

实 验 报 告

|  |  |
| --- | --- |
| **课程名称：** | 大数据分析基础训练 |
| **项目名称：** | 澳元对人民币汇率的大数据分析与预测 |
| **姓 名：** | 余力 |
| **学 号：** | 2016034643040 |
| **班 级：** | 16信计 |
| **指导教师：** | 吴世枫 |

**数学与系统科学 学院 工业中心 实验室**

二〇一九 年 十二 月 二十八 日

**广东技术师范学院实验报告**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 学院： | 数学与系统科学学院 | | | 专业： | | 16信息与计算科学 | | | 班级： | |  | | 成绩： |  |
| 姓名： | 余力 | | 学号： | | 2016034643040 | | | 组别： | |  | | 组员： |  | |
| 实验地点： | | 工业中心 | | | 实验日期： | | 2019.12.28 | | | | 指导教师签名： | | |  |

项目名称： 澳元对人民币汇率的大数据分析与预测

目录

[一、 实验项目名称 7](#_Toc28590455)

[二、 实验说明 7](#_Toc28590456)

[2.1 实验内容 7](#_Toc28590457)

[2.2 实验知识点 7](#_Toc28590458)

[2.2.1 Python基础知识 7](#_Toc28590459)

[2.2.2 Python基础库 7](#_Toc28590460)

[2.2.3 Tensorflow模块 8](#_Toc28590461)

[2.2.4 RNN和Lstm神经网络 8](#_Toc28590462)

[2.2.5 一些经济学理论 9](#_Toc28590463)

[2.3 实验环境 10](#_Toc28590464)

[2.4 适合人群 10](#_Toc28590465)

[2.5 代码获取 10](#_Toc28590466)

[三、 开发准备 10](#_Toc28590467)

[四、 项目文件结构 11](#_Toc28590468)

[五、 实验内容及步骤 11](#_Toc28590469)

[5.1 导入实验所需的Python库 11](#_Toc28590470)

[5.2 导入数据及处理 11](#_Toc28590471)

[5.3 设置参数 12](#_Toc28590472)

[5.4 样本生成函数 13](#_Toc28590473)

[5.5 构建Lstm模型 14](#_Toc28590474)

[六、 实验数据记录和处理 14](#_Toc28590475)

[6.1 模型保存 14](#_Toc28590476)

[6.2 数据处理 14](#_Toc28590477)

[6.3 训练数据 15](#_Toc28590478)

[七、 实验结果与分析 15](#_Toc28590479)

[7.1 预测测试样本 15](#_Toc28590480)

[7.2 计算准确率 18](#_Toc28590481)

[7.3 预测未来一个月的值 18](#_Toc28590482)

[八、 问题与讨论 20](#_Toc28590483)

[8.1 分析结果 20](#_Toc28590484)

[8.2 不足与改进 20](#_Toc28590485)

1. 实验项目名称

本实验研究的是澳元对人民币汇率的大数据分析与预测。

1. 实验说明

2.1 实验内容

利用所提供的数据，包括从2000年至2018年每天汇率的变化，通过Python等工具，建立数学模型，实现汇率变化的数据分析及预测。本实验将采用Tensorflow和Lstm搭建模型，预测的时间是未来一个月。

2.2 实验知识点

通过本实验，应该掌握以下几个知识点：Python基础以及基础库的知识；Tensorflow模块的应用；RNN和Lstm神经网络的知识；一些基础经济学理论。下面将一一介绍。

2.2.1 Python基础知识

Python 是一个高层次的结合了解释性、编译性、互动性和面向对象的脚本语言。它的设计具有很强的可读性，相比其他语言经常使用英文关键字，其他语言的一些标点符号，它具有比其他语言更有特色语法结构。Python是一种集解释型、交互式、面向对象等特点于一身的语言，非常适合初学者学习，它支持广泛的应用程序开发，从简单的文字处理到 WWW 浏览器再到游戏都容易上手。

2.2.2 Python基础库

针对此实验，需要掌握如下几个python库的应用：numpy、matplotlib、panda。

Numpy库是Python的一种开源的数值计算扩展。这种工具可用来存储和处理大型矩阵，比Python自身的嵌套列表结构要高效的多，它可用来生成常见的数组，甚至可以用来进行逻辑运算、统计运算等数组剑的运算。

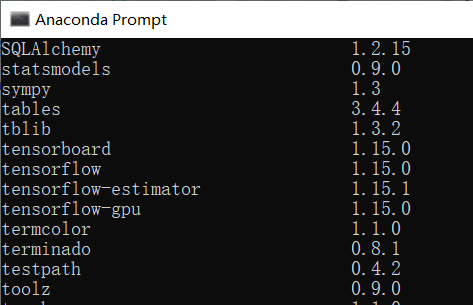
Matplotlib是一个很强大的2D画图库，它以各种硬拷贝格式和跨平台的交互式环境生成出版质量级别的图形，通过 Matplotlib，开发者可以仅需要几行代码，便可以生成绘图，直方图，功率谱，条形图，错误图，散点图等。它非常实用，几乎在所有需要画图的代码中都可以用到它。

