

## 82.10 pandas.to\_datetime函数

批量处理时间数据。

主要参数，官方文档在本文最后给出

**pandas.to\_datetime** (*arg*, *errors*='raise', *utc* = None, *format* = None, *unit* = None)

参数	意义
errors	三种取值，'ignore', 'raise', 'coerce'，默认为raise。 'raise'，则无效的解析将引发异常 'coerce'，那么无效解析将被设置为NaT 'ignore'，那么无效的解析将返回输入值
utc	布尔值，默认为none。返回utc即协调世界时。
format	格式化显示时间的格式。
unit	默认值为'ns'，则将会精确到微妙，'s'为秒。

官方文档中有几个简单的例子，这里稍微提一下：

```
1、
df = pd.DataFrame({'year': [2015, 2016],
                   'month': [2, 3],
                   'day': [4, 5]})

pd.to_datetime(df)
#0    2015-02-04
#1    2016-03-05
```

```
#1      2010-03-03
#dtype: datetime64[ns]
#可以看到将字典形式时间转换为可读时间

2、
pd.to_datetime('13000101', format='%Y%m%d', errors='ignore
#datetime.datetime(1300, 1, 1, 0, 0)

pd.to_datetime('13000101', format='%Y%m%d', errors='coerce
#NaT
#如果日期不符合时间戳限制, 则errors = 'ignore' 将返回原始输入, 而不会
#errors='coerce' 将强制超出NaT的日期, 返回NaT。
```

然而实际中遇到的可能是这样的数据：

2018-01-01T17:40:05.355Z	2018-01-01T17:38:43.519Z	2018-03-22 16:31:44
		2018-03-22 16:31:18
2018-01-01T18:01:10.298Z	2018-01-01T18:01:08.368Z	2018-03-22 16:31:15
		2018-03-06 12:08:51
2018-01-01T20:20:11.953Z	2018-01-01T20:20:11.953Z	2018-03-09 14:40:22
		2018-03-16 17:45:42
2018-01-01T21:37:08.222Z	2018-01-01T21:35:44.899Z	2018-03-15 10:40:35
		2018-03-06 12:07:02

通过pandas.read\_csv()或者pandas.read\_excel()读取文件过后, 得到的数据列对应的类型是“object”, 这样没法对时间数据处理, 可以用过pd.to\_datetime将该列数据转换为时间类型, 即datetime。

```
data.dtypes
# object

data= pd.to_datetime(data)
data.dtypes
# datetime64[ns]
```

转换过后就可以对这些时间数据操作了，可以相减求时间差，计算相差的秒数和天数，调用的方法和datetime库的方法一致，分别是 **data.dt.days()**、**data.dt.seconds()**、**data.dt.total\_seconds()**。

到这里就结束了吗？不不，还没有。这里再教大家通过pandas将时间与时间戳相互转换，原理是一样的通过datetime、time库，技巧是用apply()函数来实现它。

将data的所有时间转换为时间戳（此时数据类型已经是datetime，是object的报错不要找我。。）

```
data = data.apply(lambda x:time.mktime(x.timetuple()))
#x.timetuple() 将时间转换为时间元组，提前导入time模块
```

将data的所有时间戳转换为可读时间：

```
data = data.apply(lambda x:time.strftime('%Y/%m/%d %H:%M:%S',time.localtime(x)))
#代码含义为：先读取时间戳数据，将其转换为时间元组，在通过格式化时间转换
```

---

官方文档: [pandas.to\\_datetime](#)