

Replicated Regressions-Copy1

March 7, 2019

```
In [22]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats
from sklearn import linear_model
from tqdm import tqdm

In [23]: column_names = ['Stock Symbol', 'YearMonth', 'NumEmployees', 'AverageAge', 'NumPeople']
month_current = pd.read_csv('fastwhitepaper_month_current.csv', sep="\t", error_bad_lines=False)
month_join = pd.read_csv('fastwhitepaper_month_join.csv', sep="\t", error_bad_lines=False)
month_leave = pd.read_csv('fastwhitepaper_month_leave.csv', sep="\t", error_bad_lines=False)

In [24]: month_current = month_current.rename(columns={'NumEmployees': 'NumEmployeesCurrent'})
month_join = month_join.rename(columns={'NumEmployees': 'NumEmployeesJoin'})
month_leave = month_leave.rename(columns={'NumEmployees': 'NumEmployeesLeave'})

In [25]: month_current_employees = month_current.iloc[:, 0:3]
month_join_employees = month_join.iloc[:, 2:3]
month_leave_employees = month_leave.iloc[:, 2:3]

In [26]: frames = [month_current_employees, month_join_employees, month_leave_employees]
month_combined = pd.concat(frames, axis=1, join='inner')
month_combined.head()
```

```
Out[26]:
```

| | Stock Symbol | YearMonth | NumEmployeesCurrent | NumEmployeesJoin | \ |
|---|--------------|-----------|---------------------|------------------|---|
| 0 | AGO | 199001 | 2 | 0 | |
| 1 | AGO | 199002 | 2 | 0 | |
| 2 | AGO | 199003 | 2 | 0 | |
| 3 | AGO | 199004 | 3 | 1 | |
| 4 | AGO | 199005 | 3 | 0 | |

| | NumEmployeesLeave |
|---|-------------------|
| 0 | 0 |
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 0 |

```
In [27]: num_employees_current = month_combined.loc[:, 'NumEmployeesCurrent'].values.astype(float)
num_employees_join = month_combined.loc[:, 'NumEmployeesJoin'].values.astype(float)
num_employees_leave = month_combined.loc[:, 'NumEmployeesLeave'].values.astype(float)
join_depart_sum = np.add(num_employees_join, num_employees_leave)
turnover = np.divide(join_depart_sum,
                     num_employees_current,
                     out=(np.zeros_like(join_depart_sum)),
                     where=(num_employees_current > 0))
turnover_df = pd.DataFrame(turnover, columns=['Turnover Rate'])
```

```
In [28]: frames = [month_combined, turnover_df]
month_combined_with_turnover = pd.concat(frames, axis=1, join='inner')
month_combined_with_turnover.head()
```

```
Out [28]:
```

| | Stock Symbol | YearMonth | NumEmployeesCurrent | NumEmployeesJoin | \ |
|---|--------------|-----------|---------------------|------------------|---|
| 0 | AGO | 199001 | 2 | 0 | |
| 1 | AGO | 199002 | 2 | 0 | |
| 2 | AGO | 199003 | 2 | 0 | |
| 3 | AGO | 199004 | 3 | 1 | |
| 4 | AGO | 199005 | 3 | 0 | |

| | | NumEmployeesLeave | Turnover Rate |
|---|--|-------------------|---------------|
| 0 | | 0 | 0.000000 |
| 1 | | 0 | 0.000000 |
| 2 | | 0 | 0.000000 |
| 3 | | 0 | 0.333333 |
| 4 | | 0 | 0.000000 |

```
In [29]: minDate = min(month_combined_with_turnover.loc[:, 'YearMonth'])
maxDate = max(month_combined_with_turnover.loc[:, 'YearMonth'])
maxDate
```

```
Out [29]: 201707
```

```
In [30]: winsorized_turnover = scipy.stats.mstats.winsorize(month_combined_with_turnover["Turnover Rate"])
winsorized_turnover
```

```
Out [30]: masked_array(data=[0., 0., 0., ..., 0., 0., 0.],
                       mask=False,
                       fill_value=1e+20)
```

```
In [31]: winsorized_turnover_df = pd.DataFrame(winsorized_turnover, columns=['Winsorized Turnover Rate'])
```

```
In [32]: frames = [month_combined, winsorized_turnover_df]
month_combined_with_winsorized_turnover = pd.concat(frames, axis=1, join='inner')
month_combined_with_winsorized_turnover.head()
```

```
Out [32]:
```

| | Stock Symbol | YearMonth | NumEmployeesCurrent | NumEmployeesJoin | \ |
|---|--------------|-----------|---------------------|------------------|---|
| 0 | AGO | 199001 | 2 | 0 | |

