2019

基于膜蛋白序列的药物作用位点发现

信息科学与技术学院
翟宇豪
2019.3.10

提　纲

OUTLINE

1 背景、目标、课题

2 数据

3 处理过程

4 现阶段结果

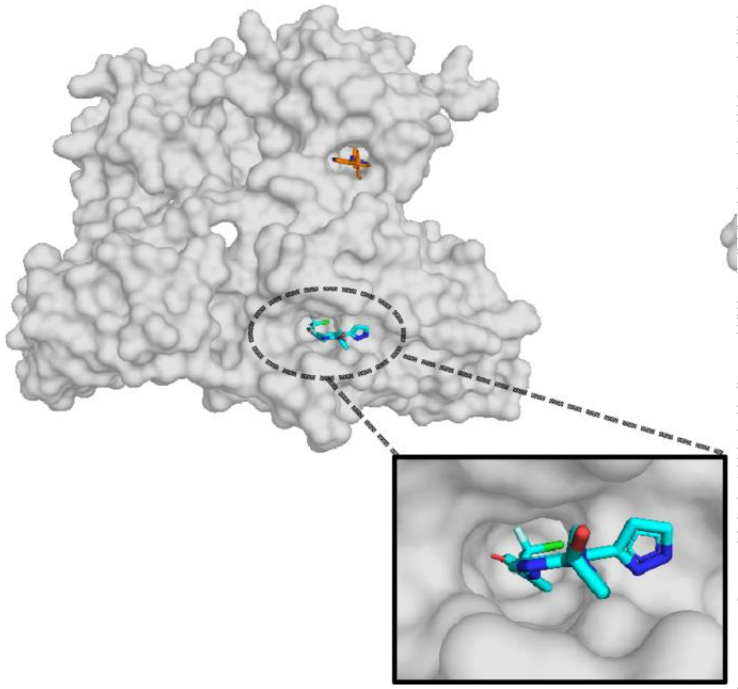
1.1 选题背景

医疗制药业 1

人类对健康的需求不断提高，医疗制药业迅速发展

科研发现 3

科学家发现，大部分药物通过作用在蛋白质上发挥其效力



2 分子生物学

随着科学的进步，人类对生物大分子的了解及认知也不断加深

4 靶向药物

药物能瞄准特定的病变部位，并在目标部位蓄积或释放有效成分。

制约因素：资金+时间+审批

资金消耗巨大，研发周期长，更重要的是，很多都不能通过药物管理局的审批

计算生物学：使用计算的方法预测药物作用位点

在计算机的帮助下，通过计算的方法，发现其作用位点，会是一种省时省力的方案

该领域主要有两种研究方向：

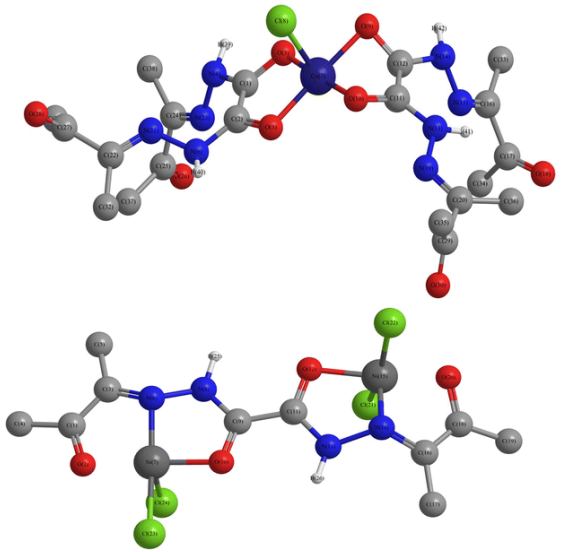
