

Homework3

付宇辉 无 05 2020010768

(1) 二分类 SVM 带正则项的 Hinge loss 为:

$$L = \sum_i \max(0, 1 - y_i(\mathbf{w}^T \mathbf{x}_i + w_0)) + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

推广到多分类的损失函数:

$$L = \sum_{i \neq y_i} \max(0, \mathbf{w}_j^T \mathbf{x}_i - \mathbf{w}_{y_i}^T \mathbf{x}_i + 1) + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

其中 y_i 是数据 \mathbf{x}_i 的 label, 该损失函数的直观理解是, 当使用现在的权重算出各个类别的得分时, 如果正确类别的得分比其他类别的得分大一定的间隔, 则损失为 0, 否则为两者得分的差。即分类正确没有损失, 分类错误有一定的损失。

对于损失函数 $L_i = \sum_{j \neq y_i} \max(0, \mathbf{w}_j^T \mathbf{x}_i - \mathbf{w}_{y_i}^T \mathbf{x}_i + 1)$, 求其梯度得到:

$$\nabla_{\mathbf{w}_{y_i}} L_i = - \sum_{j \neq y_i} \mathbf{x}_i, \quad \text{if } \mathbf{w}_j^T \mathbf{x}_i - \mathbf{w}_{y_i}^T \mathbf{x}_i + 1 > 0$$

$$\nabla_{\mathbf{w}_j} L_i = \mathbf{x}_i, \quad \text{if } \mathbf{w}_j^T \mathbf{x}_i - \mathbf{w}_{y_i}^T \mathbf{x}_i + 1 > 0$$

根据上面的分析编写代码, 见压缩包中的代码。

(2) 使用默认参数, 在图像原像素特征上进行 SVM 分类

得到下面的结果:

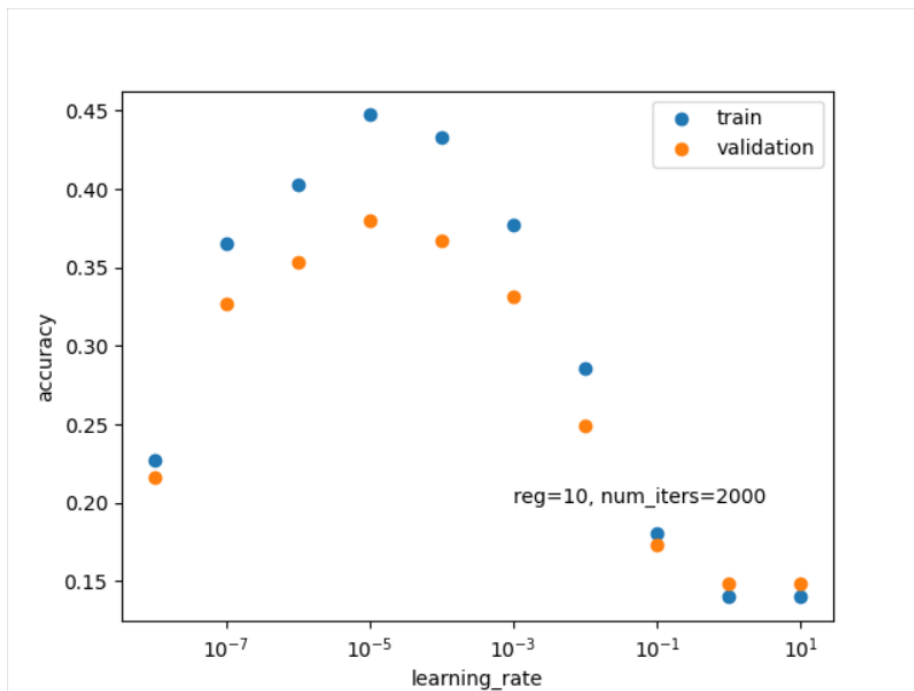
training accuracy: 0.413973

validation accuracy: 0.312789

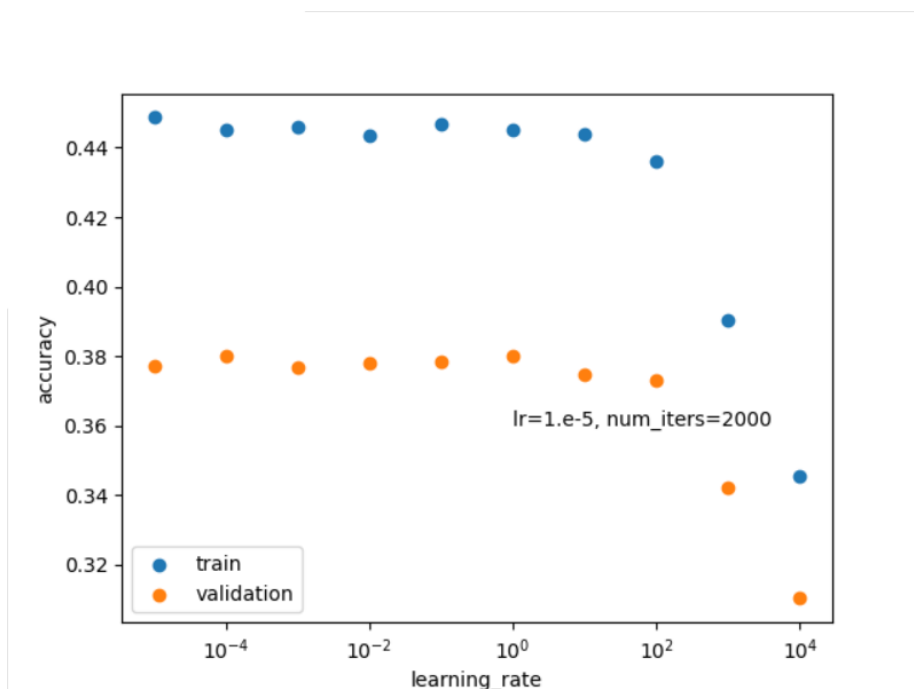
test accuracy: 0.354672

(3)

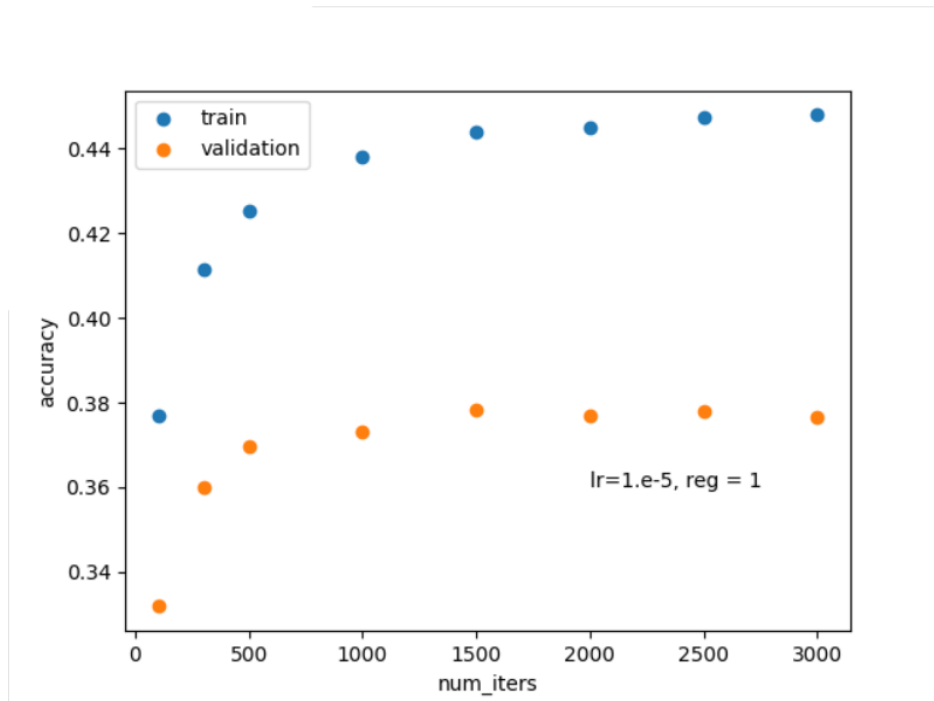
1. 首先对学习速率进行分析: 固定正则化参数为 10, 迭代步数为 2000 次, 将学习率从 10^{-8} 到 10^1 进行遍历, 得到下面的散点图, 横轴为学习率, 纵轴为在训练集和验证集上的准确率。可以看到当学习率较小时, 在训练集和验证集上的准确率都比较小, 这可能是由于迭代次数不够, 没有收敛导致的; 当学习率比较大时, 同样准确率比较小, 这可能是由于学习率大, 梯度下降时步长较大, 损失函数的值在来回振荡, 难以收敛导致。从图中可以看出, 大概在学习率为 10^{-5} 时, 训练的效果比较好。



