实验 1: 线性回归

介绍

在本练习中，你将实现线性回归，并看到它在数据上的作用。

用到的数据

1. ex1data1.txt-用于单变量线性回归的数据集
2. ex2data2.txt-用于多变量线性回归的数据集

1. 单变量线性回归

在本练习的这一部分中，假设你是一家特许经营餐厅的首席执行官，并且正在考虑在不同的城市开设一家新的特许经营餐厅。该连锁店已经在不同的城市有了餐车，你有各个城市的人口和利润数据。

在文件 ex1data1.txt 中包含了我们本次线性回归实验的数据集，第一列为一个城市的人口数量，第二列为餐车在对应城市获得的利润。

1.1 读取数据

首先你需要做的是将 ex1data1.txt 文件中的数据进行读取，使用的方法是 numpy.loadtxt()，读取完成后将所读数据的规模和维数进行打印。

```
ex1data1 = './ex1data1.txt'
data1 = np.loadtxt(ex1data1,delimiter=',')
print(data1.shape, data1.ndim)
```

✓ 0.3s

1.2 可视化数据

对数据进行可视化有助于更好的理解数据集的分布，对于本次实验的数据，可以通过绘制散点图进行可视化，因为数据只有两个特征（人口数量、利润）。画图需要使用 matplotlib 库中的相关函数。

```
x=data1[:,0]
y=data1[:,1]
print(x.shape, x.ndim)
```

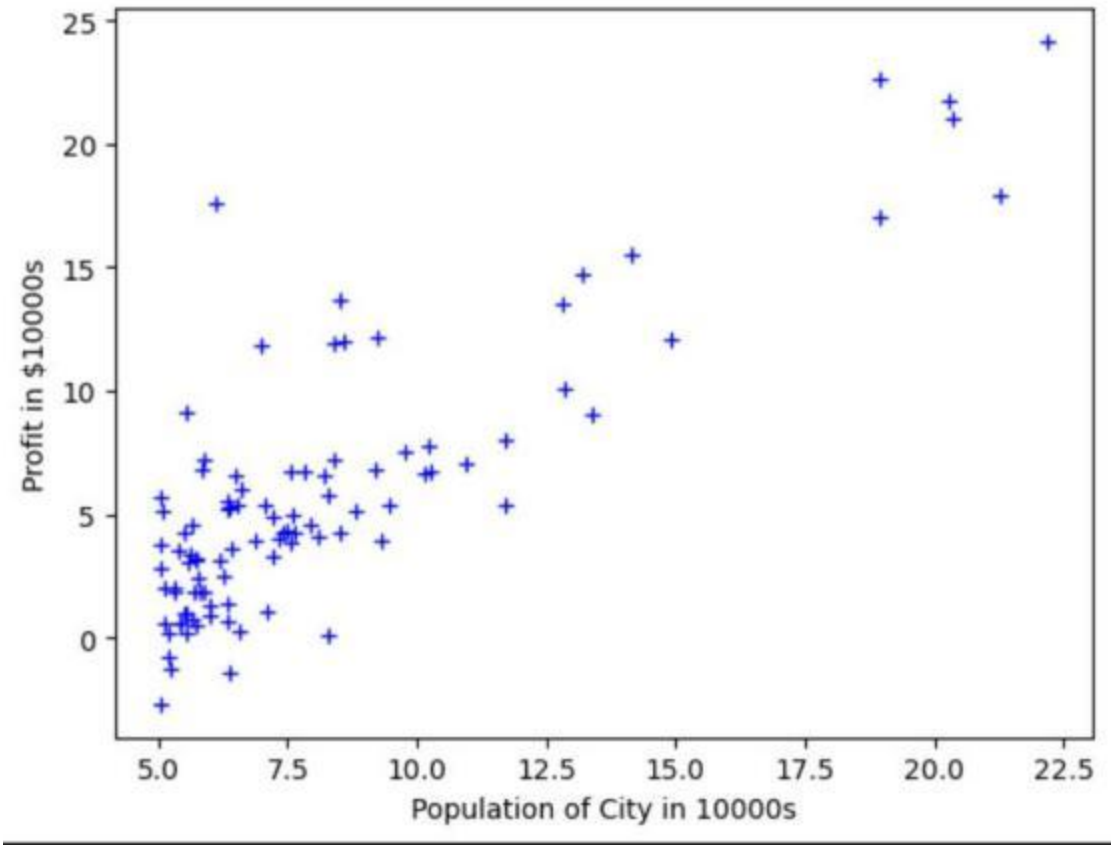
✓ 0.3s

```
(97,) 1
```

```
#使用plt.xlabel plt.ylabel plt.plot函数来进行数据可视化
plt.xlabel('Population of City in 10000s')
plt.ylabel('Profit in $10000s')
plt.plot(x, y, '+b')
```

