

Lesson-01

In [1]:

```
from sklearn.datasets import load_boston
```

In [2]:

```
data = load_boston()
```

In [3]:

```
import pandas as pd
```

In [4]:

```
dataframe = pd.DataFrame(data['data'])
```

In [5]:

```
dataframe.columns = data['feature_names']
```

In [6]:

```
dataframe['price'] = data['target']
```

In [7]:

```
import seaborn as sns
```

In [8]:

```
%matplotlib inline
```

In [9]:

```
sns.heatmap(dataframe.corr(), annot=True, fmt='.1f')
```

Out[9]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7843aa1d10>
```



In [10]:

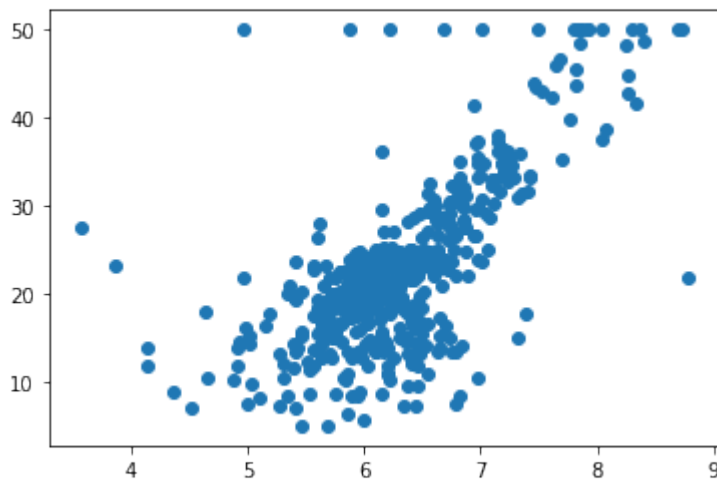
```
import matplotlib.pyplot as plt
```

In [11]:

```
plt.scatter(dataframe['RM'], dataframe['price'])
```

Out[11]:

```
<matplotlib.collections.PathCollection at 0x7f783db6b750>
```



In [12]:

```
x = dataframe['RM']
```

In [13]:

```
y = dataframe['price']
```

In [14]:

```
history_notes = {_x : _y for _x, _y in zip(x, y)}  
history_notes[6.58]
```

In [16]:

```
similarity_ys = [y for _, y in sorted(history_notes.items(),  
key=lambda x_y: (x_y[0] - 6.57) ** 2)[:3]]
```

In [17]:

```
import numpy as np
```

In [18]:

```
np.mean(similarity_ys)
```

Out[18]:

```
24.2
```

用历史数据来预测未曾见到的过的数据，最直接的方法

K-Neighbor-Nearst

In [19]:

```
def knn(query_x, history, top_n=3):
    sorted_notes = sorted(history.items(), key=lambda x_y:
        (x_y[0] - query_x) ** 2)
    similar_notes = sorted_notes[:top_n]
    similar_ys = [y for _, y in similar_notes]

    return np.mean(similar_ys)
```

In [20]:

```
knn(5.4, history_notes)
```

Out[20]:

```
15.700000000000001
```

为了更快的获得结果，我们希望通过拟合函数来获得预测能力

$$f(rm) = k * rm + b$$

Random Approach

$$\text{Loss}(k, b) = \frac{1}{n} \sum_{i \in N} (\hat{y}_i - y_i)^2$$
$$\text{Loss}(k, b) = \frac{1}{n} \sum_{i \in N} ((k * rm_i + b) - y_i)^2$$

In [21]:

```
def loss(y_hat, y):
    return np.mean((y_hat - y) ** 2)
```

In [22]:

```
import random
```

In [23]:

```
min_loss = float('inf')
best_k, best_b = None, None

for step in range(1000):
    min_v, max_v = -100, 100
    k, b = random.randrange(min_v, max_v),
    random.randrange(min_v, max_v)
    y_hats = [k * rm_i + b for rm_i in x]
    current_loss = loss(y_hats, y)
```

```

    if current_loss < min_loss:
        min_loss = current_loss
        best_k, best_b = k, b
        print('在第{}步, 我们获得了函数  $f(rm) = {} * rm + {}$ ,
此时loss是: {}'.format(step, k, b, current_loss))

```

```

在第0步, 我们获得了函数  $f(rm) = -48 * rm + 91$ , 此时loss是:
56030.13547566797
在第3步, 我们获得了函数  $f(rm) = 19 * rm + 49$ , 此时loss是:
21371.456735903175
在第8步, 我们获得了函数  $f(rm) = -19 * rm + 16$ , 此时loss是:
16293.796158037558
在第10步, 我们获得了函数  $f(rm) = 8 * rm + 92$ , 此时loss是:
14382.888905549413
在第13步, 我们获得了函数  $f(rm) = 5 * rm + -100$ , 此时loss是:
8352.856759239126
在第19步, 我们获得了函数  $f(rm) = -5 * rm + 11$ , 此时loss是:
1986.798660424901
在第20步, 我们获得了函数  $f(rm) = -8 * rm + 88$ , 此时loss是:
418.444801201581
在第62步, 我们获得了函数  $f(rm) = 9 * rm + -15$ , 此时loss是:
405.7048442826085
在第82步, 我们获得了函数  $f(rm) = 7 * rm + -31$ , 此时loss是:
136.79628004545464
在第90步, 我们获得了函数  $f(rm) = -3 * rm + 40$ , 此时loss是:
117.68416961067193
在第194步, 我们获得了函数  $f(rm) = 7 * rm + -18$ , 此时loss是:
57.74677411660077
在第518步, 我们获得了函数  $f(rm) = 11 * rm + -49$ , 此时loss是:
51.14401321541502

```

In [24]:

```

plt.scatter(x, y)
plt.scatter(x, [best_k * rm + best_b for rm in x])

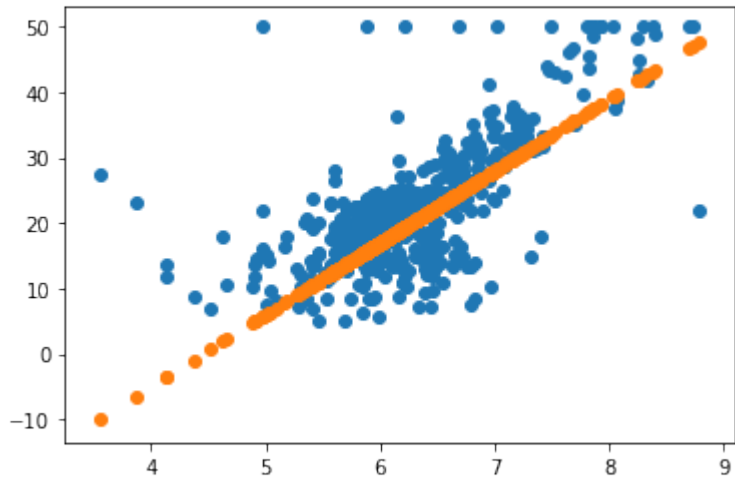
```

Out[24]:

```

<matplotlib.collections.PathCollection at 0x7f783c27f9d0>

```



蒙特卡洛模拟

Supervisor

$$\text{Loss}(k, b) = \frac{1}{n} \sum_{i \in N} ((k * x_i + b) - y_i)^2$$

$$\frac{\partial \text{loss}(k, b)}{\partial k} = \frac{2}{n} \sum_{i \in N} (k * x_i + b - y_i) * x_i$$

$$\frac{\partial \text{loss}(k, b)}{\partial b} = \frac{2}{n} \sum_{i \in N} (k * x_i + b - y_i)$$

In [25]:

```
def partial_k(k, b, x, y):
    return 2 * np.mean((k * x + b - y) * x)

def partial_b(k, b, x, y):
    return 2 * np.mean(k * x + b - y)

k, b = random.random(), random.random()
min_loss = float('inf')
best_k, best_b = None, None
learning_rate = 1e-2

for step in range(2000):
    k, b = k + (-1 * partial_k(k, b, x, y) * learning_rate),
    b + (-1 * partial_b(k, b, x, y) * learning_rate)
    y_hats = k * x + b
    current_loss = loss(y_hats, y)

    if current_loss < min_loss:
        min_loss = current_loss
        best_k, best_b = k, b
```

