

Section 1 SVM

1. 依據 `linear_classifier.ipynb` 的檔案初始化訓練、驗證、測試集

- 'X_train': `dtype` tensor of shape (N_train, D) giving training images
- 'X_val': `dtype` tensor of shape (N_val, D) giving val images
- 'X_test': `dtype` tensor of shape (N_test, D) giving test images
- 'y_train': int64 tensor of shape (N_train,) giving training labels
- 'y_val': int64 tensor of shape (N_val,) giving val labels
- 'y_test': int64 tensor of shape (N_test,) giving test labels

where D is the dimension of the image data

```
Train data shape: torch.Size([40000, 3073])
Train labels shape: torch.Size([40000])
Validation data shape: torch.Size([10000, 3073])
Validation labels shape: torch.Size([10000])
Test data shape: torch.Size([10000, 3073])
Test labels shape: torch.Size([10000])
```

2. 對驗證集呼叫 `svm_loss_naive()` 可先得到 `loss: 9.001455`
3. 不使用正規化情況下的部分抽樣微分，計算 `loss` 和 `gradient` `relative error` 應小於 $1e-5$ 。

```
numerical: 0.031599 analytic: 0.031599, relative error: 3.284032e-08
numerical: 0.111444 analytic: 0.111444, relative error: 1.584046e-07
numerical: 0.011204 analytic: 0.011204, relative error: 1.860695e-07
numerical: -0.046128 analytic: -0.046128, relative error: 2.741202e-07
numerical: 0.071948 analytic: 0.071948, relative error: 2.343481e-08
numerical: 0.025688 analytic: 0.025688, relative error: 6.774323e-07
numerical: 0.185388 analytic: 0.185388, relative error: 6.482448e-08
numerical: -0.021740 analytic: -0.021740, relative error: 7.159385e-08
numerical: -0.159613 analytic: -0.159613, relative error: 8.523654e-09
numerical: 0.092690 analytic: 0.092690, relative error: 1.126152e-07
```

4. 給定正規化 `reg = 1000`，計算 `loss` 和 `gradient`，`relative error` 應小於 $1e-5$ 。

```
numerical: 0.077269 analytic: 0.077269, relative error: 3.692407e-08
numerical: 0.388860 analytic: 0.388860, relative error: 5.582723e-08
numerical: -0.150628 analytic: -0.150628, relative error: 3.446075e-08
numerical: -0.148681 analytic: -0.148681, relative error: 7.277165e-08
numerical: -0.075842 analytic: -0.075842, relative error: 1.601786e-08
numerical: -0.004539 analytic: -0.004538, relative error: 4.379537e-06
numerical: 0.376041 analytic: 0.376041, relative error: 2.567064e-08
numerical: 0.032924 analytic: 0.032924, relative error: 9.885950e-08
numerical: 0.105138 analytic: 0.105138, relative error: 1.886019e-08
numerical: -0.033852 analytic: -0.033852, relative error: 3.237507e-07
```

