

法律声明

■ 本课件包括演示文稿、示例、代码、题库、视频和声音等内容，北风网和讲师拥有完全知识产权；只限于善意学习者在本课程使用，不得在课程范围外向任何第三方散播。任何其他人或者机构不得盗版、复制、仿造其中的创意和内容，我们保留一切通过法律手段追究违反者的权利。

■ 课程详情请咨询

◆ 微信公众号：北风教育

◆ 官方网址：<http://www.ibeifeng.com/>



人工智能之机器学习

晚自习

主讲人：Gerry

上海育创网络科技有限公司



课程要求

■ 课上课下 “九字” 真言

- ◆ 认真听，善摘录，勤思考
- ◆ **多温故，乐实践**，再发散

■ 四不原则

- ◆ **不懒散惰性，不迟到早退**
- ◆ **不请假旷课，不拖延作业**

■ 一点注意事项

- ◆ 违反 “四不原则” ， 不包就业和推荐就业

严格是大爱



寄语



做别人不愿做的事，
做别人不敢做的事，
做别人做不到的事。

课程内容

- MLE最大似然估计
- 梯度下降再次讲解

梯度下降作业

梯度下降作业：

目标函数：

$$y = x^2 + b * x + c$$

需求：求解最小值对应的x和y

要去：写代码

数据：

b：服从均值为-1，方差为10的随机数

c：服从均值为0，方差为1的随机数

假定a、b、c这样的数据组合总共10、100、10w、100w条数据，求解在现在的数据情况下，目标函数的取最小值的时候，x和y分别对应多少？

梯度下降作业 $y = f(x) = x^2$

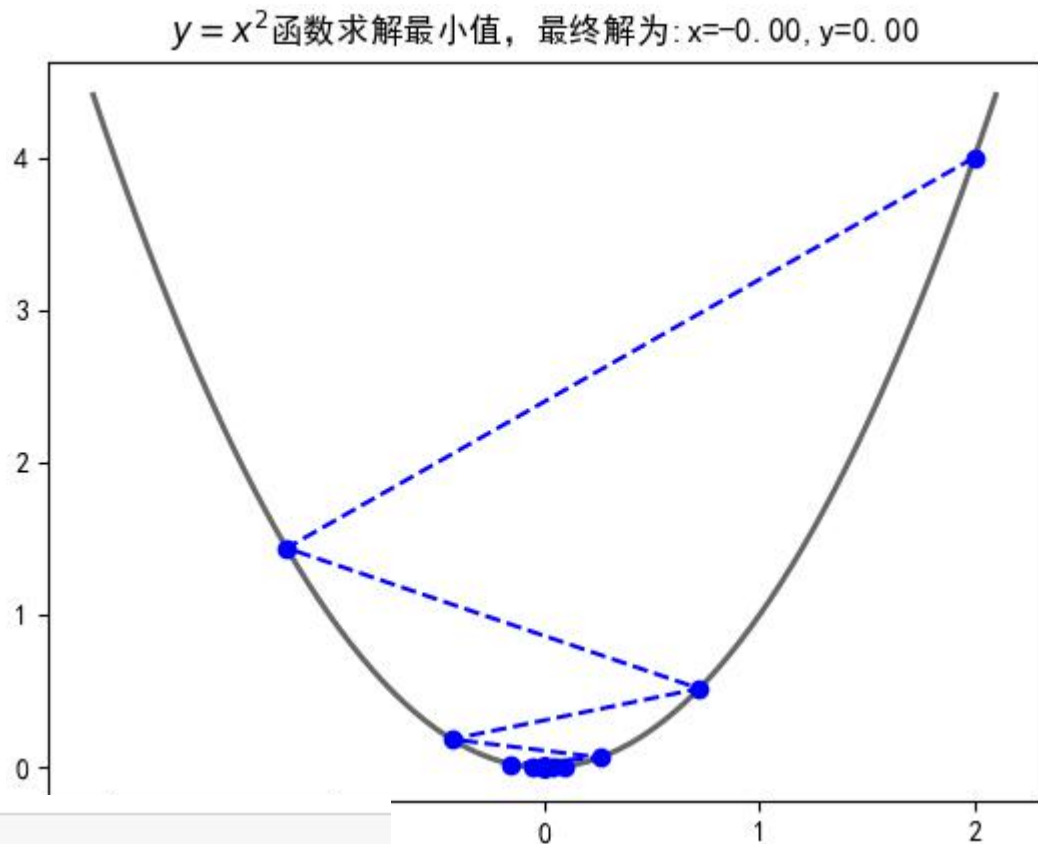
```
## 原函数
def f(x):
    return x ** 2
## 导数
def h(x):
    return 2 * x

X = []
Y = []

x = 2
step = 0.8
f_change = f(x)
f_current = f(x)
X.append(x)
Y.append(f_current)
while f_change > 1e-10:
    x = x - step * h(x)
    tmp = f(x)
    f_change = np.abs(f_current - tmp)
    f_current = tmp
    X.append(x)
    Y.append(f_current)
```

```
fig = plt.figure()
X2 = np.arange(-2.1, 2.15, 0.05)
Y2 = X2 ** 2

plt.plot(X2, Y2, '-', color='#666666', linewidth=2)
plt.plot(X, Y, 'bo—')
plt.title(u'$y=x^2$函数求解最小值, 最终解为: x=%.2f, y=%.2f'
        % (x, f_current))
plt.show()
```



