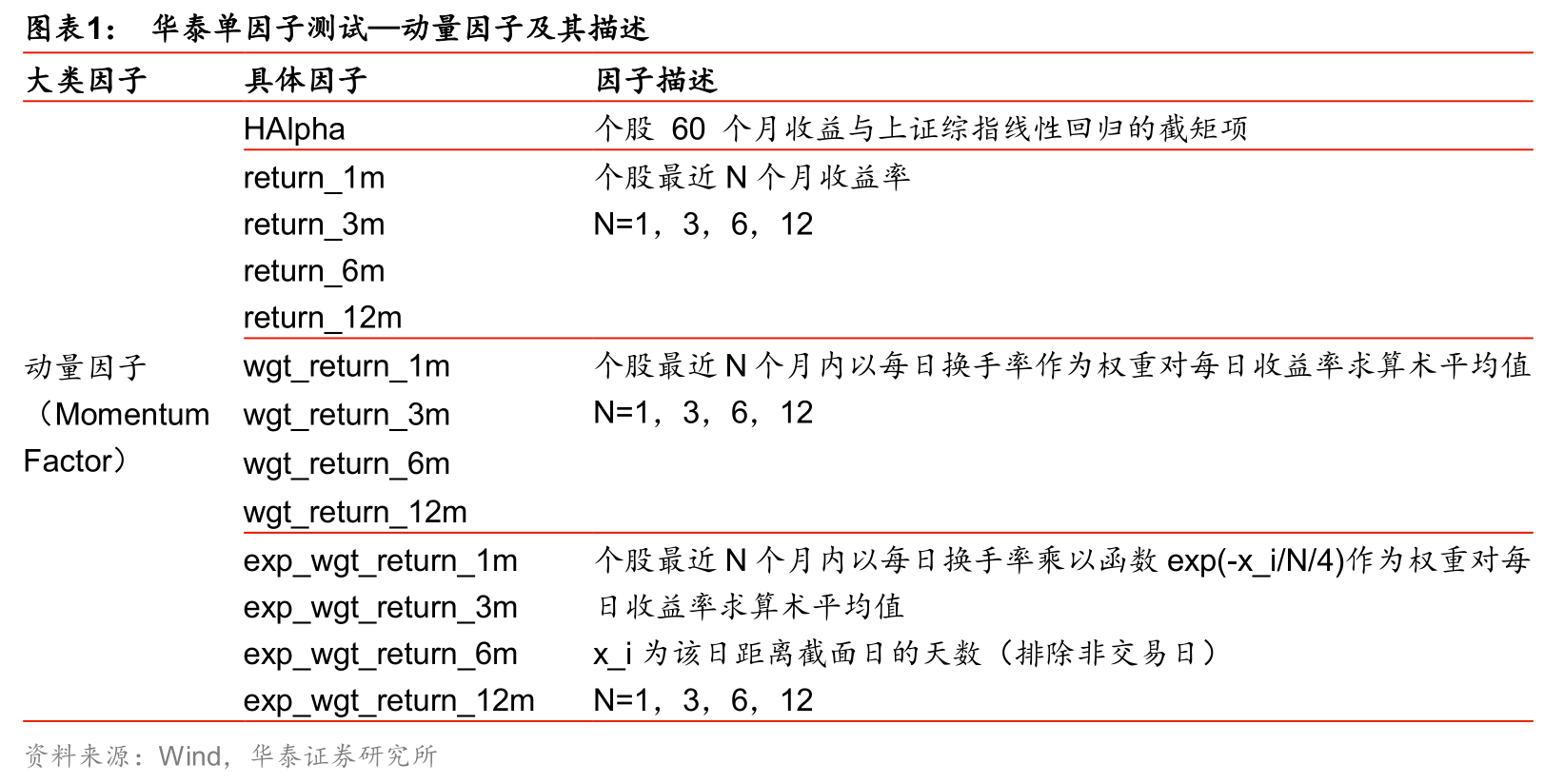
**三、计算方式及数据获取**

3.1 动量因子



已经在factor.py里有完整调用代码

*# 个股最近N月收益率 写好了 大概调用方式*

df\_monthly = pro.monthly(ts\_code=self.code, start\_date=**'20100101'**, end\_date=**'20191231'**,fields=**'ts\_code,trade\_date,close,pre\_close,pct\_chg'**)

