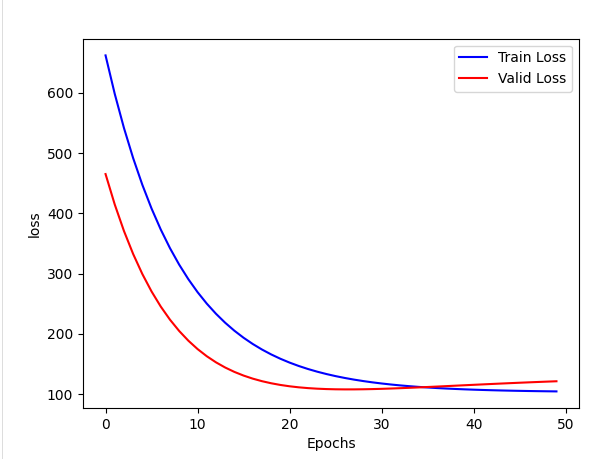
## #1 原始数据

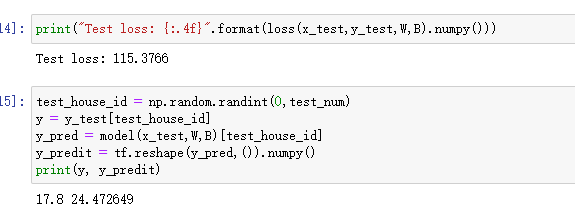
training\_epochs = 50

learning\_rate = 0.001

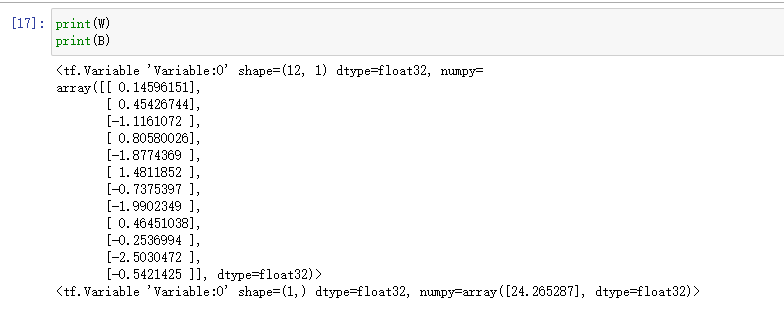
batch\_size = 10

**

*损失值和预测值*

**

*WB 值*

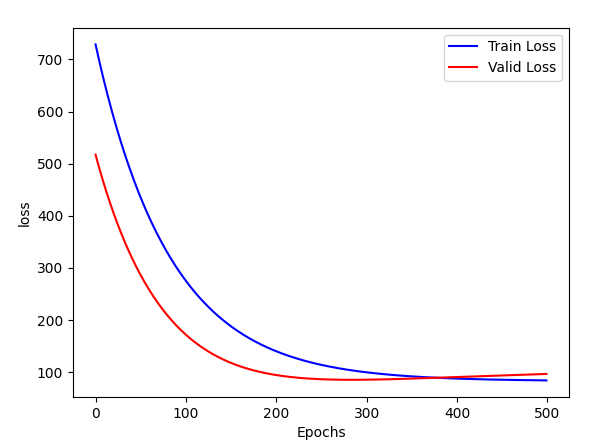
**

## #2 增加batch

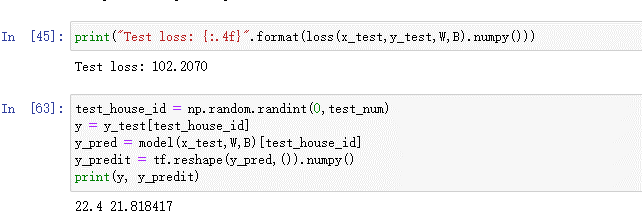
training\_epochs = 500

learning\_rate = 0.001

batch\_size = 100



*损失值和预测值*



W和B值

<tf.Variable 'Variable:0' shape=(12, 1) dtype=float32, numpy=

array([[-0.48623094],

[ 0.73265284],

[ 0.23815054],

[ 0.7604567 ],

[ 0.14206278],

[ 0.7644955 ],

[-0.8743601 ],

[-0.54762083],

[ 1.1886998 ],

[-0.54201096],

[-0.5693404 ],

[ 0.6463582 ]], dtype=float32)>

<tf.Variable 'Variable:0' shape=(1,) dtype=float32, numpy=array([24.229465], dtype=float32)>

