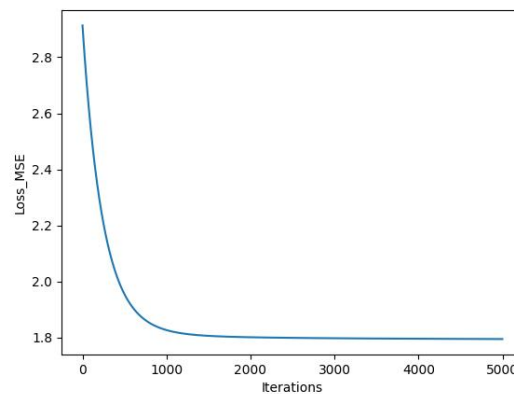


# Report

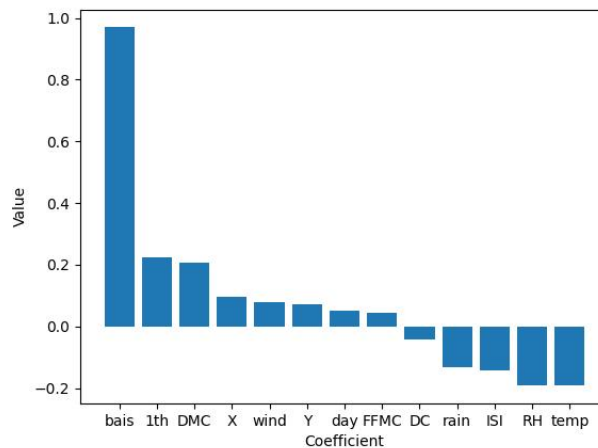
杨子董 521120910138

## 1. Linear Regression

在 linear regression 部分，采用梯度下降的方式来计算。为了方便调参，我做了梯度下降的可视化。



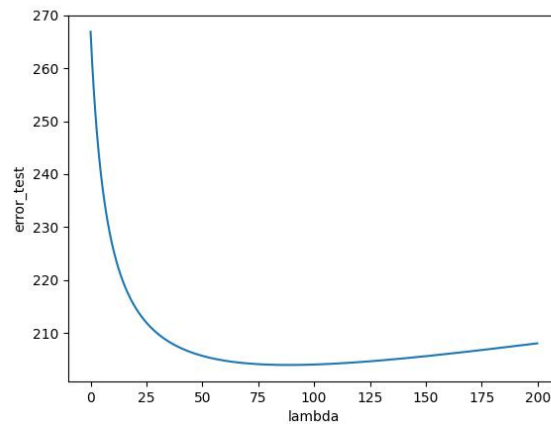
经过调参，取  $\text{step}=0.001$ ， $\text{iteration} = 2500$  得到  $\text{error test} = 196.52003882304226$ ，应该算是比较好的结果。下面是由大到小排序好的  $\beta$  向量：



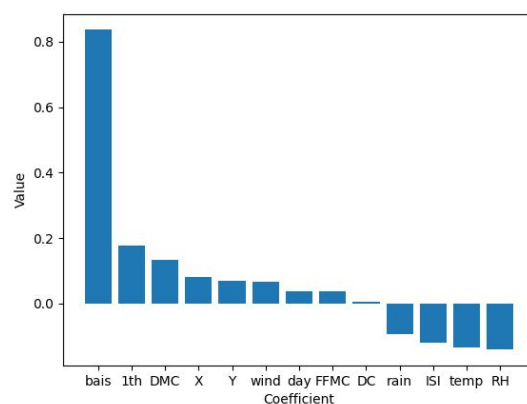
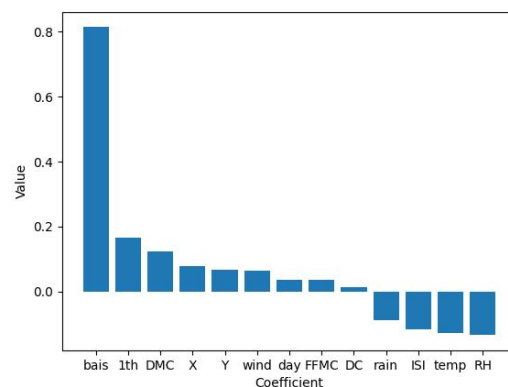
为了评估其质量，我又计算了  $\beta = 0$  和  $\beta =$  解析解的情况，分别得到， $\text{error test initial} = 275.82619023111454$  和  $\text{error test math} = 266.8897131584854$ 。可见 linear regression 还是有效果的，比什么都不做要好。但是经过归一化后的解析解比通过梯度下降的结果要差，解析解几乎没什么作用。我又输出了  $\text{train\_loss\_mse} = 1.801437589478932$  和  $\text{test\_loss\_mse} = 2.933134907806601$ ，相差较大，发现可能是这些数据本身不太适合线性回归或者数据量过少，线性回归不能很好抓住特征，存在一定程度的过拟合，而梯度下降环可以缓解这个问题。

## 2. Ridge Regression

在 Ridge regression 部分，直接采用解析解来计算。为了取到合适的  $\lambda$ ，我做了 test error 关于  $\lambda$  的可视化：



在  $\lambda$  位于 75-100 时可以取到合适结果，我们取  $\lambda=75$  和 90，结果为 error test 75 = 204.12446803814012，error test 90 = 203.9726897018685。下面是由大到小排序好的的  $\beta$  向量（上面是  $\lambda=75$ ，下面是  $\lambda=90$ ）：

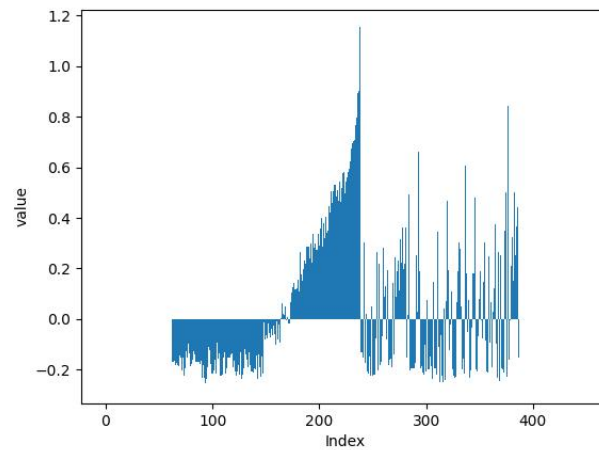


这里 Ridge regression 结果和 linear regression 梯度下降的结果差不多，甚至还比其差一点。差不多指的是，结合上面的讨论发现，这是 ridge regression 发挥了其减少过拟合风险的作用，

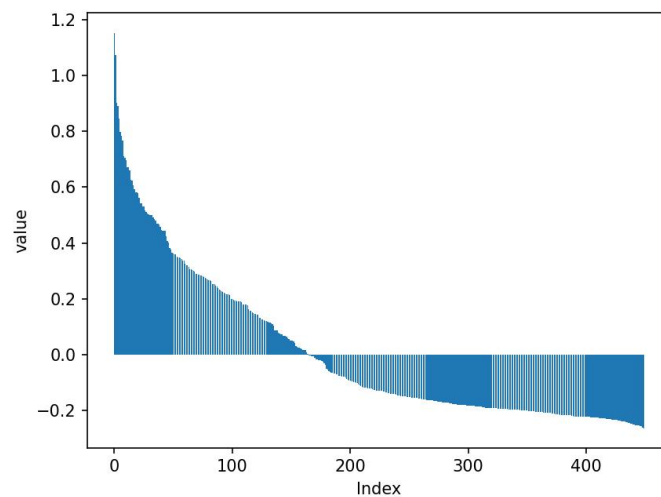
使预测结果更精确。比其差一点，可能是因为文件中存在很多的稀疏特征，比如 `rain` 系数大部分为 0，`ridge regression` 处理这些数据时会平衡，存在副作用。

### 3. Kernel Regression

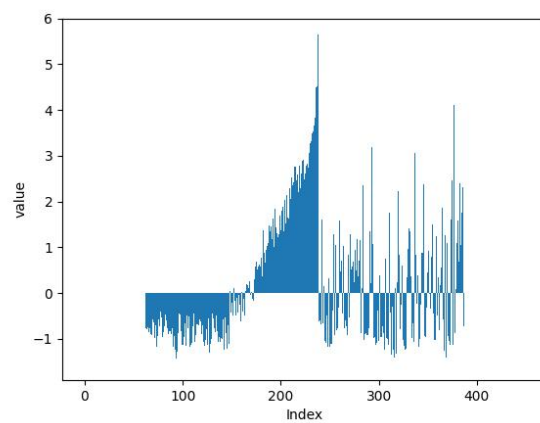
在 `Ridge regression` 部分，直接采用解析解来计算。取 `sigma_ = 0.01` 和 `lambda_ = 5`，得到 `error test = 174.18647883762262`，是比较好的结果。下面是 `c` 的可视化：



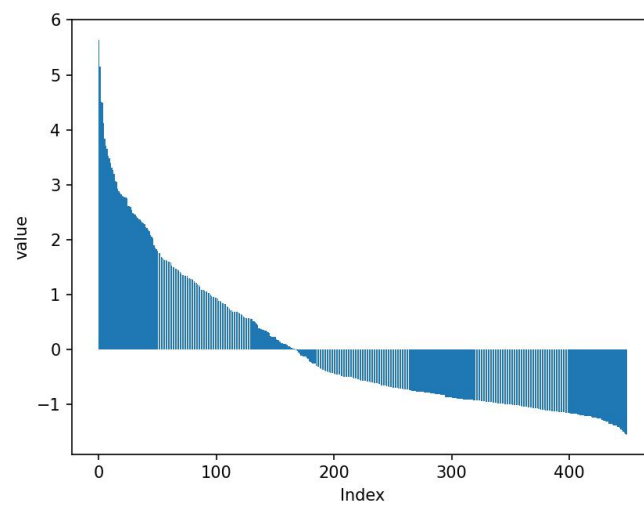
从大到小版本：



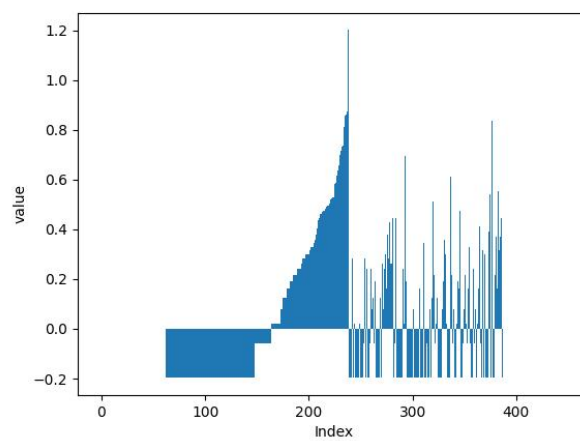
取  $\sigma = 0.01$  和  $\lambda = 1$ ，得到  $\text{error test} = 183.66891566265016$  下面是  $c$  的可视化：



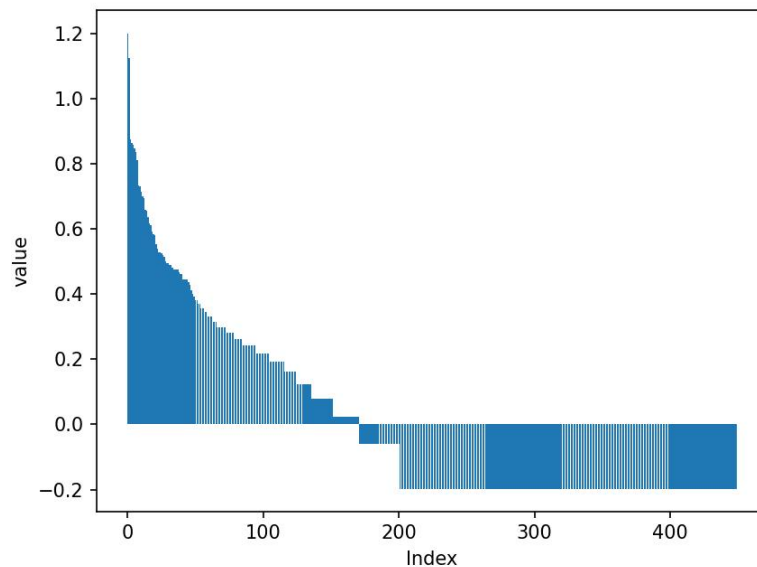
从大到小版本：



取  $\sigma = 0$  和  $\lambda = 5$ ，得到  $\text{error test} = 155.68897385110265$  下面是  $c$  的可视化：