梯度下降案例

$$z = f(x, y) = x^2 + y^2$$

梯度下降法求解，最终解为：x=0.00, y=0.00, z=0.00

```
## 原函数
def f(x, y):
    return x ** 2 + y ** 2
## 偏函数
def h(t):
    return 2 * t

X = []
Y = []
Z = []

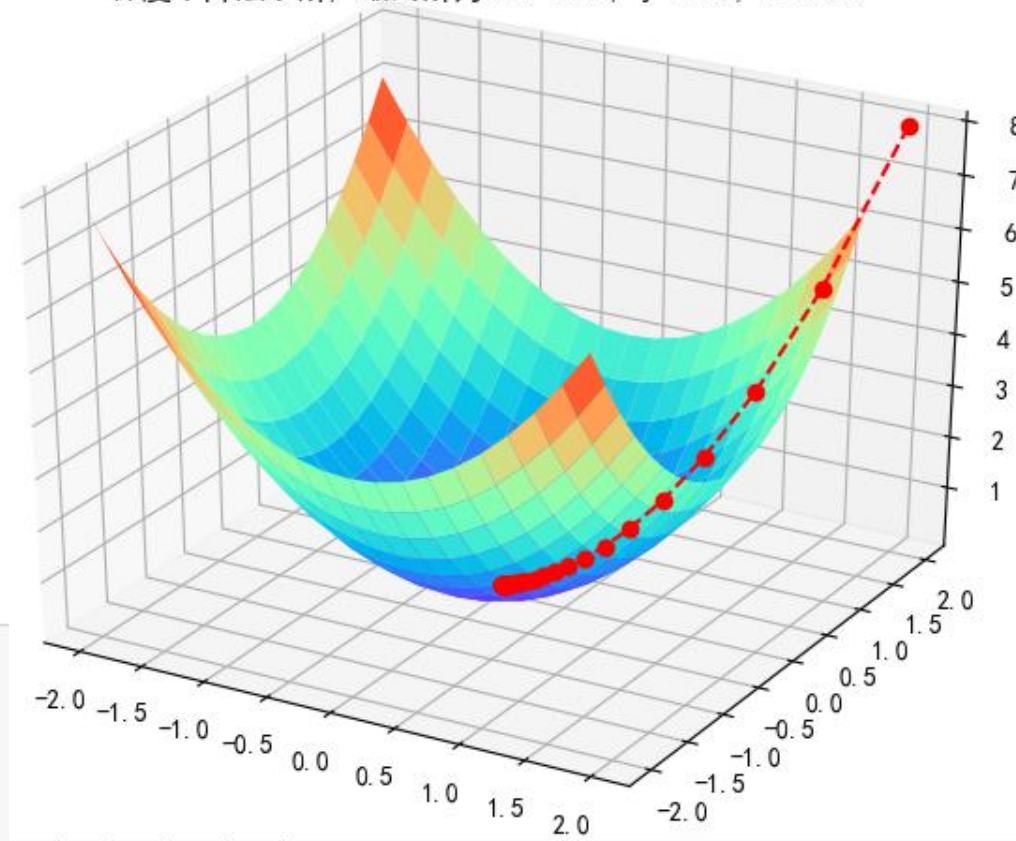
x = 2
y = 2
f_change = x ** 2 + y ** 2
f_current = f(x, y)
step = 0.1
X.append(x)
Y.append(y)
Z.append(f_current)
while f_change > 1e-10:
    x = x - step * h(x)
    y = y - step * h(y)
    f_change = f_current - f(x, y)
    f_current = f(x, y)
    X.append(x)
    Y.append(y)
    Z.append(f_current)
```

```
fig = plt.figure()
ax = Axes3D(fig)
X2 = np.arange(-2, 2, 0.2)
Y2 = np.arange(-2, 2, 0.2)
X2, Y2 = np.meshgrid(X2, Y2)
Z2 = X2 ** 2 + Y2 ** 2

ax.plot_surface(X2, Y2, Z2, rstride=1, cstride=1, cmap='rainbow')
ax.plot(X, Y, Z, 'ro—')

ax.set_title(u'梯度下降法求解，最终解为：x=%.2f, y=%.2f, z=%.2f' % (x, y, f_current))

plt.show()
```