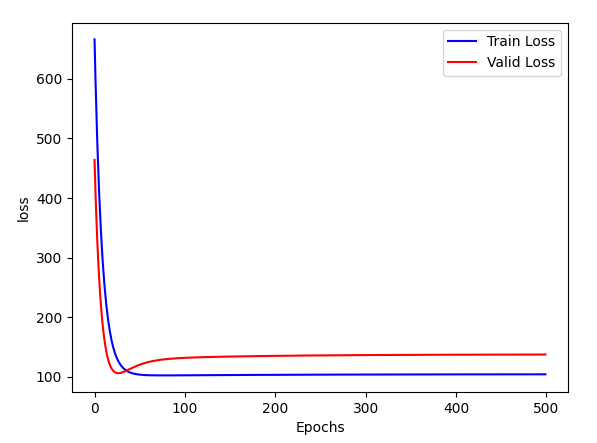
可以看到表现比前者要好，但训练时间较前者时间长

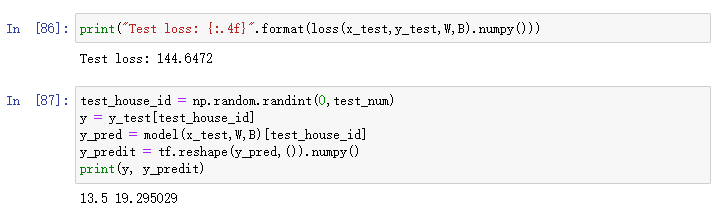
## #3 增加batch为整个集合

training\_epochs = 50

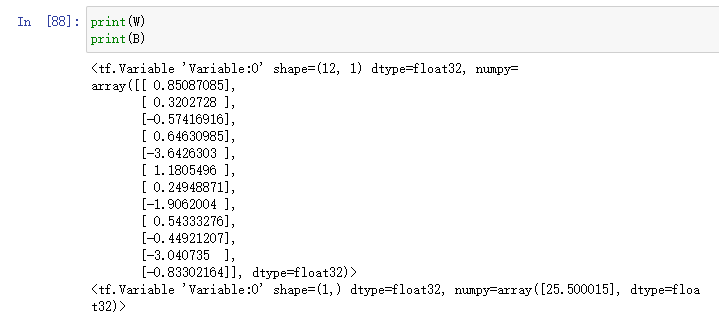
learning\_rate = 0.001

batch\_size = 506





W和B

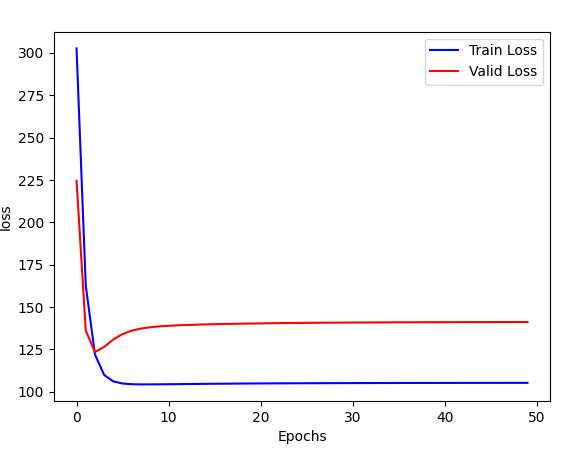


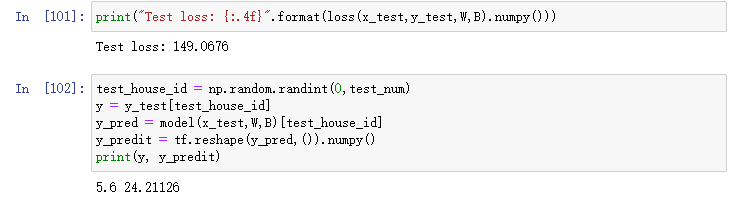
## #4 减少rate

training\_epochs = 50

