海上降水量预测

指导老师：于彦伟
主讲人：颜丙齐

问题分析

问题定义：

1、根据已有数据，预测海上降水量

2、要预测15-20天后的降水量

3、预测6月-9月的降水量

4、现有数据CIO数据和降水量数据

问题分析：

1、根据经验，降水量应该有很强的时序上的规律，所以我们利用lstm模型来捕获该规律，预测降水量。

2、降水量应该也有空间上的一些规律，既降水是呈现区域性的，正在考虑如何使用。

输入输出

现有数据：

1、5D的cio数据，该数据的组成为每天一个5D的数据，该数据和降水量有着高度的相关关系。

2、降水量数据，该数据为，每天每个空间点上各一个数据。该数据是有关降水量的数据。

3、数据取得是cio数据和降水量数据的交集区域，纬度范围[-20,20]，经度范围[40,120]，故得22X43的区域。

存在问题：

1、有很多点的降水量数据为零

2、降水量数据较小

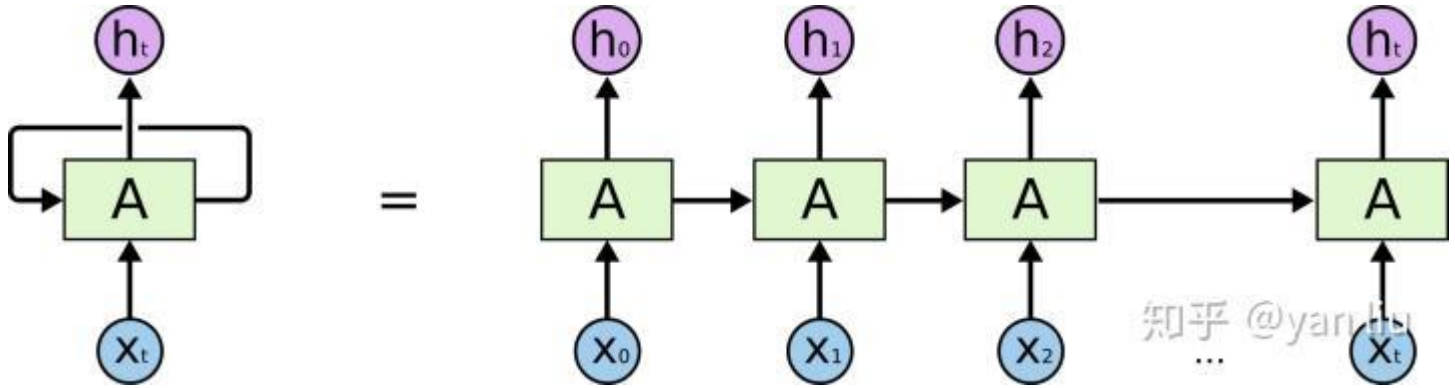
模型

1、根据经验，降水量有很强的时序关系，即，去年的这个月份多雨，今年同样的月份下雨的可能性会很大。

2、现有的数据为每天一个数据，即有很强的时序性数据，很方便我们的使用。

3、我们选用LSTM模型来捕获时序关系，预测降水量。

4、输入为cio数据，输出为降水量数据



评价指标

- 使用单一指标，皮尔森系数，其公式表达为：

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

- 皮尔森系数常用于度量两个变量X和Y之间的相关（线性相关），其值介于-1与1之间。
- 其值越接近与1说明X与Y正相关程度越高；其值越接近-1说明X与Y负相关程度越高；其值越接近0说明X与Y相关程度很低。

实验设置

- 实验说明:
- 在之前的工作中，做了很多实验，因为是一个探索的过程，所以有些实验的价值没有那么大，这里，我只讲述相对重要的一些实验。
- 在实验过程中，模型是一步一步完善的，所以，刚开始的模型简单，参数设置的也存在一定问题，我会探索性的对模型展开讲解。
- 对于实验的参数，在很多实验的基础上，确定了一个效果较好的参数范围，再在其中选择合适的参数，最终一步步确定所有的参数。
- 在实验的过程中，也发现了一些问题，将在后面部分，展开讨论。

实验进行时

第一部分实验:

运行环境：python3， pytorch

实验模型：双层的LSTM模型

模型输入：1Dcio数据

模型输出：降水量数据

思想：用降水量数据去匹配 各个点进而预测出所有点的降水量

实验简述:

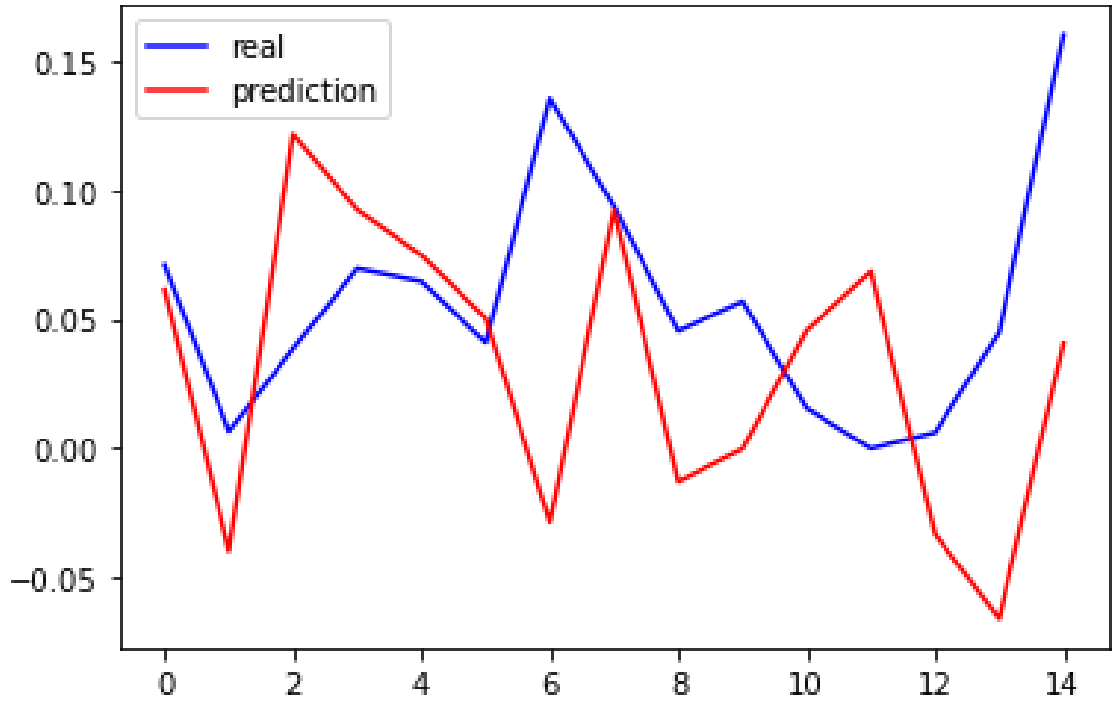
1、使用模型训练850次的一个效果图

2、隐藏层使用不同数量的神经元的效果对比

3、输入数据使用不同的batch和不同的训练次数对比

实验一

- LSTM网络配置：两层LSTM叠加，隐藏层15个神经元，训练850次。
- LSTM(input_size=15, hidden_size=15, num_layers=2)
- input(seq_len, batch=1, input_size=15)
- 皮尔森系数：
- 0.10604126137238266



实验二

- 随机取了10个连续点进行预测。
- LSTM网络配置：两层LSTM叠加，训练850次。
- LSTM(input_size=15, hidden_size=*, num_layers=2)
- 预测15天后的降水量。
- 这次用了40%的数据以提高算法的运行效率
- hidden_size分别取15、25、30、60、365.对比其不同的值对结果的影响。
- 实验发现hidden_size取30时，效果较好。

实验二

点\数据	隐藏层15个神经元	25个神经元	30个神经元	60个神经元	365个神经元
S1	0.314405598	-0.04051	-0.12698162	0.291851	0.022357664436
S2	-0.008880479	-0.00961	0.363939418	0.526753	0.253566361100
S3	0.285273037	0.41503	0.513305358	-0.05184	0.153702819612
S4	0.276011928	0.192823	-0.090521737	-0.03795	
S5	0.21797024	0.073908	-0.36217402	-0.18668	
S6	0.018863612	0.042437	0.194706082	-0.18261	
S7	-0.188002019	0.182807	-0.2073635	0.019703	
S8	0.344841686	-0.21563	0.171127667	0.113001	
S9	0.083092507	0.174363	0.42483937	0.212219	
S10	0.37221211	0.014209	-0.243015977	-0.4155	

实验三

- 利用LSTM网络，用第一模态的CIO数据作为输入，用降水量数据作为输出，进行训练。
- 随机取了10个连续点进行预测。
- LSTM网络配置：两层LSTM叠加，隐藏层个神经元，训练850次。
  LSTM(input_size=1, hidden_size=120, num_layers=2)
- input(seq_len, batch=*, input_size=1)
- 预测15天后的降水量。
- 使用用了40%的数据以提高算法的运行效率

- batch分别取2倍的hidden_size和4倍的hidden_size;train_time分别取750和1750。对比不同值对实验结果的影响。
- 最终发现batch取4倍的hidden_size且train_time取750效果较好。

实验三

batch=2/train_time=750	batch=4/train_time=750	batch=4/train_time=1750
0.14232	0.201466	0.141595
0.221423	0.243427	0.196129
0.520444	0.539235	-0.08953
0.214156	-0.23808	-0.39813
-0.43049	-0.33	-0.33841
-0.22521	-0.51565	-0.42828
0.074463	-0.32501	-0.29419
-0.67843	-0.67056	-0.71885
-0.46857	-0.17331	-0.12893
-0.00956	-0.02108	-0.02996

