机器学习导论
习题四

141180016, 丁俊峰, 1411800116@smail.nju.edu.cn

2017 年 5 月 17 日

1 [20pts] Reading Materials on CNN

卷积神经网络(Convolution Neural Network,简称CNN)是一类具有特殊结构的神经网络，在深度学习的发展中具有里程碑式的意义。其中，Hinton于2012年提出的AlexNet可以说是深度神经网络在计算机视觉问题上一次重大的突破。

关于AlexNet的具体技术细节总结在经典文章"ImageNet Classification with Deep Convolutional Neural Networks", by Alex Krizhevsky, Ilya Sutskever and Geoffrey E. Hinton in NIPS'12，目前已逾万次引用。在这篇文章中，它提出使用ReLU作为激活函数，并创新性地使用GPU对运算进行加速。请仔细阅读该论文，并回答下列问题(请用1-2句话简要回答每个小问题，中英文均可)。

(a) **[5pts]** Describe your understanding of how ReLU helps its success? And, how do the GPUs help out?

(b) **[5pts]** Using the average of predictions from several networks help reduce the error rates. Why?

(c) **[5pts]** Where is the dropout technique applied? How does it help? And what is the cost of using dropout?

(d) **[5pts]** How many parameters are there in AlexNet? Why the dataset size(1.2 million) is important for the success of AlexNet?

关于CNN，推荐阅读一份非常优秀的学习材料，由南京大学计算机系吴建鑫教授[1]所编写的讲义Introduction to Convolutional Neural Networks[2]，本题目为此讲义的Exercise-5，已获得吴建鑫老师授权使用。

Solution.

(a)

[1] 吴建鑫教授主页链接为`cs.nju.edu.cn/wujx`

[2] 由此链接可访问讲义`https://cs.nju.edu.cn/wujx/paper/CNN.pdf`

(1)ReLU实际是max(0,x)，相较tanh和sigmoid等非线性函数，计算梯度更快。
(2)由于网络规模超出了单个GPU的存储能力，因此使用2块GPU的并行架构，在每个GPU上存储一半的kenel，这2块GPU只在特定的层上通信，这样耗时更少。

(b) 因为不同的模型在不同的特征上训练，得到的平均结果泛化能力更好。

(c)
（1）用在前两个全连接层。
（2）以0.5的概率将每个隐层神经元的输出设置为零,以这种方式屏蔽的神经元既不参与前向传播，也不参与反向传播。每次输入，该神经网络就尝试一个不同的结构，但是所有这些结构之间共享权重，所以网络要被迫学习更为鲁棒的特征，提高了泛化能力，防止过拟合。
（3）由于每次只能训练一半神经元，故需要双倍收敛迭代次数。
(d)
（1）一共有六千万参数。
（2）因为Alexnet的模型容量很大，若是图片数据不够很容易欠拟合，无法表征图像特征。而1.2million的数量足以支撑网络的训练而不至于欠拟合。

2 [20pts] Kernel Functions

(1) 试通过定义证明以下函数都是一个合法的核函数：

- (i) **[5pts]** 多项式核: $\kappa(\mathbf{x}_i,\mathbf{x}_j)=(\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j)^d$;
- (ii) **[10pts]** 高斯核：$\kappa(\mathbf{x}_i,\mathbf{x}_j)=\exp(-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{2\sigma^2})$, 其中$\sigma>0$.

(2) **[5pts]** 试证明$\kappa(\mathbf{x}_i,\mathbf{x}_j)=\frac{1}{1+e^{-\mathbf{x}_i^T\mathbf{x}_j}}$不是合法的核函数。

Proof.
（1）
（i）$(\mathbf{x}_i^T\mathbf{x}_j)^d=(x_{i1}y_{j1}+x_{i2}y_{j2}+\cdots)^d$是一个多项式，展开后是$\mathbf{x}_i\mathbf{y}_i$中分量的线性组合，必然能够表示成两个特征空间向量的内积$\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$