✓ 0.4s

数据可视化的效果如下图所示。



1.3 训练线性回归模型

该部分你要完成输入数据的准备、损失函数的编写、梯度下降法求模型参数、解析法直接计算最优解。

代价函数公式：

代价函数(或损失函数)：

$$J(\theta_0,\theta_1)=\frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)})-y^{(i)}\right)^2$$

$$J(\boldsymbol{\theta})=\frac{1}{2m}\sum_{i=1}^{m}(h_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})-y^{(i)})^2$$

将上面的式子改成矩阵形式，即

$$J(\theta)=\frac{1}{2m}(\theta X-Y)^2$$

这里，θ 是一个向量，X 与 Y 则是两个矩阵。

梯度下降法公式：

Gradient Descent Algorithm

Initialize θ_0, θ_1

repeat until convergence {

$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)$$

$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right) \cdot x^{(i)}$$

} update θ_0 and θ_1 simultaneously

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

与计算损失函数相同，将上述式子改写成矩阵形式，即

$$\theta_j := \theta_j - \alpha \frac{1}{m} (\theta X - Y) X$$

与上相同，
numpy.matmul(X,(numpy.matmul(X,θ)-Y))

解析求导法公式：

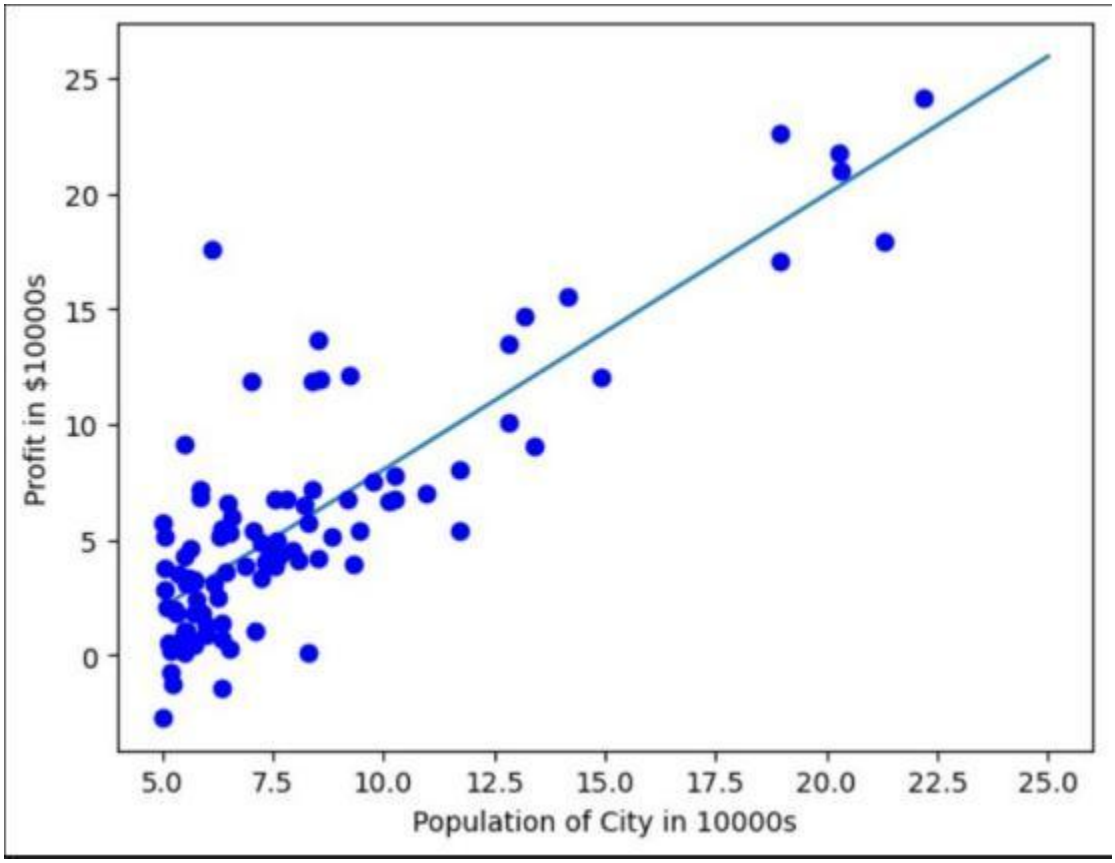
$$\boldsymbol{\theta} = (X^{\mathrm{T}} X)^{-1} X^{\mathrm{T}} \boldsymbol{y}$$

1.4 使用训练得到的模型进行预测并可视化结果

```
x0=5.0
y0=theta[0]+x0*theta[1]
x1=25.0
y1=theta[0]+x1*theta[1]
print(y0, y1)
```

✓ 0.3s

```
2.069387342636121 25.93006022642799
```



2. 多变量线性回归

在这部分中，你需要使用多元变量的线性回归来预测房价。假设你正在卖掉你的房子，你想知道一个好的市场价格。其中一种方法是首先收集最近出售的房屋的信息，并建立一个房价模型。ex1data2.txt 文件包含了俄勒冈州波特兰市的房 价培训集。第一列是房子的大小（以平方英尺），第二列是卧室的数量，第三列是房子的价格。

参考单变量线性回归的代码，完成该任务。