2. 接着对正则化参数进行筛选：固定学习率为 10^{-5} ，迭代步数为 2000 次，使正则化参数在 $10^{-5} \sim 10^4$ 中以 10 的幂次进行取值。得到下面的散点图，横轴为学习率，纵轴为在训练集和验证集上的准确率。当正则化参数比较小时，可能导致过拟合：在训练集上表现不错，但在验证集上表现较差，但在这次实验中没有出现，可能是模型比较小，数据不太多的原因；正则化参数比较大时，导致 w 参数的学习无法进行，学习效果也不好。当正则化参数为 1 时，效果比较好。



3. 最后对迭代次数进行分析，选择上述选择的最好参数，然后进行迭代训练。很容易能够想到训练次数不够的话，没有收敛，效果不好；训练次数太多，浪费计算时间和资源，没有必要。我们这里选择 2500。



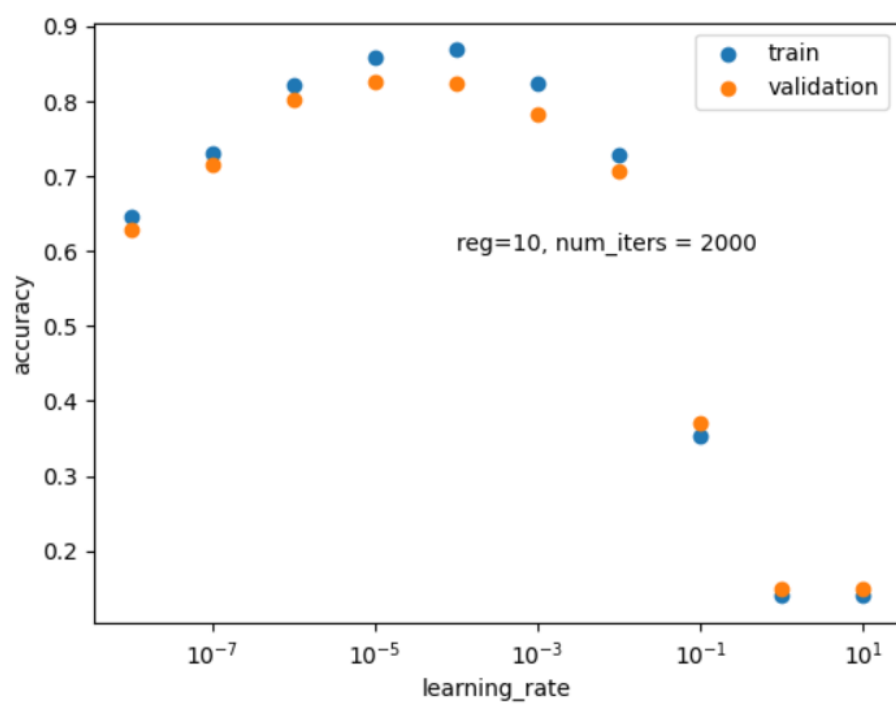
所以给出最好的一组参数: $\text{learning_rate}=1.e-5$, $\text{reg}=1$, $\text{num_iters}=2500$, 得到如下结果

```
training accuracy: 0.450253
validation accuracy: 0.377007
test accuracy: 0.415481
class angry accuracy: 0.20981210855949894
class disgust accuracy: 0.036036036036036036
class fear accuracy: 0.22265625
class happy accuracy: 0.6093573844419391
class neutral accuracy: 0.4630981346309814
class sad accuracy: 0.35605453087409783
class surprise accuracy: 0.5467800729040098
```

