L3 损失函数和优化损失函数

2020年3月28日 18:33

1. 损失函数

a. 概念诠释：将参数W作为输入数据，用于定量估计W的优劣程度

b.

$$\{(x_i, y_i)\}_{i=1}^{N}$$

Where x_i is image and
y_i is (integer) label

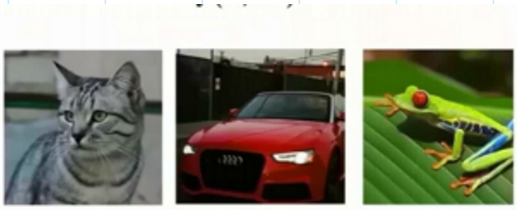
Loss over the dataset is a sum of loss over examples:

$$L = \frac{1}{N}\sum L_i(f(x_i, W), y_i)$$

c. N代表样本容量，最终的损失值L即各样本损失值的算术平均，f即上一讲中线性分类的函数模型

d. e.g.:

i. 多分类SVM损失函数（合页损失函数）



Given an example (x_i, y_i) where x_i is the image and where y_i is the (integer) label,

and using the shorthand for the scores vector: $s = f(x_i, W)$

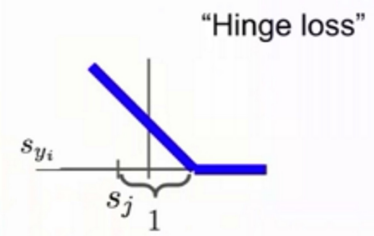
the SVM loss has the form:

$$L_i = \sum_{j \neq y_i} \begin{cases} 0 & \text{if } s_{y_i} \geq s_j + 1 \\ s_j - s_{y_i} + 1 & \text{otherwise} \end{cases}$$

$$= \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + 1)$$

cat	**3.2**	1.3	2.2
car	5.1	**4.9**	2.5
frog	-1.7	2.0	**-3.1**

其中j是指其他类别（非正确）对应的预测分数



? 1) 损失函数可缩放，不会对损失函数产生太大影响
2) 初始化w时，通常使用很小的随机值，在第一次迭代时，各分类的分数倾向于为较小的均匀分布的值，因此损失函数接近为类别数-1，否则可能存在bug
3) 平方项损失函数可用来扩大错误（区分错误的严重性）
4) e.g.

Multiclass SVM Loss: Example code

$$L_i = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + 1)$$

```
def L_i_vectorized(x, y, W):
  scores = W.dot(x)
  margins = np.maximum(0, scores - scores[y] + 1)
  margins[y] = 0
  loss_i = np.sum(margins)
  return loss_i
```

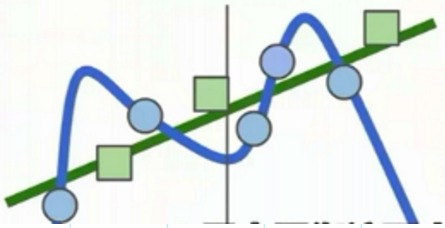
5) 如果有一个w的损失函数的值为零，对其进行缩放得到的w损失值也为0（合适的w不止一个，需要筛选出最佳的w）

网易云课堂

$$L(W) = \underbrace{\frac{1}{N}\sum_{i=1}^{N} L_i(f(x_i, W), y_i)}_{\text{Data loss}} + \underbrace{\lambda R(W)}_{\text{Regularization}}$$

Data loss: Model predictions should match training data

Regularization: Model should be "simple", so it works on test data



Occam's Razor:
"Among competing hypotheses, the simplest is the best"
William of Ockham, 1285 - 1347

引入正则项，其中λ为超参数

$$L = \frac{1}{N}\sum_{i=1}^{N}\sum_{j\neq y_i}\max(0, f(x_i;W)_j - f(x_i;W)_{y_i} + 1) + \boxed{\lambda R(W)}$$

In common use:

L2 regularization	$R(W) = \sum_k \sum_l W_{k,l}^2$
L1 regularization	$R(W) = \sum_k \sum_l \lvert W_{k,l}\rvert$
Elastic net (L1 + L2)	$R(W) = \sum_k \sum_l \beta W_{k,l}^2 + \lvert W_{k,l}\rvert$

Max norm regularization (might see later)
Dropout (will see later)
Fancier: Batch normalization, stochastic depth

ii.

Softmax Classifier (Multinomial Logistic Regression)

网易云课堂

$$L_i = -\log\left(\frac{e^{sy_i}}{\sum_j e^{s_j}}\right)$$

unnormalized probabilities

Q: What is the min/max possible loss L_i?

	unnormalized log probabilities		exp		normalize		probabilities
cat	3.2	→	24.5	→	0.13	→	L_i = -log(0.13) = 0.89
car	5.1		164.0		0.87		
frog	-1.7		0.18		0.00		

1) 纠错：初始化时，损失函数值倾向于 ln（总类别数）

iii. 差异：前者在差距大于安全边际后不再关

注数据点，后者会努力进一步扩大差距。

2. 优化
    a. 梯度：偏导数组成的向量（多元函数）指向函数增大的最快方向
    b. 通常先计算梯度表达式，然后用数值梯度进行单元检查（此过程最好减少问题的参数数量以减少运行时间）
    c. 步长（学习率）关键超参
    d. 当样本容量过大时，往往使用随机梯度下降