实验进行时

第二部分实验:
运行环境：python3， pytorch
实验模型：双层的LSTM模型
模型输入：5Dcio数据
模型输出：降水量数据
思想：用降水量数据去匹配 各个点进而预测出所有点的降水量
区别：这里与第一部分实验的主要区别是，引入了5D的cio数据
实验详情:
4、使用1D数据和5D数据效果对比
5、使用5D数据探究合适的隐藏层神经元数量

实验四

- 随机取了10个连续点进行预测。
- LSTM网络配置：两层LSTM叠加，隐藏层个神经元，训练850次。
  LSTM(input_size=1, hidden_size=30, num_layers=2)
- input(seq_len, batch=4*hidden_size, input_size=1)
- 预测15天后的降水量。
- 使用用了40%的数据以提高算法的运行效率

- 输入1Dcio数据和5Dcio数据，对比不同的结果

实验四

点\数据	1Dcio	5Dcio
S1	-0.12698162	0.291851
S2	0.363939418	0.526753
S3	0.513305358	-0.05184
S4	-0.090521737	-0.03795
S5	-0.36217402	-0.18668
S6	0.194706082	-0.18261
S7	-0.2073635	0.019703
S8	0.171127667	0.113001
S9	0.42483937	0.212219
S10	-0.243015977	-0.4155

实验五

- 随机取了10个连续点进行预测。
- LSTM网络配置：两层LSTM叠加，隐藏层个神经元，训练850次。
  LSTM(input_size=1, hidden_size=30, num_layers=2)
- input(seq_len, batch=4*hidden_size, input_size=1)
- 预测15天后的降水量。
- 使用用了40%的数据以提高算法的运行效率

- 对比不同的hidden_size对实验结果的影响。

实验五

点\数据	30个神经元	60个神经元	90个神经元
S1	-0.12698162	0.291851	
S2	0.363939418	0.526753	
S3	0.513305358	-0.05184	
S4	-0.090521737	-0.03795	
S5	-0.36217402	-0.18668	
S6	0.194706082	-0.18261	
S7	-0.2073635	0.019703	
S8	0.171127667	0.113001	
S9	0.42483937	0.212219	
S10	-0.243015977	-0.4155	

实验进行时

第三部分实验:

- 运行环境： python3 ， pytorch
- 实验模型： 双层的LSTM模型+MLP
- 模型输入： 5Dcio数据
- 模型输出： 降水量数据
- 思想： 用降水量数据去匹配 各个点进而预测出所有点的降水量
- 区别： 在第二部分实验的基础上更新了实验模型， 在LSTM模型的基础上，增加了MLP

实验详情:

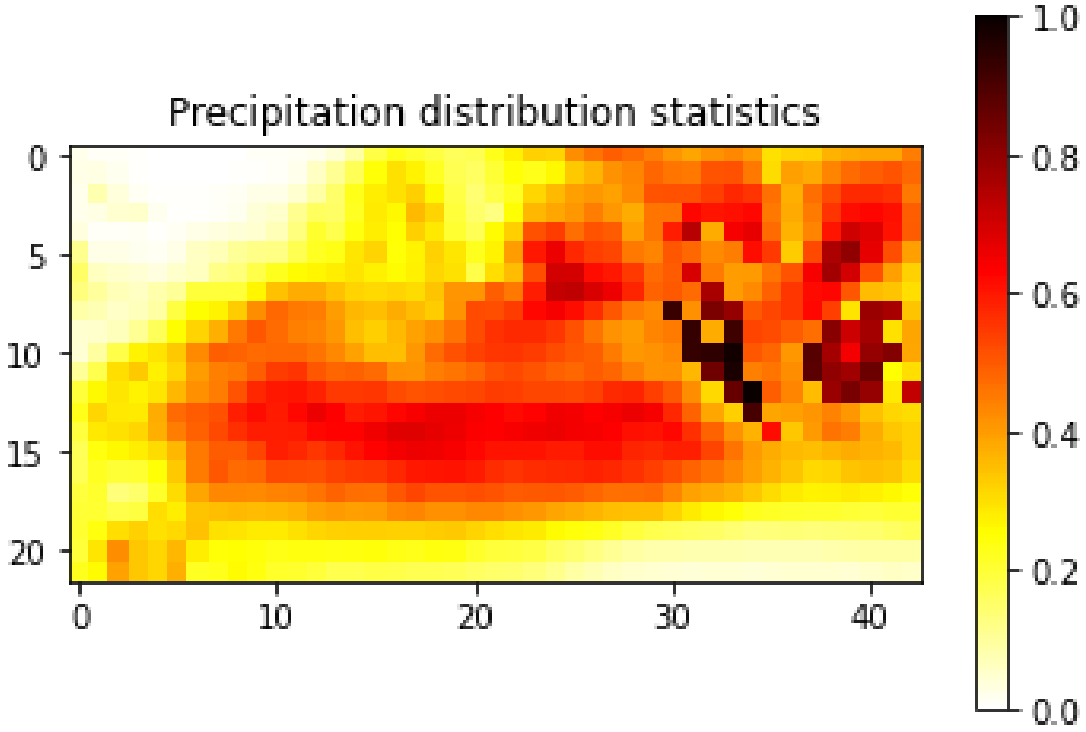
6、比较增加MLP对实验的影响。

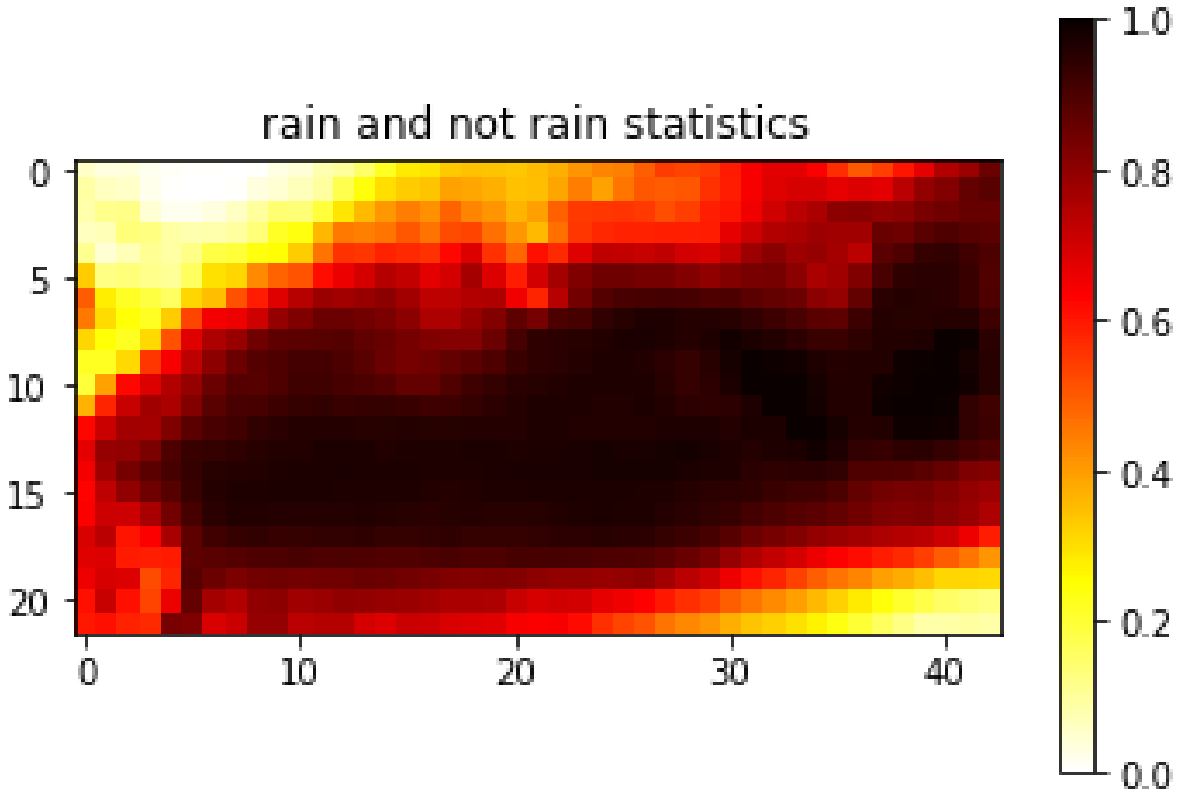
实验六

随机挑选的数据	减少了MLP的层数	去掉MLP	使用F.ReLU	使用nn.ReLU	使用两层MLP 加一个ReLU
-0.031017232	-0.043647039	0.059231474	0	0.01806153	-0.091418138
0.561641137	0.376482169	0.547938262	0	0	0.526045486
0.060910631	0.047256675	0.152843556	0.12020982	0.070583539	0.061452651
-0.28099276	-0.243177304	-0.30932793	0	-0.167639988	-0.186375574
0.462993227	0.481162378	-0.570021547	-0.0719338	0.087909451	0.44313683
-0.132382501	0.034744109	0.100922031	0	0	0.101180658
-0.444772807	-0.063819075	-0.514586078	-0.48854349	0	0.182614332
0.173859569	0.178517801	-0.056168684	0.027932172	-0.221841602	-0.277144312
-0.639868569	-0.034987819	-0.299233029	-0.00588903	0.164430535	0.072940234
-0.125619953	-0.146169406	0.52809173	0	0	-0.099296477
-0.241968089	-0.222368442	-0.234972077	-0.11194214	-0.185439562	-0.241486094
-0.753481837	-0.762179558	-0.585303753	0	-0.564301183	-0.700849934
0.055920507	-0.054607482	0.179833541	0	0.014338427	-0.164891459
0	0	0	0	0	0
-0.391979714	-0.388675889	-0.212797221	0.160200798	0	0.072405119
-0.135655822	-0.156773689	-0.118485487	-0.13271361	0.127996738	-0.002245619
0.536030116	0.490159079	0.470347874	0.431310904	-0.084327797	0.446780429
0	0	0	0	0	0
-0.42850496	-0.435782411	0.063007989	-0.33986047	0	0.600146427
-0.264852954	-0.37391918	0.152340728	-0.24651414	-0.422109386	-0.28049425