```
print('在第{}步，我们获得了函数  $f(rm) = {} * rm + {}$ ,  
此时loss是: {}'.format(step, k, b, current_loss))
```

在第0步，我们获得了函数 $f(rm) = 3.0111417302649826 * rm + 0.752026014419545$ ，此时loss是: 70.04110921910225
在第1步，我们获得了函数 $f(rm) = 3.4302473783601264 * rm + 0.8091631233491027$ ，此时loss是: 59.478050968399835
在第2步，我们获得了函数 $f(rm) = 3.5069762736063566 * rm + 0.8124789747410274$ ，此时loss是: 59.12685811839347
在第3步，我们获得了函数 $f(rm) = 3.5219216566167675 * rm + 0.8060842480337541$ ，此时loss是: 59.10820423942305
在第4步，我们获得了函数 $f(rm) = 3.525717694565284 * rm + 0.7979388905006388$ ，此时loss是: 59.100382510385046
在第5步，我们获得了函数 $f(rm) = 3.527501524520546 * rm + 0.7894793059056471$ ，此时loss是: 59.09291698482354
在第6步，我们获得了函数 $f(rm) = 3.5289219792985747 * rm + 0.7809646986209946$ ，此时loss是: 59.085466526131654
在第7步，我们获得了函数 $f(rm) = 3.5302765974073917 * rm + 0.7724418427031616$ ，此时loss是: 59.07802002399254
在第8步，我们获得了函数 $f(rm) = 3.531619070918583 * rm + 0.7639191783127174$ ，此时loss是: 59.07057711494723
在第9步，我们获得了函数 $f(rm) = 3.5329590888436884 * rm + 0.7553982281062311$ ，此时loss是: 59.06313778549417
在第10步，我们获得了函数 $f(rm) = 3.5342983996669006 * rm + 0.746879266449239$ ，此时loss是: 59.05570203352862
在第11步，我们获得了函数 $f(rm) = 3.535637318972632 * rm + 0.7383623424482884$ ，此时loss是: 59.0482698573176
在第12步，我们获得了函数 $f(rm) = 3.5369759037728095 * rm + 0.729847464561143$ ，此时loss是: 59.04084125514141
在第13步，我们获得了函数 $f(rm) = 3.538314164418913 * rm + 0.721334633910429$ ，此时loss是: 59.03341622528119
在第14步，我们获得了函数 $f(rm) = 3.539652102842343 * rm + 0.7128238502952143$ ，此时loss是: 59.02599476601903
在第15步，我们获得了函数 $f(rm) = 3.5409897194550473 * rm + 0.704315113275823$ ，此时loss是: 59.01857687563797
在第16步，我们获得了函数 $f(rm) = 3.5423270143947683 * rm + 0.6958084223695936$ ，此时loss是: 59.01116255242172
在第17步，我们获得了函数 $f(rm) = 3.543663987749749 * rm + 0.6873037770862048$ ，此时loss是: 59.0037517946549
在第18步，我们获得了函数 $f(rm) = 3.545000639599288 * rm + 0.6788011769340501$ ，此时loss是: 58.99634460062289
在第19步，我们获得了函数 $f(rm) = 3.546336970021053 * rm + 0.6703006214213878$ ，此时loss是: 58.988940968612084
在第20步，我们获得了函数 $f(rm) = 3.5476729790924035 * rm + 0.6618021100565491$ ，此时loss是: 58.98154089690936
在第21步，我们获得了函数 $f(rm) = 3.549008666890625 * rm + 0.6533056423479746$ ，此时loss是: 58.97414438380269

在第22步, 我们获得了函数 $f(rm) = 3.5503440334929786 * rm + 0.6448112178042222$, 此时loss是: 58.96675142758082
在第23步, 我们获得了函数 $f(rm) = 3.5516790789767017 * rm + 0.6363188359339674$, 此时loss是: 58.9593620265333
在第24步, 我们获得了函数 $f(rm) = 3.5530138034190157 * rm + 0.6278284962460041$, 此时loss是: 58.95197617895038
在第25步, 我们获得了函数 $f(rm) = 3.5543482068971217 * rm + 0.6193401982492439$, 此时loss是: 58.94459388312339
在第26步, 我们获得了函数 $f(rm) = 3.5556822894882028 * rm + 0.6108539414527168$, 此时loss是: 58.93721513734426
在第27步, 我们获得了函数 $f(rm) = 3.5570160512694238 * rm + 0.6023697253655707$, 此时loss是: 58.92983993990589
在第28步, 我们获得了函数 $f(rm) = 3.5583494923179297 * rm + 0.5938875494970716$, 此时loss是: 58.92246828910188
在第29步, 我们获得了函数 $f(rm) = 3.5596826127108487 * rm + 0.5854074133566034$, 此时loss是: 58.91510018322667
在第30步, 我们获得了函数 $f(rm) = 3.5610154125252897 * rm + 0.5769293164536683$, 此时loss是: 58.90773562057561
在第31步, 我们获得了函数 $f(rm) = 3.5623478918383418 * rm + 0.5684532582978861$, 此时loss是: 58.90037459944476
在第32步, 我们获得了函数 $f(rm) = 3.5636800507270774 * rm + 0.5599792383989947$, 此时loss是: 58.89301711813102
在第33步, 我们获得了函数 $f(rm) = 3.5650118892685496 * rm + 0.5515072562668498$, 此时loss是: 58.88566317493221
在第34步, 我们获得了函数 $f(rm) = 3.5663434075397933 * rm + 0.5430373114114251$, 此时loss是: 58.878312768146934
在第35步, 我们获得了函数 $f(rm) = 3.567674605617824 * rm + 0.534569403342812$, 此时loss是: 58.87096589607447
在第36步, 我们获得了函数 $f(rm) = 3.56900548357964 * rm + 0.5261035315712198$, 此时loss是: 58.86362255701515
在第37步, 我们获得了函数 $f(rm) = 3.570336041502219 * rm + 0.5176396956069754$, 此时loss是: 58.85628274926979
在第38步, 我们获得了函数 $f(rm) = 3.571666279462524 * rm + 0.5091778949605239$, 此时loss是: 58.84894647114048
在第39步, 我们获得了函数 $f(rm) = 3.572996197537495 * rm + 0.5007181291424276$, 此时loss是: 58.84161372092972
在第40步, 我们获得了函数 $f(rm) = 3.574325795804056 * rm + 0.4922603976633667$, 此时loss是: 58.83428449694094
在第41步, 我们获得了函数 $f(rm) = 3.5756550743391124 * rm + 0.4838047000341392$, 此时loss是: 58.826958797478625
在第42步, 我们获得了函数 $f(rm) = 3.5769840332195506 * rm + 0.47535103576566057$, 此时loss是: 58.819636620847625
在第43步, 我们获得了函数 $f(rm) = 3.5783126725222383 * rm + 0.4668994043689639$, 此时loss是: 58.81231796535405
在第44步, 我们获得了函数 $f(rm) = 3.5796409923240256 * rm + 0.4584498053552$, 此时loss是: 58.80500282930457

在第45步, 我们获得了函数 $f(rm) = 3.5809689927017434 * rm + 0.4500022382356372$, 此时loss是: 58.797691211006764
在第46步, 我们获得了函数 $f(rm) = 3.5822966737322046 * rm + 0.4415567025216611$, 此时loss是: 58.79038310876884
在第47步, 我们获得了函数 $f(rm) = 3.583624035492203 * rm + 0.43311319772477513$, 此时loss是: 58.783078520900155
在第48步, 我们获得了函数 $f(rm) = 3.584951078058514 * rm + 0.4246717233566001$, 此时loss是: 58.77577744571059
在第49步, 我们获得了函数 $f(rm) = 3.5862778015078955 * rm + 0.4162322789288742$, 此时loss是: 58.76847988151103
在第50步, 我们获得了函数 $f(rm) = 3.5876042059170863 * rm + 0.40779486395345305$, 此时loss是: 58.761185826613044
在第51步, 我们获得了函数 $f(rm) = 3.5889302913628067 * rm + 0.3993594779423096$, 此时loss是: 58.753895279329
在第52步, 我们获得了函数 $f(rm) = 3.5902560579217577 * rm + 0.3909261204075342$, 此时loss是: 58.74660823797224
在第53步, 我们获得了函数 $f(rm) = 3.5915815056706233 * rm + 0.3824947908613346$, 此时loss是: 58.73932470085676
在第54步, 我们获得了函数 $f(rm) = 3.592906634686069 * rm + 0.37406548881603585$, 此时loss是: 58.73204466629732
在第55步, 我们获得了函数 $f(rm) = 3.5942314450447403 * rm + 0.36563821378408007$, 此时loss是: 58.72476813260967
在第56步, 我们获得了函数 $f(rm) = 3.595555936823266 * rm + 0.35721296527802693$, 此时loss是: 58.71749509811041
在第57步, 我们获得了函数 $f(rm) = 3.596880110098256 * rm + 0.3487897428105531$, 此时loss是: 58.71022556111666
在第58步, 我们获得了函数 $f(rm) = 3.5982039649463005 * rm + 0.3403685458944524$, 此时loss是: 58.70295951994657
在第59步, 我们获得了函数 $f(rm) = 3.5995275014439736 * rm + 0.33194937404263614$, 此时loss是: 58.695696972919
在第60步, 我们获得了函数 $f(rm) = 3.6008507196678288 * rm + 0.32353222676813237$, 此时loss是: 58.68843791835367
在第61步, 我们获得了函数 $f(rm) = 3.6021736196944034 * rm + 0.3151171035840864$, 此时loss是: 58.681182354571234
在第62步, 我们获得了函数 $f(rm) = 3.6034962016002137 * rm + 0.3067040040037606$, 此时loss是: 58.67393027989288
在第63步, 我们获得了函数 $f(rm) = 3.6048184654617597 * rm + 0.29829292754053444$, 此时loss是: 58.66668169264081
在第64步, 我们获得了函数 $f(rm) = 3.606140411355521 * rm + 0.2898838737079043$, 此时loss是: 58.65943659113793
在第65步, 我们获得了函数 $f(rm) = 3.6074620393579613 * rm + 0.28147684201948375$, 此时loss是: 58.65219497370801
在第66步, 我们获得了函数 $f(rm) = 3.6087833495455244 * rm + 0.27307183198900314$, 此时loss是: 58.644956838675604
在第67步, 我们获得了函数 $f(rm) = 3.6101043419946355 * rm + 0.2646688431303098$, 此时loss是: 58.63772218436602

在第68步, 我们获得了函数 $f(rm) = 3.6114250167817024 * rm + 0.256267874957368$, 此时loss是: 58.630491009105576
在第69步, 我们获得了函数 $f(rm) = 3.6127453739831146 * rm + 0.24786892698425889$, 此时loss是: 58.62326331122109
在第70步, 我们获得了函数 $f(rm) = 3.6140654136752404 * rm + 0.23947199872518024$, 此时loss是: 58.616039089040385
在第71步, 我们获得了函数 $f(rm) = 3.6153851359344347 * rm + 0.23107708969444715$, 此时loss是: 58.60881834089202
在第72步, 我们获得了函数 $f(rm) = 3.6167045408370293 * rm + 0.222684199406491$, 此时loss是: 58.60160106510544
在第73步, 我们获得了函数 $f(rm) = 3.618023628459341 * rm + 0.21429332737586024$, 此时loss是: 58.59438726001079
在第74步, 我们获得了函数 $f(rm) = 3.619342398877666 * rm + 0.2059044731172199$, 此时loss是: 58.5871769239391
在第75步, 我们获得了函数 $f(rm) = 3.6206608521682835 * rm + 0.19751763614535184$, 此时loss是: 58.579970055222034
在第76步, 我们获得了函数 $f(rm) = 3.6219789884074536 * rm + 0.18913281597515447$, 此时loss是: 58.572766652192335
在第77步, 我们获得了函数 $f(rm) = 3.6232968076714185 * rm + 0.18075001212164302$, 此时loss是: 58.56556671318336
在第78步, 我们获得了函数 $f(rm) = 3.6246143100364026 * rm + 0.1723692240999492$, 此时loss是: 58.55837023652931
在第79步, 我们获得了函数 $f(rm) = 3.62593149557861 * rm + 0.16399045142532134$, 此时loss是: 58.551177220565116
在第80步, 我们获得了函数 $f(rm) = 3.6272483643742293 * rm + 0.15561369361312444$, 此时loss是: 58.54398766362667
在第81步, 我们获得了函数 $f(rm) = 3.6285649164994283 * rm + 0.14723895017883992$, 此时loss是: 58.536801564050606
在第82步, 我们获得了函数 $f(rm) = 3.6298811520303573 * rm + 0.13886622063806575$, 此时loss是: 58.52961892017419
在第83步, 我们获得了函数 $f(rm) = 3.631197071043149 * rm + 0.13049550450651645$, 此时loss是: 58.52243973033564
在第84步, 我们获得了函数 $f(rm) = 3.632512673613917 * rm + 0.1221268013000229$, 此时loss是: 58.515263992874054
在第85步, 我们获得了函数 $f(rm) = 3.6338279598187566 * rm + 0.11376011053453249$, 此时loss是: 58.50809170612916
在第86步, 我们获得了函数 $f(rm) = 3.6351429297337456 * rm + 0.10539543172610899$, 此时loss是: 58.5009228684416
在第87步, 我们获得了函数 $f(rm) = 3.6364575834349426 * rm + 0.09703276439093247$, 此时loss是: 58.493757478152766
在第88步, 我们获得了函数 $f(rm) = 3.6377719209983885 * rm + 0.0886721080452995$, 此时loss是: 58.48659553360469
在第89步, 我们获得了函数 $f(rm) = 3.6390859425001056 * rm + 0.08031346220562277$, 此时loss是: 58.47943703314057
在第90步, 我们获得了函数 $f(rm) = 3.6403996480160967 * rm + 0.07195682638843134$, 此时loss是: 58.47228197510417

在第91步, 我们获得了函数 $f(rm) = 3.64171303762235 * rm + 0.06360220011037071$, 此时loss是: 58.465130357839996
在第92步, 我们获得了函数 $f(rm) = 3.643026111394831 * rm + 0.05524958288820231$, 此时loss是: 58.457982179693445
在第93步, 我们获得了函数 $f(rm) = 3.64433886940949 * rm + 0.04689897423880403$, 此时loss是: 58.450837439010655
在第94步, 我们获得了函数 $f(rm) = 3.6456513117422578 * rm + 0.0385503736791697$, 此时loss是: 58.44369613413868
在第95步, 我们获得了函数 $f(rm) = 3.646963438469047 * rm + 0.03020378072640955$, 此时loss是: 58.43655826342521
在第96步, 我们获得了函数 $f(rm) = 3.6482752496657516 * rm + 0.02185919489774973$, 此时loss是: 58.42942382521886
在第97步, 我们获得了函数 $f(rm) = 3.6495867454082487 * rm + 0.013516615710532632$, 此时loss是: 58.42229281786896
在第98步, 我们获得了函数 $f(rm) = 3.650897925772396 * rm + 0.005176042682216589$, 此时loss是: 58.41516523972563
在第99步, 我们获得了函数 $f(rm) = 3.6522087908340333 * rm + -0.0031625246696239274$, 此时loss是: 58.40804108913982
在第100步, 我们获得了函数 $f(rm) = 3.6535193406689817 * rm + -0.011499086827298537$, 此时loss是: 58.400920364463325
在第101步, 我们获得了函数 $f(rm) = 3.654829575353045 * rm + -0.019833644273000696$, 此时loss是: 58.3938030640486
在第102步, 我们获得了函数 $f(rm) = 3.6561394949620083 * rm + -0.028166197488808192$, 此时loss是: 58.38668918624896
在第103步, 我们获得了函数 $f(rm) = 3.6574490995716378 * rm + -0.03649674695668263$, 此时loss是: 58.37957872941864
在第104步, 我们获得了函数 $f(rm) = 3.6587583892576827 * rm + -0.044825293158469864$, 此时loss是: 58.37247169191231
在第105步, 我们获得了函数 $f(rm) = 3.660067364095873 * rm + -0.05315183657589974$, 此时loss是: 58.36536807208584
在第106步, 我们获得了函数 $f(rm) = 3.661376024161921 * rm + -0.061476377690586455$, 此时loss是: 58.35826786829567
在第107步, 我们获得了函数 $f(rm) = 3.6626843695315205 * rm + -0.06979891698402824$, 此时loss是: 58.351171078899085
在第108步, 我们获得了函数 $f(rm) = 3.663992400280348 * rm + -0.07811945493760751$, 此时loss是: 58.34407770225413
在第109步, 我们获得了函数 $f(rm) = 3.66530011648406 * rm + -0.08643799203259109$, 此时loss是: 58.3369877367197
在第110步, 我们获得了函数 $f(rm) = 3.6666075182182962 * rm + -0.09475452875012992$, 此时loss是: 58.329901180655284
在第111步, 我们获得了函数 $f(rm) = 3.6679146055586784 * rm + -0.1030690655712592$, 此时loss是: 58.322818032421566
在第112步, 我们获得了函数 $f(rm) = 3.669221378580809 * rm + -0.11138160297689871$, 此时loss是: 58.31573829037957
在第113步, 我们获得了函数 $f(rm) = 3.6705278373602734 * rm + -0.11969214144785235$, 此时loss是: 58.30866195289136

在第114步, 我们获得了函数 $f(\text{rm}) = 3.6718339819726378 * \text{rm} + -0.1280006814648085$, 此时loss是: 58.301589018319746
在第115步, 我们获得了函数 $f(\text{rm}) = 3.6731398124934507 * \text{rm} + -0.1363072235083399$, 此时loss是: 58.29451948502828
在第116步, 我们获得了函数 $f(\text{rm}) = 3.6744453289982424 * \text{rm} + -0.14461176805890374$, 此时loss是: 58.28745335138133
在第117步, 我们获得了函数 $f(\text{rm}) = 3.675750531562525 * \text{rm} + -0.1529143155968417$, 此时loss是: 58.28039061574417
在第118步, 我们获得了函数 $f(\text{rm}) = 3.677055420261792 * \text{rm} + -0.1612148666023799$, 此时loss是: 58.27333127648255
在第119步, 我们获得了函数 $f(\text{rm}) = 3.6783599951715202 * \text{rm} + -0.169513421555629$, 此时loss是: 58.266275331963335
在第120步, 我们获得了函数 $f(\text{rm}) = 3.679664256367167 * \text{rm} + -0.17780998093658415$, 此时loss是: 58.25922278055401
在第121步, 我们获得了函数 $f(\text{rm}) = 3.680968203924172 * \text{rm} + -0.1861045452251252$, 此时loss是: 58.2521736206228
在第122步, 我们获得了函数 $f(\text{rm}) = 3.682271837917956 * \text{rm} + -0.19439711490101658$, 此时loss是: 58.24512785053893
在第123步, 我们获得了函数 $f(\text{rm}) = 3.683575158423923 * \text{rm} + -0.20268769044390728$, 此时loss是: 58.238085468672104
在第124步, 我们获得了函数 $f(\text{rm}) = 3.6848781655174574 * \text{rm} + -0.21097627233333097$, 此时loss是: 58.23104647339315
在第125步, 我们获得了函数 $f(\text{rm}) = 3.686180859273927 * \text{rm} + -0.219262861048706$, 此时loss是: 58.224010863073275
在第126步, 我们获得了函数 $f(\text{rm}) = 3.68748323976868 * \text{rm} + -0.22754745706933543$, 此时loss是: 58.21697863608494
在第127步, 我们获得了函数 $f(\text{rm}) = 3.6887853070770475 * \text{rm} + -0.23583006087440708$, 此时loss是: 58.209949790801
在第128步, 我们获得了函数 $f(\text{rm}) = 3.690087061274342 * \text{rm} + -0.2441106729429936$, 此时loss是: 58.202924325595255
在第129步, 我们获得了函数 $f(\text{rm}) = 3.691388502435858 * \text{rm} + -0.2523892937540523$, 此时loss是: 58.19590223884235
在第130步, 我们获得了函数 $f(\text{rm}) = 3.6926896306368726 * \text{rm} + -0.2606659237864254$, 此时loss是: 58.18888352891752
在第131步, 我们获得了函数 $f(\text{rm}) = 3.693990445952643 * \text{rm} + -0.26894056351883994$, 此时loss是: 58.181868194196966
在第132步, 我们获得了函数 $f(\text{rm}) = 3.69529094845841 * \text{rm} + -0.27721321342990796$, 此时loss是: 58.17485623305759
在第133步, 我们获得了函数 $f(\text{rm}) = 3.6965911382293952 * \text{rm} + -0.28548387399812625$, 此时loss是: 58.167847643877074
在第134步, 我们获得了函数 $f(\text{rm}) = 3.6978910153408036 * \text{rm} + -0.2937525457018765$, 此时loss是: 58.16084242503379
在第135步, 我们获得了函数 $f(\text{rm}) = 3.699190579867821 * \text{rm} + -0.3020192290194254$, 此时loss是: 58.15384057490719
在第136步, 我们获得了函数 $f(\text{rm}) = 3.7004898318856143 * \text{rm} + -0.31028392442892494$, 此时loss是: 58.146842091877204

在第137步, 我们获得了函数 $f(\text{rm}) = 3.7017887714693347 * \text{rm} + -0.31854663240841163$, 此时loss是: 58.13984697432448
在第138步, 我们获得了函数 $f(\text{rm}) = 3.7030873986941124 * \text{rm} + -0.32680735343580747$, 此时loss是: 58.13285522063086
在第139步, 我们获得了函数 $f(\text{rm}) = 3.704385713635062 * \text{rm} + -0.3350660879889192$, 此时loss是: 58.12586682917861
在第140步, 我们获得了函数 $f(\text{rm}) = 3.7056837163672793 * \text{rm} + -0.3433228365454387$, 此时loss是: 58.11888179835079
在第141步, 我们获得了函数 $f(\text{rm}) = 3.7069814069658413 * \text{rm} + -0.35157759958294327$, 此时loss是: 58.111900126531566
在第142步, 我们获得了函数 $f(\text{rm}) = 3.7082787855058075 * \text{rm} + -0.3598303775788951$, 此时loss是: 58.10492181210527
在第143步, 我们获得了函数 $f(\text{rm}) = 3.7095758520622195 * \text{rm} + -0.3680811710106415$, 此时loss是: 58.097946853457664
在第144步, 我们获得了函数 $f(\text{rm}) = 3.710872606710101 * \text{rm} + -0.3763299803554152$, 此时loss是: 58.09097524897488
在第145步, 我们获得了函数 $f(\text{rm}) = 3.712169049524457 * \text{rm} + -0.3845768060903341$, 此时loss是: 58.08400699704399
在第146步, 我们获得了函数 $f(\text{rm}) = 3.713465180580274 * \text{rm} + -0.3928216486924014$, 此时loss是: 58.07704209605279
在第147步, 我们获得了函数 $f(\text{rm}) = 3.714760999952523 * \text{rm} + -0.40106450863850535$, 此时loss是: 58.070080544389796
在第148步, 我们获得了函数 $f(\text{rm}) = 3.7160565077161545 * \text{rm} + -0.4093053864054199$, 此时loss是: 58.063122340444444
在第149步, 我们获得了函数 $f(\text{rm}) = 3.7173517039461017 * \text{rm} + -0.4175442824698042$, 此时loss是: 58.056167482606824
在第150步, 我们获得了函数 $f(\text{rm}) = 3.718646588717279 * \text{rm} + -0.42578119730820263$, 此时loss是: 58.04921596926785
在第151步, 我们获得了函数 $f(\text{rm}) = 3.719941162104585 * \text{rm} + -0.4340161313970451$, 此时loss是: 58.04226779881924
在第152步, 我们获得了函数 $f(\text{rm}) = 3.7212354241828978 * \text{rm} + -0.44224908521264694$, 此时loss是: 58.03532296965331
在第153步, 我们获得了函数 $f(\text{rm}) = 3.722529375027079 * \text{rm} + -0.450480059231209$, 此时loss是: 58.028381480163304
在第154步, 我们获得了函数 $f(\text{rm}) = 3.7238230147119715 * \text{rm} + -0.4587090539288175$, 此时loss是: 58.02144332874341
在第155步, 我们获得了函数 $f(\text{rm}) = 3.725116343312402 * \text{rm} + -0.4669360697814442$, 此时loss是: 58.01450851378816
在第156步, 我们获得了函数 $f(\text{rm}) = 3.726409360903175 * \text{rm} + -0.4751611072649466$, 此时loss是: 58.00757703369329
在第157步, 我们获得了函数 $f(\text{rm}) = 3.7277020675590813 * \text{rm} + -0.4833841668550674$, 此时loss是: 58.00064888685489
在第158步, 我们获得了函数 $f(\text{rm}) = 3.7289944633548924 * \text{rm} + -0.49160524902743513$, 此时loss是: 57.993724071670265
在第159步, 我们获得了函数 $f(\text{rm}) = 3.7302865483653616 * \text{rm} + -0.49982435425756394$, 此时loss是: 57.98680258653712

在第160步, 我们获得了函数 $f(\text{rm}) = 3.7315783226652224 * \text{rm} + -0.5080414830208537$, 此时loss是: 57.97988442985405
在第161步, 我们获得了函数 $f(\text{rm}) = 3.7328697863291938 * \text{rm} + -0.5162566357925897$, 此时loss是: 57.97296960002063
在第162步, 我们获得了函数 $f(\text{rm}) = 3.7341609394319746 * \text{rm} + -0.5244698130479433$, 此时loss是: 57.966058095436814
在第163步, 我们获得了函数 $f(\text{rm}) = 3.7354517820482465 * \text{rm} + -0.5326810152619711$, 此时loss是: 57.95914991450376
在第164步, 我们获得了函数 $f(\text{rm}) = 3.7367423142526723 * \text{rm} + -0.540890242909616$, 此时loss是: 57.95224505562294
在第165步, 我们获得了函数 $f(\text{rm}) = 3.738032536119898 * \text{rm} + -0.5490974964657064$, 此时loss是: 57.94534351719692
在第166步, 我们获得了函数 $f(\text{rm}) = 3.73932244772455 * \text{rm} + -0.5573027764049566$, 此时loss是: 57.938445297628995
在第167步, 我们获得了函数 $f(\text{rm}) = 3.740612049141239 * \text{rm} + -0.5655060832019667$, 此时loss是: 57.93155039532304
在第168步, 我们获得了函数 $f(\text{rm}) = 3.741901340444557 * \text{rm} + -0.5737074173312225$, 此时loss是: 57.92465880868402
在第169步, 我们获得了函数 $f(\text{rm}) = 3.743190321709076 * \text{rm} + -0.5819067792670962$, 此时loss是: 57.91777053611723
在第170步, 我们获得了函数 $f(\text{rm}) = 3.7444789930093534 * \text{rm} + -0.5901041694838454$, 此时loss是: 57.910885576029216
在第171步, 我们获得了函数 $f(\text{rm}) = 3.7457673544199257 * \text{rm} + -0.5982995884556139$, 此时loss是: 57.904003926826846
在第172步, 我们获得了函数 $f(\text{rm}) = 3.7470554060153134 * \text{rm} + -0.6064930366564315$, 此时loss是: 57.89712558691814
在第173步, 我们获得了函数 $f(\text{rm}) = 3.748343147870018 * \text{rm} + -0.6146845145602139$, 此时loss是: 57.89025055471149
在第174步, 我们获得了函数 $f(\text{rm}) = 3.7496305800585246 * \text{rm} + -0.6228740226407627$, 此时loss是: 57.8833788286165
在第175步, 我们获得了函数 $f(\text{rm}) = 3.7509177026552982 * \text{rm} + -0.631061561371766$, 此时loss是: 57.87651040704318
在第176步, 我们获得了函数 $f(\text{rm}) = 3.752204515734788 * \text{rm} + -0.6392471312267977$, 此时loss是: 57.86964528840248
在第177步, 我们获得了函数 $f(\text{rm}) = 3.7534910193714235 * \text{rm} + -0.6474307326793178$, 此时loss是: 57.86278347110597
在第178步, 我们获得了函数 $f(\text{rm}) = 3.754777213639617 * \text{rm} + -0.6556123662026726$, 此时loss是: 57.85592495356614
在第179步, 我们获得了函数 $f(\text{rm}) = 3.7560630986137635 * \text{rm} + -0.6637920322700944$, 此时loss是: 57.84906973419616
在第180步, 我们获得了函数 $f(\text{rm}) = 3.7573486743682403 * \text{rm} + -0.6719697313547017$, 此时loss是: 57.84221781141006
在第181步, 我们获得了函数 $f(\text{rm}) = 3.7586339409774046 * \text{rm} + -0.6801454639294996$, 此时loss是: 57.83536918362243
在第182步, 我们获得了函数 $f(\text{rm}) = 3.7599188985155987 * \text{rm} + -0.6883192304673789$, 此时loss是: 57.82852384924888

在第183步, 我们获得了函数 $f(rm) = 3.761203547057145 * rm + -0.6964910314411169$, 此时loss是: 57.821681806705634
在第184步, 我们获得了函数 $f(rm) = 3.7624878866763485 * rm + -0.7046608673233775$, 此时loss是: 57.81484305440947
在第185步, 我们获得了函数 $f(rm) = 3.7637719174474964 * rm + -0.7128287385867106$, 此时loss是: 57.80800759077852
在第186步, 我们获得了函数 $f(rm) = 3.765055639444859 * rm + -0.7209946457035524$, 此时loss是: 57.80117541423105
在第187步, 我们获得了函数 $f(rm) = 3.766339052742686 * rm + -0.729158589146226$, 此时loss是: 57.794346523186356
在第188步, 我们获得了函数 $f(rm) = 3.767622157415213 * rm + -0.7373205693869401$, 此时loss是: 57.787520916064594
在第189步, 我们获得了函数 $f(rm) = 3.7689049535366546 * rm + -0.7454805868977907$, 此时loss是: 57.78069859128654
在第190步, 我们获得了函数 $f(rm) = 3.7701874411812097 * rm + -0.7536386421507598$, 此时loss是: 57.7738795472737
在第191步, 我们获得了函数 $f(rm) = 3.7714696204230576 * rm + -0.761794735617716$, 此时loss是: 57.76706378244849
在第192步, 我们获得了函数 $f(rm) = 3.7727514913363605 * rm + -0.7699488677704144$, 此时loss是: 57.76025129523389
在第193步, 我们获得了函数 $f(rm) = 3.7740330539952636 * rm + -0.7781010390804967$, 此时loss是: 57.75344208405371
在第194步, 我们获得了函数 $f(rm) = 3.7753143084738934 * rm + -0.7862512500194909$, 此时loss是: 57.74663614733276
在第195步, 我们获得了函数 $f(rm) = 3.7765952548463586 * rm + -0.7943995010588123$, 此时loss是: 57.73983348349622
在第196步, 我们获得了函数 $f(rm) = 3.77787589318675 * rm + -0.8025457926697622$, 此时loss是: 57.73303409097014
在第197步, 我们获得了函数 $f(rm) = 3.779156223569141 * rm + -0.8106901253235288$, 此时loss是: 57.726237968181614
在第198步, 我们获得了函数 $f(rm) = 3.7804362460675867 * rm + -0.818832499491187$, 此时loss是: 57.71944511355817
在第199步, 我们获得了函数 $f(rm) = 3.7817159607561246 * rm + -0.8269729156436982$, 此时loss是: 57.71265552552815
在第200步, 我们获得了函数 $f(rm) = 3.782995367708775 * rm + -0.8351113742519111$, 此时loss是: 57.70586920252069
在第201步, 我们获得了函数 $f(rm) = 3.7842744669995394 * rm + -0.8432478757865605$, 此时loss是: 57.69908614296568
在第202步, 我们获得了函数 $f(rm) = 3.785553258702402 * rm + -0.8513824207182684$, 此时loss是: 57.692306345293844
在第203步, 我们获得了函数 $f(rm) = 3.786831742891329 * rm + -0.8595150095175436$, 此时loss是: 57.685529807936476
在第204步, 我们获得了函数 $f(rm) = 3.788109919640269 * rm + -0.8676456426547816$, 此时loss是: 57.67875652932588
在第205步, 我们获得了函数 $f(rm) = 3.789387789023154 * rm + -0.8757743206002649$, 此时loss是: 57.671986507894836

在第206步, 我们获得了函数 $f(\text{rm}) = 3.7906653511138946 * \text{rm} + -0.883901043824163$, 此时loss是: 57.665219742077035
在第207步, 我们获得了函数 $f(\text{rm}) = 3.791942605986388 * \text{rm} + -0.892025812796532$, 此时loss是: 57.65845623030697
在第208步, 我们获得了函数 $f(\text{rm}) = 3.793219553714511 * \text{rm} + -0.9001486279873153$, 此时loss是: 57.65169597101974
在第209步, 我们获得了函数 $f(\text{rm}) = 3.7944961943721234 * \text{rm} + -0.9082694898663431$, 此时loss是: 57.64493896265128
在第210步, 我们获得了函数 $f(\text{rm}) = 3.795772528033067 * \text{rm} + -0.916388398903333$, 此时loss是: 57.638185203638294
在第211步, 我们获得了函数 $f(\text{rm}) = 3.7970485547711665 * \text{rm} + -0.924505355567889$, 此时loss是: 57.63143469241822
在第212步, 我们获得了函数 $f(\text{rm}) = 3.798324274660228 * \text{rm} + -0.9326203603295028$, 此时loss是: 57.62468742742922
在第213步, 我们获得了函数 $f(\text{rm}) = 3.7995996877740392 * \text{rm} + -0.9407334136575527$, 此时loss是: 57.61794340711019
在第214步, 我们获得了函数 $f(\text{rm}) = 3.800874794186373 * \text{rm} + -0.9488445160213044$, 此时loss是: 57.61120262990072
在第215步, 我们获得了函数 $f(\text{rm}) = 3.802149593970981 * \text{rm} + -0.9569536678899108$, 此时loss是: 57.6044650942415
在第216步, 我们获得了函数 $f(\text{rm}) = 3.803424087201599 * \text{rm} + -0.9650608697324117$, 此时loss是: 57.59773079857355
在第217步, 我们获得了函数 $f(\text{rm}) = 3.8046982739519453 * \text{rm} + -0.9731661220177343$, 此时loss是: 57.59099974133877
在第218步, 我们获得了函数 $f(\text{rm}) = 3.8059721542957194 * \text{rm} + -0.9812694252146933$, 此时loss是: 57.58427192097998
在第219步, 我们获得了函数 $f(\text{rm}) = 3.807245728306604 * \text{rm} + -0.9893707797919902$, 此时loss是: 57.57754733594043
在第220步, 我们获得了函数 $f(\text{rm}) = 3.8085189960582633 * \text{rm} + -0.9974701862182139$, 此时loss是: 57.57082598466448
在第221步, 我们获得了函数 $f(\text{rm}) = 3.809791957624344 * \text{rm} + -1.005567644961841$, 此时loss是: 57.56410786559683
在第222步, 我们获得了函数 $f(\text{rm}) = 3.811064613078475 * \text{rm} + -1.0136631564912348$, 此时loss是: 57.55739297718338
在第223步, 我们获得了函数 $f(\text{rm}) = 3.8123369624942693 * \text{rm} + -1.0217567212746466$, 此时loss是: 57.55068131787052
在第224步, 我们获得了函数 $f(\text{rm}) = 3.813609005945319 * \text{rm} + -1.0298483397802145$, 此时loss是: 57.54397288610525
在第225步, 我们获得了函数 $f(\text{rm}) = 3.814880743505201 * \text{rm} + -1.0379380124759645$, 此时loss是: 57.5372676803357
在第226步, 我们获得了函数 $f(\text{rm}) = 3.816152175247473 * \text{rm} + -1.0460257398298098$, 此时loss是: 57.53056569901038
在第227步, 我们获得了函数 $f(\text{rm}) = 3.8174233012456757 * \text{rm} + -1.0541115223095514$, 此时loss是: 57.5238669405787
在第228步, 我们获得了函数 $f(\text{rm}) = 3.818694121573333 * \text{rm} + -1.0621953603828773$, 此时loss是: 57.517171403490906

在第229步, 我们获得了函数 $f(\text{rm}) = 3.8199646363039492 * \text{rm} + -1.0702772545173636$, 此时loss是: 57.51047908619787
在第230步, 我们获得了函数 $f(\text{rm}) = 3.8212348455110123 * \text{rm} + -1.0783572051804735$, 此时loss是: 57.50378998715117
在第231步, 我们获得了函数 $f(\text{rm}) = 3.822504749267992 * \text{rm} + -1.086435212839558$, 此时loss是: 57.49710410480318
在第232步, 我们获得了函数 $f(\text{rm}) = 3.823774347648341 * \text{rm} + -1.0945112779618558$, 此时loss是: 57.49042143760725
在第233步, 我们获得了函数 $f(\text{rm}) = 3.825043640725494 * \text{rm} + -1.1025854010144926$, 此时loss是: 57.48374198401695
在第234步, 我们获得了函数 $f(\text{rm}) = 3.826312628572867 * \text{rm} + -1.1106575824644829$, 此时loss是: 57.477065742487135
在第235步, 我们获得了函数 $f(\text{rm}) = 3.82758131126386 * \text{rm} + -1.118727822778728$, 此时loss是: 57.47039271147304
在第236步, 我们获得了函数 $f(\text{rm}) = 3.828849688871854 * \text{rm} + -1.1267961224240175$, 此时loss是: 57.463722889430734
在第237步, 我们获得了函数 $f(\text{rm}) = 3.830117761470215 * \text{rm} + -1.1348624818670279$, 此时loss是: 57.457056274817276
在第238步, 我们获得了函数 $f(\text{rm}) = 3.8313855291322874 * \text{rm} + -1.1429269015743244$, 此时loss是: 57.45039286609003
在第239步, 我们获得了函数 $f(\text{rm}) = 3.8326529919313996 * \text{rm} + -1.1509893820123598$, 此时loss是: 57.443732661707465
在第240步, 我们获得了函数 $f(\text{rm}) = 3.8339201499408637 * \text{rm} + -1.1590499236474743$, 此时loss是: 57.43707566012861
在第241步, 我们获得了函数 $f(\text{rm}) = 3.8351870032339734 * \text{rm} + -1.1671085269458963$, 此时loss是: 57.43042185981323
在第242步, 我们获得了函数 $f(\text{rm}) = 3.836453551884004 * \text{rm} + -1.1751651923737418$, 此时loss是: 57.42377125922199
在第243步, 我们获得了函数 $f(\text{rm}) = 3.8377197959642144 * \text{rm} + -1.1832199203970153$, 此时loss是: 57.41712385681608
在第244步, 我们获得了函数 $f(\text{rm}) = 3.838985735547844 * \text{rm} + -1.1912727114816086$, 此时loss是: 57.41047965105758
在第245步, 我们获得了函数 $f(\text{rm}) = 3.8402513707081165 * \text{rm} + -1.199323566093302$, 此时loss是: 57.403838640409305
在第246步, 我们获得了函数 $f(\text{rm}) = 3.8415167015182368 * \text{rm} + -1.2073724846977631$, 此时loss是: 57.39720082333468
在第247步, 我们获得了函数 $f(\text{rm}) = 3.8427817280513934 * \text{rm} + -1.2154194677605483$, 此时loss是: 57.3905661982981
在第248步, 我们获得了函数 $f(\text{rm}) = 3.8440464503807563 * \text{rm} + -1.2234645157471016$, 此时loss是: 57.38393476376434
在第249步, 我们获得了函数 $f(\text{rm}) = 3.8453108685794777 * \text{rm} + -1.2315076291227551$, 此时loss是: 57.377306518199305
在第250步, 我们获得了函数 $f(\text{rm}) = 3.8465749827206928 * \text{rm} + -1.239548808352729$, 此时loss是: 57.370681460069385
在第251步, 我们获得了函数 $f(\text{rm}) = 3.847838792877519 * \text{rm} + -1.2475880539021318$, 此时loss是: 57.364059587841844

在第252步, 我们获得了函数 $f(rm) = 3.849102299123056 * rm + -1.25562536623596$, 此时loss是: 57.357440899984525
在第253步, 我们获得了函数 $f(rm) = 3.850365501530386 * rm + -1.2636607458190983$, 此时loss是: 57.3508253949662
在第254步, 我们获得了函数 $f(rm) = 3.851628400172574 * rm + -1.2716941931163197$, 此时loss是: 57.3442130712563
在第255步, 我们获得了函数 $f(rm) = 3.8528909951226664 * rm + -1.2797257085922853$, 此时loss是: 57.33760392732486
在第256步, 我们获得了函数 $f(rm) = 3.854153286453694 * rm + -1.2877552927115448$, 此时loss是: 57.330997961642865
在第257步, 我们获得了函数 $f(rm) = 3.855415274238667 * rm + -1.2957829459385355$, 此时loss是: 57.32439517268196
在第258步, 我们获得了函数 $f(rm) = 3.8566769585505813 * rm + -1.3038086687375836$, 此时loss是: 57.31779555891435
在第259步, 我们获得了函数 $f(rm) = 3.8579383394624136 * rm + -1.3118324615729036$, 此时loss是: 57.31119911881328
在第260步, 我们获得了函数 $f(rm) = 3.8591994170471224 * rm + -1.319854324908598$, 此时loss是: 57.304605850852546
在第261步, 我们获得了函数 $f(rm) = 3.8604601913776495 * rm + -1.3278742592086583$, 此时loss是: 57.29801575350669
在第262步, 我们获得了函数 $f(rm) = 3.8617206625269205 * rm + -1.3358922649369638$, 此时loss是: 57.291428825250925
在第263步, 我们获得了函数 $f(rm) = 3.8629808305678397 * rm + -1.3439083425572826$, 此时loss是: 57.284845064561395
在第264步, 我们获得了函数 $f(rm) = 3.8642406955732986 * rm + -1.3519224925332711$, 此时loss是: 57.27826446991482
在第265步, 我们获得了函数 $f(rm) = 3.865500257616168 * rm + -1.3599347153284744$, 此时loss是: 57.271687039788674
在第266步, 我们获得了函数 $f(rm) = 3.866759516769301 * rm + -1.367945011406326$, 此时loss是: 57.26511277266119
在第267步, 我们获得了函数 $f(rm) = 3.868018473105536 * rm + -1.375953381230148$, 此时loss是: 57.2585416670114
在第268步, 我们获得了函数 $f(rm) = 3.8692771266976904 * rm + -1.383959825263151$, 此时loss是: 57.25197372131893
在第269步, 我们获得了函数 $f(rm) = 3.8705354776185668 * rm + -1.3919643439684344$, 此时loss是: 57.24540893406411
在第270步, 我们获得了函数 $f(rm) = 3.871793525940949 * rm + -1.3999669378089858$, 此时loss是: 57.23884730372823
在第271步, 我们获得了函数 $f(rm) = 3.873051271737603 * rm + -1.407967607247682$, 此时loss是: 57.232288828793074
在第272步, 我们获得了函数 $f(rm) = 3.8743087150812783 * rm + -1.4159663527472885$, 此时loss是: 57.22573350774129
在第273步, 我们获得了函数 $f(rm) = 3.875565856044707 * rm + -1.423963174770459$, 此时loss是: 57.219181339056256
在第274步, 我们获得了函数 $f(rm) = 3.876822694700602 * rm + -1.4319580737797364$, 此时loss是: 57.21263232122192

在第275步, 我们获得了函数 $f(\text{rm}) = 3.8780792311216605 * \text{rm} + -1.439951050237552$, 此时loss是: 57.206086452723206
在第276步, 我们获得了函数 $f(\text{rm}) = 3.8793354653805623 * \text{rm} + -1.4479421046062262$, 此时loss是: 57.19954373204562
在第277步, 我们获得了函数 $f(\text{rm}) = 3.880591397549968 * \text{rm} + -1.4559312373479683$, 此时loss是: 57.19300415767536
在第278步, 我们获得了函数 $f(\text{rm}) = 3.8818470277025217 * \text{rm} + -1.463918448924876$, 此时loss是: 57.1864677280994
在第279步, 我们获得了函数 $f(\text{rm}) = 3.8831023559108506 * \text{rm} + -1.4719037397989363$, 此时loss是: 57.17993444180557
在第280步, 我们获得了函数 $f(\text{rm}) = 3.8843573822475634 * \text{rm} + -1.4798871104320248$, 此时loss是: 57.17340429728216
在第281步, 我们获得了函数 $f(\text{rm}) = 3.885612106785253 * \text{rm} + -1.4878685612859064$, 此时loss是: 57.16687729301849
在第282步, 我们获得了函数 $f(\text{rm}) = 3.8868665295964924 * \text{rm} + -1.4958480928222346$, 此时loss是: 57.16035342750429
在第283步, 我们获得了函数 $f(\text{rm}) = 3.8881206507538395 * \text{rm} + -1.5038257055025521$, 此时loss是: 57.15383269923028
在第284步, 我们获得了函数 $f(\text{rm}) = 3.889374470329832 * \text{rm} + -1.5118013997882906$, 此时loss是: 57.14731510668777
在第285步, 我们获得了函数 $f(\text{rm}) = 3.8906279883969934 * \text{rm} + -1.519775176140771$, 此时loss是: 57.1408006483688
在第286步, 我们获得了函数 $f(\text{rm}) = 3.891881205027828 * \text{rm} + -1.5277470350212026$, 此时loss是: 57.134289322766264
在第287步, 我们获得了函数 $f(\text{rm}) = 3.893134120294823 * \text{rm} + -1.5357169768906844$, 此时loss是: 57.127781128373485
在第288步, 我们获得了函数 $f(\text{rm}) = 3.8943867342704466 * \text{rm} + -1.5436850022102044$, 此时loss是: 57.121276063684874
在第289步, 我们获得了函数 $f(\text{rm}) = 3.895639047027153 * \text{rm} + -1.5516511114406397$, 此时loss是: 57.11477412719541
在第290步, 我们获得了函数 $f(\text{rm}) = 3.896891058637376 * \text{rm} + -1.5596153050427566$, 此时loss是: 57.10827531740078
在第291步, 我们获得了函数 $f(\text{rm}) = 3.8981427691735333 * \text{rm} + -1.5675775834772105$, 此时loss是: 57.10177963279728
在第292步, 我们获得了函数 $f(\text{rm}) = 3.899394178708024 * \text{rm} + -1.5755379472045463$, 此时loss是: 57.09528707188209
在第293步, 我们获得了函数 $f(\text{rm}) = 3.9006452873132313 * \text{rm} + -1.5834963966851978$, 此时loss是: 57.08879763315307
在第294步, 我们获得了函数 $f(\text{rm}) = 3.901896095061521 * \text{rm} + -1.5914529323794881$, 此时loss是: 57.08231131510892
在第295步, 我们获得了函数 $f(\text{rm}) = 3.90314660202524 * \text{rm} + -1.59940755474763$, 此时loss是: 57.07582811624877
在第296步, 我们获得了函数 $f(\text{rm}) = 3.904396808276718 * \text{rm} + -1.6073602642497256$, 此时loss是: 57.069348035072835
在第297步, 我们获得了函数 $f(\text{rm}) = 3.90564671388827 * \text{rm} + -1.6153110613457655$, 此时loss是: 57.06287107008159

在第298步, 我们获得了函数 $f(\text{rm}) = 3.9068963189321892 * \text{rm} + -1.6232599464956308$, 此时loss是: 57.05639721977684
在第299步, 我们获得了函数 $f(\text{rm}) = 3.908145623480756 * \text{rm} + -1.631206920159091$, 此时loss是: 57.049926482660496
在第300步, 我们获得了函数 $f(\text{rm}) = 3.90939462760623 * \text{rm} + -1.639151982795806$, 此时loss是: 57.04345885723562
在第301步, 我们获得了函数 $f(\text{rm}) = 3.910643331380854 * \text{rm} + -1.6470951348653249$, 此时loss是: 57.03699434200574
在第302步, 我们获得了函数 $f(\text{rm}) = 3.9118917348768556 * \text{rm} + -1.6550363768270857$, 此时loss是: 57.03053293547532
在第303步, 我们获得了函数 $f(\text{rm}) = 3.913139838166442 * \text{rm} + -1.6629757091404165$, 此时loss是: 57.024074636149265
在第304步, 我们获得了函数 $f(\text{rm}) = 3.9143876413218055 * \text{rm} + -1.6709131322645348$, 此时loss是: 57.01761944253352
在第305步, 我们获得了函数 $f(\text{rm}) = 3.91563514441512 * \text{rm} + -1.6788486466585475$, 此时loss是: 57.0111673531345
在第306步, 我们获得了函数 $f(\text{rm}) = 3.9168823475185417 * \text{rm} + -1.6867822527814513$, 此时loss是: 57.00471836645965
在第307步, 我们获得了函数 $f(\text{rm}) = 3.91812925070421 * \text{rm} + -1.6947139510921325$, 此时loss是: 56.99827248101642
在第308步, 我们获得了函数 $f(\text{rm}) = 3.919375854044247 * \text{rm} + -1.7026437420493672$, 此时loss是: 56.991829695313946
在第309步, 我们获得了函数 $f(\text{rm}) = 3.9206221576107563 * \text{rm} + -1.710571626111821$, 此时loss是: 56.985390007861426
在第310步, 我们获得了函数 $f(\text{rm}) = 3.9218681614758264 * \text{rm} + -1.7184976037380486$, 此时loss是: 56.978953417168896
在第311步, 我们获得了函数 $f(\text{rm}) = 3.9231138657115263 * \text{rm} + -1.7264216753864958$, 此时loss是: 56.97251992174732
在第312步, 我们获得了函数 $f(\text{rm}) = 3.92435927038991 * \text{rm} + -1.734343841515497$, 此时loss是: 56.966089520108085
在第313步, 我们获得了函数 $f(\text{rm}) = 3.925604375583011 * \text{rm} + -1.742264102583277$, 此时loss是: 56.959662210763426
在第314步, 我们获得了函数 $f(\text{rm}) = 3.9268491813628477 * \text{rm} + -1.75018245904795$, 此时loss是: 56.95323799222641
在第315步, 我们获得了函数 $f(\text{rm}) = 3.9280936878014208 * \text{rm} + -1.7580989113675203$, 此时loss是: 56.946816863010554
在第316步, 我们获得了函数 $f(\text{rm}) = 3.9293378949707134 * \text{rm} + -1.7660134599998818$, 此时loss是: 56.9403988216304
在第317步, 我们获得了函数 $f(\text{rm}) = 3.930581802942692 * \text{rm} + -1.7739261054028186$, 此时loss是: 56.933983866601025
在第318步, 我们获得了函数 $f(\text{rm}) = 3.931825411789305 * \text{rm} + -1.7818368480340048$, 此时loss是: 56.92757199643819
在第319步, 我们获得了函数 $f(\text{rm}) = 3.933068721582483 * \text{rm} + -1.789745688351004$, 此时loss是: 56.92116320965844
在第320步, 我们获得了函数 $f(\text{rm}) = 3.9343117323941414 * \text{rm} + -1.7976526268112698$, 此时loss是: 56.91475750477895

在第321步, 我们获得了函数 $f(\text{rm}) = 3.9355544442961765 * \text{rm} + -1.8055576638721464$, 此时loss是: 56.908354880317766
在第322步, 我们获得了函数 $f(\text{rm}) = 3.9367968573604672 * \text{rm} + -1.8134607999908674$, 此时loss是: 56.901955334793556
在第323步, 我们获得了函数 $f(\text{rm}) = 3.9380389716588757 * \text{rm} + -1.8213620356245568$, 此时loss是: 56.89555886672573
在第324步, 我们获得了函数 $f(\text{rm}) = 3.9392807872632476 * \text{rm} + -1.8292613712302284$, 此时loss是: 56.88916547463426
在第325步, 我们获得了函数 $f(\text{rm}) = 3.94052230424541 * \text{rm} + -1.8371588072647862$, 此时loss是: 56.88277515703998
在第326步, 我们获得了函数 $f(\text{rm}) = 3.941763522677174 * \text{rm} + -1.8450543441850245$, 此时loss是: 56.87638791246453
在第327步, 我们获得了函数 $f(\text{rm}) = 3.9430044426303317 * \text{rm} + -1.8529479824476276$, 此时loss是: 56.87000373943005
在第328步, 我们获得了函数 $f(\text{rm}) = 3.9442450641766595 * \text{rm} + -1.86083972250917$, 此时loss是: 56.863622636459375
在第329步, 我们获得了函数 $f(\text{rm}) = 3.9454853873879157 * \text{rm} + -1.8687295648261162$, 此时loss是: 56.857244602076335
在第330步, 我们获得了函数 $f(\text{rm}) = 3.946725412335842 * \text{rm} + -1.8766175098548215$, 此时loss是: 56.85086963480518
在第331步, 我们获得了函数 $f(\text{rm}) = 3.9479651390921617 * \text{rm} + -1.8845035580515308$, 此时loss是: 56.84449773317102
在第332步, 我们获得了函数 $f(\text{rm}) = 3.9492045677285827 * \text{rm} + -1.8923877098723796$, 此时loss是: 56.83812889569952
在第333步, 我们获得了函数 $f(\text{rm}) = 3.950443698316794 * \text{rm} + -1.900269965773394$, 此时loss是: 56.831763120917316
在第334步, 我们获得了函数 $f(\text{rm}) = 3.9516825309284678 * \text{rm} + -1.9081503262104897$, 此时loss是: 56.82540040735158
在第335步, 我们获得了函数 $f(\text{rm}) = 3.9529210656352585 * \text{rm} + -1.9160287916394736$, 此时loss是: 56.819040753530196
在第336步, 我们获得了函数 $f(\text{rm}) = 3.9541593025088053 * \text{rm} + -1.9239053625160423$, 此时loss是: 56.81268415798171
在第337步, 我们获得了函数 $f(\text{rm}) = 3.955397241620729 * \text{rm} + -1.9317800392957833$, 此时loss是: 56.80633061923538
在第338步, 我们获得了函数 $f(\text{rm}) = 3.9566348830426317 * \text{rm} + -1.9396528224341743$, 此时loss是: 56.79998013582147
在第339步, 我们获得了函数 $f(\text{rm}) = 3.9578722268460997 * \text{rm} + -1.9475237123865834$, 此时loss是: 56.793632706270515
在第340步, 我们获得了函数 $f(\text{rm}) = 3.9591092731027033 * \text{rm} + -1.9553927096082695$, 此时loss是: 56.78728832911408
在第341步, 我们获得了函数 $f(\text{rm}) = 3.9603460218839928 * \text{rm} + -1.9632598145543818$, 此时loss是: 56.780947002884155
在第342步, 我们获得了函数 $f(\text{rm}) = 3.961582473261504 * \text{rm} + -1.97112502767996$, 此时loss是: 56.7746087261137
在第343步, 我们获得了函数 $f(\text{rm}) = 3.9628186273067536 * \text{rm} + -1.9789883494399343$, 此时loss是: 56.76827349733627

在第344步, 我们获得了函数 $f(\text{rm}) = 3.9640544840912413 * \text{rm} + -1.986849780289126$, 此时loss是: 56.76194131508606
在第345步, 我们获得了函数 $f(\text{rm}) = 3.9652900436864513 * \text{rm} + -1.9947093206822466$, 此时loss是: 56.755612177898065
在第346步, 我们获得了函数 $f(\text{rm}) = 3.9665253061638484 * \text{rm} + -2.0025669710738985$, 此时loss是: 56.74928608430805
在第347步, 我们获得了函数 $f(\text{rm}) = 3.9677602715948814 * \text{rm} + -2.0104227319185743$, 此时loss是: 56.74296303285229
在第348步, 我们获得了函数 $f(\text{rm}) = 3.9689949400509814 * \text{rm} + -2.018276603670658$, 此时loss是: 56.73664302206785
在第349步, 我们获得了函数 $f(\text{rm}) = 3.970229311603563 * \text{rm} + -2.0261285867844236$, 此时loss是: 56.73032605049253
在第350步, 我们获得了函数 $f(\text{rm}) = 3.971463386324024 * \text{rm} + -2.0339786817140366$, 此时loss是: 56.72401211666486
在第351步, 我们获得了函数 $f(\text{rm}) = 3.972697164283743 * \text{rm} + -2.0418268889135525$, 此时loss是: 56.717701219123974
在第352步, 我们获得了函数 $f(\text{rm}) = 3.9739306455540837 * \text{rm} + -2.0496732088369183$, 此时loss是: 56.711393356409715
在第353步, 我们获得了函数 $f(\text{rm}) = 3.975163830206391 * \text{rm} + -2.0575176419379715$, 此时loss是: 56.705088527062834
在第354步, 我们获得了函数 $f(\text{rm}) = 3.9763967183119937 * \text{rm} + -2.0653601886704407$, 此时loss是: 56.69878672962447
在第355步, 我们获得了函数 $f(\text{rm}) = 3.9776293099422033 * \text{rm} + -2.073200849487945$, 此时loss是: 56.692487962636754
在第356步, 我们获得了函数 $f(\text{rm}) = 3.9788616051683126 * \text{rm} + -2.0810396248439944$, 此时loss是: 56.68619222464227
在第357步, 我们获得了函数 $f(\text{rm}) = 3.980093604061601 * \text{rm} + -2.0888765151919904$, 此时loss是: 56.67989951418448
在第358步, 我们获得了函数 $f(\text{rm}) = 3.9813253066933254 * \text{rm} + -2.096711520985225$, 此时loss是: 56.67360982980744
在第359步, 我们获得了函数 $f(\text{rm}) = 3.98255671313473 * \text{rm} + -2.104544642676882$, 此时loss是: 56.667323170055944
在第360步, 我们获得了函数 $f(\text{rm}) = 3.9837878234570407 * \text{rm} + -2.1123758807200352$, 此时loss是: 56.661039533475524
在第361步, 我们获得了函数 $f(\text{rm}) = 3.9850186377314656 * \text{rm} + -2.1202052355676493$, 此时loss是: 56.65475891861234
在第362步, 我们获得了函数 $f(\text{rm}) = 3.9862491560291957 * \text{rm} + -2.128032707672581$, 此时loss是: 56.648481324013346
在第363步, 我们获得了函数 $f(\text{rm}) = 3.9874793784214053 * \text{rm} + -2.135858297487578$, 此时loss是: 56.64220674822613
在第364步, 我们获得了函数 $f(\text{rm}) = 3.9887093049792517 * \text{rm} + -2.143682005465278$, 此时loss是: 56.635935189798936
在第365步, 我们获得了函数 $f(\text{rm}) = 3.989938935773875 * \text{rm} + -2.1515038320582116$, 此时loss是: 56.62966664728077
在第366步, 我们获得了函数 $f(\text{rm}) = 3.9911682708763974 * \text{rm} + -2.1593237777187992$, 此时loss是: 56.62340111922132

在第367步, 我们获得了函数 $f(rm) = 3.992397310357925 * rm + -$
2.1671418428993525, 此时loss是: 56.61713860417103
在第368步, 我们获得了函数 $f(rm) = 3.993626054289547 * rm + -$
2.1749580280520755, 此时loss是: 56.61087910068087
在第369步, 我们获得了函数 $f(rm) = 3.9948545027423337 * rm + -$
2.1827723336290625, 此时loss是: 56.60462260730273
在第370步, 我们获得了函数 $f(rm) = 3.996082655787341 * rm + -$
2.190584760082299, 此时loss是: 56.59836912258907
在第371步, 我们获得了函数 $f(rm) = 3.997310513495605 * rm + -$
2.1983953078636627, 此时loss是: 56.59211864509297
在第372步, 我们获得了函数 $f(rm) = 3.998538075938147 * rm + -$
2.2062039774249214, 此时loss是: 56.58587117336846
在第373步, 我们获得了函数 $f(rm) = 3.999765343185969 * rm + -$
2.2140107692177353, 此时loss是: 56.579626705969915
在第374步, 我们获得了函数 $f(rm) = 4.000992315310058 * rm + -$
2.221815683693656, 此时loss是: 56.57338524145276
在第375步, 我们获得了函数 $f(rm) = 4.002218992381384 * rm + -$
2.229618721304125, 此时loss是: 56.56714677837284
在第376步, 我们获得了函数 $f(rm) = 4.003445374470897 * rm + -$
2.237419882500478, 此时loss是: 56.560911315286845
在第377步, 我们获得了函数 $f(rm) = 4.004671461649534 * rm + -$
2.2452191677339393, 此时loss是: 56.554678850752154
在第378步, 我们获得了函数 $f(rm) = 4.005897253988211 * rm + -$
2.2530165774556266, 此时loss是: 56.548449383326755
在第379步, 我们获得了函数 $f(rm) = 4.00712275155783 * rm + -$
2.260812112116548, 此时loss是: 56.54222291156934
在第380步, 我们获得了函数 $f(rm) = 4.008347954429274 * rm + -$
2.268605772167604, 此时loss是: 56.535999434039475
在第381步, 我们获得了函数 $f(rm) = 4.00957286267341 * rm + -$
2.276397558059586, 此时loss是: 56.52977894929709
在第382步, 我们获得了函数 $f(rm) = 4.010797476361089 * rm + -$
2.284187470243177, 此时loss是: 56.52356145590317
在第383步, 我们获得了函数 $f(rm) = 4.012021795563142 * rm + -$
2.2919755091689527, 此时loss是: 56.517346952419054
在第384步, 我们获得了函数 $f(rm) = 4.013245820350385 * rm + -$
2.299761675287379, 此时loss是: 56.511135437407134
在第385步, 我们获得了函数 $f(rm) = 4.014469550793617 * rm + -$
2.3075459690488147, 此时loss是: 56.50492690942998
在第386步, 我们获得了函数 $f(rm) = 4.015692986963619 * rm + -$
2.315328390903509, 此时loss是: 56.49872136705151
在第387步, 我们获得了函数 $f(rm) = 4.016916128931157 * rm + -$
2.323108941301604, 此时loss是: 56.49251880883579
在第388步, 我们获得了函数 $f(rm) = 4.018138976766976 * rm + -$
2.3308876206931335, 此时loss是: 56.486319233347835
在第389步, 我们获得了函数 $f(rm) = 4.019361530541808 * rm + -$
2.338664429528022, 此时loss是: 56.48012263915323

在第390步, 我们获得了函数 $f(\text{rm}) = 4.020583790326367 * \text{rm} + -2.3464393682560867$, 此时loss是: 56.47392902481845
在第391步, 我们获得了函数 $f(\text{rm}) = 4.021805756191349 * \text{rm} + -2.3542124373270363$, 此时loss是: 56.46773838891031
在第392步, 我们获得了函数 $f(\text{rm}) = 4.023027428207433 * \text{rm} + -2.361983637190472$, 此时loss是: 56.461550729996745
在第393步, 我们获得了函数 $f(\text{rm}) = 4.024248806445282 * \text{rm} + -2.3697529682958853$, 此时loss是: 56.45536604664605
在第394步, 我们获得了函数 $f(\text{rm}) = 4.0254698909755415 * \text{rm} + -2.3775204310926616$, 此时loss是: 56.44918433742728
在第395步, 我们获得了函数 $f(\text{rm}) = 4.026690681868841 * \text{rm} + -2.385286026030077$, 此时loss是: 56.443005600910254
在第396步, 我们获得了函数 $f(\text{rm}) = 4.02791117919579 * \text{rm} + -2.3930497535573$, 此时loss是: 56.436829835665435
在第397步, 我们获得了函数 $f(\text{rm}) = 4.029131383026985 * \text{rm} + -2.400811614123391$, 此时loss是: 56.43065704026408
在第398步, 我们获得了函数 $f(\text{rm}) = 4.030351293433002 * \text{rm} + -2.408571608177302$, 此时loss是: 56.42448721327785
在第399步, 我们获得了函数 $f(\text{rm}) = 4.031570910484402 * \text{rm} + -2.416329736167878$, 此时loss是: 56.41832035327938
在第400步, 我们获得了函数 $f(\text{rm}) = 4.0327902342517294 * \text{rm} + -2.424085998543855$, 此时loss是: 56.412156458841864
在第401步, 我们获得了函数 $f(\text{rm}) = 4.03400926480551 * \text{rm} + -2.4318403957538615$, 此时loss是: 56.4059955285392
在第402步, 我们获得了函数 $f(\text{rm}) = 4.035228002216255 * \text{rm} + -2.439592928246418$, 此时loss是: 56.39983756094594
在第403步, 我们获得了函数 $f(\text{rm}) = 4.036446446554455 * \text{rm} + -2.4473435964699384$, 此时loss是: 56.393682554637515
在第404步, 我们获得了函数 $f(\text{rm}) = 4.037664597890588 * \text{rm} + -2.4550924008727266$, 此时loss是: 56.387530508189656
在第405步, 我们获得了函数 $f(\text{rm}) = 4.038882456295111 * \text{rm} + -2.4628393419029804$, 此时loss是: 56.38138142017919
在第406步, 我们获得了函数 $f(\text{rm}) = 4.040100021838468 * \text{rm} + -2.4705844200087888$, 此时loss是: 56.37523528918331
在第407步, 我们获得了函数 $f(\text{rm}) = 4.041317294591082 * \text{rm} + -2.478327635638134$, 此时loss是: 56.36909211378007
在第408步, 我们获得了函数 $f(\text{rm}) = 4.042534274623361 * \text{rm} + -2.486068989238889$, 此时loss是: 56.36295189254823
在第409步, 我们获得了函数 $f(\text{rm}) = 4.0437509620056975 * \text{rm} + -2.493808481258821$, 此时loss是: 56.35681462406706
在第410步, 我们获得了函数 $f(\text{rm}) = 4.044967356808466 * \text{rm} + -2.5015461121455878$, 此时loss是: 56.35068030691669
在第411步, 我们获得了函数 $f(\text{rm}) = 4.046183459102021 * \text{rm} + -2.509281882346741$, 此时loss是: 56.344548939677956
在第412步, 我们获得了函数 $f(\text{rm}) = 4.047399268956707 * \text{rm} + -2.5170157923097234$, 此时loss是: 56.33842052093211

在第413步, 我们获得了函数 $f(\text{rm}) = 4.048614786442844 * \text{rm} + -2.524747842481871$, 此时loss是: 56.332295049261305
在第414步, 我们获得了函数 $f(\text{rm}) = 4.04983001163074 * \text{rm} + -2.5324780333104115$, 此时loss是: 56.32617252324839
在第415步, 我们获得了函数 $f(\text{rm}) = 4.051044944590684 * \text{rm} + -2.540206365242466$, 此时loss是: 56.32005294147674
在第416步, 我们获得了函数 $f(\text{rm}) = 4.052259585392949 * \text{rm} + -2.547932838725047$, 此时loss是: 56.313936302530635
在第417步, 我们获得了函数 $f(\text{rm}) = 4.053473934107791 * \text{rm} + -2.5556574542050603$, 此时loss是: 56.307822604994826
在第418步, 我们获得了函数 $f(\text{rm}) = 4.05468799080545 * \text{rm} + -2.563380212129304$, 此时loss是: 56.30171184745488
在第419步, 我们获得了函数 $f(\text{rm}) = 4.055901755556144 * \text{rm} + -2.571101112944469$, 此时loss是: 56.29560402849701
在第420步, 我们获得了函数 $f(\text{rm}) = 4.057115228430083 * \text{rm} + -2.578820157097138$, 此时loss是: 56.2894991467079
在第421步, 我们获得了函数 $f(\text{rm}) = 4.058328409497453 * \text{rm} + -2.586537345033787$, 此时loss是: 56.283397200675395
在第422步, 我们获得了函数 $f(\text{rm}) = 4.059541298828425 * \text{rm} + -2.594252677200785$, 此时loss是: 56.277298188987544
在第423步, 我们获得了函数 $f(\text{rm}) = 4.060753896493155 * \text{rm} + -2.6019661540443924$, 此时loss是: 56.27120211023322
在第424步, 我们获得了函数 $f(\text{rm}) = 4.06196620256178 * \text{rm} + -2.6096777760107632$, 此时loss是: 56.26510896300216
在第425步, 我们获得了函数 $f(\text{rm}) = 4.0631782171044195 * \text{rm} + -2.617387543545944$, 此时loss是: 56.25901874588461
在第426步, 我们获得了函数 $f(\text{rm}) = 4.064389940191179 * \text{rm} + -2.6250954570958744$, 此时loss是: 56.252931457471334
在第427步, 我们获得了函数 $f(\text{rm}) = 4.065601371892145 * \text{rm} + -2.632801517106386$, 此时loss是: 56.2468470963542
在第428步, 我们获得了函数 $f(\text{rm}) = 4.066812512277388 * \text{rm} + -2.6405057240232037$, 此时loss是: 56.240765661125415
在第429步, 我们获得了函数 $f(\text{rm}) = 4.0680233614169605 * \text{rm} + -2.6482080782919453$, 此时loss是: 56.23468715037793
在第430步, 我们获得了函数 $f(\text{rm}) = 4.0692339193809 * \text{rm} + -2.655908580358121$, 此时loss是: 56.22861156270536
在第431步, 我们获得了函数 $f(\text{rm}) = 4.070444186239225 * \text{rm} + -2.663607230667134$, 此时loss是: 56.22253889670217
在第432步, 我们获得了函数 $f(\text{rm}) = 4.07165416206194 * \text{rm} + -2.6713040296642814$, 此时loss是: 56.21646915096326
在第433步, 我们获得了函数 $f(\text{rm}) = 4.07286384691903 * \text{rm} + -2.6789989777947514$, 此时loss是: 56.210402324084455
在第434步, 我们获得了函数 $f(\text{rm}) = 4.074073240880464 * \text{rm} + -2.6866920755036268$, 此时loss是: 56.20433841466184
在第435步, 我们获得了函数 $f(\text{rm}) = 4.075282344016194 * \text{rm} + -2.6943833232358823$, 此时loss是: 56.19827742129273

在第436步, 我们获得了函数 $f(\text{rm}) = 4.076491156396157 * \text{rm} + -2.702072721436386$, 此时loss是: 56.192219342574646
在第437步, 我们获得了函数 $f(\text{rm}) = 4.07769967809027 * \text{rm} + -2.709760270549899$, 此时loss是: 56.1861641771061
在第438步, 我们获得了函数 $f(\text{rm}) = 4.078907909168436 * \text{rm} + -2.7174459710210757$, 此时loss是: 56.180111923486066
在第439步, 我们获得了函数 $f(\text{rm}) = 4.080115849700539 * \text{rm} + -2.7251298232944627$, 此时loss是: 56.17406258031433
在第440步, 我们获得了函数 $f(\text{rm}) = 4.081323499756449 * \text{rm} + -2.732811827814501$, 此时loss是: 56.16801614619128
在第441步, 我们获得了函数 $f(\text{rm}) = 4.082530859406017 * \text{rm} + -2.740491985025523$, 此时loss是: 56.16197261971793
在第442步, 我们获得了函数 $f(\text{rm}) = 4.083737928719076 * \text{rm} + -2.7481702953717564$, 此时loss是: 56.15593199949614
在第443步, 我们获得了函数 $f(\text{rm}) = 4.0849447077654455 * \text{rm} + -2.7558467592973206$, 此时loss是: 56.14989428412821
在第444步, 我们获得了函数 $f(\text{rm}) = 4.086151196614925 * \text{rm} + -2.7635213772462284$, 此时loss是: 56.143859472217365
在第445步, 我们获得了函数 $f(\text{rm}) = 4.0873573953373015 * \text{rm} + -2.771194149662386$, 此时loss是: 56.1378275623673
在第446步, 我们获得了函数 $f(\text{rm}) = 4.08856330400234 * \text{rm} + -2.778865076989593$, 此时loss是: 56.131798553182406
在第447步, 我们获得了函数 $f(\text{rm}) = 4.089768922679792 * \text{rm} + -2.786534159671542$, 此时loss是: 56.12577244326794
在第448步, 我们获得了函数 $f(\text{rm}) = 4.090974251439393 * \text{rm} + -2.794201398151819$, 此时loss是: 56.119749231229484
在第449步, 我们获得了函数 $f(\text{rm}) = 4.092179290350858 * \text{rm} + -2.801866792873903$, 此时loss是: 56.11372891567362
在第450步, 我们获得了函数 $f(\text{rm}) = 4.093384039483888 * \text{rm} + -2.8095303442811677$, 此时loss是: 56.107711495207475
在第451步, 我们获得了函数 $f(\text{rm}) = 4.094588498908167 * \text{rm} + -2.8171920528168783$, 此时loss是: 56.10169696843877
在第452步, 我们获得了函数 $f(\text{rm}) = 4.095792668693362 * \text{rm} + -2.8248519189241947$, 此时loss是: 56.09568533397604
在第453步, 我们获得了函数 $f(\text{rm}) = 4.096996548909123 * \text{rm} + -2.8325099430461695$, 此时loss是: 56.08967659042828
在第454步, 我们获得了函数 $f(\text{rm}) = 4.098200139625083 * \text{rm} + -2.8401661256257493$, 此时loss是: 56.08367073640544
在第455步, 我们获得了函数 $f(\text{rm}) = 4.09940344091086 * \text{rm} + -2.847820467105774$, 此时loss是: 56.077667770517934
在第456步, 我们获得了函数 $f(\text{rm}) = 4.100606452836053 * \text{rm} + -2.855472967928977$, 此时loss是: 56.07166769137684
在第457步, 我们获得了函数 $f(\text{rm}) = 4.101809175470245 * \text{rm} + -2.8631236285379855$, 此时loss是: 56.065670497594084
在第458步, 我们获得了函数 $f(\text{rm}) = 4.103011608883002 * \text{rm} + -2.8707724493753193$, 此时loss是: 56.059676187782

在第459步, 我们获得了函数 $f(\text{rm}) = 4.104213753143876 * \text{rm} + -2.8784194308833926$, 此时loss是: 56.053684760553736
在第460步, 我们获得了函数 $f(\text{rm}) = 4.105415608322398 * \text{rm} + -2.8860645735045134$, 此时loss是: 56.04769621452321
在第461步, 我们获得了函数 $f(\text{rm}) = 4.1066171744880835 * \text{rm} + -2.893707877680883$, 此时loss是: 56.04171054830478
在第462步, 我们获得了函数 $f(\text{rm}) = 4.107818451710434 * \text{rm} + -2.901349343854596$, 此时loss是: 56.035727760513666
在第463步, 我们获得了函数 $f(\text{rm}) = 4.109019440058932 * \text{rm} + -2.9089889724676414$, 此时loss是: 56.0297478497656
在第464步, 我们获得了函数 $f(\text{rm}) = 4.110220139603044 * \text{rm} + -2.9166267639619017$, 此时loss是: 56.02377081467713
在第465步, 我们获得了函数 $f(\text{rm}) = 4.111420550412218 * \text{rm} + -2.924262718779153$, 此时loss是: 56.017796653865275
在第466步, 我们获得了函数 $f(\text{rm}) = 4.112620672555887 * \text{rm} + -2.931896837361064$, 此时loss是: 56.01182536594793
在第467步, 我们获得了函数 $f(\text{rm}) = 4.113820506103468 * \text{rm} + -2.9395291201491998$, 此时loss是: 56.00585694954353
在第468步, 我们获得了函数 $f(\text{rm}) = 4.11502005112436 * \text{rm} + -2.9471595675850173$, 此时loss是: 55.999891403271185
在第469步, 我们获得了函数 $f(\text{rm}) = 4.116219307687946 * \text{rm} + -2.954788180109868$, 此时loss是: 55.99392872575071
在第470步, 我们获得了函数 $f(\text{rm}) = 4.117418275863591 * \text{rm} + -2.9624149581649974$, 此时loss是: 55.987968915602636
在第471步, 我们获得了函数 $f(\text{rm}) = 4.118616955720645 * \text{rm} + -2.970039902191544$, 此时loss是: 55.98201197144787
在第472步, 我们获得了函数 $f(\text{rm}) = 4.119815347328441 * \text{rm} + -2.977663012630541$, 此时loss是: 55.97605789190834
在第473步, 我们获得了函数 $f(\text{rm}) = 4.121013450756293 * \text{rm} + -2.985284289922915$, 此时loss是: 55.970106675606516
在第474步, 我们获得了函数 $f(\text{rm}) = 4.122211266073503 * \text{rm} + -2.9929037345094875$, 此时loss是: 55.964158321165485
在第475步, 我们获得了函数 $f(\text{rm}) = 4.123408793349352 * \text{rm} + -3.0005213468309733$, 此时loss是: 55.95821282720892
在第476步, 我们获得了函数 $f(\text{rm}) = 4.124606032653106 * \text{rm} + -3.008137127327981$, 此时loss是: 55.952270192361325
在第477步, 我们获得了函数 $f(\text{rm}) = 4.125802984054015 * \text{rm} + -3.0157510764410143$, 此时loss是: 55.94633041524789
在第478步, 我们获得了函数 $f(\text{rm}) = 4.126999647621312 * \text{rm} + -3.0233631946104693$, 此时loss是: 55.940393494494195
在第479步, 我们获得了函数 $f(\text{rm}) = 4.128196023424211 * \text{rm} + -3.030973482276638$, 此时loss是: 55.93445942872678
在第480步, 我们获得了函数 $f(\text{rm}) = 4.129392111531913 * \text{rm} + -3.0385819398797054$, 此时loss是: 55.928528216572616
在第481步, 我们获得了函数 $f(\text{rm}) = 4.1305879120136 * \text{rm} + -3.0461885678597502$, 此时loss是: 55.9225998566595

在第482步, 我们获得了函数 $f(rm) = 4.131783424938439 * rm + -3.053793366656747$, 此时loss是: 55.9166743476158
在第483步, 我们获得了函数 $f(rm) = 4.132978650375579 * rm + -3.0613963367105628$, 此时loss是: 55.910751688070654
在第484步, 我们获得了函数 $f(rm) = 4.134173588394152 * rm + -3.06899747846096$, 此时loss是: 55.904831876653724
在第485步, 我们获得了函数 $f(rm) = 4.135368239063275 * rm + -3.076596792347595$, 此时loss是: 55.898914911995305
在第486步, 我们获得了函数 $f(rm) = 4.136562602452047 * rm + -3.0841942788100183$, 此时loss是: 55.8930007927265
在第487步, 我们获得了函数 $f(rm) = 4.137756678629552 * rm + -3.0917899382876746$, 此时loss是: 55.88708951747905
在第488步, 我们获得了函数 $f(rm) = 4.138950467664856 * rm + -3.099383771219903$, 此时loss是: 55.881181084885206
在第489步, 我们获得了函数 $f(rm) = 4.140143969627008 * rm + -3.1069757780459373$, 此时loss是: 55.875275493578016
在第490步, 我们获得了函数 $f(rm) = 4.1413371845850415 * rm + -3.114565959204905$, 此时loss是: 55.86937274219112
在第491步, 我们获得了函数 $f(rm) = 4.142530112607974 * rm + -3.1221543151358286$, 此时loss是: 55.86347282935895
在第492步, 我们获得了函数 $f(rm) = 4.143722753764804 * rm + -3.129740846277625$, 此时loss是: 55.85757575371627
在第493步, 我们获得了函数 $f(rm) = 4.144915108124515 * rm + -3.137325553069105$, 此时loss是: 55.85168151389897
在第494步, 我们获得了函数 $f(rm) = 4.146107175756074 * rm + -3.1449084359489743$, 此时loss是: 55.84579010854314
在第495步, 我们获得了函数 $f(rm) = 4.147298956728433 * rm + -3.1524894953558333$, 此时loss是: 55.83990153628578
在第496步, 我们获得了函数 $f(rm) = 4.148490451110522 * rm + -3.1600687317281766$, 此时loss是: 55.83401579576445
在第497步, 我们获得了函数 $f(rm) = 4.149681658971261 * rm + -3.1676461455043934$, 此时loss是: 55.828132885617585
在第498步, 我们获得了函数 $f(rm) = 4.1508725803795485 * rm + -3.1752217371227673$, 此时loss是: 55.82225280448389
在第499步, 我们获得了函数 $f(rm) = 4.1520632154042705 * rm + -3.182795507021477$, 此时loss是: 55.816375551003055
在第500步, 我们获得了函数 $f(rm) = 4.153253564114292 * rm + -3.1903674556385955$, 此时loss是: 55.81050112381518
在第501步, 我们获得了函数 $f(rm) = 4.154443626578464 * rm + -3.1979375834120902$, 此时loss是: 55.80462952156121
在第502步, 我们获得了函数 $f(rm) = 4.1556334028656225 * rm + -3.205505890779824$, 此时loss是: 55.79876074288273
在第503步, 我们获得了函数 $f(rm) = 4.156822893044585 * rm + -3.213072378179554$, 此时loss是: 55.79289478642186
在第504步, 我们获得了函数 $f(rm) = 4.158012097184151 * rm + -3.2206370460489313$, 此时loss是: 55.78703165082137

在第505步, 我们获得了函数 $f(\text{rm}) = 4.159201015353105 * \text{rm} + -3.228199894825503$, 此时loss是: 55.78117133472491
在第506步, 我们获得了函数 $f(\text{rm}) = 4.1603896476202165 * \text{rm} + -3.2357609249467107$, 此时loss是: 55.77531383677649
在第507步, 我们获得了函数 $f(\text{rm}) = 4.161577994054236 * \text{rm} + -3.2433201368498903$, 此时loss是: 55.769459155620865
在第508步, 我们获得了函数 $f(\text{rm}) = 4.162766054723898 * \text{rm} + -3.250877530972273$, 此时loss是: 55.76360728990365
在第509步, 我们获得了函数 $f(\text{rm}) = 4.163953829697922 * \text{rm} + -3.258433107750984$, 此时loss是: 55.757758238270775
在第510步, 我们获得了函数 $f(\text{rm}) = 4.16514131904501 * \text{rm} + -3.2659868676230444$, 此时loss是: 55.75191199936899
在第511步, 我们获得了函数 $f(\text{rm}) = 4.166328522833846 * \text{rm} + -3.2735388110253703$, 此时loss是: 55.7460685718458
在第512步, 我们获得了函数 $f(\text{rm}) = 4.1675154411331 * \text{rm} + -3.281088938394772$, 此时loss是: 55.74022795434918
在第513步, 我们获得了函数 $f(\text{rm}) = 4.168702074011423 * \text{rm} + -3.2886372501679553$, 此时loss是: 55.734390145527804
在第514步, 我们获得了函数 $f(\text{rm}) = 4.169888421537452 * \text{rm} + -3.2961837467815203$, 此时loss是: 55.7285551440311
在第515步, 我们获得了函数 $f(\text{rm}) = 4.171074483779806 * \text{rm} + -3.3037284286719624$, 此时loss是: 55.72272294850896
在第516步, 我们获得了函数 $f(\text{rm}) = 4.172260260807089 * \text{rm} + -3.3112712962756725$, 此时loss是: 55.71689355761208
在第517步, 我们获得了函数 $f(\text{rm}) = 4.173445752687886 * \text{rm} + -3.3188123500289364$, 此时loss是: 55.711066969991805
在第518步, 我们获得了函数 $f(\text{rm}) = 4.1746309594907665 * \text{rm} + -3.3263515903679344$, 此时loss是: 55.70524318429999
在第519步, 我们获得了函数 $f(\text{rm}) = 4.175815881284284 * \text{rm} + -3.3338890177287426$, 此时loss是: 55.699422199189264
在第520步, 我们获得了函数 $f(\text{rm}) = 4.1770005181369765 * \text{rm} + -3.341424632547332$, 此时loss是: 55.69360401331284
在第521步, 我们获得了函数 $f(\text{rm}) = 4.178184870117363 * \text{rm} + -3.3489584352595685$, 此时loss是: 55.68778862532471
在第522步, 我们获得了函数 $f(\text{rm}) = 4.1793689372939475 * \text{rm} + -3.356490426301214$, 此时loss是: 55.68197603387928
在第523步, 我们获得了函数 $f(\text{rm}) = 4.180552719735219 * \text{rm} + -3.3640206061079243$, 此时loss是: 55.676166237631705
在第524步, 我们获得了函数 $f(\text{rm}) = 4.181736217509646 * \text{rm} + -3.3715489751152514$, 此时loss是: 55.67035923523795
在第525步, 我们获得了函数 $f(\text{rm}) = 4.1829194306856845 * \text{rm} + -3.3790755337586424$, 此时loss是: 55.66455502535433
在第526步, 我们获得了函数 $f(\text{rm}) = 4.1841023593317725 * \text{rm} + -3.38660028247344$, 此时loss是: 55.65875360663808
在第527步, 我们获得了函数 $f(\text{rm}) = 4.185285003516332 * \text{rm} + -3.3941232216948807$, 此时loss是: 55.65295497774698

在第528步, 我们获得了函数 $f(\text{rm}) = 4.1864673633077665 * \text{rm} + -$
3.4016443518580988, 此时loss是: 55.64715913733932
在第529步, 我们获得了函数 $f(\text{rm}) = 4.187649438774466 * \text{rm} + -$
3.4091636733981217, 此时loss是: 55.64136608407431
在第530步, 我们获得了函数 $f(\text{rm}) = 4.188831229984803 * \text{rm} + -$
3.4166811867498734, 此时loss是: 55.635575816611514
在第531步, 我们获得了函数 $f(\text{rm}) = 4.190012737007131 * \text{rm} + -$
3.4241968923481734, 此时loss是: 55.62978833361138
在第532步, 我们获得了函数 $f(\text{rm}) = 4.191193959909793 * \text{rm} + -$
3.4317107906277355, 此时loss是: 55.62400363373478
在第533步, 我们获得了函数 $f(\text{rm}) = 4.192374898761108 * \text{rm} + -$
3.4392228820231705, 此时loss是: 55.61822171564342
在第534步, 我们获得了函数 $f(\text{rm}) = 4.193555553629385 * \text{rm} + -$
3.4467331669689836, 此时loss是: 55.61244257799963
在第535步, 我们获得了函数 $f(\text{rm}) = 4.194735924582914 * \text{rm} + -$
3.4542416458995757, 此时loss是: 55.60666621946625
在第536步, 我们获得了函数 $f(\text{rm}) = 4.195916011689967 * \text{rm} + -$
3.4617483192492435, 此时loss是: 55.600892638706895
在第537步, 我们获得了函数 $f(\text{rm}) = 4.197095815018802 * \text{rm} + -$
3.4692531874521793, 此时loss是: 55.59512183438562
在第538步, 我们获得了函数 $f(\text{rm}) = 4.1982753346376604 * \text{rm} + -$
3.4767562509424703, 此时loss是: 55.58935380516752
在第539步, 我们获得了函数 $f(\text{rm}) = 4.199454570614766 * \text{rm} + -$
3.4842575101541002, 此时loss是: 55.583588549717945
在第540步, 我们获得了函数 $f(\text{rm}) = 4.200633523018328 * \text{rm} + -$
3.491756965520948, 此时loss是: 55.57782606670294
在第541步, 我们获得了函数 $f(\text{rm}) = 4.201812191916535 * \text{rm} + -$
3.4992546174767885, 此时loss是: 55.572066354789506
在第542步, 我们获得了函数 $f(\text{rm}) = 4.202990577377566 * \text{rm} + -$
3.5067504664552915, 此时loss是: 55.56630941264487
在第543步, 我们获得了函数 $f(\text{rm}) = 4.204168679469577 * \text{rm} + -$
3.5142445128900235, 此时loss是: 55.56055523893711
在第544步, 我们获得了函数 $f(\text{rm}) = 4.2053464982607105 * \text{rm} + -$
3.521736757214446, 此时loss是: 55.55480383233501
在第545步, 我们获得了函数 $f(\text{rm}) = 4.206524033819094 * \text{rm} + -$
3.5292271998619165, 此时loss是: 55.549055191507826
在第546步, 我们获得了函数 $f(\text{rm}) = 4.207701286212837 * \text{rm} + -$
3.5367158412656887, 此时loss是: 55.54330931512553
在第547步, 我们获得了函数 $f(\text{rm}) = 4.208878255510031 * \text{rm} + -$
3.5442026818589114, 此时loss是: 55.537566201858844
在第548步, 我们获得了函数 $f(\text{rm}) = 4.210054941778754 * \text{rm} + -$
3.5516877220746297, 此时loss是: 55.53182585037891
在第549步, 我们获得了函数 $f(\text{rm}) = 4.211231345087066 * \text{rm} + -$
3.559170962345785, 此时loss是: 55.52608825935764
在第550步, 我们获得了函数 $f(\text{rm}) = 4.212407465503013 * \text{rm} + -$
3.5666524031052127, 此时loss是: 55.52035342746764

在第551步, 我们获得了函数 $f(rm) = 4.21358330309462 * rm + -$
3.574132044785647, 此时loss是: 55.51462135338191
在第552步, 我们获得了函数 $f(rm) = 4.214758857929901 * rm + -$
3.5816098878197153, 此时loss是: 55.50889203577447
在第553步, 我们获得了函数 $f(rm) = 4.21593413007685 * rm + -$
3.589085932639943, 此时loss是: 55.50316547331967
在第554步, 我们获得了函数 $f(rm) = 4.217109119603444 * rm + -$
3.5965601796787503, 此时loss是: 55.49744166469256
在第555步, 我们获得了函数 $f(rm) = 4.218283826577648 * rm + -$
3.6040326293684535, 此时loss是: 55.4917206085689
在第556步, 我们获得了函数 $f(rm) = 4.219458251067406 * rm + -$
3.611503282141266, 此时loss是: 55.48600230362501
在第557步, 我们获得了函数 $f(rm) = 4.22063239314065 * rm + -$
3.6189721384292954, 此时loss是: 55.48028674853795
在第558步, 我们获得了函数 $f(rm) = 4.22180625286529 * rm + -$
3.6264391986645474, 此时loss是: 55.474573941985355
在第559步, 我们获得了函数 $f(rm) = 4.222979830309226 * rm + -$
3.633904463278922, 此时loss是: 55.46886388264538
在第560步, 我们获得了函数 $f(rm) = 4.2241531255403375 * rm + -$
3.641367932704217, 此时loss是: 55.463156569197
在第561步, 我们获得了函数 $f(rm) = 4.22532613862649 * rm + -$
3.6488296073721256, 此时loss是: 55.45745200031974
在第562步, 我们获得了函数 $f(rm) = 4.226498869635529 * rm + -$
3.6562894877142367, 此时loss是: 55.451750174693935
在第563步, 我们获得了函数 $f(rm) = 4.227671318635288 * rm + -$
3.663747574162036, 此时loss是: 55.44605109100006
在第564步, 我们获得了函数 $f(rm) = 4.228843485693583 * rm + -$
3.671203867146905, 此时loss是: 55.44035474791984
在第565步, 我们获得了函数 $f(rm) = 4.230015370878211 * rm + -$
3.6786583671001227, 此时loss是: 55.43466114413517
在第566步, 我们获得了函数 $f(rm) = 4.231186974256957 * rm + -$
3.6861110744528625, 此时loss是: 55.42897027832883
在第567步, 我们获得了函数 $f(rm) = 4.232358295897586 * rm + -$
3.6935619896361955, 此时loss是: 55.42328214918421
在第568步, 我们获得了函数 $f(rm) = 4.233529335867848 * rm + -$
3.701011113081089, 此时loss是: 55.41759675538524
在第569步, 我们获得了函数 $f(rm) = 4.234700094235479 * rm + -$
3.7084584452184064, 此时loss是: 55.41191409561663
在第570步, 我们获得了函数 $f(rm) = 4.235870571068195 * rm + -$
3.715903986478907, 此时loss是: 55.4062341685634
在第571步, 我们获得了函数 $f(rm) = 4.237040766433696 * rm + -$
3.7233477372932473, 此时loss是: 55.40055697291157
在第572步, 我们获得了函数 $f(rm) = 4.23821068039967 * rm + -$
3.73078969809198, 此时loss是: 55.39488250734763
在第573步, 我们获得了函数 $f(rm) = 4.239380313033785 * rm + -$
3.738229869305554, 此时loss是: 55.3892107705587

在第574步, 我们获得了函数 $f(\text{rm}) = 4.240549664403693 * \text{rm} + -3.745668251364315$, 此时loss是: 55.383541761232685
在第575步, 我们获得了函数 $f(\text{rm}) = 4.241718734577029 * \text{rm} + -3.753104844698505$, 此时loss是: 55.37787547805766
在第576步, 我们获得了函数 $f(\text{rm}) = 4.242887523621414 * \text{rm} + -3.760539649738263$, 此时loss是: 55.372211919722936
在第577步, 我们获得了函数 $f(\text{rm}) = 4.244056031604453 * \text{rm} + -3.7679726669136238$, 此时loss是: 55.36655108491813
在第578步, 我们获得了函数 $f(\text{rm}) = 4.245224258593732 * \text{rm} + -3.7754038966545194$, 此时loss是: 55.36089297233349
在第579步, 我们获得了函数 $f(\text{rm}) = 4.246392204656822 * \text{rm} + -3.782833339390778$, 此时loss是: 55.35523758065987
在第580步, 我们获得了函数 $f(\text{rm}) = 4.247559869861279 * \text{rm} + -3.7902609955521247$, 此时loss是: 55.34958490858894
在第581步, 我们获得了函数 $f(\text{rm}) = 4.248727254274641 * \text{rm} + -3.7976868655681817$, 此时loss是: 55.343934954812866
在第582步, 我们获得了函数 $f(\text{rm}) = 4.249894357964431 * \text{rm} + -3.805110949868467$, 此时loss是: 55.33828771802433
在第583步, 我们获得了函数 $f(\text{rm}) = 4.251061180998154 * \text{rm} + -3.8125332488823958$, 此时loss是: 55.332643196916905
在第584步, 我们获得了函数 $f(\text{rm}) = 4.252227723443302 * \text{rm} + -3.8199537630392797$, 此时loss是: 55.32700139018458
在第585步, 我们获得了函数 $f(\text{rm}) = 4.253393985367347 * \text{rm} + -3.8273724927683275$, 此时loss是: 55.321362296522146
在第586步, 我们获得了函数 $f(\text{rm}) = 4.254559966837747 * \text{rm} + -3.8347894384986447$, 此时loss是: 55.31572591462485
在第587步, 我们获得了函数 $f(\text{rm}) = 4.255725667921944 * \text{rm} + -3.842204600659233$, 此时loss是: 55.310092243188556
在第588步, 我们获得了函数 $f(\text{rm}) = 4.256891088687363 * \text{rm} + -3.8496179796789924$, 此时loss是: 55.304461280910004
在第589步, 我们获得了函数 $f(\text{rm}) = 4.258056229201412 * \text{rm} + -3.857029575986718$, 此时loss是: 55.29883302648634
在第590步, 我们获得了函数 $f(\text{rm}) = 4.259221089531485 * \text{rm} + -3.8644393900111034$, 此时loss是: 55.29320747861531
在第591步, 我们获得了函数 $f(\text{rm}) = 4.260385669744958 * \text{rm} + -3.8718474221807373$, 此时loss是: 55.28758463599546
在第592步, 我们获得了函数 $f(\text{rm}) = 4.26154996990919 * \text{rm} + -3.879253672924107$, 此时loss是: 55.281964497325845
在第593步, 我们获得了函数 $f(\text{rm}) = 4.262713990091526 * \text{rm} + -3.886658142669596$, 此时loss是: 55.27634706130618
在第594步, 我们获得了函数 $f(\text{rm}) = 4.263877730359293 * \text{rm} + -3.8940608318454846$, 此时loss是: 55.27073232663682
在第595步, 我们获得了函数 $f(\text{rm}) = 4.2650411907798045 * \text{rm} + -3.9014617408799506$, 此时loss是: 55.26512029201856
在第596步, 我们获得了函数 $f(\text{rm}) = 4.266204371420356 * \text{rm} + -3.9088608702010688$, 此时loss是: 55.2595109561532

在第597步, 我们获得了函数 $f(\text{rm}) = 4.267367272348225 * \text{rm} + -3.916258220236811$, 此时loss是: 55.25390431774285
在第598步, 我们获得了函数 $f(\text{rm}) = 4.268529893630675 * \text{rm} + -3.9236537914150458$, 此时loss是: 55.24830037549033
在第599步, 我们获得了函数 $f(\text{rm}) = 4.269692235334955 * \text{rm} + -3.931047584163539$, 此时loss是: 55.24269912809912
在第600步, 我们获得了函数 $f(\text{rm}) = 4.2708542975282935 * \text{rm} + -3.9384395989099534$, 此时loss是: 55.23710057427329
在第601步, 我们获得了函数 $f(\text{rm}) = 4.272016080277906 * \text{rm} + -3.94582983608185$, 此时loss是: 55.23150471271754
在第602步, 我们获得了函数 $f(\text{rm}) = 4.273177583650991 * \text{rm} + -3.953218296106685$, 此时loss是: 55.22591154213719
在第603步, 我们获得了函数 $f(\text{rm}) = 4.274338807714731 * \text{rm} + -3.960604979411814$, 此时loss是: 55.22032106123813
在第604步, 我们获得了函数 $f(\text{rm}) = 4.275499752536292 * \text{rm} + -3.9679898864244882$, 此时loss是: 55.21473326872702
在第605步, 我们获得了函数 $f(\text{rm}) = 4.276660418182823 * \text{rm} + -3.975373017571857$, 此时loss是: 55.209148163311035
在第606步, 我们获得了函数 $f(\text{rm}) = 4.27782080472146 * \text{rm} + -3.9827543732809665$, 此时loss是: 55.203565743697915
在第607步, 我们获得了函数 $f(\text{rm}) = 4.2789809122193185 * \text{rm} + -3.9901339539787606$, 此时loss是: 55.19798600859619
在第608步, 我们获得了函数 $f(\text{rm}) = 4.2801407407435015 * \text{rm} + -3.99751176009208$, 此时loss是: 55.192408956714836
在第609步, 我们获得了函数 $f(\text{rm}) = 4.281300290361094 * \text{rm} + -4.004887792047664$, 此时loss是: 55.18683458676359
在第610步, 我们获得了函数 $f(\text{rm}) = 4.282459561139165 * \text{rm} + -4.012262050272147$, 此时loss是: 55.18126289745266
在第611步, 我们获得了函数 $f(\text{rm}) = 4.283618553144768 * \text{rm} + -4.019634535192063$, 此时loss是: 55.175693887493075
在第612步, 我们获得了函数 $f(\text{rm}) = 4.284777266444939 * \text{rm} + -4.027005247233842$, 此时loss是: 55.17012755559627
在第613步, 我们获得了函数 $f(\text{rm}) = 4.285935701106701 * \text{rm} + -4.034374186823812$, 此时loss是: 55.1645639004745
在第614步, 我们获得了函数 $f(\text{rm}) = 4.287093857197058 * \text{rm} + -4.041741354388199$, 此时loss是: 55.159002920840486
在第615步, 我们获得了函数 $f(\text{rm}) = 4.288251734782998 * \text{rm} + -4.0491067503531255$, 此时loss是: 55.153444615407494
在第616步, 我们获得了函数 $f(\text{rm}) = 4.289409333931494 * \text{rm} + -4.056470375144612$, 此时loss是: 55.14788898288976
在第617步, 我们获得了函数 $f(\text{rm}) = 4.290566654709502 * \text{rm} + -4.063832229188578$, 此时loss是: 55.14233602200176
在第618步, 我们获得了函数 $f(\text{rm}) = 4.291723697183963 * \text{rm} + -4.071192312910838$, 此时loss是: 55.136785731458886
在第619步, 我们获得了函数 $f(\text{rm}) = 4.2928804614218015 * \text{rm} + -4.078550626737105$, 此时loss是: 55.131238109976906

在第620步, 我们获得了函数 $f(\text{rm}) = 4.294036947489925 * \text{rm} + -4.085907171092989$, 此时loss是: 55.125693156272256
在第621步, 我们获得了函数 $f(\text{rm}) = 4.295193155455226 * \text{rm} + -4.093261946404001$, 此时loss是: 55.12015086906207
在第622步, 我们获得了函数 $f(\text{rm}) = 4.2963490853845805 * \text{rm} + -4.100614953095546$, 此时loss是: 55.11461124706412
在第623步, 我们获得了函数 $f(\text{rm}) = 4.297504737344848 * \text{rm} + -4.1079661915929275$, 此时loss是: 55.109074288996744
在第624步, 我们获得了函数 $f(\text{rm}) = 4.2986601114028735 * \text{rm} + -4.115315662321348$, 此时loss是: 55.10353999357885
在第625步, 我们获得了函数 $f(\text{rm}) = 4.299815207625484 * \text{rm} + -4.122663365705906$, 此时loss是: 55.09800835953003
在第626步, 我们获得了函数 $f(\text{rm}) = 4.30097002607949 * \text{rm} + -4.130009302171599$, 此时loss是: 55.092479385570456
在第627步, 我们获得了函数 $f(\text{rm}) = 4.302124566831689 * \text{rm} + -4.137353472143323$, 此时loss是: 55.0869530704209
在第628步, 我们获得了函数 $f(\text{rm}) = 4.3032788299488605 * \text{rm} + -4.144695876045869$, 此时loss是: 55.0814294128028
在第629步, 我们获得了函数 $f(\text{rm}) = 4.304432815497767 * \text{rm} + -4.152036514303929$, 此时loss是: 55.07590841143816
在第630步, 我们获得了函数 $f(\text{rm}) = 4.305586523545157 * \text{rm} + -4.159375387342091$, 此时loss是: 55.07039006504976
在第631步, 我们获得了函数 $f(\text{rm}) = 4.306739954157761 * \text{rm} + -4.166712495584841$, 此时loss是: 55.06487437236063
在第632步, 我们获得了函数 $f(\text{rm}) = 4.307893107402294 * \text{rm} + -4.1740478394565645$, 此时loss是: 55.05936133209481
在第633步, 我们获得了函数 $f(\text{rm}) = 4.309045983345457 * \text{rm} + -4.181381419381543$, 此时loss是: 55.053850942976695
在第634步, 我们获得了函数 $f(\text{rm}) = 4.310198582053932 * \text{rm} + -4.188713235783957$, 此时loss是: 55.04834320373149
在第635步, 我们获得了函数 $f(\text{rm}) = 4.311350903594387 * \text{rm} + -4.196043289087885$, 此时loss是: 55.04283811308478
在第636步, 我们获得了函数 $f(\text{rm}) = 4.312502948033472 * \text{rm} + -4.203371579717302$, 此时loss是: 55.037335669762946
在第637步, 我们获得了函数 $f(\text{rm}) = 4.313654715437823 * \text{rm} + -4.210698108096085$, 此时loss是: 55.03183587249297
在第638步, 我们获得了函数 $f(\text{rm}) = 4.31480620587406 * \text{rm} + -4.218022874648003$, 此时loss是: 55.02633872000226
在第639步, 我们获得了函数 $f(\text{rm}) = 4.315957419408785 * \text{rm} + -4.22534587979673$, 此时loss是: 55.02084421101911
在第640步, 我们获得了函数 $f(\text{rm}) = 4.3171083561085855 * \text{rm} + -4.232667123965832$, 此时loss是: 55.015352344272266
在第641步, 我们获得了函数 $f(\text{rm}) = 4.318259016040033 * \text{rm} + -4.239986607578776$, 此时loss是: 55.009863118491154
在第642步, 我们获得了函数 $f(\text{rm}) = 4.319409399269681 * \text{rm} + -4.247304331058928$, 此时loss是: 55.00437653240563

在第643步, 我们获得了函数 $f(\text{rm}) = 4.320559505864071 * \text{rm} + -4.25462029482955$, 此时loss是: 54.99889258474646
在第644步, 我们获得了函数 $f(\text{rm}) = 4.321709335889724 * \text{rm} + -4.261934499313805$, 此时loss是: 54.993411274244714
在第645步, 我们获得了函数 $f(\text{rm}) = 4.3228588894131486 * \text{rm} + -4.269246944934751$, 此时loss是: 54.987932599632366
在第646步, 我们获得了函数 $f(\text{rm}) = 4.324008166500835 * \text{rm} + -4.276557632115346$, 此时loss是: 54.9824565596417
在第647步, 我们获得了函数 $f(\text{rm}) = 4.325157167219259 * \text{rm} + -4.2838665612784474$, 此时loss是: 54.97698315300581
在第648步, 我们获得了函数 $f(\text{rm}) = 4.326305891634879 * \text{rm} + -4.291173732846809$, 此时loss是: 54.97151237845846
在第649步, 我们获得了函数 $f(\text{rm}) = 4.327454339814139 * \text{rm} + -4.298479147243082$, 此时loss是: 54.9660442347338
在第650步, 我们获得了函数 $f(\text{rm}) = 4.3286025118234654 * \text{rm} + -4.3057828048898195$, 此时loss是: 54.96057872056673
在第651步, 我们获得了函数 $f(\text{rm}) = 4.329750407729271 * \text{rm} + -4.31308470620947$, 此时loss是: 54.955115834692705
在第652步, 我们获得了函数 $f(\text{rm}) = 4.330898027597948 * \text{rm} + -4.320384851624382$, 此时loss是: 54.9496555758479
在第653步, 我们获得了函数 $f(\text{rm}) = 4.332045371495878 * \text{rm} + -4.3276832415568025$, 此时loss是: 54.94419794276895
在第654步, 我们获得了函数 $f(\text{rm}) = 4.333192439489424 * \text{rm} + -4.334979876428875$, 此时loss是: 54.93874293419311
在第655步, 我们获得了函数 $f(\text{rm}) = 4.334339231644933 * \text{rm} + -4.342274756662643$, 此时loss是: 54.93329054885836
在第656步, 我们获得了函数 $f(\text{rm}) = 4.335485748028737 * \text{rm} + -4.3495678826800495$, 此时loss是: 54.9278407855032
在第657步, 我们获得了函数 $f(\text{rm}) = 4.3366319887071505 * \text{rm} + -4.356859254902934$, 此时loss是: 54.922393642866794
在第658步, 我们获得了函数 $f(\text{rm}) = 4.337777953746474 * \text{rm} + -4.364148873753036$, 此时loss是: 54.916949119688816
在第659步, 我们获得了函数 $f(\text{rm}) = 4.33892364321299 * \text{rm} + -4.371436739651992$, 此时loss是: 54.911507214709616
在第660步, 我们获得了函数 $f(\text{rm}) = 4.3400690571729665 * \text{rm} + -4.37872285302134$, 此时loss是: 54.906067926670154
在第661步, 我们获得了函数 $f(\text{rm}) = 4.341214195692656 * \text{rm} + -4.386007214282512$, 此时loss是: 54.90063125431197
在第662步, 我们获得了函数 $f(\text{rm}) = 4.3423590588382925 * \text{rm} + -4.393289823856843$, 此时loss是: 54.89519719637714
在第663步, 我们获得了函数 $f(\text{rm}) = 4.343503646676098 * \text{rm} + -4.400570682165566$, 此时loss是: 54.88976575160861
在第664步, 我们获得了函数 $f(\text{rm}) = 4.344647959272275 * \text{rm} + -4.40784978962981$, 此时loss是: 54.88433691874954
在第665步, 我们获得了函数 $f(\text{rm}) = 4.345791996693012 * \text{rm} + -4.4151271466706055$, 此时loss是: 54.87891069654408

在第666步, 我们获得了函数 $f(\text{rm}) = 4.34693575900448 * \text{rm} + -4.422402753708881$, 此时loss是: 54.87348708373668
在第667步, 我们获得了函数 $f(\text{rm}) = 4.348079246272836 * \text{rm} + -4.429676611165463$, 此时loss是: 54.86806607907251
在第668步, 我们获得了函数 $f(\text{rm}) = 4.349222458564221 * \text{rm} + -4.4369487194610775$, 此时loss是: 54.862647681297425
在第669步, 我们获得了函数 $f(\text{rm}) = 4.350365395944759 * \text{rm} + -4.444219079016349$, 此时loss是: 54.857231889157724
在第670步, 我们获得了函数 $f(\text{rm}) = 4.351508058480557 * \text{rm} + -4.451487690251803$, 此时loss是: 54.85181870140048
在第671步, 我们获得了函数 $f(\text{rm}) = 4.352650446237709 * \text{rm} + -4.458754553587859$, 此时loss是: 54.846408116773226
在第672步, 我们获得了函数 $f(\text{rm}) = 4.353792559282291 * \text{rm} + -4.46601966944484$, 此时loss是: 54.84100013402416
在第673步, 我们获得了函数 $f(\text{rm}) = 4.354934397680363 * \text{rm} + -4.473283038242966$, 此时loss是: 54.835594751902036
在第674步, 我们获得了函数 $f(\text{rm}) = 4.3560759614979725 * \text{rm} + -4.480544660402355$, 此时loss是: 54.830191969156296
在第675步, 我们获得了函数 $f(\text{rm}) = 4.3572172508011455 * \text{rm} + -4.487804536343027$, 此时loss是: 54.824791784536934
在第676步, 我们获得了函数 $f(\text{rm}) = 4.358358265655897 * \text{rm} + -4.495062666484897$, 此时loss是: 54.81939419679449
在第677步, 我们获得了函数 $f(\text{rm}) = 4.359499006128224 * \text{rm} + -4.502319051247785$, 此时loss是: 54.813999204680314
在第678步, 我们获得了函数 $f(\text{rm}) = 4.360639472284107 * \text{rm} + -4.509573691051402$, 此时loss是: 54.80860680694605
在第679步, 我们获得了函数 $f(\text{rm}) = 4.361779664189512 * \text{rm} + -4.516826586315365$, 此时loss是: 54.80321700234413
在第680步, 我们获得了函数 $f(\text{rm}) = 4.362919581910388 * \text{rm} + -4.524077737459186$, 此时loss是: 54.79782978962752
在第681步, 我们获得了函数 $f(\text{rm}) = 4.36405922551267 * \text{rm} + -4.531327144902279$, 此时loss是: 54.79244516754986
在第682步, 我们获得了函数 $f(\text{rm}) = 4.365198595062275 * \text{rm} + -4.538574809063953$, 此时loss是: 54.78706313486538
在第683步, 我们获得了函数 $f(\text{rm}) = 4.366337690625106 * \text{rm} + -4.545820730363422$, 此时loss是: 54.781683690328855
在第684步, 我们获得了函数 $f(\text{rm}) = 4.367476512267048 * \text{rm} + -4.553064909219795$, 此时loss是: 54.77630683269558
在第685步, 我们获得了函数 $f(\text{rm}) = 4.368615060053973 * \text{rm} + -4.560307346052079$, 此时loss是: 54.770932560721704