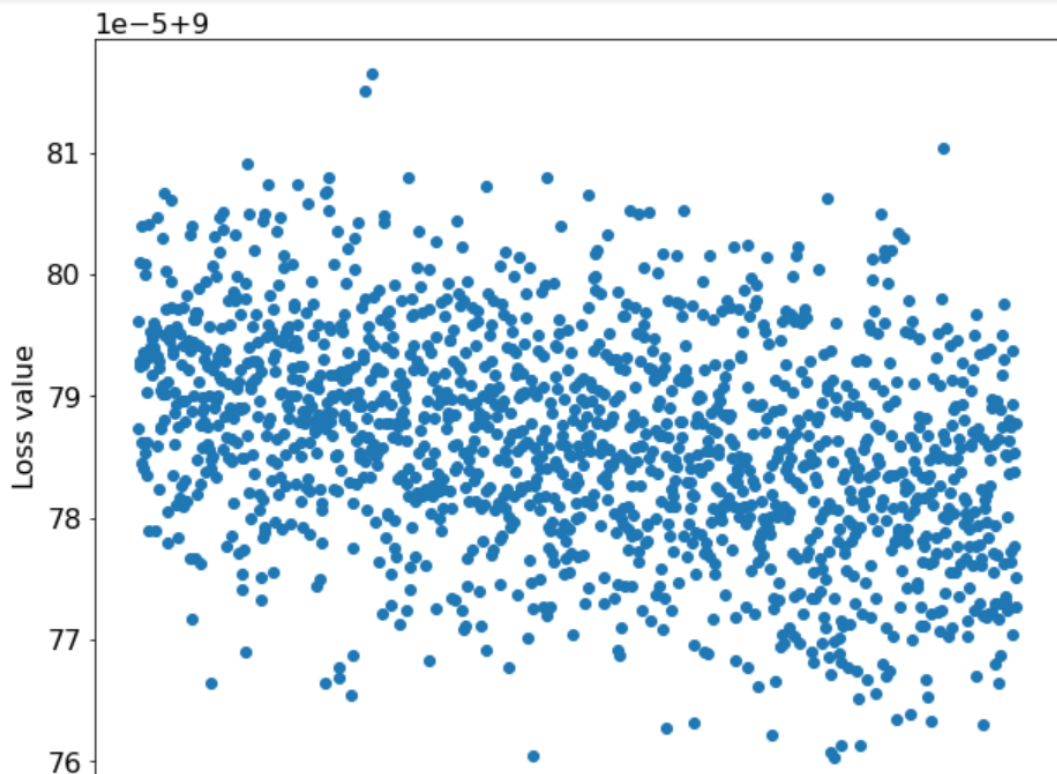
5. 比較使用 `svm_loss_naive()` 和 `svm_loss_vectorized()` 的執行速度

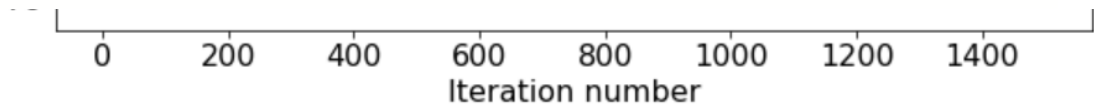
```
Naive loss: 9.000718e+00 computed in 129.65ms
Vectorized loss: 9.000718e+00 computed in 40.00ms
Difference: -1.78e-15
Speedup: 3.24X
```

6. 由於 **vectorized** 的執行速度較快，我們使用該方法 train a linear classifier using some default hyperparameters(`learning_rate=3e-11`, `reg=2.5e4`, `num_iters=1500`, `verbose=True`)

```
iteration 0 / 1500: loss 9.000796
iteration 100 / 1500: loss 9.000782
iteration 200 / 1500: loss 9.000793
iteration 300 / 1500: loss 9.000787
iteration 400 / 1500: loss 9.000789
iteration 500 / 1500: loss 9.000780
iteration 600 / 1500: loss 9.000795
iteration 700 / 1500: loss 9.000778
iteration 800 / 1500: loss 9.000782
iteration 900 / 1500: loss 9.000763
iteration 1000 / 1500: loss 9.000791
iteration 1100 / 1500: loss 9.000787
iteration 1200 / 1500: loss 9.000772
iteration 1300 / 1500: loss 9.000786
iteration 1400 / 1500: loss 9.000787
That took 7.312340s
```