（ii）

$$
\begin{aligned}
\kappa(\mathbf{x}_i,\mathbf{x}_j) &= \exp(-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{2\sigma^2}) \\
&= \exp(-\frac{(\mathbf{x}_i-\mathbf{x}_j)^T(\mathbf{x}_i-\mathbf{x}_j)}{2\sigma^2}) \\
&= \exp(-\frac{\mathbf{x}_i^T\mathbf{x}_i-\mathbf{x}_j^T\mathbf{x}_i+\mathbf{x}_j^T\mathbf{x}_j}{2\sigma^2}) \\
&= \exp(-\frac{\|\mathbf{x}_i\|^2+\|\mathbf{x}_j\|^2-2\mathbf{x}_i^T\mathbf{x}_j}{2\sigma^2}) \\
&= C\exp(\frac{\mathbf{x}_i^T\mathbf{x}_j}{\sigma^2}) \\
&= C\sum_{n=0}^{\infty}\frac{(\mathbf{x}_i^T\mathbf{x}_j)^n}{n!}(U\Gamma) \\
&= C\sum_{n=0}^{\infty}\frac{\kappa_{poly}(\mathbf{x}_i,\mathbf{x}_j)}{n!}
\end{aligned}
$$

可以看到展开的形式仍是多项式，是输入分量的线性组合，可以表示成两个特征空间向量的内积$\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$。
（2）通过构建合适的输入使其核矩阵不满足半正定特性，通过Mercer定理即可证明其是不合法的核函数。这里输入$\mathbf{x}_i=(1,1),\mathbf{x}_j=(2,2)$，核矩阵为

$$
\begin{bmatrix}
\frac{1}{1+\exp(-2)} & \frac{1}{1+\exp(-4)} \\
\frac{1}{1+\exp(-4)} & \frac{1}{1+\exp(-8)}
\end{bmatrix}
\tag{2.1}
$$

其最大余子式的值为$-0.08<0$,故其核矩阵不是半正定矩阵，故其是不合法的核函数。

□

3 [25pts] SVM with Weighted Penalty

考虑标准的SVM优化问题如下(即课本公式(6.35))，

$$
\begin{aligned}
\min_{\mathbf{w},b,\xi_i} \quad & \frac{1}{2}\|\mathbf{w}\|^2+C\sum_{i=1}^{m}\xi_i \\
\text{s.t.} \quad & y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i+b)\geq 1-\xi_i \\
& \xi_i\geq 0, i=1,2,\cdots,m.
\end{aligned}
\tag{3.1}
$$

注意到，在(3.4)中，对于正例和负例，其在目标函数中分类错误的“惩罚”是相同的。在实际场景中，很多时候正例和负例错分的“惩罚”代价是不同的，比如考虑癌症诊断，将一个确实患有癌症的人误分类为健康人，以及将健康人误分类为患有癌症，产生的错误影响以及代价不应该认为是等同的。

现在，我们希望对负例分类错误的样本(即false positive)施加$k > 0$倍于正例中被分错的样本的“惩罚”。对于此类场景下，

(1) **[10pts]** 请给出相应的SVM优化问题;

(2) **[15pts]** 请给出相应的对偶问题，要求详细的推导步骤，尤其是如KKT条件等。

Solution.

（1）

$$\begin{aligned} \min_{\mathbf{w},b,\xi_i} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C_+\sum_{i=1}^{p}\xi_i + kC_+\sum_{i=p+1}^{m}\xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0, i = 1,2,\cdots,m. \end{aligned} \tag{3.2}$$

(2)运用拉格朗日乘子法可以得到拉格朗日函数

$$L(\mathbf{w},b,\alpha,\xi,\mu) = \frac{1}{2}\|\mathbf{w}\|^2 + C_+\sum_{i=1}^{p}\xi_i + kC_+\sum_{i=p+1}^{m}\xi_i + \sum_{i=1}^{m}\alpha_i(1-\xi_i-y_i(\mathbf{w}^T\mathbf{x}_i+b)) - \sum_{i=1}^{m}\mu_i\xi_i \tag{3.3}$$

令拉格朗日函数对$\mathbf{w}, b, \mathbf{xi_i}$求偏导为0可得

$$\begin{aligned} \mathbf{w} &= \sum_{i=1}^{m}\alpha_i y_i \mathbf{x}_i \\ 0 &= \sum_{i=1}^{m}\alpha_i y_i \\ C_+ &= \begin{cases} \alpha_i + \mu_i & 1 \leq i \leq p \\ \frac{\alpha_i+\mu_i}{k} & p+1 \leq i \leq m \end{cases} \end{aligned} \tag{3.4}$$