df\_monthly[‘pct\_chg’] #一个月

df\_monthly[‘close’].pct\_change(periods=3); #三个月

df\_monthly[‘close’].pct\_change(periods=6); #六个月

df\_monthly[‘close’].pct\_change(periods=12); #十二个月

*# 个股最近N月内以当月算数平均每日换手率为权重的收益率 写好了*

df\_monthly = pro.monthly(ts\_code=self.code, start\_date=**'20100101'**, end\_date=**'20191231'**,fields=**'ts\_code,trade\_date,close,pre\_close,pct\_chg'**)

df\_daily\_basic=pro.daily\_basic(ts\_code=self.code,start\_date=**'20100101'**, end\_date=**'20191231'**,fields=**'ts\_code,trade\_date,turnover\_rate'**)

mon\_avg\_turnover=data\_basic\_daily.resample(**'M'**,on=**'trade\_date'**).mean();

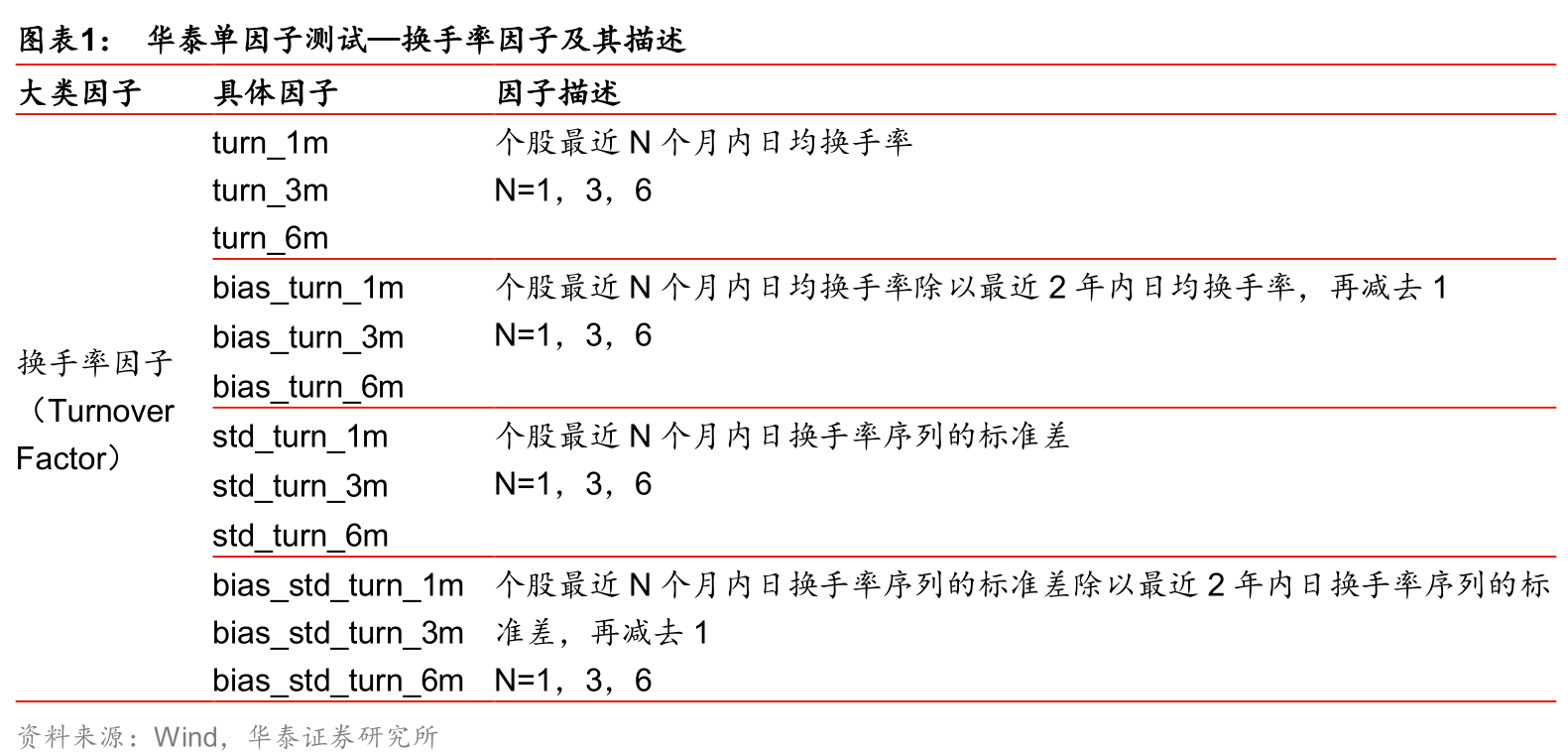
df\_monthly[‘pct\_chg’]/mon\_avg\_turnover #一个月