药物作用位点

配体为中心

相似的配体，将会共享相似的作用位点。

蛋白质为中心

基于序列和基于结构两种方法。

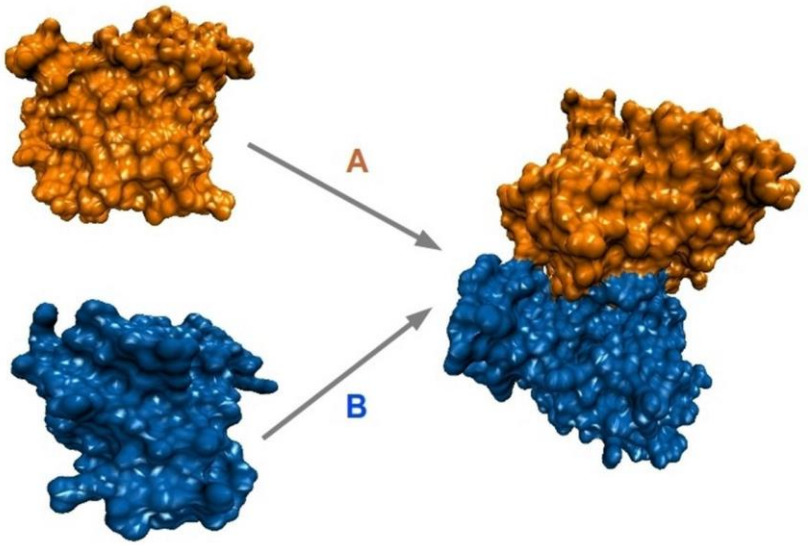


缺点与不足

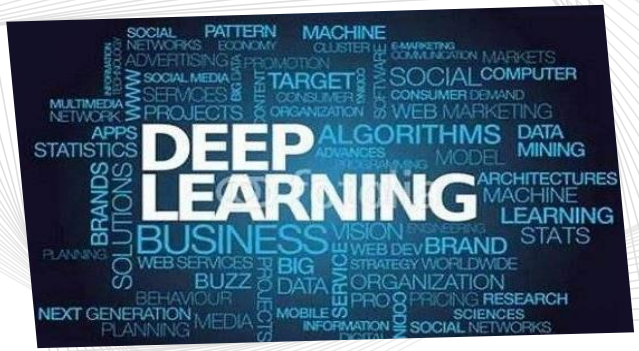
1，取决于所选配体的性质

2，已知结构的蛋白不多

3，精确度有待提升，还有提升空间



1.3 研究目标



蛋白质序列信息

采用深度学习的方法
对数据进行挖掘

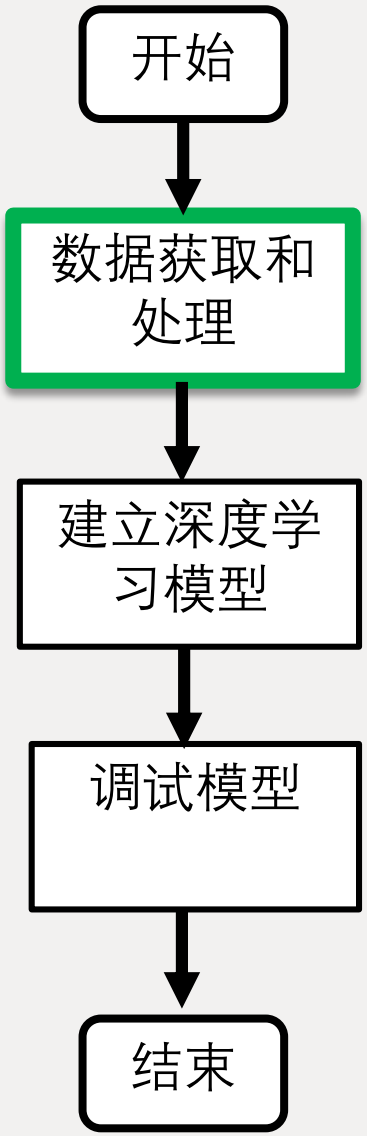
对药物在蛋白质上的
作用位点进行预测

蛋白质

深度学习

药物作用位点

2 数据收集



DRUGBANK

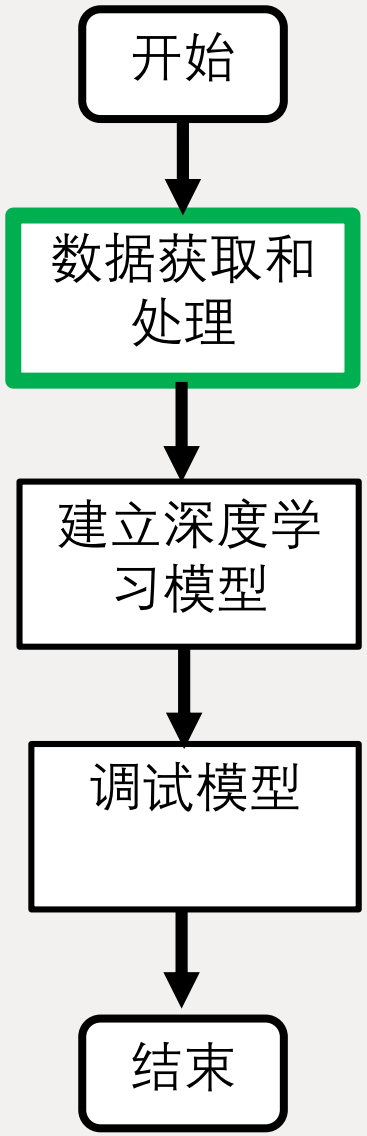
提取了药物与蛋白质作用残基名称列表。
“HET”字段信息。

数据收集

PROTEIN DATA BANK

筛选得到包含药物作用位点的蛋白质
及其序列信息和作用位点

3.1 数据清洗



根据需求对数据进行整理及清洗：

数据清洗

❶ 标记作用位点

```
>2PX8_B
GRAGGRTLGEQWKEXLNAMGKEEFFSYRKEA
GRTLGEQWKE_LNAMGKEEFFSYRKEAILEV
0000000000001000000000000000000
```

❷ 比对标准序列

```
>2PX8_B
GRAGGRTLGEQWKEXLNAMGKEEFFSYRKEAILEVDRTEARRAR
GRTLGEQWKE_LNAMGKEEFFSYRKEAILEVDRTEARRARREGN
00000000000010000000000000000000000000000000
>2PX8_B
GRAGGRTLGEQWKEXLNAMGKEEFFSYRKEAILEVDRTEARRAR
    GRTLGEQWKEXLNAMGKEEFFSYRKEAILEVDRTEARRAR
    00000000000010000000000000000000000000000
```

