**Python数据分析在COVID-19预测中的应用**

——利用LSTM算法对新增确诊趋势的预测

姓名： 刘森林

武汉大学

电子信息学院

2021.7

# LSTM算法简介

## Recurrent Neural Networks(RNN)

举个生动形象的例子：今天我跑了十公里，明天肯定会腿酸——这是短时间内的对应关系。在这种短时间的结构中，我们只需传入上一级的状态，再结合当前输入，得到当前的输出，如图所示。

手机屏幕截图

低可信度描述已自动生成

## LSTM Networks (for long-term dependencies)

再举另外一个例子：对于数据结构课，如果该学图算法的时候不学，考前也不学，那么期末考试中关于图的题目必然不会做。与上个例子不同，这种情况属于长期和短期的对应，一个月前的行为和近几天的行为都会对结果产生影响。为了描述这种对应关系，研究人员在RNN的基础上提出了LSTM模型，用于描述长短期的对应关系。

图示

描述已自动生成

接下来对LSTM的数学模型做一个更详细的解释。

白板上的文字

描述已自动生成

C^(t-1)代表长期记忆状态，它是由第一个细胞的状态经过所有之前的记忆细胞传到当前，因此包含有长期记忆效应。

H^(t-1)是上一级传来的短期记忆状态，它的维度就是隐藏节点数——hidden layer size。

X^t是当前输入，Y^t是当前输出。

图片包含 图表

描述已自动生成

首先使用LSTM的当前输入x^t 和上一个状态传递下来的h^(h-1)拼接训练得到四个状态值

对应上面手画草图理解，通过将输入x^t 和上个时刻隐状态h^(t-1)拼接后乘以权重参数，再通过sigmoid激活函数转换到0到1间的数值，来作为一种门控状态值。而 z, 则是将结果通过一个 tanh激活函数将转换成-1到1之间的值。其中W, W^i, W^f, W^o 就是模型要学习的权重参数，x^t的维度自定义，在NLP中即就是词向量的维度，每一列代表一个词向量，维度自定义。矩阵的列数为一个句子的token个数，也是time\_step length。LSTM在自然语言处理的应用中，每个sentence的每个token的embedding(词向量)对应了每个时序 t 的输入。如output, (hn, cn) = lstm(input, (h0, c0))，time\_step=1时，input为第一个词的词向量，词向量中的每一个元素对应输入层的每个节点（每个神经元），常看到hidden\_size就是我们的隐藏节点（神经元），根据上面公式可看出input\_size+hidden\_size肯定等于权重参数的列数。

文本, 信件

描述已自动生成

最后是LSTM结构单元的输出：忘记阶段，遗忘门z^f, 主要对上一个时刻cell状态c^(t-1) 进行选择性遗忘，即将不重要的忘记。

记忆阶段，输入门z^i, 主要是对当前这个阶段的输入进行‘记忆’，当前的输入内容由前面计算得到的z来表示，输入门控由z^i来表示。

将上面两步得到的结果相加，即可得到传输给下个状态的长时记忆c^t, 也就是上面公式的第一个公式

输出阶段，输出门z\*o, 主要是决定哪些c^t 将会被当成当前隐状态的输出，通过输出门z^o 来进行控制。并且还需要对c^t进行缩放(通过一个tanh激活函数进行变化)

# 预测算法的实现过程

## 2.1导入相应的包和模块

文本

描述已自动生成

## 2.2数据获取及预处理

从约翰霍普金斯大学网站上获取美国累计确诊病例，读入CSV文件，将528天的累计确诊数据变换为单日新增确诊数据。

文本

描述已自动生成

设置训练集和验证集的数据，选取最后二十天作为验证集

文本

描述已自动生成

训练集做归一化处理

文本

描述已自动生成

得到一个输入输出列表，每个元素是每个元素是(长度为20)训练序列和训练标签(21)的元组。文本

描述已自动生成

## 2.3建立LSTM模型

1. 初始化，传参

文本

描述已自动生成

2.前向传播

文本

描述已自动生成

3.训练

文本

描述已自动生成

4.预测

文本

描述已自动生成

5.图形绘制

文本

描述已自动生成

# 结果分析及个人思考

## 结果分析

（以下图片是**隐藏节点数为45**的结果）

可以清晰地看到，训练过程中，损失函数在递减

文本

描述已自动生成

我们通过前508天的数据做训练和测试，对最后二十天的新增确诊人数曲线做预测（红色为预测，蓝色为真实）。当隐藏节点数为45时，曲线拟合较好。

图表, 折线图

描述已自动生成

## 个人思考

1. LSTM模型最好用来预测**单日新增曲线**而非累计确诊曲线。

当我们用LSTM模型预测累计确诊人数时,由于曲线上升过快，每个单元的输入梯度过大。而LSTM具有**反梯度爆炸机制**，过大的梯度会导致门控装置的关闭，因此梯度无法传播，导致预测结果趋于水平，与实际走势大相径庭。

