机器学习学习笔记

Project：机器学习

Author：胡小忠

Version：2019/5/6-1

前言

该文档产生自本人在学习Google深度学习框架TensorFlow过程后。主要内容包括机器学习基本概念、TensorFlow基本使用、神经网络基本概念等。

机器学习速成课程：

<https://developers.google.cn/machine-learning/>

公式使用TeX格式渲染。

机器学习简介

机器学习可以帮助我：

* 缩短编程时间。
* 定义自己的产品。
* 解决自己作为编程人员却不知道如何用人工方法解决的问题（提供足够的样本和合适的模型）。

框架处理

（监督式）机器学习：机器学习系统通过学习如何组合输入信息来对从未见过的数据做出有用的预测。

标签：要预测的对象，即简单线性回归中的y变量。

特征：输入变量，即简单线性回归中的x变量。复杂的机器学习可能使用数百万个特征，例如：。

样本：数据的特定实例：。可以为：无标签样本和有标签样本。使用有标签的样本训练模型，对无标签样本进行预测。

模型：特征与标签之间的关系。

回归模型：预测连续值。

分类模型：预测离散值。

深入了解机器学习

线性回归：。机器学习中为：。找到一种最适合一组点的直线或超平面的方法。

训练：让损失最小化。

损失：

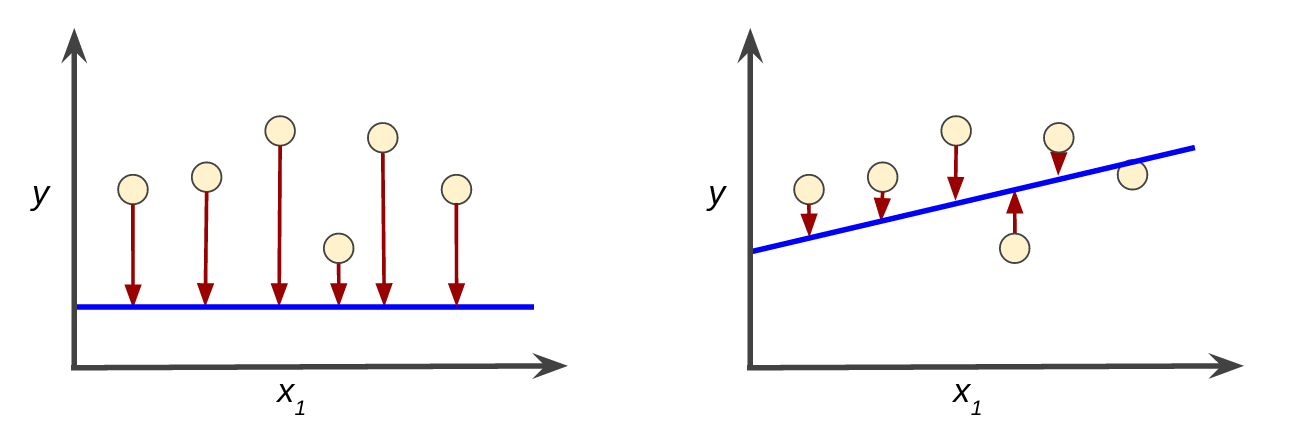


图 1 左侧损失大，右侧损失小

* 均方损失：，需要注意的是，该损失函数并不是唯一的损失函数，也不是适用于所有情况的损失函数。
* 平方损失：又称损失。即。

降低损失

迭代方法：

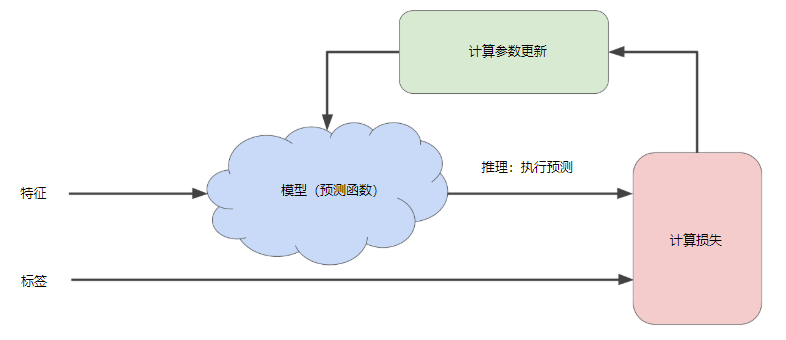


图 2 计算参数更新

梯度下降法：

梯度是一个矢量，因此具有两个重要特征：方向和大小。梯度始终指向损失函数中增长最为迅猛的方向。梯度下降算法会沿着负梯度的方向走，以便尽快降低损失。

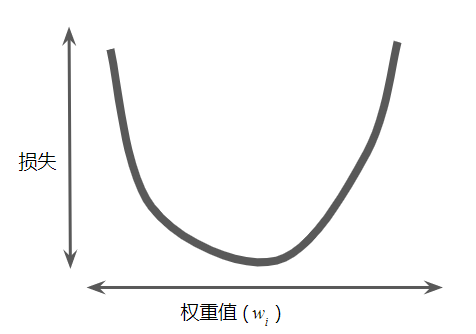


图 3 回归问题产生的损失与权重图为凸型

学习速率：

也就是梯度下降算法中每次下降的步长。过小，会收敛太慢。太大，会来回弹跳，不收敛。

* 一维空间中，最理想的学习速率是，也是对二阶导数的倒数。
* 二维或者多维空间中理想的学习速率是海森距离（由二阶偏导数构成的矩阵）的倒数。

优化学习速率：多次试用。

随机梯度下降法（SGD）：

每次迭代使用全批量不太现实，计算量太大，所示可以每次迭代只使用一个样本（批量大小为1）。或者使用折衷方案，小批量SGD，10-1000个随机选择的样本。**梯度下降法也同样适用于包含多个特征的特征集。**

使用TensorFlow基本步骤

|  |
| --- |
| 1. # Import TensorFlow package. 2. import tensorflow as tf 3. # Set up a linear classifier. 4. classifier = tf.estimator.LinearClassifier() 5. # Train the model on some example data. 6. classifier.train(input\_fn=train\_input\_fn, steps=2000) 7. # Use it to predict. 8. predictions = classifier.predict(input\_fn=predict\_input\_fn) |

代码 1 使用TensorFlow创建线性预测器并进行预测

具体参考：

书籍：TensorFlow实战Google深度学习框架（第二版）

文档：TensorFlow官网文档<https://tensorflow.google.cn/learn>

常用参数表：

|  |  |
| --- | --- |
| 名称 | 说明 |
| step | 训练迭代的总次数。一步计算一批样本产生的损失 |
| batch size | 单步的样本数量，随机选择。例如SGD的批次大小为1。 |
| periods | 控制报告的粒度，例如periods为7，steps为70，则每10步输出一次损失值。 |

泛化

训练集和测试集

验证

表示法

特征组合

正则化：简单性正则

逻辑回归

分类

正则化：稀疏性正则

神经网络简介

多类别神经网络

嵌入

静态训练与动态训练

静态推理与动态推理

数据依赖关系

应用准则