附加

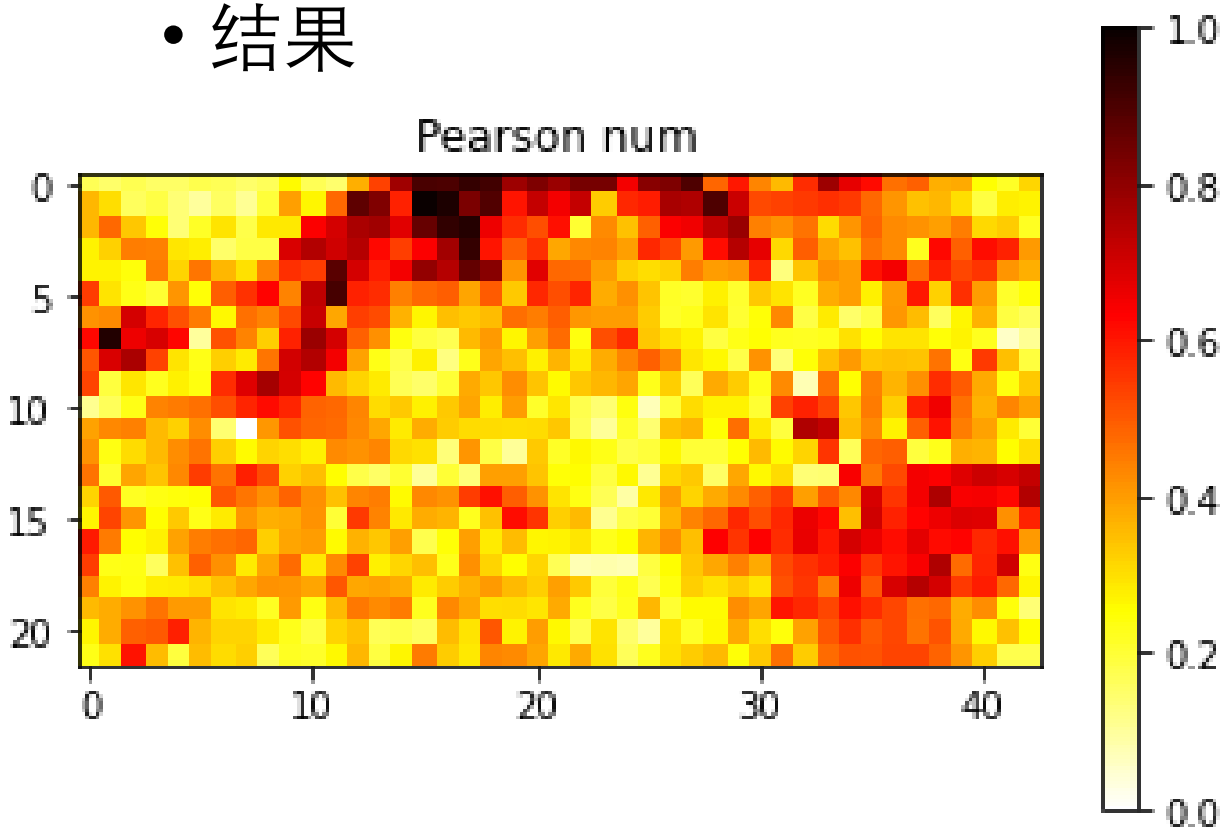
- 数据处理

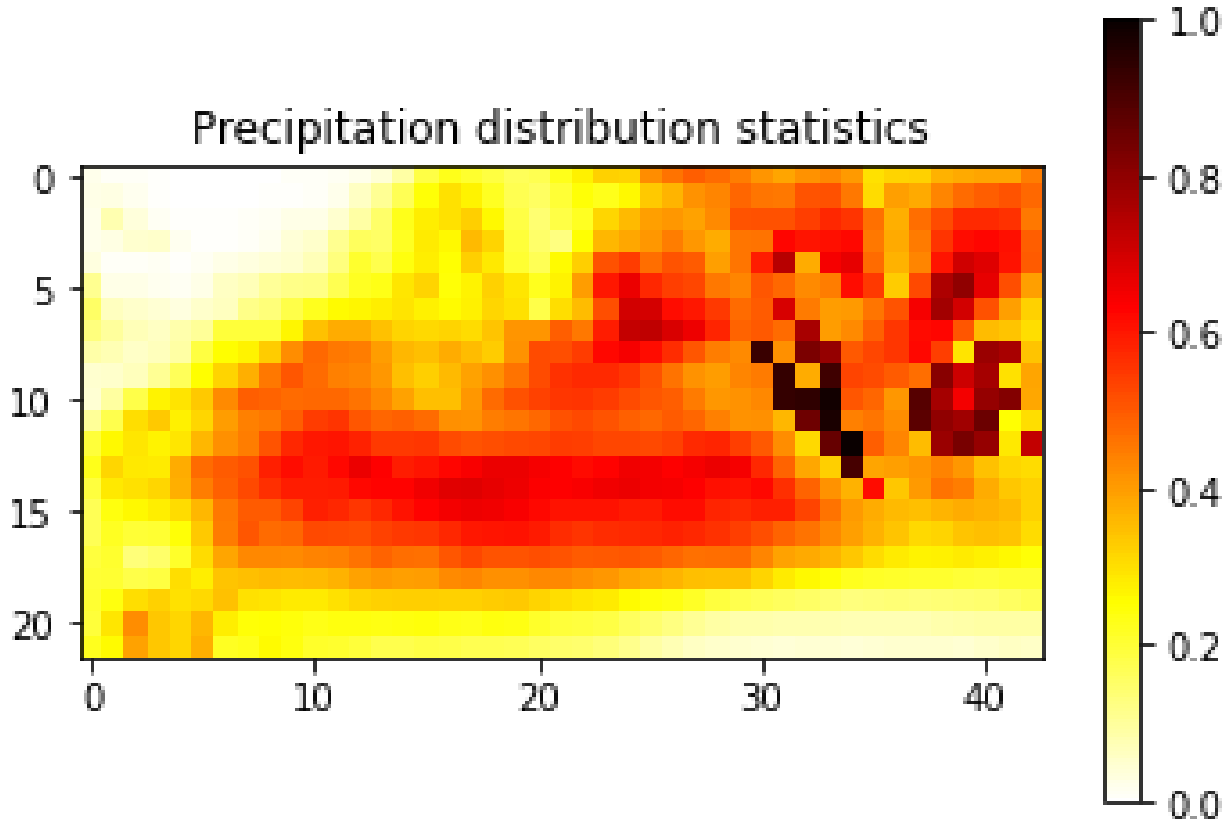

Precipitation distribution statistics
0
5
10
15
20
0
10
20
30
40
1.0
0.8
0.6
0.4
0.2
0.0


