# SDU\_NLP\_2020期中考试

姓名：

学号：

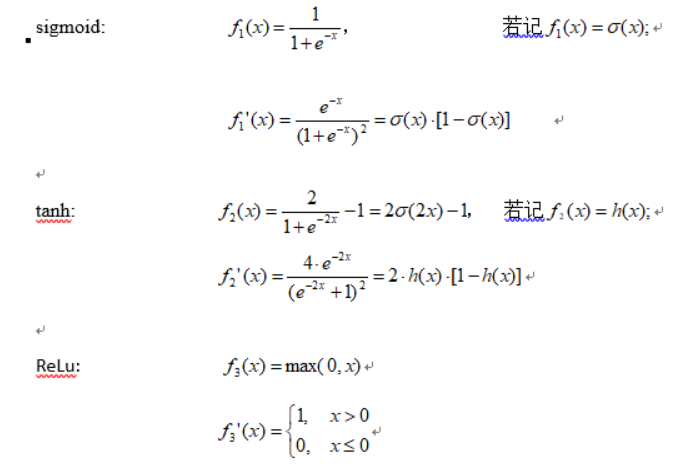
## 一、激活函数（10分）

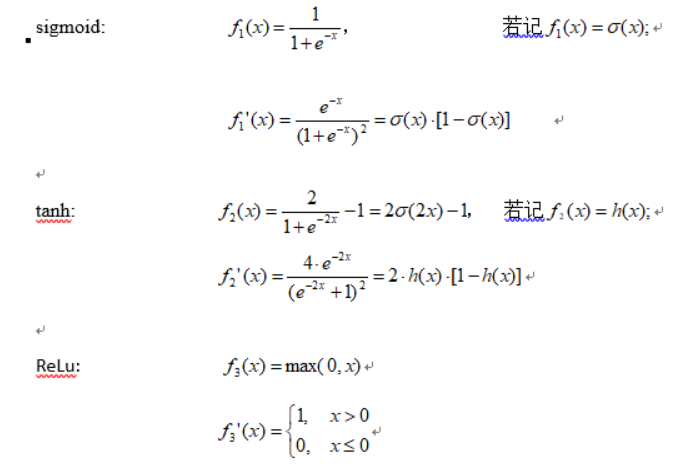
1. 推导常见激活函数如sigmoid函数，tanh函数和Relu函数的导数，其中

sigmoid的函数形式为，

Relu函数形式为R（x）=max(0,x)。（3分）

答：





1. 证明针对任何输入向量和常数c，softmax函数的输出不会随着输入向量偏移（也就是常数c）而改变。即：

其中就是给每个元素加上常数c。注意 （3分）

答：针对所有维度1<=i<=dim()

1. 已知一个N行d列的输入矩阵x，计算每一行的softmax概率。请补全代码

import numpy as np

def softmax(x):

"""

#此处补全代码

"""

return x （4分）

答：

assert len(x.shape) > 1, "Softmax的得分向量要求维度高于1"

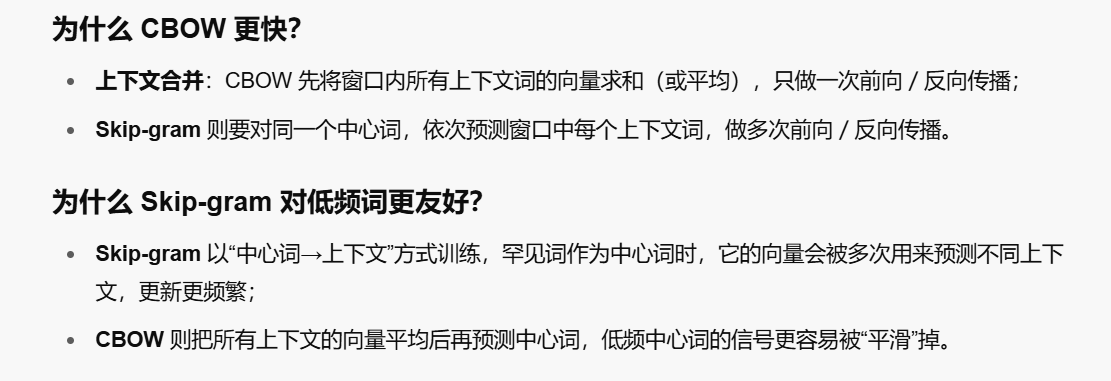
x -= np.max(x, axis=1, keepdims=True)

x = np.exp(x) / np.sum(np.exp(x), axis=1, keepdims=True)