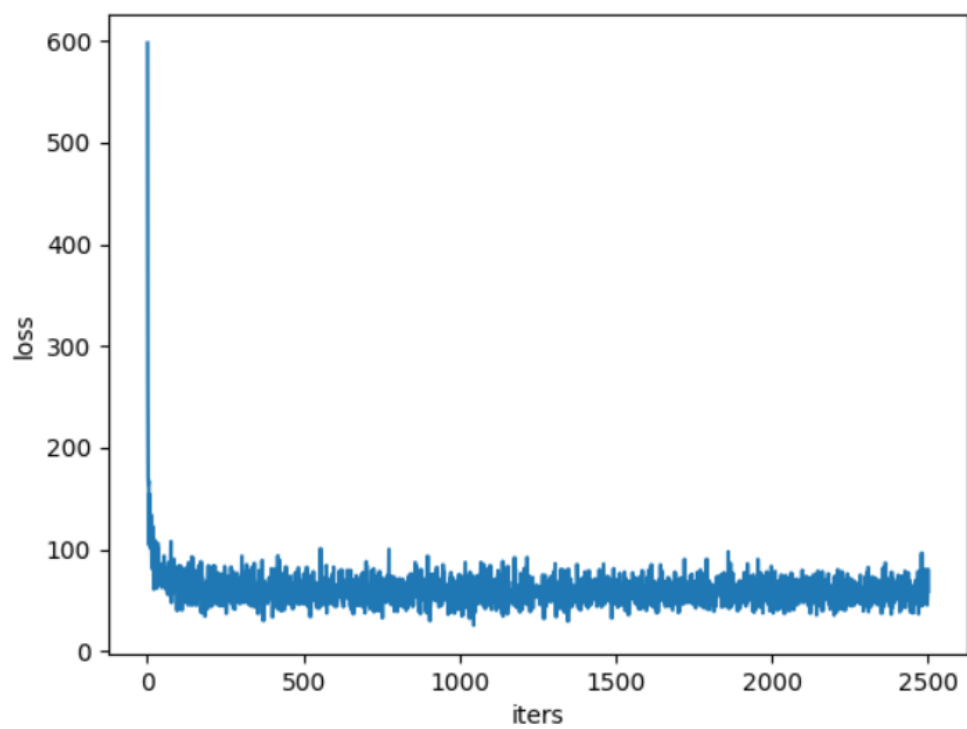
- (4) 和上述进行相同的筛选，得到在验证集上表现的最好的一组参数：

$\text{learning_rate}=1.e-4$, $\text{reg}=10$, $\text{num_iters}=2500$

(下图是进行选择学习率过程中做的图表，其他参数不再赘述)



在这组超参数下，得到 loss 随迭代次数的变化曲线：



并得到输出结果:

```
training accuracy: 0.867845
validation accuracy: 0.824898
test accuracy: 0.664435
class angry accuracy: 0.6263048016701461
class disgust accuracy: 0.6036036036036037
class fear accuracy: 0.484375
class happy accuracy: 0.8478015783540023
class neutral accuracy: 0.6399026763990268
class sad accuracy: 0.5308740978348035
class surprise accuracy: 0.7849331713244229
```

总结

提高 SVM 图像分类性能的一些方法:

1. 损失函数加入正则化项, 减少 overfitting 的影响。
2. 提取更好的分类特征 feature, 能够使性能大幅度提高。
3. 合理选择超参数, 使训练的效果更好。
4. 对图像数据进行预处理, 将数据分布更均匀。