Homework 1：线性回归模型

522030910135 陈元杰

1. 数据集构建

我们先生成一系列输入特征为房屋面积和房龄，标签为房屋售价的数据。我们以线性函数加上随机噪声的方法来生成训练和测试数据，具体方法如下：

假设房屋面积为 x（单位为千平方米），房龄为 y（单位为十年）；按照常识，房屋面积 $x \in [0.05, 1.05]$，而房龄 $y \in [0.1, 3.1]$。按照一般规律，房屋售价与房屋面积成正相关而与房龄成负相关，因此我们以下面的经验公式来生成房屋售价 z.

$$z = 2x - y + 4 + 0.1\epsilon$$

在上面的公式中 $\epsilon \sim N(0, 1)$，是一个服从标准正态分布的高斯噪声。下图清晰的显示了我们生成的数据房屋售价与房屋面积和房龄的关系。

图 1: 房屋售价

2. 线性模型搭建

我们利用线性回归的模型来训练拟合训练集的数据。在训练过程中，首先通过构建一个线性函数作为需要拟合的预测函数

$$h(x) = \sum_{i=0}^{d} \theta_i x_i = \theta^T x$$

在上面的公式中 d 是数据特征的维度（第 0 维是 1，引入偏置项）。然后计算与真实的数据标签的损失函数，这里我们采用的是平方损失

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

在上式中，n 是训练样本的数量。我们的目标是使损失函数最小，可以采用梯度下降方法来不断更新权重等参数。

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

计算损失函数的梯度可以得到

$$\begin{aligned}
\frac{\partial}{\partial \theta_j} J(\theta) &= \frac{\partial}{\partial \theta_j} \frac{1}{2} \left(h_\theta(x) - y\right)^2 \\
&= 2 \cdot \frac{1}{2} \left(h_\theta(x) - y\right) \cdot \frac{\partial}{\partial \theta_j} \left(h_\theta(x) - y\right) \\
&= \left(h_\theta(x) - y\right) \cdot \frac{\partial}{\partial \theta_j} \left(\sum_{i=0}^{d} \theta_i x_i - y \right) \\
&= \left(h_\theta(x) - y\right) x_j
\end{aligned}$$

最终可以得到参数更新的公式如下

$$\theta_j := \theta_j + \alpha \left(y^{(i)} - h_\theta(x^{(i)})\right) x_j^{(i)}$$

按照这个方式一直更新参数直至收敛就可以训练得到线性回归模型。线性回归算法的伪代码如下所示：

Algorithm 1 Linear Regression Using Gradient Descent

Require: Training set $\{(x^{(i)}, y^{(i)})\}_{i=1}^{m}$, learning rate α, number of iterations num_iters

Ensure: Parameters θ

1: Initialize $\theta := 0$ {Or small random values}
2: **for** iter $= 1$ to num_iters **do**
3: **for** $j = 0$ to n **do**
4: Compute the gradient for θ_j:

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right) x_j^{(i)}$$

5: Update θ_j:

$$\theta_j := \theta_j - \alpha \frac{\partial J(\theta)}{\partial \theta_j}$$

6: **end for**
7: **end for**
8: **return** θ

3. 实验结果与分析

在实际实验中，我们搭建一层线性层神经网络作为线性回归的权重。生成 700 个数据作为训练集，300 个数据作为测试集。将 MSE loss 作为损失函数，利用随机梯度下降，以学习率为 0.1 训练 1000 个 epoch。做出训练时的 loss 值随训练轮数的变化曲线，如下图所示：

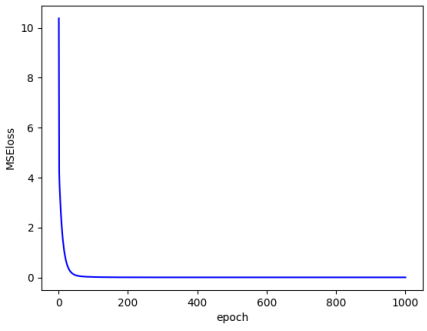

图 2: *loss* 变化图

可见模型很快达到收敛。对于训练好的模型，将它应用在测试集上进行测试。对于测试集中的每一个样本，我们做出它对应的真实的房屋售价和预测价格，如下图所示：

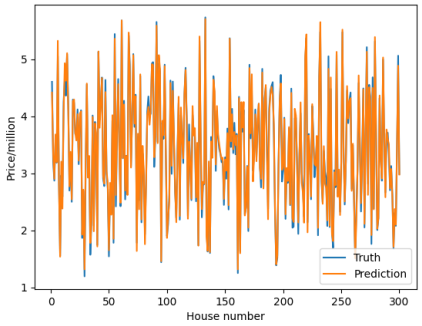

图 3: 预测效果图

从图中可以看出，我们模型的预测值与真实值非常接近。然后我们打印出线性层的权重，如下表所示：

表 1: 模型权重

W_1	W_2	b
1.977	-0.996	4.010

$W_1 \approx 2$; $W_2 \approx -1$; $b \approx 4$. 与我们生成数据时预设的参数十分接近，可见我们的模型取得了很好的效果。