《用 Python 玩转数据》之利用免费财经数据接口 TuShare 获取和分析数据

by Dazhuang@NJU

1. 安装

在 Anaconda Prompt 窗口中输入如下命令安装：

> pip install tushare

2. 介绍

"TuShare 是一个免费、开源的 python 财经数据接口包。主要实现对股票等金融数据从数据采集、清洗加工 到 数据存储的过程，能够为金融分析人员提供快速、整洁、和多样的便于分析的数据，为他们在数据获取方面极大地减轻工作量，使他们更加专注于策略和模型的研究与实现上。考虑到 Python pandas 包在金融量化分析中体现出的优势，TuShare 返回的绝大部分的数据格式都是 pandas DataFrame 类型，非常便于用 pandas/NumPy/Matplotlib 进行数据分析和可视化。当然，如果您习惯了用 Excel 或者关系型数据库做分析，您也可以通过 TuShare 的数据存储功能，将数据全部保存到本地后进行分析。"这是 TuShare 官网(**http://tushare.org/index.html**)上对于 TuShare 的描述，它提供了便捷的各类财经数据和新闻等的接口。

3. 简单示例

例如要想获取股票代码是 600848 的股票在 2020 年 3 月 1 日至 3 月 8 日间的基本历史数据，只要使用如下代码即可：

```
>>> import tushare as ts
>>> ts.get_hist_data('600848',start='2020-03-01',end='2020-03-08')
```

date	open	high	close	low	volume	price_chang
2020-03-06	21.60	21.61	21.46	21.38	44897.09	-0.1
2020-03-05	21.87	21.87	21.64	21.50	63113.81	0.1
2020-03-04	21.13	21.54	21.47	20.97	52519.09	0.1
2020-03-03	21.28	21.77	21.30	21.10	62690.06	0.2
2020-03-02	20.52	21.10	21.08	20.52	51902.65	0.6

...

get_hist_data()函数可以获取三年内 A 股历史行情，其他 Tushare 中功能相似的函数还有 get_h_data()和 get_k_data()等，大家可自行查看函数说明并进行测试。

提示：如果要做正式发表的研究，数据尽量要与权威的财经网站比对核对。

小项目任务：

利用 Tushare 包中的接口函数获取招商银行（股票代码 600036）2019 年第一季度的股票数据并完成如下数据处理和分析任务：

1. 数据只保留 date、open、high、close、low 和 volume 这几个属性，并按时间先后顺序对数据进行排序；
2. 选择 2019 年一季度和 1 月该股票最高价 high 和最低价 low 数据。
3. 输出这一季度内成交量最低和最高那两天的日期和分别的成交量；
4. 列出成交量在 100000 以上的记录；
5. 计算这一季度中收盘价（close）高于开盘价（open）的天数；
6. 计算前后两天开盘价的涨跌情况，用两种方式表示，第一种输出每两天之间的差值（后一

天减去前一天），第二种输出一个开盘价涨跌列表，涨用 1 表示，跌用-1 表示；[提示：可使用 diff()方法和 sign()函数]
7. 绘制 2019 年 1 月该股票最高价 high 和最低价 low 的折线图；
8. 绘制该股票在此季度内每日收盘价与开盘价之差与当日成交量之间的散点图。。

【参考程序】

```
import tushare as ts
import numpy as np

# 1
df = ts.get_hist_data('600036', start = '2019-01-01', end = '2019-03-31')
df = df.iloc[:, :5]    #  获取前 5 列
df.sort_index(inplace = True)      #  按 date 列进行排序
# 2
print(df.loc[:, ['high', 'low']])            #  或 print(df[['high', 'low']])
print(df.loc['2019-01-01':'2019-01-31', ['high', 'low']])
# 3
volume_sorted = df.sort_values(by = 'volume')
min_day = volume_sorted.iloc[0,:]
min_volume = min_day.volume
min_volume_date = min_day.name
print("the min volume of {} is at {}".format(min_volume, min_volume_date))
max_day = volume_sorted.iloc[-1,]
max_volume = max_day.volume
max_volume_date = max_day.name
print("the max volume of {} is at {}".format(max_volume, max_volume_date))
# 4
print(df[df.volume >= 100000])
# 5
print(len(df[df.close > df.open]))
# 6
print(df.open.diff())
print(np.sign(np.diff(df.open)))
# 7
df_new = df.loc['2019-01-01':'2019-01-31',['high', 'low']]
df_new.sort_index().plot()
# 8
plt.scatter(df.close-df.open, df.volume)
```