| | | | | |
|---|-----|--------|---|---|
| 1 | AGO | 199002 | 2 | 0 |
| 2 | AGO | 199003 | 2 | 0 |
| 3 | AGO | 199004 | 3 | 1 |
| 4 | AGO | 199005 | 3 | 0 |

| | NumEmployeesLeave | Winsorized Turnover Rate |
|---|-------------------|--------------------------|
| 0 | 0 | 0.00 |
| 1 | 0 | 0.00 |
| 2 | 0 | 0.00 |
| 3 | 0 | 0.25 |
| 4 | 0 | 0.00 |

```
In [33]: column_names = ['gvkey', 'datadate', 'fyearq', 'fqtr', 'indfmt', 'consol', 'popsrc', 'datafmt', 'tic', 'cusip', 'rdq', 'ceqq', 'cshoq', 'epsf12', 'epsfxq', 'xrdq', 'naics']
compustat_data = pd.read_csv('Compustat_2000_2016.csv', sep="\t", names = column_names)
compustat_data.head()
```

```
Out [33]:
```

| | gvkey | datadate | fyearq | fqtr | indfmt | consol | popsrc | datafmt | tic | cusip | \ |
|---|-------|----------|--------|------|--------|--------|--------|---------|-----|-----------|---|
| 0 | 1004 | 20000229 | 1999 | 3 | INDL | C | D | STD | AIR | 000361105 | |
| 1 | 1004 | 20000531 | 1999 | 4 | INDL | C | D | STD | AIR | 000361105 | |
| 2 | 1004 | 20000831 | 2000 | 1 | INDL | C | D | STD | AIR | 000361105 | |
| 3 | 1004 | 20001130 | 2000 | 2 | INDL | C | D | STD | AIR | 000361105 | |
| 4 | 1004 | 20010228 | 2000 | 3 | INDL | C | D | STD | AIR | 000361105 | |

| | ... | datafqtr | rdq | ceqq | cshoq | epsf12 | epsfxq | xrdq | \ |
|---|-----|----------|------------|---------|--------|--------|--------|------|---|
| 0 | ... | 1999Q3 | 20000315.0 | 342.482 | 26.963 | 1.61 | 0.40 | NaN | |
| 1 | ... | 1999Q4 | 20000628.0 | 339.515 | 26.865 | 1.28 | 0.09 | NaN | |
| 2 | ... | 2000Q1 | 20000920.0 | 339.253 | 26.857 | 1.01 | 0.12 | NaN | |
| 3 | ... | 2000Q2 | 20001220.0 | 341.264 | 26.932 | 0.77 | 0.16 | NaN | |
| 4 | ... | 2000Q3 | 20010320.0 | 344.865 | 26.945 | 0.57 | 0.20 | NaN | |

| | costat | prccq | naics |
|---|--------|--------|----------|
| 0 | A | 23.750 | 423860.0 |
| 1 | A | 13.875 | 423860.0 |
| 2 | A | 11.250 | 423860.0 |
| 3 | A | 10.375 | 423860.0 |
| 4 | A | 13.600 | 423860.0 |

[5 rows x 22 columns]

```
In [34]: report_date_df = pd.DataFrame(compustat_data['datadate'].values, columns=["Report Date"])
report_date_df.head()
```

```
Out [34]:
```

| | Report Date |
|---|-------------|
| 0 | 20000229 |
| 1 | 20000531 |
| 2 | 20000831 |
| 3 | 20001130 |
| 4 | 20010228 |

```
In [35]: ticker = compustat_data['tic'].values
        ticker_df = pd.DataFrame(ticker, columns=["Stock Symbol"])
        frames = [report_date_df, ticker_df]
        date_ticker_df = pd.concat(frames, axis=1, join='inner')
        date_ticker_df.head()
```

```
Out[35]:
```

| | Report Date | Stock Symbol |
|---|-------------|--------------|
| 0 | 20000229 | AIR |
| 1 | 20000531 | AIR |
| 2 | 20000831 | AIR |
| 3 | 20001130 | AIR |
| 4 | 20010228 | AIR |

```
In [36]: shares_outstanding = compustat_data['cshoq'].values
        price_per_share = compustat_data['prccq'].values
        market_capitalization = np.multiply(shares_outstanding, price_per_share)
        size = np.log(market_capitalization)
        size_df = pd.DataFrame(size, columns=["Size"])
        frames = [report_date_df, size_df]
        date_size_df = pd.concat(frames, axis=1, join='inner')
        date_size_df.head()
```