7. 由於 training loss 並未遞減得很快，目前的 hyperparameters 仍不好





8. 由下圖可知一開始在訓練和驗證集的表現非常差

Training accuracy: 8.91%
Validation accuracy: 9.08%

9. Tune hyperparameters：透過調整 learning rate 和 regularization strengths 去獲得最佳參數

觀察 learning rate 給定為 0.1，regularization strengths 設定從 $1e-1$, $1e-2$, ..., $1e-10$ ，可觀察到從 $reg = 1e-6$ 開始，模型 accuracy 便維持在 30.18%，最佳為 30.29%

```
lr 1.000000e-01 reg 1.000000e-10 train accuracy: 0.309775 val accuracy: 0.301800
lr 1.000000e-01 reg 1.000000e-09 train accuracy: 0.309775 val accuracy: 0.301800
lr 1.000000e-01 reg 1.000000e-08 train accuracy: 0.309775 val accuracy: 0.301800
lr 1.000000e-01 reg 1.000000e-07 train accuracy: 0.309775 val accuracy: 0.301800
lr 1.000000e-01 reg 1.000000e-06 train accuracy: 0.309775 val accuracy: 0.301800
lr 1.000000e-01 reg 1.000000e-05 train accuracy: 0.303625 val accuracy: 0.295400
lr 1.000000e-01 reg 1.000000e-04 train accuracy: 0.309650 val accuracy: 0.302900
lr 1.000000e-01 reg 1.000000e-03 train accuracy: 0.304125 val accuracy: 0.296200
lr 1.000000e-01 reg 1.000000e-02 train accuracy: 0.301175 val accuracy: 0.290700
lr 1.000000e-01 reg 1.000000e-01 train accuracy: 0.260575 val accuracy: 0.246900
best validation accuracy achieved during cross-validation: 0.302900
```

觀察 learning rate 給定為 0.01，regularization strengths 設定從 $1e-1$, $1e-2$, ..., $1e-7$ ，可觀察到從 $reg = 1e-6$ 開始，模型 accuracy 便維持在 35.81%，最佳為 35.82%

```
lr 1.000000e-02 reg 1.000000e-07 train accuracy: 0.372725 val accuracy: 0.358100
lr 1.000000e-02 reg 1.000000e-06 train accuracy: 0.372725 val accuracy: 0.358100
lr 1.000000e-02 reg 1.000000e-05 train accuracy: 0.372475 val accuracy: 0.357700
lr 1.000000e-02 reg 1.000000e-04 train accuracy: 0.372275 val accuracy: 0.357600
lr 1.000000e-02 reg 1.000000e-03 train accuracy: 0.372375 val accuracy: 0.358200
lr 1.000000e-02 reg 1.000000e-02 train accuracy: 0.372275 val accuracy: 0.357400
lr 1.000000e-02 reg 1.000000e-01 train accuracy: 0.371850 val accuracy: 0.356200
```