在第686步, 我们获得了函数 $f(\text{rm}) = 4.369753334051733 * \text{rm} + -4.567548041279185$, 此时loss是: 54.76556087316373
在第687步, 我们获得了函数 $f(\text{rm}) = 4.37089133432617 * \text{rm} + -4.57478699531992$, 此时loss是: 54.76019176877883
在第688步, 我们获得了函数 $f(\text{rm}) = 4.372029060943103 * \text{rm} + -4.582024208592991$, 此时loss是: 54.75482524632485

在第689步, 我们获得了函数 $f(\text{rm}) = 4.373166513968343 * \text{rm} + -4.589259681517005$, 此时loss是: 54.74946130456007
在第690步, 我们获得了函数 $f(\text{rm}) = 4.374303693467678 * \text{rm} + -4.596493414510467$, 此时loss是: 54.7440999422437
在第691步, 我们获得了函数 $f(\text{rm}) = 4.375440599506885 * \text{rm} + -4.603725407991781$, 此时loss是: 54.73874115813502
在第692步, 我们获得了函数 $f(\text{rm}) = 4.376577232151724 * \text{rm} + -4.6109556623792525$, 此时loss是: 54.73338495099442
在第693步, 我们获得了函数 $f(\text{rm}) = 4.377713591467938 * \text{rm} + -4.618184178091086$, 此时loss是: 54.728031319582655
在第694步, 我们获得了函数 $f(\text{rm}) = 4.378849677521254 * \text{rm} + -4.625410955545385$, 此时loss是: 54.7226802626609
在第695步, 我们获得了函数 $f(\text{rm}) = 4.379985490377385 * \text{rm} + -4.6326359951601495$, 此时loss是: 54.71733177899127
在第696步, 我们获得了函数 $f(\text{rm}) = 4.381121030102029 * \text{rm} + -4.639859297353285$, 此时loss是: 54.711985867336445
在第697步, 我们获得了函数 $f(\text{rm}) = 4.382256296760864 * \text{rm} + -4.647080862542591$, 此时loss是: 54.70664252645926
在第698步, 我们获得了函数 $f(\text{rm}) = 4.383391290419556 * \text{rm} + -4.654300691145769$, 此时loss是: 54.701301755123744
在第699步, 我们获得了函数 $f(\text{rm}) = 4.3845260111437545 * \text{rm} + -4.661518783580421$, 此时loss是: 54.69596355209409
在第700步, 我们获得了函数 $f(\text{rm}) = 4.385660458999091 * \text{rm} + -4.6687351402640465$, 此时loss是: 54.69062791613525
在第701步, 我们获得了函数 $f(\text{rm}) = 4.386794634051184 * \text{rm} + -4.675949761614046$, 此时loss是: 54.685294846012845
在第702步, 我们获得了函数 $f(\text{rm}) = 4.387928536365635 * \text{rm} + -4.683162648047718$, 此时loss是: 54.67996434049291
在第703步, 我们获得了函数 $f(\text{rm}) = 4.3890621660080305 * \text{rm} + -4.690373799982263$, 此时loss是: 54.674636398342166
在第704步, 我们获得了函数 $f(\text{rm}) = 4.390195523043939 * \text{rm} + -4.697583217834779$, 此时loss是: 54.669311018327974
在第705步, 我们获得了函数 $f(\text{rm}) = 4.391328607538916 * \text{rm} + -4.704790902022266$, 此时loss是: 54.66398819921817
在第706步, 我们获得了函数 $f(\text{rm}) = 4.392461419558499 * \text{rm} + -4.7119968529616205$, 此时loss是: 54.658667939781296
在第707步, 我们获得了函数 $f(\text{rm}) = 4.3935939591682125 * \text{rm} + -4.719201071069642$, 此时loss是: 54.65335023878643
在第708步, 我们获得了函数 $f(\text{rm}) = 4.394726226433562 * \text{rm} + -4.726403556763027$, 此时loss是: 54.64803509500327
在第709步, 我们获得了函数 $f(\text{rm}) = 4.395858221420039 * \text{rm} + -4.733604310458375$, 此时loss是: 54.642722507202
在第710步, 我们获得了函数 $f(\text{rm}) = 4.39698994419312 * \text{rm} + -4.740803332572183$, 此时loss是: 54.637412474153614
在第711步, 我们获得了函数 $f(\text{rm}) = 4.398121394818263 * \text{rm} + -4.748000623520847$, 此时loss是: 54.63210499462946

在第712步, 我们获得了函数 $f(\text{rm}) = 4.399252573360914 * \text{rm} + -4.755196183720665$, 此时loss是: 54.626800067401646
在第713步, 我们获得了函数 $f(\text{rm}) = 4.4003834798865 * \text{rm} + -4.762390013587835$, 此时loss是: 54.621497691242745
在第714步, 我们获得了函数 $f(\text{rm}) = 4.401514114460434 * \text{rm} + -4.769582113538452$, 此时loss是: 54.61619786492605
在第715步, 我们获得了函数 $f(\text{rm}) = 4.402644477148112 * \text{rm} + -4.776772483988514$, 此时loss是: 54.61090058722535
在第716步, 我们获得了函数 $f(\text{rm}) = 4.403774568014916 * \text{rm} + -4.783961125353919$, 此时loss是: 54.605605856915
在第717步, 我们获得了函数 $f(\text{rm}) = 4.404904387126212 * \text{rm} + -4.791148038050462$, 此时loss是: 54.60031367277011
在第718步, 我们获得了函数 $f(\text{rm}) = 4.406033934547349 * \text{rm} + -4.7983332224938415$, 此时loss是: 54.59502403356612
在第719步, 我们获得了函数 $f(\text{rm}) = 4.40716321034366 * \text{rm} + -4.805516679099654$, 此时loss是: 54.58973693807928
在第720步, 我们获得了函数 $f(\text{rm}) = 4.408292214580465 * \text{rm} + -4.812698408283396$, 此时loss是: 54.58445238508635
在第721步, 我们获得了函数 $f(\text{rm}) = 4.409420947323065 * \text{rm} + -4.819878410460465$, 此时loss是: 54.579170373364704
在第722步, 我们获得了函数 $f(\text{rm}) = 4.410549408636748 * \text{rm} + -4.827056686046158$, 此时loss是: 54.57389090169229
在第723步, 我们获得了函数 $f(\text{rm}) = 4.411677598586784 * \text{rm} + -4.834233235455673$, 此时loss是: 54.568613968847494
在第724步, 我们获得了函数 $f(\text{rm}) = 4.4128055172384295 * \text{rm} + -4.841408059104106$, 此时loss是: 54.563339573609575
在第725步, 我们获得了函数 $f(\text{rm}) = 4.413933164656923 * \text{rm} + -4.848581157406456$, 此时loss是: 54.55806771475812
在第726步, 我们获得了函数 $f(\text{rm}) = 4.41506054090749 * \text{rm} + -4.855752530777621$, 此时loss是: 54.5527983910735
在第727步, 我们获得了函数 $f(\text{rm}) = 4.416187646055337 * \text{rm} + -4.862922179632398$, 此时loss是: 54.54753160133657
在第728步, 我们获得了函数 $f(\text{rm}) = 4.4173144801656585 * \text{rm} + -4.870090104385486$, 此时loss是: 54.54226734432876
在第729步, 我们获得了函数 $f(\text{rm}) = 4.418441043303631 * \text{rm} + -4.877256305451484$, 此时loss是: 54.537005618832126
在第730步, 我们获得了函数 $f(\text{rm}) = 4.419567335534416 * \text{rm} + -4.88442078324489$, 此时loss是: 54.53174642362933
在第731步, 我们获得了函数 $f(\text{rm}) = 4.420693356923158 * \text{rm} + -4.891583538180104$, 此时loss是: 54.52648975750353
在第732步, 我们获得了函数 $f(\text{rm}) = 4.421819107534988 * \text{rm} + -4.898744570671425$, 此时loss是: 54.5212356192386
在第733步, 我们获得了函数 $f(\text{rm}) = 4.422944587435021 * \text{rm} + -4.905903881133054$, 此时loss是: 54.51598400761891
在第734步, 我们获得了函数 $f(\text{rm}) = 4.424069796688354 * \text{rm} + -4.91306146997909$, 此时loss是: 54.51073492142934

在第735步, 我们获得了函数 $f(\text{rm}) = 4.425194735360071 * \text{rm} + -4.920217337623535$, 此时loss是: 54.50548835945546
在第736步, 我们获得了函数 $f(\text{rm}) = 4.426319403515239 * \text{rm} + -4.9273714844802905$, 此时loss是: 54.50024432048357
在第737步, 我们获得了函数 $f(\text{rm}) = 4.427443801218911 * \text{rm} + -4.934523910963157$, 此时loss是: 54.4950028033002
在第738步, 我们获得了函数 $f(\text{rm}) = 4.428567928536121 * \text{rm} + -4.9416746174858375$, 此时loss是: 54.48976380669275
在第739步, 我们获得了函数 $f(\text{rm}) = 4.429691785531891 * \text{rm} + -4.948823604461934$, 此时loss是: 54.48452732944908
在第740步, 我们获得了函数 $f(\text{rm}) = 4.430815372271225 * \text{rm} + -4.95597087230495$, 此时loss是: 54.479293370357695
在第741步, 我们获得了函数 $f(\text{rm}) = 4.4319386888191135 * \text{rm} + -4.96311642142829$, 此时loss是: 54.474061928207654
在第742步, 我们获得了函数 $f(\text{rm}) = 4.4330617352405275 * \text{rm} + -4.970260252245258$, 此时loss是: 54.46883300178848
在第743步, 我们获得了函数 $f(\text{rm}) = 4.434184511600427 * \text{rm} + -4.977402365169059$, 此时loss是: 54.463606589890595
在第744步, 我们获得了函数 $f(\text{rm}) = 4.435307017963755 * \text{rm} + -4.984542760612799$, 此时loss是: 54.45838269130458
在第745步, 我们获得了函数 $f(\text{rm}) = 4.436429254395436 * \text{rm} + -4.991681438989483$, 此时loss是: 54.45316130482199
在第746步, 我们获得了函数 $f(\text{rm}) = 4.437551220960382 * \text{rm} + -4.998818400712019$, 此时loss是: 54.44794242923466
在第747步, 我们获得了函数 $f(\text{rm}) = 4.438672917723488 * \text{rm} + -5.005953646193215$, 此时loss是: 54.44272606333518
在第748步, 我们获得了函数 $f(\text{rm}) = 4.439794344749635 * \text{rm} + -5.013087175845779$, 此时loss是: 54.43751220591671
在第749步, 我们获得了函数 $f(\text{rm}) = 4.440915502103686 * \text{rm} + -5.020218990082321$, 此时loss是: 54.432300855772986
在第750步, 我们获得了函数 $f(\text{rm}) = 4.442036389850491 * \text{rm} + -5.027349089315349$, 此时loss是: 54.42709201169818
在第751步, 我们获得了函数 $f(\text{rm}) = 4.4431570080548815 * \text{rm} + -5.034477473957275$, 此时loss是: 54.42188567248719
在第752步, 我们获得了函数 $f(\text{rm}) = 4.444277356781677 * \text{rm} + -5.041604144420411$, 此时loss是: 54.4166818369355
在第753步, 我们获得了函数 $f(\text{rm}) = 4.445397436095678 * \text{rm} + -5.04872910111697$, 此时loss是: 54.41148050383908
在第754步, 我们获得了函数 $f(\text{rm}) = 4.44651724606167 * \text{rm} + -5.055852344459064$, 此时loss是: 54.406281671994655
在第755步, 我们获得了函数 $f(\text{rm}) = 4.447636786744425 * \text{rm} + -5.062973874858707$, 此时loss是: 54.40108534019933
在第756步, 我们获得了函数 $f(\text{rm}) = 4.448756058208698 * \text{rm} + -5.070093692727815$, 此时loss是: 54.39589150725077
在第757步, 我们获得了函数 $f(\text{rm}) = 4.449875060519228 * \text{rm} + -5.077211798478204$, 此时loss是: 54.3907001719475

在第758步, 我们获得了函数 $f(\text{rm}) = 4.45099379374074 * \text{rm} + -5.084328192521591$, 此时loss是: 54.38551133308819
在第759步, 我们获得了函数 $f(\text{rm}) = 4.452112257937942 * \text{rm} + -5.0914428752695935$, 此时loss是: 54.38032498947255
在第760步, 我们获得了函数 $f(\text{rm}) = 4.453230453175527 * \text{rm} + -5.098555847133731$, 此时loss是: 54.375141139900634
在第761步, 我们获得了函数 $f(\text{rm}) = 4.454348379518172 * \text{rm} + -5.105667108525425$, 此时loss是: 54.36995978317294
在第762步, 我们获得了函数 $f(\text{rm}) = 4.455466037030538 * \text{rm} + -5.112776659855994$, 此时loss是: 54.36478091809085
在第763步, 我们获得了函数 $f(\text{rm}) = 4.456583425777273 * \text{rm} + -5.119884501536662$, 此时loss是: 54.35960454345606
在第764步, 我们获得了函数 $f(\text{rm}) = 4.457700545823006 * \text{rm} + -5.1269906339785525$, 此时loss是: 54.354430658070974
在第765步, 我们获得了函数 $f(\text{rm}) = 4.458817397232354 * \text{rm} + -5.134095057592689$, 此时loss是: 54.34925926073867
在第766步, 我们获得了函数 $f(\text{rm}) = 4.459933980069915 * \text{rm} + -5.141197772789998$, 此时loss是: 54.34409035026246
在第767步, 我们获得了函数 $f(\text{rm}) = 4.461050294400274 * \text{rm} + -5.148298779981307$, 此时loss是: 54.33892392544655
在第768步, 我们获得了函数 $f(\text{rm}) = 4.4621663402879985 * \text{rm} + -5.155398079577342$, 此时loss是: 54.33375998509567
在第769步, 我们获得了函数 $f(\text{rm}) = 4.463282117797643 * \text{rm} + -5.162495671988734$, 此时loss是: 54.32859852801498
在第770步, 我们获得了函数 $f(\text{rm}) = 4.464397626993743 * \text{rm} + -5.1695915576260125$, 此时loss是: 54.32343955301041
在第771步, 我们获得了函数 $f(\text{rm}) = 4.465512867940823 * \text{rm} + -5.17668573689961$, 此时loss是: 54.31828305888825
在第772步, 我们获得了函数 $f(\text{rm}) = 4.466627840703387 * \text{rm} + -5.1837782102198595$, 此时loss是: 54.31312904445556
在第773步, 我们获得了函数 $f(\text{rm}) = 4.467742545345927 * \text{rm} + -5.190868977996995$, 此时loss是: 54.30797750851981
在第774步, 我们获得了函数 $f(\text{rm}) = 4.468856981932919 * \text{rm} + -5.197958040641153$, 此时loss是: 54.302828449889276
在第775步, 我们获得了函数 $f(\text{rm}) = 4.4699711505288215 * \text{rm} + -5.205045398562371$, 此时loss是: 54.297681867372525
在第776步, 我们获得了函数 $f(\text{rm}) = 4.47108505119808 * \text{rm} + -5.212131052170586$, 此时loss是: 54.29253775977885
在第777步, 我们获得了函数 $f(\text{rm}) = 4.472198684005123 * \text{rm} + -5.21921500187564$, 此时loss是: 54.287396125918065
在第778步, 我们获得了函数 $f(\text{rm}) = 4.473312049014365 * \text{rm} + -5.226297248087273$, 此时loss是: 54.282256964600656
在第779步, 我们获得了函数 $f(\text{rm}) = 4.474425146290202 * \text{rm} + -5.233377791215129$, 此时loss是: 54.27712027463754
在第780步, 我们获得了函数 $f(\text{rm}) = 4.475537975897018 * \text{rm} + -5.240456631668751$, 此时loss是: 54.27198605484035

在第781步, 我们获得了函数 $f(\text{rm}) = 4.47665053789918 * \text{rm} + -5.247533769857586$, 此时loss是: 54.26685430402119
在第782步, 我们获得了函数 $f(\text{rm}) = 4.477762832361038 * \text{rm} + -5.254609206190982$, 此时loss是: 54.26172502099274
在第783步, 我们获得了函数 $f(\text{rm}) = 4.47887485934693 * \text{rm} + -5.2616829410781865$, 此时loss是: 54.25659820456824
在第784步, 我们获得了函数 $f(\text{rm}) = 4.479986618921175 * \text{rm} + -5.268754974928351$, 此时loss是: 54.25147385356157
在第785步, 我们获得了函数 $f(\text{rm}) = 4.4810981111480785 * \text{rm} + -5.2758253081505275$, 此时loss是: 54.24635196678718
在第786步, 我们获得了函数 $f(\text{rm}) = 4.48220933609193 * \text{rm} + -5.28289394115367$, 此时loss是: 54.24123254306002
在第787步, 我们获得了函数 $f(\text{rm}) = 4.483320293817005 * \text{rm} + -5.289960874346634$, 此时loss是: 54.236115581195584
在第788步, 我们获得了函数 $f(\text{rm}) = 4.4844309843875605 * \text{rm} + -5.297026108138176$, 此时loss是: 54.23100108001011
在第789步, 我们获得了函数 $f(\text{rm}) = 4.48554140786784 * \text{rm} + -5.304089642936956$, 此时loss是: 54.22588903832024
在第790步, 我们获得了函数 $f(\text{rm}) = 4.486651564322073 * \text{rm} + -5.311151479151535$, 此时loss是: 54.2207794549432
在第791步, 我们获得了函数 $f(\text{rm}) = 4.4877614538144694 * \text{rm} + -5.318211617190374$, 此时loss是: 54.21567232869686
在第792步, 我们获得了函数 $f(\text{rm}) = 4.488871076409228 * \text{rm} + -5.325270057461838$, 此时loss是: 54.210567658399626
在第793步, 我们获得了函数 $f(\text{rm}) = 4.489980432170529 * \text{rm} + -5.332326800374192$, 此时loss是: 54.20546544287045
在第794步, 我们获得了函数 $f(\text{rm}) = 4.491089521162539 * \text{rm} + -5.339381846335605$, 此时loss是: 54.20036568092884
在第795步, 我们获得了函数 $f(\text{rm}) = 4.492198343449409 * \text{rm} + -5.346435195754146$, 此时loss是: 54.195268371394945
在第796步, 我们获得了函数 $f(\text{rm}) = 4.493306899095273 * \text{rm} + -5.353486849037786$, 此时loss是: 54.19017351308939
在第797步, 我们获得了函数 $f(\text{rm}) = 4.494415188164252 * \text{rm} + -5.3605368065944$, 此时loss是: 54.18508110483349
在第798步, 我们获得了函数 $f(\text{rm}) = 4.49552321072045 * \text{rm} + -5.367585068831762$, 此时loss是: 54.17999114544891
在第799步, 我们获得了函数 $f(\text{rm}) = 4.496630966827956 * \text{rm} + -5.37463163615755$, 此时loss是: 54.17490363375815
在第800步, 我们获得了函数 $f(\text{rm}) = 4.4977384565508425 * \text{rm} + -5.381676508979342$, 此时loss是: 54.16981856858408
在第801步, 我们获得了函数 $f(\text{rm}) = 4.4988456799531695 * \text{rm} + -5.38871968770462$, 此时loss是: 54.16473594875019
在第802步, 我们获得了函数 $f(\text{rm}) = 4.499952637098978 * \text{rm} + -5.395761172740769$, 此时loss是: 54.15965577308067
在第803步, 我们获得了函数 $f(\text{rm}) = 4.501059328052296 * \text{rm} + -5.4028009644950705$, 此时loss是: 54.15457804039999

在第804步, 我们获得了函数 $f(\text{rm}) = 4.502165752877136 * \text{rm} + -5.409839063374715$, 此时loss是: 54.14950274953341
在第805步, 我们获得了函数 $f(\text{rm}) = 4.503271911637494 * \text{rm} + -5.416875469786791$, 此时loss是: 54.144429899306765
在第806步, 我们获得了函数 $f(\text{rm}) = 4.50437780439735 * \text{rm} + -5.423910184138289$, 此时loss是: 54.13935948854631
在第807步, 我们获得了函数 $f(\text{rm}) = 4.5054834312206715 * \text{rm} + -5.430943206836104$, 此时loss是: 54.13429151607895
在第808步, 我们获得了函数 $f(\text{rm}) = 4.506588792171408 * \text{rm} + -5.437974538287031$, 此时loss是: 54.12922598073216
在第809步, 我们获得了函数 $f(\text{rm}) = 4.507693887313494 * \text{rm} + -5.445004178897768$, 此时loss是: 54.12416288133392
在第810步, 我们获得了函数 $f(\text{rm}) = 4.508798716710851 * \text{rm} + -5.452032129074915$, 此时loss是: 54.119102216712825
在第811步, 我们获得了函数 $f(\text{rm}) = 4.509903280427381 * \text{rm} + -5.4590583892249755$, 此时loss是: 54.11404398569808
在第812步, 我们获得了函数 $f(\text{rm}) = 4.511007578526975 * \text{rm} + -5.4660829597543525$, 此时loss是: 54.108988187119365
在第813步, 我们获得了函数 $f(\text{rm}) = 4.512111611073505 * \text{rm} + -5.473105841069354$, 此时loss是: 54.1039348198069
在第814步, 我们获得了函数 $f(\text{rm}) = 4.513215378130829 * \text{rm} + -5.480127033576189$, 此时loss是: 54.09888388259158
在第815步, 我们获得了函数 $f(\text{rm}) = 4.51431887976279 * \text{rm} + -5.487146537680969$, 此时loss是: 54.09383537430486
在第816步, 我们获得了函数 $f(\text{rm}) = 4.515422116033215 * \text{rm} + -5.494164353789708$, 此时loss是: 54.08878929377856
在第817步, 我们获得了函数 $f(\text{rm}) = 4.516525087005919 * \text{rm} + -5.501180482308322$, 此时loss是: 54.08374563984525
在第818步, 我们获得了函数 $f(\text{rm}) = 4.517627792744695 * \text{rm} + -5.5081949236426295$, 此时loss是: 54.07870441133811
在第819步, 我们获得了函数 $f(\text{rm}) = 4.518730233313327 * \text{rm} + -5.515207678198352$, 此时loss是: 54.07366560709069
在第820步, 我们获得了函数 $f(\text{rm}) = 4.519832408775579 * \text{rm} + -5.522218746381113$, 此时loss是: 54.068629225937165
在第821步, 我们获得了函数 $f(\text{rm}) = 4.520934319195203 * \text{rm} + -5.529228128596438$, 此时loss是: 54.06359526671242
在第822步, 我们获得了函数 $f(\text{rm}) = 4.5220359646359345 * \text{rm} + -5.536235825249755$, 此时loss是: 54.058563728251734
在第823步, 我们获得了函数 $f(\text{rm}) = 4.5231373451614925 * \text{rm} + -5.543241836746396$, 此时loss是: 54.05353460939093
在第824步, 我们获得了函数 $f(\text{rm}) = 4.524238460835583 * \text{rm} + -5.550246163491593$, 此时loss是: 54.04850790896645
在第825步, 我们获得了函数 $f(\text{rm}) = 4.525339311721895 * \text{rm} + -5.557248805890484$, 此时loss是: 54.0434836258154
在第826步, 我们获得了函数 $f(\text{rm}) = 4.526439897884102 * \text{rm} + -5.564249764348106$, 此时loss是: 54.038461758775306

在第827步, 我们获得了函数 $f(\text{rm}) = 4.527540219385864 * \text{rm} + -5.5712490392694$, 此时loss是: 54.03344230668427
在第828步, 我们获得了函数 $f(\text{rm}) = 4.528640276290823 * \text{rm} + -5.578246631059211$, 此时loss是: 54.028425268380985
在第829步, 我们获得了函数 $f(\text{rm}) = 4.529740068662609 * \text{rm} + -5.585242540122285$, 此时loss是: 54.023410642704725
在第830步, 我们获得了函数 $f(\text{rm}) = 4.530839596564833 * \text{rm} + -5.59223676686327$, 此时loss是: 54.01839842849519
在第831步, 我们获得了函数 $f(\text{rm}) = 4.531938860061095 * \text{rm} + -5.59922931168672$, 此时loss是: 54.013388624592906
在第832步, 我们获得了函数 $f(\text{rm}) = 4.533037859214975 * \text{rm} + -5.606220174997088$, 此时loss是: 54.0083812298386
在第833步, 我们获得了函数 $f(\text{rm}) = 4.53413659409004 * \text{rm} + -5.613209357198731$, 此时loss是: 54.003376243073845
在第834步, 我们获得了函数 $f(\text{rm}) = 4.535235064749843 * \text{rm} + -5.62019685869591$, 此时loss是: 53.99837366314071
在第835步, 我们获得了函数 $f(\text{rm}) = 4.53633327125792 * \text{rm} + -5.627182679892787$, 此时loss是: 53.99337348888168
在第836步, 我们获得了函数 $f(\text{rm}) = 4.537431213677793 * \text{rm} + -5.6341668211934275$, 此时loss是: 53.98837571914
在第837步, 我们获得了函数 $f(\text{rm}) = 4.538528892072966 * \text{rm} + -5.641149283001801$, 此时loss是: 53.98338035275928
在第838步, 我们获得了函数 $f(\text{rm}) = 4.539626306506931 * \text{rm} + -5.648130065721778$, 此时loss是: 53.9783873885838
在第839步, 我们获得了函数 $f(\text{rm}) = 4.540723457043163 * \text{rm} + -5.655109169757133$, 此时loss是: 53.973396825458465
在第840步, 我们获得了函数 $f(\text{rm}) = 4.541820343745122 * \text{rm} + -5.662086595511544$, 此时loss是: 53.968408662228484
在第841步, 我们获得了函数 $f(\text{rm}) = 4.542916966676253 * \text{rm} + -5.669062343388589$, 此时loss是: 53.963422897739875
在第842步, 我们获得了函数 $f(\text{rm}) = 4.544013325899986 * \text{rm} + -5.676036413791752$, 此时loss是: 53.958439530839144
在第843步, 我们获得了函数 $f(\text{rm}) = 4.545109421479734 * \text{rm} + -5.683008807124419$, 此时loss是: 53.95345856037321
在第844步, 我们获得了函数 $f(\text{rm}) = 4.546205253478897 * \text{rm} + -5.689979523789879$, 此时loss是: 53.94847998518984
在第845步, 我们获得了函数 $f(\text{rm}) = 4.547300821960858 * \text{rm} + -5.696948564191324$, 此时loss是: 53.943503804136974
在第846步, 我们获得了函数 $f(\text{rm}) = 4.548396126988988 * \text{rm} + -5.703915928731848$, 此时loss是: 53.93853001606338
在第847步, 我们获得了函数 $f(\text{rm}) = 4.549491168626638 * \text{rm} + -5.710881617814451$, 此时loss是: 53.93355861981836
在第848步, 我们获得了函数 $f(\text{rm}) = 4.550585946937146 * \text{rm} + -5.7178456318420325$, 此时loss是: 53.92858961425161
在第849步, 我们获得了函数 $f(\text{rm}) = 4.551680461983835 * \text{rm} + -5.724807971217397$, 此时loss是: 53.9236229982136

在第850步, 我们获得了函数 $f(\text{rm}) = 4.552774713830015 * \text{rm} + -5.7317686363432525$, 此时loss是: 53.918658770555204
在第851步, 我们获得了函数 $f(\text{rm}) = 4.553868702538976 * \text{rm} + -5.738727627622209$, 此时loss是: 53.91369693012787
在第852步, 我们获得了函数 $f(\text{rm}) = 4.5549624281739955 * \text{rm} + -5.745684945456781$, 此时loss是: 53.908737475783596
在第853步, 我们获得了函数 $f(\text{rm}) = 4.556055890798336 * \text{rm} + -5.752640590249385$, 此时loss是: 53.903780406374906
在第854步, 我们获得了函数 $f(\text{rm}) = 4.557149090475244 * \text{rm} + -5.7595945624023415$, 此时loss是: 53.89882572075507
在第855步, 我们获得了函数 $f(\text{rm}) = 4.558242027267951 * \text{rm} + -5.766546862317873$, 此时loss是: 53.893873417777506
在第856步, 我们获得了函数 $f(\text{rm}) = 4.559334701239674 * \text{rm} + -5.773497490398107$, 此时loss是: 53.88892349629671
在第857步, 我们获得了函数 $f(\text{rm}) = 4.560427112453613 * \text{rm} + -5.780446447045073$, 此时loss是: 53.88397595516731
在第858步, 我们获得了函数 $f(\text{rm}) = 4.561519260972955 * \text{rm} + -5.787393732660705$, 此时loss是: 53.87903079324449
在第859步, 我们获得了函数 $f(\text{rm}) = 4.5626111468608705 * \text{rm} + -5.79433934764684$, 此时loss是: 53.87408800938441
在第860步, 我们获得了函数 $f(\text{rm}) = 4.563702770180515 * \text{rm} + -5.801283292405217$, 此时loss是: 53.86914760244342
在第861步, 我们获得了函数 $f(\text{rm}) = 4.564794130995029 * \text{rm} + -5.80822556733748$, 此时loss是: 53.86420957127823
在第862步, 我们获得了函数 $f(\text{rm}) = 4.565885229367538 * \text{rm} + -5.815166172845176$, 此时loss是: 53.85927391474668
在第863步, 我们获得了函数 $f(\text{rm}) = 4.566976065361151 * \text{rm} + -5.822105109329755$, 此时loss是: 53.85434063170676
在第864步, 我们获得了函数 $f(\text{rm}) = 4.568066639038963 * \text{rm} + -5.829042377192571$, 此时loss是: 53.849409721016904
在第865步, 我们获得了函数 $f(\text{rm}) = 4.569156950464055 * \text{rm} + -5.83597797683488$, 此时loss是: 53.84448118153647
在第866步, 我们获得了函数 $f(\text{rm}) = 4.570246999699489 * \text{rm} + -5.842911908657845$, 此时loss是: 53.839555012125146
在第867步, 我们获得了函数 $f(\text{rm}) = 4.571336786808318 * \text{rm} + -5.849844173062529$, 此时loss是: 53.834631211643156
在第868步, 我们获得了函数 $f(\text{rm}) = 4.572426311853573 * \text{rm} + -5.856774770449899$, 此时loss是: 53.82970977895131
在第869步, 我们获得了函数 $f(\text{rm}) = 4.573515574898275 * \text{rm} + -5.863703701220828$, 此时loss是: 53.82479071291105
在第870步, 我们获得了函数 $f(\text{rm}) = 4.574604576005426 * \text{rm} + -5.870630965776091$, 此时loss是: 53.81987401238416
在第871步, 我们获得了函数 $f(\text{rm}) = 4.575693315238016 * \text{rm} + -5.877556564516366$, 此时loss是: 53.814959676233215
在第872步, 我们获得了函数 $f(\text{rm}) = 4.576781792659019 * \text{rm} + -5.884480497842236$, 此时loss是: 53.81004770332115

在第873步, 我们获得了函数 $f(\text{rm}) = 4.577870008331392 * \text{rm} + -5.891402766154186$, 此时loss是: 53.80513809251149
在第874步, 我们获得了函数 $f(\text{rm}) = 4.57895796231808 * \text{rm} + -5.898323369852605$, 此时loss是: 53.800230842668455
在第875步, 我们获得了函数 $f(\text{rm}) = 4.580045654682009 * \text{rm} + -5.905242309337788$, 此时loss是: 53.79532595265654
在第876步, 我们获得了函数 $f(\text{rm}) = 4.581133085486094 * \text{rm} + -5.912159585009931$, 此时loss是: 53.790423421341
在第877步, 我们获得了函数 $f(\text{rm}) = 4.582220254793232 * \text{rm} + -5.9190751972691364$, 此时loss是: 53.78552324758757
在第878步, 我们获得了函数 $f(\text{rm}) = 4.583307162666306 * \text{rm} + -5.925989146515408$, 此时loss是: 53.780625430262575
在第879步, 我们获得了函数 $f(\text{rm}) = 4.584393809168183 * \text{rm} + -5.932901433148654$, 此时loss是: 53.775729968232724
在第880步, 我们获得了函数 $f(\text{rm}) = 4.585480194361716 * \text{rm} + -5.939812057568687$, 此时loss是: 53.77083686036544
在第881步, 我们获得了函数 $f(\text{rm}) = 4.5865663183097425 * \text{rm} + -5.9467210201752225$, 此时loss是: 53.765946105528755
在第882步, 我们获得了函数 $f(\text{rm}) = 4.587652181075084 * \text{rm} + -5.953628321367882$, 此时loss是: 53.76105770259096
在第883步, 我们获得了函数 $f(\text{rm}) = 4.58873778272055 * \text{rm} + -5.960533961546188$, 此时loss是: 53.75617165042113
在第884步, 我们获得了函数 $f(\text{rm}) = 4.58982312330893 * \text{rm} + -5.967437941109569$, 此时loss是: 53.75128794788878
在第885步, 我们获得了函数 $f(\text{rm}) = 4.590908202903002 * \text{rm} + -5.974340260457358$, 此时loss是: 53.746406593864094
在第886步, 我们获得了函数 $f(\text{rm}) = 4.591993021565528 * \text{rm} + -5.981240919988788$, 此时loss是: 53.74152758721756
在第887步, 我们获得了函数 $f(\text{rm}) = 4.593077579359255 * \text{rm} + -5.988139920103001$, 此时loss是: 53.73665092682054
在第888步, 我们获得了函数 $f(\text{rm}) = 4.594161876346914 * \text{rm} + -5.995037261199041$, 此时loss是: 53.7317766115445
在第889步, 我们获得了函数 $f(\text{rm}) = 4.5952459125912215 * \text{rm} + -6.001932943675855$, 此时loss是: 53.726904640261935
在第890步, 我们获得了函数 $f(\text{rm}) = 4.5963296881548805 * \text{rm} + -6.008826967932294$, 此时loss是: 53.72203501184553
在第891步, 我们获得了函数 $f(\text{rm}) = 4.597413203100576 * \text{rm} + -6.015719334367116$, 此时loss是: 53.71716772516872
在第892步, 我们获得了函数 $f(\text{rm}) = 4.59849645749098 * \text{rm} + -6.02261004337898$, 此时loss是: 53.71230277910534
在第893步, 我们获得了函数 $f(\text{rm}) = 4.599579451388749 * \text{rm} + -6.029499095366449$, 此时loss是: 53.70744017252977
在第894步, 我们获得了函数 $f(\text{rm}) = 4.600662184856524 * \text{rm} + -6.0363864907279945$, 此时loss是: 53.702579904316956
在第895步, 我们获得了函数 $f(\text{rm}) = 4.601744657956931 * \text{rm} + -6.043272229861986$, 此时loss是: 53.69772197334266

在第896步, 我们获得了函数 $f(\text{rm}) = 4.602826870752581 * \text{rm} + -6.050156313166703$, 此时loss是: 53.6928663784826
在第897步, 我们获得了函数 $f(\text{rm}) = 4.60390882330607 * \text{rm} + -6.057038741040325$, 此时loss是: 53.68801311861366
在第898步, 我们获得了函数 $f(\text{rm}) = 4.60499051567998 * \text{rm} + -6.063919513880936$, 此时loss是: 53.683162192612805
在第899步, 我们获得了函数 $f(\text{rm}) = 4.606071947936877 * \text{rm} + -6.070798632086528$, 此时loss是: 53.678313599357644
在第900步, 我们获得了函数 $f(\text{rm}) = 4.60715312013931 * \text{rm} + -6.077676096054994$, 此时loss是: 53.673467337726606
在第901步, 我们获得了函数 $f(\text{rm}) = 4.6082340323498165 * \text{rm} + -6.0845519061841316$, 此时loss是: 53.668623406598336
在第902步, 我们获得了函数 $f(\text{rm}) = 4.609314684630918 * \text{rm} + -6.091426062871643$, 此时loss是: 53.66378180485216
在第903步, 我们获得了函数 $f(\text{rm}) = 4.610395077045118 * \text{rm} + -6.098298566515137$, 此时loss是: 53.65894253136772
在第904步, 我们获得了函数 $f(\text{rm}) = 4.61147520965491 * \text{rm} + -6.105169417512123$, 此时loss是: 53.65410558502565
在第905步, 我们获得了函数 $f(\text{rm}) = 4.612555082522768 * \text{rm} + -6.112038616260016$, 此时loss是: 53.649270964706666
在第906步, 我们获得了函数 $f(\text{rm}) = 4.613634695711152 * \text{rm} + -6.118906163156139$, 此时loss是: 53.644438669292356
在第907步, 我们获得了函数 $f(\text{rm}) = 4.61471404928251 * \text{rm} + -6.125772058597715$, 此时loss是: 53.639608697664684
在第908步, 我们获得了函数 $f(\text{rm}) = 4.615793143299271 * \text{rm} + -6.132636302981871$, 此时loss是: 53.63478104870611
在第909步, 我们获得了函数 $f(\text{rm}) = 4.616871977823851 * \text{rm} + -6.139498896705644$, 此时loss是: 53.62995572129963
在第910步, 我们获得了函数 $f(\text{rm}) = 4.617950552918651 * \text{rm} + -6.146359840165969$, 此时loss是: 53.625132714329
在第911步, 我们获得了函数 $f(\text{rm}) = 4.6190288686460566 * \text{rm} + -6.15321913375969$, 此时loss是: 53.620312026678214
在第912步, 我们获得了函数 $f(\text{rm}) = 4.6201069250684395 * \text{rm} + -6.160076777883555$, 此时loss是: 53.615493657232065
在第913步, 我们获得了函数 $f(\text{rm}) = 4.621184722248152 * \text{rm} + -6.1669327729342145$, 此时loss是: 53.61067760487566
在第914步, 我们获得了函数 $f(\text{rm}) = 4.622262260247538 * \text{rm} + -6.173787119308225$, 此时loss是: 53.605863868494836
在第915步, 我们获得了函数 $f(\text{rm}) = 4.623339539128923 * \text{rm} + -6.1806398174020485$, 此时loss是: 53.60105244697571
在第916步, 我们获得了函数 $f(\text{rm}) = 4.624416558954617 * \text{rm} + -6.187490867612049$, 此时loss是: 53.59624333920533
在第917步, 我们获得了函数 $f(\text{rm}) = 4.625493319786915 * \text{rm} + -6.194340270334498$, 此时loss是: 53.59143654407081
在第918步, 我们获得了函数 $f(\text{rm}) = 4.626569821688099 * \text{rm} + -6.20118802596557$, 此时loss是: 53.586632060460204

在第919步, 我们获得了函数 $f(\text{rm}) = 4.627646064720433 * \text{rm} + -6.208034134901346$, 此时loss是: 53.58182988726188
在第920步, 我们获得了函数 $f(\text{rm}) = 4.62872204894617 * \text{rm} + -6.214878597537808$, 此时loss是: 53.577030023364784
在第921步, 我们获得了函数 $f(\text{rm}) = 4.629797774427543 * \text{rm} + -6.221721414270848$, 此时loss是: 53.572232467658324
在第922步, 我们获得了函数 $f(\text{rm}) = 4.630873241226776 * \text{rm} + -6.22856258549626$, 此时loss是: 53.5674372190326
在第923步, 我们获得了函数 $f(\text{rm}) = 4.631948449406073 * \text{rm} + -6.235402111609741$, 此时loss是: 53.5626442763782
在第924步, 我们获得了函数 $f(\text{rm}) = 4.633023399027625 * \text{rm} + -6.242239993006896$, 此时loss是: 53.55785363858617
在第925步, 我们获得了函数 $f(\text{rm}) = 4.634098090153609 * \text{rm} + -6.249076230083233$, 此时loss是: 53.55306530454817
在第926步, 我们获得了函数 $f(\text{rm}) = 4.635172522846185 * \text{rm} + -6.255910823234166$, 此时loss是: 53.54827927315627
在第927步, 我们获得了函数 $f(\text{rm}) = 4.636246697167499 * \text{rm} + -6.262743772855014$, 此时loss是: 53.54349554330329
在第928步, 我们获得了函数 $f(\text{rm}) = 4.637320613179683 * \text{rm} + -6.269575079341$, 此时loss是: 53.538714113882335
在第929步, 我们获得了函数 $f(\text{rm}) = 4.638394270944852 * \text{rm} + -6.2764047430872525$, 此时loss是: 53.53393498378719
在第930步, 我们获得了函数 $f(\text{rm}) = 4.639467670525109 * \text{rm} + -6.283232764488804$, 此时loss是: 53.52915815191215
在第931步, 我们获得了函数 $f(\text{rm}) = 4.640540811982539 * \text{rm} + -6.290059143940593$, 此时loss是: 53.52438361715204
在第932步, 我们获得了函数 $f(\text{rm}) = 4.641613695379212 * \text{rm} + -6.296883881837464$, 此时loss是: 53.519611378402125
在第933步, 我们获得了函数 $f(\text{rm}) = 4.642686320777187 * \text{rm} + -6.303706978574164$, 此时loss是: 53.514841434558434
在第934步, 我们获得了函数 $f(\text{rm}) = 4.643758688238505 * \text{rm} + -6.310528434545348$, 此时loss是: 53.510073784517274
在第935步, 我们获得了函数 $f(\text{rm}) = 4.644830797825192 * \text{rm} + -6.3173482501455736$, 此时loss是: 53.50530842717559
在第936步, 我们获得了函数 $f(\text{rm}) = 4.64590264959926 * \text{rm} + -6.324166425769304$, 此时loss是: 53.50054536143098
在第937步, 我们获得了函数 $f(\text{rm}) = 4.646974243622706 * \text{rm} + -6.33098296181091$, 此时loss是: 53.495784586181216
在第938步, 我们获得了函数 $f(\text{rm}) = 4.648045579957512 * \text{rm} + -6.337797858664664$, 此时loss是: 53.49102610032497
在第939步, 我们获得了函数 $f(\text{rm}) = 4.649116658665644 * \text{rm} + -6.344611116724746$, 此时loss是: 53.486269902761315
在第940步, 我们获得了函数 $f(\text{rm}) = 4.650187479809054 * \text{rm} + -6.351422736385239$, 此时loss是: 53.48151599238981
在第941步, 我们获得了函数 $f(\text{rm}) = 4.65125804344968 * \text{rm} + -6.358232718040135$, 此时loss是: 53.47676436811056

在第942步, 我们获得了函数 $f(\text{rm}) = 4.652328349649444 * \text{rm} + -6.3650410620833275$, 此时loss是: 53.472015028824224
在第943步, 我们获得了函数 $f(\text{rm}) = 4.653398398470254 * \text{rm} + -6.371847768908617$, 此时loss是: 53.46726797343204
在第944步, 我们获得了函数 $f(\text{rm}) = 4.6544681899740015 * \text{rm} + -6.378652838909708$, 此时loss是: 53.462523200835584
在第945步, 我们获得了函数 $f(\text{rm}) = 4.655537724222564 * \text{rm} + -6.385456272480212$, 此时loss是: 53.45778070993724
在第946步, 我们获得了函数 $f(\text{rm}) = 4.6566070012778065 * \text{rm} + -6.3922580700136455$, 此时loss是: 53.45304049963965
在第947步, 我们获得了函数 $f(\text{rm}) = 4.657676021201574 * \text{rm} + -6.39905823190343$, 此时loss是: 53.44830256884619
在第948步, 我们获得了函数 $f(\text{rm}) = 4.658744784055701 * \text{rm} + -6.4058567585428925$, 此时loss是: 53.44356691646051
在第949步, 我们获得了函数 $f(\text{rm}) = 4.659813289902004 * \text{rm} + -6.412653650325265$, 此时loss是: 53.43883354138719
在第950步, 我们获得了函数 $f(\text{rm}) = 4.660881538802289 * \text{rm} + -6.419448907643686$, 此时loss是: 53.434102442530985
在第951步, 我们获得了函数 $f(\text{rm}) = 4.661949530818342 * \text{rm} + -6.426242530891197$, 此时loss是: 53.42937361879731
在第952步, 我们获得了函数 $f(\text{rm}) = 4.663017266011939 * \text{rm} + -6.433034520460748$, 此时loss是: 53.42464706909203
在第953步, 我们获得了函数 $f(\text{rm}) = 4.6640847444448355 * \text{rm} + -6.439824876745194$, 此时loss是: 53.419922792321664
在第954步, 我们获得了函数 $f(\text{rm}) = 4.665151966178778 * \text{rm} + -6.446613600137294$, 此时loss是: 53.41520078739319
在第955步, 我们获得了函数 $f(\text{rm}) = 4.666218931275493 * \text{rm} + -6.453400691029713$, 此时loss是: 53.41048105321401
在第956步, 我们获得了函数 $f(\text{rm}) = 4.667285639796697 * \text{rm} + -6.460186149815022$, 此时loss是: 53.40576358869228
在第957步, 我们获得了函数 $f(\text{rm}) = 4.668352091804088 * \text{rm} + -6.466969976885697$, 此时loss是: 53.401048392736534
在第958步, 我们获得了函数 $f(\text{rm}) = 4.669418287359351 * \text{rm} + -6.473752172634121$, 此时loss是: 53.396335464255706
在第959步, 我们获得了函数 $f(\text{rm}) = 4.670484226524155 * \text{rm} + -6.480532737452582$, 此时loss是: 53.39162480215956
在第960步, 我们获得了函数 $f(\text{rm}) = 4.671549909360156 * \text{rm} + -6.487311671733273$, 此时loss是: 53.386916405358186
在第961步, 我们获得了函数 $f(\text{rm}) = 4.6726153359289935 * \text{rm} + -6.4940889758682925$, 此时loss是: 53.38221027276218
在第962步, 我们获得了函数 $f(\text{rm}) = 4.673680506292292 * \text{rm} + -6.500864650249646$, 此时loss是: 53.37750640328279
在第963步, 我们获得了函数 $f(\text{rm}) = 4.674745420511662 * \text{rm} + -6.507638695269244$, 此时loss是: 53.3728047958316
在第964步, 我们获得了函数 $f(\text{rm}) = 4.6758100786486985 * \text{rm} + -6.5144111113189025$, 此时loss是: 53.368105449321

在第965步, 我们获得了函数 $f(\text{rm}) = 4.676874480764983 * \text{rm} + -6.521181898790344$, 此时loss是: 53.363408362663534
在第966步, 我们获得了函数 $f(\text{rm}) = 4.6779386269220815 * \text{rm} + -6.527951058075196$, 此时loss是: 53.35871353477267
在第967步, 我们获得了函数 $f(\text{rm}) = 4.679002517181545 * \text{rm} + -6.534718589564992$, 此时loss是: 53.35402096456205
在第968步, 我们获得了函数 $f(\text{rm}) = 4.68006615160491 * \text{rm} + -6.541484493651172$, 此时loss是: 53.349330650945994
在第969步, 我们获得了函数 $f(\text{rm}) = 4.681129530253697 * \text{rm} + -6.548248770725081$, 此时loss是: 53.34464259283939
在第970步, 我们获得了函数 $f(\text{rm}) = 4.682192653189413 * \text{rm} + -6.555011421177971$, 此时loss是: 53.339956789157654
在第971步, 我们获得了函数 $f(\text{rm}) = 4.683255520473551 * \text{rm} + -6.561772445400999$, 此时loss是: 53.33527323881651
在第972步, 我们获得了函数 $f(\text{rm}) = 4.684318132167587 * \text{rm} + -6.5685318437852285$, 此时loss是: 53.330591940732475
在第973步, 我们获得了函数 $f(\text{rm}) = 4.685380488332984 * \text{rm} + -6.575289616721628$, 此时loss是: 53.32591289382247
在第974步, 我们获得了函数 $f(\text{rm}) = 4.686442589031189 * \text{rm} + -6.582045764601073$, 此时loss是: 53.3212360970038
在第975步, 我们获得了函数 $f(\text{rm}) = 4.687504434323635 * \text{rm} + -6.588800287814344$, 此时loss是: 53.316561549194546
在第976步, 我们获得了函数 $f(\text{rm}) = 4.68856602427174 * \text{rm} + -6.595553186752129$, 此时loss是: 53.31188924931317
在第977步, 我们获得了函数 $f(\text{rm}) = 4.689627358936907 * \text{rm} + -6.602304461805021$, 此时loss是: 53.307219196278716
在第978步, 我们获得了函数 $f(\text{rm}) = 4.6906884383805245 * \text{rm} + -6.609054113363519$, 此时loss是: 53.30255138901063
在第979步, 我们获得了函数 $f(\text{rm}) = 4.691749262663967 * \text{rm} + -6.615802141818028$, 此时loss是: 53.297885826428946
在第980步, 我们获得了函数 $f(\text{rm}) = 4.692809831848592 * \text{rm} + -6.622548547558861$, 此时loss是: 53.293222507454274
在第981步, 我们获得了函数 $f(\text{rm}) = 4.693870145995746 * \text{rm} + -6.629293330976234$, 此时loss是: 53.28856143100763
在第982步, 我们获得了函数 $f(\text{rm}) = 4.694930205166756 * \text{rm} + -6.6360364924602715$, 此时loss是: 53.28390259601071
在第983步, 我们获得了函数 $f(\text{rm}) = 4.695990009422937 * \text{rm} + -6.6427780324010035$, 此时loss是: 53.279246001385616
在第984步, 我们获得了函数 $f(\text{rm}) = 4.69704955882559 * \text{rm} + -6.649517951188366$, 此时loss是: 53.2745916460549
在第985步, 我们获得了函数 $f(\text{rm}) = 4.698108853436 * \text{rm} + -6.656256249212201$, 此时loss是: 53.269939528941805
在第986步, 我们获得了函数 $f(\text{rm}) = 4.699167893315438 * \text{rm} + -6.662992926862258$, 此时loss是: 53.26528964896986
在第987步, 我们获得了函数 $f(\text{rm}) = 4.700226678525158 * \text{rm} + -6.669727984528191$, 此时loss是: 53.26064200506344

在第988步, 我们获得了函数 $f(\text{rm}) = 4.701285209126402 * \text{rm} + -6.676461422599563$, 此时loss是: 53.2559965961471
在第989步, 我们获得了函数 $f(\text{rm}) = 4.702343485180396 * \text{rm} + -6.68319324146584$, 此时loss是: 53.2513534211461
在第990步, 我们获得了函数 $f(\text{rm}) = 4.703401506748352 * \text{rm} + -6.689923441516397$, 此时loss是: 53.246712478986325
在第991步, 我们获得了函数 $f(\text{rm}) = 4.704459273891468 * \text{rm} + -6.696652023140513$, 此时loss是: 53.24207376859382
在第992步, 我们获得了函数 $f(\text{rm}) = 4.705516786670923 * \text{rm} + -6.703378986727375$, 此时loss是: 53.237437288895485
在第993步, 我们获得了函数 $f(\text{rm}) = 4.706574045147887 * \text{rm} + -6.710104332666077$, 此时loss是: 53.23280303881853
在第994步, 我们获得了函数 $f(\text{rm}) = 4.707631049383512 * \text{rm} + -6.716828061345618$, 此时loss是: 53.22817101729081
在第995步, 我们获得了函数 $f(\text{rm}) = 4.708687799438935 * \text{rm} + -6.723550173154903$, 此时loss是: 53.223541223240744
在第996步, 我们获得了函数 $f(\text{rm}) = 4.709744295375281 * \text{rm} + -6.730270668482746$, 此时loss是: 53.218913655596936
在第997步, 我们获得了函数 $f(\text{rm}) = 4.710800537253657 * \text{rm} + -6.736989547717864$, 此时loss是: 53.2142883132889
在第998步, 我们获得了函数 $f(\text{rm}) = 4.7118565251351585 * \text{rm} + -6.7437068112488845$, 此时loss是: 53.209665195246444
在第999步, 我们获得了函数 $f(\text{rm}) = 4.712912259080864 * \text{rm} + -6.750422459464339$, 此时loss是: 53.20504430039995
在第1000步, 我们获得了函数 $f(\text{rm}) = 4.713967739151837 * \text{rm} + -6.757136492752665$, 此时loss是: 53.20042562768047
在第1001步, 我们获得了函数 $f(\text{rm}) = 4.715022965409129 * \text{rm} + -6.76384891150221$, 此时loss是: 53.195809176019154
在第1002步, 我们获得了函数 $f(\text{rm}) = 4.716077937913774 * \text{rm} + -6.770559716101224$, 此时loss是: 53.19119494434804
在第1003步, 我们获得了函数 $f(\text{rm}) = 4.717132656726792 * \text{rm} + -6.7772689069378655$, 此时loss是: 53.18658293159961
在第1004步, 我们获得了函数 $f(\text{rm}) = 4.718187121909191 * \text{rm} + -6.7839764844002$, 此时loss是: 53.18197313670676
在第1005步, 我们获得了函数 $f(\text{rm}) = 4.71924133352196 * \text{rm} + -6.790682448876199$, 此时loss是: 53.17736555860294
在第1006步, 我们获得了函数 $f(\text{rm}) = 4.720295291626076 * \text{rm} + -6.797386800753741$, 此时loss是: 53.172760196222164
在第1007步, 我们获得了函数 $f(\text{rm}) = 4.721348996282502 * \text{rm} + -6.804089540420612$, 此时loss是: 53.16815704849895
在第1008步, 我们获得了函数 $f(\text{rm}) = 4.7224024475521835 * \text{rm} + -6.810790668264502$, 此时loss是: 53.16355611436825
在第1009步, 我们获得了函数 $f(\text{rm}) = 4.7234556454960535 * \text{rm} + -6.817490184673012$, 此时loss是: 53.15895739276551
在第1010步, 我们获得了函数 $f(\text{rm}) = 4.72450859017503 * \text{rm} + -6.824188090033647$, 此时loss是: 53.15436088262688

在第1011步, 我们获得了函数 $f(\text{rm}) = 4.725561281650016 * \text{rm} + -6.830884384733817$, 此时loss是: 53.14976658288885
在第1012步, 我们获得了函数 $f(\text{rm}) = 4.7266137199819 * \text{rm} + -6.837579069160844$, 此时loss是: 53.14517449248839
在第1013步, 我们获得了函数 $f(\text{rm}) = 4.727665905231556 * \text{rm} + -6.844272143701953$, 此时loss是: 53.14058461036322
在第1014步, 我们获得了函数 $f(\text{rm}) = 4.728717837459844 * \text{rm} + -6.8509636087442765$, 此时loss是: 53.135996935451224
在第1015步, 我们获得了函数 $f(\text{rm}) = 4.729769516727608 * \text{rm} + -6.857653464674855$, 此时loss是: 53.131411466691176
在第1016步, 我们获得了函数 $f(\text{rm}) = 4.730820943095678 * \text{rm} + -6.864341711880635$, 此时loss是: 53.12682820302202
在第1017步, 我们获得了函数 $f(\text{rm}) = 4.731872116624869 * \text{rm} + -6.87102835074847$, 此时loss是: 53.122247143383426
在第1018步, 我们获得了函数 $f(\text{rm}) = 4.732923037375982 * \text{rm} + -6.877713381665122$, 此时loss是: 53.11766828671548
在第1019步, 我们获得了函数 $f(\text{rm}) = 4.733973705409802 * \text{rm} + -6.884396805017256$, 此时loss是: 53.11309163195877
在第1020步, 我们获得了函数 $f(\text{rm}) = 4.735024120787104 * \text{rm} + -6.891078621191449$, 此时loss是: 53.10851717805452
在第1021步, 我们获得了函数 $f(\text{rm}) = 4.736074283568641 * \text{rm} + -6.8977588305741815$, 此时loss是: 53.10394492394425
在第1022步, 我们获得了函数 $f(\text{rm}) = 4.737124193815157 * \text{rm} + -6.904437433551843$, 此时loss是: 53.09937486857025
在第1023步, 我们获得了函数 $f(\text{rm}) = 4.73817385158738 * \text{rm} + -6.911114430510729$, 此时loss是: 53.09480701087503
在第1024步, 我们获得了函数 $f(\text{rm}) = 4.739223256946022 * \text{rm} + -6.917789821837042$, 此时loss是: 53.09024134980184
在第1025步, 我们获得了函数 $f(\text{rm}) = 4.740272409951782 * \text{rm} + -6.924463607916893$, 此时loss是: 53.08567788429437
在第1026步, 我们获得了函数 $f(\text{rm}) = 4.741321310665344 * \text{rm} + -6.931135789136299$, 此时loss是: 53.08111661329673
在第1027步, 我们获得了函数 $f(\text{rm}) = 4.742369959147378 * \text{rm} + -6.937806365881184$, 此时loss是: 53.076557535753736
在第1028步, 我们获得了函数 $f(\text{rm}) = 4.7434183554585365 * \text{rm} + -6.94447533853738$, 此时loss是: 53.072000650610384
在第1029步, 我们获得了函数 $f(\text{rm}) = 4.744466499659461 * \text{rm} + -6.951142707490626$, 此时loss是: 53.06744595681254
在第1030步, 我们获得了函数 $f(\text{rm}) = 4.7455143918107785 * \text{rm} + -6.957808473126567$, 此时loss是: 53.06289345330639
在第1031步, 我们获得了函数 $f(\text{rm}) = 4.746562031973097 * \text{rm} + -6.9644726358307585$, 此时loss是: 53.058343139038556
在第1032步, 我们获得了函数 $f(\text{rm}) = 4.747609420207015 * \text{rm} + -6.971135195988659$, 此时loss是: 53.05379501295641
在第1033步, 我们获得了函数 $f(\text{rm}) = 4.748656556573113 * \text{rm} + -6.977796153985637$, 此时loss是: 53.049249074007626

在第1034步, 我们获得了函数 $f(\text{rm}) = 4.749703441131959 * \text{rm} + -6.984455510206969$, 此时loss是: 53.04470532114035
在第1035步, 我们获得了函数 $f(\text{rm}) = 4.750750073944106 * \text{rm} + -6.991113265037836$, 此时loss是: 53.040163753303375
在第1036步, 我们获得了函数 $f(\text{rm}) = 4.751796455070091 * \text{rm} + -6.997769418863328$, 此时loss是: 53.03562436944605
在第1037步, 我们获得了函数 $f(\text{rm}) = 4.7528425845704385 * \text{rm} + -7.004423972068444$, 此时loss是: 53.03108716851803
在第1038步, 我们获得了函数 $f(\text{rm}) = 4.753888462505657 * \text{rm} + -7.011076925038087$, 此时loss是: 53.026552149469566
在第1039步, 我们获得了函数 $f(\text{rm}) = 4.75493408893624 * \text{rm} + -7.017728278157071$, 此时loss是: 53.02201931125142
在第1040步, 我们获得了函数 $f(\text{rm}) = 4.755979463922669 * \text{rm} + -7.024378031810114$, 此时loss是: 53.017488652814954
在第1041步, 我们获得了函数 $f(\text{rm}) = 4.757024587525409 * \text{rm} + -7.031026186381844$, 此时loss是: 53.012960173111885
在第1042步, 我们获得了函数 $f(\text{rm}) = 4.7580694598049105 * \text{rm} + -7.037672742256795$, 此时loss是: 53.00843387109445
在第1043步, 我们获得了函数 $f(\text{rm}) = 4.759114080821609 * \text{rm} + -7.044317699819411$, 此时loss是: 53.00390974571539
在第1044步, 我们获得了函数 $f(\text{rm}) = 4.760158450635928 * \text{rm} + -7.05096105945404$, 此时loss是: 52.99938779592814
在第1045步, 我们获得了函数 $f(\text{rm}) = 4.761202569308273 * \text{rm} + -7.05760282154494$, 此时loss是: 52.99486802068632
在第1046步, 我们获得了函数 $f(\text{rm}) = 4.762246436899039 * \text{rm} + -7.064242986476276$, 此时loss是: 52.99035041894441
在第1047步, 我们获得了函数 $f(\text{rm}) = 4.763290053468602 * \text{rm} + -7.070881554632121$, 此时loss是: 52.98583498965708
在第1048步, 我们获得了函数 $f(\text{rm}) = 4.764333419077325 * \text{rm} + -7.077518526396454$, 此时loss是: 52.981321731779545
在第1049步, 我们获得了函数 $f(\text{rm}) = 4.76537653378556 * \text{rm} + -7.084153902153164$, 此时loss是: 52.976810644267765
在第1050步, 我们获得了函数 $f(\text{rm}) = 4.766419397653639 * \text{rm} + -7.090787682286047$, 此时loss是: 52.972301726077994
在第1051步, 我们获得了函数 $f(\text{rm}) = 4.767462010741883 * \text{rm} + -7.097419867178805$, 此时loss是: 52.96779497616696
在第1052步, 我们获得了函数 $f(\text{rm}) = 4.768504373110598 * \text{rm} + -7.104050457215049$, 此时loss是: 52.963290393492116
在第1053步, 我们获得了函数 $f(\text{rm}) = 4.769546484820074 * \text{rm} + -7.110679452778299$, 此时loss是: 52.95878797701104
在第1054步, 我们获得了函数 $f(\text{rm}) = 4.770588345930589 * \text{rm} + -7.1173068542519795$, 此时loss是: 52.95428772568221
在第1055步, 我们获得了函数 $f(\text{rm}) = 4.771629956502404 * \text{rm} + -7.123932662019427$, 此时loss是: 52.94978963846437
在第1056步, 我们获得了函数 $f(\text{rm}) = 4.772671316595767 * \text{rm} + -7.130556876463882$, 此时loss是: 52.945293714316996

在第1057步, 我们获得了函数 $f(\text{rm}) = 4.773712426270912 * \text{rm} + -7.137179497968496$, 此时loss是: 52.940799952199505
在第1058步, 我们获得了函数 $f(\text{rm}) = 4.774753285588057 * \text{rm} + -7.143800526916326$, 此时loss是: 52.93630835107262
在第1059步, 我们获得了函数 $f(\text{rm}) = 4.775793894607405 * \text{rm} + -7.1504199636903385$, 此时loss是: 52.93181890989704
在第1060步, 我们获得了函数 $f(\text{rm}) = 4.776834253389147 * \text{rm} + -7.157037808673405$, 此时loss是: 52.92733162763382
在第1061步, 我们获得了函数 $f(\text{rm}) = 4.777874361993457 * \text{rm} + -7.16365406224831$, 此时loss是: 52.92284650324506
在第1062步, 我们获得了函数 $f(\text{rm}) = 4.778914220480497 * \text{rm} + -7.170268724797741$, 此时loss是: 52.91836353569294
在第1063步, 我们获得了函数 $f(\text{rm}) = 4.779953828910414 * \text{rm} + -7.176881796704296$, 此时loss是: 52.91388272394033
在第1064步, 我们获得了函数 $f(\text{rm}) = 4.780993187343338 * \text{rm} + -7.183493278350481$, 此时loss是: 52.9094040669505
在第1065步, 我们获得了函数 $f(\text{rm}) = 4.782032295839387 * \text{rm} + -7.190103170118708$, 此时loss是: 52.904927563687224
在第1066步, 我们获得了函数 $f(\text{rm}) = 4.783071154458664 * \text{rm} + -7.1967114723913$, 此时loss是: 52.90045321311487
在第1067步, 我们获得了函数 $f(\text{rm}) = 4.784109763261257 * \text{rm} + -7.203318185550486$, 此时loss是: 52.89598101419812
在第1068步, 我们获得了函数 $f(\text{rm}) = 4.785148122307239 * \text{rm} + -7.209923309978405$, 此时loss是: 52.8915109659024
在第1069步, 我们获得了函数 $f(\text{rm}) = 4.786186231656672 * \text{rm} + -7.216526846057101$, 此时loss是: 52.8870430671935
在第1070步, 我们获得了函数 $f(\text{rm}) = 4.787224091369601 * \text{rm} + -7.223128794168528$, 此时loss是: 52.882577317037594
在第1071步, 我们获得了函数 $f(\text{rm}) = 4.7882617015060545 * \text{rm} + -7.2297291546945495$, 此时loss是: 52.87811371440159
在第1072步, 我们获得了函数 $f(\text{rm}) = 4.78929906212605 * \text{rm} + -7.236327928016935$, 此时loss是: 52.87365225825272
在第1073步, 我们获得了函数 $f(\text{rm}) = 4.790336173289589 * \text{rm} + -7.242925114517362$, 此时loss是: 52.869192947558766
在第1074步, 我们获得了函数 $f(\text{rm}) = 4.791373035056659 * \text{rm} + -7.249520714577419$, 此时loss是: 52.86473578128795
在第1075步, 我们获得了函数 $f(\text{rm}) = 4.792409647487233 * \text{rm} + -7.2561147285786$, 此时loss是: 52.86028075840923
在第1076步, 我们获得了函数 $f(\text{rm}) = 4.793446010641269 * \text{rm} + -7.262707156902309$, 此时loss是: 52.85582787789179
在第1077步, 我们获得了函数 $f(\text{rm}) = 4.794482124578712 * \text{rm} + -7.269297999929855$, 此时loss是: 52.851377138705274
在第1078步, 我们获得了函数 $f(\text{rm}) = 4.795517989359491 * \text{rm} + -7.27588725804246$, 此时loss是: 52.84692853982002
在第1079步, 我们获得了函数 $f(\text{rm}) = 4.7965536050435205 * \text{rm} + -7.282474931621252$, 此时loss是: 52.84248208020682

在第1080步, 我们获得了函数 $f(\text{rm}) = 4.797588971690703 * \text{rm} + -7.289061021047266$, 此时loss是: 52.838037758836926
在第1081步, 我们获得了函数 $f(\text{rm}) = 4.798624089360924 * \text{rm} + -7.295645526701449$, 此时loss是: 52.8335955746821
在第1082步, 我们获得了函数 $f(\text{rm}) = 4.7996589581140565 * \text{rm} + -7.3022284489646525$, 此时loss是: 52.82915552671451
在第1083步, 我们获得了函数 $f(\text{rm}) = 4.800693578009957 * \text{rm} + -7.308809788217638$, 此时loss是: 52.82471761390691
在第1084步, 我们获得了函数 $f(\text{rm}) = 4.80172794910847 * \text{rm} + -7.315389544841076$, 此时loss是: 52.82028183523268
在第1085步, 我们获得了函数 $f(\text{rm}) = 4.802762071469422 * \text{rm} + -7.321967719215546$, 此时loss是: 52.81584818966535
在第1086步, 我们获得了函数 $f(\text{rm}) = 4.80379594515263 * \text{rm} + -7.328544311721533$, 此时loss是: 52.81141667617917
在第1087步, 我们获得了函数 $f(\text{rm}) = 4.804829570217893 * \text{rm} + -7.335119322739434$, 此时loss是: 52.806987293748946
在第1088步, 我们获得了函数 $f(\text{rm}) = 4.805862946724997 * \text{rm} + -7.341692752649553$, 此时loss是: 52.802560041349764
在第1089步, 我们获得了函数 $f(\text{rm}) = 4.8068960747337135 * \text{rm} + -7.348264601832102$, 此时loss是: 52.79813491795744
在第1090步, 我们获得了函数 $f(\text{rm}) = 4.807928954303798 * \text{rm} + -7.354834870667202$, 此时loss是: 52.79371192254799
在第1091步, 我们获得了函数 $f(\text{rm}) = 4.808961585494995 * \text{rm} + -7.361403559534883$, 此时loss是: 52.78929105409832
在第1092步, 我们获得了函数 $f(\text{rm}) = 4.809993968367032 * \text{rm} + -7.367970668815083$, 此时loss是: 52.78487231158547
在第1093步, 我们获得了函数 $f(\text{rm}) = 4.811026102979624 * \text{rm} + -7.374536198887649$, 此时loss是: 52.78045569398716
在第1094步, 我们获得了函数 $f(\text{rm}) = 4.812057989392468 * \text{rm} + -7.3811001501323386$, 此时loss是: 52.776041200281405
在第1095步, 我们获得了函数 $f(\text{rm}) = 4.813089627665251 * \text{rm} + -7.387662522928814$, 此时loss是: 52.77162882944706
在第1096步, 我们获得了函数 $f(\text{rm}) = 4.8141210178576435 * \text{rm} + -7.394223317656648$, 此时loss是: 52.767218580463116
在第1097步, 我们获得了函数 $f(\text{rm}) = 4.815152160029301 * \text{rm} + -7.400782534695323$, 此时loss是: 52.76281045230929
在第1098步, 我们获得了函数 $f(\text{rm}) = 4.816183054239868 * \text{rm} + -7.40734017442423$, 此时loss是: 52.75840444396564
在第1099步, 我们获得了函数 $f(\text{rm}) = 4.81721370054897 * \text{rm} + -7.4138962372226676$, 此时loss是: 52.754000554412784
在第1100步, 我们获得了函数 $f(\text{rm}) = 4.818244099016221 * \text{rm} + -7.420450723469844$, 此时loss是: 52.74959878263191
在第1101步, 我们获得了函数 $f(\text{rm}) = 4.81927424970122 * \text{rm} + -7.427003633544876$, 此时loss是: 52.74519912760446
在第1102步, 我们获得了函数 $f(\text{rm}) = 4.820304152663552 * \text{rm} + -7.433554967826789$, 此时loss是: 52.740801588312614

在第1103步, 我们获得了函数 $f(\text{rm}) = 4.821333807962787 * \text{rm} + -7.440104726694519$, 此时loss是: 52.73640616373895
在第1104步, 我们获得了函数 $f(\text{rm}) = 4.822363215658482 * \text{rm} + -7.4466529105269075$, 此时loss是: 52.7320128528665
在第1105步, 我们获得了函数 $f(\text{rm}) = 4.823392375810179 * \text{rm} + -7.453199519702708$, 此时loss是: 52.727621654678806
在第1106步, 我们获得了函数 $f(\text{rm}) = 4.824421288477404 * \text{rm} + -7.459744554600581$, 此时loss是: 52.723232568159936
在第1107步, 我们获得了函数 $f(\text{rm}) = 4.825449953719671 * \text{rm} + -7.466288015599098$, 此时loss是: 52.71884559229444
在第1108步, 我们获得了函数 $f(\text{rm}) = 4.826478371596479 * \text{rm} + -7.472829903076736$, 此时loss是: 52.714460726067195
在第1109步, 我们获得了函数 $f(\text{rm}) = 4.827506542167312 * \text{rm} + -7.479370217411884$, 此时loss是: 52.710077968463864
在第1110步, 我们获得了函数 $f(\text{rm}) = 4.82853446549164 * \text{rm} + -7.485908958982841$, 此时loss是: 52.70569731847045
在第1111步, 我们获得了函数 $f(\text{rm}) = 4.82956214162892 * \text{rm} + -7.492446128167811$, 此时loss是: 52.70131877507322
在第1112步, 我们获得了函数 $f(\text{rm}) = 4.830589570638593 * \text{rm} + -7.498981725344909$, 此时loss是: 52.69694233725933
在第1113步, 我们获得了函数 $f(\text{rm}) = 4.831616752580087 * \text{rm} + -7.505515750892161$, 此时loss是: 52.69256800401617
在第1114步, 我们获得了函数 $f(\text{rm}) = 4.832643687512814 * \text{rm} + -7.512048205187499$, 此时loss是: 52.68819577433171
在第1115步, 我们获得了函数 $f(\text{rm}) = 4.8336703754961725 * \text{rm} + -7.5185790886087664$, 此时loss是: 52.68382564719433
在第1116步, 我们获得了函数 $f(\text{rm}) = 4.834696816589548 * \text{rm} + -7.525108401533714$, 此时loss是: 52.67945762159302
在第1117步, 我们获得了函数 $f(\text{rm}) = 4.835723010852311 * \text{rm} + -7.531636144340004$, 此时loss是: 52.67509169651708
在第1118步, 我们获得了函数 $f(\text{rm}) = 4.836748958343817 * \text{rm} + -7.538162317405204$, 此时loss是: 52.67072787095644
在第1119步, 我们获得了函数 $f(\text{rm}) = 4.837774659123406 * \text{rm} + -7.544686921106796$, 此时loss是: 52.6663661439015
在第1120步, 我们获得了函数 $f(\text{rm}) = 4.838800113250407 * \text{rm} + -7.551209955822166$, 此时loss是: 52.662006514343055
在第1121步, 我们获得了函数 $f(\text{rm}) = 4.839825320784135 * \text{rm} + -7.557731421928613$, 此时loss是: 52.657648981272494
在第1122步, 我们获得了函数 $f(\text{rm}) = 4.840850281783885 * \text{rm} + -7.564251319803344$, 此时loss是: 52.65329354368165
在第1123步, 我们获得了函数 $f(\text{rm}) = 4.841874996308944 * \text{rm} + -7.570769649823475$, 此时loss是: 52.648940200562784
在第1124步, 我们获得了函数 $f(\text{rm}) = 4.84289946441858 * \text{rm} + -7.577286412366031$, 此时loss是: 52.64458895090879
在第1125步, 我们获得了函数 $f(\text{rm}) = 4.843923686172053 * \text{rm} + -7.583801607807947$, 此时loss是: 52.64023979371285

在第1126步, 我们获得了函数 $f(\text{rm}) = 4.844947661628601 * \text{rm} + -7.590315236526068$, 此时loss是: 52.635892727968724
在第1127步, 我们获得了函数 $f(\text{rm}) = 4.8459713908474535 * \text{rm} + -7.596827298897147$, 此时loss是: 52.631547752670755
在第1128步, 我们获得了函数 $f(\text{rm}) = 4.846994873887824 * \text{rm} + -7.603337795297848$, 此时loss是: 52.6272048668136
在第1129步, 我们获得了函数 $f(\text{rm}) = 4.84801811080891 * \text{rm} + -7.609846726104742$, 此时loss是: 52.62286406939244
在第1130步, 我们获得了函数 $f(\text{rm}) = 4.849041101669899 * \text{rm} + -7.616354091694311$, 此时loss是: 52.61852535940311
在第1131步, 我们获得了函数 $f(\text{rm}) = 4.8500638465299595 * \text{rm} + -7.622859892442947$, 此时loss是: 52.61418873584168
在第1132步, 我们获得了函数 $f(\text{rm}) = 4.851086345448247 * \text{rm} + -7.629364128726952$, 此时loss是: 52.609854197704855
在第1133步, 我们获得了函数 $f(\text{rm}) = 4.852108598483907 * \text{rm} + -7.635866800922534$, 此时loss是: 52.605521743989826
在第1134步, 我们获得了函数 $f(\text{rm}) = 4.853130605696065 * \text{rm} + -7.642367909405814$, 此时loss是: 52.601191373694036
在第1135步, 我们获得了函数 $f(\text{rm}) = 4.8541523671438345 * \text{rm} + -7.648867454552821$, 此时loss是: 52.59686308581579
在第1136步, 我们获得了函数 $f(\text{rm}) = 4.855173882886317 * \text{rm} + -7.655365436739495$, 此时loss是: 52.592536879353666
在第1137步, 我们获得了函数 $f(\text{rm}) = 4.856195152982596 * \text{rm} + -7.661861856341684$, 此时loss是: 52.588212753306635
在第1138步, 我们获得了函数 $f(\text{rm}) = 4.857216177491743 * \text{rm} + -7.668356713735146$, 此时loss是: 52.5838907066743
在第1139步, 我们获得了函数 $f(\text{rm}) = 4.858236956472816 * \text{rm} + -7.6748500092955485$, 此时loss是: 52.57957073845675
在第1140步, 我们获得了函数 $f(\text{rm}) = 4.859257489984856 * \text{rm} + -7.6813417433984705$, 此时loss是: 52.57525284765439
在第1141步, 我们获得了函数 $f(\text{rm}) = 4.860277778086893 * \text{rm} + -7.687831916419398$, 此时loss是: 52.57093703326839
在第1142步, 我们获得了函数 $f(\text{rm}) = 4.86129782083794 * \text{rm} + -7.694320528733728$, 此时loss是: 52.566623294300086
在第1143步, 我们获得了函数 $f(\text{rm}) = 4.8623176182969985 * \text{rm} + -7.700807580716768$, 此时loss是: 52.56231162975144
在第1144步, 我们获得了函数 $f(\text{rm}) = 4.863337170523053 * \text{rm} + -7.707293072743734$, 此时loss是: 52.55800203862496
在第1145步, 我们获得了函数 $f(\text{rm}) = 4.864356477575076 * \text{rm} + -7.713777005189751$, 此时loss是: 52.55369451992359
在第1146步, 我们获得了函数 $f(\text{rm}) = 4.8653755395120255 * \text{rm} + -7.720259378429857$, 此时loss是: 52.54938907265061
在第1147步, 我们获得了函数 $f(\text{rm}) = 4.866394356392844 * \text{rm} + -7.726740192838996$, 此时loss是: 52.54508569581003
在第1148步, 我们获得了函数 $f(\text{rm}) = 4.86741292827646 * \text{rm} + -7.733219448792024$, 此时loss是: 52.540784388406095

在第1149步, 我们获得了函数 $f(\text{rm}) = 4.8684312552217905 * \text{rm} + -7.7396971466637074$, 此时loss是: 52.53648514944363

在第1150步, 我们获得了函数 $f(\text{rm}) = 4.869449337287736 * \text{rm} + -7.7461732868287205$, 此时loss是: 52.532187977928054

在第1151步, 我们获得了函数 $f(\text{rm}) = 4.870467174533181 * \text{rm} + -7.752647869661649$, 此时loss是: 52.52789287286508

在第1152步, 我们获得了函数 $f(\text{rm}) = 4.871484767017001 * \text{rm} + -7.759120895536988$, 此时loss是: 52.52359983326105

在第1153步, 我们获得了函数 $f(\text{rm}) = 4.872502114798053 * \text{rm} + -7.765592364829143$, 此时loss是: 52.51930885812264

在第1154步, 我们获得了函数 $f(\text{rm}) = 4.87351921793518 * \text{rm} + -7.772062277912428$, 此时loss是: 52.51501994645713

在第1155步, 我们获得了函数 $f(\text{rm}) = 4.874536076487215 * \text{rm} + -7.7785306351610695$, 此时loss是: 52.51073309727215

在第1156步, 我们获得了函数 $f(\text{rm}) = 4.87555269051297 * \text{rm} + -7.784997436949202$, 此时loss是: 52.506448309575994

在第1157步, 我们获得了函数 $f(\text{rm}) = 4.876569060071251 * \text{rm} + -7.79146268365087$, 此时loss是: 52.50216558237722

在第1158步, 我们获得了函数 $f(\text{rm}) = 4.877585185220843 * \text{rm} + -7.79792637564003$, 此时loss是: 52.497884914685024

在第1159步, 我们获得了函数 $f(\text{rm}) = 4.87860106602052 * \text{rm} + -7.8043885132905455$, 此时loss是: 52.49360630550897

在第1160步, 我们获得了函数 $f(\text{rm}) = 4.8796167025290424 * \text{rm} + -7.8108490969761935$, 此时loss是: 52.489329753859124

在第1161步, 我们获得了函数 $f(\text{rm}) = 4.880632094805154 * \text{rm} + -7.817308127070659$, 此时loss是: 52.48505525874616

在第1162步, 我们获得了函数 $f(\text{rm}) = 4.8816472429075874 * \text{rm} + -7.823765603947536$, 此时loss是: 52.48078281918101

在第1163步, 我们获得了函数 $f(\text{rm}) = 4.882662146895059 * \text{rm} + -7.830221527980332$, 此时loss是: 52.476512434175234

在第1164步, 我们获得了函数 $f(\text{rm}) = 4.88367680682627 * \text{rm} + -7.836675899542463$, 此时loss是: 52.47224410274076

在第1165步, 我们获得了函数 $f(\text{rm}) = 4.884691222759911 * \text{rm} + -7.843128719007255$, 此时loss是: 52.46797782389017

在第1166步, 我们获得了函数 $f(\text{rm}) = 4.885705394754656 * \text{rm} + -7.849579986747943$, 此时loss是: 52.46371359663632

在第1167步, 我们获得了函数 $f(\text{rm}) = 4.886719322869167 * \text{rm} + -7.856029703137675$, 此时loss是: 52.45945141999262

在第1168步, 我们获得了函数 $f(\text{rm}) = 4.887733007162089 * \text{rm} + -7.862477868549507$, 此时loss是: 52.455191292972984

在第1169步, 我们获得了函数 $f(\text{rm}) = 4.8887464476920535 * \text{rm} + -7.868924483356407$, 此时loss是: 52.45093321459176

在第1170步, 我们获得了函数 $f(\text{rm}) = 4.88975964451768 * \text{rm} + -7.875369547931252$, 此时loss是: 52.44667718386392

在第1171步, 我们获得了函数 $f(\text{rm}) = 4.890772597697573 * \text{rm} + -7.88181306264683$, 此时loss是: 52.44242319980444

在第1172步, 我们获得了函数 $f(\text{rm}) = 4.891785307290322 * \text{rm} + -7.888255027875839$, 此时loss是: 52.43817126142955
在第1173步, 我们获得了函数 $f(\text{rm}) = 4.892797773354502 * \text{rm} + -7.894695443990888$, 此时loss是: 52.43392136775517
在第1174步, 我们获得了函数 $f(\text{rm}) = 4.8938099959486765 * \text{rm} + -7.901134311364495$, 此时loss是: 52.42967351779808
在第1175步, 我们获得了函数 $f(\text{rm}) = 4.894821975131392 * \text{rm} + -7.90757163036909$, 此时loss是: 52.42542771057561
在第1176步, 我们获得了函数 $f(\text{rm}) = 4.895833710961184 * \text{rm} + -7.914007401377013$, 此时loss是: 52.4211839451054
在第1177步, 我们获得了函数 $f(\text{rm}) = 4.89684520349657 * \text{rm} + -7.920441624760514$, 此时loss是: 52.416942220405396
在第1178步, 我们获得了函数 $f(\text{rm}) = 4.897856452796056 * \text{rm} + -7.926874300891753$, 此时loss是: 52.41270253549456
在第1179步, 我们获得了函数 $f(\text{rm}) = 4.898867458918135 * \text{rm} + -7.933305430142802$, 此时loss是: 52.40846488939172
在第1180步, 我们获得了函数 $f(\text{rm}) = 4.899878221921282 * \text{rm} + -7.939735012885643$, 此时loss是: 52.404229281116564
在第1181步, 我们获得了函数 $f(\text{rm}) = 4.900888741863962 * \text{rm} + -7.946163049492169$, 此时loss是: 52.399995709689115
在第1182步, 我们获得了函数 $f(\text{rm}) = 4.901899018804625 * \text{rm} + -7.9525895403341815$, 此时loss是: 52.39576417412977
在第1183步, 我们获得了函数 $f(\text{rm}) = 4.9029090528017045 * \text{rm} + -7.959014485783395$, 此时loss是: 52.39153467345973
在第1184步, 我们获得了函数 $f(\text{rm}) = 4.903918843913623 * \text{rm} + -7.965437886211432$, 此时loss是: 52.38730720670024
在第1185步, 我们获得了函数 $f(\text{rm}) = 4.904928392198787 * \text{rm} + -7.971859741989829$, 此时loss是: 52.38308177287333
在第1186步, 我们获得了函数 $f(\text{rm}) = 4.905937697715589 * \text{rm} + -7.97828005349003$, 此时loss是: 52.37885837100122
在第1187步, 我们获得了函数 $f(\text{rm}) = 4.90694676052241 * \text{rm} + -7.984698821083392$, 此时loss是: 52.374637000107015
在第1188步, 我们获得了函数 $f(\text{rm}) = 4.907955580677613 * \text{rm} + -7.991116045141182$, 此时loss是: 52.3704176592139
在第1189步, 我们获得了函数 $f(\text{rm}) = 4.90896415823955 * \text{rm} + -7.997531726034577$, 此时loss是: 52.36620034734573
在第1190步, 我们获得了函数 $f(\text{rm}) = 4.909972493266557 * \text{rm} + -8.003945864134664$, 此时loss是: 52.361985063526696
在第1191步, 我们获得了函数 $f(\text{rm}) = 4.910980585816958 * \text{rm} + -8.010358459812444$, 此时loss是: 52.357771806781656
在第1192步, 我们获得了函数 $f(\text{rm}) = 4.911988435949062 * \text{rm} + -8.016769513438827$, 此时loss是: 52.353560576135685
在第1193步, 我们获得了函数 $f(\text{rm}) = 4.912996043721162 * \text{rm} + -8.023179025384632$, 此时loss是: 52.34935137061456
在第1194步, 我们获得了函数 $f(\text{rm}) = 4.914003409191542 * \text{rm} + -8.02958699602059$, 此时loss是: 52.34514418924441

在第1195步, 我们获得了函数 $f(\text{rm}) = 4.915010532418466 * \text{rm} + -8.035993425717345$, 此时loss是: 52.34093903105178
在第1196步, 我们获得了函数 $f(\text{rm}) = 4.916017413460188 * \text{rm} + -8.04239831484545$, 此时loss是: 52.33673589506379
在第1197步, 我们获得了函数 $f(\text{rm}) = 4.917024052374947 * \text{rm} + -8.048801663775368$, 此时loss是: 52.33253478030803
在第1198步, 我们获得了函数 $f(\text{rm}) = 4.918030449220968 * \text{rm} + -8.055203472877475$, 此时loss是: 52.32833568581249
在第1199步, 我们获得了函数 $f(\text{rm}) = 4.919036604056461 * \text{rm} + -8.061603742522056$, 此时loss是: 52.324138610605644
在第1200步, 我们获得了函数 $f(\text{rm}) = 4.920042516939623 * \text{rm} + -8.068002473079309$, 此时loss是: 52.319943553716406
在第1201步, 我们获得了函数 $f(\text{rm}) = 4.921048187928638 * \text{rm} + -8.07439966491934$, 此时loss是: 52.315750514174276
在第1202步, 我们获得了函数 $f(\text{rm}) = 4.922053617081673 * \text{rm} + -8.08079531841217$, 此时loss是: 52.31155949100902
在第1203步, 我们获得了函数 $f(\text{rm}) = 4.9230588044568835 * \text{rm} + -8.087189433927726$, 此时loss是: 52.30737048325118
在第1204步, 我们获得了函数 $f(\text{rm}) = 4.924063750112412 * \text{rm} + -8.093582011835851$, 此时loss是: 52.30318348993137
在第1205步, 我们获得了函数 $f(\text{rm}) = 4.925068454106382 * \text{rm} + -8.099973052506297$, 此时loss是: 52.29899851008089
在第1206步, 我们获得了函数 $f(\text{rm}) = 4.926072916496909 * \text{rm} + -8.106362556308726$, 此时loss是: 52.29481554273173
在第1207步, 我们获得了函数 $f(\text{rm}) = 4.927077137342092 * \text{rm} + -8.112750523612712$, 此时loss是: 52.29063458691591
在第1208步, 我们获得了函数 $f(\text{rm}) = 4.928081116700015 * \text{rm} + -8.119136954787741$, 此时loss是: 52.28645564166605
在第1209步, 我们获得了函数 $f(\text{rm}) = 4.929084854628749 * \text{rm} + -8.12552185020321$, 此时loss是: 52.28227870601545
在第1210步, 我们获得了函数 $f(\text{rm}) = 4.930088351186351 * \text{rm} + -8.131905210228426$, 此时loss是: 52.27810377899758
在第1211步, 我们获得了函数 $f(\text{rm}) = 4.931091606430864 * \text{rm} + -8.138287035232608$, 此时loss是: 52.27393085964666
在第1212步, 我们获得了函数 $f(\text{rm}) = 4.932094620420319 * \text{rm} + -8.144667325584885$, 此时loss是: 52.2697599469971
在第1213步, 我们获得了函数 $f(\text{rm}) = 4.933097393212728 * \text{rm} + -8.151046081654298$, 此时loss是: 52.26559104008408
在第1214步, 我们获得了函数 $f(\text{rm}) = 4.934099924866095 * \text{rm} + -8.157423303809802$, 此时loss是: 52.26142413794279
在第1215步, 我们获得了函数 $f(\text{rm}) = 4.935102215438405 * \text{rm} + -8.163798992420258$, 此时loss是: 52.25725923960943
在第1216步, 我们获得了函数 $f(\text{rm}) = 4.936104264987633 * \text{rm} + -8.170173147854442$, 此时loss是: 52.25309634412019
在第1217步, 我们获得了函数 $f(\text{rm}) = 4.937106073571737 * \text{rm} + -8.17654577048104$, 此时loss是: 52.24893545051212

在第1218步, 我们获得了函数 $f(\text{rm}) = 4.938107641248664 * \text{rm} + -8.182916860668652$, 此时loss是: 52.24477655782238
在第1219步, 我们获得了函数 $f(\text{rm}) = 4.9391089680763445 * \text{rm} + -8.189286418785784$, 此时loss是: 52.24061966508889
在第1220步, 我们获得了函数 $f(\text{rm}) = 4.940110054112695 * \text{rm} + -8.195654445200859$, 此时loss是: 52.23646477134975
在第1221步, 我们获得了函数 $f(\text{rm}) = 4.941110899415622 * \text{rm} + -8.202020940282207$, 此时loss是: 52.23231187564379
在第1222步, 我们获得了函数 $f(\text{rm}) = 4.942111504043012 * \text{rm} + -8.208385904398071$, 此时loss是: 52.22816097701016
在第1223步, 我们获得了函数 $f(\text{rm}) = 4.9431118680527435 * \text{rm} + -8.214749337916606$, 此时loss是: 52.224012074488456
在第1224步, 我们获得了函数 $f(\text{rm}) = 4.944111991502677 * \text{rm} + -8.221111241205879$, 此时loss是: 52.219865167118805
在第1225步, 我们获得了函数 $f(\text{rm}) = 4.94511187445066 * \text{rm} + -8.227471614633869$, 此时loss是: 52.215720253941754
在第1226步, 我们获得了函数 $f(\text{rm}) = 4.946111516954528 * \text{rm} + -8.233830458568463$, 此时loss是: 52.211577333998285
在第1227步, 我们获得了函数 $f(\text{rm}) = 4.9471109190721 * \text{rm} + -8.240187773377462$, 此时loss是: 52.207436406329975
在第1228步, 我们获得了函数 $f(\text{rm}) = 4.948110080861182 * \text{rm} + -8.246543559428579$, 此时loss是: 52.203297469978665
在第1229步, 我们获得了函数 $f(\text{rm}) = 4.9491090023795685 * \text{rm} + -8.252897817089437$, 此时loss是: 52.199160523986826
在第1230步, 我们获得了函数 $f(\text{rm}) = 4.950107683685035 * \text{rm} + -8.259250546727573$, 此时loss是: 52.19502556739725
在第1231步, 我们获得了函数 $f(\text{rm}) = 4.951106124835348 * \text{rm} + -8.265601748710433$, 此时loss是: 52.19089259925339
在第1232步, 我们获得了函数 $f(\text{rm}) = 4.952104325888257 * \text{rm} + -8.271951423405374$, 此时loss是: 52.18676161859884
在第1233步, 我们获得了函数 $f(\text{rm}) = 4.953102286901499 * \text{rm} + -8.27829957117967$, 此时loss是: 52.18263262447804
在第1234步, 我们获得了函数 $f(\text{rm}) = 4.954100007932798 * \text{rm} + -8.284646192400501$, 此时loss是: 52.17850561593556
在第1235步, 我们获得了函数 $f(\text{rm}) = 4.955097489039862 * \text{rm} + -8.290991287434961$, 此时loss是: 52.17438059201653
在第1236步, 我们获得了函数 $f(\text{rm}) = 4.956094730280387 * \text{rm} + -8.297334856650055$, 此时loss是: 52.17025755176672
在第1237步, 我们获得了函数 $f(\text{rm}) = 4.957091731712052 * \text{rm} + -8.303676900412702$, 此时loss是: 52.16613649423215
在第1238步, 我们获得了函数 $f(\text{rm}) = 4.958088493392527 * \text{rm} + -8.310017419089728$, 此时loss是: 52.16201741845927
在第1239步, 我们获得了函数 $f(\text{rm}) = 4.959085015379464 * \text{rm} + -8.316356413047878$, 此时loss是: 52.157900323495184
在第1240步, 我们获得了函数 $f(\text{rm}) = 4.960081297730504 * \text{rm} + -8.3226938826538$, 此时loss是: 52.15378520838732

在第1241步, 我们获得了函数 $f(\text{rm}) = 4.961077340503271 * \text{rm} + -8.329029828274061$, 此时loss是: 52.149672072183506
在第1242步, 我们获得了函数 $f(\text{rm}) = 4.9620731437553784 * \text{rm} + -8.335364250275138$, 此时loss是: 52.14556091393229
在第1243步, 我们获得了函数 $f(\text{rm}) = 4.963068707544424 * \text{rm} + -8.341697149023418$, 此时loss是: 52.14145173268241
在第1244步, 我们获得了函数 $f(\text{rm}) = 4.9640640319279905 * \text{rm} + -8.3480285248852$, 此时loss是: 52.13734452748301
在第1245步, 我们获得了函数 $f(\text{rm}) = 4.96505911696365 * \text{rm} + -8.354358378226697$, 此时loss是: 52.13323929738413
在第1246步, 我们获得了函数 $f(\text{rm}) = 4.966053962708959 * \text{rm} + -8.360686709414034$, 此时loss是: 52.12913604143568
在第1247步, 我们获得了函数 $f(\text{rm}) = 4.967048569221459 * \text{rm} + -8.367013518813247$, 此时loss是: 52.125034758688464
在第1248步, 我们获得了函数 $f(\text{rm}) = 4.968042936558681 * \text{rm} + -8.373338806790281$, 此时loss是: 52.12093544819362
在第1249步, 我们获得了函数 $f(\text{rm}) = 4.969037064778137 * \text{rm} + -8.379662573710998$, 此时loss是: 52.11683810900268
在第1250步, 我们获得了函数 $f(\text{rm}) = 4.97003095393733 * \text{rm} + -8.38598481994117$, 此时loss是: 52.11274274016755
在第1251步, 我们获得了函数 $f(\text{rm}) = 4.971024604093748 * \text{rm} + -8.39230554584648$, 此时loss是: 52.10864934074094
在第1252步, 我们获得了函数 $f(\text{rm}) = 4.972018015304864 * \text{rm} + -8.398624751792523$, 此时loss是: 52.104557909775615
在第1253步, 我们获得了函数 $f(\text{rm}) = 4.973011187628137 * \text{rm} + -8.40494243814481$, 此时loss是: 52.1004684463251
在第1254步, 我们获得了函数 $f(\text{rm}) = 4.974004121121013 * \text{rm} + -8.411258605268758$, 此时loss是: 52.09638094944307
在第1255步, 我们获得了函数 $f(\text{rm}) = 4.974996815840924 * \text{rm} + -8.417573253529701$, 此时loss是: 52.09229541818402
在第1256步, 我们获得了函数 $f(\text{rm}) = 4.97598927184529 * \text{rm} + -8.423886383292883$, 此时loss是: 52.088211851602594
在第1257步, 我们获得了函数 $f(\text{rm}) = 4.9769814891915125 * \text{rm} + -8.43019799492346$, 此时loss是: 52.084130248753986
在第1258步, 我们获得了函数 $f(\text{rm}) = 4.977973467936986 * \text{rm} + -8.436508088786503$, 此时loss是: 52.08005060869393
在第1259步, 我们获得了函数 $f(\text{rm}) = 4.978965208139083 * \text{rm} + -8.44281666524699$, 此时loss是: 52.07597293047851
在第1260步, 我们获得了函数 $f(\text{rm}) = 4.97995670985517 * \text{rm} + -8.449123724669818$, 此时loss是: 52.071897213164306
在第1261步, 我们获得了函数 $f(\text{rm}) = 4.980947973142594 * \text{rm} + -8.455429267419788$, 此时loss是: 52.067823455808394
在第1262步, 我们获得了函数 $f(\text{rm}) = 4.981938998058691 * \text{rm} + -8.46173329386162$, 此时loss是: 52.06375165746813
在第1263步, 我们获得了函数 $f(\text{rm}) = 4.982929784660785 * \text{rm} + -8.468035804359944$, 此时loss是: 52.05968181720163

在第1264步, 我们获得了函数 $f(\text{rm}) = 4.983920333006181 * \text{rm} + -8.474336799279303$, 此时loss是: 52.05561393406716
在第1265步, 我们获得了函数 $f(\text{rm}) = 4.984910643152174 * \text{rm} + -8.480636278984152$, 此时loss是: 52.05154800712349
在第1266步, 我们获得了函数 $f(\text{rm}) = 4.9859007151560455 * \text{rm} + -8.486934243838856$, 此时loss是: 52.047484035430095
在第1267步, 我们获得了函数 $f(\text{rm}) = 4.986890549075061 * \text{rm} + -8.493230694207696$, 此时loss是: 52.04342201804666
在第1268步, 我们获得了函数 $f(\text{rm}) = 4.987880144966473 * \text{rm} + -8.499525630454864$, 此时loss是: 52.039361954033325
在第1269步, 我们获得了函数 $f(\text{rm}) = 4.988869502887521 * \text{rm} + -8.505819052944462$, 此时loss是: 52.03530384245079
在第1270步, 我们获得了函数 $f(\text{rm}) = 4.989858622895431 * \text{rm} + -8.51211096204051$, 此时loss是: 52.03124768236019
在第1271步, 我们获得了函数 $f(\text{rm}) = 4.990847505047413 * \text{rm} + -8.518401358106933$, 此时loss是: 52.02719347282293
在第1272步, 我们获得了函数 $f(\text{rm}) = 4.991836149400666 * \text{rm} + -8.524690241507578$, 此时loss是: 52.02314121290121
在第1273步, 我们获得了函数 $f(\text{rm}) = 4.992824556012373 * \text{rm} + -8.530977612606195$, 此时loss是: 52.01909090165733
在第1274步, 我们获得了函数 $f(\text{rm}) = 4.993812724939705 * \text{rm} + -8.537263471766453$, 此时loss是: 52.015042538154226
在第1275步, 我们获得了函数 $f(\text{rm}) = 4.994800656239819 * \text{rm} + -8.543547819351929$, 此时loss是: 52.01099612145536
在第1276步, 我们获得了函数 $f(\text{rm}) = 4.995788349969856 * \text{rm} + -8.549830655726117$, 此时loss是: 52.006951650624444
在第1277步, 我们获得了函数 $f(\text{rm}) = 4.996775806186946 * \text{rm} + -8.556111981252421$, 此时loss是: 52.002909124725804
在第1278步, 我们获得了函数 $f(\text{rm}) = 4.997763024948205 * \text{rm} + -8.562391796294158$, 此时loss是: 51.99886854282408
在第1279步, 我们获得了函数 $f(\text{rm}) = 4.9987500063107335 * \text{rm} + -8.568670101214556$, 此时loss是: 51.99482990398442
在第1280步, 我们获得了函数 $f(\text{rm}) = 4.9997367503316195 * \text{rm} + -8.574946896376758$, 此时loss是: 51.99079320727251
在第1281步, 我们获得了函数 $f(\text{rm}) = 5.000723257067937 * \text{rm} + -8.58122218214382$, 此时loss是: 51.986758451754326
在第1282步, 我们获得了函数 $f(\text{rm}) = 5.001709526576747 * \text{rm} + -8.58749595887871$, 此时loss是: 51.982725636496376
在第1283步, 我们获得了函数 $f(\text{rm}) = 5.002695558915096 * \text{rm} + -8.593768226944308$, 此时loss是: 51.97869476056565
在第1284步, 我们获得了函数 $f(\text{rm}) = 5.003681354140016 * \text{rm} + -8.600038986703405$, 此时loss是: 51.97466582302963
在第1285步, 我们获得了函数 $f(\text{rm}) = 5.004666912308528 * \text{rm} + -8.606308238518709$, 此时loss是: 51.970638822956026
在第1286步, 我们获得了函数 $f(\text{rm}) = 5.005652233477635 * \text{rm} + -8.612575982752839$, 此时loss是: 51.96661375941326

在第1287步, 我们获得了函数 $f(\text{rm}) = 5.006637317704331 * \text{rm} + -8.618842219768323$, 此时loss是: 51.96259063146998
在第1288步, 我们获得了函数 $f(\text{rm}) = 5.007622165045593 * \text{rm} + -8.62510694992761$, 此时loss是: 51.958569438195376
在第1289步, 我们获得了函数 $f(\text{rm}) = 5.008606775558385 * \text{rm} + -8.631370173593053$, 此时loss是: 51.95455017865922
在第1290步, 我们获得了函数 $f(\text{rm}) = 5.009591149299659 * \text{rm} + -8.637631891126926$, 此时loss是: 51.950532851931435
在第1291步, 我们获得了函数 $f(\text{rm}) = 5.010575286326351 * \text{rm} + -8.643892102891408$, 此时loss是: 51.94651745708266
在第1292步, 我们获得了函数 $f(\text{rm}) = 5.011559186695385 * \text{rm} + -8.650150809248599$, 此时loss是: 51.94250399318394
在第1293步, 我们获得了函数 $f(\text{rm}) = 5.012542850463669 * \text{rm} + -8.656408010560504$, 此时loss是: 51.93849245930657
在第1294步, 我们获得了函数 $f(\text{rm}) = 5.013526277688101 * \text{rm} + -8.662663707189047$, 此时loss是: 51.93448285452248
在第1295步, 我们获得了函数 $f(\text{rm}) = 5.014509468425562 * \text{rm} + -8.66891789949606$, 此时loss是: 51.93047517790407
在第1296步, 我们获得了函数 $f(\text{rm}) = 5.015492422732921 * \text{rm} + -8.675170587843294$, 此时loss是: 51.92646942852399
在第1297步, 我们获得了函数 $f(\text{rm}) = 5.0164751406670325 * \text{rm} + -8.681421772592406$, 此时loss是: 51.92246560545552
在第1298步, 我们获得了函数 $f(\text{rm}) = 5.017457622284739 * \text{rm} + -8.687671454104972$, 此时loss是: 51.918463707772325
在第1299步, 我们获得了函数 $f(\text{rm}) = 5.018439867642865 * \text{rm} + -8.693919632742478$, 此时loss是: 51.91446373454846
在第1300步, 我们获得了函数 $f(\text{rm}) = 5.019421876798227 * \text{rm} + -8.700166308866324$, 此时loss是: 51.9104656848586
在第1301步, 我们获得了函数 $f(\text{rm}) = 5.020403649807625 * \text{rm} + -8.706411482837822$, 此时loss是: 51.90646955777757
在第1302步, 我们获得了函数 $f(\text{rm}) = 5.021385186727845 * \text{rm} + -8.7126551550182$, 此时loss是: 51.90247535238097
在第1303步, 我们获得了函数 $f(\text{rm}) = 5.022366487615661 * \text{rm} + -8.718897325768596$, 此时loss是: 51.898483067744536
在第1304步, 我们获得了函数 $f(\text{rm}) = 5.02334755252783 * \text{rm} + -8.725137995450062$, 此时loss是: 51.89449270294481
在第1305步, 我们获得了函数 $f(\text{rm}) = 5.0243283815211 * \text{rm} + -8.731377164423563$, 此时loss是: 51.890504257058375
在第1306步, 我们获得了函数 $f(\text{rm}) = 5.025308974652201 * \text{rm} + -8.737614833049978$, 此时loss是: 51.886517729162584
在第1307步, 我们获得了函数 $f(\text{rm}) = 5.026289331977852 * \text{rm} + -8.7438510016901$, 此时loss是: 51.88253311833493
在第1308步, 我们获得了函数 $f(\text{rm}) = 5.027269453554758 * \text{rm} + -8.750085670704632$, 此时loss是: 51.87855042365369
在第1309步, 我们获得了函数 $f(\text{rm}) = 5.0282493394396095 * \text{rm} + -8.756318840454194$, 此时loss是: 51.87456964419741

在第1310步, 我们获得了函数 $f(\text{rm}) = 5.029228989689084 * \text{rm} + -$
8.762550511299317, 此时loss是: 51.87059077904491
在第1311步, 我们获得了函数 $f(\text{rm}) = 5.030208404359846 * \text{rm} + -$
8.768780683600447, 此时loss是: 51.86661382727585
在第1312步, 我们获得了函数 $f(\text{rm}) = 5.031187583508546 * \text{rm} + -$
8.775009357717941, 此时loss是: 51.86263878796993
在第1313步, 我们获得了函数 $f(\text{rm}) = 5.032166527191819 * \text{rm} + -$
8.781236534012072, 此时loss是: 51.85866566020758
在第1314步, 我们获得了函数 $f(\text{rm}) = 5.0331452354662884 * \text{rm} + -$
8.787462212843023, 此时loss是: 51.85469444306949
在第1315步, 我们获得了函数 $f(\text{rm}) = 5.034123708388564 * \text{rm} + -$
8.793686394570893, 此时loss是: 51.850725135636914
在第1316步, 我们获得了函数 $f(\text{rm}) = 5.03510194601524 * \text{rm} + -$
8.799909079555693, 此时loss是: 51.84675773699151
在第1317步, 我们获得了函数 $f(\text{rm}) = 5.036079948402901 * \text{rm} + -$
8.80613026815735, 此时loss是: 51.84279224621545
在第1318步, 我们获得了函数 $f(\text{rm}) = 5.037057715608114 * \text{rm} + -$
8.812349960735702, 此时loss是: 51.83882866239101
在第1319步, 我们获得了函数 $f(\text{rm}) = 5.038035247687433 * \text{rm} + -$
8.818568157650502, 此时loss是: 51.83486698460137
在第1320步, 我们获得了函数 $f(\text{rm}) = 5.0390125446974015 * \text{rm} + -$
8.824784859261413, 此时loss是: 51.83090721192982
在第1321步, 我们获得了函数 $f(\text{rm}) = 5.039989606694545 * \text{rm} + -$
8.831000065928016, 此时loss是: 51.82694934346042
在第1322步, 我们获得了函数 $f(\text{rm}) = 5.040966433735378 * \text{rm} + -$
8.837213778009804, 此时loss是: 51.82299337827726
在第1323步, 我们获得了函数 $f(\text{rm}) = 5.041943025876402 * \text{rm} + -$
8.843425995866184, 此时loss是: 51.819039315465126
在第1324步, 我们获得了函数 $f(\text{rm}) = 5.0429193831741035 * \text{rm} + -$
8.849636719856473, 此时loss是: 51.815087154109236
在第1325步, 我们获得了函数 $f(\text{rm}) = 5.043895505684955 * \text{rm} + -$
8.855845950339907, 此时loss是: 51.811136893295156
在第1326步, 我们获得了函数 $f(\text{rm}) = 5.044871393465417 * \text{rm} + -$
8.86205368767563, 此时loss是: 51.807188532109016
在第1327步, 我们获得了函数 $f(\text{rm}) = 5.045847046571936 * \text{rm} + -$
8.868259932222704, 此时loss是: 51.803242069637186
在第1328步, 我们获得了函数 $f(\text{rm}) = 5.046822465060943 * \text{rm} + -$
8.874464684340104, 此时loss是: 51.79929750496675
在第1329步, 我们获得了函数 $f(\text{rm}) = 5.047797648988859 * \text{rm} + -$
8.880667944386719, 此时loss是: 51.79535483718491
在第1330步, 我们获得了函数 $f(\text{rm}) = 5.0487725984120875 * \text{rm} + -$
8.886869712721348, 此时loss是: 51.79141406537958
在第1331步, 我们获得了函数 $f(\text{rm}) = 5.049747313387022 * \text{rm} + -$
8.893069989702708, 此时loss是: 51.7874751886391
在第1332步, 我们获得了函数 $f(\text{rm}) = 5.05072179397004 * \text{rm} + -$
8.899268775689427, 此时loss是: 51.78353820605196

在第1333步, 我们获得了函数 $f(\text{rm}) = 5.051696040217506 * \text{rm} + -8.905466071040049$, 此时loss是: 51.77960311670744
在第1334步, 我们获得了函数 $f(\text{rm}) = 5.052670052185772 * \text{rm} + -8.91166187611303$, 此时loss是: 51.7756699196951
在第1335步, 我们获得了函数 $f(\text{rm}) = 5.053643829931176 * \text{rm} + -8.91785619126674$, 此时loss是: 51.77173861410477
在第1336步, 我们获得了函数 $f(\text{rm}) = 5.054617373510041 * \text{rm} + -8.924049016859463$, 此时loss是: 51.76780919902707
在第1337步, 我们获得了函数 $f(\text{rm}) = 5.055590682978677 * \text{rm} + -8.930240353249399$, 此时loss是: 51.76388167355281
在第1338步, 我们获得了函数 $f(\text{rm}) = 5.0565637583933825 * \text{rm} + -8.936430200794659$, 此时loss是: 51.75995603677333
在第1339步, 我们获得了函数 $f(\text{rm}) = 5.05753659981044 * \text{rm} + -8.942618559853269$, 此时loss是: 51.75603228778039
在第1340步, 我们获得了函数 $f(\text{rm}) = 5.058509207286121 * \text{rm} + -8.948805430783167$, 此时loss是: 51.75211042566609
在第1341步, 我们获得了函数 $f(\text{rm}) = 5.05948158087668 * \text{rm} + -8.95499081394221$, 此时loss是: 51.748190449523214
在第1342步, 我们获得了函数 $f(\text{rm}) = 5.06045372063836 * \text{rm} + -8.961174709688162$, 此时loss是: 51.74427235844464
在第1343步, 我们获得了函数 $f(\text{rm}) = 5.061425626627392 * \text{rm} + -8.967357118378708$, 此时loss是: 51.740356151524026
在第1344步, 我们获得了函数 $f(\text{rm}) = 5.062397298899991 * \text{rm} + -8.97353804037144$, 此时loss是: 51.73644182785513
在第1345步, 我们获得了函数 $f(\text{rm}) = 5.063368737512358 * \text{rm} + -8.979717476023872$, 此时loss是: 51.73252938653249
在第1346步, 我们获得了函数 $f(\text{rm}) = 5.064339942520684 * \text{rm} + -8.985895425693423$, 此时loss是: 51.72861882665088
