In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:
```
df=pd.read_csv(r"C:\Users\user\Downloads\6_Salesworkload1.csv")
df.fillna(0,inplace=True)
df
```

Out[2]:

| | MonthYear | Time index | Country | StoreID | City | Dept_ID | Dept. Name | HoursOwn | HoursLea |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 1.0 | Dry | 3184.764 | |
| **1** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 2.0 | Frozen | 1582.941 | |
| **2** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 3.0 | other | 47.205 | |
| **3** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 4.0 | Fish | 1623.852 | |
| **4** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 5.0 | Fruits & Vegetables | 1759.173 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | |
| **7653** | 06.2017 | 9.0 | Sweden | 29650.0 | Gothenburg | 12.0 | Checkout | 6322.323 | |
| **7654** | 06.2017 | 9.0 | Sweden | 29650.0 | Gothenburg | 16.0 | Customer Services | 4270.479 | |
| **7655** | 06.2017 | 9.0 | Sweden | 29650.0 | Gothenburg | 11.0 | Delivery | 0 | |
| **7656** | 06.2017 | 9.0 | Sweden | 29650.0 | Gothenburg | 17.0 | others | 2224.929 | |
| **7657** | 06.2017 | 9.0 | Sweden | 29650.0 | Gothenburg | 18.0 | all | 39652.2 | |

7658 rows × 14 columns

In [3]:
```
df.head()
```

Out[3]:

| | MonthYear | Time index | Country | StoreID | City | Dept_ID | Dept. Name | HoursOwn | HoursLease | |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 1.0 | Dry | 3184.764 | 0.0 | 39 |
| **1** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 2.0 | Frozen | 1582.941 | 0.0 | 8 |
| **2** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 3.0 | other | 47.205 | 0.0 | 43 |
| **3** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 4.0 | Fish | 1623.852 | 0.0 | 30 |
| **4** | 10.2016 | 1.0 | United Kingdom | 88253.0 | London (I) | 5.0 | Fruits & Vegetables | 1759.173 | 0.0 | 16 |