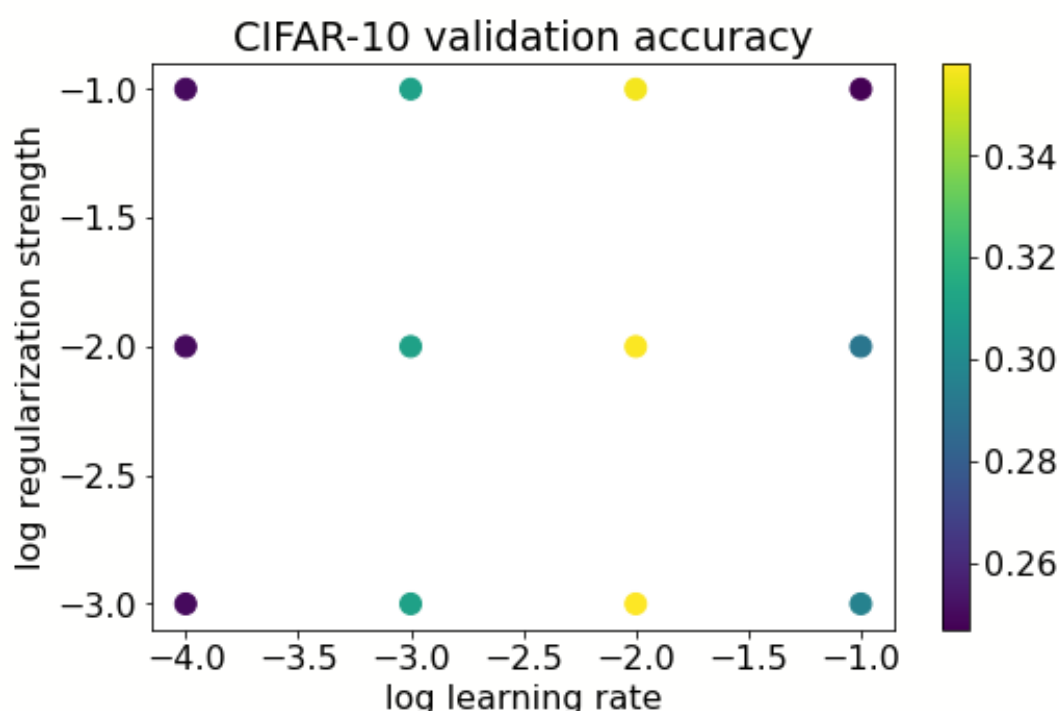
觀察 learning rate 給定為 0.001，regularization strengths 設定從 $1e-1$, $1e-2$, ..., $1e-7$ ，可觀察到從 $reg = 1e-3$ 開始，模型 accuracy 便維持在 31.07%，最佳為 31.08%

```
lr 1.000000e-03 reg 1.000000e-07 train accuracy: 0.311200 val accuracy: 0.310700
lr 1.000000e-03 reg 1.000000e-06 train accuracy: 0.311200 val accuracy: 0.310700
lr 1.000000e-03 reg 1.000000e-05 train accuracy: 0.311200 val accuracy: 0.310700
lr 1.000000e-03 reg 1.000000e-04 train accuracy: 0.311200 val accuracy: 0.310700
lr 1.000000e-03 reg 1.000000e-03 train accuracy: 0.311200 val accuracy: 0.310700
lr 1.000000e-03 reg 1.000000e-02 train accuracy: 0.311350 val accuracy: 0.310800
lr 1.000000e-03 reg 1.000000e-01 train accuracy: 0.311125 val accuracy: 0.310300
```

觀察 learning rate 給定為 0.00001，regularization strengths 設定從 $1e-1$, $1e-2$, ..., $1e-6$ ，可觀察到從 $\text{reg} = 1e-1$ 開始，模型 accuracy 便維持在 24.96%，最佳為 24.96%

```
lr 1.000000e-04 reg 1.000000e-06 train accuracy: 0.247100 val accuracy: 0.249600
lr 1.000000e-04 reg 1.000000e-05 train accuracy: 0.247100 val accuracy: 0.249600
lr 1.000000e-04 reg 1.000000e-04 train accuracy: 0.247100 val accuracy: 0.249600
lr 1.000000e-04 reg 1.000000e-03 train accuracy: 0.247100 val accuracy: 0.249600
lr 1.000000e-04 reg 1.000000e-02 train accuracy: 0.247100 val accuracy: 0.249600
lr 1.000000e-04 reg 1.000000e-01 train accuracy: 0.247150 val accuracy: 0.249600
```

經由上述嘗試，learning rate 為 0.01 且 regularization strengths 為 0.001 時，SVM 最佳 accuracy 為 35.82%。下圖為驗證集顯示 learning rate 和 regularization strengths 值不同時的精確度狀況，與結果吻合。



10. 將校正過的模型放入測試集測試

```
linear SVM on raw pixels final test set accuracy: 0.366400
```

accuracy 有 36.64%，超過 35% 的標準，結果還算不錯。

11. 將各個 class 視覺化

plane



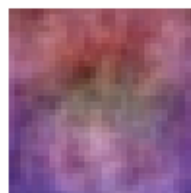
car



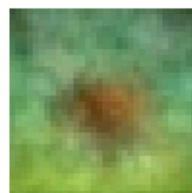
bird



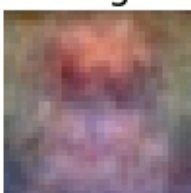
cat



deer



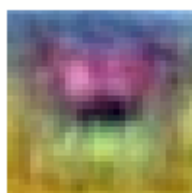
dog



frog



horse



ship



truck