## 二、算法类对比（7分）

1. 对比CBOW和skip-gram的区别？

答：(1) 训练速度上 CBOW 应该会更快一点。因为每次会更新 context(w) 的词向量，而 Skip-gram 只更新核心词的词向量。两者的预测时间复杂度分别是 O(V)，O(KV)。(2) Skip-gram 对低频词效果比 CBOW好。因为是尝试用当前词去预测上下文，当前词是低频词还是高频词没有区别。但是 CBOW会选择最常见或者说概率最大的词来补全，因此不太会选择低频词。Skip-gram 在大一点的数据集可以提取更多的信息。SG 总体比 CBOW 要好一些。



## 三、模型细节内部（7分）

1. 请简述在Word2Vec训练过程中，优化方法负采样的采样步骤？

答：如果词汇表的大小为V,那么我们就将一段长度为1的线段分成V份，每份对应词汇表中的一个词。当然每个词对应的线段长度是不一样的，高频词对应的线段长，低频词对应的线段短。每个词w的线段长度由下式决定： 在采样前，我们将这段长度为1的线段划分成M等份，这里M>>V，这样可以保证每个词对应的线段都会划分成对应的小块。而M份中的每一份都会落在某一个词对应的线段上。在采样的时候，我们只需要从M个位置中采样出neg个位置就行，此时采样到的每一个位置对应到的线段所属的词就是我们的负例词。在word2vec中，M取值默认为108

## 四、词向量的评价指标（6分）

1. 词向量的评价方法有哪些？其价值和局限分别是什么？

答：词向量的评价方法可以分为内部评价和外部评价。内部评价是在一个中间任务上评价词向量的好坏，包括词向量类比实验，直接计算余弦相似度等。该方法计算速度快，能帮助我们更好地理解系统。但是在内部评价上表现好并不意味着词向量在真实任务中会有好的效果。外部评价是在一个真正的NLP任务中使用词向量，以此来评判词向量的好坏。但是计算外部评价结果会消耗很长的时间。即使出现了问题，我们也不清楚是词向量的问题还是其他子任务的问题。

## 五、python基础（9分）

1. python语言的优点有哪些？（5分）

答：优势：

1.便于理解和编写，语法与英文相似

2.可以在不同系统上执行（Windows，Mac，Linux等）

3.面向对象语言，带来编码的可重用性、可维护性

4.有强大的标准库

5.动态数据类型的优势:不需要预先声明变量类型

1. 给定张量A和张量B，他们的形状分别为 (2,1,5,3)和(3,3,1)，如果调用numpy.matmul(A,B)是否满足广播机制的条件。如果是写出输出张量的形状。（4分）

答: 1.满足，因为后两位(5,3)和(3,1)可做矩阵乘法，剩余部分(2,1),(3,)也满足条件。

2.输出张量形状为(2,3,5,1)

## 六、词窗口分类（6分）

1. 在softmax分类器中，假设输入向量为x，模型权重,（C、d分别为类别个数和x的维度），写出计算的过程。（6分）

**答：**

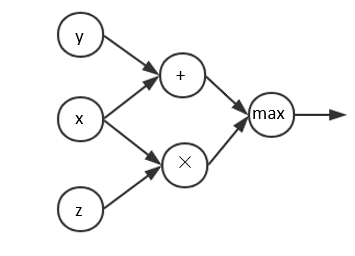
**1. 或文字叙述，计算输入x与W第y行的向量内积。**

**2.对于c=1,2,…,C计算**

**3..=softmax() 或文字叙述，用softmax求fy归一化后输出的概率。**

## 七、反向传播算法（15分）

1. 有表达式,且初始化。利用反向传播算法，计算输出f对输入x,y,z的偏导数，写出中间计算过程。（9分）



**答：**

**1.计算正向值的传播， (1分，符号a、b可以自定义)**

**2.计算局部梯度(local gradient)**

**3.计算梯度反向传播，,**

**,**

1. 补全下列关于反向传播算法应用在深度神经网络中的伪代码中的空缺：（6分）

假设有多层前馈神经网络：

网络深度：

模型权重矩阵：

模型偏置参数：

程序输入:x

目标输出:y

有前向传播过程(不考虑正则化项):

#补全下划线，f为激活函数

后向传播过程:

#补全下划线，关于偏置的梯度

#补全下划线，梯度向下一隐层传播。

**答：**



## 八、依存句法分析（15分）

1. 给定句子“scientists count whales from space”，请给出两种句法分析结构，形式为（count- subj - scientists），并说明歧义类型（3分）

