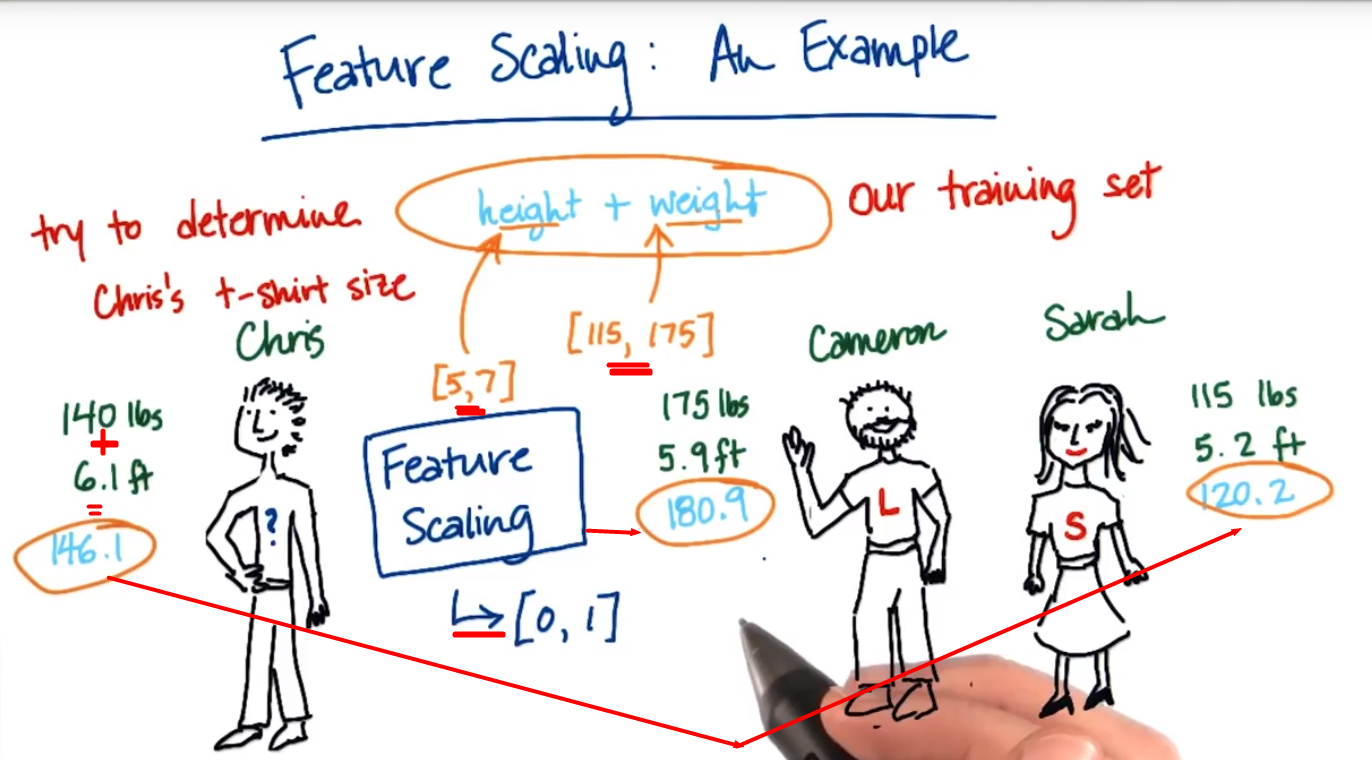
**特征缩放（Feature scaling）**

特征缩放是特征预处理的一项重要步骤。

# 1、Feature Scaling 案例



现在 问题在于 身高与体重两个特征的 度量标准非常不平衡