Panda是一个强大的分析结构化数据的工具集；它的使用基础是Numpy；用于数据挖掘和数据分析，同时也提供数据清洗功能。它可以创建一个非常牢固的用于数据挖掘与分析的基础，在分析数据时非常实用。它基于两种类型：series与dataframe。series是一个一维的数据类型，其中每一个元素都有一个标签；而dataframe是一个二维的表结构，它可以存储许多种不同的数据类型，并且每一个坐标轴都有自己的标签。

2.2.3 Tensorflow模块

这是这个实验的第一个重点，也是实验成功必不可少的一个库，在了解它之前，我们先得了解Anaconda。

Anaconda 是专门为了方便使用 Python 进行数据科学研究而建立的一组软件包，涵盖了数据科学领域常见的 Python 库，并且自带了专门用来解决软件环境依赖问题的 conda 包管理系统。conda可以理解为一个工具，也是一个可执行命令，其核心功能是包管理与环境管理。包管理与pip的使用类似，环境管理则允许用户方便地安装不同版本的python并可以快速切换。Anaconda则是一个打包的集合，里面预装好了conda、某个版本的python、众多packages、科学计算工具等等，所以也称为python的一种发行版，它里面包含着python中各式各样的库，在控制台中输入pip list即可查询你所安装的库有多少。

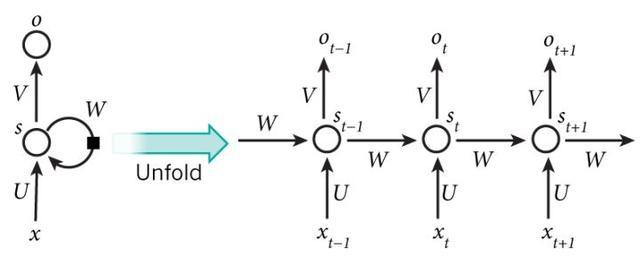
在我的电脑中，已安装了anaconda，所以自然就配置好了python环境，上述所提到过的基础库如numpy、panda等都有，但是tensorflow没有，所以需自行安装。我安装的版本是1.15.0版本，本来最新更新的版本已经到了2.0，但是鉴于2.0版本无法使用contrib模块，而contribute模块有着非常强大的功能，且本实验也会用到，所以只得降低tensorflow的版本，使contirb兼容，且附带的tensorflow-estimator和tensorflow-gpu这两个模块也得和本体版本相同，不然无法使用，如图所示：

那么Tensorflow是干嘛用的呢？TensorFlow 是目前最流行的深度学习库，它是 Google 开源的一款人工智能学习系统。Tensor 的意思是张量，代表 N 维数组；Flow 的意思是流，代表基于数据流图的计算。把 N 维数字从流图的一端流动到另一端的过程，就是人工智能神经网络进行分析和处理的过程。它与机器学习和神经网络的内容息息相关，而TensorFlow所支持CNN、RNN和Lstm的算法，正是目前最流行的深度神经网络模型。所以说它对于本实验的帮助是非常大的，因为我们就是要利用神经网络来对大数据进行分析和预测。

