**标准线性回归**

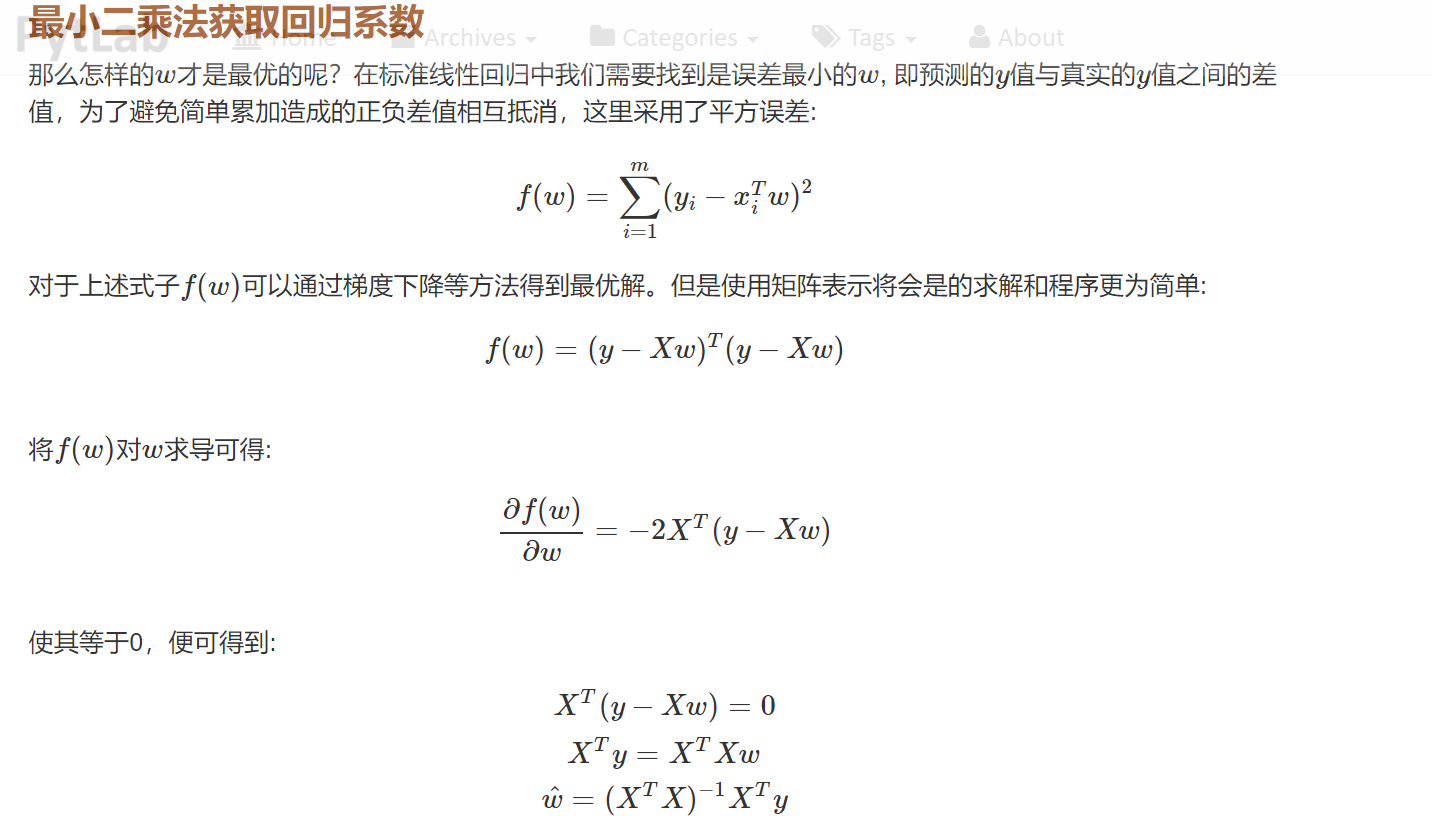
获取回归系数w：

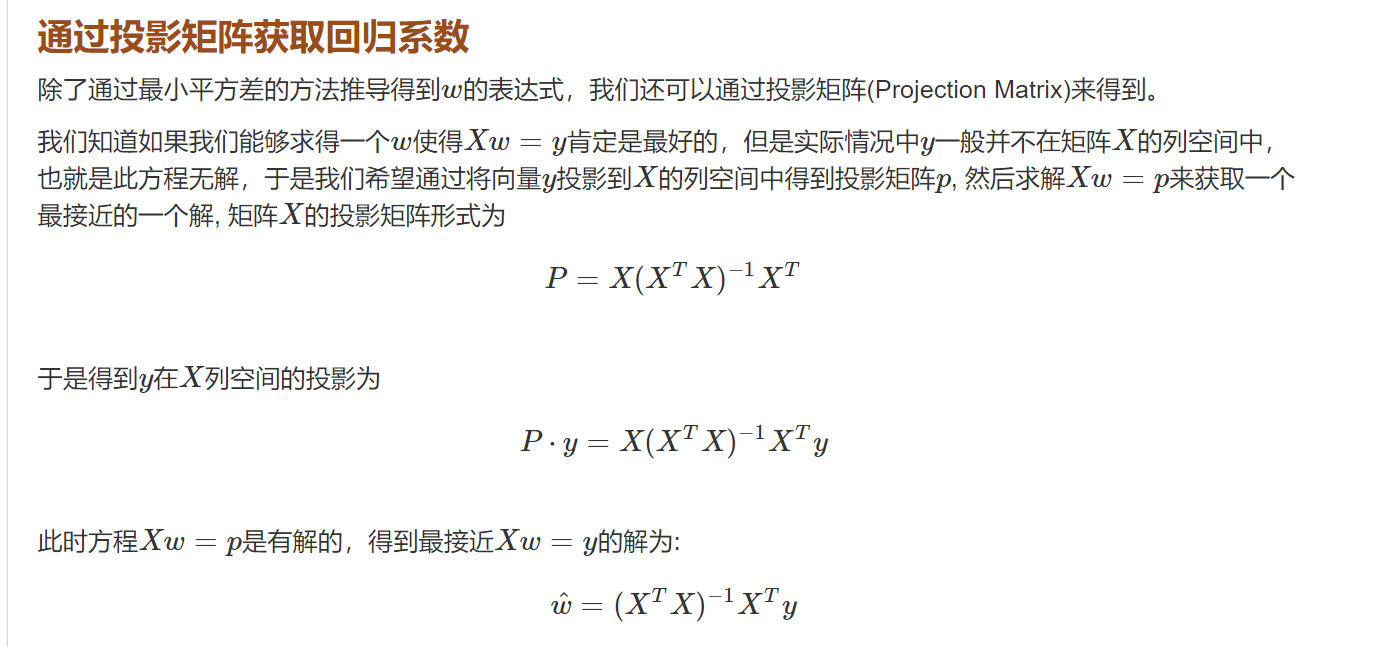
1.最小二乘法

梯度下降

矩阵表示

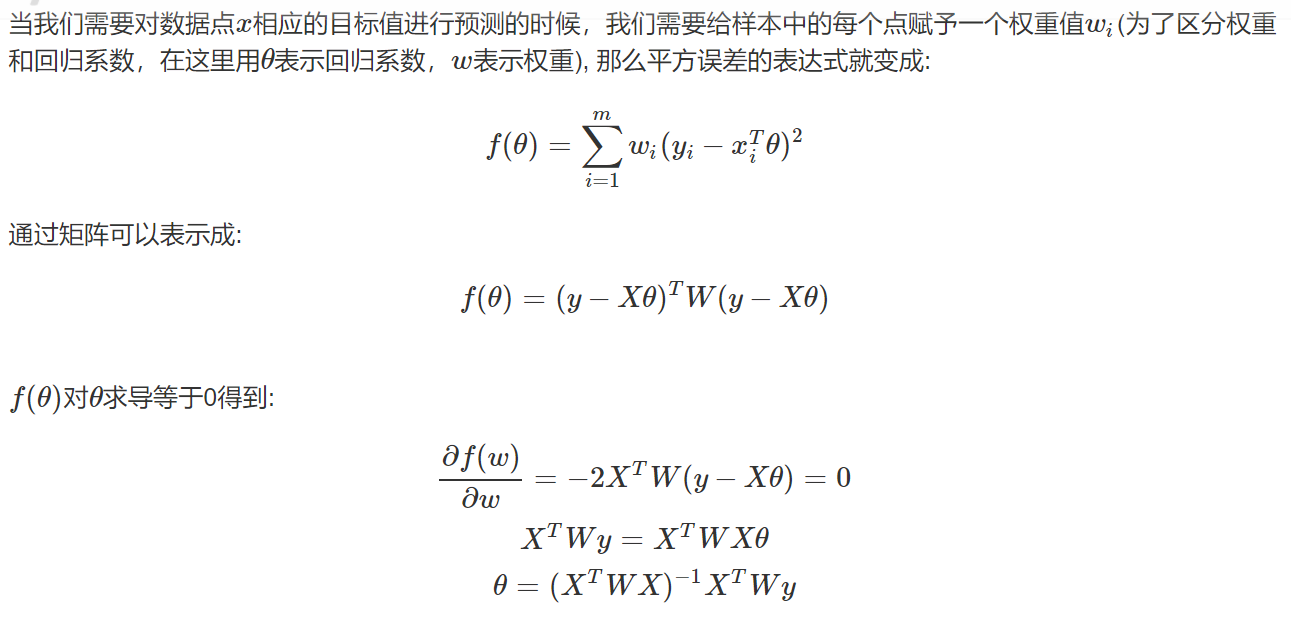
