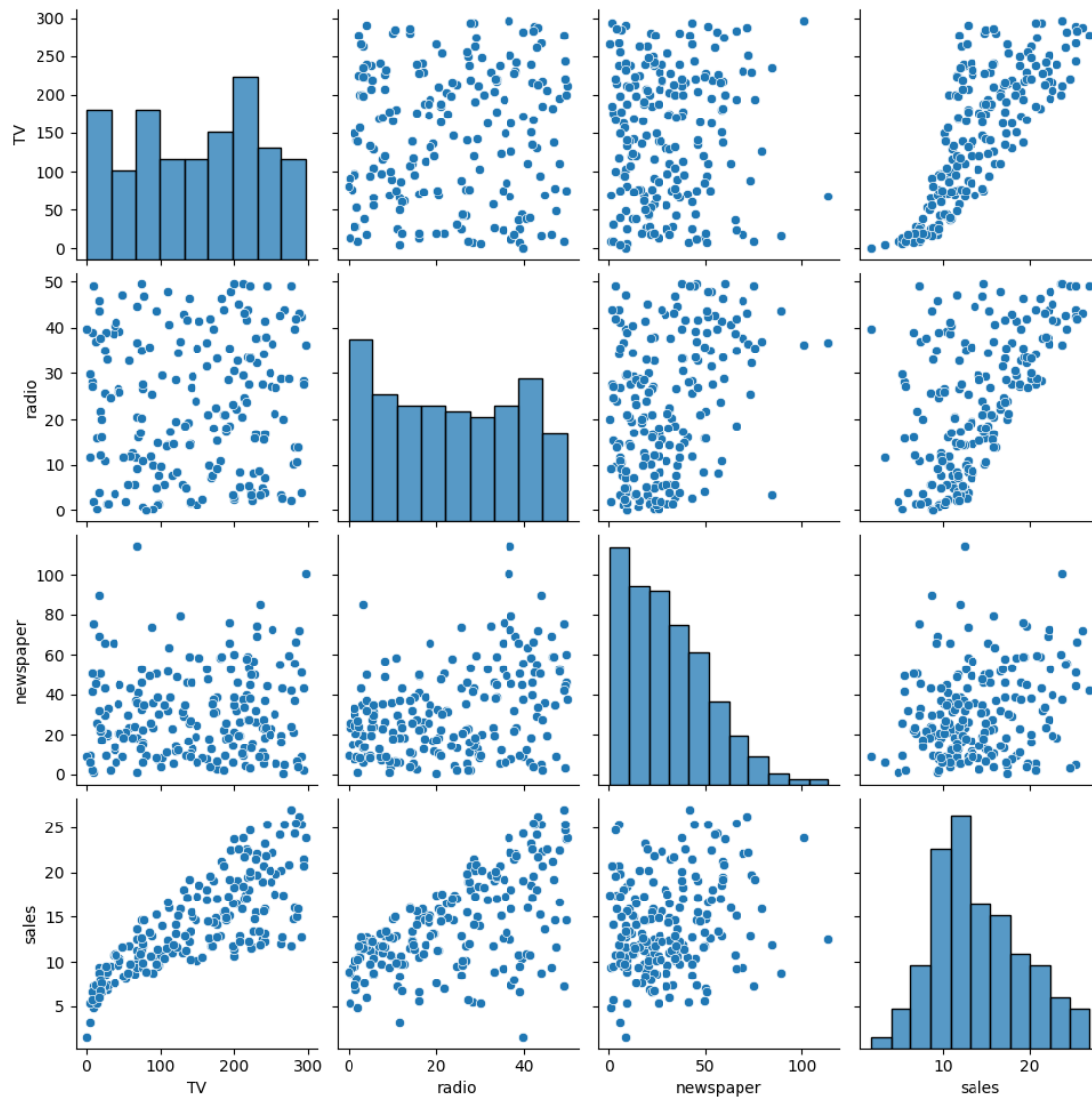[67]:
```python
df.head()
```

[67]:
```
      TV  radio  newspaper  sales
0  230.1   37.8       69.2   22.1
1   44.5   39.3       45.1   10.4
2   17.2   45.9       69.3    9.3
3  151.5   41.3       58.5   18.5
4  180.8   10.8       58.4   12.9
```