/Users/timothyhuang/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:4: RuntimeWarning: after removing the cwd from sys.path.

```
Out[36]:
```

| | Report Date | Size |
|---|-------------|----------|
| 0 | 20000229 | 6.462048 |
| 1 | 20000531 | 5.920913 |
| 2 | 20000831 | 5.710895 |
| 3 | 20001130 | 5.632714 |
| 4 | 20010228 | 5.903868 |

```
In [37]: book_value = compustat_data['ceqq'].values
        book_to_market_ratio = np.divide(book_value, market_capitalization, out=np.zeros_like
        book_to_market_ratio_df = pd.DataFrame(book_to_market_ratio, columns=["Book to Market
        frames = [report_date_df, book_to_market_ratio_df]
        date_book_to_market_ratio_df = pd.concat(frames, axis=1, join='inner')
        date_book_to_market_ratio_df.head()
```

```
Out[37]:
```

| | Report Date | Book to Market Ratio |
|---|-------------|----------------------|
| 0 | 20000229 | 0.534818 |
| 1 | 20000531 | 0.910834 |
| 2 | 20000831 | 1.122829 |
| 3 | 20001130 | 1.221332 |
| 4 | 20010228 | 0.941092 |

```
In [38]: def slicer_vectorized(a,start,end):
        b = a.view((str,1)).reshape(len(a),-1)[:start:end]
```

```

        return np.fromstring(b.tostring(),dtype=(str,end-start))

naics_industry = compustat_data['naics'].values
naics_industry = slicer_vectorized(naics_industry.astype(str),0,2) #slicing naics by

naics_industry_df = pd.DataFrame(naics_industry, columns=["NAICS Industry Classification"])
frames = [report_date_df, naics_industry_df]
date_naics_industry_df = pd.concat(frames, axis=1, join='inner')
date_naics_industry_df.head()

```

/Users/timothyhuang/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:3: DeprecationWarning: This is separate from the ipykernel package so we can avoid doing imports until

```

Out[38]: Report Date NAICS Industry Classification
0      20000229      42
1      20000531      42
2      20000831      42
3      20001130      42
4      20010228      42

```

```

In [39]: # combined controls data
frames = [report_date_df, ticker_df, size_df, book_to_market_ratio_df, naics_industry_df]
combined_controls = pd.concat(frames, axis=1, join='inner')
combined_controls.head()

```

```

Out[39]: Report Date Stock Symbol      Size Book to Market Ratio \
0      20000229      AIR  6.462048      0.534818
1      20000531      AIR  5.920913      0.910834
2      20000831      AIR  5.710895      1.122829
3      20001130      AIR  5.632714      1.221332
4      20010228      AIR  5.903868      0.941092

      NAICS Industry Classification
0      42
1      42
2      42
3      42
4      42

```

```

In [40]: len(month_combined_with_winsorized_turnover.loc[month_combined_with_winsorized_turnover['Stock Symbol'] == 'PM'])

```

```

Out[40]: 956885

```

```

In [43]: df1 = month_combined_with_winsorized_turnover
df2 = df1.loc[(df1['Stock Symbol'] == 'PM') & (df1['YearMonth'] == 201405)]
df2 = df2.append(df1.loc[(df1['Stock Symbol'] == 'MRK') & (df1['YearMonth'] == 200102)])
df2 = df2.append(df1.loc[(df1['Stock Symbol'] == 'C') & (df1['YearMonth'] == 200312)])
df2 = df2.append(df1.loc[(df1['Stock Symbol'] == 'JNJ') & (df1['YearMonth'] == 200805)])

```

```

df2 = df2.append(df1.loc[(df1['Stock Symbol'] == 'GOOG') & (df1['YearMonth'] == 201412)])
df2 = df2.append(df1.loc[(df1['Stock Symbol'] == 'APPL') & (df1['YearMonth'] == 200501)])
df2 = df2.append(df1.loc[(df1['Stock Symbol'] == 'IBM') & (df1['YearMonth'] == 201001)])
df2 = df2.append(df1.loc[(df1['Stock Symbol'] == 'GS') & (df1['YearMonth'] == 200708)])
df2 = df2.append(df1.loc[(df1['Stock Symbol'] == 'MCD') & (df1['YearMonth'] == 200212)])
df2 = df2.append(df1.loc[(df1['Stock Symbol'] == 'DAL') & (df1['YearMonth'] == 201105)])
df2

```

```

Out [43]:
      Stock Symbol  YearMonth  NumEmployeesCurrent  NumEmployeesJoin  \
1348786          PM    201405                12397                469
104067          MRK    200102                13255                244
220613           C    200312                34250                502
22397          JNJ    200805                22435                683
1021103         GOOG    201412                46809                959
828733          IBM    201001               189924               8171
13782           GS    200708                13823                307
1310253         MCD    200212                20822                209
1406344         DAL    201105                14695                332

      NumEmployeesLeave  Winsorized Turnover Rate
1348786              366                0.067355
104067              114                0.027009
220613              633                0.033139
22397              451                0.050546
1021103            1267                0.047555
828733            3328                0.060545
13782              711                0.073645
1310253            332                0.025982
1406344            135                0.031780

