**线性回归**

首先我们要了解插值、拟合、回归和预测的区别，虽然这几个概念比较类似，但是这几个方向的侧重点并不同，预测是一种笼统的说法，并非特指某一种方法。插值和拟合都是用离散数据确定一个最能体现离散数据的函数，插值要求每一个离散值都在函数上，而拟合并不要求。回归问题是比较常见的问题，主要用模型反应离散数据，并用这个模型预测和分析新的未知数据。下面是这几个概念的具体解释：

插值是指在离散数据的基础上插上连续函数，使每一个点都在这个连续函数上。

拟合：用一个连续函数使得离散数据都尽可能的靠近，它与插值的差别主要在它并不要求每一个离散数据都在函数上。

回归：主要是建立一个模型能够尽可能的反映离散数据的特征，并检验该模型的可信度。并用这一个模型预测和分析未知数据。

预测：这是一种非常笼统的说法，从某种角度上来说上述几种都可以说是预测。

总而言之，插值和拟合更多的是利用离散数据确定函数，回归更多的是利用离散数据建立模型，并用模型对未知数据进行预测和控制。

在回归问题中，需要用一个模型来反应离散数据的情况，在所有模型中，线性模型是最为简单的模型，有时候由于它的简单反而能够更好的反映数据，不产生过拟合现象。

举个例子，在一个有两个特征的问题中，拟合平面如下：



在这个拟合平面中，是偏置项，是权重项，扩展到n维特征中，这个拟合超平面是：

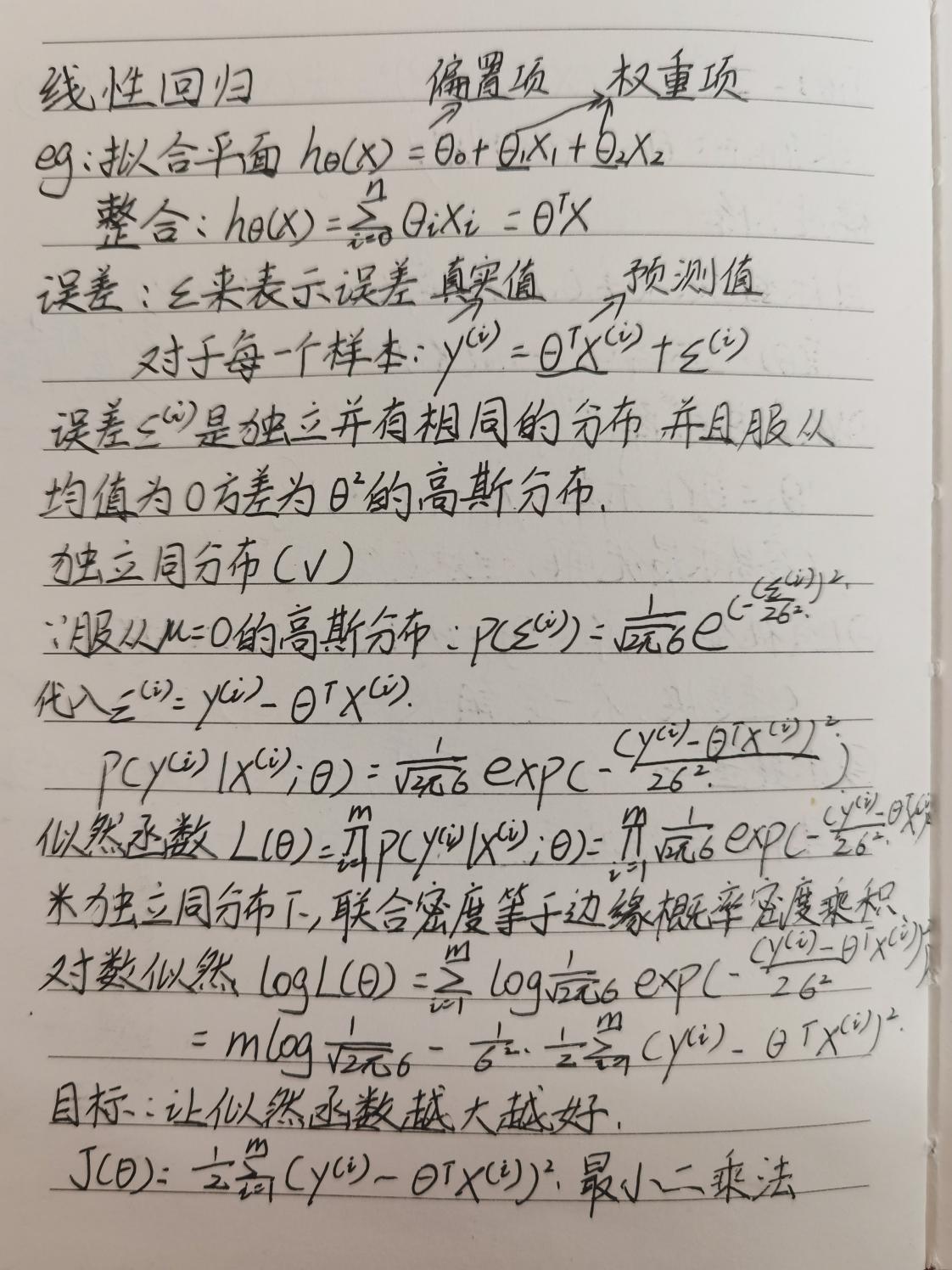


在所有机器学习问题中，大部分都是以矩阵形式进行运算的，在这个问题中，是参数矩阵，X是样本矩阵。

在这个问题中，我们现在的目标是求出最合适的值，对于这个问题我们可以直接使用最小二乘法来计算，得到的结果是：



当然真正推导过程中，预测值与真实值之间并不相等，两者之间还存在误差项，我们的目的就是使误差值最小，基本假设是误差服从高斯分布，并且误差项之间独立同分布。使用最大似然方法进行计算，最后能够得到与最小二乘法相同的式子。具体式子我就不写了，我在纸上写了，感兴趣的可以看看。



我们的目的是使得最小二乘法的式子最小，直接求解的方法是计算它的偏导数，当它的导数为0时，求得最小值。直接求导可以得到结果：



当然可以直接使用这个式子计算参数值，但是在实际运算中不会使用直接计算，原因无他，计算量太大，当特征数和样本数比较多的时候，计算量会大量增加。这时我们就可以使用优化算法求解一个不是最优的，但是足够好的参数值。这里使用的优化算法是梯度下降算法，如果后续有时间我会单独讲讲梯度下降算法。

这个时候我们的目标函数是：



梯度下降简而言之就是梯度按照梯度的方向进行迭代，该目标函数的梯度是：

，其中j代表第j个特征。

梯度下降的策略有以下几种：

1. 批量梯度下降：，每一次采用所有样本进行更新参数，容易求出最优解，但速度很慢。
2. 随机梯度下降：，每次都采用一个样本进行参数迭代，速度很快，但是不一定朝收敛方向迭代。
3. 小批量梯度下降：，每次选取其中一些进行迭代，其中代表学习率，对结果产生巨大影响，尽可能小一点。

接下来，就是代码实现：注释已经很详细了，就不一步一步解释了。