df\_monthly[‘close’].pct\_change(periods=3) /mon\_avg\_turnover; #三个月

df\_monthly[‘close’].pct\_change(periods=6) /mon\_avg\_turnover; #六个月

df\_monthly[‘close’].pct\_change(periods=12) /mon\_avg\_turnover; #十二个月

3.2 换手率因子



已经在factor.py里有完整调用代码

# 个股最近N个月日均换手率

df\_daily\_basic=pro.daily\_basic(ts\_code=self.code,start\_date=**'20100101'**, end\_date=**'20191231'**,fields=**'ts\_code,trade\_date,turnover\_rate'**)

mon\_avg\_turnover=data\_basic\_daily.resample(**'M'**,on=**'trade\_date'**).mean();

turn\_1m = pd.DataFrame({**'month'**:mon\_avg\_turnover[**'trade\_date'**],  
 i:mon\_avg\_turnover[**'turnover\_rate'**]});  
turn\_3m = pd.DataFrame({**'month'**: mon\_avg\_turnover[**'trade\_date'**],  
 i: mon\_avg\_turnover[**'turnover\_rate'**].pct\_change(periods=3)});  
turn\_6m = pd.DataFrame({**'month'**: mon\_avg\_turnover[**'trade\_date'**],  
 i: mon\_avg\_turnover[**'turnover\_rate'**].pct\_change(periods=6)});

# 个股最近N个月日均换手率除以最近1年内日均换手率

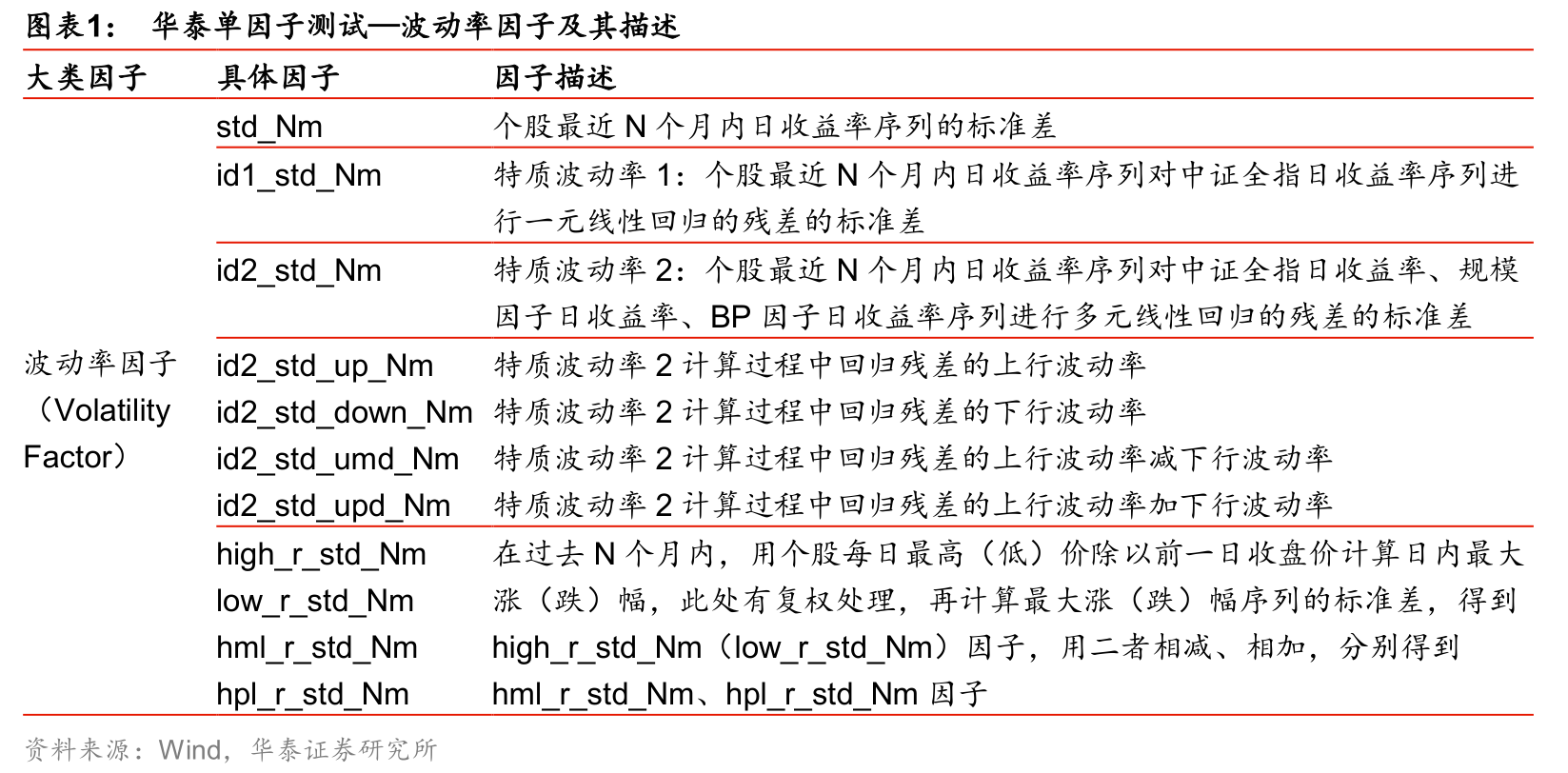
df\_daily\_basic=pro.daily\_basic(ts\_code=self.code,start\_date=**'20100101'**, end\_date=**'20191231'**,fields=**'ts\_code,trade\_date,turnover\_rate'**)