❸ 最终结果

```
>2PX8_B
GRTLGEQWKEXLNAMGKEEFFSYRKEA
00000000000010000000000000000
```

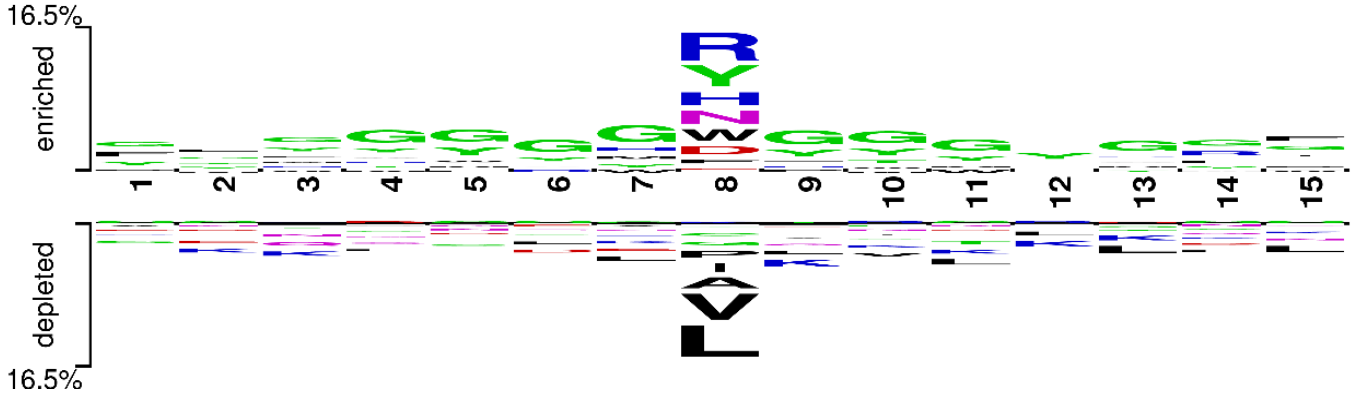
3.2 数据验证

CD-HIT	蛋白条数	正样本	负样本	正负比
0.3	446	2729	23930	1：47
0.9	653	4235	199300	1：48
全部	3232	22141	1063764	1：49