rain and not rain statistics
0
5
10
15
20
0
10
20
30
40
1.0
0.8
0.6
0.4
0.2
0.0

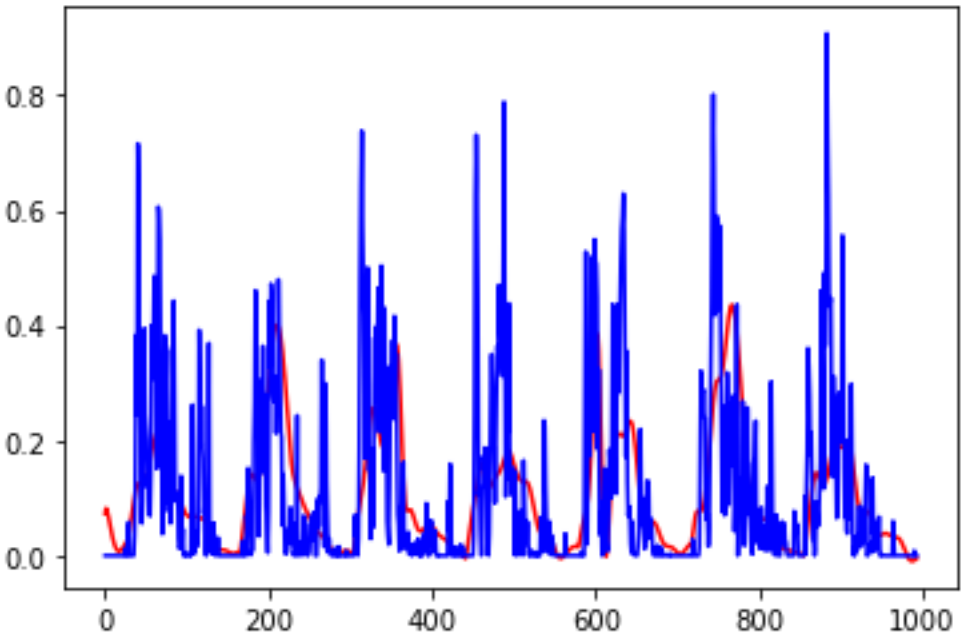
附加

- 结果

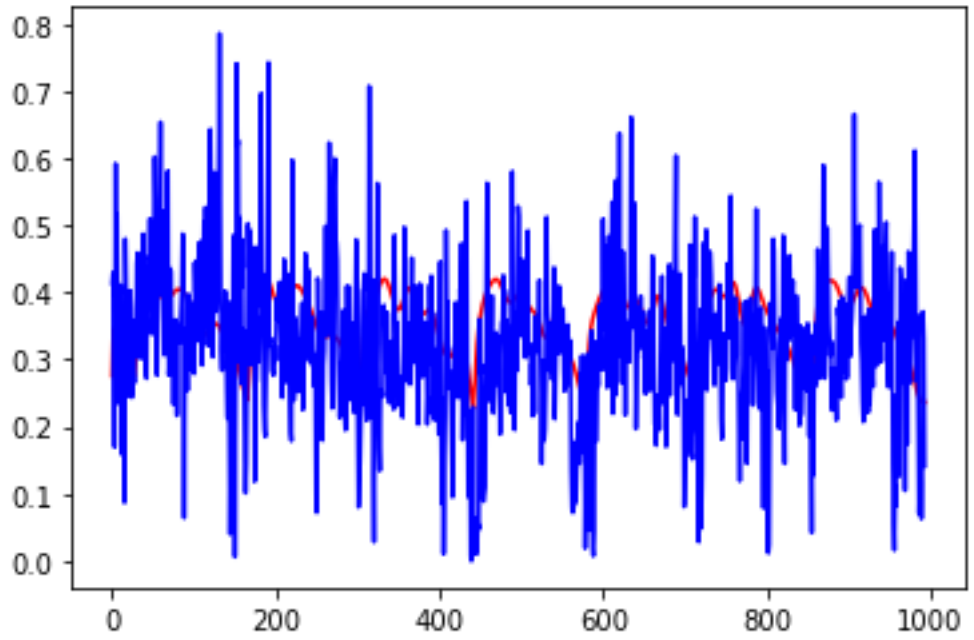

Pearson num
0
5
10
15
20
0
10
20
30
40
1.0
0.8
0.6
0.4
0.2
0.0


Precipitation distribution statistics
0
5
10
15
20
0
10
20
30
40
1.0
0.8
0.6
0.4
0.2
0.0