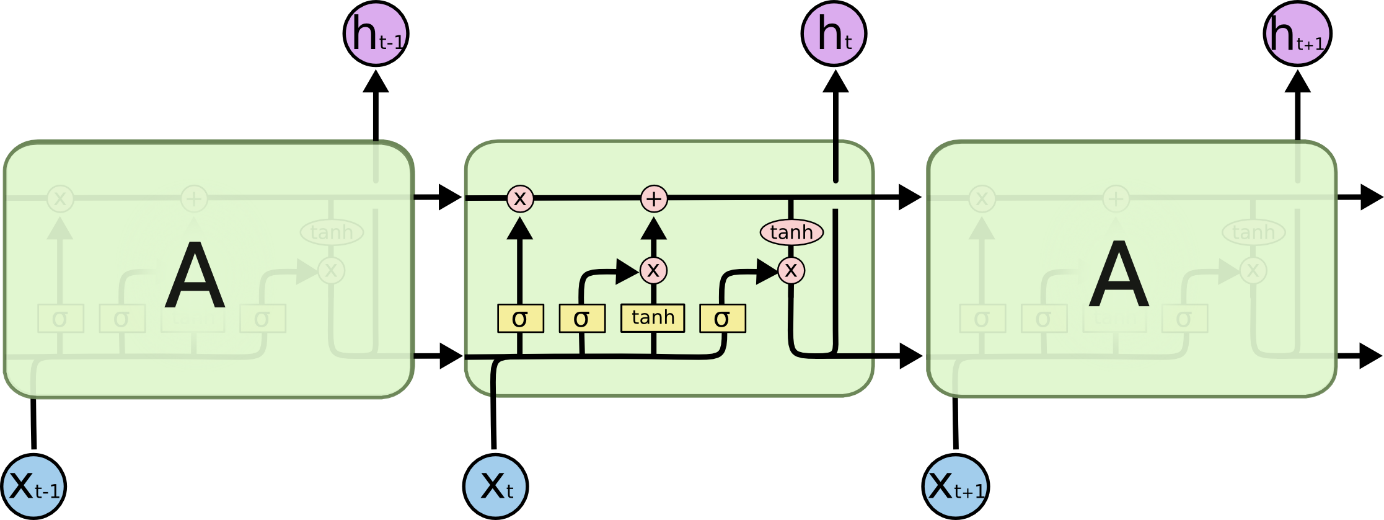
2.2.4 RNN和Lstm神经网络

接下来就进入到本实验的第二个重点——深度神经网络。在之前的学习中，我们都学习过CNN，即卷积神经网络，它包含着输入层，输出层，隐含层等基本结构，隐含层中还包含着卷积层、池化层等内部构造。针对它我们也做过一些项目，因此也有一定的神经网络的基础。而这一次的RNN和CNN很类似。

RNN即循环神经网络，是一类以序列数据为输入，在序列的演进方向进行递归且所有节点按链式连接的递归神经网络。这是专业概念，我们先来了解什么是序列数据：它是指在不同时间点上收集到的数据，这类数据反映了某一事物、现象等随时间的变化状态或程度，其特点是后面的数据跟前面的数据是相关联的。RNN利用了这些顺序信息，针对系列中的每一个元素都执行相同的操作，而每一个操作都依赖于之前的计算结果，从而形成了循环。这就是RNN的工作原理，它可以将信息持久化，结构图如下：



而我们要使用的Lstm网络，即长短期记忆网络，是一种特殊的RNN的变体，RNN由于梯度消失的原因只能有短期记忆，Lstm网络通过精妙的门控制将短期记忆与长期记忆结合起来，并且一定程度上解决了梯度消失的问题。所有RNN都有一种重复神经网络模块的链式的形式，这个重复的模块只有一个简单的结构——tanh层。但Lstm中重复的模块有一个不同的结构，不同于 单一神经网络层，这里是有四个，以一种非常特殊的方式进行交互，如下图：



Lstm 的关键就是细胞状态，它通过精心设计的被称为“门”的结构来去除或者增加信息到细胞状态的能力。门是一种让信息选择式通过的方法。他们包含一个 sigmoid 神经网络层和一个 pointwise 乘法操作。Lstm拥有三个门，分别是遗忘门，输入门，输出门。遗忘门能决定应丢弃或保留哪些信息，输入门用来更新单元状态，输出门能决定下个隐藏状态的值，隐藏状态中包含了先前输入的相关信息。这就是简单的Lstm介绍，具体的内容还需进行深度学习才能了解。

2.2.5 一些经济学理论

汇率是指不同货币买卖之间的兑换比率，即一个国家的货币折算为另一个国家货币的价格，它分为即期汇率和远期汇率。即期汇率是指目前的汇率，用于外汇的现货买卖，是外汇买卖双方在成交当日或两天以内交割使用的汇率，也称现汇汇率；远期汇率是指现在由外汇买卖双方签订元气和同事规定的、在未来一定时期进行交割时使用的汇率。如果远期汇率高于即期汇率，则称远期汇率升水，反之则称贴水；如二者相等，则称平价。

在某一特定时点，某一外币的供给量正好等于需求量，这一汇率就是均衡汇率。而在外汇市场中，如果外币的供给量、需求量、发生变动，原有汇率的均衡状况会被打破，新的均衡汇率就会形成，从原有均衡状态达到新的汇率均衡，就叫汇率波动。

2.3 实验环境

本实验运行环境为Windows10，开发环境为Python3.7.1，调试环境为Spyder

2.4 适合人群

本实验适合所有学习Python的新手及老手，以及初次接触Tensorflow模块和Lstm神经网络的学习者们，从中可以了解掌握他们的基本概念，以便在以后的学习中熟练运用。

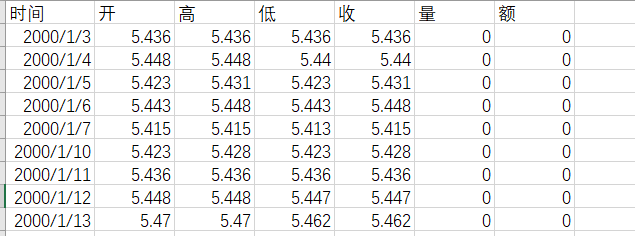
2.5 代码获取

代码将在以下环节进行展示和讲解。具体代码见<https://github.com/HardenMVB/12.28>。

1. 开发准备

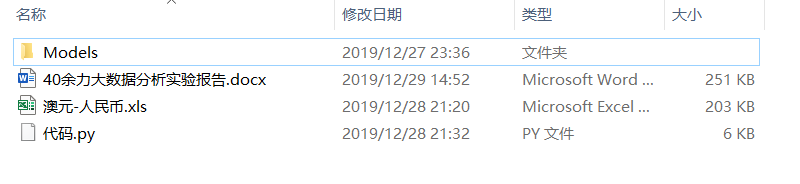
在上面的环节已经提到，要开发首先要安装anaconda，因为里面有我们所需要的各式各样的python库大全，且不需要配置环境变量，打开即可使用。安装完之后由于tensorflow库是没有的，所以要自行下载，我的版本是1.15.0，这样某些功能才可以兼容使用。