使用CD-HIT中cut-off参数为0.3，0.9对序列进行去冗余

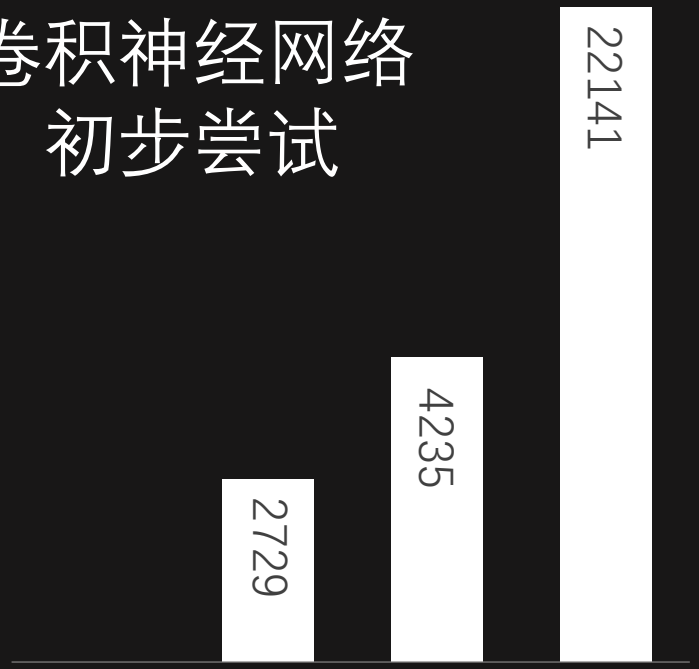
使用**two samples logos工具**对位点特异性进行分析：

本图来自全部正样本及0.3cut-off后的负样本

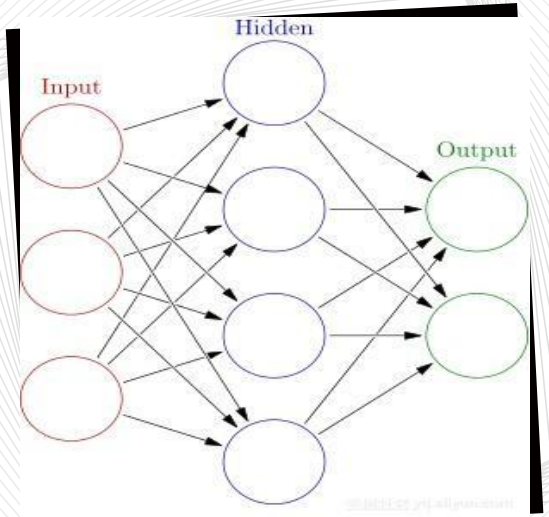


4 现阶段结果

卷积神经网络初步尝试



CD-HIT	0.3	0.9	全部
val_loss	0.68	0.67	0.64
Val_acc	0.54	0.58	0.63



数据采用one-of–key编码方式。训练集80%，验证集20% 。

标志为1的位点为正样本。窗口大小为17时，达到最优效果

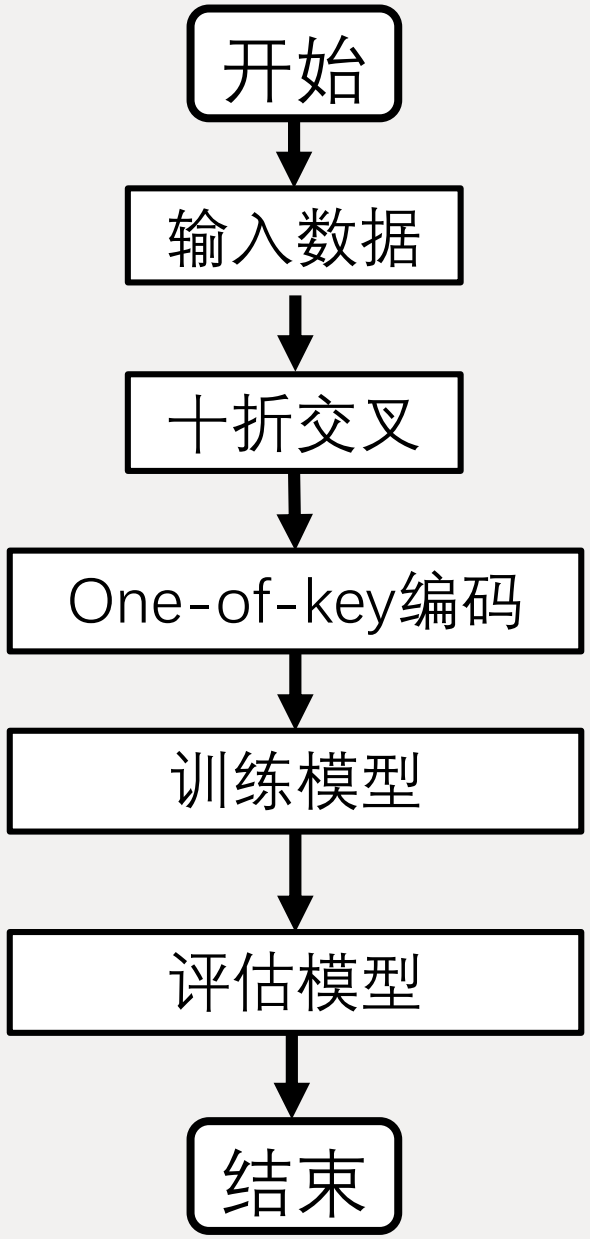
1层卷积网络
Dropout:0.5
batch_size:120
Epoch:500

4 最近一次较好实验的设置：

实验运行时间：1天及以上。
实验使用数据信息：消除位于边界的负样本，包含正样本位点的负样本。
数据集正负样本设置，比例设置，
正负样本1：33。采用十则交叉验证模式构建10个数据集的数据，依次进行学习。目前没完成整体评估+存取模型+选取最优模型等步骤，只对十折中第一部数据分进行学习。

