

1.环境配置&需要安装的库

- pandas
- pytorch
- joblib
- pmdarima
- tqdm
- openpyxl: 3.1.0 (pip3 install openpyxl==3.1.0)
- xlrd
-

需要安装的库较多，同学们根据实际情况安装以及配置实验环境。本次实验运行需要GPU。

本次实验&数据代码地址：<https://github.com/tongjidam2023/benchmark>

数据集说明

模拟数据集系列：每个患者的血糖数据有8个特征，其中time(时间), CGM(动态时刻血糖值)是实验中需要关注的特征。

Ohio数据集：每个患者的血糖数据有18个特征，其中ts(时间), gl_value(glucose level value, CGM data)是实验中需要关注的特征。

Shanghai_T1DM&Shanghai_T2DM数据集：每个患者的血糖数据有11个特征，其中Date(时间), CGM (mg / dl)是实验中需要关注的特征。Dietary intake是摄入饮食信息，如在进行 GRU_{meal} 实验使用该特征时，需自行预处理并且修改相应代码逻辑。

任务1: 计算可预测性值

论文名称：Predicting Taxi and Uber Demand in Cities: Approaching the Limit of Predictability

论文地址：https://www.researchgate.net/publication/337876513_Predicting_Taxi_and_Uber_Demand_in_Cities_Approaching_the_Limit_of_Predictability

实现代码：`benchmark/predictability` 文件夹下，里面包含jupyter实例文件。

要求：对每一个患者根据代码计算其可预测性，作为该患者血糖预测的上限值。请将结果保留成csv格式的文件，随报告一同提交。

任务2: 根据不同模型对4个数据集实验

本次所有实验代码文件和数据文件都置于 `benchmark` 文件夹下。表格中的每一种方法与对应的代码文件关系如下表：

方法	文件/链接	备注
ARIMA	<code>ariam_train.py</code>	更换 <code>data_type</code> 调整数据集
LR	<code>classic_model.py</code>	更换 <code>one_train</code> 函数的第一个参数调整数据集
RF	<code>classic_model.py</code>	更换 <code>one_train</code> 函数的第一个参数调整数据集
Ridge	<code>classic_model.py</code>	更换 <code>one_train</code> 函数的第一个参数调整数据集
SVR	<code>classic_model.py</code>	更换 <code>one_train</code> 函数的第一个参数调整数据集
KNN	<code>classic_model.py</code>	更换 <code>one_train</code> 函数的第一个参数调整数据集
GRU_r	<code>DeepModels.py</code>	递归预测下GRU模型，运行py文件第1150行
GRU_m	<code>DeepModels.py</code>	多输出预测下GRU模型，运行py文件第1158行
GRU_{meal}	<code>DeepModels.py</code>	添加了饮食信息GRU模型，运行py文件第1166行
MICN	code available	自行根据GitHub链接复现

notes:

- 训练模型需要一定时间，同学们尽量尽早开始实验。
- 代码量较多，同学们需要对代码进行理解并根据自己的需求调整。需要在实验报告里详细描述你的实验环境和配置，包括所使用的python, pytorch的版本等。
- 每种时间序列方法中有可调整的超参数，同学们需要自己对不同的超参数进行试验调整，选择最优的超参设置保留并汇报在报告中。
- 在运行ARIMA, LR, RF, Ridge, SVR, KNN方法后，每个患者（如 `data/Simulator_adult/` 下的 `adult#002.csv` 和 `adult#006.csv` 分别代表两个不同的患者）可以得到30min和60min下的两个rmse值，仅代表该方法在患者下的血糖预测效果。同学们应将所有患者的平均rmse（必要时代码实现求均值）作为最终结果填写在表格里。最终分数根据表格中的值衡量。
- Simulator系列数据集中除了血糖数据CGM值可用之外，摄入碳水CHO值和胰岛素值insulin值也可以作为数据特征进行探索，同学们可以在代码基础之上改进探索。

任务3:

对代码debug和优化(不改变原有代码的逻辑的基础上)，在报告中阐述对代码的理解。在对代码理解的基础上，调整各种参数（如训练轮次epoch，学习率lr，深度学习优化器optim等）尽可能提升实验效果，需要在报告里说明对**实验代码的理解**，**调整了哪些参数**，以及**参数含义**，**调整原因**，以及**结果的分析与对比**等。代码里有一些函数，是本次实验并未涉及到的部分，忽略即可。

实验结果的保存：可将实验结果保存成CSV格式，txt格式，如保存成pkl或npz等格式，请附上解压缩脚本。

附录

- 部分代码运行后会自动将结果保存在pkl文件中，读 `result` 文件下pkl文件的代码（需自行安装joblib库）：

```
import joblib
import pprint

file=open("Ridge.pkl","rb")
data=joblib.load(file)
pprint.pprint(data)
file.close()
```