Loading [MathJax]/extensions/Safe.js

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7658 entries, 0 to 7657
Data columns (total 14 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   MonthYear      7658 non-null   object 
 1   Time index     7658 non-null   float64
 2   Country        7658 non-null   object 
 3   StoreID        7658 non-null   float64
 4   City           7658 non-null   object 
 5   Dept_ID        7658 non-null   float64
 6   Dept. Name     7658 non-null   object 
 7   HoursOwn       7658 non-null   object 
 8   HoursLease     7658 non-null   float64
 9   Sales units    7658 non-null   float64
 10  Turnover       7658 non-null   float64
 11  Customer       7658 non-null   float64
 12  Area (m2)      7658 non-null   object 
 13  Opening hours  7658 non-null   object 
dtypes: float64(7), object(7)
memory usage: 837.7+ KB
```

In [5]:
```
import seaborn as sns
```

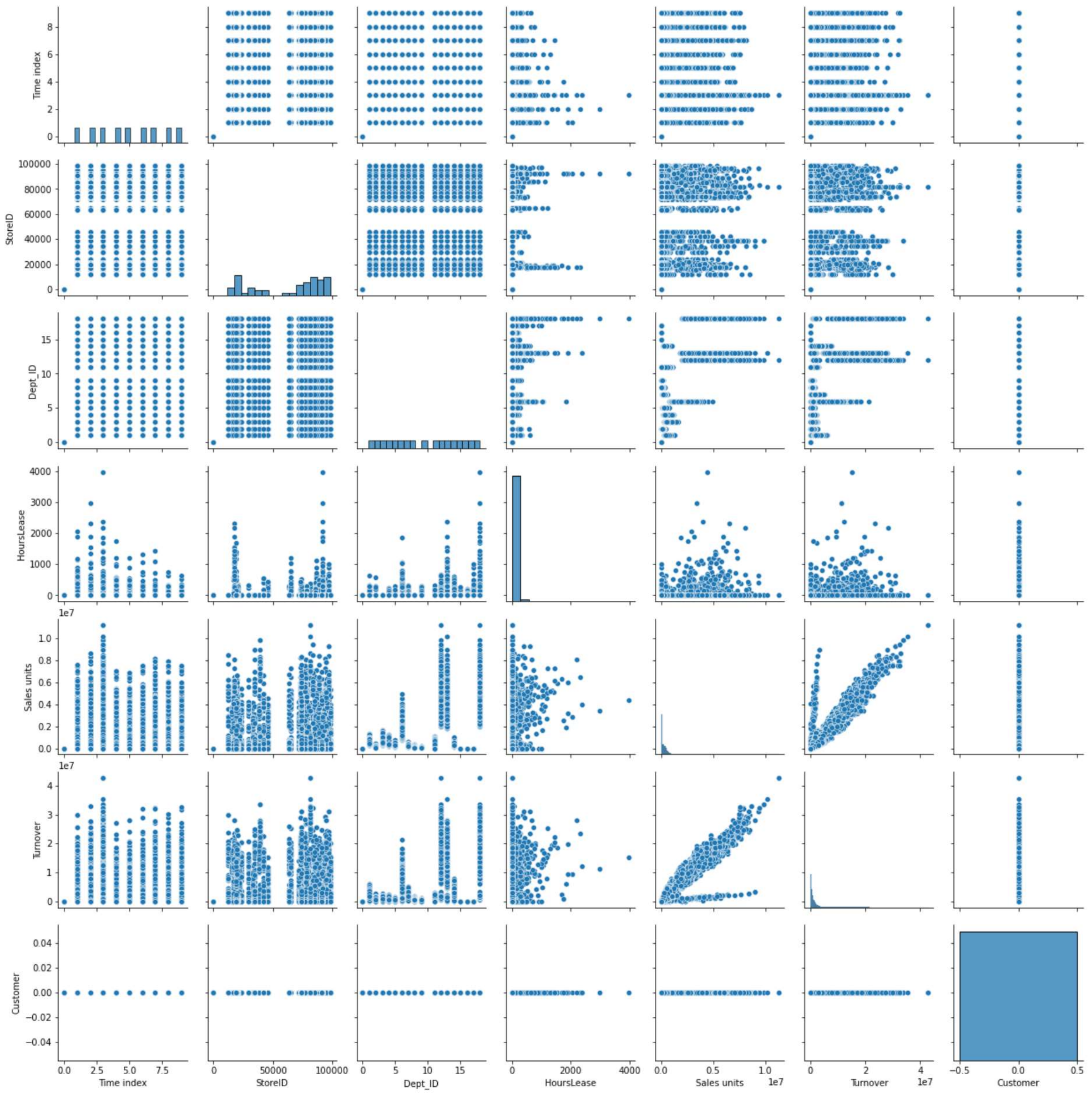
In [6]:
```
df.describe()
```

Out[6]:

| | Time index | StoreID | Dept_ID | HoursLease | Sales units | Turnover | Custon |
|---|---|---|---|---|---|---|---|
| **count** | 7658.000000 | 7658.000000 | 7658.000000 | 7658.000000 | 7.658000e+03 | 7.658000e+03 | 765 |
| **mean** | 4.994777 | 61930.456124 | 9.460695 | 22.013058 | 1.075346e+06 | 3.717505e+06 | ( |
| **std** | 2.585859 | 29975.929873 | 5.343407 | 133.231761 | 1.727560e+06 | 6.001448e+06 | ( |