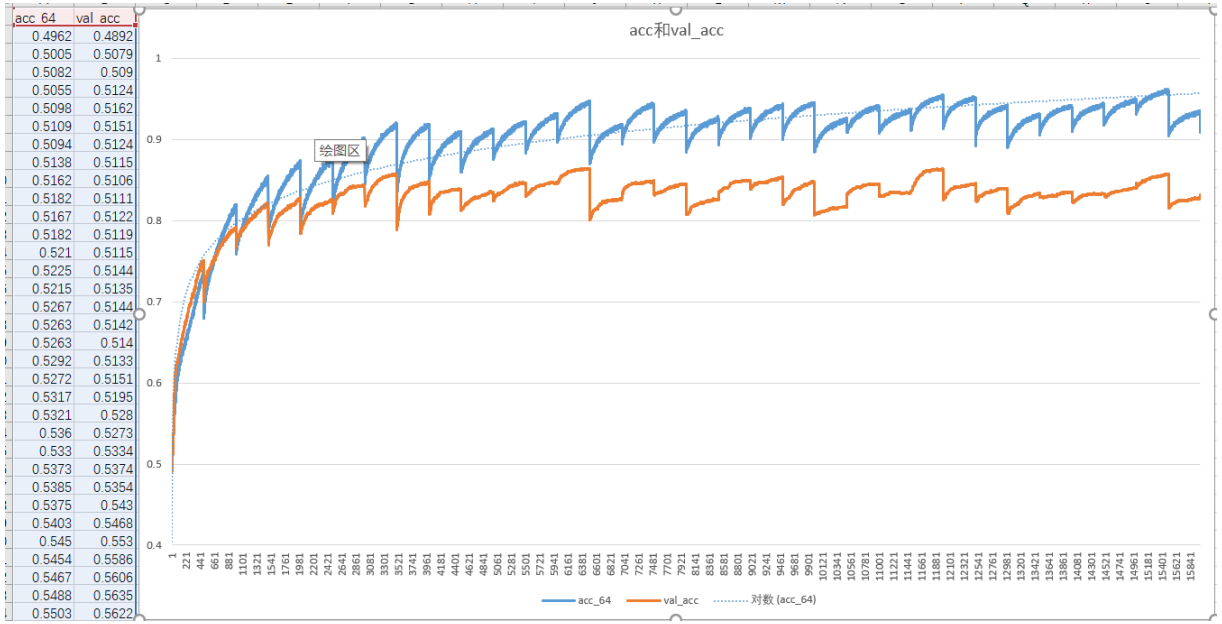
一些关键参数设置及网络：
（nb_epoch=500,batch_size = 512）
filter1 = 16-32-48-64
filtersize1 = 2
dropout1 = 0.25
L1CNN = 0
nb_classes = 2
batch_size = batch_size
actfun = "relu";
nadam = Nadam(lr=0.00001)
optimization = nadam
dense_size1 = 128 dense_size2 = 64 dense_size3 = 8 dropout_dense1 = 0.298224 dropout_dense2 = 0 dropout_dense3 = 0
input = Input(shape=(input_row, input_col))
x = conv.Conv1D(filter1, filtersize1, init='glorot_normal', W_regularizer=regularizers.l2(L1CNN),border_mode="same")(input)
x = Dropout(dropout1)(x) x = Activation(actfun)(x)
x = core.Flatten()(x) output = x
output = Dropout(dropout1)(output)
output = Dense(dense_size1, init='glorot_normal', activation='relu')(output)
output = Dropout(dropout_dense1)(output)
output = Dense(dense_size3, activation="relu", init='glorot_normal')(output)
output = Dropout(dropout_dense3)(output)
out = Dense(nb_classes, init='glorot_normal', activation='softmax')(output)
cnn = Model(input, out)
cnn.compile(loss='binary_crossentropy', optimizer=optimization, metrics=['accuracy'])