图表, 折线图

描述已自动生成

如果我们预测的是单日新增确诊曲线，由于这条曲线的上升梯度始终趋于平缓，不会出现反梯度爆炸的现象，预测结果则会正常很多。

图表, 折线图

描述已自动生成

1. LSTM预测曲线很大程度上取决于**隐藏节点数**。

当我设置不同的**隐藏节点数**参数时，得到的结果也出现了很大的差异。

a.隐藏节点数较小时（5~10）

隐藏节点数hidden\_layer\_size=5时：

图表, 折线图

描述已自动生成

H\_l\_s=7 H\_l\_s=10

图表, 折线图

描述已自动生成 图表, 折线图

描述已自动生成

H\_l\_s较小时，得到的预测曲线都比较平滑，但误差很大。

b. h\_l\_s在30-45时得到的结果较为满意

h\_l\_s=30 H\_l\_s=40

图表, 折线图

描述已自动生成 图表, 折线图

描述已自动生成

H\_l\_s=45

图表, 折线图

描述已自动生成

由此可见，在此案例的分析中，应该在30-45之间选择隐藏节点数

c. h\_l\_s大于50出现明显的过拟合现象

h\_l\_S=50 h\_l\_s=70

图表, 折线图

描述已自动生成 图表, 折线图

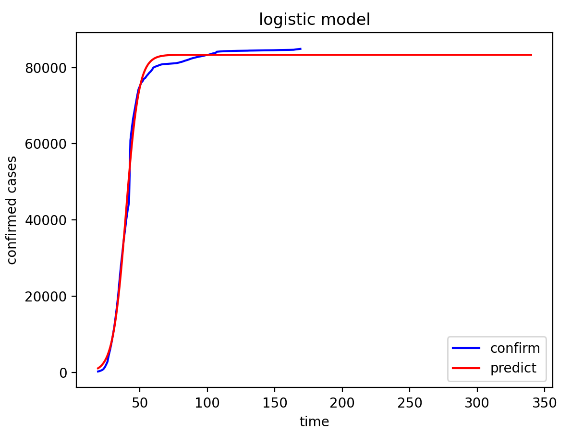
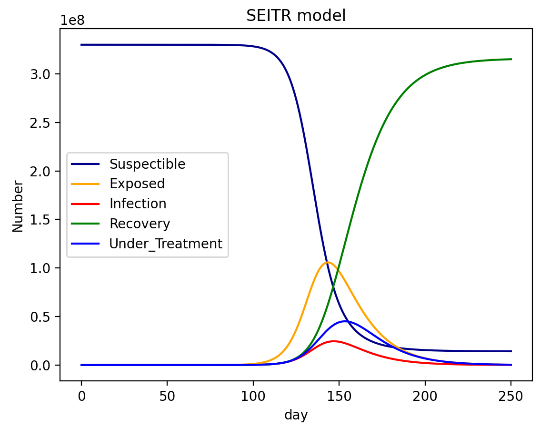
描述已自动生成

1. 对比分析：关于**LSTM**与**传统流行病学模型**。

传统流行病学模型的产生都是在长期在病毒传播机制的基础上总结出的经验和规律，具有较强的可靠性。

而LSTM看似是一种能够通过“学习”过去数据从而预测未来的具有普适性的复杂网络，实则在很多应用场景下具有相当的局限性。比如疫情预测的案例中，任何传染病在一个地区的传播必然经过从爆发到趋于平缓最后清零的过程，因此最终的总体趋势一定是逻辑斯蒂曲线。而LSTM不能通过已有的数据理解这种机制。

Logistic model result: SEITR model result:

(如图)**逻辑斯蒂模型**和**SEITR模型**等传统流行病学模型都能得到较为可靠的预测结果*，****符合疫情的总体走势***

再比如，我们用LSTM模型预测中美疫情走势都会出现两种截然相反的结果。两者都用前五十天的数据做训练，LSTM不会预料到中国政府的管控如此得力，清零的速度如此之快；LSTM也不会预想到美国政府疏于管控，美国疫情从最初几乎没有确诊演变成蔓延全国、迟迟不退的史上最大公共卫生事件。

## 4课程结语和感想

首先非常感谢曾老师在本课程中的付出，让我对python数据分析有了较为系统的应用和掌握，我也在一个星期内收获了许多之前没有接触过的方法和操作，比如网页爬虫，我不仅学会了疫情数据“字典”式的爬虫，也学习了新闻网页链接和标题的爬虫方法，并将其应用在同样是本周结课的公选，可以说极大促进了个人的历练和能力的提升。

再比如传统的传染病模型实现，让我对如何建模并通过代码实现最终可视化的完整过程有了深刻的体会，这种数学建模的操作流程是今后收益终生的。

最后，我实现了LSTM模型对疫情的预测，并做了我力所能及的分析。由于之前自己了解过深度学习，所以我在实现的过程中更加侧重对数学表达式的分析，以及模型中关键参数的改变对结果的影响的分析。总体而言，得到了满意的结果！