在第1347步, 我们获得了函数 $f(\text{rm}) = 5.0653109139811425 * \text{rm} + -8.992071889737433$, 此时loss是: 51.72471014730547
在第1348步, 我们获得了函数 $f(\text{rm}) = 5.066281651949897 * \text{rm} + -8.998246868513155$, 此时loss是: 51.720803347591996
在第1349步, 我们获得了函数 $f(\text{rm}) = 5.067252156483095 * \text{rm} + -9.004420362377752$, 此时loss是: 51.71689842660654
在第1350步, 我们获得了函数 $f(\text{rm}) = 5.06822242763687 * \text{rm} + -9.010592371688306$, 此时loss是: 51.712995383445666
在第1351步, 我们获得了函数 $f(\text{rm}) = 5.069192465467345 * \text{rm} + -9.01676289680181$, 此时loss是: 51.70909421720627
在第1352步, 我们获得了函数 $f(\text{rm}) = 5.070162270030627 * \text{rm} + -9.022931938075173$, 此时loss是: 51.705194926985904
在第1353步, 我们获得了函数 $f(\text{rm}) = 5.071131841382811 * \text{rm} + -9.029099495865216$, 此时loss是: 51.70129751188236
在第1354步, 我们获得了函数 $f(\text{rm}) = 5.072101179579976 * \text{rm} + -9.035265570528678$, 此时loss是: 51.697401970993816
在第1355步, 我们获得了函数 $f(\text{rm}) = 5.073070284678191 * \text{rm} + -9.041430162422207$, 此时loss是: 51.69350830341912

在第1356步, 我们获得了函数 $f(\text{rm}) = 5.07403915673351 * \text{rm} + -9.04759327190237$, 此时loss是: 51.689616508257394
在第1357步, 我们获得了函数 $f(\text{rm}) = 5.075007795801972 * \text{rm} + -9.053754899325646$, 此时loss是: 51.685726584608176
在第1358步, 我们获得了函数 $f(\text{rm}) = 5.075976201939605 * \text{rm} + -9.05991504504843$, 此时loss是: 51.68183853157155
在第1359步, 我们获得了函数 $f(\text{rm}) = 5.076944375202422 * \text{rm} + -9.066073709427025$, 此时loss是: 51.67795234824791
在第1360步, 我们获得了函数 $f(\text{rm}) = 5.077912315646423 * \text{rm} + -9.072230892817657$, 此时loss是: 51.67406803373805
在第1361步, 我们获得了函数 $f(\text{rm}) = 5.078880023327595 * \text{rm} + -9.078386595576463$, 此时loss是: 51.670185587143486
在第1362步, 我们获得了函数 $f(\text{rm}) = 5.07984749830191 * \text{rm} + -9.084540818059493$, 此时loss是: 51.6663050075658
在第1363步, 我们获得了函数 $f(\text{rm}) = 5.080814740625328 * \text{rm} + -9.09069356062271$, 此时loss是: 51.662426294107206
在第1364步, 我们获得了函数 $f(\text{rm}) = 5.081781750353795 * \text{rm} + -9.096844823621998$, 此时loss是: 51.65854944587041
在第1365步, 我们获得了函数 $f(\text{rm}) = 5.082748527543244 * \text{rm} + -9.102994607413148$, 此时loss是: 51.65467446195829
在第1366步, 我们获得了函数 $f(\text{rm}) = 5.083715072249594 * \text{rm} + -9.109142912351869$, 此时loss是: 51.650801341474455
在第1367步, 我们获得了函数 $f(\text{rm}) = 5.08468138452875 * \text{rm} + -9.115289738793784$, 此时loss是: 51.64693008352275
在第1368步, 我们获得了函数 $f(\text{rm}) = 5.0856474644366045 * \text{rm} + -9.121435087094431$, 此时loss是: 51.64306068720743
在第1369步, 我们获得了函数 $f(\text{rm}) = 5.086613312029038 * \text{rm} + -9.127578957609263$, 此时loss是: 51.639193151633314
在第1370步, 我们获得了函数 $f(\text{rm}) = 5.087578927361914 * \text{rm} + -9.133721350693644$, 此时loss是: 51.63532747590573
在第1371步, 我们获得了函数 $f(\text{rm}) = 5.088544310491085 * \text{rm} + -9.139862266702858$, 此时loss是: 51.631463659130176
在第1372步, 我们获得了函数 $f(\text{rm}) = 5.08950946147239 * \text{rm} + -9.146001705992099$, 此时loss是: 51.62760170041265
在第1373步, 我们获得了函数 $f(\text{rm}) = 5.090474380361655 * \text{rm} + -9.152139668916476$, 此时loss是: 51.623741598859795
在第1374步, 我们获得了函数 $f(\text{rm}) = 5.091439067214689 * \text{rm} + -9.158276155831015$, 此时loss是: 51.61988335357836
在第1375步, 我们获得了函数 $f(\text{rm}) = 5.0924035220872925 * \text{rm} + -9.164411167090655$, 此时loss是: 51.616026963675836
在第1376步, 我们获得了函数 $f(\text{rm}) = 5.093367745035249 * \text{rm} + -9.17054470305025$, 此时loss是: 51.612172428259896
在第1377步, 我们获得了函数 $f(\text{rm}) = 5.094331736114331 * \text{rm} + -9.17667676406457$, 此时loss是: 51.60831974643878
在第1378步, 我们获得了函数 $f(\text{rm}) = 5.095295495380296 * \text{rm} + -9.182807350488298$, 此时loss是: 51.604468917321135

在第1379步, 我们获得了函数 $f(\text{rm}) = 5.09625902288889 * \text{rm} + -9.18893646267603$, 此时loss是: 51.60061994001588
在第1380步, 我们获得了函数 $f(\text{rm}) = 5.097222318695842 * \text{rm} + -9.195064100982282$, 此时loss是: 51.59677281363275
在第1381步, 我们获得了函数 $f(\text{rm}) = 5.098185382856871 * \text{rm} + -9.20119026576148$, 此时loss是: 51.59292753728147
在第1382步, 我们获得了函数 $f(\text{rm}) = 5.099148215427681 * \text{rm} + -9.207314957367968$, 此时loss是: 51.589084110072456
在第1383步, 我们获得了函数 $f(\text{rm}) = 5.100110816463963 * \text{rm} + -9.213438176156$, 此时loss是: 51.585242531116535
在第1384步, 我们获得了函数 $f(\text{rm}) = 5.101073186021396 * \text{rm} + -9.21955992247975$, 此时loss是: 51.58140279952481
在第1385步, 我们获得了函数 $f(\text{rm}) = 5.102035324155642 * \text{rm} + -9.225680196693304$, 此时loss是: 51.57756491440885
在第1386步, 我们获得了函数 $f(\text{rm}) = 5.1029972309223535 * \text{rm} + -9.231798999150666$, 此时loss是: 51.57372887488087
在第1387步, 我们获得了函数 $f(\text{rm}) = 5.1039589063771675 * \text{rm} + -9.23791633020575$, 此时loss是: 51.56989468005327
在第1388步, 我们获得了函数 $f(\text{rm}) = 5.104920350575708 * \text{rm} + -9.244032190212387$, 此时loss是: 51.56606232903896
在第1389步, 我们获得了函数 $f(\text{rm}) = 5.105881563573585 * \text{rm} + -9.250146579524326$, 此时loss是: 51.56223182095132
在第1390步, 我们获得了函数 $f(\text{rm}) = 5.106842545426398 * \text{rm} + -9.256259498495227$, 此时loss是: 51.55840315490405
在第1391步, 我们获得了函数 $f(\text{rm}) = 5.107803296189729 * \text{rm} + -9.262370947478665$, 此时loss是: 51.55457633001138
在第1392步, 我们获得了函数 $f(\text{rm}) = 5.108763815919149 * \text{rm} + -9.268480926828133$, 此时loss是: 51.5507513453878
在第1393步, 我们获得了函数 $f(\text{rm}) = 5.1097241046702155 * \text{rm} + -9.274589436897037$, 此时loss是: 51.5469282001486
在第1394步, 我们获得了函数 $f(\text{rm}) = 5.110684162498473 * \text{rm} + -9.280696478038697$, 此时loss是: 51.54310689340903
在第1395步, 我们获得了函数 $f(\text{rm}) = 5.111643989459451 * \text{rm} + -9.286802050606351$, 此时loss是: 51.53928742428501
在第1396步, 我们获得了函数 $f(\text{rm}) = 5.112603585608667 * \text{rm} + -9.292906154953148$, 此时loss是: 51.53546979189291
在第1397步, 我们获得了函数 $f(\text{rm}) = 5.113562951001625 * \text{rm} + -9.299008791432156$, 此时loss是: 51.53165399534959
在第1398步, 我们获得了函数 $f(\text{rm}) = 5.114522085693816 * \text{rm} + -9.305109960396356$, 此时loss是: 51.52784003377195
在第1399步, 我们获得了函数 $f(\text{rm}) = 5.115480989740715 * \text{rm} + -9.311209662198646$, 此时loss是: 51.52402790627769
在第1400步, 我们获得了函数 $f(\text{rm}) = 5.116439663197789 * \text{rm} + -9.317307897191835$, 此时loss是: 51.52021761198487
在第1401步, 我们获得了函数 $f(\text{rm}) = 5.117398106120485 * \text{rm} + -9.323404665728653$, 此时loss是: 51.51640915001195

在第1402步, 我们获得了函数 $f(\text{rm}) = 5.118356318564243 * \text{rm} + -9.32949996816174$, 此时loss是: 51.51260251947767
在第1403步, 我们获得了函数 $f(\text{rm}) = 5.1193143005844854 * \text{rm} + -9.335593804843652$, 此时loss是: 51.50879771950142
在第1404步, 我们获得了函数 $f(\text{rm}) = 5.120272052236623 * \text{rm} + -9.341686176126865$, 此时loss是: 51.504994749202915
在第1405步, 我们获得了函数 $f(\text{rm}) = 5.121229573576053 * \text{rm} + -9.347777082363766$, 此时loss是: 51.50119360770217
在第1406步, 我们获得了函数 $f(\text{rm}) = 5.1221868646581585 * \text{rm} + -9.353866523906657$, 此时loss是: 51.49739429411985
在第1407步, 我们获得了函数 $f(\text{rm}) = 5.1231439255383115 * \text{rm} + -9.359954501107755$, 此时loss是: 51.49359680757689
在第1408步, 我们获得了函数 $f(\text{rm}) = 5.124100756271868 * \text{rm} + -9.366041014319196$, 此时loss是: 51.4898011471948
在第1409步, 我们获得了函数 $f(\text{rm}) = 5.125057356914172 * \text{rm} + -9.372126063893027$, 此时loss是: 51.48600731209513
在第1410步, 我们获得了函数 $f(\text{rm}) = 5.126013727520554 * \text{rm} + -9.378209650181214$, 此时loss是: 51.48221530140051
在第1411步, 我们获得了函数 $f(\text{rm}) = 5.126969868146332 * \text{rm} + -9.384291773535635$, 此时loss是: 51.478425114233254
在第1412步, 我们获得了函数 $f(\text{rm}) = 5.127925778846809 * \text{rm} + -9.390372434308086$, 此时loss是: 51.47463674971665
在第1413步, 我们获得了函数 $f(\text{rm}) = 5.128881459677276 * \text{rm} + -9.396451632850276$, 此时loss是: 51.47085020697407
在第1414步, 我们获得了函数 $f(\text{rm}) = 5.12983691069301 * \text{rm} + -9.402529369513832$, 此时loss是: 51.46706548512962
在第1415步, 我们获得了函数 $f(\text{rm}) = 5.130792131949277 * \text{rm} + -9.408605644650295$, 此时loss是: 51.46328258330755
在第1416步, 我们获得了函数 $f(\text{rm}) = 5.1317471235013254 * \text{rm} + -9.414680458611123$, 此时loss是: 51.45950150063267
在第1417步, 我们获得了函数 $f(\text{rm}) = 5.132701885404393 * \text{rm} + -9.420753811747687$, 此时loss是: 51.45572223623014
在第1418步, 我们获得了函数 $f(\text{rm}) = 5.133656417713706 * \text{rm} + -9.426825704411275$, 此时loss是: 51.451944789225635
在第1419步, 我们获得了函数 $f(\text{rm}) = 5.134610720484473 * \text{rm} + -9.43289613695309$, 此时loss是: 51.44816915874509
在第1420步, 我们获得了函数 $f(\text{rm}) = 5.135564793771893 * \text{rm} + -9.43896510972425$, 此时loss是: 51.44439534391508
在第1421步, 我们获得了函数 $f(\text{rm}) = 5.136518637631149 * \text{rm} + -9.445032623075791$, 此时loss是: 51.44062334386241
在第1422步, 我们获得了函数 $f(\text{rm}) = 5.137472252117415 * \text{rm} + -9.451098677358663$, 此时loss是: 51.436853157714445
在第1423步, 我们获得了函数 $f(\text{rm}) = 5.138425637285845 * \text{rm} + -9.457163272923731$, 此时loss是: 51.43308478459881
在第1424步, 我们获得了函数 $f(\text{rm}) = 5.139378793191587 * \text{rm} + -9.463226410121775$, 此时loss是: 51.429318223643676

在第1425步, 我们获得了函数 $f(\text{rm}) = 5.140331719889771 * \text{rm} + -9.469288089303493$, 此时loss是: 51.4255534739776

在第1426步, 我们获得了函数 $f(\text{rm}) = 5.1412844174355135 * \text{rm} + -9.475348310819497$, 此时loss是: 51.42179053472963

在第1427步, 我们获得了函数 $f(\text{rm}) = 5.142236885883921 * \text{rm} + -9.481407075020314$, 此时loss是: 51.418029405029046

在第1428步, 我们获得了函数 $f(\text{rm}) = 5.1431891252900845 * \text{rm} + -9.487464382256391$, 此时loss是: 51.41427008400576

在第1429步, 我们获得了函数 $f(\text{rm}) = 5.144141135709083 * \text{rm} + -9.493520232878085$, 此时loss是: 51.410512570789905

在第1430步, 我们获得了函数 $f(\text{rm}) = 5.145092917195981 * \text{rm} + -9.499574627235672$, 此时loss是: 51.40675686451221

在第1431步, 我们获得了函数 $f(\text{rm}) = 5.146044469805831 * \text{rm} + -9.505627565679344$, 此时loss是: 51.40300296430364

在第1432步, 我们获得了函数 $f(\text{rm}) = 5.14699579359367 * \text{rm} + -9.511679048559207$, 此时loss是: 51.399250869295905

在第1433步, 我们获得了函数 $f(\text{rm}) = 5.147946888614525 * \text{rm} + -9.517729076225285$, 此时loss是: 51.39550057862065

在第1434步, 我们获得了函数 $f(\text{rm}) = 5.148897754923408 * \text{rm} + -9.523777649027515$, 此时loss是: 51.391752091410275

在第1435步, 我们获得了函数 $f(\text{rm}) = 5.149848392575317 * \text{rm} + -9.529824767315752$, 此时loss是: 51.388005406797575

在第1436步, 我们获得了函数 $f(\text{rm}) = 5.150798801625237 * \text{rm} + -9.535870431439767$, 此时loss是: 51.38426052391574

在第1437步, 我们获得了函数 $f(\text{rm}) = 5.151748982128143 * \text{rm} + -9.541914641749244$, 此时loss是: 51.3805174418983

在第1438步, 我们获得了函数 $f(\text{rm}) = 5.152698934138992 * \text{rm} + -9.547957398593788$, 此时loss是: 51.37677615987913

在第1439步, 我们获得了函数 $f(\text{rm}) = 5.153648657712731 * \text{rm} + -9.553998702322914$, 此时loss是: 51.37303667699272

在第1440步, 我们获得了函数 $f(\text{rm}) = 5.154598152904292 * \text{rm} + -9.560038553286057$, 此时loss是: 51.369298992373906

在第1441步, 我们获得了函数 $f(\text{rm}) = 5.155547419768595 * \text{rm} + -9.566076951832569$, 此时loss是: 51.36556310515795

在第1442步, 我们获得了函数 $f(\text{rm}) = 5.1564964583605475 * \text{rm} + -9.572113898311713$, 此时loss是: 51.36182901448045

在第1443步, 我们获得了函数 $f(\text{rm}) = 5.1574452687350405 * \text{rm} + -9.578149393072673$, 此时loss是: 51.358096719477516

在第1444步, 我们获得了函数 $f(\text{rm}) = 5.158393850946955 * \text{rm} + -9.584183436464546$, 此时loss是: 51.35436621928558

在第1445步, 我们获得了函数 $f(\text{rm}) = 5.159342205051158 * \text{rm} + -9.590216028836345$, 此时loss是: 51.350637513041605

在第1446步, 我们获得了函数 $f(\text{rm}) = 5.160290331102503 * \text{rm} + -9.596247170537001$, 此时loss是: 51.34691059988281

在第1447步, 我们获得了函数 $f(\text{rm}) = 5.161238229155829 * \text{rm} + -9.602276861915362$, 此时loss是: 51.34318547894703

在第1448步, 我们获得了函数 $f(\text{rm}) = 5.162185899265967 * \text{rm} + -9.608305103320188$, 此时loss是: 51.33946214937239
在第1449步, 我们获得了函数 $f(\text{rm}) = 5.163133341487727 * \text{rm} + -9.614331895100158$, 此时loss是: 51.335740610297364
在第1450步, 我们获得了函数 $f(\text{rm}) = 5.164080555875912 * \text{rm} + -9.620357237603866$, 此时loss是: 51.332020860861014
在第1451步, 我们获得了函数 $f(\text{rm}) = 5.165027542485308 * \text{rm} + -9.626381131179825$, 此时loss是: 51.3283029002027
在第1452步, 我们获得了函数 $f(\text{rm}) = 5.165974301370691 * \text{rm} + -9.63240357617646$, 此时loss是: 51.324586727462176
在第1453步, 我们获得了函数 $f(\text{rm}) = 5.166920832586822 * \text{rm} + -9.638424572942114$, 此时loss是: 51.32087234177971
在第1454步, 我们获得了函数 $f(\text{rm}) = 5.167867136188448 * \text{rm} + -9.644444121825046$, 此时loss是: 51.31715974229601
在第1455步, 我们获得了函数 $f(\text{rm}) = 5.168813212230306 * \text{rm} + -9.650462223173433$, 此时loss是: 51.31344892815195
在第1456步, 我们获得了函数 $f(\text{rm}) = 5.169759060767117 * \text{rm} + -9.656478877335367$, 此时loss是: 51.309739898489006
在第1457步, 我们获得了函数 $f(\text{rm}) = 5.1707046818535884 * \text{rm} + -9.662494084658855$, 此时loss是: 51.30603265244916
在第1458步, 我们获得了函数 $f(\text{rm}) = 5.171650075544417 * \text{rm} + -9.668507845491822$, 此时loss是: 51.30232718917466
在第1459步, 我们获得了函数 $f(\text{rm}) = 5.172595241894284 * \text{rm} + -9.67452016018211$, 此时loss是: 51.298623507808145
在第1460步, 我们获得了函数 $f(\text{rm}) = 5.173540180957861 * \text{rm} + -9.680531029077475$, 此时loss是: 51.29492160749277
在第1461步, 我们获得了函数 $f(\text{rm}) = 5.174484892789802 * \text{rm} + -9.686540452525591$, 此时loss是: 51.29122148737191
在第1462步, 我们获得了函数 $f(\text{rm}) = 5.1754293774447495 * \text{rm} + -9.692548430874048$, 此时loss是: 51.2875231465897
在第1463步, 我们获得了函数 $f(\text{rm}) = 5.176373634977335 * \text{rm} + -9.69855496447035$, 此时loss是: 51.28382658429033
在第1464步, 我们获得了函数 $f(\text{rm}) = 5.177317665442174 * \text{rm} + -9.704560053661924$, 此时loss是: 51.280131799618694
在第1465步, 我们获得了函数 $f(\text{rm}) = 5.1782614688938695 * \text{rm} + -9.710563698796106$, 此时loss是: 51.276438791719784
在第1466步, 我们获得了函数 $f(\text{rm}) = 5.179205045387014 * \text{rm} + -9.716565900220154$, 此时loss是: 51.27274755973929
在第1467步, 我们获得了函数 $f(\text{rm}) = 5.180148394976182 * \text{rm} + -9.722566658281238$, 此时loss是: 51.26905810282319
在第1468步, 我们获得了函数 $f(\text{rm}) = 5.18109151771594 * \text{rm} + -9.728565973326448$, 此时loss是: 51.26537042011775
在第1469步, 我们获得了函数 $f(\text{rm}) = 5.182034413660837 * \text{rm} + -9.734563845702791$, 此时loss是: 51.26168451077
在第1470步, 我们获得了函数 $f(\text{rm}) = 5.182977082865413 * \text{rm} + -9.740560275757186$, 此时loss是: 51.25800037392691

在第1471步, 我们获得了函数 $f(\text{rm}) = 5.183919525384191 * \text{rm} + -9.746555263836473$, 此时loss是: 51.25431800873639
在第1472步, 我们获得了函数 $f(\text{rm}) = 5.184861741271684 * \text{rm} + -9.752548810287406$, 此时loss是: 51.25063741434618
在第1473步, 我们获得了函数 $f(\text{rm}) = 5.185803730582388 * \text{rm} + -9.758540915456658$, 此时loss是: 51.246958589904914
在第1474步, 我们获得了函数 $f(\text{rm}) = 5.1867454933707915 * \text{rm} + -9.764531579690816$, 此时loss是: 51.243281534561376
在第1475步, 我们获得了函数 $f(\text{rm}) = 5.187687029691365 * \text{rm} + -9.770520803336385$, 此时loss是: 51.23960624746488
在第1476步, 我们获得了函数 $f(\text{rm}) = 5.188628339598569 * \text{rm} + -9.776508586739787$, 此时loss是: 51.23593272776505
在第1477步, 我们获得了函数 $f(\text{rm}) = 5.189569423146849 * \text{rm} + -9.78249493024736$, 此时loss是: 51.23226097461194
在第1478步, 我们获得了函数 $f(\text{rm}) = 5.190510280390637 * \text{rm} + -9.78847983420536$, 此时loss是: 51.22859098715616
在第1479步, 我们获得了函数 $f(\text{rm}) = 5.191450911384354 * \text{rm} + -9.794463298959958$, 此时loss是: 51.2249227645485
在第1480步, 我们获得了函数 $f(\text{rm}) = 5.192391316182407 * \text{rm} + -9.800445324857241$, 此时loss是: 51.221256305940294
在第1481步, 我们获得了函数 $f(\text{rm}) = 5.19333149483919 * \text{rm} + -9.806425912243217$, 此时loss是: 51.21759161048331
在第1482步, 我们获得了函数 $f(\text{rm}) = 5.194271447409082 * \text{rm} + -9.812405061463807$, 此时loss是: 51.21392867732963
在第1483步, 我们获得了函数 $f(\text{rm}) = 5.195211173946452 * \text{rm} + -9.81838277286485$, 此时loss是: 51.21026750563175
在第1484步, 我们获得了函数 $f(\text{rm}) = 5.196150674505656 * \text{rm} + -9.8243590467921$, 此时loss是: 51.206608094542645
在第1485步, 我们获得了函数 $f(\text{rm}) = 5.1970899491410325 * \text{rm} + -9.830333883591232$, 此时loss是: 51.20295044321573
在第1486步, 我们获得了函数 $f(\text{rm}) = 5.198028997906912 * \text{rm} + -9.836307283607834$, 此时loss是: 51.19929455080463
在第1487步, 我们获得了函数 $f(\text{rm}) = 5.198967820857609 * \text{rm} + -9.842279247187413$, 此时loss是: 51.19564041646355
在第1488步, 我们获得了函数 $f(\text{rm}) = 5.1999064180474255 * \text{rm} + -9.848249774675393$, 此时loss是: 51.191988039347144
在第1489步, 我们获得了函数 $f(\text{rm}) = 5.200844789530652 * \text{rm} + -9.854218866417114$, 此时loss是: 51.18833741861027
在第1490步, 我们获得了函数 $f(\text{rm}) = 5.201782935361564 * \text{rm} + -9.86018652275783$, 此时loss是: 51.184688553408385
在第1491步, 我们获得了函数 $f(\text{rm}) = 5.202720855594424 * \text{rm} + -9.86615274404272$, 此时loss是: 51.18104144289722
在第1492步, 我们获得了函数 $f(\text{rm}) = 5.203658550283483 * \text{rm} + -9.872117530616872$, 此时loss是: 51.17739608623299
在第1493步, 我们获得了函数 $f(\text{rm}) = 5.204596019482977 * \text{rm} + -9.878080882825294$, 此时loss是: 51.17375248257233

在第1494步, 我们获得了函数 $f(\text{rm}) = 5.205533263247131 * \text{rm} + -9.884042801012914$, 此时loss是: 51.17011063107215
在第1495步, 我们获得了函数 $f(\text{rm}) = 5.206470281630155 * \text{rm} + -9.890003285524571$, 此时loss是: 51.16647053089001
在第1496步, 我们获得了函数 $f(\text{rm}) = 5.207407074686247 * \text{rm} + -9.895962336705026$, 此时loss是: 51.162832181183575
在第1497步, 我们获得了函数 $f(\text{rm}) = 5.208343642469593 * \text{rm} + -9.901919954898954$, 此时loss是: 51.159195581111106
在第1498步, 我们获得了函数 $f(\text{rm}) = 5.209279985034365 * \text{rm} + -9.90787614045095$, 此时loss是: 51.15556072983129
在第1499步, 我们获得了函数 $f(\text{rm}) = 5.21021610243472 * \text{rm} + -9.913830893705525$, 此时loss是: 51.15192762650314
在第1500步, 我们获得了函数 $f(\text{rm}) = 5.211151994724805 * \text{rm} + -9.919784215007105$, 此时loss是: 51.14829627028598
在第1501步, 我们获得了函数 $f(\text{rm}) = 5.212087661958752 * \text{rm} + -9.925736104700036$, 此时loss是: 51.144666660339794
在第1502步, 我们获得了函数 $f(\text{rm}) = 5.2130231041906825 * \text{rm} + -9.93168656312858$, 此时loss是: 51.14103879582474
在第1503步, 我们获得了函数 $f(\text{rm}) = 5.213958321474701 * \text{rm} + -9.937635590636916$, 此时loss是: 51.13741267590147
在第1504步, 我们获得了函数 $f(\text{rm}) = 5.214893313864903 * \text{rm} + -9.943583187569141$, 此时loss是: 51.13378829973103
在第1505步, 我们获得了函数 $f(\text{rm}) = 5.215828081415368 * \text{rm} + -9.94952935426927$, 此时loss是: 51.13016566647493
在第1506步, 我们获得了函数 $f(\text{rm}) = 5.216762624180165 * \text{rm} + -9.955474091081232$, 此时loss是: 51.12654477529494
在第1507步, 我们获得了函数 $f(\text{rm}) = 5.217696942213347 * \text{rm} + -9.961417398348877$, 此时loss是: 51.122925625353346
在第1508步, 我们获得了函数 $f(\text{rm}) = 5.218631035568957 * \text{rm} + -9.96735927641597$, 此时loss是: 51.11930821581288
在第1509步, 我们获得了函数 $f(\text{rm}) = 5.2195649043010235 * \text{rm} + -9.973299725626195$, 此时loss是: 51.115692545836495
在第1510步, 我们获得了函数 $f(\text{rm}) = 5.220498548463562 * \text{rm} + -9.979238746323151$, 此时loss是: 51.11207861458776
在第1511步, 我们获得了函数 $f(\text{rm}) = 5.221431968110576 * \text{rm} + -9.985176338850358$, 此时loss是: 51.108466421230474
在第1512步, 我们获得了函数 $f(\text{rm}) = 5.222365163296054 * \text{rm} + -9.99111250355125$, 此时loss是: 51.10485596492899
在第1513步, 我们获得了函数 $f(\text{rm}) = 5.223298134073974 * \text{rm} + -9.997047240769179$, 此时loss是: 51.10124724484788
在第1514步, 我们获得了函数 $f(\text{rm}) = 5.224230880498298 * \text{rm} + -10.002980550847415$, 此时loss是: 51.09764026015223
在第1515步, 我们获得了函数 $f(\text{rm}) = 5.225163402622979 * \text{rm} + -10.008912434129146$, 此时loss是: 51.09403501000761
在第1516步, 我们获得了函数 $f(\text{rm}) = 5.226095700501953 * \text{rm} + -10.014842890957476$, 此时loss是: 51.090431493579786

在第1517步, 我们获得了函数 $f(\text{rm}) = 5.227027774189146 * \text{rm} + -10.020771921675431$, 此时loss是: 51.08682971003508
在第1518步, 我们获得了函数 $f(\text{rm}) = 5.227959623738471 * \text{rm} + -10.026699526625947$, 此时loss是: 51.08322965854024
在第1519步, 我们获得了函数 $f(\text{rm}) = 5.228891249203824 * \text{rm} + -10.032625706151883$, 此时loss是: 51.079631338262224
在第1520步, 我们获得了函数 $f(\text{rm}) = 5.229822650639093 * \text{rm} + -10.038550460596015$, 此时loss是: 51.07603474836861
在第1521步, 我们获得了函数 $f(\text{rm}) = 5.230753828098151 * \text{rm} + -10.044473790301035$, 此时loss是: 51.072439888027205
在第1522步, 我们获得了函数 $f(\text{rm}) = 5.231684781634857 * \text{rm} + -10.050395695609552$, 此时loss是: 51.0688467564064
在第1523步, 我们获得了函数 $f(\text{rm}) = 5.2326155113030595 * \text{rm} + -10.056316176864096$, 此时loss是: 51.06525535267468
在第1524步, 我们获得了函数 $f(\text{rm}) = 5.233546017156591 * \text{rm} + -10.06223523440711$, 此时loss是: 51.06166567600135
在第1525步, 我们获得了函数 $f(\text{rm}) = 5.234476299249274 * \text{rm} + -10.06815286858096$, 此时loss是: 51.05807772555577
在第1526步, 我们获得了函数 $f(\text{rm}) = 5.235406357634917 * \text{rm} + -10.074069079727924$, 此时loss是: 51.05449150050779
在第1527步, 我们获得了函数 $f(\text{rm}) = 5.236336192367314 * \text{rm} + -10.079983868190203$, 此时loss是: 51.050907000027884
在第1528步, 我们获得了函数 $f(\text{rm}) = 5.237265803500248 * \text{rm} + -10.085897234309911$, 此时loss是: 51.04732422328641
在第1529步, 我们获得了函数 $f(\text{rm}) = 5.23819519108749 * \text{rm} + -10.091809178429084$, 此时loss是: 51.043743169454686
在第1530步, 我们获得了函数 $f(\text{rm}) = 5.239124355182794 * \text{rm} + -10.097719700889673$, 此时loss是: 51.04016383770413
在第1531步, 我们获得了函数 $f(\text{rm}) = 5.240053295839905 * \text{rm} + -10.103628802033546$, 此时loss是: 51.036586227206634
在第1532步, 我们获得了函数 $f(\text{rm}) = 5.2409820131125535 * \text{rm} + -10.109536482202492$, 此时loss是: 51.033010337134414
在第1533步, 我们获得了函数 $f(\text{rm}) = 5.241910507054458 * \text{rm} + -10.115442741738214$, 此时loss是: 51.02943616666017
在第1534步, 我们获得了函数 $f(\text{rm}) = 5.242838777719322 * \text{rm} + -10.121347580982336$, 此时loss是: 51.025863714957
在第1535步, 我们获得了函数 $f(\text{rm}) = 5.243766825160837 * \text{rm} + -10.1272510002764$, 此时loss是: 51.022292981198284
在第1536步, 我们获得了函数 $f(\text{rm}) = 5.2446946494326845 * \text{rm} + -10.133152999961863$, 此时loss是: 51.01872396455801
在第1537步, 我们获得了函数 $f(\text{rm}) = 5.245622250588529 * \text{rm} + -10.139053580380104$, 此时loss是: 51.01515666421031
在第1538步, 我们获得了函数 $f(\text{rm}) = 5.246549628682023 * \text{rm} + -10.144952741872414$, 此时loss是: 51.011591079329925
在第1539步, 我们获得了函数 $f(\text{rm}) = 5.247476783766808 * \text{rm} + -10.150850484780007$, 此时loss是: 51.00802720909188

在第1540步, 我们获得了函数 $f(\text{rm}) = 5.248403715896512 * \text{rm} + -10.156746809444012$, 此时loss是: 51.00446505267167
在第1541步, 我们获得了函数 $f(\text{rm}) = 5.249330425124747 * \text{rm} + -10.16264171620548$, 此时loss是: 51.000904609245126
在第1542步, 我们获得了函数 $f(\text{rm}) = 5.250256911505117 * \text{rm} + -10.168535205405375$, 此时loss是: 50.9973458779884
在第1543步, 我们获得了函数 $f(\text{rm}) = 5.25118317509121 * \text{rm} + -10.174427277384583$, 此时loss是: 50.993788858078254
在第1544步, 我们获得了函数 $f(\text{rm}) = 5.252109215936602 * \text{rm} + -10.180317932483904$, 此时loss是: 50.99023354869166
在第1545步, 我们获得了函数 $f(\text{rm}) = 5.2530350340948555 * \text{rm} + -10.18620717104406$, 此时loss是: 50.98667994900605
在第1546步, 我们获得了函数 $f(\text{rm}) = 5.253960629619521 * \text{rm} + -10.19209499340569$, 此时loss是: 50.98312805819931
在第1547步, 我们获得了函数 $f(\text{rm}) = 5.2548860025641355 * \text{rm} + -10.197981399909347$, 此时loss是: 50.97957787544955
在第1548步, 我们获得了函数 $f(\text{rm}) = 5.255811152982223 * \text{rm} + -10.203866390895508$, 此时loss是: 50.976029399935456
在第1549步, 我们获得了函数 $f(\text{rm}) = 5.256736080927296 * \text{rm} + -10.209749966704566$, 此时loss是: 50.97248263083614
在第1550步, 我们获得了函数 $f(\text{rm}) = 5.257660786452852 * \text{rm} + -10.21563212767683$, 此时loss是: 50.96893756733083
在第1551步, 我们获得了函数 $f(\text{rm}) = 5.258585269612378 * \text{rm} + -10.221512874152532$, 此时loss是: 50.965394208599434
在第1552步, 我们获得了函数 $f(\text{rm}) = 5.259509530459344 * \text{rm} + -10.227392206471817$, 此时loss是: 50.961852553822105
在第1553步, 我们获得了函数 $f(\text{rm}) = 5.260433569047214 * \text{rm} + -10.233270124974752$, 此时loss是: 50.95831260217949
在第1554步, 我们获得了函数 $f(\text{rm}) = 5.261357385429432 * \text{rm} + -10.239146630001319$, 此时loss是: 50.954774352852525
在第1555步, 我们获得了函数 $f(\text{rm}) = 5.262280979659434 * \text{rm} + -10.245021721891419$, 此时loss是: 50.95123780502262
在第1556步, 我们获得了函数 $f(\text{rm}) = 5.263204351790641 * \text{rm} + -10.250895400984875$, 此时loss是: 50.94770295787148
在第1557步, 我们获得了函数 $f(\text{rm}) = 5.2641275018764615 * \text{rm} + -10.256767667621425$, 此时loss是: 50.944169810581435
在第1558步, 我们获得了函数 $f(\text{rm}) = 5.265050429970291 * \text{rm} + -10.262638522140723$, 此时loss是: 50.94063836233484
在第1559步, 我们获得了函数 $f(\text{rm}) = 5.265973136125512 * \text{rm} + -10.268507964882346$, 此时loss是: 50.93710861231474
在第1560步, 我们获得了函数 $f(\text{rm}) = 5.266895620395497 * \text{rm} + -10.274375996185787$, 此时loss是: 50.93358055970446
在第1561步, 我们获得了函数 $f(\text{rm}) = 5.2678178828336 * \text{rm} + -10.280242616390458$, 此时loss是: 50.930054203687824
在第1562步, 我们获得了函数 $f(\text{rm}) = 5.268739923493167 * \text{rm} + -10.286107825835689$, 此时loss是: 50.926529543448886

在第1563步, 我们获得了函数 $f(\text{rm}) = 5.26966174242753 * \text{rm} + -$
10.291971624860729, 此时loss是: 50.92300657817215
在第1564步, 我们获得了函数 $f(\text{rm}) = 5.270583339690007 * \text{rm} + -$
10.297834013804744, 此时loss是: 50.91948530704255
在第1565步, 我们获得了函数 $f(\text{rm}) = 5.271504715333905 * \text{rm} + -$
10.30369499300682, 此时loss是: 50.91596572924543
在第1566步, 我们获得了函数 $f(\text{rm}) = 5.2724258694125155 * \text{rm} + -$
10.30955456280596, 此时loss是: 50.91244784396642
在第1567步, 我们获得了函数 $f(\text{rm}) = 5.27334680197912 * \text{rm} + -$
10.315412723541087, 此时loss是: 50.90893165039171
在第1568步, 我们获得了函数 $f(\text{rm}) = 5.274267513086985 * \text{rm} + -$
10.321269475551041, 此时loss是: 50.905417147707716
在第1569步, 我们获得了函数 $f(\text{rm}) = 5.275188002789366 * \text{rm} + -$
10.327124819174584, 此时loss是: 50.90190433510136
在第1570步, 我们获得了函数 $f(\text{rm}) = 5.276108271139505 * \text{rm} + -$
10.33297875475039, 此时loss是: 50.898393211759796
在第1571步, 我们获得了函数 $f(\text{rm}) = 5.277028318190631 * \text{rm} + -$
10.33883128261706, 此时loss是: 50.894883776870735
在第1572步, 我们获得了函数 $f(\text{rm}) = 5.2779481439959595 * \text{rm} + -$
10.344682403113104, 此时loss是: 50.89137602962229
在第1573步, 我们获得了函数 $f(\text{rm}) = 5.278867748608695 * \text{rm} + -$
10.350532116576959, 此时loss是: 50.88786996920287
在第1574步, 我们获得了函数 $f(\text{rm}) = 5.279787132082028 * \text{rm} + -$
10.356380423346975, 此时loss是: 50.88436559480123
在第1575步, 我们获得了函数 $f(\text{rm}) = 5.280706294469135 * \text{rm} + -$
10.362227323761426, 此时loss是: 50.88086290560666
在第1576步, 我们获得了函数 $f(\text{rm}) = 5.281625235823182 * \text{rm} + -$
10.368072818158497, 此时loss是: 50.87736190080865
在第1577步, 我们获得了函数 $f(\text{rm}) = 5.282543956197323 * \text{rm} + -$
10.3739169068763, 此时loss是: 50.87386257959733
在第1578步, 我们获得了函数 $f(\text{rm}) = 5.283462455644695 * \text{rm} + -$
10.37975959025286, 此时loss是: 50.87036494116307
在第1579步, 我们获得了函数 $f(\text{rm}) = 5.284380734218426 * \text{rm} + -$
10.385600868626124, 此时loss是: 50.86686898469665
在第1580步, 我们获得了函数 $f(\text{rm}) = 5.28529879197163 * \text{rm} + -$
10.391440742333955, 此时loss是: 50.86337470938917
在第1581步, 我们获得了函数 $f(\text{rm}) = 5.2862166289574075 * \text{rm} + -$
10.397279211714137, 此时loss是: 50.85988211443216
在第1582步, 我们获得了函数 $f(\text{rm}) = 5.287134245228848 * \text{rm} + -$
10.403116277104372, 此时loss是: 50.856391199017644
在第1583步, 我们获得了函数 $f(\text{rm}) = 5.288051640839027 * \text{rm} + -$
10.408951938842279, 此时loss是: 50.852901962337974
在第1584步, 我们获得了函数 $f(\text{rm}) = 5.288968815841006 * \text{rm} + -$
10.414786197265398, 此时loss是: 50.84941440358572
在第1585步, 我们获得了函数 $f(\text{rm}) = 5.289885770287837 * \text{rm} + -$
10.420619052711189, 此时loss是: 50.845928521954114

在第1586步, 我们获得了函数 $f(\text{rm}) = 5.290802504232556 * \text{rm} + -$
10.426450505517026, 此时loss是: 50.84244431663666
在第1587步, 我们获得了函数 $f(\text{rm}) = 5.291719017728188 * \text{rm} + -$
10.43228055602021, 此时loss是: 50.838961786827184
在第1588步, 我们获得了函数 $f(\text{rm}) = 5.2926353108277455 * \text{rm} + -$
10.438109204557952, 此时loss是: 50.83548093172003
在第1589步, 我们获得了函数 $f(\text{rm}) = 5.293551383584227 * \text{rm} + -$
10.443936451467385, 此时loss是: 50.83200175050973
在第1590步, 我们获得了函数 $f(\text{rm}) = 5.29446723605062 * \text{rm} + -$
10.449762297085565, 此时loss是: 50.82852424239141
在第1591步, 我们获得了函数 $f(\text{rm}) = 5.295382868279898 * \text{rm} + -$
10.45558674174946, 此时loss是: 50.82504840656052
在第1592步, 我们获得了函数 $f(\text{rm}) = 5.296298280325019 * \text{rm} + -$
10.461409785795965, 此时loss是: 50.82157424221283
在第1593步, 我们获得了函数 $f(\text{rm}) = 5.297213472238934 * \text{rm} + -$
10.467231429561886, 此时loss是: 50.818101748544656
在第1594步, 我们获得了函数 $f(\text{rm}) = 5.298128444074579 * \text{rm} + -$
10.473051673383953, 此时loss是: 50.814630924752414
在第1595步, 我们获得了函数 $f(\text{rm}) = 5.299043195884876 * \text{rm} + -$
10.478870517598814, 此时loss是: 50.8111617700332
在第1596步, 我们获得了函数 $f(\text{rm}) = 5.299957727722733 * \text{rm} + -$
10.484687962543035, 此时loss是: 50.807694283584304
在第1597步, 我们获得了函数 $f(\text{rm}) = 5.300872039641049 * \text{rm} + -$
10.490504008553103, 此时loss是: 50.8042284646036
在第1598步, 我们获得了函数 $f(\text{rm}) = 5.301786131692709 * \text{rm} + -$
10.496318655965421, 此时loss是: 50.80076431228912
在第1599步, 我们获得了函数 $f(\text{rm}) = 5.302700003930584 * \text{rm} + -$
10.502131905116315, 此时loss是: 50.79730182583938
在第1600步, 我们获得了函数 $f(\text{rm}) = 5.303613656407533 * \text{rm} + -$
10.507943756342026, 此时loss是: 50.79384100445339
在第1601步, 我们获得了函数 $f(\text{rm}) = 5.304527089176403 * \text{rm} + -$
10.513754209978718, 此时loss是: 50.79038184733029
在第1602步, 我们获得了函数 $f(\text{rm}) = 5.3054403022900285 * \text{rm} + -$
10.519563266362471, 此时loss是: 50.786924353669924
在第1603步, 我们获得了函数 $f(\text{rm}) = 5.3063532958012285 * \text{rm} + -$
10.525370925829288, 此时loss是: 50.78346852267224
在第1604步, 我们获得了函数 $f(\text{rm}) = 5.307266069762812 * \text{rm} + -$
10.531177188715086, 此时loss是: 50.7800143535377
在第1605步, 我们获得了函数 $f(\text{rm}) = 5.308178624227575 * \text{rm} + -$
10.536982055355704, 此时loss是: 50.77656184546716
在第1606步, 我们获得了函数 $f(\text{rm}) = 5.309090959248301 * \text{rm} + -$
10.542785526086902, 此时loss是: 50.77311099766184
在第1607步, 我们获得了函数 $f(\text{rm}) = 5.310003074877759 * \text{rm} + -$
10.548587601244357, 此时loss是: 50.769661809323374
在第1608步, 我们获得了函数 $f(\text{rm}) = 5.310914971168707 * \text{rm} + -$
10.554388281163666, 此时loss是: 50.76621427965371

在第1609步, 我们获得了函数 $f(\text{rm}) = 5.31182664817389 * \text{rm} + -10.560187566180344$, 此时loss是: 50.762768407855106
在第1610步, 我们获得了函数 $f(\text{rm}) = 5.312738105946039 * \text{rm} + -10.565985456629827$, 此时loss是: 50.75932419313061
在第1611步, 我们获得了函数 $f(\text{rm}) = 5.3136493445378745 * \text{rm} + -10.57178195284747$, 此时loss是: 50.755881634682986
在第1612步, 我们获得了函数 $f(\text{rm}) = 5.314560364002104 * \text{rm} + -10.577577055168547$, 此时loss是: 50.75244073171603
在第1613步, 我们获得了函数 $f(\text{rm}) = 5.31547116439142 * \text{rm} + -10.58337076392825$, 此时loss是: 50.74900148343347
在第1614步, 我们获得了函数 $f(\text{rm}) = 5.316381745758505 * \text{rm} + -10.589163079461695$, 此时loss是: 50.745563889039744
在第1615步, 我们获得了函数 $f(\text{rm}) = 5.317292108156028 * \text{rm} + -10.594954002103911$, 此时loss是: 50.74212794773944
在第1616步, 我们获得了函数 $f(\text{rm}) = 5.318202251636644 * \text{rm} + -10.600743532189853$, 此时loss是: 50.738693658737574
在第1617步, 我们获得了函数 $f(\text{rm}) = 5.319112176252998 * \text{rm} + -10.60653167005439$, 此时loss是: 50.73526102123965
在第1618步, 我们获得了函数 $f(\text{rm}) = 5.320021882057719 * \text{rm} + -10.612318416032311$, 此时loss是: 50.73183003445143
在第1619步, 我们获得了函数 $f(\text{rm}) = 5.320931369103426 * \text{rm} + -10.618103770458328$, 此时loss是: 50.72840069757912
在第1620步, 我们获得了函数 $f(\text{rm}) = 5.321840637442724 * \text{rm} + -10.623887733667072$, 此时loss是: 50.724973009829284
在第1621步, 我们获得了函数 $f(\text{rm}) = 5.322749687128208 * \text{rm} + -10.62967030599309$, 此时loss是: 50.721546970408916
在第1622步, 我们获得了函数 $f(\text{rm}) = 5.323658518212454 * \text{rm} + -10.635451487770851$, 此时loss是: 50.71812257852531
在第1623步, 我们获得了函数 $f(\text{rm}) = 5.324567130748034 * \text{rm} + -10.641231279334745$, 此时loss是: 50.714699833386284
在第1624步, 我们获得了函数 $f(\text{rm}) = 5.3254755247875 * \text{rm} + -10.647009681019078$, 此时loss是: 50.71127873419982
在第1625步, 我们获得了函数 $f(\text{rm}) = 5.326383700383395 * \text{rm} + -10.65278669315808$, 此时loss是: 50.70785928017452
在第1626步, 我们获得了函数 $f(\text{rm}) = 5.3272916575882485 * \text{rm} + -10.658562316085893$, 此时loss是: 50.70444147051911
在第1627步, 我们获得了函数 $f(\text{rm}) = 5.328199396454578 * \text{rm} + -10.66433655013659$, 此时loss是: 50.7010253044429
在第1628步, 我们获得了函数 $f(\text{rm}) = 5.3291069170348875 * \text{rm} + -10.670109395644152$, 此时loss是: 50.69761078115555
在第1629步, 我们获得了函数 $f(\text{rm}) = 5.3300142193816695 * \text{rm} + -10.67588085294249$, 此时loss是: 50.69419789986704
在第1630步, 我们获得了函数 $f(\text{rm}) = 5.330921303547402 * \text{rm} + -10.681650922365426$, 此时loss是: 50.690786659787726
在第1631步, 我们获得了函数 $f(\text{rm}) = 5.331828169584552 * \text{rm} + -10.687419604246708$, 此时loss是: 50.687377060128455

在第1632步, 我们获得了函数 $f(\text{rm}) = 5.332734817545574 * \text{rm} + -$
10.69318689892, 此时loss是: 50.68396910010026
在第1633步, 我们获得了函数 $f(\text{rm}) = 5.333641247482907 * \text{rm} + -$
10.698952806718887, 此时loss是: 50.680562778914734
在第1634步, 我们获得了函数 $f(\text{rm}) = 5.334547459448983 * \text{rm} + -$
10.704717327976875, 此时loss是: 50.67715809578379
在第1635步, 我们获得了函数 $f(\text{rm}) = 5.335453453496215 * \text{rm} + -$
10.710480463027386, 此时loss是: 50.673755049919635
在第1636步, 我们获得了函数 $f(\text{rm}) = 5.336359229677008 * \text{rm} + -$
10.716242212203767, 此时loss是: 50.670353640535
在第1637步, 我们获得了函数 $f(\text{rm}) = 5.337264788043752 * \text{rm} + -$
10.722002575839282, 此时loss是: 50.666953866842874
在第1638步, 我们获得了函数 $f(\text{rm}) = 5.338170128648827 * \text{rm} + -$
10.727761554267113, 此时loss是: 50.66355572805677
在第1639步, 我们获得了函数 $f(\text{rm}) = 5.339075251544596 * \text{rm} + -$
10.733519147820367, 此时loss是: 50.66015922339039
在第1640步, 我们获得了函数 $f(\text{rm}) = 5.339980156783414 * \text{rm} + -$
10.739275356832067, 此时loss是: 50.656764352057884
在第1641步, 我们获得了函数 $f(\text{rm}) = 5.340884844417621 * \text{rm} + -$
10.745030181635157, 此时loss是: 50.65337111327387
在第1642步, 我们获得了函数 $f(\text{rm}) = 5.341789314499545 * \text{rm} + -$
10.7507836225625, 此时loss是: 50.6499795062533
在第1643步, 我们获得了函数 $f(\text{rm}) = 5.3426935670815014 * \text{rm} + -$
10.756535679946879, 此时loss是: 50.64658953021138
在第1644步, 我们获得了函数 $f(\text{rm}) = 5.343597602215792 * \text{rm} + -$
10.762286354120999, 此时loss是: 50.64320118436389
在第1645步, 我们获得了函数 $f(\text{rm}) = 5.344501419954708 * \text{rm} + -$
10.768035645417482, 此时loss是: 50.639814467926854
在第1646步, 我们获得了函数 $f(\text{rm}) = 5.345405020350526 * \text{rm} + -$
10.773783554168874, 此时loss是: 50.63642938011666
在第1647步, 我们获得了函数 $f(\text{rm}) = 5.346308403455511 * \text{rm} + -$
10.779530080707639, 此时loss是: 50.63304592015028
在第1648步, 我们获得了函数 $f(\text{rm}) = 5.3472115693219155 * \text{rm} + -$
10.785275225366158, 此时loss是: 50.6296640872447
在第1649步, 我们获得了函数 $f(\text{rm}) = 5.348114518001982 * \text{rm} + -$
10.791018988476736, 此时loss是: 50.62628388061763
在第1650步, 我们获得了函数 $f(\text{rm}) = 5.349017249547933 * \text{rm} + -$
10.796761370371598, 此时loss是: 50.62290529948693
在第1651步, 我们获得了函数 $f(\text{rm}) = 5.349919764011988 * \text{rm} + -$
10.802502371382888, 此时loss是: 50.61952834307099
在第1652步, 我们获得了函数 $f(\text{rm}) = 5.350822061446345 * \text{rm} + -$
10.808241991842669, 此时loss是: 50.6161530105884
在第1653步, 我们获得了函数 $f(\text{rm}) = 5.351724141903197 * \text{rm} + -$
10.813980232082926, 此时loss是: 50.61277930125839
在第1654步, 我们获得了函数 $f(\text{rm}) = 5.352626005434718 * \text{rm} + -$
10.819717092435564, 此时loss是: 50.609407214300354

在第1655步, 我们获得了函数 $f(\text{rm}) = 5.353527652093075 * \text{rm} + -10.825452573232408$, 此时loss是: 50.60603674893405
在第1656步, 我们获得了函数 $f(\text{rm}) = 5.35442908193042 * \text{rm} + -10.8311866748052$, 此时loss是: 50.60266790437974
在第1657步, 我们获得了函数 $f(\text{rm}) = 5.3553302949988915 * \text{rm} + -10.83691939748561$, 此时loss是: 50.59930067985788
在第1658步, 我们获得了函数 $f(\text{rm}) = 5.356231291350615 * \text{rm} + -10.84265074160522$, 此时loss是: 50.59593507458963
在第1659步, 我们获得了函数 $f(\text{rm}) = 5.357132071037707 * \text{rm} + -10.848380707495537$, 此时loss是: 50.59257108779611
在第1660步, 我们获得了函数 $f(\text{rm}) = 5.358032634112269 * \text{rm} + -10.854109295487985$, 此时loss是: 50.58920871869915
在第1661步, 我们获得了函数 $f(\text{rm}) = 5.35893298062639 * \text{rm} + -10.859836505913913$, 此时loss是: 50.58584796652084
在第1662步, 我们获得了函数 $f(\text{rm}) = 5.359833110632147 * \text{rm} + -10.865562339104585$, 此时loss是: 50.582488830483385
在第1663步, 我们获得了函数 $f(\text{rm}) = 5.360733024181604 * \text{rm} + -10.871286795391189$, 此时loss是: 50.57913130980989
在第1664步, 我们获得了函数 $f(\text{rm}) = 5.361632721326813 * \text{rm} + -10.877009875104832$, 此时loss是: 50.57577540372344
在第1665步, 我们获得了函数 $f(\text{rm}) = 5.362532202119812 * \text{rm} + -10.882731578576541$, 此时loss是: 50.57242111144747
在第1666步, 我们获得了函数 $f(\text{rm}) = 5.363431466612629 * \text{rm} + -10.888451906137265$, 此时loss是: 50.56906843220612
在第1667步, 我们获得了函数 $f(\text{rm}) = 5.3643305148572775 * \text{rm} + -10.894170858117873$, 此时loss是: 50.565717365223634
在第1668步, 我们获得了函数 $f(\text{rm}) = 5.36522934690576 * \text{rm} + -10.899888434849153$, 此时loss是: 50.56236790972469
在第1669步, 我们获得了函数 $f(\text{rm}) = 5.366127962810066 * \text{rm} + -10.905604636661813$, 此时loss是: 50.559020064934366
在第1670步, 我们获得了函数 $f(\text{rm}) = 5.367026362622171 * \text{rm} + -10.911319463886485$, 此时loss是: 50.555673830078064
在第1671步, 我们获得了函数 $f(\text{rm}) = 5.367924546394039 * \text{rm} + -10.917032916853719$, 此时loss是: 50.552329204381586
在第1672步, 我们获得了函数 $f(\text{rm}) = 5.368822514177623 * \text{rm} + -10.922744995893984$, 此时loss是: 50.548986187071094
在第1673步, 我们获得了函数 $f(\text{rm}) = 5.369720266024861 * \text{rm} + -10.928455701337674$, 此时loss是: 50.54564477737323
在第1674步, 我们获得了函数 $f(\text{rm}) = 5.37061780198768 * \text{rm} + -10.934165033515098$, 此时loss是: 50.5423049745148
在第1675步, 我们获得了函数 $f(\text{rm}) = 5.371515122117994 * \text{rm} + -10.939872992756491$, 此时loss是: 50.53896677772316
在第1676步, 我们获得了函数 $f(\text{rm}) = 5.372412226467706 * \text{rm} + -10.945579579392005$, 此时loss是: 50.53563018622598
在第1677步, 我们获得了函数 $f(\text{rm}) = 5.373309115088704 * \text{rm} + -10.951284793751713$, 此时loss是: 50.532295199251294

在第1678步, 我们获得了函数 $f(\text{rm}) = 5.3742057880328655 * \text{rm} + -10.95698863616561$, 此时loss是: 50.52896181602742
在第1679步, 我们获得了函数 $f(\text{rm}) = 5.375102245352054 * \text{rm} + -10.96269110696361$, 此时loss是: 50.52563003578324
在第1680步, 我们获得了函数 $f(\text{rm}) = 5.375998487098122 * \text{rm} + -10.968392206475551$, 此时loss是: 50.522299857747875
在第1681步, 我们获得了函数 $f(\text{rm}) = 5.376894513322909 * \text{rm} + -10.974091935031186$, 此时loss是: 50.51897128115083
在第1682步, 我们获得了函数 $f(\text{rm}) = 5.377790324078241 * \text{rm} + -10.979790292960194$, 此时loss是: 50.51564430522199
在第1683步, 我们获得了函数 $f(\text{rm}) = 5.378685919415933 * \text{rm} + -10.985487280592173$, 此时loss是: 50.51231892919169
在第1684步, 我们获得了函数 $f(\text{rm}) = 5.3795812993877865 * \text{rm} + -10.991182898256639$, 此时loss是: 50.508995152290446
在第1685步, 我们获得了函数 $f(\text{rm}) = 5.380476464045592 * \text{rm} + -10.996877146283033$, 此时loss是: 50.50567297374931
在第1686步, 我们获得了函数 $f(\text{rm}) = 5.381371413441126 * \text{rm} + -11.002570025000717$, 此时loss是: 50.50235239279965
在第1687步, 我们获得了函数 $f(\text{rm}) = 5.382266147626153 * \text{rm} + -11.008261534738967$, 此时loss是: 50.49903340867318
在第1688步, 我们获得了函数 $f(\text{rm}) = 5.383160666652425 * \text{rm} + -11.013951675826988$, 此时loss是: 50.495716020602075
在第1689步, 我们获得了函数 $f(\text{rm}) = 5.3840549705716825 * \text{rm} + -11.019640448593902$, 此时loss是: 50.492400227818806
在第1690步, 我们获得了函数 $f(\text{rm}) = 5.384949059435652 * \text{rm} + -11.025327853368752$, 此时loss是: 50.48908602955605
在第1691步, 我们获得了函数 $f(\text{rm}) = 5.385842933296048 * \text{rm} + -11.031013890480502$, 此时loss是: 50.48577342504729
在第1692步, 我们获得了函数 $f(\text{rm}) = 5.386736592204573 * \text{rm} + -11.036698560258037$, 此时loss是: 50.482462413525916
在第1693步, 我们获得了函数 $f(\text{rm}) = 5.3876300362129195 * \text{rm} + -11.042381863030162$, 此时loss是: 50.479152994226006
在第1694步, 我们获得了函数 $f(\text{rm}) = 5.388523265372761 * \text{rm} + -11.048063799125606$, 此时loss是: 50.475845166381745
在第1695步, 我们获得了函数 $f(\text{rm}) = 5.389416279735766 * \text{rm} + -11.053744368873016$, 此时loss是: 50.472538929228
在第1696步, 我们获得了函数 $f(\text{rm}) = 5.390309079353585 * \text{rm} + -11.05942357260096$, 此时loss是: 50.46923428199962
在第1697步, 我们获得了函数 $f(\text{rm}) = 5.391201664277859 * \text{rm} + -11.065101410637928$, 此时loss是: 50.465931223932195
在第1698步, 我们获得了函数 $f(\text{rm}) = 5.392094034560216 * \text{rm} + -11.070777883312331$, 此时loss是: 50.462629754261506
在第1699步, 我们获得了函数 $f(\text{rm}) = 5.392986190252271 * \text{rm} + -11.076452990952502$, 此时loss是: 50.45932987222362
在第1700步, 我们获得了函数 $f(\text{rm}) = 5.393878131405628 * \text{rm} + -11.082126733886692$, 此时loss是: 50.45603157705505

在第1701步, 我们获得了函数 $f(\text{rm}) = 5.394769858071876 * \text{rm} + -11.087799112443076$, 此时loss是: 50.4527348679929
在第1702步, 我们获得了函数 $f(\text{rm}) = 5.395661370302594 * \text{rm} + -11.093470126949748$, 此时loss是: 50.449439744274244
在第1703步, 我们获得了函数 $f(\text{rm}) = 5.396552668149348 * \text{rm} + -11.099139777734726$, 此时loss是: 50.44614620513668
在第1704步, 我们获得了函数 $f(\text{rm}) = 5.397443751663692 * \text{rm} + -11.104808065125946$, 此时loss是: 50.44285424981838
在第1705步, 我们获得了函数 $f(\text{rm}) = 5.398334620897166 * \text{rm} + -11.110474989451266$, 此时loss是: 50.43956387755757
在第1706步, 我们获得了函数 $f(\text{rm}) = 5.399225275901299 * \text{rm} + -11.116140551038466$, 此时loss是: 50.43627508759305
在第1707步, 我们获得了函数 $f(\text{rm}) = 5.400115716727607 * \text{rm} + -11.121804750215247$, 此时loss是: 50.43298787916384
在第1708步, 我们获得了函数 $f(\text{rm}) = 5.401005943427593 * \text{rm} + -11.127467587309232$, 此时loss是: 50.42970225150952
在第1709步, 我们获得了函数 $f(\text{rm}) = 5.40189595605275 * \text{rm} + -11.133129062647962$, 此时loss是: 50.42641820386974
在第1710步, 我们获得了函数 $f(\text{rm}) = 5.402785754654557 * \text{rm} + -11.138789176558902$, 此时loss是: 50.42313573548493
在第1711步, 我们获得了函数 $f(\text{rm}) = 5.40367533928448 * \text{rm} + -11.144447929369438$, 此时loss是: 50.419854845595346
在第1712步, 我们获得了函数 $f(\text{rm}) = 5.4045647099939735 * \text{rm} + -11.150105321406876$, 此时loss是: 50.416575533442206
在第1713步, 我们获得了函数 $f(\text{rm}) = 5.4054538668344785 * \text{rm} + -11.155761352998445$, 此时loss是: 50.41329779826654
在第1714步, 我们获得了函数 $f(\text{rm}) = 5.406342809857426 * \text{rm} + -11.161416024471295$, 此时loss是: 50.41002163931024
在第1715步, 我们获得了函数 $f(\text{rm}) = 5.407231539114233 * \text{rm} + -11.167069336152496$, 此时loss是: 50.40674705581513
在第1716步, 我们获得了函数 $f(\text{rm}) = 5.408120054656303 * \text{rm} + -11.17272128836904$, 此时loss是: 50.40347404702372
在第1717步, 我们获得了函数 $f(\text{rm}) = 5.40900835653503 * \text{rm} + -11.178371881447841$, 此时loss是: 50.400202612178674
在第1718步, 我们获得了函数 $f(\text{rm}) = 5.4098964448017925 * \text{rm} + -11.184021115715735$, 此时loss是: 50.3969327505232
在第1719步, 我们获得了函数 $f(\text{rm}) = 5.410784319507961 * \text{rm} + -11.189668991499477$, 此时loss是: 50.39366446130067
在第1720步, 我们获得了函数 $f(\text{rm}) = 5.411671980704888 * \text{rm} + -11.195315509125745$, 此时loss是: 50.39039774375496
在第1721步, 我们获得了函数 $f(\text{rm}) = 5.412559428443918 * \text{rm} + -11.200960668921137$, 此时loss是: 50.38713259713025
在第1722步, 我们获得了函数 $f(\text{rm}) = 5.413446662776381 * \text{rm} + -11.206604471212176$, 此时loss是: 50.383869020671156
在第1723步, 我们获得了函数 $f(\text{rm}) = 5.414333683753596 * \text{rm} + -11.212246916325302$, 此时loss是: 50.38060701362252

在第1724步, 我们获得了函数 $f(\text{rm}) = 5.415220491426869 * \text{rm} + -11.21788800458688$, 此时loss是: 50.37734657522973
在第1725步, 我们获得了函数 $f(\text{rm}) = 5.416107085847493 * \text{rm} + -11.223527736323195$, 此时loss是: 50.37408770473842
在第1726步, 我们获得了函数 $f(\text{rm}) = 5.41699346706675 * \text{rm} + -11.229166111860453$, 此时loss是: 50.37083040139454
在第1727步, 我们获得了函数 $f(\text{rm}) = 5.417879635135909 * \text{rm} + -11.234803131524783$, 此时loss是: 50.367574664444604
在第1728步, 我们获得了函数 $f(\text{rm}) = 5.418765590106227 * \text{rm} + -11.240438795642236$, 此时loss是: 50.3643204931352
在第1729步, 我们获得了函数 $f(\text{rm}) = 5.419651332028947 * \text{rm} + -11.24607310453878$, 此时loss是: 50.36106788671348
在第1730步, 我们获得了函数 $f(\text{rm}) = 5.420536860955303 * \text{rm} + -11.25170605854031$, 此时loss是: 50.357816844427006
在第1731步, 我们获得了函数 $f(\text{rm}) = 5.421422176936512 * \text{rm} + -11.257337657972643$, 此时loss是: 50.35456736552344
在第1732步, 我们获得了函数 $f(\text{rm}) = 5.422307280023784 * \text{rm} + -11.262967903161512$, 此时loss是: 50.3513194492511
在第1733步, 我们获得了函数 $f(\text{rm}) = 5.423192170268313 * \text{rm} + -11.268596794432575$, 此时loss是: 50.34807309485849
在第1734步, 我们获得了函数 $f(\text{rm}) = 5.424076847721282 * \text{rm} + -11.274224332111414$, 此时loss是: 50.344828301594546
在第1735步, 我们获得了函数 $f(\text{rm}) = 5.424961312433862 * \text{rm} + -11.27985051652353$, 此时loss是: 50.341585068708596
在第1736步, 我们获得了函数 $f(\text{rm}) = 5.42584556445721 * \text{rm} + -11.285475347994343$, 此时loss是: 50.338343395450096
在第1737步, 我们获得了函数 $f(\text{rm}) = 5.426729603842472 * \text{rm} + -11.2910988268492$, 此时loss是: 50.33510328106919
在第1738步, 我们获得了函数 $f(\text{rm}) = 5.427613430640782 * \text{rm} + -11.296720953413368$, 此时loss是: 50.331864724816185
在第1739步, 我们获得了函数 $f(\text{rm}) = 5.428497044903261 * \text{rm} + -11.302341728012037$, 此时loss是: 50.32862772594182
在第1740步, 我们获得了函数 $f(\text{rm}) = 5.4293804466810185 * \text{rm} + -11.307961150970314$, 此时loss是: 50.325392283697155
在第1741步, 我们获得了函数 $f(\text{rm}) = 5.430263636025152 * \text{rm} + -11.313579222613232$, 此时loss是: 50.322158397333645
在第1742步, 我们获得了函数 $f(\text{rm}) = 5.431146612986743 * \text{rm} + -11.319195943265745$, 此时loss是: 50.31892606610309
在第1743步, 我们获得了函数 $f(\text{rm}) = 5.432029377616867 * \text{rm} + -11.324811313252729$, 此时loss是: 50.315695289257654
在第1744步, 我们获得了函数 $f(\text{rm}) = 5.432911929966581 * \text{rm} + -11.330425332898981$, 此时loss是: 50.31246606604977
在第1745步, 我们获得了函数 $f(\text{rm}) = 5.433794270086934 * \text{rm} + -11.336038002529222$, 此时loss是: 50.309238395732436
在第1746步, 我们获得了函数 $f(\text{rm}) = 5.4346763980289605 * \text{rm} + -11.341649322468092$, 此时loss是: 50.30601227755876

在第1747步, 我们获得了函数 $f(\text{rm}) = 5.435558313843685 * \text{rm} + -11.347259293040155$, 此时loss是: 50.30278771078243
在第1748步, 我们获得了函数 $f(\text{rm}) = 5.436440017582116 * \text{rm} + -11.352867914569895$, 此时loss是: 50.2995646946574
在第1749步, 我们获得了函数 $f(\text{rm}) = 5.437321509295255 * \text{rm} + -11.358475187381721$, 此时loss是: 50.296343228438026
在第1750步, 我们获得了函数 $f(\text{rm}) = 5.438202789034085 * \text{rm} + -11.364081111799962$, 此时loss是: 50.29312331137884
在第1751步, 我们获得了函数 $f(\text{rm}) = 5.439083856849582 * \text{rm} + -11.369685688148868$, 此时loss是: 50.289904942734935
在第1752步, 我们获得了函数 $f(\text{rm}) = 5.439964712792706 * \text{rm} + -11.375288916752615$, 此时loss是: 50.28668812176168
在第1753步, 我们获得了函数 $f(\text{rm}) = 5.4408453569144095 * \text{rm} + -11.380890797935294$, 此时loss是: 50.28347284771492
在第1754步, 我们获得了函数 $f(\text{rm}) = 5.441725789265626 * \text{rm} + -11.386491332020926$, 此时loss是: 50.28025911985064
在第1755步, 我们获得了函数 $f(\text{rm}) = 5.442606009897283 * \text{rm} + -11.392090519333449$, 此时loss是: 50.27704693742535
在第1756步, 我们获得了函数 $f(\text{rm}) = 5.443486018860292 * \text{rm} + -11.397688360196724$, 此时loss是: 50.2738362996959
在第1757步, 我们获得了函数 $f(\text{rm}) = 5.444365816205553 * \text{rm} + -11.403284854934537$, 此时loss是: 50.270627205919425
在第1758步, 我们获得了函数 $f(\text{rm}) = 5.445245401983956 * \text{rm} + -11.40888000387059$, 此时loss是: 50.26741965535345
在第1759步, 我们获得了函数 $f(\text{rm}) = 5.446124776246376 * \text{rm} + -11.414473807328516$, 此时loss是: 50.26421364725587
在第1760步, 我们获得了函数 $f(\text{rm}) = 5.447003939043675 * \text{rm} + -11.420066265631862$, 此时loss是: 50.261009180884955
在第1761步, 我们获得了函数 $f(\text{rm}) = 5.447882890426707 * \text{rm} + -11.4256573791041$, 此时loss是: 50.25780625549929
在第1762步, 我们获得了函数 $f(\text{rm}) = 5.448761630446311 * \text{rm} + -11.431247148068627$, 此时loss是: 50.25460487035793
在第1763步, 我们获得了函数 $f(\text{rm}) = 5.449640159153313 * \text{rm} + -11.436835572848757$, 此时loss是: 50.251405024719965
在第1764步, 我们获得了函数 $f(\text{rm}) = 5.450518476598528 * \text{rm} + -11.442422653767732$, 此时loss是: 50.2482067178453
在第1765步, 我们获得了函数 $f(\text{rm}) = 5.451396582832759 * \text{rm} + -11.448008391148711$, 此时loss是: 50.24500994899382
在第1766步, 我们获得了函数 $f(\text{rm}) = 5.452274477906796 * \text{rm} + -11.453592785314779$, 此时loss是: 50.241814717426074
在第1767步, 我们获得了函数 $f(\text{rm}) = 5.453152161871418 * \text{rm} + -11.45917583658894$, 此时loss是: 50.23862102240258
在第1768步, 我们获得了函数 $f(\text{rm}) = 5.45402963477739 * \text{rm} + -11.464757545294125$, 此时loss是: 50.23542886318456
在第1769步, 我们获得了函数 $f(\text{rm}) = 5.454906896675467 * \text{rm} + -11.470337911753182$, 此时loss是: 50.23223823903345

在第1770步, 我们获得了函数 $f(\text{rm}) = 5.455783947616388 * \text{rm} + -11.475916936288884$, 此时loss是: 50.229049149211136
在第1771步, 我们获得了函数 $f(\text{rm}) = 5.456660787650884 * \text{rm} + -11.481494619223929$, 此时loss是: 50.2258615929797
在第1772步, 我们获得了函数 $f(\text{rm}) = 5.457537416829673 * \text{rm} + -11.487070960880933$, 此时loss是: 50.222675569601584
在第1773步, 我们获得了函数 $f(\text{rm}) = 5.458413835203459 * \text{rm} + -11.492645961582436$, 此时loss是: 50.21949107833973
在第1774步, 我们获得了函数 $f(\text{rm}) = 5.459290042822934 * \text{rm} + -11.498219621650902$, 此时loss是: 50.216308118457505
在第1775步, 我们获得了函数 $f(\text{rm}) = 5.46016603973878 * \text{rm} + -11.503791941408714$, 此时loss是: 50.21312668921818
在第1776步, 我们获得了函数 $f(\text{rm}) = 5.461041826001665 * \text{rm} + -11.50936292117818$, 此时loss是: 50.20994678988599
在第1777步, 我们获得了函数 $f(\text{rm}) = 5.461917401662245 * \text{rm} + -11.514932561281531$, 此时loss是: 50.20676841972502
在第1778步, 我们获得了函数 $f(\text{rm}) = 5.462792766771164 * \text{rm} + -11.52050086204092$, 此时loss是: 50.203591578000015
在第1779步, 我们获得了函数 $f(\text{rm}) = 5.463667921379053 * \text{rm} + -11.52606782377842$, 此时loss是: 50.20041626397601
在第1780步, 我们获得了函数 $f(\text{rm}) = 5.464542865536533 * \text{rm} + -11.53163344681603$, 此时loss是: 50.19724247691818
在第1781步, 我们获得了函数 $f(\text{rm}) = 5.465417599294211 * \text{rm} + -11.53719773147567$, 此时loss是: 50.194070216092385
在第1782步, 我们获得了函数 $f(\text{rm}) = 5.466292122702683 * \text{rm} + -11.542760678079183$, 此时loss是: 50.19089948076463
在第1783步, 我们获得了函数 $f(\text{rm}) = 5.467166435812531 * \text{rm} + -11.548322286948334$, 此时loss是: 50.187730270201314
在第1784步, 我们获得了函数 $f(\text{rm}) = 5.468040538674326 * \text{rm} + -11.553882558404812$, 此时loss是: 50.18456258366918
在第1785步, 我们获得了函数 $f(\text{rm}) = 5.468914431338627 * \text{rm} + -11.559441492770226$, 此时loss是: 50.18139642043548
在第1786步, 我们获得了函数 $f(\text{rm}) = 5.469788113855981 * \text{rm} + -11.564999090366111$, 此时loss是: 50.17823177976748
在第1787步, 我们获得了函数 $f(\text{rm}) = 5.470661586276923 * \text{rm} + -11.570555351513923$, 此时loss是: 50.17506866093312
在第1788步, 我们获得了函数 $f(\text{rm}) = 5.471534848651975 * \text{rm} + -11.576110276535038$, 此时loss是: 50.17190706320049
在第1789步, 我们获得了函数 $f(\text{rm}) = 5.472407901031646 * \text{rm} + -11.58166386575076$, 此时loss是: 50.16874698583817
在第1790步, 我们获得了函数 $f(\text{rm}) = 5.473280743466436 * \text{rm} + -11.587216119482314$, 此时loss是: 50.16558842811516
在第1791步, 我们获得了函数 $f(\text{rm}) = 5.4741533760068295 * \text{rm} + -11.592767038050843$, 此时loss是: 50.162431389300345
在第1792步, 我们获得了函数 $f(\text{rm}) = 5.4750257987033 * \text{rm} + -11.59831662177742$, 此时loss是: 50.15927586866364

在第1793步, 我们获得了函数 $f(\text{rm}) = 5.4758980116063105 * \text{rm} + -11.603864870983035$, 此时loss是: 50.15612186547476
在第1794步, 我们获得了函数 $f(\text{rm}) = 5.47677001476631 * \text{rm} + -11.609411785988605$, 此时loss是: 50.15296937900413
在第1795步, 我们获得了函数 $f(\text{rm}) = 5.477641808233736 * \text{rm} + -11.61495736711497$, 此时loss是: 50.149818408522314
在第1796步, 我们获得了函数 $f(\text{rm}) = 5.478513392059013 * \text{rm} + -11.620501614682887$, 此时loss是: 50.14666895330023
在第1797步, 我们获得了函数 $f(\text{rm}) = 5.479384766292555 * \text{rm} + -11.626044529013042$, 此时loss是: 50.14352101260931
在第1798步, 我们获得了函数 $f(\text{rm}) = 5.480255930984764 * \text{rm} + -11.63158611042604$, 此时loss是: 50.14037458572127
在第1799步, 我们获得了函数 $f(\text{rm}) = 5.481126886186026 * \text{rm} + -11.637126359242414$, 此时loss是: 50.137229671908
在第1800步, 我们获得了函数 $f(\text{rm}) = 5.481997631946719 * \text{rm} + -11.642665275782614$, 此时loss是: 50.13408627044196
在第1801步, 我们获得了函数 $f(\text{rm}) = 5.482868168317208 * \text{rm} + -11.648202860367016$, 此时loss是: 50.13094438059594
在第1802步, 我们获得了函数 $f(\text{rm}) = 5.483738495347845 * \text{rm} + -11.653739113315918$, 此时loss是: 50.1278040016429
在第1803步, 我们获得了函数 $f(\text{rm}) = 5.484608613088973 * \text{rm} + -11.659274034949542$, 此时loss是: 50.12466513285633
在第1804步, 我们获得了函数 $f(\text{rm}) = 5.485478521590917 * \text{rm} + -11.66480762558803$, 此时loss是: 50.12152777351006
在第1805步, 我们获得了函数 $f(\text{rm}) = 5.486348220903994 * \text{rm} + -11.670339885551455$, 此时loss是: 50.11839192287808
在第1806步, 我们获得了函数 $f(\text{rm}) = 5.487217711078509 * \text{rm} + -11.675870815159803$, 此时loss是: 50.11525758023507
在第1807步, 我们获得了函数 $f(\text{rm}) = 5.488086992164755 * \text{rm} + -11.681400414732988$, 此时loss是: 50.1121247448557
在第1808步, 我们获得了函数 $f(\text{rm}) = 5.488956064213008 * \text{rm} + -11.686928684590846$, 此时loss是: 50.10899341601521
在第1809步, 我们获得了函数 $f(\text{rm}) = 5.48982492727354 * \text{rm} + -11.692455625053139$, 此时loss是: 50.105863592989124
在第1810步, 我们获得了函数 $f(\text{rm}) = 5.490693581396604 * \text{rm} + -11.697981236439547$, 此时loss是: 50.102735275053284
在第1811步, 我们获得了函数 $f(\text{rm}) = 5.491562026632446 * \text{rm} + -11.70350551906968$, 此时loss是: 50.099608461483946
在第1812步, 我们获得了函数 $f(\text{rm}) = 5.492430263031296 * \text{rm} + -11.709028473263063$, 此时loss是: 50.09648315155766
在第1813步, 我们获得了函数 $f(\text{rm}) = 5.493298290643374 * \text{rm} + -11.71455009933915$, 此时loss是: 50.09335934455134
在第1814步, 我们获得了函数 $f(\text{rm}) = 5.494166109518886 * \text{rm} + -11.720070397617315$, 此时loss是: 50.090237039742256
在第1815步, 我们获得了函数 $f(\text{rm}) = 5.4950337197080295 * \text{rm} + -11.725589368416857$, 此时loss是: 50.087116236408114

在第1816步, 我们获得了函数 $f(\text{rm}) = 5.495901121260987 * \text{rm} + -11.731107012056999$, 此时loss是: 50.08399693382669
在第1817步, 我们获得了函数 $f(\text{rm}) = 5.496768314227929 * \text{rm} + -11.736623328856885$, 此时loss是: 50.08087913127642
在第1818步, 我们获得了函数 $f(\text{rm}) = 5.4976352986590165 * \text{rm} + -11.742138319135584$, 此时loss是: 50.077762828035915
在第1819步, 我们获得了函数 $f(\text{rm}) = 5.498502074604395 * \text{rm} + -11.747651983212087$, 此时loss是: 50.07464802338428
在第1820步, 我们获得了函数 $f(\text{rm}) = 5.4993686421142 * \text{rm} + -11.75316432140531$, 此时loss是: 50.071534716600645
在第1821步, 我们获得了函数 $f(\text{rm}) = 5.500235001238553 * \text{rm} + -11.75867533403409$, 此时loss是: 50.068422906964834
在第1822步, 我们获得了函数 $f(\text{rm}) = 5.501101152027568 * \text{rm} + -11.764185021417187$, 此时loss是: 50.065312593757014
在第1823步, 我们获得了函数 $f(\text{rm}) = 5.501967094531342 * \text{rm} + -11.769693383873289$, 此时loss是: 50.062203776257455
在第1824步, 我们获得了函数 $f(\text{rm}) = 5.502832828799962 * \text{rm} + -11.775200421721003$, 此时loss是: 50.05909645374675
在第1825步, 我们获得了函数 $f(\text{rm}) = 5.503698354883502 * \text{rm} + -11.780706135278859$, 此时loss是: 50.055990625506254
在第1826步, 我们获得了函数 $f(\text{rm}) = 5.504563672832026 * \text{rm} + -11.786210524865313$, 此时loss是: 50.05288629081716
在第1827步, 我们获得了函数 $f(\text{rm}) = 5.505428782695584 * \text{rm} + -11.791713590798745$, 此时loss是: 50.049783448961314
在第1828步, 我们获得了函数 $f(\text{rm}) = 5.506293684524215 * \text{rm} + -11.797215333397455$, 此时loss是: 50.04668209922093
在第1829步, 我们获得了函数 $f(\text{rm}) = 5.507158378367946 * \text{rm} + -11.802715752979669$, 此时loss是: 50.043582240878294
在第1830步, 我们获得了函数 $f(\text{rm}) = 5.508022864276791 * \text{rm} + -11.808214849863536$, 此时loss是: 50.04048387321641
在第1831步, 我们获得了函数 $f(\text{rm}) = 5.508887142300752 * \text{rm} + -11.813712624367126$, 此时loss是: 50.037386995518254
在第1832步, 我们获得了函数 $f(\text{rm}) = 5.509751212489821 * \text{rm} + -11.819209076808438$, 此时loss是: 50.03429160706741
在第1833步, 我们获得了函数 $f(\text{rm}) = 5.510615074893975 * \text{rm} + -11.82470420750539$, 此时loss是: 50.03119770714771
在第1834步, 我们获得了函数 $f(\text{rm}) = 5.511478729563182 * \text{rm} + -11.830198016775824$, 此时loss是: 50.028105295043346
在第1835步, 我们获得了函数 $f(\text{rm}) = 5.512342176547395 * \text{rm} + -11.835690504937508$, 此时loss是: 50.02501437003881
在第1836步, 我们获得了函数 $f(\text{rm}) = 5.513205415896558 * \text{rm} + -11.841181672308132$, 此时loss是: 50.021924931418994
在第1837步, 我们获得了函数 $f(\text{rm}) = 5.5140684476606 * \text{rm} + -11.846671519205307$, 此时loss是: 50.018836978469146
在第1838步, 我们获得了函数 $f(\text{rm}) = 5.514931271889441 * \text{rm} + -11.852160045946572$, 此时loss是: 50.015750510474795

在第1839步, 我们获得了函数 $f(\text{rm}) = 5.515793888632986 * \text{rm} + -11.85764725284939$, 此时loss是: 50.01266552672189
在第1840步, 我们获得了函数 $f(\text{rm}) = 5.51665629794113 * \text{rm} + -11.863133140231142$, 此时loss是: 50.009582026496524
在第1841步, 我们获得了函数 $f(\text{rm}) = 5.5175184998637565 * \text{rm} + -11.868617708409138$, 此时loss是: 50.006500009085556
在第1842步, 我们获得了函数 $f(\text{rm}) = 5.518380494450734 * \text{rm} + -11.874100957700609$, 此时loss是: 50.003419473775715
在第1843步, 我们获得了函数 $f(\text{rm}) = 5.519242281751922 * \text{rm} + -11.87958288842271$, 此时loss是: 50.00034041985429
在第1844步, 我们获得了函数 $f(\text{rm}) = 5.520103861817168 * \text{rm} + -11.885063500892523$, 此时loss是: 49.99726284660896
在第1845步, 我们获得了函数 $f(\text{rm}) = 5.520965234696304 * \text{rm} + -11.890542795427049$, 此时loss是: 49.994186753327725
在第1846步, 我们获得了函数 $f(\text{rm}) = 5.521826400439155 * \text{rm} + -11.896020772343215$, 此时loss是: 49.9911121392988
在第1847步, 我们获得了函数 $f(\text{rm}) = 5.52268735909553 * \text{rm} + -11.901497431957873$, 此时loss是: 49.98803900381092
在第1848步, 我们获得了函数 $f(\text{rm}) = 5.523548110715228 * \text{rm} + -11.906972774587796$, 此时loss是: 49.98496734615295
在第1849步, 我们获得了函数 $f(\text{rm}) = 5.524408655348036 * \text{rm} + -11.912446800549684$, 此时loss是: 49.9818971656143
在第1850步, 我们获得了函数 $f(\text{rm}) = 5.525268993043729 * \text{rm} + -11.917919510160159$, 此时loss是: 49.97882846148474
在第1851步, 我们获得了函数 $f(\text{rm}) = 5.526129123852067 * \text{rm} + -11.923390903735765$, 此时loss是: 49.975761233054065
在第1852步, 我们获得了函数 $f(\text{rm}) = 5.526989047822804 * \text{rm} + -11.928860981592972$, 此时loss是: 49.97269547961277
在第1853步, 我们获得了函数 $f(\text{rm}) = 5.527848765005677 * \text{rm} + -11.934329744048176$, 此时loss是: 49.96963120045158
在第1854步, 我们获得了函数 $f(\text{rm}) = 5.5287082754504135 * \text{rm} + -11.939797191417695$, 此时loss是: 49.966568394861405
在第1855步, 我们获得了函数 $f(\text{rm}) = 5.529567579206726 * \text{rm} + -11.945263324017768$, 此时loss是: 49.96350706213379
在第1856步, 我们获得了函数 $f(\text{rm}) = 5.53042667632432 * \text{rm} + -11.95072814216456$, 此时loss是: 49.96044720156033
在第1857步, 我们获得了函数 $f(\text{rm}) = 5.531285566852887 * \text{rm} + -11.956191646174165$, 此时loss是: 49.95738881243312
在第1858步, 我们获得了函数 $f(\text{rm}) = 5.532144250842103 * \text{rm} + -11.961653836362593$, 此时loss是: 49.954331894044564
在第1859步, 我们获得了函数 $f(\text{rm}) = 5.533002728341637 * \text{rm} + -11.967114713045783$, 此时loss是: 49.95127644568741
在第1860步, 我们获得了函数 $f(\text{rm}) = 5.533860999401143 * \text{rm} + -11.972574276539595$, 此时loss是: 49.94822246665475
在第1861步, 我们获得了函数 $f(\text{rm}) = 5.5347190640702655 * \text{rm} + -11.978032527159815$, 此时loss是: 49.945169956239994

在第1862步, 我们获得了函数 $f(\text{rm}) = 5.535576922398634 * \text{rm} + -11.983489465222155$, 此时loss是: 49.942118913736856
在第1863步, 我们获得了函数 $f(\text{rm}) = 5.5364345744358685 * \text{rm} + -11.988945091042245$, 此时loss是: 49.939069338439495
在第1864步, 我们获得了函数 $f(\text{rm}) = 5.537292020231576 * \text{rm} + -11.994399404935645$, 此时loss是: 49.936021229642364
在第1865步, 我们获得了函数 $f(\text{rm}) = 5.538149259835352 * \text{rm} + -11.999852407217839$, 此时loss是: 49.93297458664019
在第1866步, 我们获得了函数 $f(\text{rm}) = 5.53900629329678 * \text{rm} + -12.00530409820423$, 此时loss是: 49.929929408728064
在第1867步, 我们获得了函数 $f(\text{rm}) = 5.539863120665433 * \text{rm} + -12.01075447821015$, 此时loss是: 49.92688569520157
在第1868步, 我们获得了函数 $f(\text{rm}) = 5.540719741990868 * \text{rm} + -12.016203547550852$, 此时loss是: 49.92384344535648
在第1869步, 我们获得了函数 $f(\text{rm}) = 5.541576157322634 * \text{rm} + -12.021651306541514$, 此时loss是: 49.92080265848882
在第1870步, 我们获得了函数 $f(\text{rm}) = 5.542432366710267 * \text{rm} + -12.027097755497241$, 此时loss是: 49.91776333389509
在第1871步, 我们获得了函数 $f(\text{rm}) = 5.543288370203291 * \text{rm} + -12.03254289473306$, 此时loss是: 49.91472547087216
在第1872步, 我们获得了函数 $f(\text{rm}) = 5.544144167851216 * \text{rm} + -12.037986724563922$, 此时loss是: 49.91168906871713
在第1873步, 我们获得了函数 $f(\text{rm}) = 5.544999759703544 * \text{rm} + -12.0434292453047$, 此时loss是: 49.90865412672756
在第1874步, 我们获得了函数 $f(\text{rm}) = 5.545855145809763 * \text{rm} + -12.048870457270198$, 此时loss是: 49.90562064420119
在第1875步, 我们获得了函数 $f(\text{rm}) = 5.5467103262193485 * \text{rm} + -12.054310360775137$, 此时loss是: 49.90258862043626
在第1876步, 我们获得了函数 $f(\text{rm}) = 5.5475653009817645 * \text{rm} + -12.059748956134166$, 此时loss是: 49.89955805473115
在第1877步, 我们获得了函数 $f(\text{rm}) = 5.548420070146465 * \text{rm} + -12.06518624366186$, 此时loss是: 49.896528946384905
在第1878步, 我们获得了函数 $f(\text{rm}) = 5.549274633762889 * \text{rm} + -12.070622223672714$, 此时loss是: 49.89350129469649
在第1879步, 我们获得了函数 $f(\text{rm}) = 5.550128991880466 * \text{rm} + -12.07605689648115$, 此时loss是: 49.89047509896553
在第1880步, 我们获得了函数 $f(\text{rm}) = 5.550983144548613 * \text{rm} + -12.081490262401514$, 此时loss是: 49.88745035849184
在第1881步, 我们获得了函数 $f(\text{rm}) = 5.551837091816735 * \text{rm} + -12.086922321748077$, 此时loss是: 49.88442707257567
在第1882步, 我们获得了函数 $f(\text{rm}) = 5.552690833734222 * \text{rm} + -12.092353074835032$, 此时loss是: 49.881405240517445
在第1883步, 我们获得了函数 $f(\text{rm}) = 5.55354437035046 * \text{rm} + -12.0977825219765$, 此时loss是: 49.878384861618
在第1884步, 我们获得了函数 $f(\text{rm}) = 5.554397701714815 * \text{rm} + -12.103210663486523$, 此时loss是: 49.8753659351787

在第1885步, 我们获得了函数 $f(\text{rm}) = 5.5552508278766455 * \text{rm} + -12.108637499679071$, 此时loss是: 49.87234846050092
在第1886步, 我们获得了函数 $f(\text{rm}) = 5.556103748885296 * \text{rm} + -12.114063030868037$, 此时loss是: 49.869332436886616
在第1887步, 我们获得了函数 $f(\text{rm}) = 5.556956464790102 * \text{rm} + -12.119487257367236$, 此时loss是: 49.866317863637924
在第1888步, 我们获得了函数 $f(\text{rm}) = 5.557808975640384 * \text{rm} + -12.124910179490412$, 此时loss是: 49.86330474005745
在第1889步, 我们获得了函数 $f(\text{rm}) = 5.558661281485453 * \text{rm} + -12.130331797551229$, 此时loss是: 49.860293065447976
在第1890步, 我们获得了函数 $f(\text{rm}) = 5.559513382374604 * \text{rm} + -12.135752111863278$, 此时loss是: 49.85728283911282
在第1891步, 我们获得了函数 $f(\text{rm}) = 5.560365278357126 * \text{rm} + -12.141171122740076$, 此时loss是: 49.854274060355436
在第1892步, 我们获得了函数 $f(\text{rm}) = 5.5612169694822935 * \text{rm} + -12.146588830495062$, 此时loss是: 49.85126672847981
在第1893步, 我们获得了函数 $f(\text{rm}) = 5.562068455799368 * \text{rm} + -12.152005235441601$, 此时loss是: 49.84826084279002
在第1894步, 我们获得了函数 $f(\text{rm}) = 5.562919737357601 * \text{rm} + -12.157420337892981$, 此时loss是: 49.84525640259073
在第1895步, 我们获得了函数 $f(\text{rm}) = 5.563770814206229 * \text{rm} + -12.162834138162419$, 此时loss是: 49.84225340718668
在第1896步, 我们获得了函数 $f(\text{rm}) = 5.564621686394482 * \text{rm} + -12.168246636563051$, 此时loss是: 49.83925185588324
在第1897步, 我们获得了函数 $f(\text{rm}) = 5.5654723539715745 * \text{rm} + -12.17365783340794$, 此时loss是: 49.8362517479859
在第1898步, 我们获得了函数 $f(\text{rm}) = 5.566322816986709 * \text{rm} + -12.179067729010077$, 此时loss是: 49.833253082800525
在第1899步, 我们获得了函数 $f(\text{rm}) = 5.567173075489078 * \text{rm} + -12.184476323682372$, 此时loss是: 49.83025585963339
在第1900步, 我们获得了函数 $f(\text{rm}) = 5.568023129527858 * \text{rm} + -12.189883617737664$, 此时loss是: 49.82726007779103
在第1901步, 我们获得了函数 $f(\text{rm}) = 5.5688729791522205 * \text{rm} + -12.195289611488715$, 此时loss是: 49.82426573658027
在第1902步, 我们获得了函数 $f(\text{rm}) = 5.56972262441132 * \text{rm} + -12.200694305248211$, 此时loss是: 49.82127283530837
在第1903步, 我们获得了函数 $f(\text{rm}) = 5.570572065354301 * \text{rm} + -12.206097699328765$, 此时loss是: 49.81828137328296
在第1904步, 我们获得了函数 $f(\text{rm}) = 5.571421302030295 * \text{rm} + -12.211499794042913$, 此时loss是: 49.81529134981178
在第1905步, 我们获得了函数 $f(\text{rm}) = 5.572270334488424 * \text{rm} + -12.216900589703117$, 此时loss是: 49.81230276420315
在第1906步, 我们获得了函数 $f(\text{rm}) = 5.573119162777795 * \text{rm} + -12.222300086621765$, 此时loss是: 49.80931561576553
在第1907步, 我们获得了函数 $f(\text{rm}) = 5.573967786947507 * \text{rm} + -12.227698285111165$, 此时loss是: 49.806329903807956

在第1908步, 我们获得了函数 $f(\text{rm}) = 5.574816207046643 * \text{rm} + -12.233095185483556$, 此时loss是: 49.80334562763946
在第1909步, 我们获得了函数 $f(\text{rm}) = 5.575664423124278 * \text{rm} + -12.238490788051099$, 此时loss是: 49.800362786569785
在第1910步, 我们获得了函数 $f(\text{rm}) = 5.576512435229472 * \text{rm} + -12.24388509312588$, 此时loss是: 49.797381379908636
在第1911步, 我们获得了函数 $f(\text{rm}) = 5.577360243411277 * \text{rm} + -12.249278101019907$, 此时loss是: 49.79440140696635
在第1912步, 我们获得了函数 $f(\text{rm}) = 5.578207847718728 * \text{rm} + -12.254669812045119$, 此时loss是: 49.7914228670534
在第1913步, 我们获得了函数 $f(\text{rm}) = 5.579055248200853 * \text{rm} + -12.260060226513376$, 此时loss是: 49.78844575948063
在第1914步, 我们获得了函数 $f(\text{rm}) = 5.579902444906666 * \text{rm} + -12.265449344736465$, 此时loss是: 49.78547008355936
在第1915步, 我们获得了函数 $f(\text{rm}) = 5.58074943788517 * \text{rm} + -12.270837167026096$, 此时loss是: 49.78249583860102
在第1916步, 我们获得了函数 $f(\text{rm}) = 5.581596227185354 * \text{rm} + -12.276223693693906$, 此时loss是: 49.779523023917605
在第1917步, 我们获得了函数 $f(\text{rm}) = 5.5824428128562 * \text{rm} + -12.281608925051454$, 此时loss是: 49.77655163882113
在第1918步, 我们获得了函数 $f(\text{rm}) = 5.583289194946672 * \text{rm} + -12.28699286141023$, 此时loss是: 49.77358168262424
在第1919步, 我们获得了函数 $f(\text{rm}) = 5.5841353735057275 * \text{rm} + -12.292375503081642$, 此时loss是: 49.77061315463989
在第1920步, 我们获得了函数 $f(\text{rm}) = 5.58498134858231 * \text{rm} + -12.297756850377027$, 此时loss是: 49.767646054181085
在第1921步, 我们获得了函数 $f(\text{rm}) = 5.58582712022535 * \text{rm} + -12.303136903607648$, 此时loss是: 49.764680380561416
在第1922步, 我们获得了函数 $f(\text{rm}) = 5.586672688483769 * \text{rm} + -12.30851566308469$, 此时loss是: 49.761716133094815
在第1923步, 我们获得了函数 $f(\text{rm}) = 5.587518053406474 * \text{rm} + -12.313893129119265$, 此时loss是: 49.758753311095354
在第1924步, 我们获得了函数 $f(\text{rm}) = 5.588363215042364 * \text{rm} + -12.31926930202241$, 此时loss是: 49.75579191387756
在第1925步, 我们获得了函数 $f(\text{rm}) = 5.589208173440321 * \text{rm} + -12.32464418210509$, 此时loss是: 49.7528319407563
在第1926步, 我们获得了函数 $f(\text{rm}) = 5.59005292864922 * \text{rm} + -12.330017769678188$, 此时loss是: 49.74987339104678
在第1927步, 我们获得了函数 $f(\text{rm}) = 5.5908974807179215 * \text{rm} + -12.335390065052518$, 此时loss是: 49.74691626406449
在第1928步, 我们获得了函数 $f(\text{rm}) = 5.591741829695276 * \text{rm} + -12.34076106853882$, 此时loss是: 49.74396055912522
在第1929步, 我们获得了函数 $f(\text{rm}) = 5.5925859756301195 * \text{rm} + -12.346130780447755$, 此时loss是: 49.74100627554513
在第1930步, 我们获得了函数 $f(\text{rm}) = 5.593429918571281 * \text{rm} + -12.351499201089913$, 此时loss是: 49.738053412640724

在第1931步, 我们获得了函数 $f(\text{rm}) = 5.594273658567572 * \text{rm} + -12.356866330775809$, 此时loss是: 49.73510196972885
在第1932步, 我们获得了函数 $f(\text{rm}) = 5.595117195667796 * \text{rm} + -12.36223216981588$, 此时loss是: 49.732151946126514
在第1933步, 我们获得了函数 $f(\text{rm}) = 5.595960529920744 * \text{rm} + -12.36759671852049$, 此时loss是: 49.729203341151376
在第1934步, 我们获得了函数 $f(\text{rm}) = 5.596803661375196 * \text{rm} + -12.372959977199931$, 此时loss是: 49.72625615412118
在第1935步, 我们获得了函数 $f(\text{rm}) = 5.597646590079919 * \text{rm} + -12.378321946164418$, 此时loss是: 49.72331038435397
在第1936步, 我们获得了函数 $f(\text{rm}) = 5.598489316083668 * \text{rm} + -12.38368262572409$, 此时loss是: 49.72036603116826
在第1937步, 我们获得了函数 $f(\text{rm}) = 5.599331839435187 * \text{rm} + -12.389042016189014$, 此时loss是: 49.71742309388296
在第1938步, 我们获得了函数 $f(\text{rm}) = 5.600174160183209 * \text{rm} + -12.394400117869182$, 此时loss是: 49.71448157181697
在第1939步, 我们获得了函数 $f(\text{rm}) = 5.601016278376454 * \text{rm} + -12.39975693107451$, 此时loss是: 49.71154146428976
在第1940步, 我们获得了函数 $f(\text{rm}) = 5.60185819406363 * \text{rm} + -12.405112456114841$, 此时loss是: 49.70860277062125
在第1941步, 我们获得了函数 $f(\text{rm}) = 5.602699907293435 * \text{rm} + -12.410466693299943$, 此时loss是: 49.7056654901314
在第1942步, 我们获得了函数 $f(\text{rm}) = 5.603541418114555 * \text{rm} + -12.41581964293951$, 此时loss是: 49.70272962214071
在第1943步, 我们获得了函数 $f(\text{rm}) = 5.604382726575664 * \text{rm} + -12.42117130534316$, 此时loss是: 49.69979516596984
在第1944步, 我们获得了函数 $f(\text{rm}) = 5.605223832725422 * \text{rm} + -12.426521680820438$, 此时loss是: 49.69686212093987
在第1945步, 我们获得了函数 $f(\text{rm}) = 5.606064736612479 * \text{rm} + -12.431870769680813$, 此时loss是: 49.69393048637236
在第1946步, 我们获得了函数 $f(\text{rm}) = 5.606905438285477 * \text{rm} + -12.437218572233682$, 此时loss是: 49.691000261588854
在第1947步, 我们获得了函数 $f(\text{rm}) = 5.60774593779304 * \text{rm} + -12.442565088788367$, 此时loss是: 49.68807144591155
在第1948步, 我们获得了函数 $f(\text{rm}) = 5.608586235183783 * \text{rm} + -12.447910319654113$, 此时loss是: 49.68514403866262
在第1949步, 我们获得了函数 $f(\text{rm}) = 5.609426330506311 * \text{rm} + -12.453254265140094$, 此时loss是: 49.68221803916506
在第1950步, 我们获得了函数 $f(\text{rm}) = 5.610266223809216 * \text{rm} + -12.458596925555408$, 此时loss是: 49.67929344674165
在第1951步, 我们获得了函数 $f(\text{rm}) = 5.611105915141076 * \text{rm} + -12.46393830120908$, 此时loss是: 49.676370260715764
在第1952步, 我们获得了函数 $f(\text{rm}) = 5.611945404550461 * \text{rm} + -12.469278392410056$, 此时loss是: 49.67344848041131
在第1953步, 我们获得了函数 $f(\text{rm}) = 5.612784692085928 * \text{rm} + -12.474617199467213$, 此时loss是: 49.67052810515207

在第1954步, 我们获得了函数 $f(\text{rm}) = 5.613623777796022 * \text{rm} + -12.479954722689353$, 此时loss是: 49.66760913426247
在第1955步, 我们获得了函数 $f(\text{rm}) = 5.614462661729275 * \text{rm} + -12.485290962385202$, 此时loss是: 49.66469156706716
在第1956步, 我们获得了函数 $f(\text{rm}) = 5.61530134393421 * \text{rm} + -12.490625918863412$, 此时loss是: 49.66177540289107
在第1957步, 我们获得了函数 $f(\text{rm}) = 5.616139824459337 * \text{rm} + -12.495959592432563$, 此时loss是: 49.65886064105961
在第1958步, 我们获得了函数 $f(\text{rm}) = 5.616978103353154 * \text{rm} + -12.501291983401156$, 此时loss是: 49.655947280898275
在第1959步, 我们获得了函数 $f(\text{rm}) = 5.617816180664148 * \text{rm} + -12.506623092077623$, 此时loss是: 49.65303532173311
在第1960步, 我们获得了函数 $f(\text{rm}) = 5.618654056440794 * \text{rm} + -12.51195291877032$, 此时loss是: 49.650124762890435
在第1961步, 我们获得了函数 $f(\text{rm}) = 5.619491730731555 * \text{rm} + -12.517281463787526$, 此时loss是: 49.64721560369677
在第1962步, 我们获得了函数 $f(\text{rm}) = 5.620329203584884 * \text{rm} + -12.522608727437452$, 此时loss是: 49.644307843479076
在第1963步, 我们获得了函数 $f(\text{rm}) = 5.621166475049219 * \text{rm} + -12.52793471002823$, 此时loss是: 49.64140148156459
在第1964步, 我们获得了函数 $f(\text{rm}) = 5.622003545172991 * \text{rm} + -12.533259411867919$, 此时loss是: 49.63849651728087
在第1965步, 我们获得了函数 $f(\text{rm}) = 5.622840414004615 * \text{rm} + -12.538582833264503$, 此时loss是: 49.63559294995583
在第1966步, 我们获得了函数 $f(\text{rm}) = 5.623677081592497 * \text{rm} + -12.543904974525894$, 此时loss是: 49.632690778917755
在第1967步, 我们获得了函数 $f(\text{rm}) = 5.62451354798503 * \text{rm} + -12.549225835959929$, 此时loss是: 49.62979000349514
在第1968步, 我们获得了函数 $f(\text{rm}) = 5.625349813230597 * \text{rm} + -12.55454541787437$, 此时loss是: 49.626890623016784
在第1969步, 我们获得了函数 $f(\text{rm}) = 5.626185877377567 * \text{rm} + -12.559863720576908$, 此时loss是: 49.62399263681195
在第1970步, 我们获得了函数 $f(\text{rm}) = 5.627021740474298 * \text{rm} + -12.565180744375157$, 此时loss是: 49.6210960442102
在第1971步, 我们获得了函数 $f(\text{rm}) = 5.6278574025691395 * \text{rm} + -12.570496489576657$, 此时loss是: 49.61820084454124
在第1972步, 我们获得了函数 $f(\text{rm}) = 5.628692863710425 * \text{rm} + -12.575810956488876$, 此时loss是: 49.61530703713526
在第1973步, 我们获得了函数 $f(\text{rm}) = 5.6295281239464785 * \text{rm} + -12.581124145419206$, 此时loss是: 49.61241462132293
在第1974步, 我们获得了函数 $f(\text{rm}) = 5.6303631833256125 * \text{rm} + -12.586436056674968$, 此时loss是: 49.60952359643481
在第1975步, 我们获得了函数 $f(\text{rm}) = 5.631198041896128 * \text{rm} + -12.591746690563406$, 此时loss是: 49.60663396180206
在第1976步, 我们获得了函数 $f(\text{rm}) = 5.632032699706312 * \text{rm} + -12.597056047391693$, 此时loss是: 49.60374571675626

在第1977步, 我们获得了函数 $f(\text{rm}) = 5.632867156804444 * \text{rm} + -12.602364127466924$, 此时loss是: 49.600858860629025
在第1978步, 我们获得了函数 $f(\text{rm}) = 5.633701413238789 * \text{rm} + -12.607670931096125$, 此时loss是: 49.59797339275249
在第1979步, 我们获得了函数 $f(\text{rm}) = 5.6345354690576 * \text{rm} + -12.612976458586246$, 此时loss是: 49.595089312459095
在第1980步, 我们获得了函数 $f(\text{rm}) = 5.635369324309122 * \text{rm} + -12.61828071024416$, 此时loss是: 49.59220661908157
在第1981步, 我们获得了函数 $f(\text{rm}) = 5.636202979041582 * \text{rm} + -12.623583686376673$, 此时loss是: 49.589325311953004
在第1982步, 我们获得了函数 $f(\text{rm}) = 5.637036433303202 * \text{rm} + -12.628885387290511$, 此时loss是: 49.586445390406666
在第1983步, 我们获得了函数 $f(\text{rm}) = 5.6378696871421905 * \text{rm} + -12.63418581329233$, 此时loss是: 49.58356685377637
在第1984步, 我们获得了函数 $f(\text{rm}) = 5.638702740606742 * \text{rm} + -12.63948496468871$, 此时loss是: 49.58068970139591
在第1985步, 我们获得了函数 $f(\text{rm}) = 5.63953559374504 * \text{rm} + -12.64478284178616$, 此时loss是: 49.57781393259984
在第1986步, 我们获得了函数 $f(\text{rm}) = 5.640368246605259 * \text{rm} + -12.650079444891112$, 此时loss是: 49.574939546722796
在第1987步, 我们获得了函数 $f(\text{rm}) = 5.641200699235561 * \text{rm} + -12.655374774309927$, 此时loss是: 49.572066543099645
在第1988步, 我们获得了函数 $f(\text{rm}) = 5.642032951684093 * \text{rm} + -12.66066883034889$, 此时loss是: 49.569194921065645
在第1989步, 我们获得了函数 $f(\text{rm}) = 5.642865003998996 * \text{rm} + -12.665961613314213$, 此时loss是: 49.56632467995652
在第1990步, 我们获得了函数 $f(\text{rm}) = 5.643696856228395 * \text{rm} + -12.671253123512036$, 此时loss是: 49.56345581910825
在第1991步, 我们获得了函数 $f(\text{rm}) = 5.644528508420406 * \text{rm} + -12.676543361248424$, 此时loss是: 49.56058833785695
在第1992步, 我们获得了函数 $f(\text{rm}) = 5.645359960623131 * \text{rm} + -12.681832326829369$, 此时loss是: 49.557722235539266
在第1993步, 我们获得了函数 $f(\text{rm}) = 5.646191212884664 * \text{rm} + -12.687120020560789$, 此时loss是: 49.55485751149208
在第1994步, 我们获得了函数 $f(\text{rm}) = 5.647022265253083 * \text{rm} + -12.692406442748528$, 此时loss是: 49.55199416505257
在第1995步, 我们获得了函数 $f(\text{rm}) = 5.647853117776458 * \text{rm} + -12.697691593698357$, 此时loss是: 49.549132195558265
在第1996步, 我们获得了函数 $f(\text{rm}) = 5.6486837705028465 * \text{rm} + -12.702975473715973$, 此时loss是: 49.546271602347076
在第1997步, 我们获得了函数 $f(\text{rm}) = 5.649514223480293 * \text{rm} + -12.708258083107003$, 此时loss是: 49.54341238475707
在第1998步, 我们获得了函数 $f(\text{rm}) = 5.650344476756832 * \text{rm} + -12.713539422176995$, 此时loss是: 49.54055454212677
在第1999步, 我们获得了函数 $f(\text{rm}) = 5.651174530380487 * \text{rm} + -12.718819491231425$, 此时loss是: 49.537698073795035

In [26]:

```
best_k, best_b
```

Out[26]:

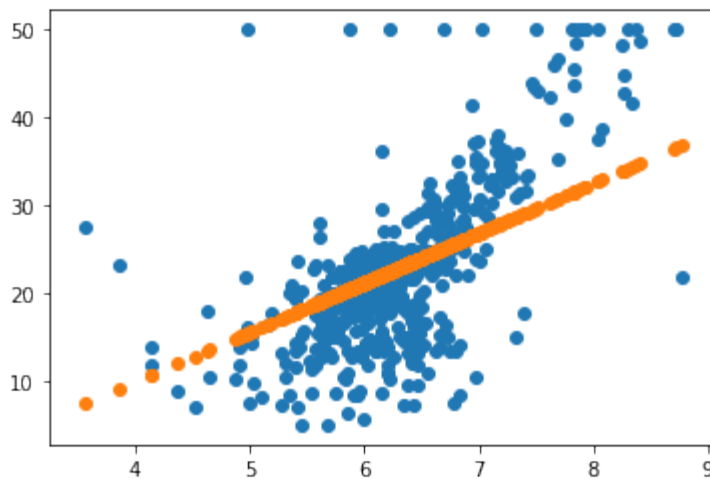
```
(5.651174530380487, -12.718819491231425)
```

In [27]:

```
plt.scatter(x, y)
plt.scatter(x, [best_k * rm + best_b for rm in x])
```

Out[27]:

```
<matplotlib.collections.PathCollection at 0x7f783c1e14d0>
```



Supervised Learning

我们把房价的预测
变成更加负责，精细的模型，该怎么做？

$$f(x) = k * x + b$$

$$f(x) = k_2 * \sigma(k_1 * x + b_1) + b_2 \quad \sigma(x) = \frac{1}{1 + e^{-x}}$$

In [28]:

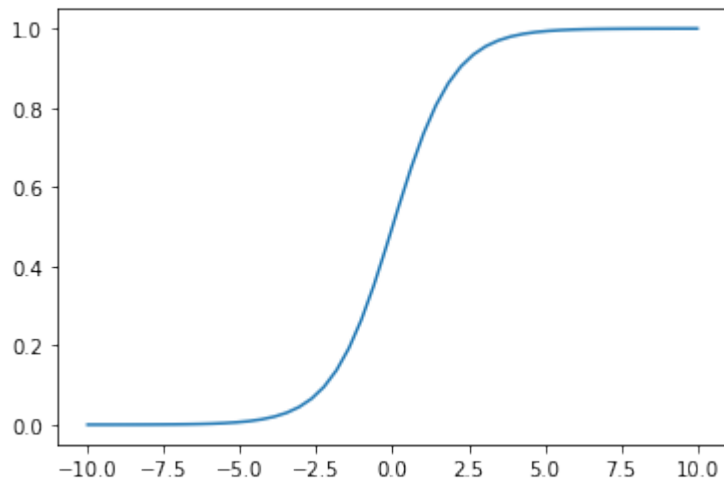
```
def sigmoid(x):
    return 1 / (1 + np.exp(-x))
```

In [29]:

```
sub_x = np.linspace(-10, 10)
plt.plot(sub_x, sigmoid(sub_x))
```

Out[29]:

```
[<matplotlib.lines.Line2D at 0x7f783c1c0f10>]
```



In [30]:

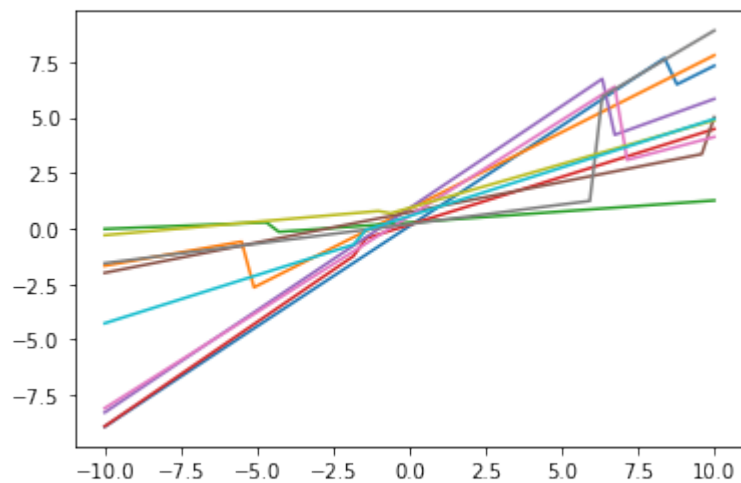
```
def random_linear(x):
    k, b = random.random(), random.random()
    return k * x + b
```

In [31]:

```
def complex_function(x):
    return (random_linear(x))
```

In [33]:

```
for _ in range(10):
    index = random.randrange(0, len(sub_x))
    sub_x_1, sub_x_2 = sub_x[:index], sub_x[index:]
    new_y = np.concatenate((complex_function(sub_x_1),
                             complex_function(sub_x_2)))
    plt.plot(sub_x, new_y)
```



我们可以通过简单的、基本的模块，经过反复的叠加，来实现更加复杂的函数

面向越来越复杂的函数？计算机如何求导？

1. 什么是机器学习？

2. KNN这种方法的缺陷，提出线性拟合的背景是什么

3.
怎么样通过监督的方法，来获得更快的函数权值更新

4.
非线性函数和线性函数的结合，可以拟合出非常复杂的函数

5.
深度学习我们可以通过基本的函数模块，来拟合更加复杂的函数

Assignment:

$$\text{L2-Loss}(y, \hat{y}) = \frac{1}{n} \sum (\hat{y} - y)^2 \quad \text{L1-Loss}(y, \hat{y}) = \frac{1}{n} \sum |(\hat{y} - y)|$$

将课堂代码中的**L2-Loss** 变成**L1Loss**
并且实现梯度下降

In [1]:

```
###Step 1: Load Data & Data Analysis
from sklearn.datasets import load_boston
```

In [2]:

```
dataset=load_boston()
```

In [3]:

```
dir(dataset)
```

Out[3]:

```
['DESCR', 'data', 'feature_names', 'filename', 'target']
```

In [4]:

```
dataset['feature_names']
```

Out[4]:

```
array(['CRIM', 'ZN', 'INDUS', 'CHAS', 'NOX', 'RM', 'AGE', 'DIS', 'RAD',
       'TAX', 'PTRATIO', 'B', 'LSTAT'], dtype='<U7')
```

In [5]:

```
###Define the problem
###Assuming you were a real state salesperson,given some description data
###about a real state => its price
```

In [6]:

```
import pandas as pd
```

In [8]:

```
dataframe=pd.DataFrame(dataset['data'])
```

In [10]:

```
dataframe.columns=dataset['feature_names']
```

In [11]:

```
dataframe['price']=dataset['target']
```

In [12]:

```
###What's the most influential factor of house price?
```

In [13]:

```
%matplotlib inline
```


In [16]:

```
import seaborn as sns
```

In [17]:

```
sns.heatmap(dataframe.corr(),annot=True,fmt='.1f')
```

Out[17]:

<matplotlib.axes._subplots.AxesSubplot at 0x7f3eed3a19d0>



In [18]:

```
###room numbers=>price?How?
```

In [19]:

```
X_rm=dataframe['RM'].values
```

In [20]:

```
Y=dataframe['price'].values
```

In [21]:

```
rm_to_price={r:y for r,y in zip(X_rm,Y)}
```

In [34]:

```
import numpy as np
```

```
def find_price_by_similar(history_price,query_x,topn=3):  
    sorted_notes=sorted(history_price.items(),key=lambda x_y:(x_y[0]-query_x)**2)  
    similar_notes=sorted_notes[:topn]  
    similar_ys=[y for _, y in similar_notes]  
    return np.mean(similar_ys)
```

In [35]:

```
find_price_by_similar(rm_to_price,5.4)
```

Out[35]:

```
15.700000000000001
```

In [27]:

```
similarity_ys = [y for _, y in sorted(rm_to_price.items(), key=lambda x_y: (x_y[0] - 6.57) ** 2)[:3]]
```

In [38]:

```
X=['张三','李四','王五']
```

```
Y1=[1,2,3]
```

```
xy={name:grade for name,grade in zip(X,Y1)}
```

In [39]:

```
xy
```

Out[39]:

```
{'张三': 1, '李四': 2, '王五': 3}
```

In [43]:

```
np.mean([g for n,g in sorted(xy.items(),key=lambda e:e[1])[:3]])
```

```
#最后一个索引是取排序中的第几位
```

Out[43]:

```
2.0
```

In [44]:

```
#以上是 knn 预测，所需的时间较长
```

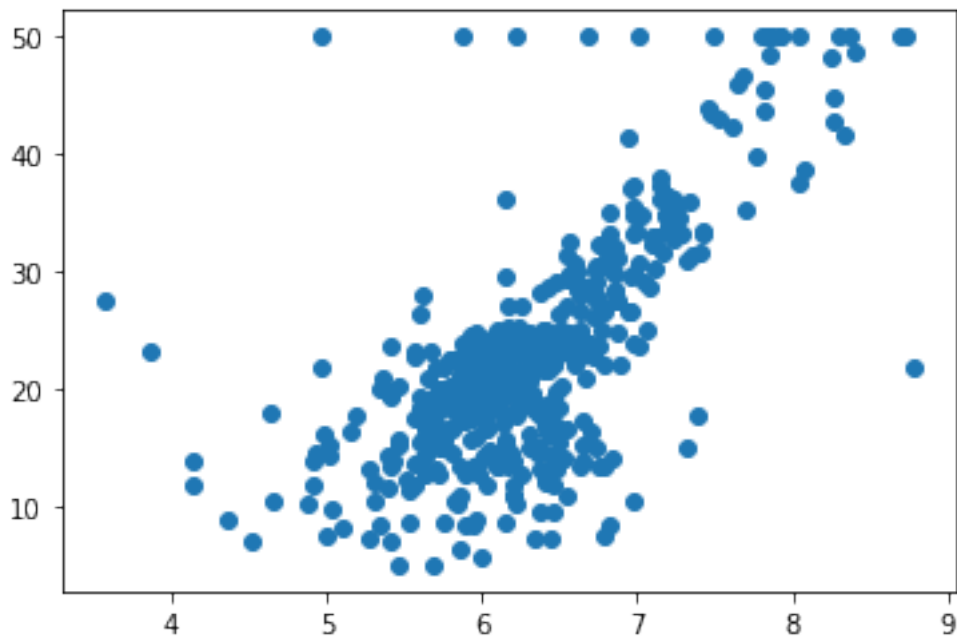
```
import matplotlib.pyplot as plt
```

In [45]:

```
plt.scatter(X_rm,Y)
```

Out[45]:

```
<matplotlib.collections.PathCollection at 0x7f3ee37f0ed0>
```



In []:

###Loss 函数，衡量信息损失，均方根误差、线性回归

In [46]:

```
def error_function(theta, X, y):  
    '''Error function J definition.'''  
    diff = np.dot(X, theta) - y  
    return (1./2*m) * np.dot(np.transpose(diff), diff)  
  
def gradient_function(theta, X, y):  
    '''Gradient of the function J definition.'''  
    diff = np.dot(X, theta) - y  
    return (1./m) * np.dot(np.transpose(X), diff)
```

In []:

###梯度下降迭代

```
def gradient_descent(X, y, alpha):  
    '''Perform gradient descent.'''  
    theta = np.array([1, 1]).reshape(2, 1)  
    gradient = gradient_function(theta, X, y)  
    while not np.all(np.absolute(gradient) <= 1e-5):  
        theta = theta - alpha * gradient  
        gradient = gradient_function(theta, X, y)  
    return theta
```

Lesson-02¶

In [1]:

```
from sklearn.datasets import load_boston
```

In [2]:

```
data = load_boston()
```

In [3]:

```
dir(data)
```

Out[3]:

```
['DESCR', 'data', 'feature_names', 'filename', 'target']
```

In [4]:

```
print(data.DESCR)
```

```
.. _boston_dataset:
```

Boston house prices dataset

****Data Set Characteristics:****

:Number of Instances: 506

:Number of Attributes: 13 numeric/categorical predictive. Median Value (attribute 14) is usually the target.

:Attribute Information (in order):

- 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 highways
- TAX	full-value property-tax rate per \$10,000
- PTRATIO	pupil-teacher ratio by town
- B	$1000(B_k - 0.63)^2$ 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

:Missing Attribute Values: None

:Creator: Harrison, D. and Rubinfeld, D.L.

This is a copy of UCI ML housing dataset.

<https://archive.ics.uci.edu/ml/machine-learning-databases/housing/>

This dataset was taken from the StatLib library which is maintained at Carnegie Mellon University.

The Boston house-price data of Harrison, D. and Rubinfeld, D.L. 'Hedonic prices and the demand for clean air', J. Environ. Economics & Management, vol.5, 81-102, 1978. Used in Belsley, Kuh & Welsch, 'Regression diagnostics ...', Wiley, 1980. N.B. Various transformations are used in the table on pages 244-261 of the latter.

The Boston house-price data has been used in many machine learning papers that address regression problems.

.. topic:: References

- Belsley, Kuh & Welsch, 'Regression diagnostics: Identifying Influential Data and Sources of Collinearity', Wiley, 1980. 244-261.

- Quinlan,R. (1993). Combining Instance-Based and Model-Based Learning. In Proceedings on the Tenth International Conference of Machine Learning, 236-243, University of Massachusetts, Amherst. Morgan Kaufmann.

In [5]:

```
X, y = data['data'], data['target']
```

In [6]:

```
X[1]
```

Out[6]:

```
array([2.7310e-02, 0.0000e+00, 7.0700e+00, 0.0000e+00, 4.6900e-01,
       6.4210e+00, 7.8900e+01, 4.9671e+00, 2.0000e+00, 2.4200e+02,
       1.7800e+01, 3.9690e+02, 9.1400e+00])
```

In [7]:

```
len(X), len(y)
```

Out[7]:

(506, 506)

In [8]:

```
import matplotlib.pyplot as plt
```

In [9]:

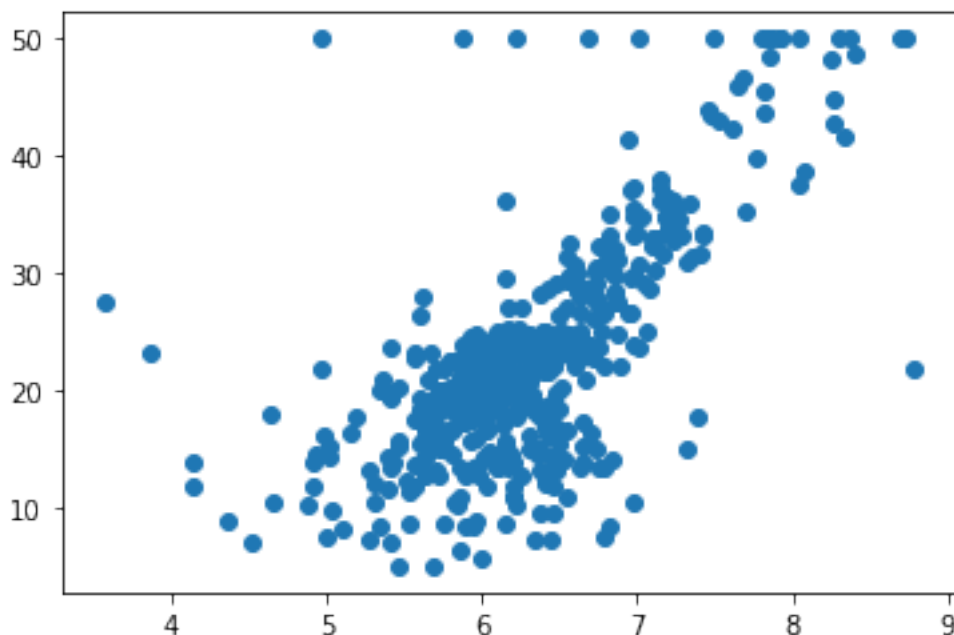
```
%matplotlib inline
```

In [10]:

```
def draw_rm_and_price():  
    plt.scatter(X[:, 5], y)
```

In [11]:

```
draw_rm_and_price()
```



需要依据一个 X ，来预测对应的 y 是多少， $rm \rightarrow y$

Simplest Method!!!

In [12]:

```
X[:, 5]
```

Out[12]:

```
array([6.575, 6.421, 7.185, 6.998, 7.147, 6.43 , 6.012, 6.172, 5.631,  
       6.004, 6.377, 6.009, 5.889, 5.949, 6.096, 5.834, 5.935, 5.99 ,  
       5.456, 5.727, 5.57 , 5.965, 6.142, 5.813, 5.924, 5.599, 5.813,  
       6.047, 6.495, 6.674, 5.713, 6.072, 5.95 , 5.701, 6.096, 5.933,  
       5.841, 5.85 , 5.966, 6.595, 7.024, 6.77 , 6.169, 6.211, 6.069,
```

5.682, 5.786, 6.03 , 5.399, 5.602, 5.963, 6.115, 6.511, 5.998,
5.888, 7.249, 6.383, 6.816, 6.145, 5.927, 5.741, 5.966, 6.456,
6.762, 7.104, 6.29 , 5.787, 5.878, 5.594, 5.885, 6.417, 5.961,
6.065, 6.245, 6.273, 6.286, 6.279, 6.14 , 6.232, 5.874, 6.727,
6.619, 6.302, 6.167, 6.389, 6.63 , 6.015, 6.121, 7.007, 7.079,
6.417, 6.405, 6.442, 6.211, 6.249, 6.625, 6.163, 8.069, 7.82 ,
7.416, 6.727, 6.781, 6.405, 6.137, 6.167, 5.851, 5.836, 6.127,
6.474, 6.229, 6.195, 6.715, 5.913, 6.092, 6.254, 5.928, 6.176,
6.021, 5.872, 5.731, 5.87 , 6.004, 5.961, 5.856, 5.879, 5.986,
5.613, 5.693, 6.431, 5.637, 6.458, 6.326, 6.372, 5.822, 5.757,
6.335, 5.942, 6.454, 5.857, 6.151, 6.174, 5.019, 5.403, 5.468,
4.903, 6.13 , 5.628, 4.926, 5.186, 5.597, 6.122, 5.404, 5.012,
5.709, 6.129, 6.152, 5.272, 6.943, 6.066, 6.51 , 6.25 , 7.489,
7.802, 8.375, 5.854, 6.101, 7.929, 5.877, 6.319, 6.402, 5.875,
5.88 , 5.572, 6.416, 5.859, 6.546, 6.02 , 6.315, 6.86 , 6.98 ,
7.765, 6.144, 7.155, 6.563, 5.604, 6.153, 7.831, 6.782, 6.556,
7.185, 6.951, 6.739, 7.178, 6.8 , 6.604, 7.875, 7.287, 7.107,
7.274, 6.975, 7.135, 6.162, 7.61 , 7.853, 8.034, 5.891, 6.326,
5.783, 6.064, 5.344, 5.96 , 5.404, 5.807, 6.375, 5.412, 6.182,
5.888, 6.642, 5.951, 6.373, 6.951, 6.164, 6.879, 6.618, 8.266,
8.725, 8.04 , 7.163, 7.686, 6.552, 5.981, 7.412, 8.337, 8.247,
6.726, 6.086, 6.631, 7.358, 6.481, 6.606, 6.897, 6.095, 6.358,
6.393, 5.593, 5.605, 6.108, 6.226, 6.433, 6.718, 6.487, 6.438,
6.957, 8.259, 6.108, 5.876, 7.454, 8.704, 7.333, 6.842, 7.203,
7.52 , 8.398, 7.327, 7.206, 5.56 , 7.014, 8.297, 7.47 , 5.92 ,
5.856, 6.24 , 6.538, 7.691, 6.758, 6.854, 7.267, 6.826, 6.482,
6.812, 7.82 , 6.968, 7.645, 7.923, 7.088, 6.453, 6.23 , 6.209,
6.315, 6.565, 6.861, 7.148, 6.63 , 6.127, 6.009, 6.678, 6.549,
5.79 , 6.345, 7.041, 6.871, 6.59 , 6.495, 6.982, 7.236, 6.616,
7.42 , 6.849, 6.635, 5.972, 4.973, 6.122, 6.023, 6.266, 6.567,
5.705, 5.914, 5.782, 6.382, 6.113, 6.426, 6.376, 6.041, 5.708,
6.415, 6.431, 6.312, 6.083, 5.868, 6.333, 6.144, 5.706, 6.031,
6.316, 6.31 , 6.037, 5.869, 5.895, 6.059, 5.985, 5.968, 7.241,
6.54 , 6.696, 6.874, 6.014, 5.898, 6.516, 6.635, 6.939, 6.49 ,
6.579, 5.884, 6.728, 5.663, 5.936, 6.212, 6.395, 6.127, 6.112,
6.398, 6.251, 5.362, 5.803, 8.78 , 3.561, 4.963, 3.863, 4.97 ,
6.683, 7.016, 6.216, 5.875, 4.906, 4.138, 7.313, 6.649, 6.794,
6.38 , 6.223, 6.968, 6.545, 5.536, 5.52 , 4.368, 5.277, 4.652,
5. , 4.88 , 5.39 , 5.713, 6.051, 5.036, 6.193, 5.887, 6.471,
6.405, 5.747, 5.453, 5.852, 5.987, 6.343, 6.404, 5.349, 5.531,
5.683, 4.138, 5.608, 5.617, 6.852, 5.757, 6.657, 4.628, 5.155,
4.519, 6.434, 6.782, 5.304, 5.957, 6.824, 6.411, 6.006, 5.648,
6.103, 5.565, 5.896, 5.837, 6.202, 6.193, 6.38 , 6.348, 6.833,
6.425, 6.436, 6.208, 6.629, 6.461, 6.152, 5.935, 5.627, 5.818,
6.406, 6.219, 6.485, 5.854, 6.459, 6.341, 6.251, 6.185, 6.417,
6.749, 6.655, 6.297, 7.393, 6.728, 6.525, 5.976, 5.936, 6.301,
6.081, 6.701, 6.376, 6.317, 6.513, 6.209, 5.759, 5.952, 6.003,

```
5.926, 5.713, 6.167, 6.229, 6.437, 6.98 , 5.427, 6.162, 6.484,  
5.304, 6.185, 6.229, 6.242, 6.75 , 7.061, 5.762, 5.871, 6.312,  
6.114, 5.905, 5.454, 5.414, 5.093, 5.983, 5.983, 5.707, 5.926,  
5.67 , 5.39 , 5.794, 6.019, 5.569, 6.027, 6.593, 6.12 , 6.976,  
6.794, 6.03 ])
```

In [13]:

y

Out[13]:

```
array([24. , 21.6, 34.7, 33.4, 36.2, 28.7, 22.9, 27.1, 16.5, 18.9, 15. ,  
      18.9, 21.7, 20.4, 18.2, 19.9, 23.1, 17.5, 20.2, 18.2, 13.6, 19.6,  
      15.2, 14.5, 15.6, 13.9, 16.6, 14.8, 18.4, 21. , 12.7, 14.5, 13.2,  
      13.1, 13.5, 18.9, 20. , 21. , 24.7, 30.8, 34.9, 26.6, 25.3, 24.7,  
      21.2, 19.3, 20. , 16.6, 14.4, 19.4, 19.7, 20.5, 25. , 23.4, 18.9,  
      35.4, 24.7, 31.6, 23.3, 19.6, 18.7, 16. , 22.2, 25. , 33. , 23.5,  
      19.4, 22. , 17.4, 20.9, 24.2, 21.7, 22.8, 23.4, 24.1, 21.4, 20. ,  
      20.8, 21.2, 20.3, 28. , 23.9, 24.8, 22.9, 23.9, 26.6, 22.5, 22.2,  
      23.6, 28.7, 22.6, 22. , 22.9, 25. , 20.6, 28.4, 21.4, 38.7, 43.8,  
      33.2, 27.5, 26.5, 18.6, 19.3, 20.1, 19.5, 19.5, 20.4, 19.8, 19.4,  
      21.7, 22.8, 18.8, 18.7, 18.5, 18.3, 21.2, 19.2, 20.4, 19.3, 22. ,  
      20.3, 20.5, 17.3, 18.8, 21.4, 15.7, 16.2, 18. , 14.3, 19.2, 19.6,  
      23. , 18.4, 15.6, 18.1, 17.4, 17.1, 13.3, 17.8, 14. , 14.4, 13.4,  
      15.6, 11.8, 13.8, 15.6, 14.6, 17.8, 15.4, 21.5, 19.6, 15.3, 19.4,  
      17. , 15.6, 13.1, 41.3, 24.3, 23.3, 27. , 50. , 50. , 50. , 22.7,  
      25. , 50. , 23.8, 23.8, 22.3, 17.4, 19.1, 23.1, 23.6, 22.6, 29.4,  
      23.2, 24.6, 29.9, 37.2, 39.8, 36.2, 37.9, 32.5, 26.4, 29.6, 50. ,  
      32. , 29.8, 34.9, 37. , 30.5, 36.4, 31.1, 29.1, 50. , 33.3, 30.3,  
      34.6, 34.9, 32.9, 24.1, 42.3, 48.5, 50. , 22.6, 24.4, 22.5, 24.4,  
      20. , 21.7, 19.3, 22.4, 28.1, 23.7, 25. , 23.3, 28.7, 21.5, 23. ,  
      26.7, 21.7, 27.5, 30.1, 44.8, 50. , 37.6, 31.6, 46.7, 31.5, 24.3,  
      31.7, 41.7, 48.3, 29. , 24. , 25.1, 31.5, 23.7, 23.3, 22. , 20.1,  
      22.2, 23.7, 17.6, 18.5, 24.3, 20.5, 24.5, 26.2, 24.4, 24.8, 29.6,  
      42.8, 21.9, 20.9, 44. , 50. , 36. , 30.1, 33.8, 43.1, 48.8, 31. ,  
      36.5, 22.8, 30.7, 50. , 43.5, 20.7, 21.1, 25.2, 24.4, 35.2, 32.4,  
      32. , 33.2, 33.1, 29.1, 35.1, 45.4, 35.4, 46. , 50. , 32.2, 22. ,  
      20.1, 23.2, 22.3, 24.8, 28.5, 37.3, 27.9, 23.9, 21.7, 28.6, 27.1,  
      20.3, 22.5, 29. , 24.8, 22. , 26.4, 33.1, 36.1, 28.4, 33.4, 28.2,  
      22.8, 20.3, 16.1, 22.1, 19.4, 21.6, 23.8, 16.2, 17.8, 19.8, 23.1,  
      21. , 23.8, 23.1, 20.4, 18.5, 25. , 24.6, 23. , 22.2, 19.3, 22.6,  
      19.8, 17.1, 19.4, 22.2, 20.7, 21.1, 19.5, 18.5, 20.6, 19. , 18.7,  
      32.7, 16.5, 23.9, 31.2, 17.5, 17.2, 23.1, 24.5, 26.6, 22.9, 24.1,  
      18.6, 30.1, 18.2, 20.6, 17.8, 21.7, 22.7, 22.6, 25. , 19.9, 20.8,  
      16.8, 21.9, 27.5, 21.9, 23.1, 50. , 50. , 50. , 50. , 50. , 13.8,  
      13.8, 15. , 13.9, 13.3, 13.1, 10.2, 10.4, 10.9, 11.3, 12.3, 8.8,  
      7.2, 10.5, 7.4, 10.2, 11.5, 15.1, 23.2, 9.7, 13.8, 12.7, 13.1,  
      12.5, 8.5, 5. , 6.3, 5.6, 7.2, 12.1, 8.3, 8.5, 5. , 11.9,
```



```
27.9, 17.2, 27.5, 15. , 17.2, 17.9, 16.3, 7. , 7.2, 7.5, 10.4,
8.8, 8.4, 16.7, 14.2, 20.8, 13.4, 11.7, 8.3, 10.2, 10.9, 11. ,
9.5, 14.5, 14.1, 16.1, 14.3, 11.7, 13.4, 9.6, 8.7, 8.4, 12.8,
10.5, 17.1, 18.4, 15.4, 10.8, 11.8, 14.9, 12.6, 14.1, 13. , 13.4,
15.2, 16.1, 17.8, 14.9, 14.1, 12.7, 13.5, 14.9, 20. , 16.4, 17.7,
19.5, 20.2, 21.4, 19.9, 19. , 19.1, 19.1, 20.1, 19.9, 19.6, 23.2,
29.8, 13.8, 13.3, 16.7, 12. , 14.6, 21.4, 23. , 23.7, 25. , 21.8,
20.6, 21.2, 19.1, 20.6, 15.2, 7. , 8.1, 13.6, 20.1, 21.8, 24.5,
23.1, 19.7, 18.3, 21.2, 17.5, 16.8, 22.4, 20.6, 23.9, 22. , 11.9])
```

In [14]:

```
rm_to_price = {rm : p for rm, p in zip(X[:, 5], y)}
```

In [15]:

```
rm_to_price[6.88]
```

```
-----
KeyError                                Traceback (most recent call last)
<ipython-input-15-57accf75d4f9> in <module>
----> 1 rm_to_price[6.88]
```

KeyError: 6.88

In [16]:

```
def distance(x1, x2):
    return (x1 - x2) **2
```

In [17]:

```
distance(0, 2)
```

Out[17]:

4

In [18]:

```
distance(-19, 18)
```

Out[18]:

1369

In [19]:

```
rm_data = X[:, 5]
```

In [20]:

```
import numpy as np
```

根据相似数据 来预测不知道的值¶

In [21]:

```
def k_nearest_model(x, k=5): # K近邻模型
    query_x = x
    k_nearest = k
    nearted_x_and_ys = sorted(rm_to_price.items(), key=lambda r_p:
distance(r_p[0], query_x))[:k_nearest]
    print(nearted_x_and_ys)
    ys = [y for x, y in nearted_x_and_ys]
    return np.mean(ys)
```

In [22]:

```
k_nearest_model(8.9)
```

```
[(8.78, 21.9), (8.725, 50.0), (8.704, 50.0), (8.398, 48.8), (8.375, 50.0)]
```

Out[22]:

```
44.14
```

时间复杂度¶

$O(N)$ # 和我们观测到的数据量成正比，观测的数据量，现在的情况下，常常会有几十万。几百万的数据

空间复杂度¶

$O(N)$ 非常耗费空间

Model -> Lazy Learning ¶

Outlinear 异常值 ¶

我们现在要有更加高级的抽象方法了！！ ¶

我们不仅仅要死记硬背！我们还要抽象出一个函数出来！
¶

最简单的找出函数的方法！！！ ¶

随机，蒙特卡洛 ¶

Function-1: Random Generate ¶

In [23]:

```
import random
```

In [24]:

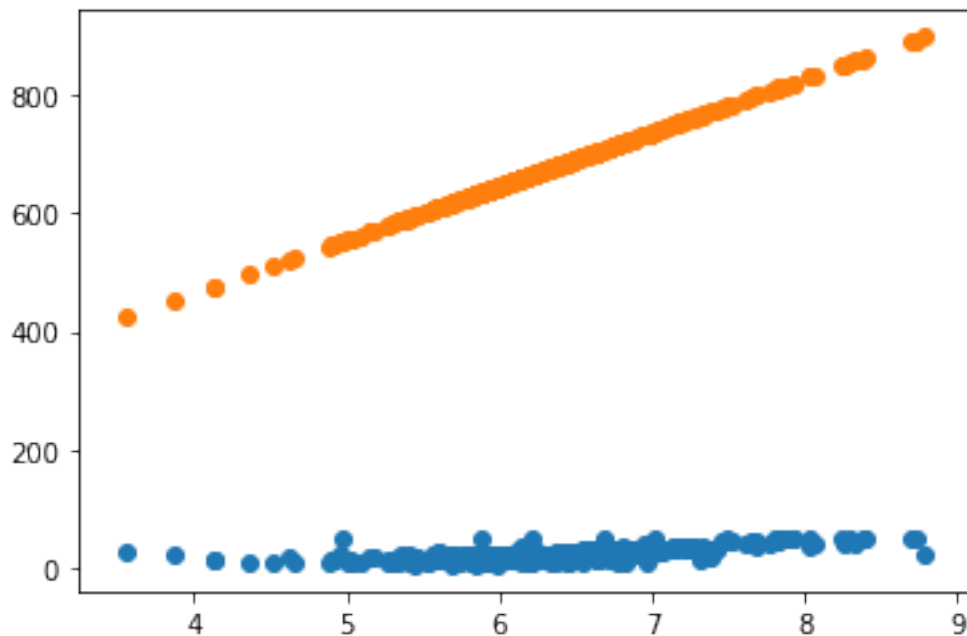
```
def random_linear_function():  
    min_v, max_v = -100, 100  
  
    k = random.randint(min_v, max_v)  
    b = random.randint(min_v, max_v)  
  
    return lambda x: k*x + b
```

In [25]:

```
function_we_get = random_linear_function()  
predicated_y = [function_we_get(x) for x in rm_data]  
  
draw_rm_and_price()  
plt.scatter(rm_data, predicated_y)
```

Out[25]:

```
<matplotlib.collections.PathCollection at 0x19a74175b48>
```



我们要找到一个好的函数 来预测¶

??什么是好? 怎么定义! ¶

In [26]:

```
def loss(predicate_y, y_true):  
    return np.mean((np.array(predicate_y) - y_true)**2)
```

In [27]:

```
list_1 = np.array([1, 2, 3, 4])
```

In [28]:

```
list_2 = np.array([0, 1, 3, 5])
```

In [29]:

```
np.mean((list_1 - list_2) **2) # numpy broadcast
```

Out[29]:

0.75

In [30]:

```
sum_ = 0  
for x1, x2 in zip(list_1, list_2):  
    sum_ += (x1 - x2)**2  
sum_ /= len(list_1)
```

In [31]:

```
from tqdm import tqdm_notebook
```

In [32]:

```
trying_time = 10000
min_loss = float('inf')
optimal_func = None

for t in range(trying_time):
    function_we_get = random_linear_function()
    predicated_y = [function_we_get(x) for x in rm_data]

    #draw_rm_and_price()
    #plt.scatter(rm_data, predicated_y)
    current_loss = loss(predicated_y, y)

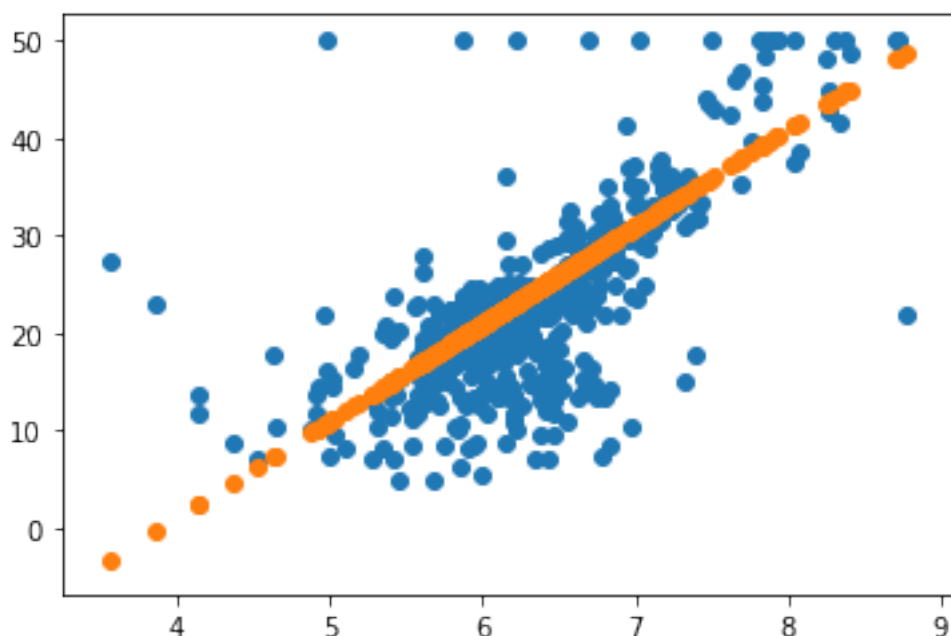
    if current_loss < min_loss:
        min_loss = current_loss
        optimal_func = function_we_get
        print('在第{} 次, 我们找到了一个更好的函数, 它的 loss 是: {}'.format(t, min_loss))

draw_rm_and_price()
plt.scatter(rm_data, optimal_func(rm_data))
```

在第 0 次, 我们找到了一个更好的函数, 它的 loss 是: 45088.61016415613
在第 4 次, 我们找到了一个更好的函数, 它的 loss 是: 1642.078932806324
在第 6 次, 我们找到了一个更好的函数, 它的 loss 是: 818.5025516837944
在第 16 次, 我们找到了一个更好的函数, 它的 loss 是: 128.28721887747037
在第 43 次, 我们找到了一个更好的函数, 它的 loss 是: 56.34183829051384
在第 1174 次, 我们找到了一个更好的函数, 它的 loss 是: 48.37242231620554
在第 4034 次, 我们找到了一个更好的函数, 它的 loss 是: 45.72314762845849

Out[32]:

<matplotlib.collections.PathCollection at 0x19a7420cc08>



Supervised Learning by Partial Loss¶

In [33]:

```
def partial_k(k, b, xi, yi):
    xi, yi = np.array(xi), np.array(yi)
    return -2 * np.sum(xi * (yi - (k*xi + b)))
```

In [34]:

```
def partial_b(k, b, xi, yi):
    xi, yi = np.array(xi), np.array(yi)
    return -2 * np.sum(yi - (k*xi + b))
```

In [35]:

```
partial_k(k=123143994, b=20, xi=rm_data, yi=y), partial_b(k=2, b=20, xi=rm_data, yi=y)
```

Out[35]:

```
(4983538469594.802, 10156.900000000001)
```

In [36]:

```
def model(x, k, b):
    return k * x + b
```

```
k, b = random.random(), random.random()
print('随机生成的 k: {}, b: {}'.format(k, b))
```

```
随机生成的 k: 0.24643382990367635, b: 0.4106473464089948
```

In [37]:

```
trying_time = 100
min_loss = float('inf')
optimal_func = None

for t in range(trying_time):
    learning_rate = 1e-3
    k = partial_k(k, b, rm_data, y) * -1 * learning_rate + k
    b = partial_b(k, b, rm_data, y) * -1 * learning_rate + b
    predicated_y = model(rm_data, k, b)

    current_loss = loss(predicated_y, y)

    if current_loss < min_loss:
        min_loss = current_loss
        optimal_func = lambda x: model(x, k, b)

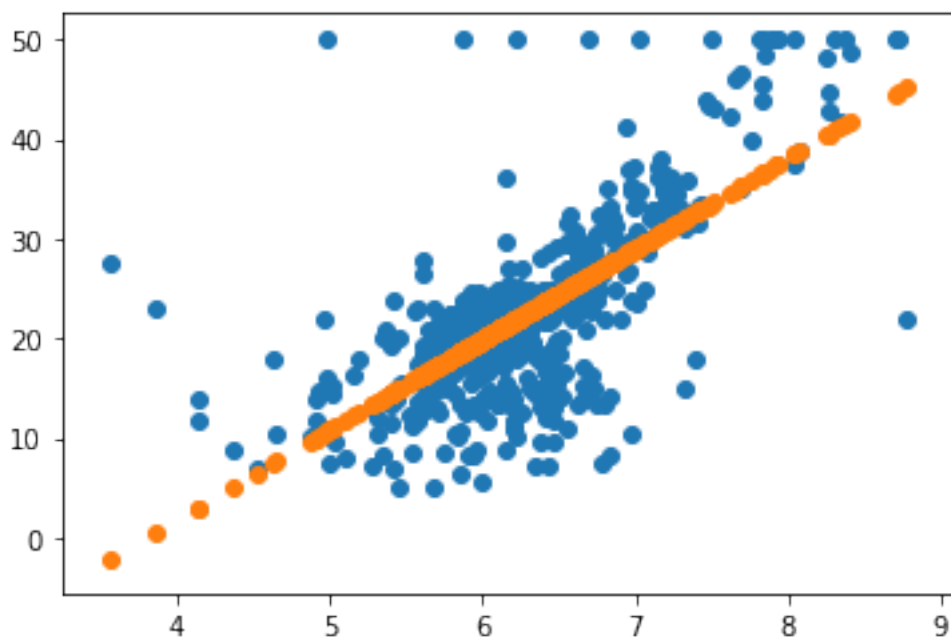
    print('在第 {} 次, 我们找到了一个更好的函数, 它的 loss 是: {}'.format(t, min_loss))
```

```
draw_rm_and_price()  
plt.scatter(rm_data, optimal_func(rm_data))
```

在第 0 次，我们找到了一个更好的函数，它的 loss 是：8015.423277757801
在第 1 次，我们找到了一个更好的函数，它的 loss 是：7951.3483093425875
在第 2 次，我们找到了一个更好的函数，它的 loss 是：2015.7627775327937
在第 3 次，我们找到了一个更好的函数，它的 loss 是：65.33244291262085
在第 7 次，我们找到了一个更好的函数，它的 loss 是：44.6712274419388
在第 11 次，我们找到了一个更好的函数，它的 loss 是：43.65595423343845
在第 15 次，我们找到了一个更好的函数，它的 loss 是：43.60355104683401
在第 19 次，我们找到了一个更好的函数，它的 loss 是：43.600720096260055
在第 23 次，我们找到了一个更好的函数，它的 loss 是：43.6005614562181
在第 27 次，我们找到了一个更好的函数，它的 loss 是：43.60055233665765
在第 31 次，我们找到了一个更好的函数，它的 loss 是：43.600551804407615
在第 35 次，我们找到了一个更好的函数，它的 loss 是：43.6005517731253
在第 39 次，我们找到了一个更好的函数，它的 loss 是：43.600551771284344
在第 43 次，我们找到了一个更好的函数，它的 loss 是：43.60055177117628
在第 46 次，我们找到了一个更好的函数，它的 loss 是：43.600551771176264
在第 47 次，我们找到了一个更好的函数，它的 loss 是：43.600551771169954
在第 50 次，我们找到了一个更好的函数，它的 loss 是：43.600551771169876
在第 51 次，我们找到了一个更好的函数，它的 loss 是：43.60055177116959
在第 54 次，我们找到了一个更好的函数，它的 loss 是：43.60055177116958
在第 55 次，我们找到了一个更好的函数，它的 loss 是：43.60055177116957
在第 60 次，我们找到了一个更好的函数，它的 loss 是：43.60055177116956
在第 71 次，我们找到了一个更好的函数，它的 loss 是：43.600551771169556

Out[37]:

<matplotlib.collections.PathCollection at 0x19a737dd588>



In [38]:

```
%%timeit
```

```
k_neartest_model(6.88)
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Lesson-03¶

In [11]:

```
import random
```

In [12]:

```
from collections import defaultdict
```

In [1]:

```
import numpy as np
```

```
class Node:
```

```
    def __init__(self, inputs=[]):
```

```
        self.inputs = inputs
```

```
        self.outputs = []
```

```
        for n in self.inputs:
```

```
            n.outputs.append(self)
```

```
            # set 'self' node as inbound_nodes's outbound_nodes
```

```
        self.value = None
```

```
        self.gradients = {}
```

```
        # keys are the inputs to this node, and their
```

```
        # values are the partials of this node with
```

```
        # respect to that input.
```

```
        # \partial{node}{input_i}
```

```
    def forward(self):
```

```
        ...
```

```
        Forward propagation.
```

```
        Compute the output value vased on 'inbound_nodes' and store the
```

```
        result in self.value
```

```
        ...
```

```
        raise NotImplemented
```

```
    def backward(self):
```

```
        raise NotImplemented
```

In [2]:

```
class Node:
    def __init__(self, inputs=[], name=None, is_trainable=True):
        self.inputs = inputs
        self.outputs = []
        self.name = name
        self.is_trainable = is_trainable      #k 和 b 的值需要更新，其它都不用训练

        for n in self.inputs:
            n.outputs.append(self)

        self.value = None

        self.gradients = {}

# node_x=Node(name='x')
# node_y=Node(name='y')
# node_k=Node(name='k')
# node_b=Node(name='b')
# node_linear = Node(inputs=[node_x,node_k,node_b],name='linear')
# node_sigmoid=Node(inputs=[node_linear])
# node_loss=Node(inputs=[node_sigmoid,node_y])

    def forward(self):
        raise NotImplementedError

    def backward(self):      #反向传播：链式求导
        raise NotImplementedError

    def __repr__(self):
        return self.name
```

In []:

```
'''need_expand=[node_x,node_y,node_k,node_b]
from collections import defaultdict
computing_graph = defaultdict(list)
while need_expand:
    n=need_expand.pop(0)
    if n in computing_graph: continue
    for m in n.outputs:
        computing_graph[n].append(m)
        need_expand.append(m)'''
```

In [3]:

```
class Placeholder(Node):      #即为输入的 k、x、b
    def __init__(self, name, is_trainable=True):
```

```

    Node.__init__(self, name=name, is_trainable=is_trainable)

def forward(self, value=None):
    if value is not None: self.value = value

def backward(self):
    self.gradients = {}
    for n in self.outputs:
        self.gradients[self] = n.gradients[self] * 1

class Linear(Node):
    def __init__(self, x=None, weigth=None, bias=None, name=None,
is_trainable=False):
        Node.__init__(self, [x, weigth, bias], name=name, is_trainable=is_trainable)

    def forward(self):
        k, x, b = self.inputs[1], self.inputs[0], self.inputs[2]
        self.value = k.value * x.value + b.value

    def backward(self):
        k, x, b = self.inputs[1], self.inputs[0], self.inputs[2]

        for n in self.outputs:
            grad_cost = n.gradients[self]

            self.gradients[k] = grad_cost * x.value

            self.gradients[x] = grad_cost * k.value

            self.gradients[b] = grad_cost * 1

class Sigmoid(Node):
    def __init__(self, x, name=None, is_trainable=False):
        Node.__init__(self, [x], name=name, is_trainable=is_trainable)
        self.x = self.inputs[0]

    def _sigmoid(self, x):
        return 1. / (1 + np.exp(-1 * x))

    def forward(self):
        self.value = self._sigmoid(self.x.value)

    def partial(self):
        return self._sigmoid(self.x.value) * (1 - self._sigmoid(self.x.value))

    def backward(self):

```



```

        for n in self.outputs:
            grad_cost = n.gradients[self]
            self.gradients[self.x] = grad_cost * self.partial()
#     print(self.gradients)

```

```

class Relu(Node):
    def __init__(self, x, name=None, is_trainable=False):
        Node.__init__(self, [x], name=name, is_trainable=is_trainable)
        self.x = x

    def forward(self):
        self.value = self.x.value * (self.x.value > 0)

    def backward(self):
        for n in self.outputs:
            grad_cost = n.gradients[self]
            self.gradients[self.x] = grad_cost * (self.x.value > 0)

class L2_LOSS(Node):
    def __init__(self, y, y_hat, name=None, is_trainable=False):
        Node.__init__(self, [y, y_hat], name=name, is_trainable=is_trainable)
        self.y = y
        self.y_hat = y_hat

    def forward(self):
        y_v = np.array(self.y.value)
        yhat_v = np.array(self.y_hat.value)
        self.value = np.mean((y_v - yhat_v) ** 2)

    def backward(self):
        # 1/n sum (y- yhat)**2
        y_v = np.array(self.y.value)
        yhat_v = np.array(self.y_hat.value)
        self.gradients[self.y] = 2 * np.mean((y_v - yhat_v))
        self.gradients[self.y_hat] = -2 * np.mean((y_v - yhat_v))
#     print(self.gradients)

```

In [5]:

```

def topologic(graph): #定义拓扑排序
    sorted_node = []

    while len(graph) > 0:

        all_inputs = []
        all_outputs = []

```

```

    for n in graph:
        all_inputs += graph[n]
        all_outputs.append(n)

    all_inputs = set(all_inputs)
    all_outputs = set(all_outputs)

    need_remove = all_outputs - all_inputs # which in all_inputs but not in
all_outputs

    if len(need_remove) > 0:
        node = random.choice(list(need_remove))

        visited_next = [node]
        if len(graph) == 1: visited_next += graph[node]    #如果是最后一个节点了，则
需保留

        graph.pop(node)
        sorted_node += visited_next

        for _, links in graph.items():
            if node in links: links.remove(node)
        else:
            break            #raise TypeError("This graph has circle and cannot
give topologic order")

    return sorted_node

```

In [4]:

```

def topological_sort_feed_dict(feed_dict):
    graph = convert_feed_dict_to_graph(feed_dict)

    return topologic(graph)

```

In [6]:

```

def convert_feed_dict_to_graph(feed_dict):
    computing_graph = defaultdict(list)

    nodes = [n for n in feed_dict]

    while nodes:
        n = nodes.pop(0)

        if isinstance(n, Placeholder):
            n.value = feed_dict[n]

```

```
if n in computing_graph: continue
```

```
for m in n.outputs:  
    computing_graph[n].append(m)  
    nodes.append(m)
```

```
return computing_graph
```

In [18]:

```
def forward(graph_order, monitor=False):  
    for node in graph_order:  
        if monitor: print('forward computing -- {}'.format(node))  
        node.forward()
```

```
def backward(graph_order, monitor=False):  
    for node in graph_order[::-1]:  
        if monitor: print('backward computing -- {}'.format(node))  
        node.backward()
```

In [19]:

```
def run_one_epoch(graph_order, monitor=False): #正向传播一次, 反向传播一次  
    forward(graph_order, monitor)  
    backward(graph_order, monitor)
```

In [8]:

```
def optimize(graph, learning_rate=1e-2):  
    # there are so many other update / optimization methods  
    # such as Adam, Mom,  
    for t in graph:  
        if t.is_trainable:  
            t.value += -1 * learning_rate * t.gradients[t] #learning_rate: 微小波动
```

In [9]:

```
from sklearn.datasets import load_boston
```

In [14]:

```
from tqdm import tqdm_notebook
```

In [124]:

```
data = load_boston()  
X_, y_ = data['data'], data['target']  
X_rm = X_[:, 5]
```

```
w1_, b1_ = np.random.normal(), np.random.normal()  
w2_, b2_ = np.random.normal(), np.random.normal()  
w3_, b3_ = np.random.normal(), np.random.normal()
```

```

X, y = Placeholder(name='X', is_trainable=False), Placeholder(name='y',
is_trainable=False)
w1, b1 = Placeholder(name='w1'), Placeholder(name='b1')
w2, b2 = Placeholder(name='w2'), Placeholder(name='b2')

# build model
output1 = Linear(X, w1, b1, name='linear-01')
output2 = Sigmoid(output1, name='activation')
#output2 = Relu(output1, name='activation')
y_hat = Linear(output2, w2, b2, name='y_hat')
cost = L2_LOSS(y, y_hat, name='cost')

feed_dict = {
    X: X_rm,
    y: y_,
    w1: w1_,
    w2: w2_,
    b1: b1_,
    b2: b2_,
}

graph_sort = topological_sort_feed_dict(feed_dict)

epoch = 1000

batch_num = len(X_rm)

learning_rate = 1e-3

losses = []

for e in tqdm_notebook(range(epoch)):
    loss = 0

    for b in range(batch_num):
        index = np.random.choice(range(len(X_rm)))
        X.value = X_rm[index]
        y.value = y_[index]

        run_one_epoch(graph_sort, monitor=False)

    optimize(graph_sort, learning_rate)

    loss += cost.value

```

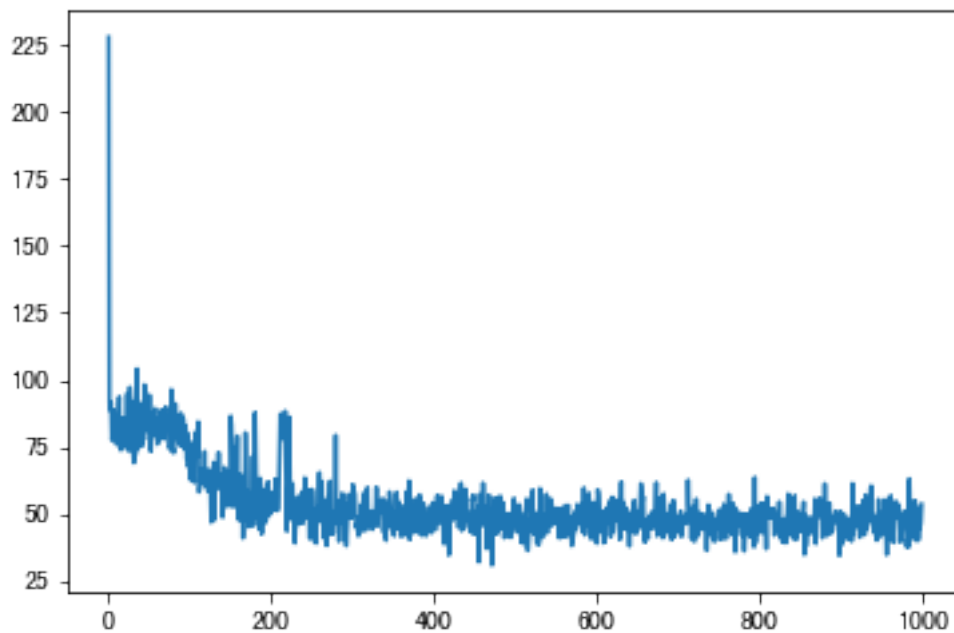
```
losses.append(loss / batch_num)
```

In [125]:

```
plt.plot(losses)
```

Out[125]:

[<matplotlib.lines.Line2D at 0x7f90569c2e80>]



In [126]:

```
def predicate(x, graph):  
    X.value = x  
    forward(graph)  
    return y_hat.value
```

In [127]:

```
import matplotlib.pyplot as plt
```

In [131]:

```
predicate(7, graph_sort)
```

Out[131]:

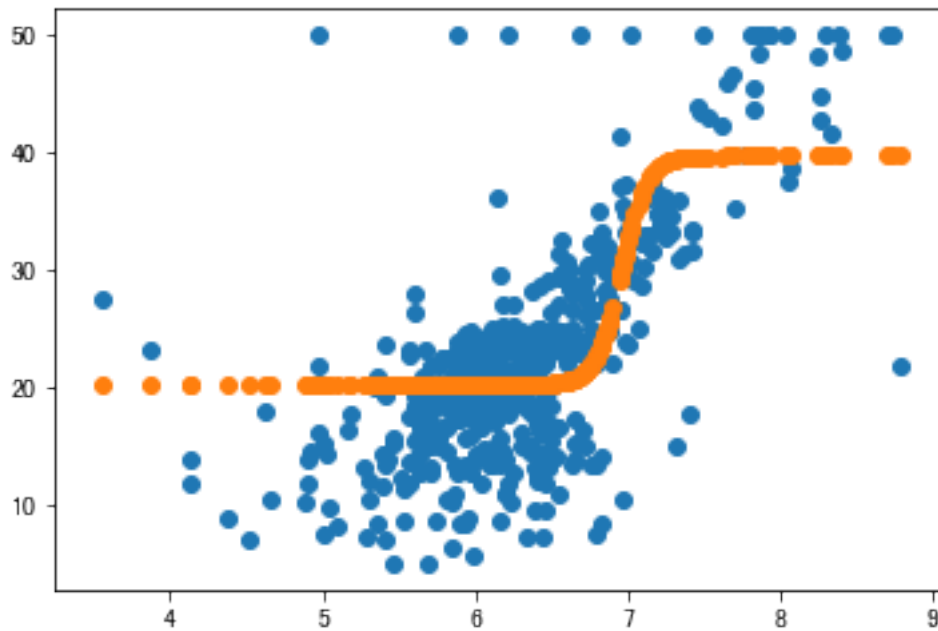
32.66592258031777

In [132]:

```
plt.scatter(X_rm, y_)  
plt.scatter(X_rm, [predicate(x, graph_sort) for x in X_rm])
```

Out[132]:

<matplotlib.collections.PathCollection at 0x7f9056878a58>



多维向量版本¶

In [156]:

```
import numpy as np
import random
```

```
class Node:
```

```
    def __init__(self, inputs=[]):
        self.inputs = inputs
        self.outputs = []

        for n in self.inputs:
            n.outputs.append(self)
            # set 'self' node as inbound_nodes's outbound_nodes
```

```
        self.value = None
```

```
        self.gradients = {}
        # keys are the inputs to this node, and their
        # values are the partials of this node with
        # respect to that input.
        # \partial{node}{input_i}
```

```
    def forward(self):
```

```
        ...
```

```
        Forward propagation.
```

```
        Compute the output value vased on 'inbound_nodes' and store the
```

```

        result in self.value
        ...

        raise NotImplemented

def backward(self):

    raise NotImplemented

class Placeholder(Node):
    def __init__(self):
        ...

        An Input node has no inbound nodes.
        So no need to pass anything to the Node instantiator.
        ...

        Node.__init__(self)

    def forward(self, value=None):
        ...

        Only input node is the node where the value may be passed
        as an argument to forward().
        All other node implementations should get the value of the
        previous node from self.inbound_nodes

        Example:
        val0: self.inbound_nodes[0].value
        ...

        if value is not None:
            self.value = value
            ## It's is input node, when need to forward, this node initiate self's
value.

        # Input subclass just holds a value, such as a data feature or a model
parameter(weight/bias)

    def backward(self):
        self.gradients = {self:0}
        for n in self.outputs:
            grad_cost = n.gradients[self]
            self.gradients[self] = grad_cost * 1

        # input N --> N1, N2
        # \partial L / \partial N
        # ==> \partial L / \partial N1 * \partial N1 / \partial N

```

```

class Add(Node):
    def __init__(self, *nodes):
        Node.__init__(self, nodes)

    def forward(self):
        self.value = sum(map(lambda n: n.value, self.inputs))
        ## when execute forward, this node caculate value as defined.

class Linear(Node):
    def __init__(self, nodes, weights, bias):
        Node.__init__(self, [nodes, weights, bias])

    def forward(self):
        inputs = self.inputs[0].value
        weights = self.inputs[1].value
        bias = self.inputs[2].value

        self.value = np.dot(inputs, weights) + bias

    def backward(self):

        # initial a partial for each of the inbound_nodes.
        self.gradients = {n: np.zeros_like(n.value) for n in self.inputs}
#初始化: 0 矩阵

        for n in self.outputs:
            # Get the partial of the cost w.r.t this node.
            grad_cost = n.gradients[self]

            self.gradients[self.inputs[0]] = np.dot(grad_cost,
self.inputs[1].value.T)          #求偏导=>使用矩阵
            self.gradients[self.inputs[1]] = np.dot(self.inputs[0].value.T,
grad_cost)
            self.gradients[self.inputs[2]] = np.sum(grad_cost, axis=0,
keepdims=False)

            #  $WX + B / W \implies X$ 
            #  $WX + B / X \implies W$ 

class Sigmoid(Node):
    def __init__(self, node):
        Node.__init__(self, [node])

    def _sigmoid(self, x):

```



```

        return 1./(1 + np.exp(-1 * x))

def forward(self):
    self.x = self.inputs[0].value
    self.value = self._sigmoid(self.x)

def backward(self):
    self.partial = self._sigmoid(self.x) * (1 - self._sigmoid(self.x))

    #  $y = 1 / (1 + e^{-x})$ 
    #  $y' = 1 / (1 + e^{-x}) (1 - 1 / (1 + e^{-x}))$ 

    self.gradients = {n: np.zeros_like(n.value) for n in self.inputs}

    for n in self.outputs:
        grad_cost = n.gradients[self] # Get the partial of the cost with
respect to this node.

        self.gradients[self.inputs[0]] = grad_cost * self.partial
        # use * to keep all the dimension same!.

class MSE(Node):
    def __init__(self, y, a):
        Node.__init__(self, [y, a])

    def forward(self):
        y = self.inputs[0].value.reshape(-1, 1)
        a = self.inputs[1].value.reshape(-1, 1)
        assert(y.shape == a.shape)

        self.m = self.inputs[0].value.shape[0]
        self.diff = y - a

        self.value = np.mean(self.diff**2)

    def backward(self):
        self.gradients[self.inputs[0]] = (2 / self.m) * self.diff
        self.gradients[self.inputs[1]] = (-2 / self.m) * self.diff

def forward_and_backward(graph):
    # execute all the forward method of sorted_nodes.

```

In practice, it's common to feed in multiple data examples in each forward pass rather than just 1. Because the examples can be processed in parallel. The number of examples is called batch size.

```
for n in graph:
    n.forward()
    ## each node execute forward, get self.value based on the topological sort
    result.
```

```
for n in graph[::-1]:
    n.backward()
```

```
### v --> a --> C
## b --> C
## b --> v -- a --> C
## v --> v ---> a --> C
```

```
def topologic(graph):
    sorted_node = []

    while len(graph) > 0:

        all_inputs = []
        all_outputs = []

        for n in graph:
            all_inputs += graph[n]
            all_outputs.append(n)

        all_inputs = set(all_inputs)
        all_outputs = set(all_outputs)

        need_remove = all_outputs - all_inputs # which in all_outputs but not in
all_inputs

        if len(need_remove) > 0:
            node = random.choice(list(need_remove))

            need_to_visited = [node]

            if len(graph) == 1: need_to_visited += graph[node]

            graph.pop(node)
            sorted_node += need_to_visited

            for _, links in graph.items():
                if node in links: links.remove(node)
            else: # have cycle
```

```
break
```

```
return sorted_node
```

```
from collections import defaultdict
```

```
def convert_feed_dict_to_graph(feed_dict):
```

```
    computing_graph = defaultdict(list)
```

```
    nodes = [n for n in feed_dict]
```

```
    while nodes:
```

```
        n = nodes.pop(0)
```

```
        if isinstance(n, Placeholder):
```

```
            n.value = feed_dict[n]
```

```
        if n in computing_graph: continue
```

```
        for m in n.outputs:
```

```
            computing_graph[n].append(m)
```

```
            nodes.append(m)
```

```
    return computing_graph
```

```
def topological_sort_feed_dict(feed_dict):
```

```
    graph = convert_feed_dict_to_graph(feed_dict)
```

```
    return topologic(graph)
```

```
def optimize(trainables, learning_rate=1e-2):
```

```
    # there are so many other update / optimization methods
```

```
    # such as Adam, Mom,
```

```
    for t in trainables:
```

```
        t.value += -1 * learning_rate * t.gradients[t]
```

```
In [134]:
```

```
import numpy as np
```

```
from sklearn.datasets import load_boston
```

```
from sklearn.utils import shuffle, resample
```

```
#from miniflow import *
```

```
# Load data
```

```
data = load_boston()
```

```
X_ = data['data']
```

```

y_ = data['target']

# Normalize data
X_ = (X_ - np.mean(X_, axis=0)) / np.std(X_, axis=0)

n_features = X_.shape[1]
n_hidden = 10
W1_ = np.random.randn(n_features, n_hidden)
b1_ = np.zeros(n_hidden)
W2_ = np.random.randn(n_hidden, 1)
b2_ = np.zeros(1)

# Neural network
X, y = Placeholder(), Placeholder()
W1, b1 = Placeholder(), Placeholder()
W2, b2 = Placeholder(), Placeholder()

l1 = Linear(X, W1, b1)
s1 = Sigmoid(l1)
l2 = Linear(s1, W2, b2)
cost = MSE(y, l2)

feed_dict = {
    X: X_,
    y: y_,
    W1: W1_,
    b1: b1_,
    W2: W2_,
    b2: b2_
}

epochs = 5000
# Total number of examples
m = X_.shape[0]
batch_size = 16
steps_per_epoch = m // batch_size

graph = topological_sort_feed_dict(feed_dict)
trainables = [W1, b1, W2, b2]

print("Total number of examples = {}".format(m))

Total number of examples = 506

In [135]:
losses = []

```

```

for i in range(epochs):
    loss = 0
    for j in range(steps_per_epoch):
        # Step 1
        # Randomly sample a batch of examples
        X_batch, y_batch = resample(X_, y_, n_samples=batch_size)

        # Reset value of X and y Inputs
        X.value = X_batch
        y.value = y_batch

        # Step 2
        _ = None
        forward_and_backward(_, graph) # set output node not important.

        # Step 3
        rate = 1e-2

        optimize(trainables, rate)

        loss += graph[-1].value

    if i % 100 == 0:
        print("Epoch: {}, Loss: {:.3f}".format(i+1, loss/steps_per_epoch))
        losses.append(loss/steps_per_epoch)

```

```

Epoch: 1, Loss: 193.699
Epoch: 101, Loss: 7.604
Epoch: 201, Loss: 5.731
Epoch: 301, Loss: 5.680
Epoch: 401, Loss: 4.977
Epoch: 501, Loss: 5.376
Epoch: 601, Loss: 5.091
Epoch: 701, Loss: 3.858
Epoch: 801, Loss: 5.181
Epoch: 901, Loss: 4.455
Epoch: 1001, Loss: 3.751
Epoch: 1101, Loss: 4.068
Epoch: 1201, Loss: 4.022
Epoch: 1301, Loss: 4.051
Epoch: 1401, Loss: 3.523
Epoch: 1501, Loss: 3.897
Epoch: 1601, Loss: 4.073
Epoch: 1701, Loss: 3.562
Epoch: 1801, Loss: 3.625
Epoch: 1901, Loss: 3.856
Epoch: 2001, Loss: 3.969
Epoch: 2101, Loss: 3.579

```

Epoch: 2201, Loss: 3.204
Epoch: 2301, Loss: 3.797
Epoch: 2401, Loss: 3.611
Epoch: 2501, Loss: 3.823
Epoch: 2601, Loss: 3.705
Epoch: 2701, Loss: 3.649
Epoch: 2801, Loss: 3.056
Epoch: 2901, Loss: 3.338
Epoch: 3001, Loss: 3.610
Epoch: 3101, Loss: 3.777
Epoch: 3201, Loss: 3.240
Epoch: 3301, Loss: 3.780
Epoch: 3401, Loss: 3.851
Epoch: 3501, Loss: 3.437
Epoch: 3601, Loss: 3.510
Epoch: 3701, Loss: 3.583
Epoch: 3801, Loss: 3.211
Epoch: 3901, Loss: 4.325
Epoch: 4001, Loss: 3.075
Epoch: 4101, Loss: 3.433
Epoch: 4201, Loss: 3.483
Epoch: 4301, Loss: 3.405
Epoch: 4401, Loss: 3.370
Epoch: 4501, Loss: 3.355
Epoch: 4601, Loss: 3.395
Epoch: 4701, Loss: 3.519
Epoch: 4801, Loss: 3.057
Epoch: 4901, Loss: 3.442

2-Dimensions Version¶

In [145]:

```
import pandas as pd
```

In [149]:

```
dataframe = pd.DataFrame(data['data'])  
dataframe.columns = data['feature_names']
```

In [150]:

```
dataframe
```

Out[150]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33
5	0.02985	0.0	2.18	0.0	0.458	6.430	58.7	6.0622	3.0	222.0	18.7	394.12	5.21
6	0.08829	12.5	7.87	0.0	0.524	6.012	66.6	5.5605	5.0	311.0	15.2	395.60	12.43
7	0.14455	12.5	7.87	0.0	0.524	6.172	96.1	5.9505	5.0	311.0	15.2	396.90	19.15
8	0.21124	12.5	7.87	0.0	0.524	5.631	100.0	6.0821	5.0	311.0	15.2	386.63	29.93
9	0.17004	12.5	7.87	0.0	0.524	6.004	85.9	6.5921	5.0	311.0	15.2	386.71	17.10
10	0.22489	12.5	7.87	0.0	0.524	6.377	94.3	6.3467	5.0	311.0	15.2	392.52	20.45
11	0.11747	12.5	7.87	0.0	0.524	6.009	82.9	6.2267	5.0	311.0	15.2	396.90	13.27
12	0.09378	12.5	7.87	0.0	0.524	5.889	39.0	5.4509	5.0	311.0	15.2	390.50	15.71
13	0.62976	0.0	8.14	0.0	0.538	5.949	61.8	4.7075	4.0	307.0	21.0	396.90	8.26
14	0.63796	0.0	8.14	0.0	0.538	6.096	84.5	4.4619	4.0	307.0	21.0	380.02	10.26
15	0.62739	0.0	8.14	0.0	0.538	5.834	56.5	4.4986	4.0	307.0	21.0	395.62	8.47
16	1.05393	0.0	8.14	0.0	0.538	5.935	29.3	4.4986	4.0	307.0	21.0	386.85	6.58
17	0.78420	0.0	8.14	0.0	0.538	5.990	81.7	4.2579	4.0	307.0	21.0	386.75	14.67
18	0.80271	0.0	8.14	0.0	0.538	5.456	36.6	3.7965	4.0	307.0	21.0	288.99	11.69
19	0.72580	0.0	8.14	0.0	0.538	5.727	69.5	3.7965	4.0	307.0	21.0	390.95	11.28
20	1.25179	0.0	8.14	0.0	0.538	5.570	98.1	3.7979	4.0	307.0	21.0	376.57	21.02
21	0.85204	0.0	8.14	0.0	0.538	5.965	89.2	4.0123	4.0	307.0	21.0	392.53	13.83
22	1.23247	0.0	8.14	0.0	0.538	6.142	91.7	3.9769	4.0	307.0	21.0	396.90	18.72
23	0.98843	0.0	8.14	0.0	0.538	5.813	100.0	4.0952	4.0	307.0	21.0	394.54	19.88
24	0.75026	0.0	8.14	0.0	0.538	5.924	94.1	4.3996	4.0	307.0	21.0	394.33	16.30
25	0.84054	0.0	8.14	0.0	0.538	5.599	85.7	4.4546	4.0	307.0	21.0	303.42	16.51
26	0.67191	0.0	8.14	0.0	0.538	5.813	90.3	4.6820	4.0	307.0	21.0	376.88	14.81
27	0.95577	0.0	8.14	0.0	0.538	6.047	88.8	4.4534	4.0	307.0	21.0	306.38	17.28
28	0.77299	0.0	8.14	0.0	0.538	6.495	94.4	4.4547	4.0	307.0	21.0	387.94	12.80
29	1.00245	0.0	8.14	0.0	0.538	6.674	87.3	4.2390	4.0	307.0	21.0	380.23	11.98

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
...
476	4.87141	0.0	18.10	0.0	0.614	6.484	93.6	2.3053	24.0	666.0	20.2	396.21	18.68
477	15.02340	0.0	18.10	0.0	0.614	5.304	97.3	2.1007	24.0	666.0	20.2	349.48	24.91
478	10.23300	0.0	18.10	0.0	0.614	6.185	96.7	2.1705	24.0	666.0	20.2	379.70	18.03
479	14.33370	0.0	18.10	0.0	0.614	6.229	88.0	1.9512	24.0	666.0	20.2	383.32	13.11
480	5.82401	0.0	18.10	0.0	0.532	6.242	64.7	3.4242	24.0	666.0	20.2	396.90	10.74
481	5.70818	0.0	18.10	0.0	0.532	6.750	74.9	3.3317	24.0	666.0	20.2	393.07	7.74
482	5.73116	0.0	18.10	0.0	0.532	7.061	77.0	3.4106	24.0	666.0	20.2	395.28	7.01
483	2.81838	0.0	18.10	0.0	0.532	5.762	40.3	4.0983	24.0	666.0	20.2	392.92	10.42
484	2.37857	0.0	18.10	0.0	0.583	5.871	41.9	3.7240	24.0	666.0	20.2	370.73	13.34
485	3.67367	0.0	18.10	0.0	0.583	6.312	51.9	3.9917	24.0	666.0	20.2	388.62	10.58
486	5.69175	0.0	18.10	0.0	0.583	6.114	79.8	3.5459	24.0	666.0	20.2	392.68	14.98
487	4.83567	0.0	18.10	0.0	0.583	5.905	53.2	3.1523	24.0	666.0	20.2	388.22	11.45
488	0.15086	0.0	27.74	0.0	0.609	5.454	92.7	1.8209	4.0	711.0	20.1	395.09	18.06
489	0.18337	0.0	27.74	0.0	0.609	5.414	98.3	1.7554	4.0	711.0	20.1	344.05	23.97
490	0.20746	0.0	27.74	0.0	0.609	5.093	98.0	1.8226	4.0	711.0	20.1	318.43	29.68
491	0.10574	0.0	27.74	0.0	0.609	5.983	98.8	1.8681	4.0	711.0	20.1	390.11	18.07
492	0.11132	0.0	27.74	0.0	0.609	5.983	83.5	2.1099	4.0	711.0	20.1	396.90	13.35
493	0.17331	0.0	9.69	0.0	0.585	5.707	54.0	2.3817	6.0	391.0	19.2	396.90	12.01
494	0.27957	0.0	9.69	0.0	0.585	5.926	42.6	2.3817	6.0	391.0	19.2	396.90	13.59
495	0.17899	0.0	9.69	0.0	0.585	5.670	28.8	2.7986	6.0	391.0	19.2	393.29	17.60
496	0.28960	0.0	9.69	0.0	0.585	5.390	72.9	2.7986	6.0	391.0	19.2	396.90	21.14
497	0.26838	0.0	9.69	0.0	0.585	5.794	70.6	2.8927	6.0	391.0	19.2	396.90	14.10
498	0.23912	0.0	9.69	0.0	0.585	6.019	65.3	2.4091	6.0	391.0	19.2	396.90	12.92
499	0.17783	0.0	9.69	0.0	0.585	5.569	73.5	2.3999	6.0	391.0	19.2	395.77	15.10
500	0.22438	0.0	9.69	0.0	0.585	6.027	79.7	2.4982	6.0	391.0	19.2	396.90	14.33
501	0.06263	0.0	11.93	0.0	0.573	6.593	69.1	2.4786	1.0	273.0	21.0	391.99	9.67
502	0.04527	0.0	11.93	0.0	0.573	6.120	76.7	2.2875	1.0	273.0	21.0	396.90	9.08
503	0.06076	0.0	11.93	0.0	0.573	6.976	91.0	2.1675	1.0	273.0	21.0	396.90	5.64

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
504	0.10959	0.0	11.93	0.0	0.573	6.794	89.3	2.3889	1.0	273.0	21.0	393.45	6.48
505	0.04741	0.0	11.93	0.0	0.573	6.030	80.8	2.5050	1.0	273.0	21.0	396.90	7.88

506 rows × 13 columns

In [151]:

```
training_data = dataframe[['RM', 'LSTAT']]
```

In [297]:

```
import numpy as np
from sklearn.datasets import load_boston
from sklearn.utils import shuffle, resample
#from miniflow import *

# Load data
data = load_boston()
X_ = training_data
y_ = data['target']

# Normalize data
X_ = (X_ - np.mean(X_, axis=0)) / np.std(X_, axis=0)

n_features = X_.shape[1]
n_hidden = 10
W1_ = np.random.randn(n_features, n_hidden)
b1_ = np.zeros(n_hidden)
W2_ = np.random.randn(n_hidden, 1)
b2_ = np.zeros(1)

# Neural network
X, y = Placeholder(), Placeholder()
W1, b1 = Placeholder(), Placeholder()
W2, b2 = Placeholder(), Placeholder()

l1 = Linear(X, W1, b1)
s1 = Sigmoid(l1)
l2 = Linear(s1, W2, b2)
cost = MSE(y, l2)

feed_dict = {
    X: X_,
    y: y_,
    W1: W1_,
    b1: b1_,
    W2: W2_,
    b2: b2_
```

```
}
```

```
epochs = 200
# Total number of examples
m = X_.shape[0]
batch_size = 1
steps_per_epoch = m // batch_size

graph = topological_sort_feed_dict(feed_dict)
trainables = [W1, b1, W2, b2]

print("Total number of examples = {}".format(m))
```

Total number of examples = 506

In [298]:

```
losses = []

for i in range(epochs):
    loss = 0
    for j in range(steps_per_epoch):
        # Step 1
        # Randomly sample a batch of examples
        X_batch, y_batch = resample(X_, y_, n_samples=batch_size)

        # Reset value of X and y Inputs
        X.value = X_batch
        y.value = y_batch

        # Step 2

        forward_and_backward(graph) # set output node not important.

        # Step 3
        rate = 1e-2

        optimize(trainables, rate)

        loss += graph[-1].value

    if i % 100 == 0:
        print("Epoch: {}, Loss: {:.3f}".format(i+1, loss/steps_per_epoch))
        losses.append(loss/steps_per_epoch)
```

Epoch: 1, Loss: 32.281

Epoch: 101, Loss: 16.730

In [300]:

```
X.value = np.array([[6, 8]])  
forward_and_backward(graph)  
graph[-2].value[0][0]
```

Out[300]:

13.185566465195175

In [301]:

```
from mpl_toolkits.mplot3d import Axes3D
```

In [302]:

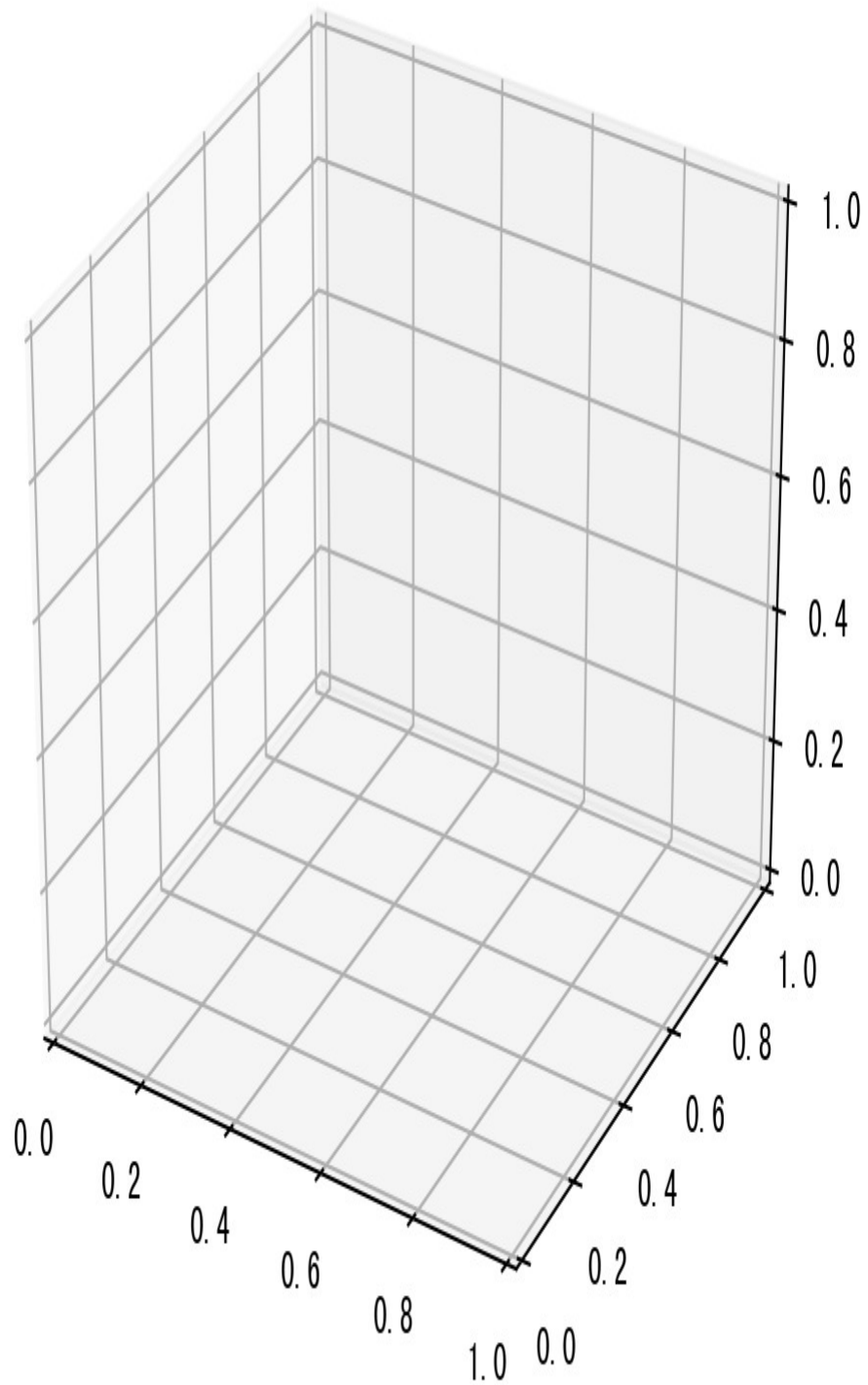
```
import matplotlib.pyplot as plt
```

In [303]:

```
from matplotlib.ticker import LinearLocator, FormatStrFormatter
```

In [304]:

```
fig = plt.figure()
```



In [305]:

```
ax = fig.gca(projection='3d')
```

In [288]:

```
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
from matplotlib import cm
from matplotlib.ticker import LinearLocator, FormatStrFormatter
import numpy as np
```

```
fig = plt.figure(figsize=(10, 10))
ax = fig.add_subplot(111, projection='3d')
```

```
# Make data.
```

```
X = X_.values[:, 0]
```

```
Y = X_.values[:, 1]
```

```
Z = y_
```

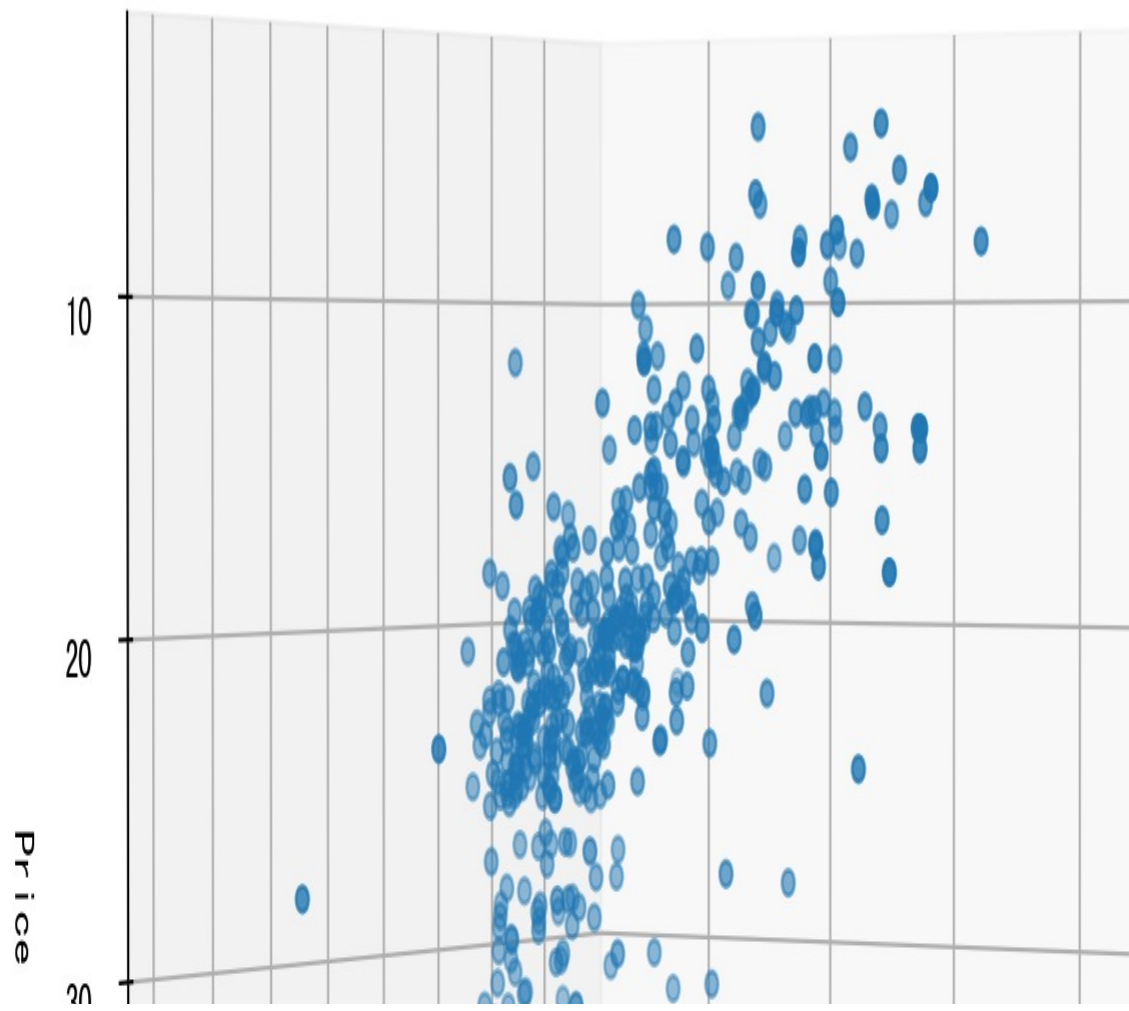
```
# Plot the surface.
```

```
rm_and_lstp_price = ax.scatter(X, Y, Z)
```

```
ax.set_xlabel('RM')
```

```
ax.set_ylabel('% of lower state')
```

```
ax.set_zlabel('Price')
```



Out[288]:

```
Text(0.5, 0, 'Price')
```

In [251]:

```
training_data.values
```

Out[251]:

```
array([[6.575, 4.98 ],
       [6.421, 9.14 ],
       [7.185, 4.03 ],
       ...,
       [6.976, 5.64 ],
       [6.794, 6.48 ],
       [6.03 , 7.88 ]])
```

In [266]:

```
predicate_result = []
for rm, ls in training_data.values:
    X.value = np.array([[rm, ls]])
    forward_and_backward(graph)
    predicate_result.append(graph[-2].value[0][0])
```

In [269]:

```
predicate_result = np.array(predicate_result)
```

In [286]:

```
%matplotlib notebook
```

In []:

In [306]:

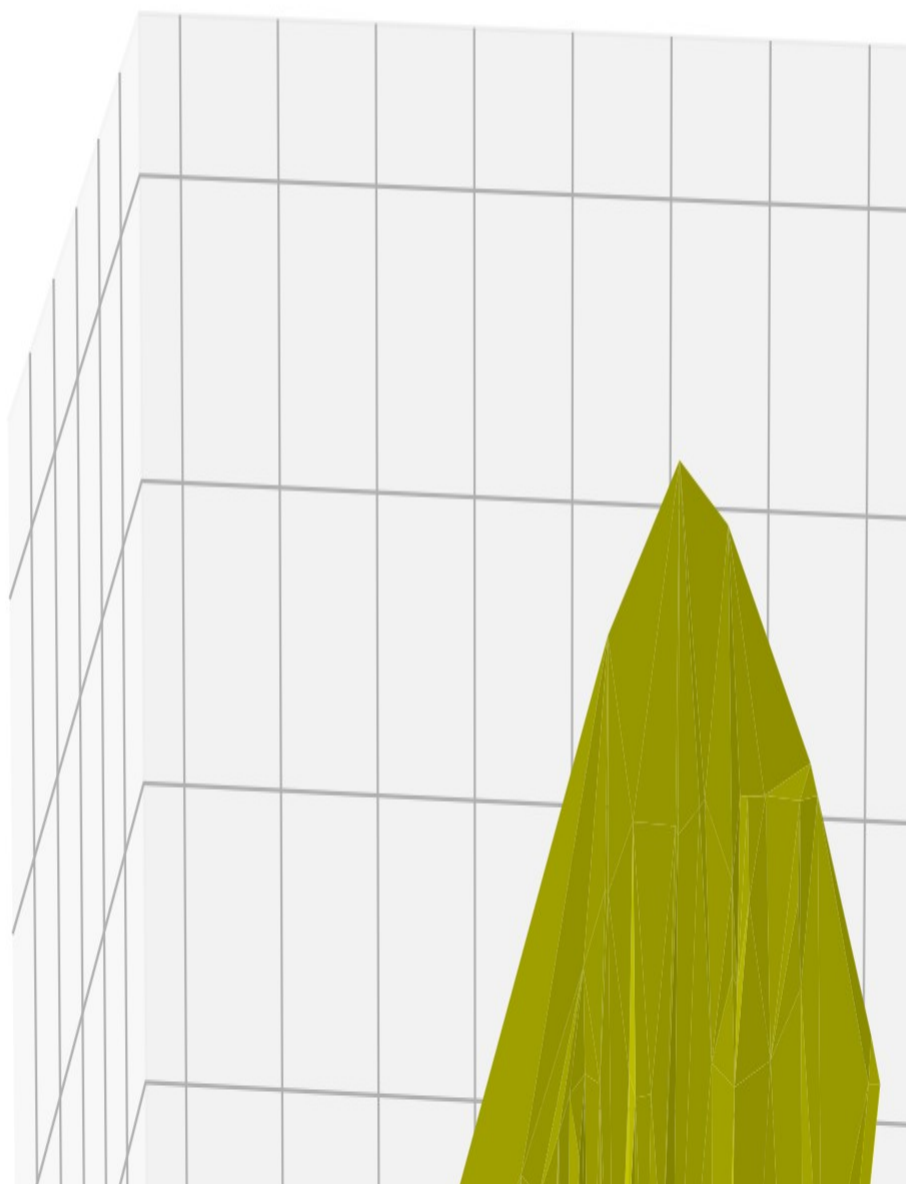
```
fig = plt.figure(figsize=(10, 10))
ax = fig.add_subplot(111, projection='3d')

# Make data.
X = X_.values[:, 0]
Y = X_.values[:, 1]
Z = predicate_result

# Plot the surface.
rm_and_lstp_price = ax.plot_trisurf(X, Y, Z, color='yellow')

ax.set_xlabel('RM')
ax.set_ylabel('% of lower state')
```

```
ax.set_zlabel('Predicated-Price')
```

Out[306]:

Text(0.5, 0, 'Predicated-Price')