2.通过投影矩阵



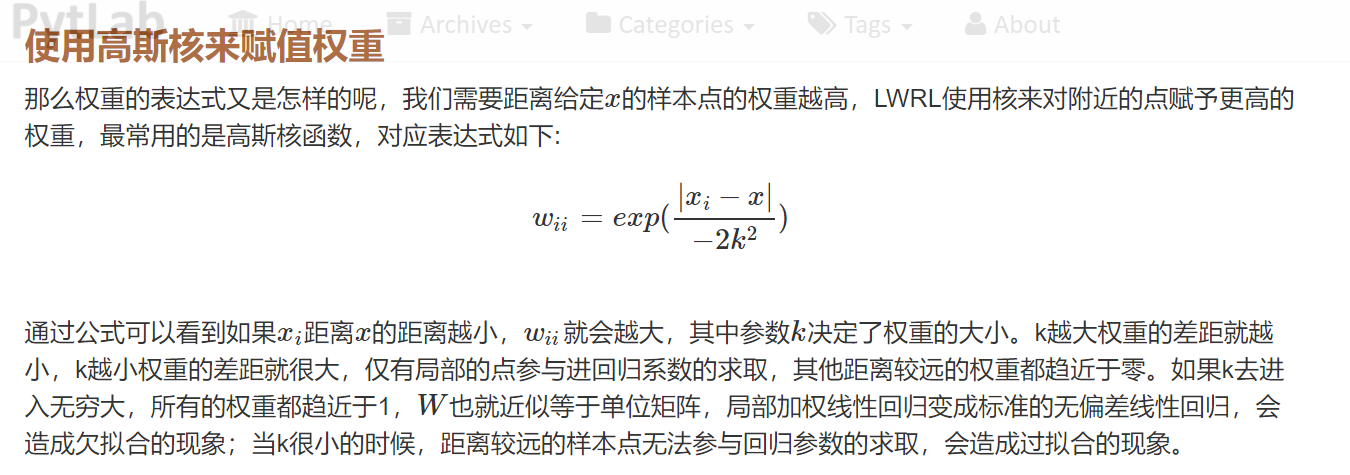


标准线性回归是一种无偏差估计，因为会造成欠拟合现象，因此就有了局部加权线性回归。

**局部加权线性回归**



使用高斯核来赋予权重：

通过调整k的大小，可发现数据的内在潜质，其缺点和KNN一样，要对每个点进行计算，其计算了较大。