```

```

In [48]: df3 = combined_controls
df4 = df3.loc[(df3['Stock Symbol'] == 'PM') & (df3['Report Date'] // 100 <= 201405)]
df4 = df4.append(df3.loc[(df3['Stock Symbol'] == 'MRK') & (df3['Report Date'] // 100 <= 200102)])
df4 = df4.append(df3.loc[(df3['Stock Symbol'] == 'C') & (df3['Report Date'] // 100 <= 200312)])
df4 = df4.append(df3.loc[(df3['Stock Symbol'] == 'JNJ') & (df3['Report Date'] // 100 <= 200805)])
df4 = df4.append(df3.loc[(df3['Stock Symbol'] == 'GOOG') & (df3['Report Date'] // 100 <= 201412)])
df4 = df4.append(df3.loc[(df3['Stock Symbol'] == 'APPL') & (df3['Report Date'] // 100 <= 200501)])
df4 = df4.append(df3.loc[(df3['Stock Symbol'] == 'IBM') & (df3['Report Date'] // 100 <= 201001)])
df4 = df4.append(df3.loc[(df3['Stock Symbol'] == 'GS') & (df3['Report Date'] // 100 <= 200708)])
df4 = df4.append(df3.loc[(df3['Stock Symbol'] == 'MCD') & (df3['Report Date'] // 100 <= 200212)])
df4 = df4.append(df3.loc[(df3['Stock Symbol'] == 'DAL') & (df3['Report Date'] // 100 <= 201105)])

```

```

In [49]: # Function to match turnover data with most recent controls data
# Winsorized turnover dataframe has columns (Stock Symbol, YearMonth, NumEmployeesCurrent,
# NumEmployeesJoin, NumEmployeesLeave, Winsorized Turnover Rate)
# Controls dataframe has columns (Report date, Stock Symbol, Size, Book to Market Ratio,
# NAICS Industry Classification)

```

```

def match_func(wins_turnover_df, controls_df):
    matched_dict = dict()
    wins_turnover_df = wins_turnover_df.loc[wins_turnover_df['YearMonth'] >= 200000]
    for index, row in tqdm(wins_turnover_df.iterrows()):
        nearest_row = None
        pruned_controls = controls_df.loc[(controls_df['Stock Symbol'] == row['Stock Symbol']) & (controls_df['Report Date'] >= row['YearMonth'])]
        if (len(pruned_controls) > 0):
            nearest_index = pruned_controls['Report Date'].idxmax()
            nearest_row = pruned_controls.loc[nearest_index]
        matched_dict[(row['Stock Symbol'], row['YearMonth'], row['Winsorized Turnover'])] = nearest_row
    return matched_dict

```

In [51]: matched_dict = match_func(df2, df4)

9it [00:00, 442.36it/s]

In [56]: matched_df = pd.DataFrame(matched_dict)
matched_df = matched_df.transpose()
matched_df.head(10)

Out [56]:

| | | | | Report Date | Stock Symbol | | Size | Book to Market Ratio \ |
|------|--------|----------|--|-------------|--------------|---------|------|------------------------|
| PM | 201405 | 0.067355 | | 20140331 | PM | 11.7679 | | -0.0667442 |
| MRK | 200102 | 0.027009 | | 20001231 | MRK | 12.2833 | | 0.068653 |
| C | 200312 | 0.033139 | | 20031231 | C | 12.4305 | | 0.387063 |
| JNJ | 200805 | 0.050546 | | 20080331 | JNJ | 12.116 | | 0.249626 |
| GOOG | 201412 | 0.047555 | | None | None | None | | None |
| IBM | 201001 | 0.060545 | | 20091231 | IBM | 12.0487 | | 0.132482 |
| GS | 200708 | 0.073645 | | 20070831 | GS | 11.2238 | | 0.480931 |
| MCD | 200212 | 0.025982 | | 20021231 | MCD | 9.92293 | | 0.504147 |
| DAL | 201105 | 0.031780 | | 20110331 | DAL | 9.02211 | | 0.0914991 |

| | | | | NAICS Industry Classification |
|------|--------|----------|--|-------------------------------|
| PM | 201405 | 0.067355 | | 31 |
| MRK | 200102 | 0.027009 | | 32 |
| C | 200312 | 0.033139 | | 52 |
| JNJ | 200805 | 0.050546 | | 32 |
| GOOG | 201412 | 0.047555 | | None |
| IBM | 201001 | 0.060545 | | 51 |
| GS | 200708 | 0.073645 | | 52 |
| MCD | 200212 | 0.025982 | | 72 |
| DAL | 201105 | 0.031780 | | 48 |

In []: