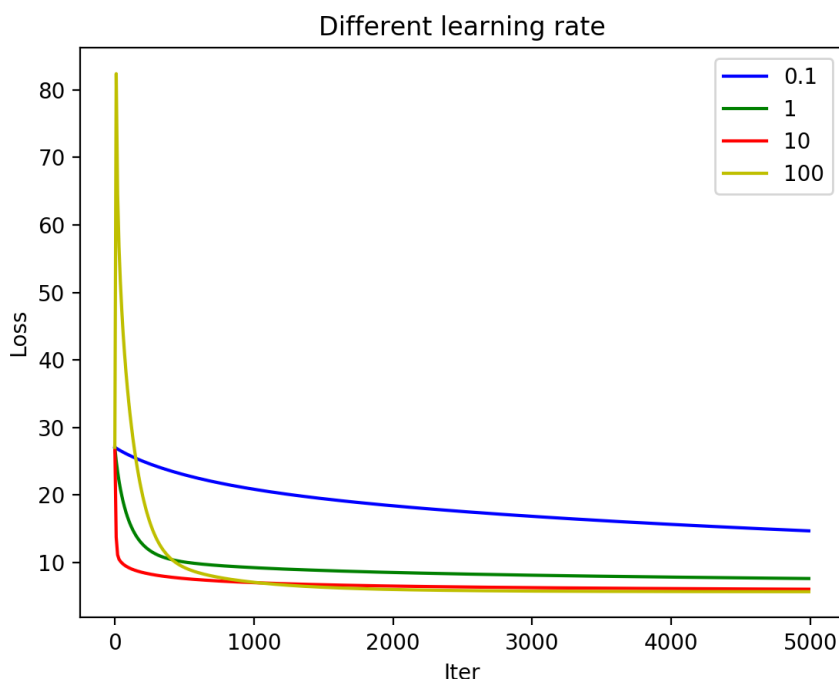


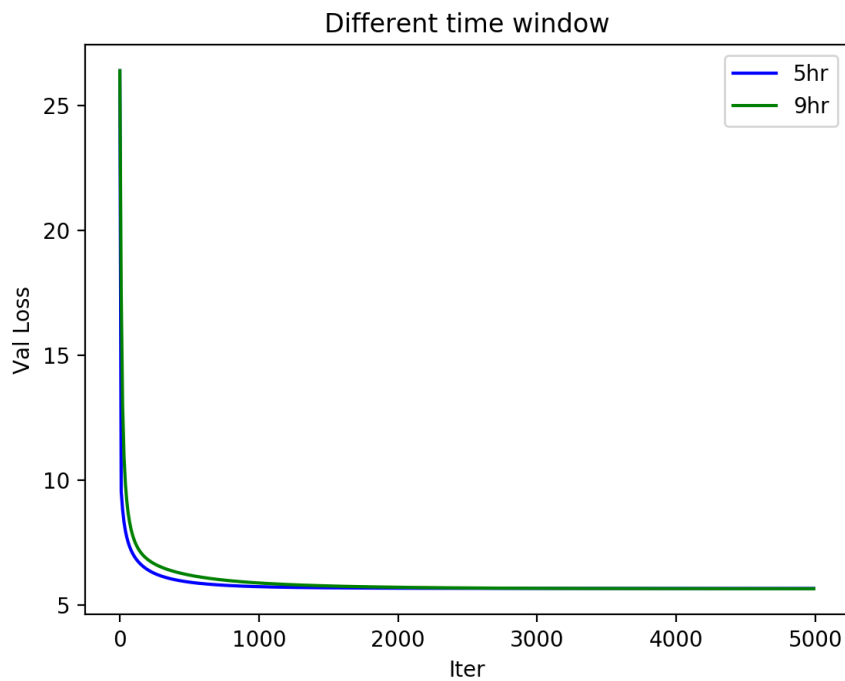
1.(2%) 使用四種不同的 learning rate 進行 training (其他參數需一致), 作圖並討論其收斂過程 (橫軸為 iteration 次數, 縱軸為 loss 的大小, 四種 learning rate 的收斂線請以不同顏色呈現在一張圖裡做比較)。

本题中我设定lr为[100, 10, 1, 0.1], 得到结果如下:



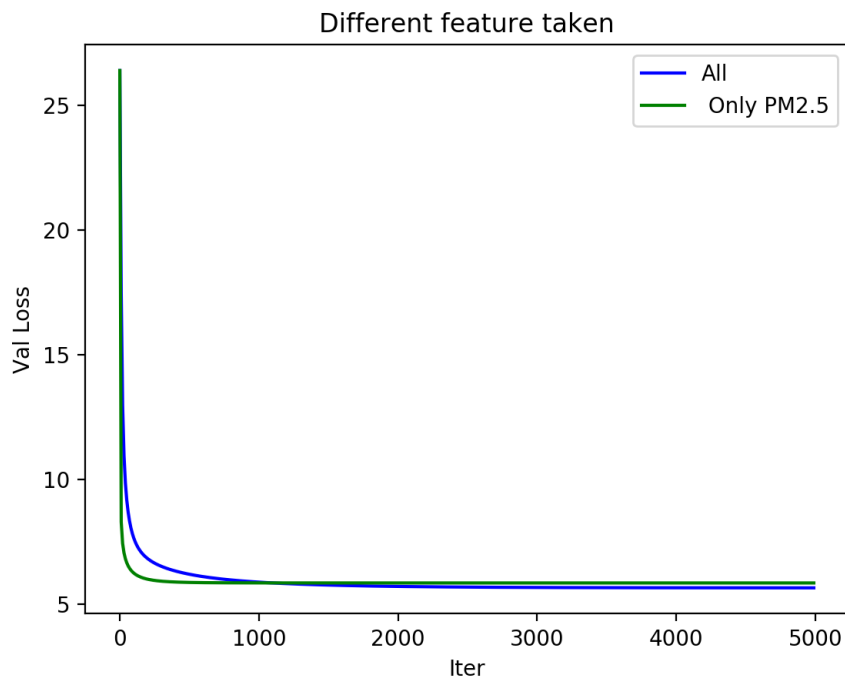
可以清晰地看到, 对于这个较简单的模型, 学习率设置为10这样大的数字就可以得到极佳的结果 (对比神经网络学习率不超过 $1e-3$ )。当学习率为0.1时, 模型缓步下降, 但需要较多步数才可收敛到全局最优, 因此其loss再开始呈现爆炸的态势, 高于同迭代次数下其他曲线。学习率为1情况同理。而当学习率为100时, 在开始的时候有可能会走过想要搜索的区域, 但因为最优解的唯一存在, 经过几次波折后总是能慢慢靠近, 并且仔细观察可发现, 随着迭代次数增多, 学习率100的loss甚至最后超过了之前表现最好的学习率10曲线。

2. (1%) 比較取前 5 hrs 和前 9 hrs 的資料 (518 + 1 v.s 918 + 1) 在 validation set 上預測的結果, 並說明造成的可能原因 (1. 因為 testing set 預測結果要上傳 Kaggle 後才能得知, 所以在報告中並不要 求同學們呈現 testing set 的結果, 至於什麼是 validation set 請參考: [https://youtu.be/D\\_S6yOJm6dQ?t=1949](https://youtu.be/D_S6yOJm6dQ?t=1949) 2. 9hr:取前9小時預測第10小時的PM2.5; 5hr:在前面的那些features中, 以5~9hr預測第10小時的PM2.5。這樣兩者在相同的validation set比例下, 會有一樣筆數的資料)。



观察发现，当利用前五小时数据时，因为这部分数据在时间上更接近第十小时，因此迭代的开始，误差表现比9小时数据好。然而随着次数的增加，两种方式的Validation loss逐渐接近，这说明随着学习的进行，模型实际上确定出前几个小时的数据重要性占比逐渐缩小，从而这些数据的权重减少，导致选用不同长度数据的测试集结果无差别。

**3. (1%) 比較只取前 9 hrs 的 PM2.5 和取所有前 9 hrs 的 features (91 + 1 vs. 918 + 1) 在 validation set 上預測的結果，並說明造成的可能原因。**



由图知，仅使用PM2.5的数据在一开始的误差极小，说明使用PM2.5预测PM2.5是正确的（数据间相关性强），但随着训练进行，使用所有数据的模型开始超越，这说明有部分其他特征会对于预测也有帮助，因此通过优化能得到更好的结果。

4. (2%) 請說明你超越 **baseline** 的 **model**(最後選擇在Kaggle上提交的) 是如何實作的 (例如: 怎麼進行 **feature selection**, 有沒有做 **pre-processing**、**learning rate** 的調整、**advanced gradient descent** 技術、不同的 **model** 等等) 。

在本人的测试集中，测试过加入二次项、三次项等，val loss差别不大；同样也试过挑选不同特征，同样收获相差不大的val loss。我认为，通过学习，模型会自动调节权重，因此特征选取无法达到满意的效果（线性方法有且只有全局最优，因此通过学习确定的最优解一定会在不需要的特征上给出较小的权重，相当于摒弃这类特征）。但实际上，通过加入这类变化，模型将会产生极大改进。接下来介绍调节过程中涉及的变动：

#### 1.高次项

首先是对高次项的选用。同讲义，仅仅加入 $w_2x^2$ 一项即可。注意求导的时候同样多出一项，应分开处理。

#### 2.特征选择

sklearn包含有SelectKBest方法，可以选取与预测目标互信息系数最大的特征。所谓互信息，实际上是类似于一种“明确的函数关系”，即变化x导致y的变化。

#### 3.神经网络

不明确是否可以用该方法，但我也尝试搭建了一套神经网络，通过非线性层ReLU的组合以得到一个拟合效果更强的模型。

在执行以上实验的过程中，我发现Public board的误差均不达标，在检查示例代码后发现，因我为了调用方便，将各个数据预处理功能封装为函数，从而导致测试集在作归一化时，用的仅仅是测试集自身的数据，当其与训练集的数据分布有差异时，就会让这些数据产生无意义的偏移，从而无法适应训练集得到的模型！因此在修改了这一细节后，重新实验，加入高次项的模型结果已符合要求。实验设定学习率10，迭代次数1000。

	Public Board	Private Board
Strong Baseline	5.55972	7.14231
Simple Baseline	6.55912	8.73773
My Result	5.95051	6.59204

注：因担心训练次数过多带来过拟合，因此学习率与迭代次数未过分调整。变动2与3都是在数据被错误归一化情况下完成的，因此结果较差，正确归一化后提交结果仅加入高次项一个改动（比不加入的Private score效果好）。