将（3.4）带入(3.3)得到对偶问题

$$\begin{aligned} & \max_{\alpha}\sum_{i=1}^{p}(\alpha_i+\mu_i)\xi_i + \sum_{i=p+1}^{m}k\times\frac{\alpha_i+\mu_i}{k}\xi_i + \sum_{i=1}^{m}\alpha_i(1-\xi_i) - \sum_{i=1}^{m}\mu_i\xi_i \\ & = \max_{\alpha}\sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j \\ & \text{s.t.} \sum_{i=1}^{m}\alpha_i y_i = 0 \\ & 0 \leq \alpha_i \geq C_+, i = 1,2,\cdots,p. \\ & 0 \leq \alpha_i \geq kC_+, i = p+1,\cdots,m. \end{aligned}$$

KKT条件要求

$$
\begin{cases}
\alpha_i \geq 0, \mu_i \geq 0 \\
y_i f(\mathbf{x}_i) - 1 + \xi_i \geq 0 \\
\alpha_i (y_i f(\mathbf{x}_i - 1 + \xi_i)) = 0 \\
\xi_i \geq 0, \mu_i \xi_i = 0
\end{cases}
\tag{3.5}
$$

4 [35pts] SVM in Practice - LIBSVM

支持向量机(Support Vector Machine，简称SVM)是在工程和科研都非常常用的分类学习算法。有非常成熟的软件包实现了不同形式SVM的高效求解，这里比较著名且常用的如LIBSVM[3]。

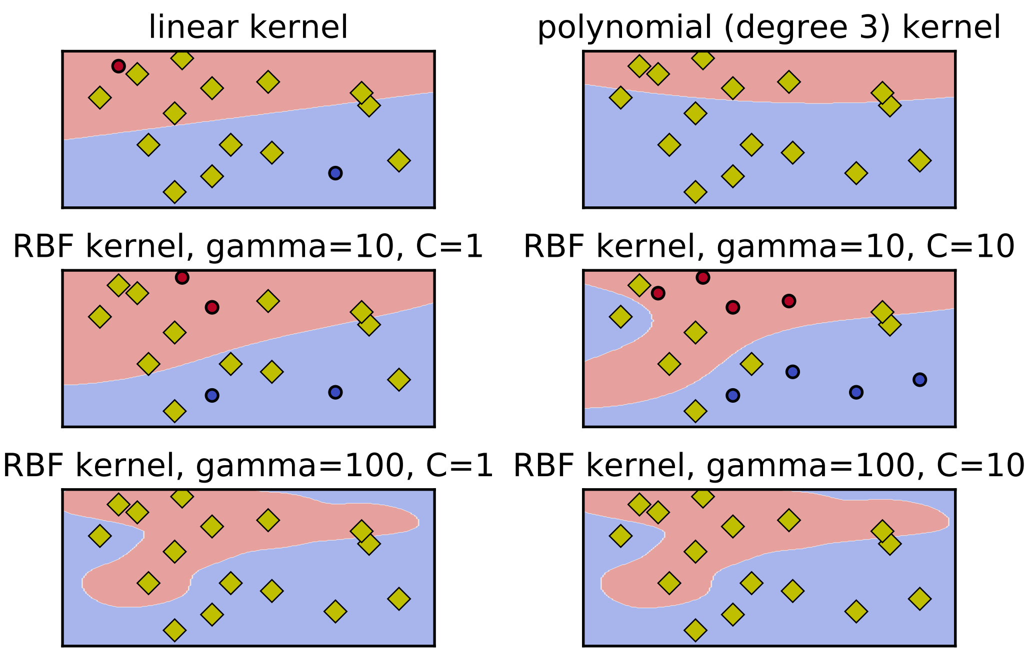
(1) **[20pts]** 调用库进行SVM的训练，但是用你自己编写的预测函数作出预测。

(2) **[10pts]** 借助我们提供的可视化代码，简要了解绘图工具的使用，通过可视化增进对SVM各项参数的理解。详细编程题指南请参见链接：`http://lamda.nju.edu.cn/ml2017/PS4/ML4_programming.html`.

(3) **[5pts]** 在完成上述实践任务之后，你对SVM及核函数技巧有什么新的认识吗？请简要谈谈。

Solution.

(2)



[3]LIBSVM主页课参见链接：`https://www.csie.ntu.edu.tw/~cjlin/libsvm/`

（3）SVM训练过程中，无论有无松弛项，都只有支持向量才能发挥作用。