基本准备完成后，还需要导入数据，我选的题目是澳元对人民币的汇率分析预测，所以要导入此文件：



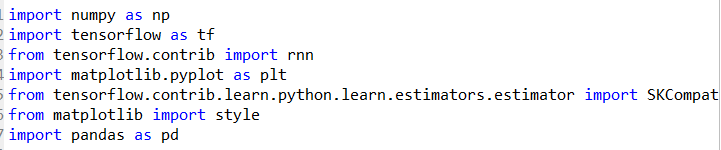
这份文件显示的是从2000年到2018年间的汇率变化，时间跨度大，汇率变化大，具有良好的分析价值和预测价值，所以我选用了此题目。

1. 项目文件结构



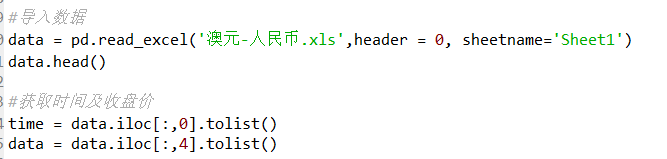
1. 实验内容及步骤

5.1 导入实验所需的Python库

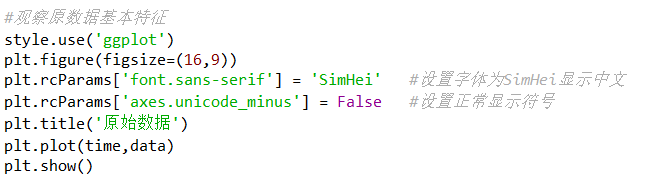


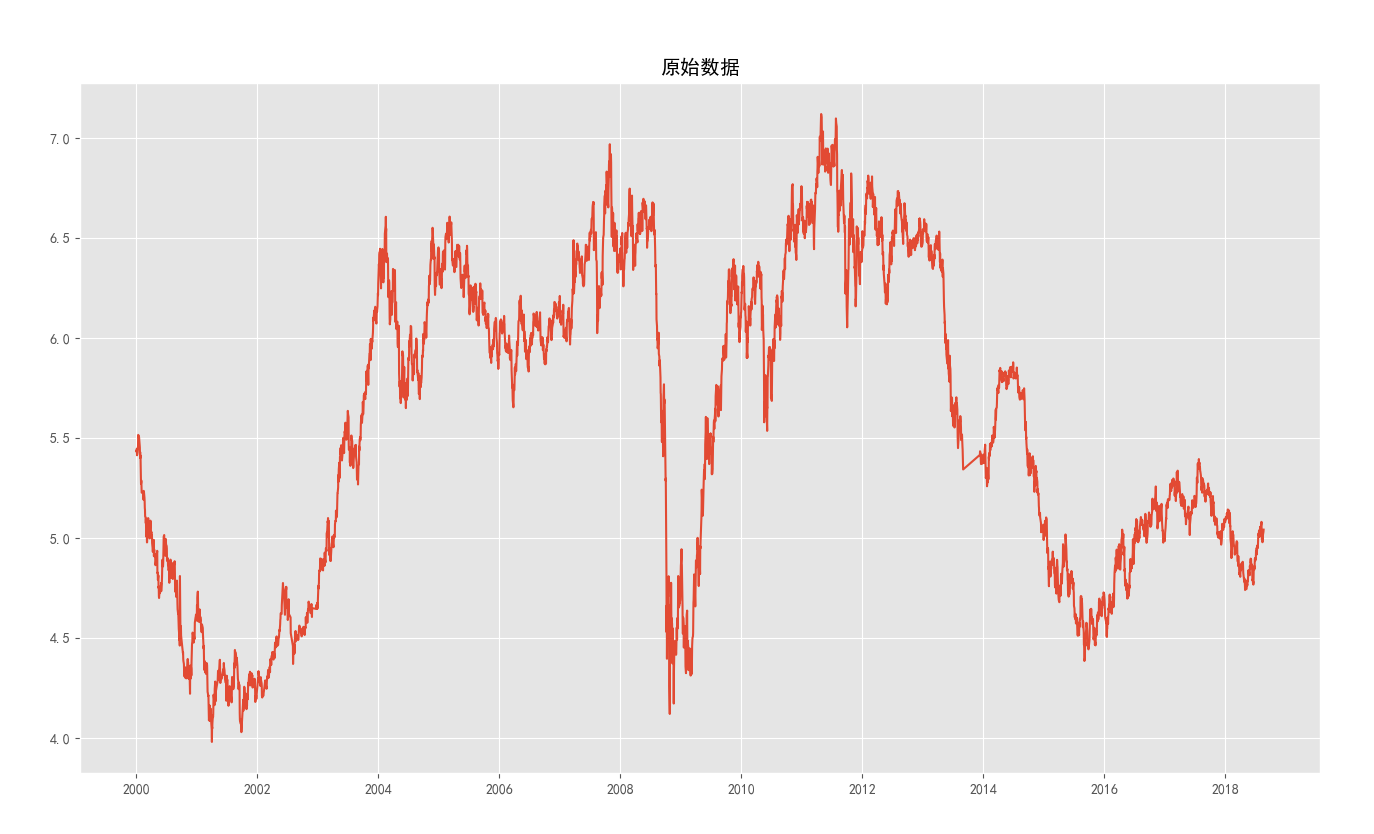
5.2 导入数据及处理

获取数据及收盘价

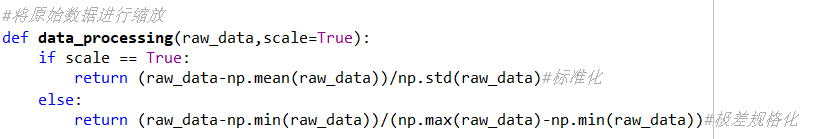


画出图像，观察曲线特征





从图中可以看出，汇率趋势大致呈显“M”字形，先上升，中间2008-2009年间猛然下跌，而后又再次上升，近五年又再次下降。总体汇率范围在4-7之间波动，而根据Lstm网络的特点，连续型数据需要用tanh输出，而由RNN被tanh激活后输出的值位于[-1，1]之间，因此要对数据进行缩放，即将数据标准化处理。