4 流程及结果



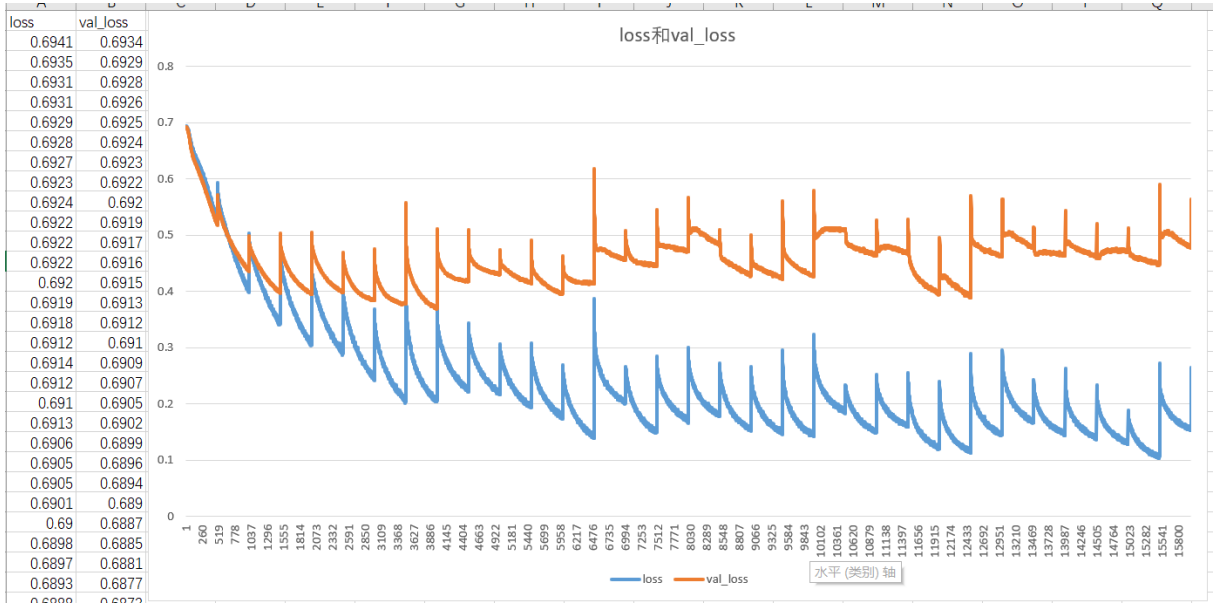
1

acc_64	val_acc
0.4962	0.4892
0.5005	0.5079
0.5082	0.509
0.5055	0.5124
0.5098	0.5162
0.5109	0.5151
0.5094	0.5124
0.5138	0.5115
0.5162	0.5106
0.5182	0.5111
0.5167	0.5122
0.5182	0.5119
0.521	0.5115
0.5225	0.5144
0.5215	0.5135
0.5267	0.5144
0.5263	0.5142
0.5263	0.514
0.5292	0.5133
0.5272	0.5151
0.5317	0.5195
0.5321	0.528
0.536	0.5273
0.533	0.5334
0.5373	0.5374
0.5385	0.5354
0.5375	0.543
0.5403	0.5468
0.545	0.553
0.5454	0.5586
0.5467	0.5606
0.5488	0.5635
0.5503	0.5622



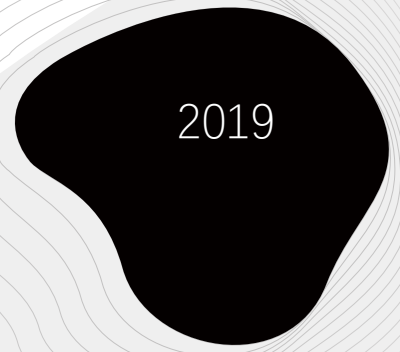
2

loss	val_loss
0.6941	0.6934
0.6935	0.6929
0.6931	0.6928
0.6931	0.6926
0.6929	0.6925
0.6928	0.6924
0.6927	0.6923
0.6923	0.6922
0.6924	0.692
0.6922	0.6919
0.6922	0.6917
0.6922	0.6916
0.692	0.6915
0.6919	0.6913
0.6918	0.6912
0.6912	0.691
0.6914	0.6909
0.6912	0.6907
0.691	0.6905
0.6913	0.6902
0.6906	0.6899
0.6905	0.6896
0.6905	0.6894
0.6901	0.689
0.69	0.6887
0.6898	0.6885
0.6897	0.6881
0.6893	0.6877