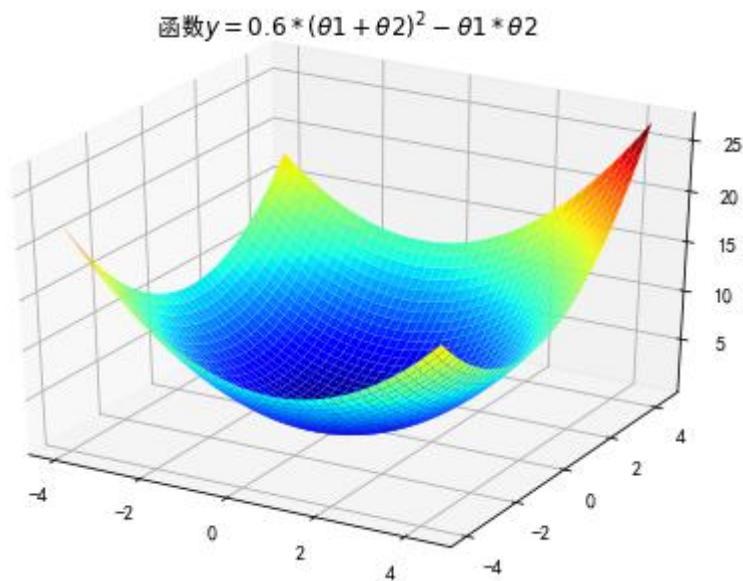
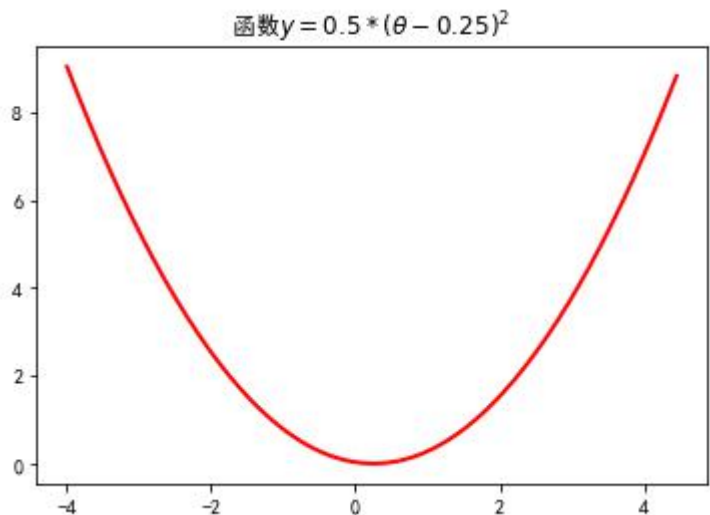


梯度下降法

- 梯度下降法的优化思想是用当前位置负梯度方向作为搜索方向，因为该方向为当前位置的最快下降方向，所以梯度下降法也被称为“最速下降法”。梯度下降法中越接近目标值，变量变化越小。计算公式如下：

$$\theta^{k+1} = \theta^k - \alpha \nabla f(\theta^k)$$

- α 被称为**步长**或者**学习率(learning rate)**，表示自变量x每次迭代变化的大小。
- 收敛条件：当目标函数的函数值变化非常小的时候或者达到最大迭代次数的时候，就结束循环。





THANK YOU

上海育创网络科技有限公司