| **min** | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000e+00 | 0.000000e+00 | ( |
| **25%** | 3.000000 | 29650.000000 | 5.000000 | 0.000000 | 5.441375e+04 | 2.720558e+05 | ( |
| **50%** | 5.000000 | 73949.000000 | 9.000000 | 0.000000 | 2.927625e+05 | 9.300810e+05 | ( |
| **75%** | 7.000000 | 87703.000000 | 14.000000 | 0.000000 | 9.154812e+05 | 3.251488e+06 | ( |
| **max** | 9.000000 | 98422.000000 | 18.000000 | 3984.000000 | 1.124296e+07 | 4.271739e+07 | ( |

In [7]:
```
sns.pairplot(df)
```

Out[7]:
```
<seaborn.axisgrid.PairGrid at 0x246dbf459d0>
```



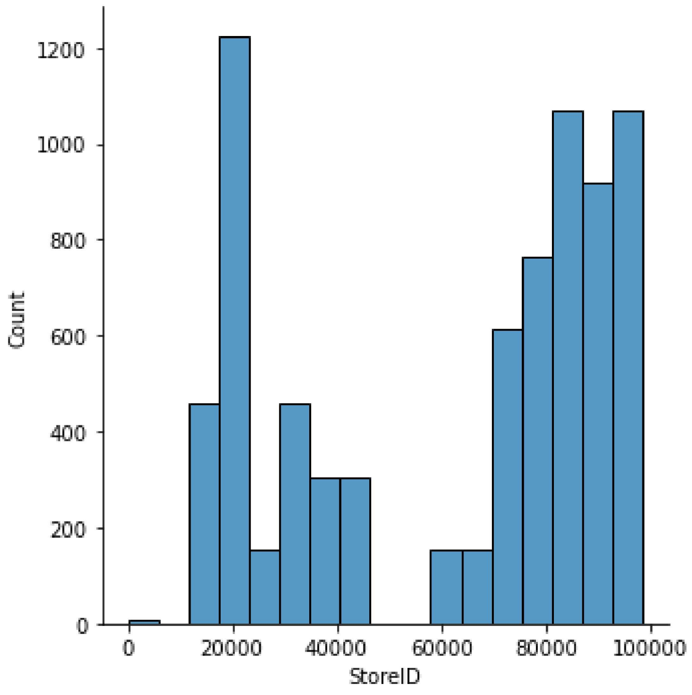
Loading [MathJax]/extensions/Safe.js

In [8]:
```
df1=df.drop(['Country'],axis=1)
df1
df1=df1.drop(df1.index[1537:])
df1.isna().sum()
```

Out[8]:
```
MonthYear        0
Time index       0
StoreID          0
City             0
Dept_ID          0
Dept. Name       0
HoursOwn         0
HoursLease       0
Sales units      0
Turnover         0
Customer         0
Area (m2)        0
Opening hours    0
dtype: int64
```

In [9]:
```
sns.displot(df['StoreID'])
```

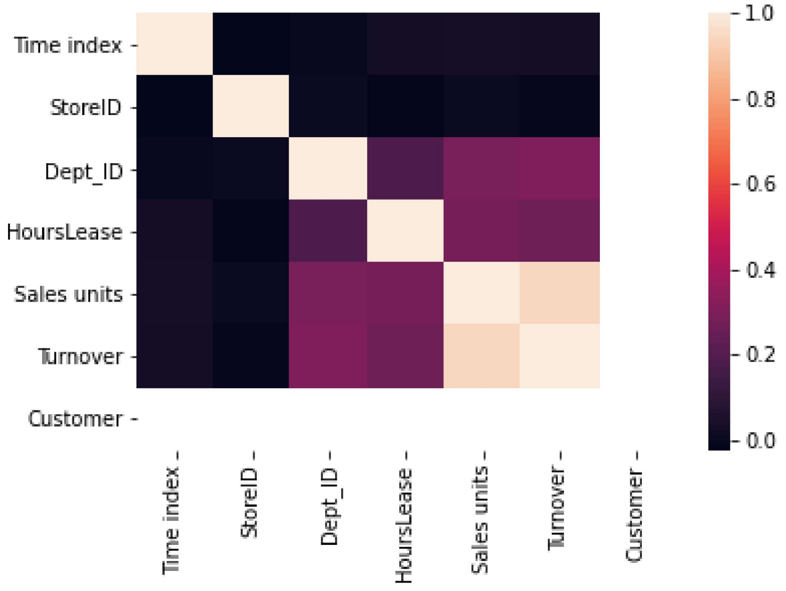
Out[9]:
```
<seaborn.axisgrid.FacetGrid at 0x246dc121310>
```



Loading [MathJax]/extensions/Safe.js

In [10]:
```
sns.heatmap(df1.corr())
```

Out[10]:
```
<AxesSubplot:>
```



In [11]:
```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [12]:
```
df1.isna().sum()
```