答： 第一种：（count- subj - scientists），（count-obj-whales），（count-nmod-space）（1分）

第二种：（count- subj - scientists），（count-obj-whales），（whales-nmod-space）（1分）

歧义类型：介词短语依附歧义 （1分）

1. 下图1是正确的依存关系结果，图2是某依存分析器给出的结果（4分）

图2

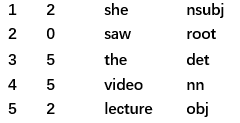
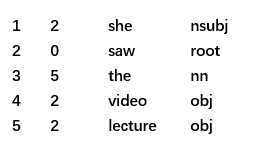


图1

请计算如下性能指标：

1. 无标记依存正确率（UAS）

答：80%

1. 带标记依存正确率（LAS）

答：60%

1. 给定句子“I ate fish”，使用transition-based方法进行依存分析，请给出时每一步的栈、缓冲区以及A的状态。（8分）

答：

1. 初始状态：

Stack：root

Buffer：I ate fish

A：空集

1. 进行shift 操作：

Stack：root I

Buffer : ate fish

A: 空集

1. 进行shift 操作：

Stack：root I ate

Buffer : fish

A: 空集

1. 进行Left- arc操作：

Stack：root ate

Buffer：fish

A：nsubj（ate-I）

1. 进行shift 操作：

Stack：root ate fish

Buffer : 空

A: nsubj（ate-I）

1. 进行Right- arc 操作：

Stack：root ate

Buffer : 空

A: nsubj（ate-I），obj（ate-fish）

1. 进行Right- arc 操作：

Stack：root

Buffer : 空

A: nsubj（ate-I），obj（ate-fish），root（root-ate）

1. 结束

## 九、语言模型与RNN（15分）

1、请写出在马尔科夫假设下，计算的n gram 公式（5分）

答：

1. 请写出RNN的前向传播公式，并给出公式中参数的含义（10分）

答：

词嵌入矩阵

：含有 T 个单词的语料库对应的离散的词。

：每个时间步 t 的隐藏层的输出特征的计算关系

：在时间步 t 的对应的词向量。

：输入词向量对应的权重矩阵。

：上一个时间步的输出对应的权重矩阵。

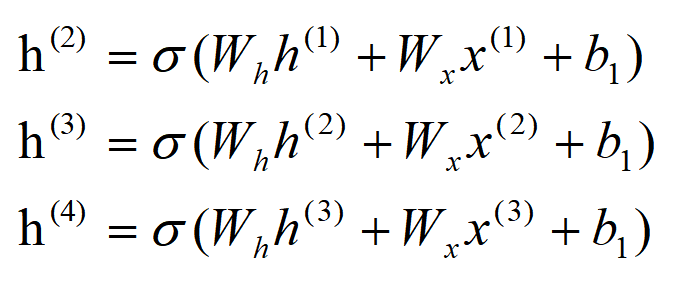
：上一个时间步 t-1 的非线性函数输出。 是在时间步 t=0 的隐藏层的一个初始化向量。