[68]:
```python
sns.pairplot(df)
```

[68]: <seaborn.axisgrid.PairGrid at 0x7fc371d54520>

```
[69]: X=df.drop('sales',axis=1)
```

```
[70]: y=df['sales']
```

```
[71]: y
```

```
[71]: 0      22.1
      1      10.4
      2       9.3
      3      18.5
      4      12.9
             ...
      195     7.6
```

```
196      9.7
197     12.8
198     25.5
199     13.4
Name: sales, Length: 200, dtype: float64
```

[72]:
```python
from sklearn.model_selection import train_test_split
```

[73]:
```python
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.
 ↪2,random_state=42)
```

[74]:
```python
len(df)
```

[74]: 200

[75]:
```python
len(X_train)
```

[75]: 160

[76]:
```python
from sklearn.linear_model import LinearRegression
```

[77]:
```python
model=LinearRegression()
```

[78]:
```python
model.fit(X_train,y_train)
```

[78]: LinearRegression()

[79]:
```python
test_pred=model.predict(X_test)
```

[80]:
```python
from sklearn.metrics import mean_absolute_error,mean_squared_error
```

[81]:
```python
df['sales'].mean()
```

[81]: 14.0225

[82]:
```python
mean_absolute_error(y_test,test_pred)
```
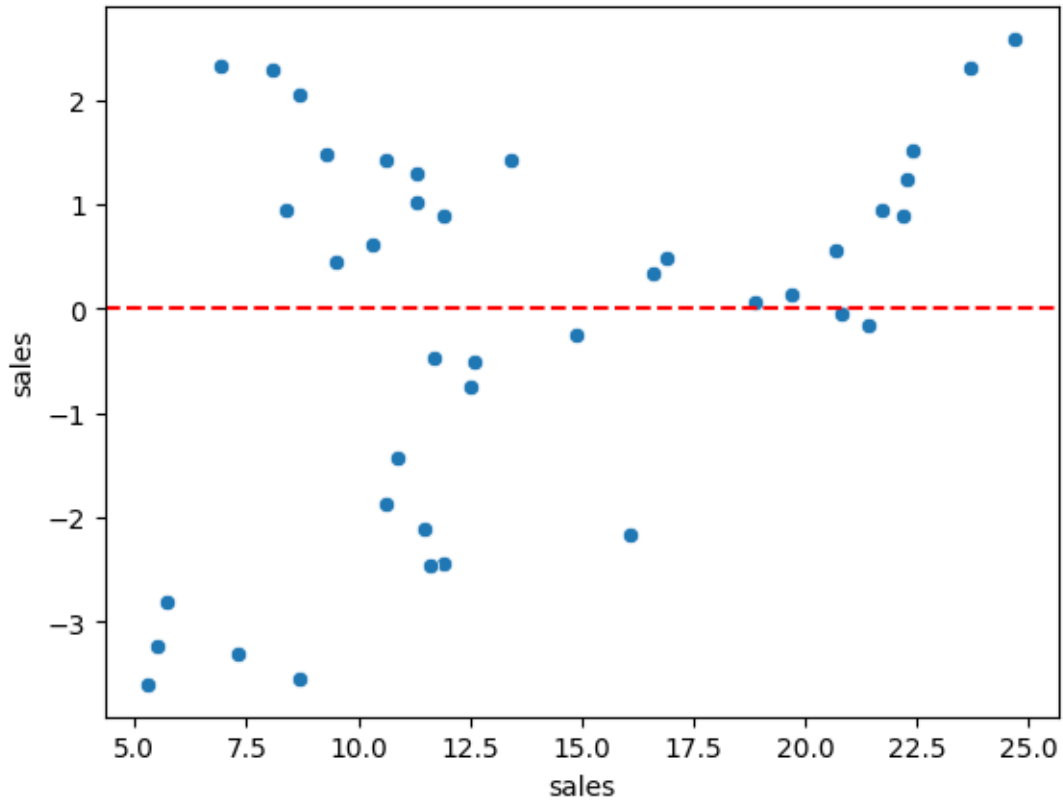
[82]: 1.4607567168117603