Out[12]:
```
MonthYear        0
Time index       0
StoreID          0
City             0
Dept_ID          0
Dept. Name       0
HoursOwn         0
HoursLease       0
Sales units      0
Turnover         0
Customer         0
Area (m2)        0
Opening hours    0
dtype: int64
```

Loading [MathJax]/extensions/Safe.js

In [13]:
```
y=df1['Turnover']
x=df1.drop(['Turnover','MonthYear','City','Opening hours','Dept. Name','Custome
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
print(x_train)
```

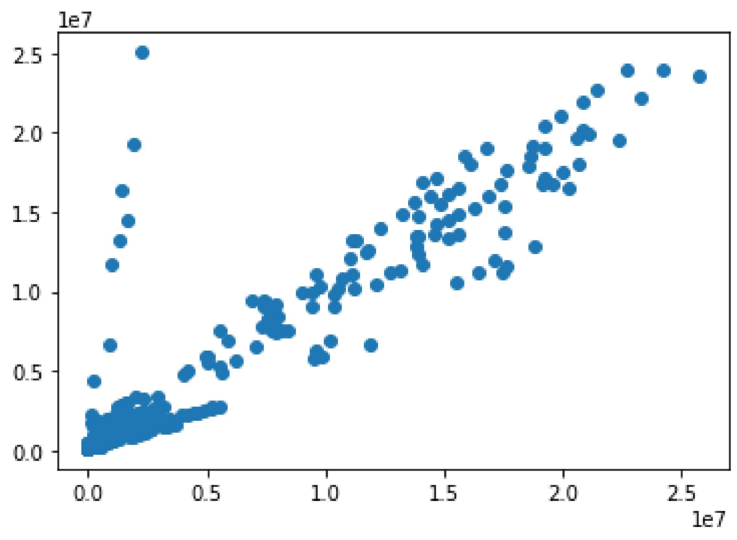
```
      Time index  StoreID  Dept_ID   HoursOwn  HoursLease  Sales units
1175         2.0  96857.0      2.0   2445.219         0.0      96480.0
647          1.0  98422.0      2.0   2634.039         0.0     119500.0
439          1.0  96493.0     11.0    5003.73         0.0     408445.0
100          1.0  18808.0     17.0   2017.227         0.0         25.0
989          2.0  23623.0      3.0     47.205         0.0     568530.0
...          ...      ...      ...        ...         ...          ...
806          1.0  81473.0      7.0  10010.607         0.0     429815.0
154          1.0  19769.0      2.0   1727.703         0.0     103060.0
42           1.0  17647.0      8.0   1957.434         0.0      76875.0
741          1.0  91973.0     14.0   6976.899       620.0     199750.0
605          1.0  34378.0     14.0  11810.691         0.0     495275.0

[1075 rows x 6 columns]
```

In [14]:
```
model=LinearRegression()
model.fit(x_train,y_train)
model.intercept_
```

Out[14]:
```
33013.282505813986
```

In [15]:
```
prediction=model.predict(x_test)
plt.scatter(y_test,prediction)
```

Out[15]:
```
<matplotlib.collections.PathCollection at 0x246e1927160>
```



In [16]:
```
model.score(x_test,y_test)
```

Out[16]:
```
0.864769102459002
```

Loading [MathJax]/extensions/Safe.js

In [17]:
```
from sklearn.linear_model import Ridge,Lasso
```

In [18]:
```
rr=Ridge(alpha=10)
rr.fit(x_train,y_train)
```

Out[18]:
```
Ridge(alpha=10)
```

In [19]:
```
rr.score(x_test,y_test)
```

Out[19]:
```
0.8647716459905493
```

In [20]:
```
la =Lasso(alpha=10)
la.fit(x_train,y_train)
```

Out[20]:
```
Lasso(alpha=10)
```

In [21]:
```
la.score(x_test,y_test)
```

Out[21]:
```
0.8647692530530686
```

Loading [MathJax]/extensions/Safe.js

In [22]:
```
from sklearn.linear_model import ElasticNet
en=ElasticNet()
en.fit(x_train,y_train)
print(en.coef_)
print(en.intercept_)
print(en.predict(x_test))
print(en.score(x_test,y_test))
from sklearn import metrics
print("Mean Absolute Error:",metrics.mean_absolute_error(y_test,prediction))
print("Mean Squared Error:",metrics.mean_squared_error(y_test,prediction))
print("Root Mean Squared Error:",np.sqrt(metrics.mean_squared_error(y_test,pred
```