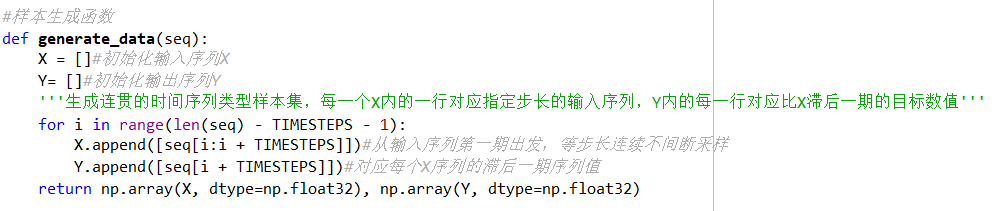
5.3 设置参数

接下来就是对神经网络进行参数定义。由于数据集较简单，所以隐藏层可以设置一层，神经元设32个，一个时间步中的递归数为12，即每12步为一组，训练轮数为2000，批处理数量为128。

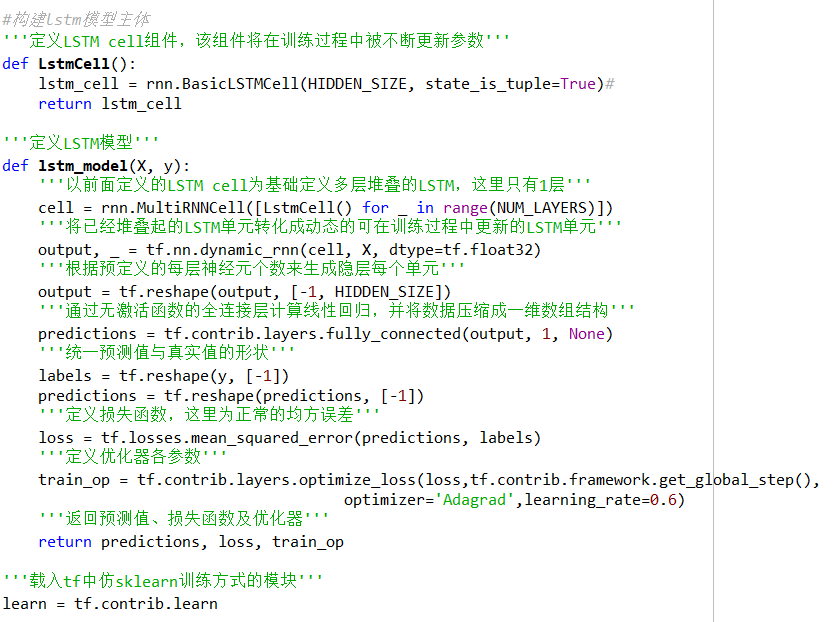


5.4 样本生成函数

为了将原始的时序数据处理成可供Lstm所接受的类型，需将原始数据做进一步处理，即从第一个开始，每12个序列作为x输入，第13个作为y标签。

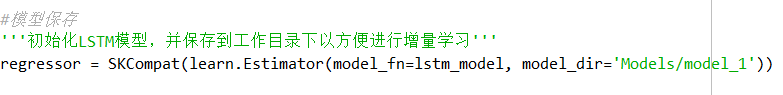


5.5 构建Lstm模型



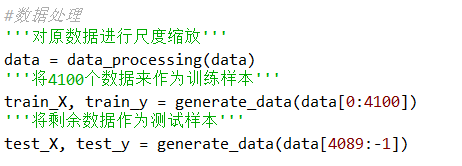
1. 实验数据记录和处理

6.1 模型保存



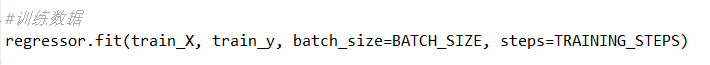
6.2 数据处理

调用前面的缩放函数处理数据，因为数据有4619个，我们选用4100个作为训练样本，这样剩下的519个数据就都能被预测出来，而鉴于之前选择的递归步数为12，即每12步为一组x，第13步为y，所以这4100个数据处理后将得到4089个，那么我们要将第4089个到最后一个数据转化为测试样本。

