

异常检测接口说明

文件结构

代码块

1

AnomalyDetection

2

|— data

3

| |— Dataset1

4

| |— Node1.csv

5

|— clustering

6

| |— classify_result

7

| |— feature_center

8

|— model_sharing

9

| |— model

10

| |— result

11

| |— detecting.py

12

| |— TransformerAndMoe.py

13

| |— utils.py

14

|— config.yml

15

|— detect.sh

16

|— metric.json

17

|— requirements.txt

样例数据

样例数据

聚类相关结果目录

聚类结果目录

簇中心和特征向量目录

模型及检测结果目录

模型存储目录

检测结果目录

异常检测接口代码

模型定义代码

实用函数

配置文件

异常检测运行命令

指标文件

环境包

输入/输出

输入

一个表示多维时间序列数据的DataFrame，或者存为csv的文件（将通过pd.read_csv读取）

文件格式为：第一行为列索引，代表指标名，每一列都是一个指标的时间序列，每一行是当前这个节点某一刻的各个指标的观测值，如下：

metric1	metric2	metric3	...
1.0	0.8	1.9	...
0.9	0.9	1.8	...
0.8	1.0	1.7	...
...

可参考样例数据文件 `AnomalyDetection/data/Dataset1/Node1.csv`

数据需包含 `AnomalyDetection/metric.json` 文件中的所有指标

算法将会对数据进行预处理、异常检测等操作，最终输出异常点列表

输出

一个元素为int的list列表，其中每一个元素代表检测出的异常点的下标

例如：[1,33,1024,...] 表示下标为1，33，1024的时刻为异常点

运行

1. 安装环境

代码块

```
1 cd AnomalyDetection
2 conda create -n XXX python=3.8
3 pip install -r requirements.txt
```

2. 准备数据与运行算法

有两种方法：

- a. 将DataFrame数据作为函数输入，运行 `AnomalyDetection/model_sharing/detecting.py` 中的 `run_detect` 函数，例如：

代码块

```
1 node_df = pd.read_csv("../data/Dataset1/Node1.csv", index_col=0, header=0)
2 run_detect(node_df)
```

- b. 修改配置文件 `AnomalyDetection/config.yml` 中的 `data_file` 路径至待检测数据路径，随后在 `AnomalyDetection` 目录下运行 **sh detect.sh** 命令