Loading [MathJax]/extensions/Safe.js

```
[ 5.91281712e+03 -8.70401122e-01  1.15971431e+04  3.24023366e+01
 -1.35065268e+02  3.08655239e+00]
52501.59375305567
[  462244.06628371  2269476.23209006  2105033.16428777   688883.64572541
   286082.7318642   6966813.22015364   941926.66443989 13941231.68914202
 16793172.74997648  1827200.34357978 16200623.67274766   311313.37539213
  1442061.034047     893595.63550655  1555228.18886612  1337914.58300572
 11987167.41732434 23936173.05428037  7981099.19548364  1411315.69838724
   180557.41029904   290902.92048172   861187.1133865    448723.07633381
  2268395.48189931   574589.61942517   183366.79419416   391637.36075703
 18567866.43877792  2781007.69456374 14215902.96033365  2233232.19538449
   327746.594971    18485842.45849872  7807843.55420761   424081.62843975
 15584515.42084916   303689.23543836  3022702.19808835   405935.41460217
   699449.94272249  7405981.33451352   347299.86585551   263645.32161727
  1419998.43541877 17505984.08132511   406968.26048323   313781.89290022
   327309.36088958  1165205.26527575  2185506.78663655  1189033.77571633
 13563480.60800539   365236.93614737  9393507.31635557  1388293.31680634
 17202913.71102853  1208305.85206723  7482999.23393212 10192556.26262396
 21097782.42865413  1936945.47206621  6857788.88676217  3382721.75406507
  1659859.98121875   260923.31114269  9869436.7042002    296196.70388037
 12330229.14661974   282920.09611068  8481579.31194763  1680502.10241783
  1297349.9025637  19951575.5564132    724407.00352319   456363.20889173
 16858993.95009464  1351521.33296119   223069.79161693  1279158.52640466
  1701646.39795547   251272.0329979    794439.3406711  14557750.28418186
   402080.05798366  1497165.25958297  1675420.07292798   320101.35532052
 13597864.35045141   269880.54639668  9508865.96829285   508469.34733507
   187241.40718631   301947.62903317   448655.38076559  2316970.11086571
   341495.15493049   293421.85600669   108734.88512159   377270.15311569
   759895.22340083   542875.62999309   853160.87817503 19722174.25707543
 13460772.19938864  1876000.37894049  5956942.67673296   929116.13742029
  5751669.94428228   336152.43005001 11274873.40200516   803179.5669999
  7539596.75086837   337311.12244633  1533233.25480557   180224.89444228
  1360328.67950027   424224.15523835   182522.676437    2448063.60080037
   171505.37039175   551881.2209463   2228921.34774416  1293042.15658461
  1590161.1425788    559699.5914434   1098997.81481291   279849.86156872
   353913.21663447  1522463.33459473  1630004.02344695  1304836.87815157
  6607245.07556955   238420.18332593   681889.97955637  2856575.320444
 13712604.13893146   335015.157492     299499.8803033  12490225.64407302
   476251.24385521  1714044.52689921 11774248.01928434  1882142.0779955
   328516.85532827 14882096.04325702 12838676.03901198 18009167.21690749
  8742032.3349558    266488.76934063 15219359.77350081  1488938.11351726
 12616089.39739024 20435860.86096258  2713228.87153567  2030706.81572237
   267890.91468996   429729.09174819   905325.58496738   324264.31322605
 11172777.44086461   190387.95191863 10261908.00829017 16547892.72907457
  4346672.31427543 13280689.55523516  1628315.99857824  1055422.57464171
   207511.39171542 13228537.84401189  4763908.37805752  5698911.53448797
  1702279.69832512 23951763.23416855  1109717.2961228    658539.80836975
   965990.58071699   324759.22728517  9048106.35009173   373301.63963163
   523046.01918086   352495.36189507  1538527.93664157  2632477.5199982
  9958027.75750918   243293.9984312   9209731.63328844   357766.2455272
   237096.02310575  5307560.13798665  1560993.55017131   248908.49503908
  6262664.08107271   416094.88086084  1266301.67380305 10451471.41110025
   450258.13586277  1311115.45093266  1645318.31057201   355124.29137595
   943734.58117586 12858835.39242516   920913.9144002   1225580.45029792
  1475515.45305615   318515.70966461 14911973.13045475   283274.37326969
   489653.76383423  1375914.9370615    369865.97167877  5559259.54036704
   322599.9659392    428092.54598513   431766.90137537   769259.02869702
```