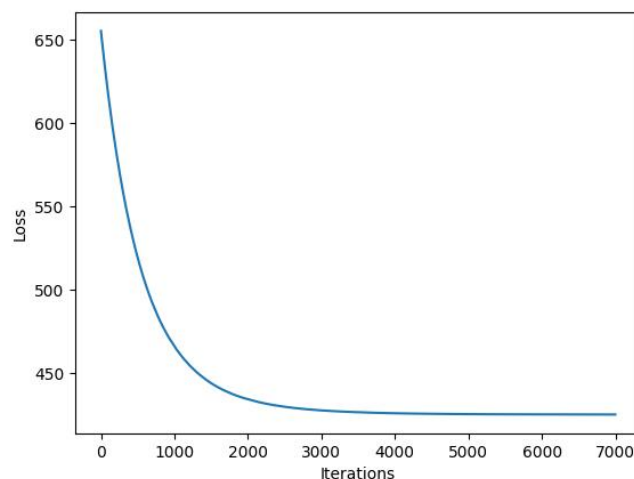
从大到小版本:



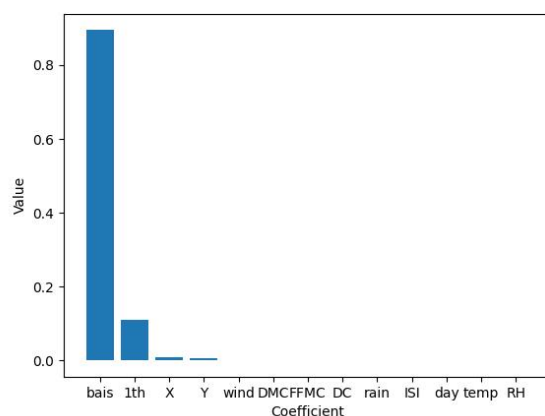
这里可以发现  $\sigma$  取的越小, 拟合效果越好,  $\lambda$  取的越大, 拟合效果也越好。在  $\sigma$  为 0 时, 取到了 4 种方法中最好的结果。但  $\sigma$  取 0 时仅仅只是对样本点进行均值化处理, 没有学习任何特征。这也说明数据集中的特征之间的距离分布比较均匀, 可能不适用 kernel regression, 更适应线性的。

## 4.Lasso Regression

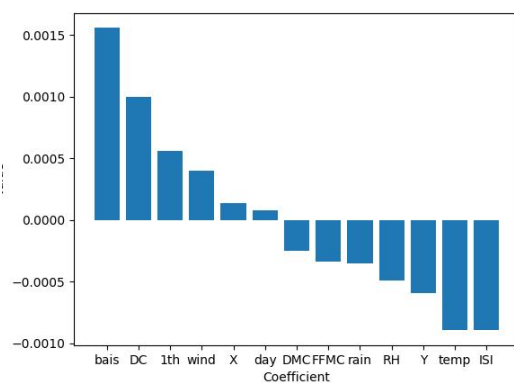
在 lasso regression 部分, 采用梯度下降的方式来计算。取  $\lambda=0.1$ 。为了方便调参, 我做了梯度下降的可视化:



经过调参, 取  $\text{step}=0.001$ ,  $\text{iteration} = 7000$  得到  $\text{error test} = 166.24442806421342$ , 应该算是比较好的结果。下面是由大到小排序好的  $\beta$  向量:



这里可以看到 **Lasso regression** 的效果是最好的，它将很多特征归 0 了，只保留了影响最大的特征，极大简化了模型，这非常符合其理论上可用于特征选择和避免过拟合的优点。后面我们取  $\lambda=1$ ，扩大 10 倍，这时由大到小排序好的的  $\beta$  向量：



所有的特征都趋向于 0 了，这是因为在  $\lambda$  过大时其正则化程度过强，将特征系数缩减太猛烈，可能会造成欠拟合。