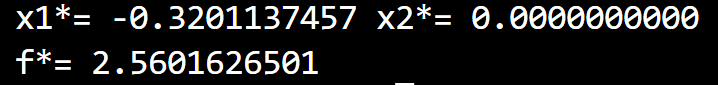
**运行截图：**



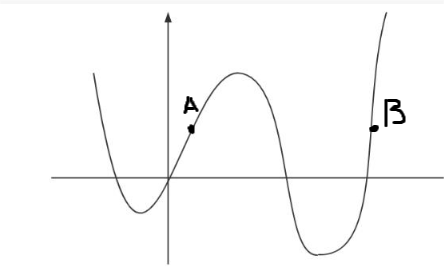
**思考**

1. **非凸函数**

对于非凸函数，局部最优解不一定是全局最优解，因为可能有多个局部最优解，程序在找到一个局部最优解之后导数变为零，解值无法移动，因此不能继续寻找下一个局部最优解。

1. **初始值的影响**

初始点对非凸函数极值点的求解影响非常大，因为不同的初始点可能对应着不同的局部最优解，如下图所示



若初始点为A最终结果只能到达左边的极值，而初始点为B时可以求得右边的极值。初始点的不同直接影响了最终答案求解的好坏。

1. **初始点设置方法**
   1. 随机初始化
   2. 基于先验知识初始化

根据函数的特性或者数据的分布情况，可以提前估计一个较好的初始点作为起点

* 1. 采样平均初始化

将数据按照某种方式分成若干个子集，对每个子集进行优化，最后取所有结果的平均值作为初始点。

1. **步长的影响**

如果步长过大，每次权重更新时会产生很大的波动，这可能导致模型无法收敛甚至发散，也可能会使得模型错失最优解而停留在一个较差的局部最小值。如果步长过小，模型收敛速度会变慢，需要更多的迭代才能达到收敛状态，也可能会使得模型卡在一个较差的局部最小值，因为它不能够足够快地跳出这个局部最小值

1. **手动条件检验结果好坏**
2. **缺点以及对应改进方案**
   1. 受到初始点选择的影响大：随机梯度下降是梯度下降的一种变体，它每次只使用一个样本计算梯度，并更新参数。相比于梯度下降，随机梯度下降具有更快的收敛速度和更好的泛化性能
   2. 对于非凸函数，陷入局部最优解：动量梯度下降是一种基于动量的优化方法，它可以加速收敛并减少震荡。在更新参数时，动量梯度下降不仅考虑当前时刻的梯度，还考虑之前的梯度方向和大小，从而使得参数更新更为平滑
   3. 选择何时的步长：自适应学习率算法可以根据梯度的大小自动调整学习率，常见的算法包括Adagrad、RMSProp和Adam等。这些算法能够在保证模型收敛的情况下，自适应地控制学习率的大小，从而提高了模型的性能