mon\_avg\_turnover=data\_basic\_daily.resample(**'M'**,on=**'trade\_date'**).mean();

year\_avg\_turnover=data\_basic\_daily.resample(**'Y'**,on=**'trade\_date'**).mean(); #最近一年内日均换手率

wgt\_turn\_1m = pd.DataFrame({**'month'**:mon\_avg\_turnover[**'trade\_date'**],  
 i:mon\_avg\_turnover[**'turnover\_rate'**]/ year\_avg\_turnover});  
wgt\_turn\_3m = pd.DataFrame({**'month'**: mon\_avg\_turnover[**'trade\_date'**],  
 i: mon\_avg\_turnover[**'turnover\_rate'**].pct\_change(periods=3)/ year\_avg\_turnover });  
wgt\_turn\_6m = pd.DataFrame({**'month'**: mon\_avg\_turnover[**'trade\_date'**],  
 i: mon\_avg\_turnover[**'turnover\_rate'**].pct\_change(periods=6)}/ year\_avg\_turnover);

3.3 波动率因子



这部分还没有写成.py 直接找调用接口

# 个股最近N个月内日收益率标准差

df\_daily= pro.daily(ts\_code=self.code, start\_date=**'20100101'**, end\_date=**'20191231'**);

daily\_return = df.daily[‘close’]-df.daily[‘pre\_close];

daily return\_1m = daily\_return.resample('M',on='trade\_date').mean();

daily return\_3m = daily\_return.resample('M',on='trade\_date').mean().pct\_change(periods=3);

daily return\_6m = daily\_return.resample('M',on='trade\_date').mean().pct\_change(periods=6);

daily return\_12m = daily\_return.resample('M',on='trade\_date').mean().pct\_change(periods=12);

……对这些daily\_return\_NM序列求标准差（交给周老师了）

# 特质波动率1：个股最近N个月内日收益率对沪深300收益率序列进行一元线性回归的残差的标准差

df\_daily= pro.daily(ts\_code=self.code, start\_date='20100101', end\_date='20191231');

daily\_return = df.daily[‘close’]-df.daily[‘pre\_close];

daily return\_1m = daily\_return.resample('M',on='trade\_date').mean();

daily return\_3m = daily\_return.resample('M',on='trade\_date').mean().pct\_change(periods=3);

daily return\_6m = daily\_return.resample('M',on='trade\_date').mean().pct\_change(periods=6);

daily return\_12m = daily\_return.resample('M',on='trade\_date').mean().pct\_change(periods=12);

index = pro.index\_daily(ts\_code = ‘000300’)

index\_return = index[‘close’]-index[‘pre\_close’];

index\_return\_1m = daily\_return.resample('M',on='trade\_date').mean();

index\_return\_3m = daily\_return.resample('M',on='trade\_date').mean().pct\_change(periods=3);

index\_return\_6m = daily\_return.resample('M',on='trade\_date').mean().pct\_change(periods=6);

index\_return\_12m = daily\_return.resample('M',on='trade\_date').mean().pct\_change(periods=12);

……daily\_return\_NM对index\_return\_NM进行一元线性回归算残差 再算残差的标准差