[83]:
```python
#RMSE
np.sqrt(mean_squared_error(y_test,test_pred))
```

[83]: 1.78159966153345

[84]:
```python
test_residuals=y_test-test_pred
```
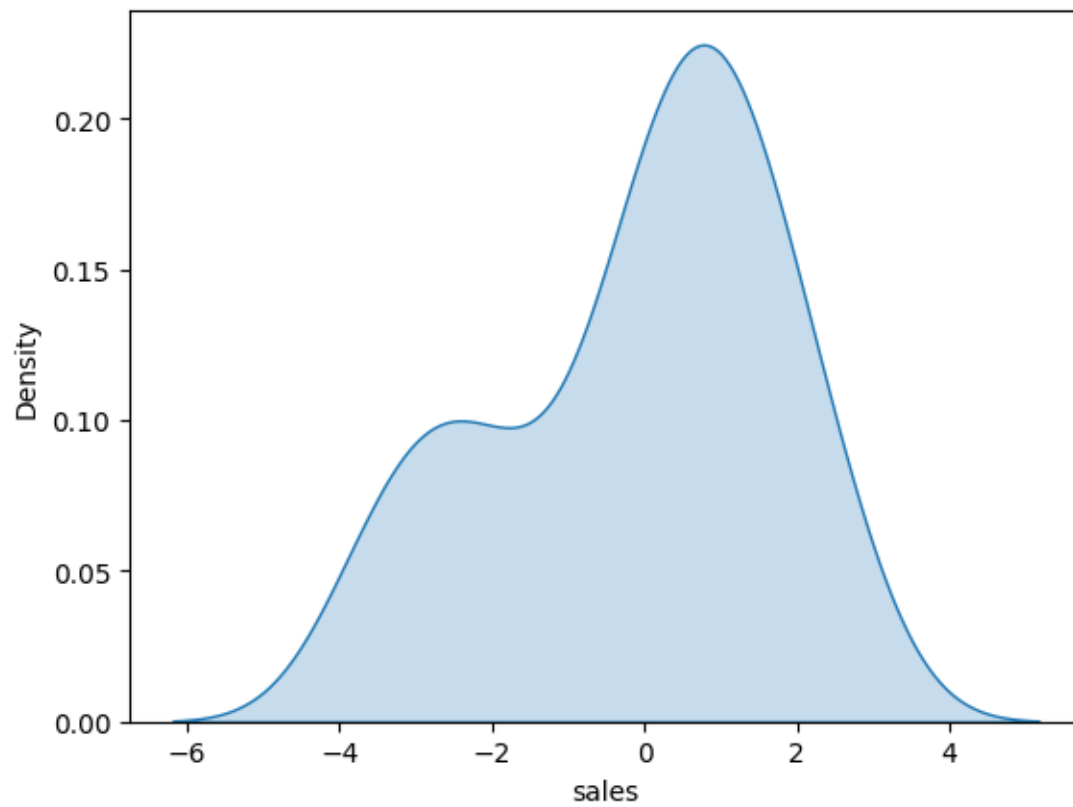
```
[85]: sns.scatterplot(x=y_test,y=test_residuals)
      plt.axhline(y=0,color='r',ls='--')
```

