



مبانی یادگیری ماشین

گزارش تمرین سری آخر

عرشیا حسینمردی

شماره دانشجویی : 98222030

فایل requirement مربوط به پروژه که در صفحه آخر همین گزارش قرار دارد همینطور داخل پروژه (فولدر finalML1) نیز قرار دارد.

خیلی از requirement در پروژه وارد شده ولی استفاده نشده برای همین زیاد است.

در این گزارش سعی شده نحوه ورودی و خروجی و همینطور توضیحاتی درباره بخش های مختلف پروژه داده شود.

خروجی ها مانند همان مثالی است که خودتان در گروه فرستاده بودید ولی ممکن است فرق های کوچکی داشته باشد پس حتما قسمت ورودی و خروجی که از صفحه 3 شروع میشود مطالعه شود.

فایل کد ها در finalML1 است که در اصل فقط بخش کد پروژه flask است که در pycharm روی آن کار شده است. (به دلیل حجم بالای پروژه فقط کد های آن قرار گرفته شده است. که بخش اصلی در app.py است و دیگر کد ها در فولدر utils قرار دارد.

در ادامه توضیحات پروژه ها را میتوانید ببینید:

سرویس شماره 1 :

که در اصل هدف آن دورنیابی زمانی است که نوع ورودی و خروجی آن در صفحه 3 مشخص است که شبیه همان مثالی که خودتان حل کردین ، حل شده است. که کد های مربوط به آن در util.interpolatinmethods وجود دارد. (توجه شود که نوع ورودی شمسی و میلادی فرق دارد که در صفحه 3 توضیح داده شده است).

سرویس شماره 2:

هدف آن مثل بخش میلادی سوال اول ولی با خروجی متفاوت که باید به شکل تاریخ شمسی باید باشد که در همان util.interpolatinmethods قرار دارد ولی تابع های آن فرق دارد.

سرویس شماره 3:

سرویس برای Outlier detection در است که در دو حالت داده با time series و بدون time series است ، که برای حالت با time series از روش های isolation forest و DBSCAN استفاده کرده ایم و در حالت بدون time series از 3 روش IQR ، zscore و isolation forest استفاده کرده ایم. که کد های مربوط به آن در کلاس util.outlierMethods.py قرار دارد

سرویس شماره 4 :

در سرویس شماره 4 که مربوط به رفع مشکل imbalanced data است 4 روش پیاده سازی شده که در config باید مشخص شود. که این 4 روش ، روش های SMOTE ، oversampling ، undersampling ، tometlinks است. این سرویس فقط بر روی یک فیچر کار میکند. و کد مربوط به آن در utils.unbalanceDataMethods.py قرار دارد.

توجه شود که در روش SMOTE در کلاس minor باید حداقل دو عضو باشد .

در ادامه به شکل ورودی و خروجی میپردازیم:

ورودی سرویس 1 در حالت میلادی مثل ورودی مثال داخل کانال تلگرام است:

```
{
  "data": {
    "time": {
      "0": 1577836800000,
      "1": 1577923200000
    },
    "vol": {
      "0": 20,
      "1": 40
    }
  },
  "config": {
    "type": "miladi",
    "time": "daily",
    "interpolation": "linear"
  }
}
```

در ادامه حالت شمسی را بررسی میکنیم (صفحه بعدی)

در حالت شمسی نوع ورودی ها باید به شکل زیر باشد.

```
{
  "data": {
    "time": {
      "0": "1399-10-11",
      "1": "1399-10-15"
    },
    "vol": {
      "0": 20,
      "1": 40
    }
  },
  "config": {
    "type": "shamsi",
    "time": "daily",
    "interpolation": "linear"
  }
}
```

در اصل در حالت شمسی ورودی time به صورت string است.

Interpolation فقط به صورت linear و time میتواند "daily" یا "monthly" باشد و type نیز میتواند "miladi" یا "shamsi" باشد.

ورودی سرویس 2 دقیقا مثل مثال گروه است ولی به config ، skip_holiday نیاز نیست چون این قسمت پیاده سازی نشده است.

```
{
  "data": {
    "time": {
      "0": 1328227200000,
      "1": 1328400000000,
      "2": 1328486400000
    },
    "vol": {
      "0": 20,
      "1": 40,
      "2": 100
    }
  },
  "config": {
    "time": "daily",
    "interpolation": "linear"
  }
}
```

بخش time میتواند "monthly" یا "daily" باشد و فقط روش "linear" در interpolation موجود است.

ورودی سرویس سوم شبیه همان ورودی است که در فایل تست شما بوده که به شکل زیر است:

```
{
  "data": {
    "id": {
      "0": 1,
      "1": 2
    },
    "feature": {
      "0": 100,
      "1": 20,
    }
  },
  "config": {
    "time_series": false
  }
}
```

که اگر `time_series` برابر با `true` باشد به جای ستون `id` ستون تایم داریم و مثل `time` مثال یک است.(بخش میلادی آن منظور است)

خروجی سرویس شماره 4 به شکل زیر است:

```
{
  "data": {
    "id": {
      "0": 1,
      "1": 2,
      "2": 3
    },
    "feature1": {
      "0": 50,
      "1": 12,
      "2": 50
    },
    "class": {
      "0": 1,
      "1": 1,
      "2": 0
    }
  },
  "config": {
    "major_class": 1,
    "minor_class": 0,
    "method": "SMOTE"
  }
}
```

(ادامه توضیحات صفحه بعد)

که خروجی آن مثل Json ورودی است فقط داده ها balance شده است.

توجه شود در روش SMOTE در کلاس minor باید حداقل دو عضو باشد و روش های دیگر باید به شکل زیر وارد شود:

"method": "SMOTE"

"method": "oversampling"

"method": "undersampling"

"method": "tomeklinks"

و در آخر فایل مربوط به نیازمندی ها که در شکل زیر میتوانید مشاهده کنید.

```
click==8.1.3
colorama==0.4.5
Flask==2.1.2
imbalanced-learn==0.9.0
importlib-metadata==4.11.4
itsdangerous==2.1.2
jalali-pandas==0.2.2
jdatetime==3.8.2
Jinja2==3.1.2
joblib==1.1.0
Khayyam==3.0.18
MarkupSafe==2.1.1
numpy==1.21.6
pandas==1.3.5
pandas-jalali==0.1.2
python-dateutil==2.8.2
pytz==2022.1
scikit-learn==1.0.2
scipy==1.7.3
six==1.16.0
sklearn==0.0
threadpoolctl==3.1.0
typing_extensions==4.2.0
Werkzeug==2.1.2
zipp==3.8.0
```