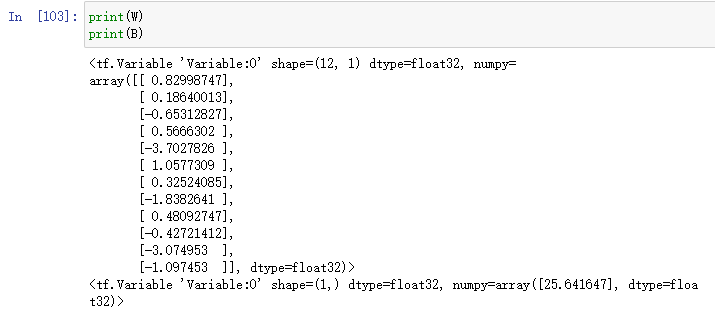
learning\_rate = 0.01

batch\_size = 10





W和B



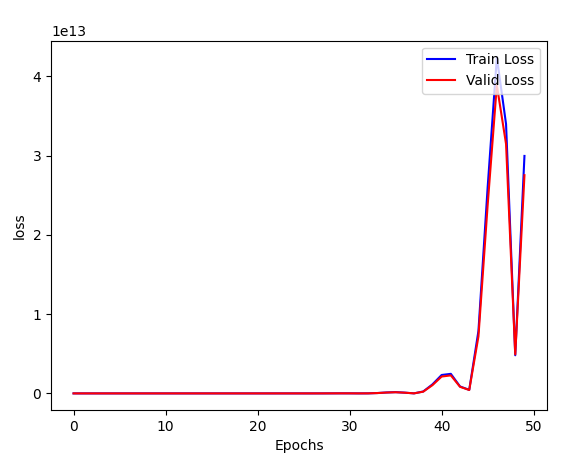
、可以看到预测值偏差较大

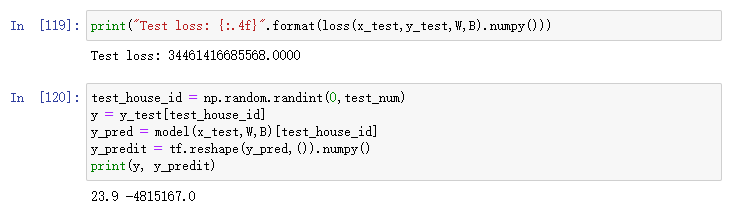
## #5 再次减少rate

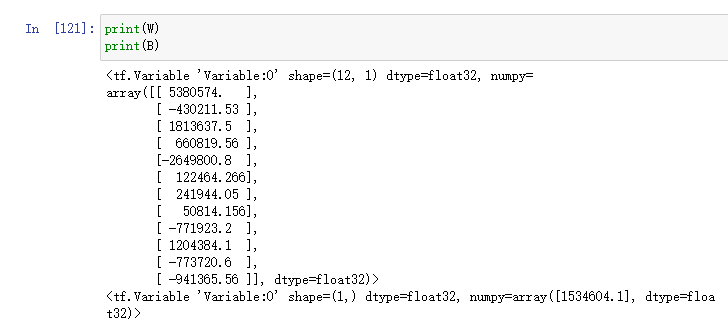
training\_epochs = 50

learning\_rate = 0.1

batch\_size = 10







可以看到数据非常离谱了，因为学习率太大了！