附录：

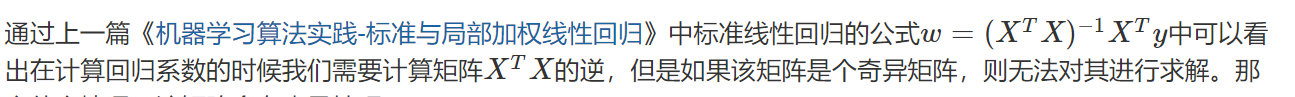
1.参考链接：

线性代数之矩阵求导：<https://blog.csdn.net/u010976453/article/details/54381248>

2.pytlab参考教程：

**岭回归和LASSO：两种回归均是在标准线性回归的基础上加上正则项来减小模型的方差**

问题的提出：



因此如果是奇异矩阵（也就是非满秩矩阵），则无法求逆，因此需要进行一定的变换才可而稳定求解，此种变化就包括岭回归和LASSO。

此处注意：就算是满秩矩阵，若数据特征中存在共线性，即相关性比较大的时候，会使得求解不稳定，则需修改损失函数。

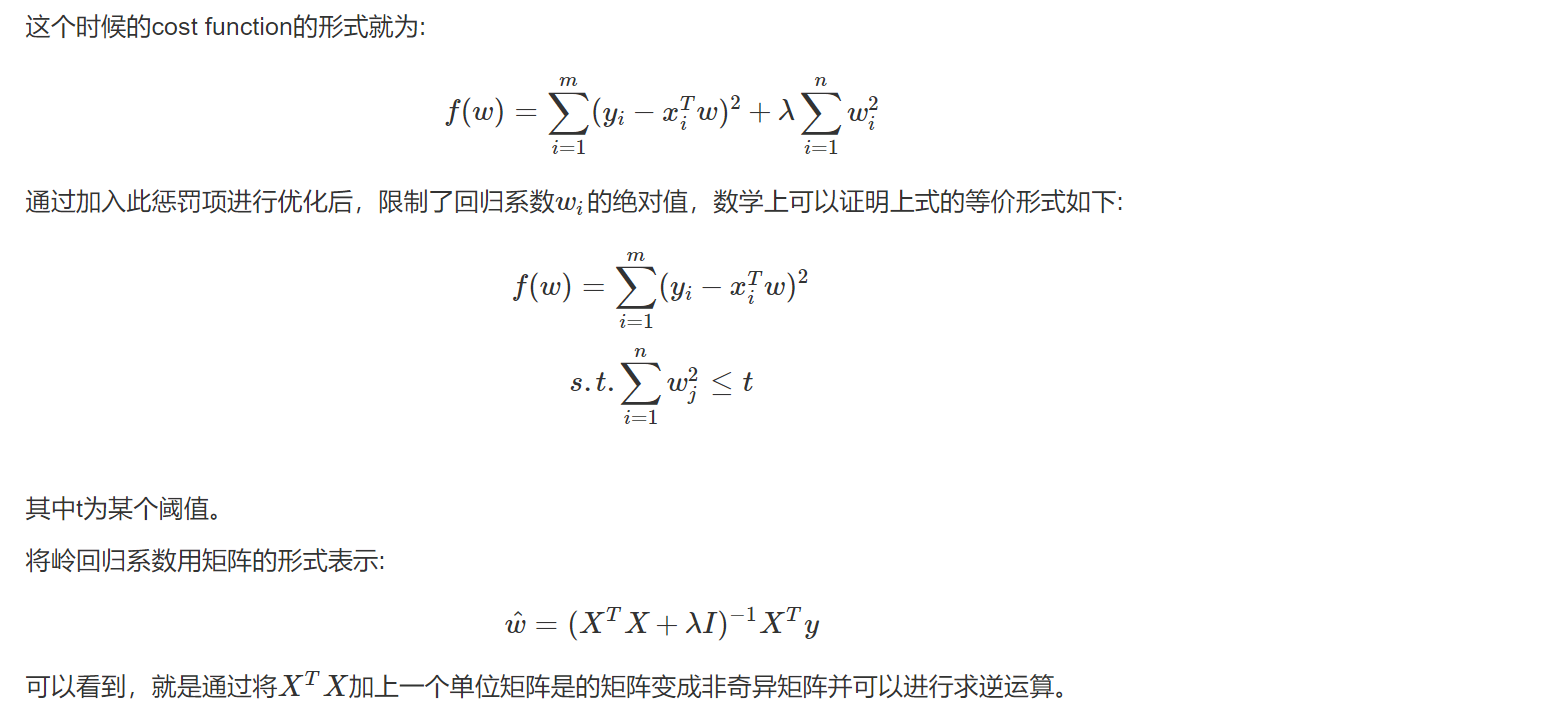
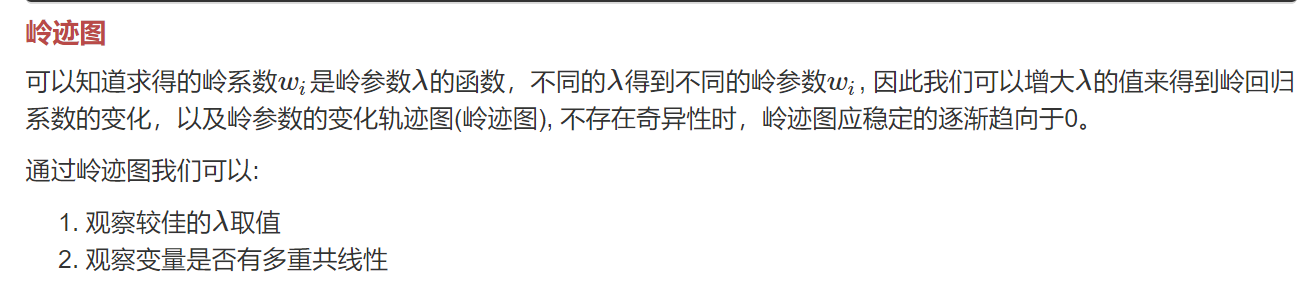
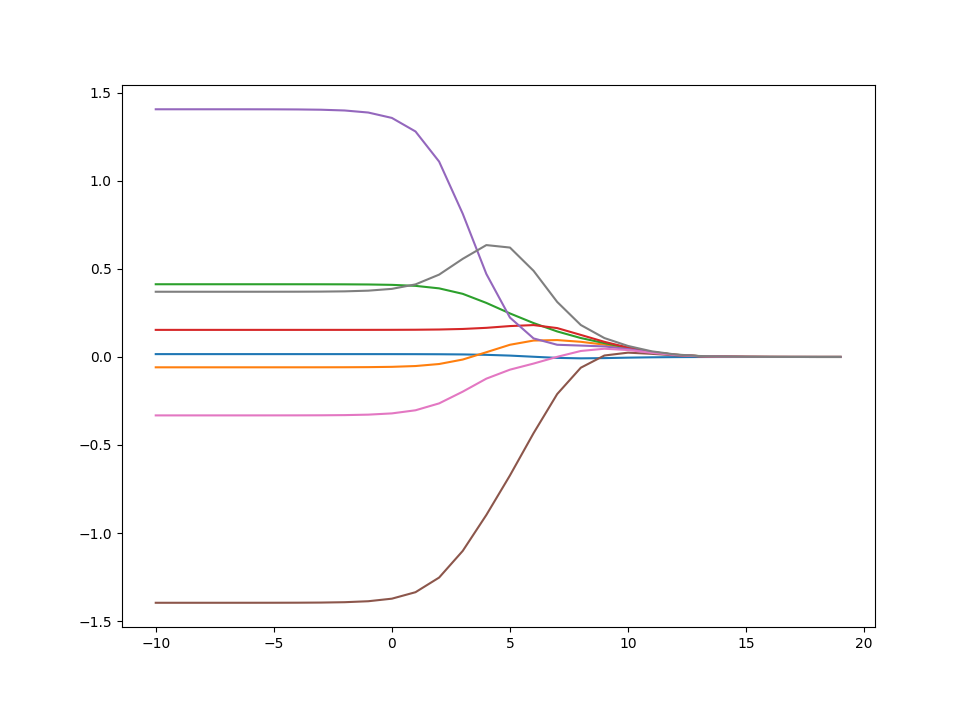
因此总结问题的提出有两点：