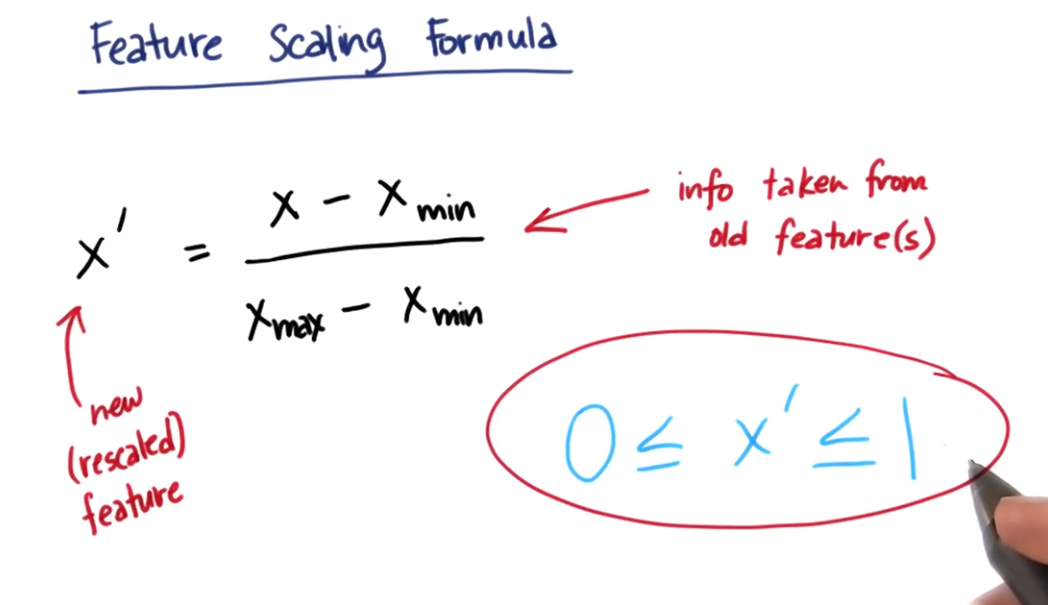
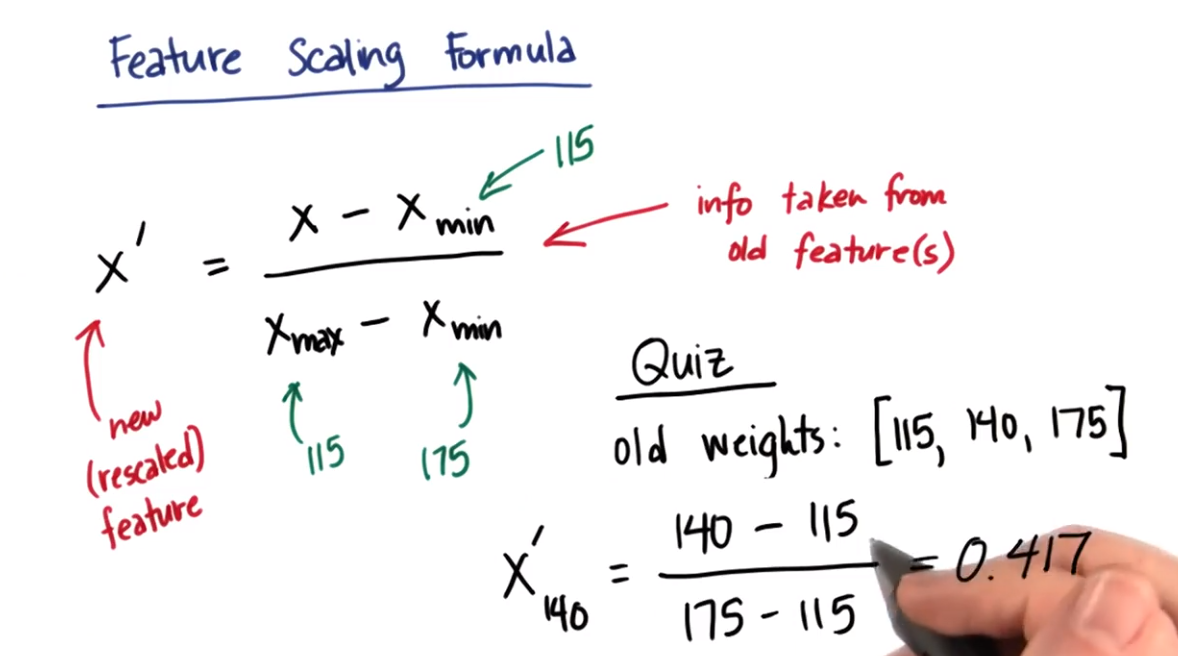
既身高的值，通常介于5和7之间

体重的值要大得多，是介于 115 和 175 之间

如果计算两种特征的合计 结果就是，体重几乎完全主导了你得到的答案，而身高结果成为舍入误差.