Loading [MathJax]/extensions/Safe.js

```
 11548044.01789074   325307.49721536   179652.35277347  1461581.16908952
  1611258.15515934  1232515.73417364   309248.00648783  1458645.88738465
   240431.02846279  6621660.71339243   984925.26729706   683420.83934594
  1882945.23677125   470011.56605186  1933887.7147775   1284133.60233309
   157971.04767214 14787347.98692013  1853476.49030939   423385.6625844
   253248.47945821   314935.80939586   472322.01571406   263484.60462845
  5884376.60891201   387904.05807238  1542949.89505714   593619.19495937
  1265615.68629705  3362466.17192858 19570795.6302726  22161347.80527036
 16783663.47879988  2370644.23597511   214365.32479846   430044.65462515
  1711784.39647948   676008.80170921  3230921.58889462 20169653.68895246
   471508.35582847  2685490.62356119  2720094.2915146    424215.87646617
  9286093.94096111   387025.24783593   431478.87548156  1822366.12031705
  2278159.65888747   106783.44580694  1379512.09729595  5889898.48960366
   114435.32437138 19150386.01800586  1897976.52553656   339013.37426465
 16568764.53661312  1242564.23964442   328758.66036102   423646.38094723
   372969.13198578   634467.59540425  1071430.91442926  8250901.19392187
   348237.48861713 13514721.25860493   715474.16444487  1433307.67747345
  1461503.42374304   300459.84855592   359971.21464282  1636146.95459774
 11127219.48740398   268908.661329    1761965.55266624  1600894.79834688
   256864.09174783   316646.06785015   351026.74733383   307266.26551087
   145430.48743589  1364187.96718216   377948.01597825   344442.77094599
   208393.30393204   283469.30292195  1353497.59540043  1482612.05429015
  1404634.65645365  1487331.75988077  3061538.90597291   362962.44819469
 12093989.13267286  1844382.14831104 16374153.11632121   350769.88291378
   427412.48741197  1685578.48950975   372758.94588981   267010.93620354
   833084.69344825   335951.79379663  2471054.60040988   297387.33540533
   588079.38994845   493021.37664847   303180.44352649 11089551.95855783
  4961049.32232805   407237.7178341  23620556.76246073   342269.22241604
  1928519.01757879  1775202.25428118  1543682.05510932 17696281.67687373
   848768.29290393   432390.07637558   961963.10880662   377339.53503644
   282360.09988384  1771500.93559468  1314228.30428657   933584.90867211
  1963483.07940489  1937940.58125869  2189791.56376562   331714.32277246
  1448810.78450138   290744.32345044 19101497.18425009  2295142.2456638
  2662096.17544094 19275762.44739357  1911319.6795918    339228.63613481
  2403377.48155319   343795.74951543   526201.62981869  7905492.29362962
   329997.97670759  1944130.75430867   258750.30600481 21891706.05050242
   306544.0068148  16722160.66226767  1917876.37090165  4903181.60466972
   119090.91766652  2304791.843298     297294.61999107 15435357.9352918
  1034858.68840672   357322.19103572  1539924.97012547  1391973.3519168
 19094108.66423761  1450404.03648296   321803.54461049   283775.24994172
   360278.57687713  1079686.84718067  8181202.78314111 13297508.88035269
 10243726.74271062   207838.60501862   375357.29799255  6493913.53504549
   884879.89223108   381677.74559724  1895548.84530323 17197615.93032748
  1838099.80283285  7540626.72736274  1175770.00331545 11156610.81936805
  1359278.52859191   383766.86407827   252005.37899258   660867.786955
  1198694.7658534    226068.50786124  2190227.72469304 22674790.53083359
  1419040.09134503  2400262.0632972    452468.8494666    709733.33203633
  1479398.91751717  1932378.72906955   319215.6511053   2521135.64339855
  2493374.90385175 15955403.74157708   764017.75035969  2151854.92770888
 10578650.39116335 17867559.6312477  18017458.40276735   275242.12375526
 15528200.96432324   741551.74964223  1549614.8645155    113801.05329687
   256850.68370544   315269.47147816   317968.90263578   279204.0100746
   937273.7356801    637016.77675626   277021.26192084  9927602.26619122
   967121.97790684  1872458.16775221 14497497.1585055    275377.75109267
  1662551.43361758   302329.64243711   980396.21279115  1871018.65642594
 15962170.07827333 11237660.0864967  25074579.67458021  1001752.45050179
   349208.09894213   152428.61683571 13300470.79946513  7573663.37465634
  1550048.03587647 10860600.47727663   424477.27483667   940079.93797458
  2244417.84632412 11700443.43224945   109167.47447903  9121005.83922358
  1298232.90655166  1712470.96883503   480836.94704025  1070938.22340348
   730391.85086242   408741.67356412  2626477.08106002  9060697.0867383
   354851.15754961  2443506.36573225]
0.8648072262322661
Mean Absolute Error: 985030.9550554731
Mean Squared Error: 4740682315027.354
Root Mean Squared Error: 2177310.798904776
```

Loading [MathJax]/extensions/Safe.js