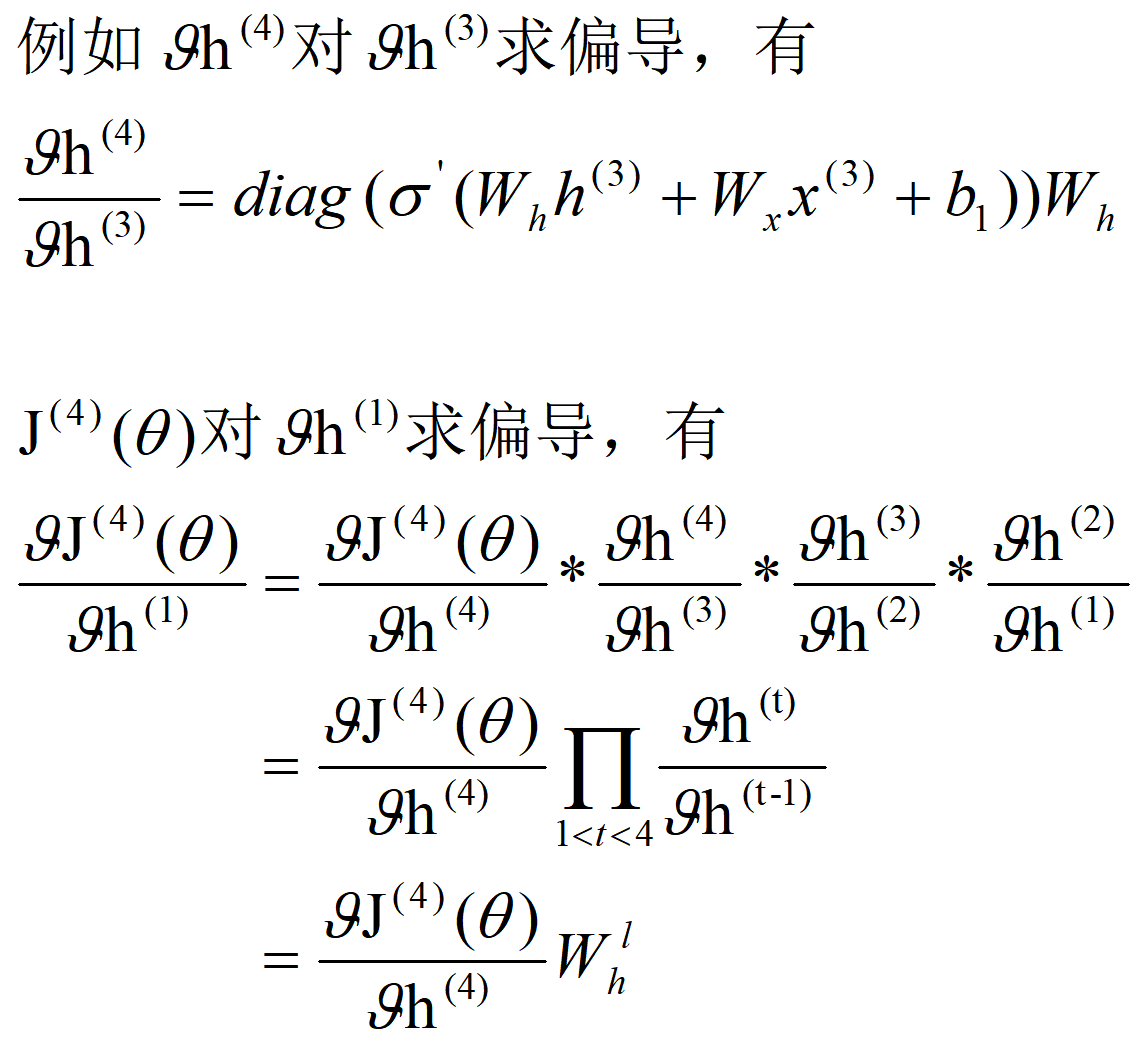
σ：非线性函数（这里是 sigmoid 函数）。

## 十、梯度消失和LSTM（10分）

1.请说明梯度消失的原因以及梯度消失的坏处？(5分)

答：层数较深的神经网络模型进行反向传播时容易出现梯度消失问题。例如在以下前向传播中



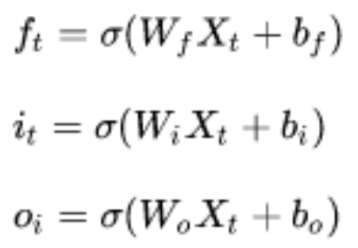


当i-j很大时，**累乘的结果会很小，从而导致梯度消失**。导致**层数比较浅的权重没有更新。**

2.为什么说LSTM在一定程度上解决了梯度消失？(5分)

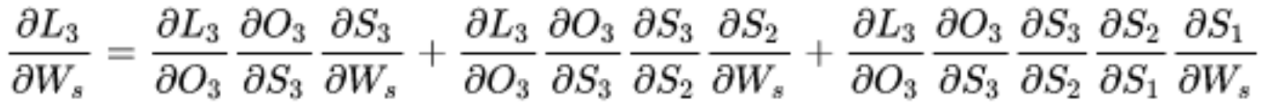
答：

LSTM有如下公式

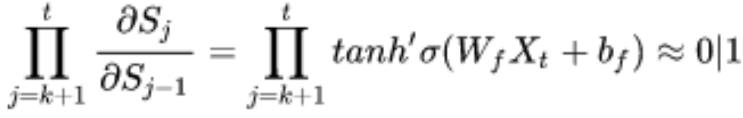




例如L3在对Ws求导时，有



因为经过LTSM中门的控制，使得



因此，**L3在对Ws求导中间的累乘为0-1的值，并可以根据网络学习到的知识选择在多大程度上对梯度进行传递**。