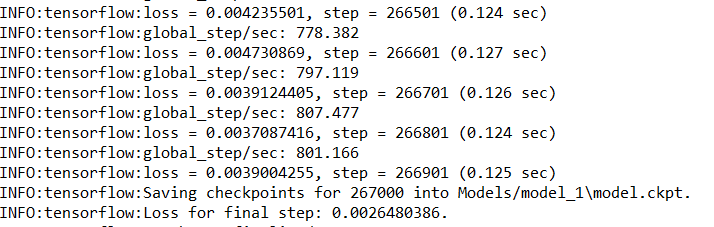


6.3 训练数据

这里运用到了之前的批处理数和训练轮数。



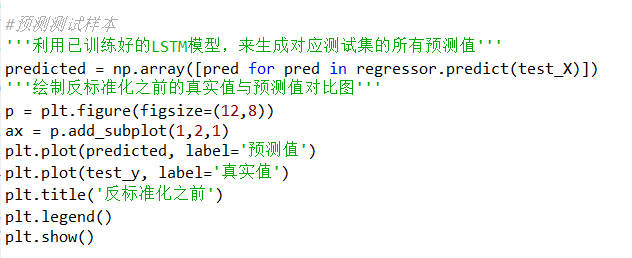
部分结果如下：



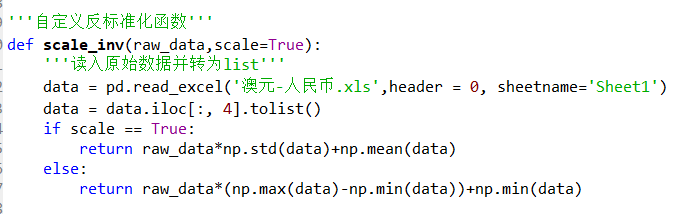
1. 实验结果与分析

7.1 预测测试样本

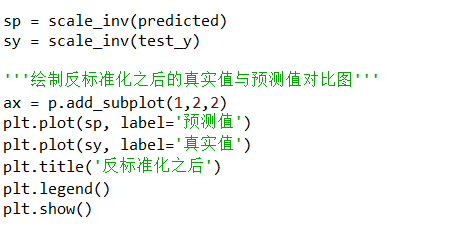
用训练好的模型计算处理过的测试集，可以得到预测值，此时得到的数据为标准化之后的。将预测值和真实值进行对比，得到反标准化之前的图像。