[85]: <matplotlib.lines.Line2D at 0x7fc3902da5f0>
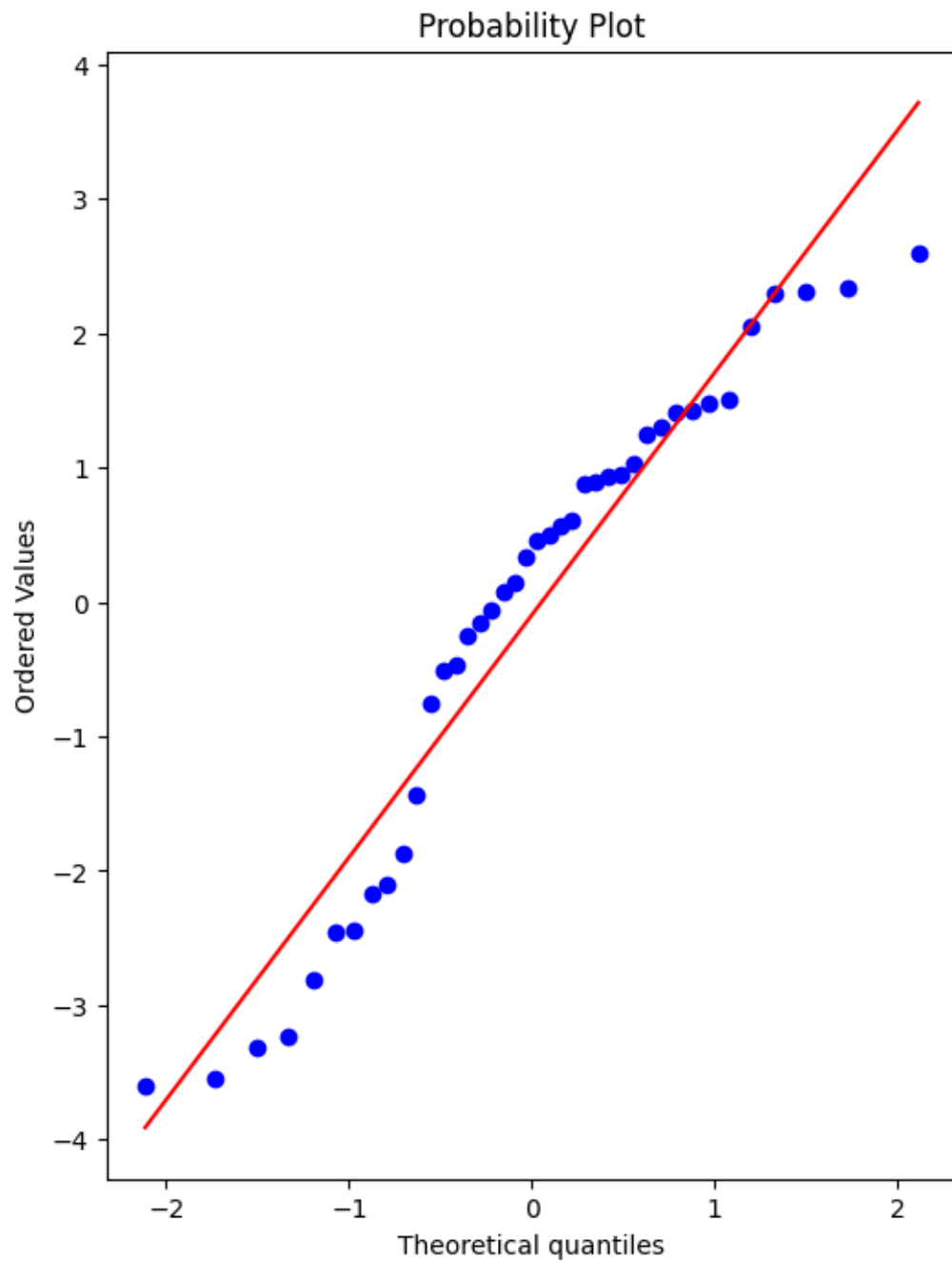


```
[86]: sns.kdeplot(test_residuals,fill=True)
```

[86]: <AxesSubplot:xlabel='sales', ylabel='Density'>

```
[87]: import scipy as sp
```

```
[88]: fig,ax=plt.subplots(figsize=(6,8))
      _=sp.stats.probplot(test_residuals,plot=ax)
```

Probability Plot


```
[89]: final_model=LinearRegression()
```

```
[90]: final_model.fit(X,y)
```

```
[90]: LinearRegression()
```

```
[91]: final_model.coef_
```

```
[91]: array([ 0.04576465,  0.18853002, -0.00103749])
```

```
[92]: y_hat=final_model.predict(X)
```

```
[93]: from joblib import dump,load
```

```
[94]: dump(final_model,'final_sales_model.joblib')
```

```
[94]: ['final_sales_model.joblib']
```

```
[ ]:
```

```
[95]: loaded_model=load('final_sales_model.joblib')
```

```
[96]: loaded_model
```

```
[96]: LinearRegression()
```

```
[97]: campaign=[[149,22,12]]
```

```
[98]: loaded_model.predict(campaign)
```

```
/home/mustafa/Desktop/ML/first/lib/python3.10/site-packages/sklearn/base.py:465:
UserWarning: X does not have valid feature names, but LinearRegression was
fitted with feature names
  warnings.warn(
```

```
[98]: array([13.893032])
```

```
[ ]:
```