其他数据

	A	B	C	D	E	F	G	H	I
1	pre_score1	pre_score2	pre	rec	SN	SP	f1	mcc	roc_auc
2	[0.43737439645217663,	0.789908462788117]	0.936599424	0.587172538	0.587172538	0.961147903	0.72182121	0.592806712	0.893897753
3	[0.396889906013137,	0.8211654396854519]	0.943008615	0.642728094	0.642728094	0.962030905	0.764437282	0.63970892	0.909505036
4	[0.3954132948237941,	0.8274168319323149]	0.915961306	0.684281843	0.684281843	0.938631347	0.783350569	0.645317525	0.91049961
5	[0.3989470023166976,	0.8254074546662152]	0.9143026	0.698735321	0.698735321	0.93598234	0.792114695	0.654537947	0.9091775
6	[0.38450601636085907,	0.8430453242135649]	0.926636569	0.741644083	0.741644083	0.942604857	0.823883593	0.699618134	0.91289596
7	[0.3790588745887683,	0.8566644340254521]	0.935086738	0.754742547	0.754742547	0.948785872	0.835291177	0.718298159	0.922907207
8	[0.36991304931003993,	0.8481803960926859]	0.952409639	0.714092141	0.714092141	0.965121413	0.816210635	0.703117121	0.933038002
9	[0.4187565763152638 6,	0.8381335115320956]	0.914158305	0.740740741	0.740740741	0.93200883	0.818363273	0.686424817	0.901733899
10	[0.4315685454916544,	0.8338914940894302]	0.900054915	0.74028907	0.74028907	0.919646799	0.812391574	0.671726039	0.899981654
11	[0.414269887917301,	0.8463942824290464]	0.923333333	0.750677507	0.750677507	0.939072848	0.828101644	0.703383594	0.909247494
12	[0.3969092429745514,	0.84728734286722]	0.914893617	0.757452575	0.757452575	0.931125828	0.828762046	0.70016461	0.919829063
13	[0.41384993633619244,	0.8642554127709571]	0.949381327	0.762420958	0.762420958	0.960264901	0.845691383	0.738489453	0.918306542
14	[0.4575412852685881,	0.8274168351926716]	0.913644214	0.716802168	0.716802168	0.933774834	0.803340926	0.667556431	0.90300486
15	[0.44593295572085634,	0.8486269238166017]	0.917118093	0.764679313	0.764679313	0.932450331	0.833990148	0.708100962	0.910794144
16	[0.4707253057660203,	0.8452779641106715]	0.915171289	0.760162602	0.760162602	0.931125828	0.830495929	0.702569628	0.897928993
17	[0.48264962467704936,	0.8263005137869385]	0.922039859	0.710478771	0.710478771	0.941280353	0.80255102	0.671035634	0.893852586
18	[0.4283320117100564,	0.8486269266245008]	0.941847206	0.746160795	0.746160795	0.954966887	0.83266129	0.718161267	0.916375922
19	[0.4224906369102243,	0.8539852622125425]	0.931567329	0.762420958	0.762420958	0.945253863	0.838549429	0.720885102	0.917140373
20	[0.4266398663092575,	0.8477338709371328]	0.919847328	0.761969286	0.761969286	0.935099338	0.833498024	0.708738365	0.9127123
21	[0.5097219468320845,	0.8169234185033569]	0.888222465	0.735772358	0.735772358	0.909492274	0.804841897	0.656067114	0.879721858
22	[0.4647861783606688,	0.8455012255972266]	0.917350848	0.757000903	0.757000903	0.933333333	0.829497649	0.702305857	0.901715353
23	[0.46747012135305954,	0.8345612860412623]	0.90027248	0.746160795	0.746160795	0.919205298	0.816003952	0.676448623	0.906467572
24	[0.3962016749139821,	0.8631390932016698]	0.945036455	0.761065944	0.761065944	0.956732892	0.843132349	0.733145284	0.936238187
25	[0.38988060645507927,	0.8457244941900287]	0.952690716	0.727642276	0.727642276	0.964679912	0.825096031	0.714024269	0.947461967
26	[0.46132764223897316,	0.8394730943179232]	0.899838449	0.754742547	0.754742547	0.917880795	0.820928519	0.68260985	0.90579545
27	[0.4786014659353121,	0.8341147573458932]	0.944981862	0.70596206	0.70596206	0.9598234	0.808169597	0.689731783	0.892658399
28	[0.4643522877729373,	0.8350078157746225]	0.904994571	0.752935863	0.752935863	0.922737307	0.82199211	0.686530297	0.907358252
29	[0.4626270006160221,	0.8334449651545247]	0.904632153	0.749774164	0.749774164	0.922737307	0.819955545	0.683713017	0.904069228
30	[0.47117900695351517,	0.8396963611807401]	0.905945946	0.757000903	0.757000903	0.923178808	0.82480315	0.690661786	0.913112423



谢谢