结果分析

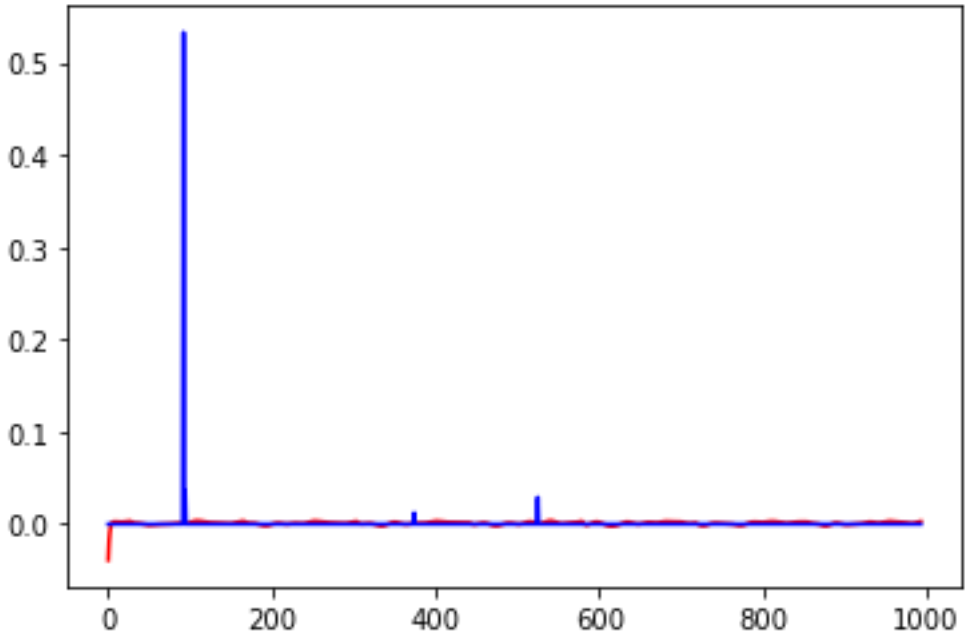


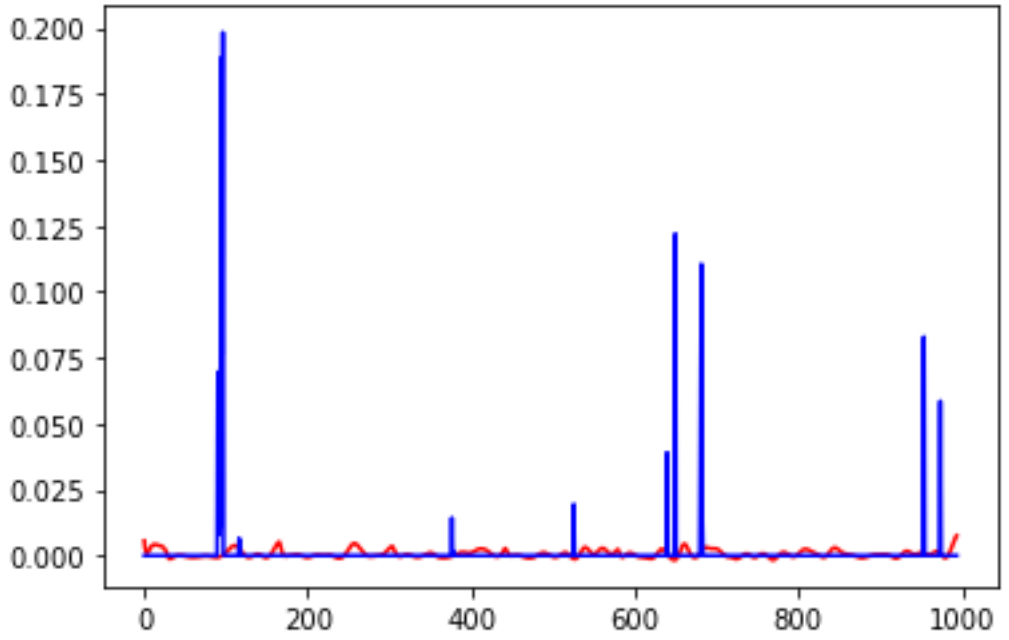
0.5403956613047615



0.2358102224603758

结果分析



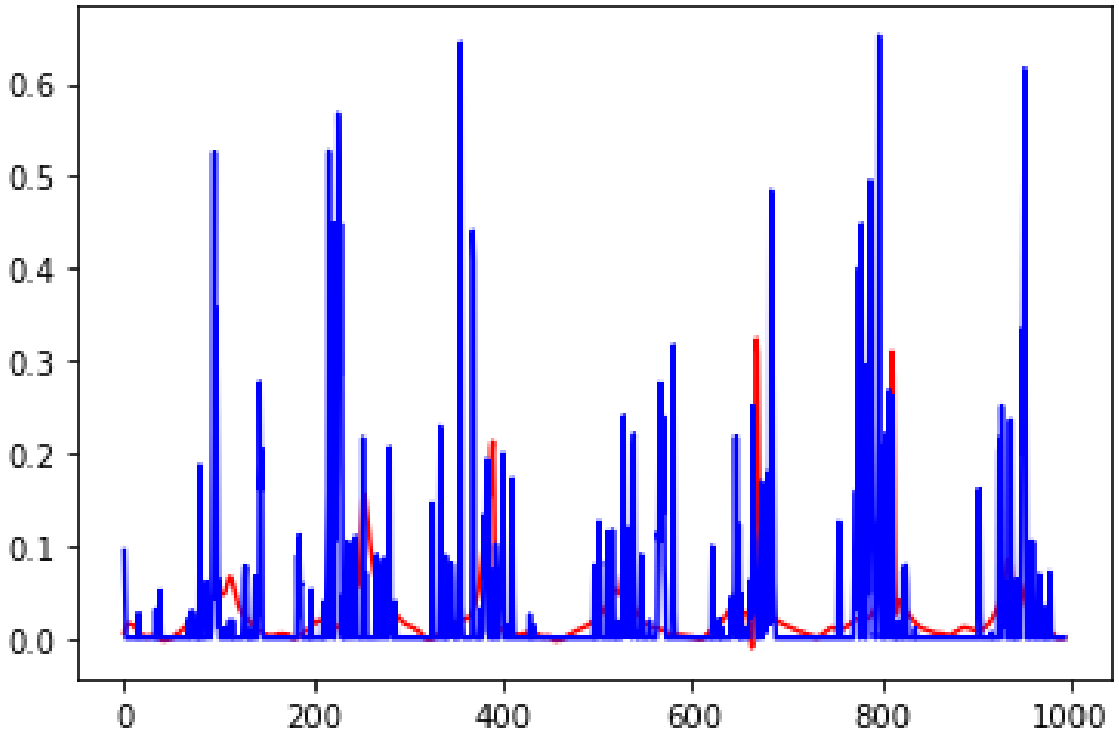


0.004483495862727628

-0.03128930978803778

预测的结果和数据存在一定的关系，有的数据有利于预测，有的数据，会影响预测

存在疑问



0.13384086399242973

看图中，基本将数据的趋势预测的差不多了，该高的高，该低的低，算出来皮尔森系数却很小，那么在这个问题中，单纯的用皮尔森系数衡量真的有效吗？

存在疑问

- 对于每天一个的cio数据，是否能和所有空间点上的降水量都相关，或者相关度很高。
- 有些点是不可预测的。
- 降水量数据过小，我认为会影响预测。
- 对于数据，和模型，请问老师们有什么好的建议吗？

Thank you