1.非满秩矩阵无法求逆的问题

2.是满秩矩阵但是存在特征共线而导致结果有误差的问题

数据的中心化和标准化：均值调整到0，标准差调整到1。

中心化是个平移的过程，将所有的数据中心平移到原点，而标准化是使得所有数据的不同特征都有相同的scale。

**岭回归**

岭轨迹图解释：

横轴：log(λ) 纵轴：w

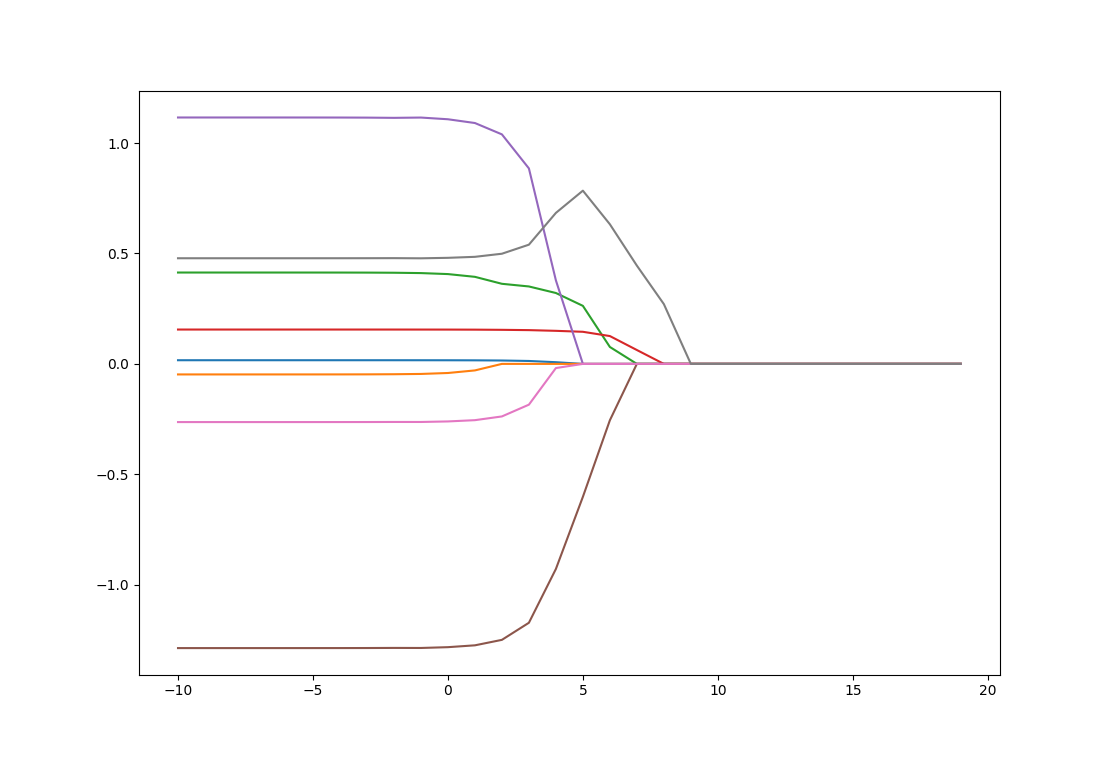
在最左边λ系数最小时，可以得到所有系数的原始值(与标准线性回归相同); 而在右边，系数全部缩减为0, 从不稳定趋于稳定；为了定量的找到最佳参数值，还需要进行交叉验证。要判断哪些变量对结果的预测最具影响力，可以观察他们的系数大小即可。

**LASSO**

岭回归的不足：当两个变量具有相关性时，可能会出现系数为很大的正数和很大的负数，但是岭回归的限定条件为所有w系数的平方和小于t。

LASSO的主要思想是构造一个一阶惩罚函数获得一个精炼的模型, 通过最终确定一些变量的系数为0进行特征筛选。

在LASSO中，当λ很小的时候，一些系数会随着变为0而岭回归却很难使得某个系数恰好缩减为0。因为岭回归的约束条件是圆，而LASSO的约束条件是方形，相比圆来说，方形更容易与抛物面相交，顶点就意味着很多系数为0，因此Lasso起到了很好的筛选变量的作用。



通过与岭轨迹图进行对比发现，随着λλ的增大，系数逐渐趋近于0，但是岭回归没有系数真正为0，而lasso中不断有系数变为0。

LASSO回归系数的获取：

1. 通过坐标下降(coordinate descent)法

2. 通过向前逐步回归法