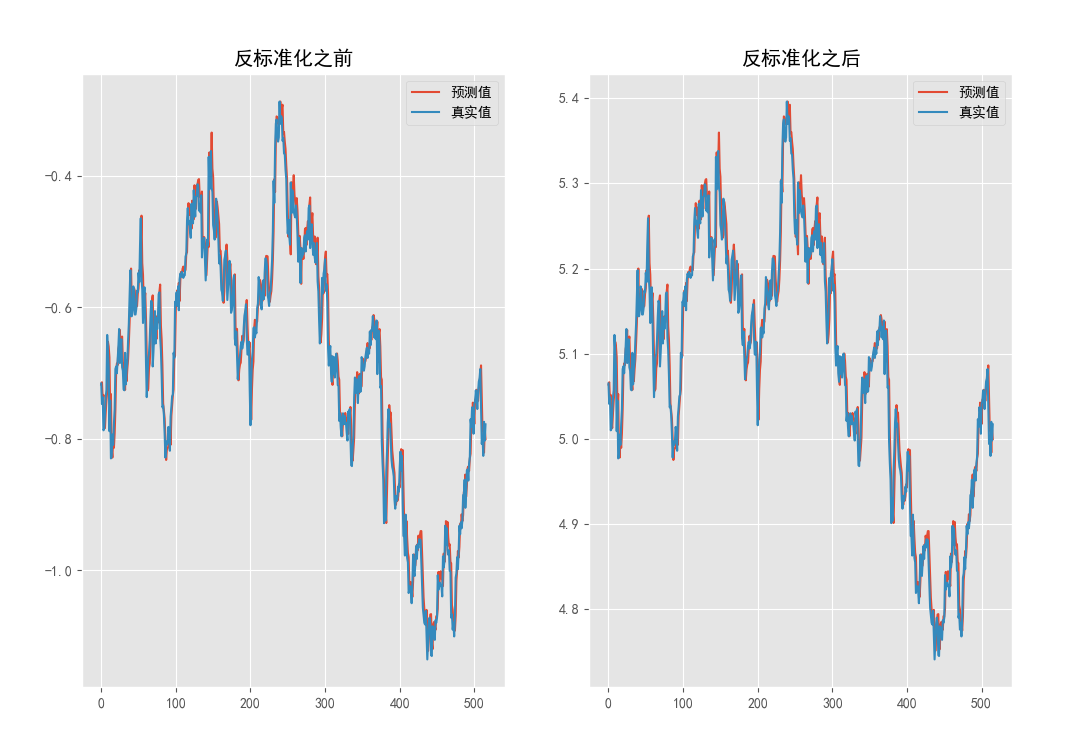


再定义一个反标准化函数，把预测出的数据转化为真实数据。

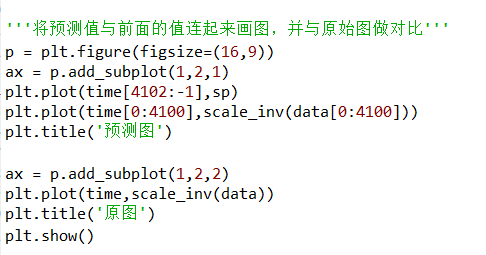


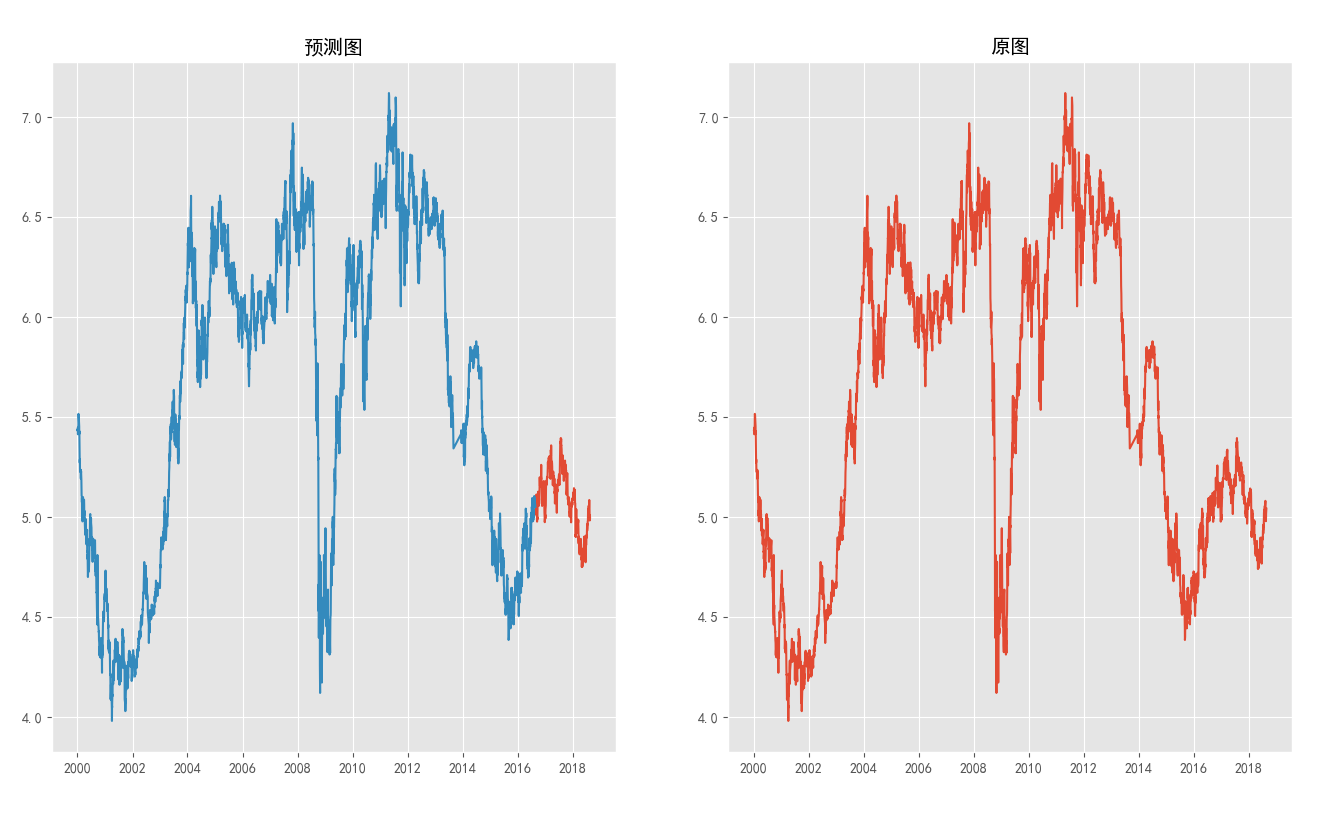
调用此函数，得到真实数据，并画出反标准化后的图像。





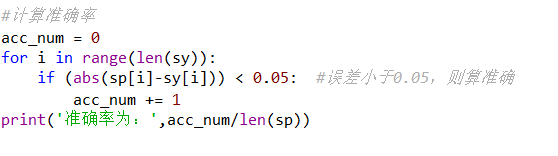
接下来要把所得到的预测值与之前的值连起来，画出图像，和原始图进行对比。





7.2 计算准确率

我这里定义的是当预测数据与原数据的误差小于0.05时，则算准确。因为差距过大时，容错率就很高，这样准确率就很高，体现不出这个模型预测的水平；而差距很小时，容错率就很小，所以会得出很小的准确率，甚至0.08这样的数字就会出现。所以经过考虑，我选择了0.05作为误差界限，既不大也不小，能充分体现模型预测的好坏程度。