如果希望在进行合计运算时，两个特征能够得到同等的重视，那么特征缩放的方法，可以使这些特征 跨越 范围 具有可比性， 通常是介于零和一之间。特征缩放仍包含相同的信息，只是使用的单位不同，不会遗漏信息

# 2、Feature scaling公式及优缺点



也就是说 0.417就是重140磅的人所拥有的新特征值

Feature Scaling优缺点

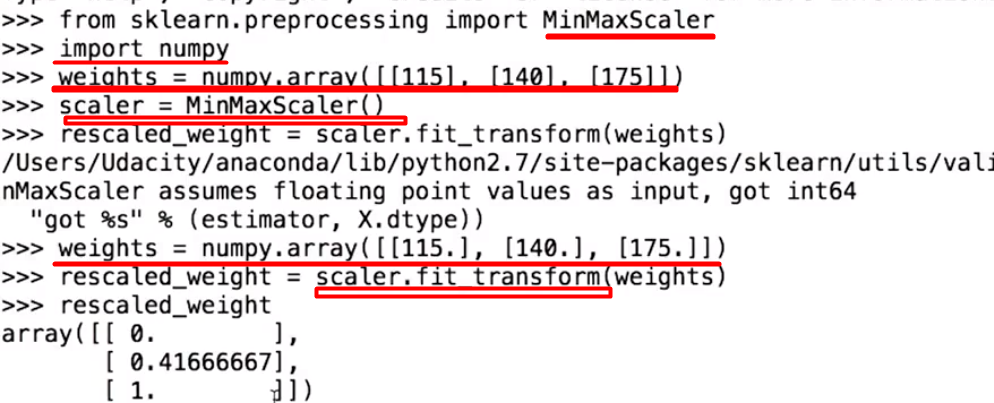
特征缩放公式的特点之一是,缩放后的特征值总是在0和1之间.

优点：预估输出相对稳定。

缺点：对异常值敏感，如果输入特征存在异常数值,那么特征缩放会比较棘手,因为 Xmin 和 Xmax 可能会是极端值的情况.

注意：当 Xmin 和 Xmax 相等的时候，要自己制定一个返回值。

# 3、Sckit-learn MinMaxScaler



# 4、特征缩放对算法的影响

## 1、受影响

受特征缩放影响的算法：支持向量机和 K-均值聚类

在计算距离时 你其实是在利用一个维度与另一个维度进行交换

举个例子 就支持向量机而言，它有一条将距离最大化的分割线，你在这里计算距离

而这个计算就是用一个维度与另一个维度进行交换，如果我们把某一点的增大至其他点的两倍 那么它的值也会扩大一倍

K-均值聚类也是这样的，你有一个集群中心，然后计算各数据点到集群中心的距离

该距离具有相同的特征，如果你将一个变量扩大一倍 它的值也会扩大一倍



## 2、不受影响

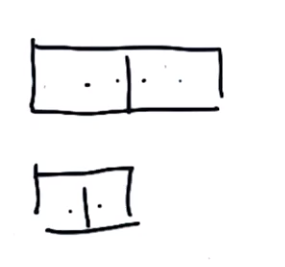
不受 特征缩放 影响的算法为：树和线性回归

### 1）决策树

决策树并不会呈现对角线 对吧？它会呈现出一系列的水平线和垂线，所以不存在两者的交换，它只是在不同的方向上进行切割。在处理某一维度时，我们不需要考虑另一个维度的情况。

如把这个方框缩小一半，就是对该特征进行缩放，图中线条的位置会有所变化

但分割的顺序是一样的，它是按比例分割的，所以 两个变量之间不存在交换



2）线性回归

每个特征都有一个相应的系数，这个系数总是与相应的特征同时出现，特征 A 的变化不会影响到，特征 B 的系数，所以不存在两者的交换，它只是在不同的方向上进行切割。

事实上 如果你把某一变量的变比例扩大一倍，它的特征会缩小一半，但它的输出不会有任何变化

解释：

特征值增加一倍，Feature Scaling的值缩小一半，例如

1,2,3 （2-1/3-1 = 1）

2,4,6 （4-2/6-2 = 0.5）