**Five\_Factor\_Replication Comparison with Answer**

**Comp部分**

1. Line7多填充0了
2. comp = comp[comp['cogs'].notna()|comp['xsga'].notna()|comp['xint'].notna()]

这一部份处理的比答案细，不错

1. revt答案是填充0，你是dropna
2. comp['op'] = comp['op'] / (comp['be'] + comp['mib'])

mib通常指的是“市场投资组合的账面价值与市场价值之比”（Market-to-Book ratio）

但是论文中的描述是 “For portfolios formed in June of year t, profitability (measured with accounting data for the fiscal year ending in t-1) is annual revenues minus cost of goods sold, interest expense, and selling, general, and administrative expenses, all divided by book equity at

the end of fiscal year t- 1.

这种调整可以帮助消除因资本结构和市场环境不同而导致的比较偏差。

1. 答案没有对lat填充0也没有dropna

CRSP部分

1. crsp = crsp.dropna(subset=['prc','shrout'])

crsp = crsp.drop\_duplicates(subset=['permno', 'jdate'], keep='first')

这两步答案都没有

我删除重复值是因为发现有股票在多家交易所上市

2. decme['year']=decme['year']+1

核查一下这一步时间是不是错了：我的逻辑是上一年年底数据应用于下一年

没问题 答案在下一步也这么干了

3. crsp\_jun = crsp\_jun[crsp\_jun['mebase'].notna()]

答案没有dropna

4. ccm1['linkdt'] = pd.to\_datetime(ccm1['linkdt']).fillna(pd.Timestamp('1960-01-01'))  
ccm1['linkenddt'] = pd.to\_datetime(ccm1['linkenddt']).fillna(pd.Timestamp('2023-12-31'))  
ccm1 = ccm1.dropna(subset=['linkdt', 'linkenddt'])

这一部份处理的较为可笑

5. ccm\_jun = ccm\_jun.dropna(subset=['be', 'op', 'agr', 'dec\_me'])

你又dropna了

1. ccm\_jun['beme'] = np.where(ccm\_jun['dec\_me'] == 0, np.nan, ccm\_jun['be'] \* 1000 / ccm\_jun['dec\_me'])

answer没有进行除0处理！！！

ccm\_jun['exchcd'].isin([1, 2, 3])） 可以学习的写法 比==更好拓展

nyse断点部分

1. nyse.loc[:, 'op'] = winsorize(nyse['op'], limits=[0.01, 0.01])
2. nyse.loc[:, 'agr'] = winsorize(nyse['agr'], limits=[0.01,

额外处理了极值

=

ccm1\_jun = ccm1\_jun.dropna(subset=['beme'])

你为啥要dropna呢

ccm1\_jun['nonmissport'] = np.where((ccm1\_jun['bmport'] != ''), 1, 0)

vs

ccm1\_jun['nonmissport']=np.where((ccm1\_jun['bmport']!='')&  
 (ccm1\_jun['szport']!='')&  
 (ccm1\_jun['opport']!='')&  
 (ccm1\_jun['agrport']!=''), 1, 0)

感觉我的更好啊

这一段让两者相互匹配是否有必要呢

years\_in\_june = june['ffyear'].unique()  
crsp3\_filtered = crsp3[crsp3['ffyear'].isin(years\_in\_june)]  
  
# Step 2: Further filter crsp3 to only include permnos present in june  
permnos\_in\_june = june['permno'].unique()  
crsp3\_filtered = crsp3\_filtered[crsp3\_filtered['permno'].isin(permnos\_in\_june)]

portfolio部分处理

答案多了这一个步骤

vwret\_szbmopinv = ccm4.groupby(['jdate', 'szport', 'bmport','opport','invport']).apply(wavg, 'retadj', 'wt').to\_frame().reset\_index().rename(  
 columns={0: 'vwret'})

答案里384行还有错的呢

\_ffcomp70\_24['WSMB'] = \_ffcomp70\_24['WSMB'] \* 100  
\_ffcomp70\_24['WHML'] = \_ffcomp70\_24['WHML'] \* 100  
\_ffcomp70\_24['WRMW'] = \_ffcomp70\_24['WRMW'] \* 100  
\_ffcomp70\_24['WCMA'] = \_ffcomp70\_24['WCMA'] \* 100

答案是在最后output的时候才\*100

Vs

vwret['vwret'] = vwret['vwret']\*100

我在计算portfolio return的时候就\*100