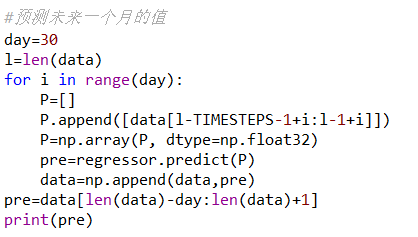


最后算出的结果为：

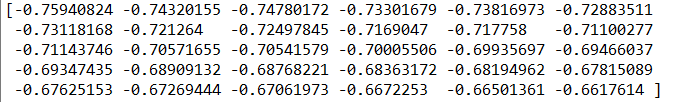


7.3 预测未来一个月的值

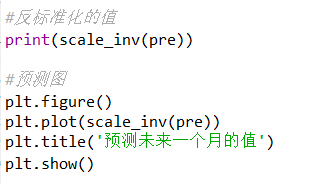
基于以上思路，我们就可以利用这个模型，通过之前定义的递归步数，进行未来汇率预测，具体方法是用已知12天的数据预测下一天的值，这个值再放入数据中继续预测下一天的值，得出结果。



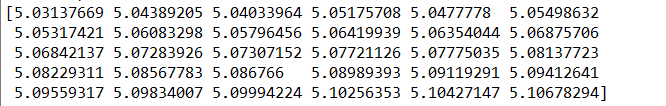
结果为：

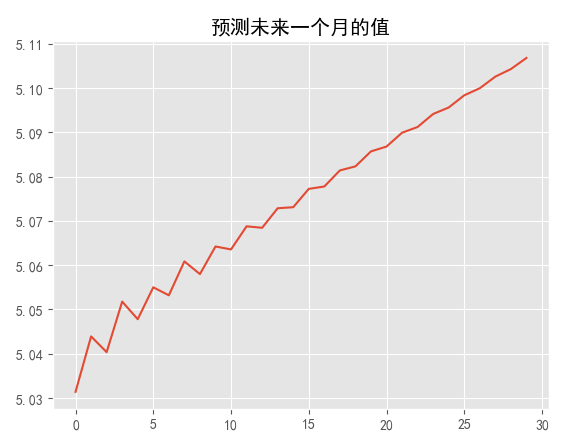


此为标准化的值，将它转化为原始数据，并画出预测图：



其结果和预测图分别是：





1. 问题与讨论

8.1 分析结果

首先从预测测试集的结果来看，大体上还是令人满意的，误差为0.002左右，算是非常精确了。

再看标准化图像，可以看出预测值和真实值的两条线非常接近，没有太大的偏差，预测效果还可以。在0.05的误差范围内，预测准确率也达到了94%，算是比较好的准确度了。当我拿其他数据进行测试时，最高的准确率也只有79%左右，说明用这个模型预测此数据集的效果很好，但预测其他的就需要改进了。

最后是预测未来一个月的值，从图上可以看出，在前15天，汇率在上下波动中上升；而在15天后，汇率就基本趋于平稳上升，没有太大波动。从30天的结果可以看出，该模型可以适用于预测短期汇率，当预测长期汇率时，效果可能就没那么好了，毕竟会受到很多别的因素的影响，这条平稳上升的直线也可能骤然猛降或猛升，到那时就得分析新的数据来进行改进了。

8.2 不足与改进

本实验的不足之处在于，预测的结果有很多是和原本的数据重合的，所以画出来的对比图基本重合，这就降低了预测效度，毕竟选用的都是已知的真实数据，达不到凭空预测的效果。还有一点在于预测的值会不准确，当测试样本为4000时，结果还和现在一样；但当测试样本为4300时，预测的结果就变成了逐步下降了，按理来说不会发生这样的情况，也许是因为4300的样本正好对应这图像的转折点，所以它的预测会有偏差，这也是模型的不足之处。预测的是以后的结果，应该和数据选取没有太大关系，并不会因为一次的下降而忽略了整体的上升，所以这是有待改进的地方。还有就是准确率偏低，有些模型可以预测出98%甚至99%的准确率，我用美元数据代入，准确率很高；而采用离岸人民币数据，准确率相当低。说明对于澳元汇率，此模型还算良好，准确率能达到94%，但是相比之下还有提升的空间。

针对本实验，有很多可以改进之处，例如用预测值取代输入值，增加实验可信度，还有就是影响汇率的因素有很多，如利息、加息、通货膨胀等等，可以把这些因素加进模型中，来观察预测结果的好坏。还可以不止用收盘价作为最后的指标，当天的汇率也可能受各种人为的影响，所以不能把模型做的太单一，要把这些都考虑进去，才能让结果更加真实可靠。