Stock Trades

Michael Chen, Siyu Chen

Summary of Findings

Introduction

The dataset was gathered from Timothy Carambat's project of House Stock Watcher (https://housestockwatcher.com/). The csv file contains 12 columns after transforming into Dataframe. The major columns we focused on include the date, type, assets, amount of the transactions and the representatives who made the transactions, as well as its party affiliation. We examined these columns to figure out potential correlations between two parties and the detail of transactions they made.

Cleaning and EDA

For easy analysis purpose, we transformed 'transaction_data' and 'disclosure_date' into datetime object. There are some dates in misordered format, so we manually handle them. Also we decided to replace 'amount' column with its mean as float value for later hypothesis questions.

Party affiliation was gathered from https://ballotpedia.org/List_of_current_members_of_the_U.S._Congress. It includes the current U.S. House members' party affiliations. To merge it with original datasets, we conducted three steps:

1. We merged with lower case of names using left join method
2. We find names with two mor common names when encountering similar but not exactly the same name.
3. we manually fill in most of the rest missing values of party when names differ too much.

Assessment of Missingness

By checking all columns values, we discovered that missing values also appeared in format of '--', so we placed them with np.NaN. Then, we performed permutation tests on column 'owner' with other columns 'type', 'amount', and 'transaction_date' to determine whether the missingness of 'owner' column is MCAR or MAR. The p-value of all three permutation test is 0 which means that we reject the null hypothesis that the data of 'owner' is missing completely at random. The missingness of 'owner' is depend on three other columns.

Hypothesis Test

1. Does one party trade more often?

- Null: Two parties trade equally often.
- Althernative: One party trades more often.
- Significance Level: 0.05
- p-value: 0.0
- Conclusion: The null is rejected, Democrats trade more often.

1. Does one party make larger trades?

- Null: Two parties trade with equal size.
- Althernative: One party trades larger size.
- Significance Level: 0.05
- p-value: around 0.38
- Conclusion: We failed to reject the null, two parties traded equally large.

1. Do the two parties invest in different stocks or sectors? For instance, do Democrats invest in Tesla more than Republicans?

- Null: Two parties invest in same stocks or sectors.
- Althernative: One party invest in different stocks or sectors.
- Significance Level: 0.05
- p-value: 0.0
- Conclusion: The null is rejected, two parties invest in different stocks or sectors.

Code

In [541…

```
import matplotlib.pyplot as plt
import numpy as np
import os
import pandas as pd
import seaborn as sns
```

Cleaning and EDA

Cleaning

Transform all missing values to NaN

In [542...

```
data = pd.read_csv("all_transactions.csv")
data = data.replace('--', np.NaN)
```

Transforming columns with dates to datetime object and replaced confusing dates

In [543...

```
data.loc[data['transaction_date'] == '20221-11-18', 'transaction_date'] = '2021-11-18'
data.loc[data['transaction_date'] == '0021-08-02', 'transaction_date'] = '2021-08-02'
data.loc[data['transaction_date'] == '0021-06-22', 'transaction_date'] = '2021-06-22'
data.loc[data['transaction_date'] == '0201-06-22', 'transaction_date'] = '2021-06-22'
data = data.drop(data.loc[data['transaction_date'] == '0009-06-09'].index)
data['transaction_date'] = pd.to_datetime(data['transaction_date'])
data['disclosure_date'] = pd.to_datetime(data['disclosure_date'])
```

Handling unfaithful transactions amount

In [544...

```
data['amount'] = data['amount'].replace('$1,001 -', '$1,001 - $15,000')
data['amount'] = data['amount'].replace('$1,000 - $15,000', '$1,001 - $15,000')
data['amount'] = data['amount'].replace('$15,000 - $50,000', '$15,001 - $50,000')
data['amount'] = data['amount'].replace('$50,000,000 +', '$50,000,000 - $1,000,000,000')
data['amount'] = data['amount'].replace('$1,000,000 +', '$1,000,001 - $5,000,000')
data['amount'] = data['amount'].replace('$1,000,000 - $5,000,000', '$1,000,001 - $5,000,000')
```

Merging party affiliation with original data

In [545...

```
# First step of Party affiliation merging

party = pd.read_csv("party_aff.csv")
party = party[['Name', 'Party']]
data['representative'] = data['representative'].apply(lambda x: x.strip()[5:])
data['representative'] = data['representative'].str.lower()
party['Name'] = party['Name'].str.lower()
data.loc[data['representative'] == 'eter meijer', 'representative'] = 'peter meijer'
merged = pd.merge(data, party, left_on='representative', right_on='Name', how='left')
merged = merged.drop(['Name'], axis=1)
```

```
merged = merged.sort_values(by=['transaction_date'])
merged
```

Out[545...

	disclosure_year	disclosure_date	transaction_date	owner	ticker	asset_description	type	amount	representative	district	
9739	2021	2021-08-26	2012-06-19	NaN	BLFSD	BioLife Solutions Inc	purchase	$1,001$ — 15,000	tom malinowski	NJ07	clerk.h
10451	2022	2022-03-03	2017-09-05	NaN	SUP	Superior Industries International Inc Common S...	purchase	$1,001$ — 15,000	thomas suozzi	NY03	clerk.h
10432	2022	2022-03-03	2017-12-06	NaN	CAT	Caterpillar Inc	purchase	$1,001$ — 15,000	thomas suozzi	NY03	clerk.h
10431	2022	2022-03-03	2018-04-17	NaN	BA	Boeing Company	purchase	$15,001$ — 50,000	thomas suozzi	NY03	clerk.h
10437	2022	2022-03-03	2018-04-30	NaN	CTRL	Control4 Corporation	purchase	$1,001$ — 15,000	thomas suozzi	NY03	clerk.h
...	...	...	...	...	...	...	...	...	...	...	
13717	2022	2022-05-04	2022-04-28	joint	MMP	Magellan Midstream Partners LP Limited Partner...	purchase	$1,001$ — 15,000	virginia foxx	NC05	clerk.h
13733	2022	2022-05-04	2022-04-29	joint	TTE	TotalEnergies Inc	purchase	$1,001$ — 15,000	virginia foxx	NC05	clerk.h
13736	2022	2022-05-04	2022-04-29	joint	ASML	ASML Holding NV - New York Registry Shares	purchase	$1,001$ — 15,000	kathy manning	NC06	clerk.h
13741	2022	2022-05-04	2022-04-29	joint	V	Visa Inc	sale_partial	$1,001$ — 15,000	kathy manning	NC06	clerk.h
9590	2022	2022-05-08	2022-05-04	joint	NaN	Sales Tax Securitization Corp 5% Due 1/1/2027	sale_full	$250,001-$ 500,000	suzan k. delbene	WA01	clerk.h

14273 rows × 13 columns

In [546…

```
# Second step of Party affiliation merging

party['Name'] = party['Name'].apply(lambda x: x.strip().split(' '))
merged_values =  pd.Series(merged.groupby('representative').count().index).apply(lambda x: x.split(' ')).values
for val in range(len(party['Name'])):
    for val2 in range(len(merged_values)):
        if len(list(set(party['Name'][val]).intersection(merged_values[val2]))) >= 2:
            replaced = merged.loc[merged['representative'] == ' '.join(merged_values[val2])]['Party'] \
                .fillna(party['Party'][val])
            merged.loc[merged['representative'] == ' '.join(merged_values[val2]), 'Party'] = replaced
names_fill = merged.loc[merged['Party'].isna()]['representative'].value_counts().index.values

merged.loc[merged['Party'].isna()]['representative'].value_counts()
```

Out[546…

```
gilbert cisneros          783
donna shalala             567
greg gianforte            497
rohit khanna              297
james r. langevin         220
kenny marchant            109
patrick fallon             92
thomas suozzi              75
william r. keating         72
francis rooney             60
michael garcia             45
richard w. allen           43
michael john gallagher     38
roger w. marshall          37
james e hon banks          27
james e. banks             25
harold dallas rogers       25
david p. roe               19
susan a. davis             18
susan w. brooks            14
george holding             13
daniel meuser              13
bradley s. schneider        9
harley e. rouda             7
linda t. sanchez            5
raúl m. grijalva            4
```

```
justin amash                3
bill flores                 3
wm. lacy clay               2
peter j. visclosky          2
joseph p. kennedy           2
j john (tj) cox             1
james hagedorn              1
robert e. latta             1
cott franklin               1
kenneth r. buck             1
nicholas v. taylor          1
james m. costa              1
Name: representative, dtype: int64
```

In [547…

```
# Last step of Party affiliation merging

fill_dict = ['Democratic', 'Democratic', 'Republican', 'Democratic', 'Democratic', 'Republican', 'Republican',
             'Democratic', 'Democratic', 'Republican', 'Republican', 'Republican', 'Democratic', 'Republican',
             'Republican', 'Republican', 'Republican', 'Republican', 'Democratic', 'Republican', 'Republican',
             'Republican']

for i in range(len(fill_dict)):
    merged.loc[merged['representative'] == names_fill[i], 'Party'] = fill_dict[i]

merged.loc[merged['Party'].isna()]['representative'].value_counts()
```

Out[547…

```
bradley s. schneider    9
harley e. rouda         7
linda t. sanchez        5
raúl m. grijalva        4
bill flores             3
justin amash            3
joseph p. kennedy       2
peter j. visclosky      2
wm. lacy clay           2
robert e. latta         1
james hagedorn          1
j john (tj) cox         1
cott franklin           1
kenneth r. buck         1
nicholas v. taylor      1
james m. costa          1
Name: representative, dtype: int64
```

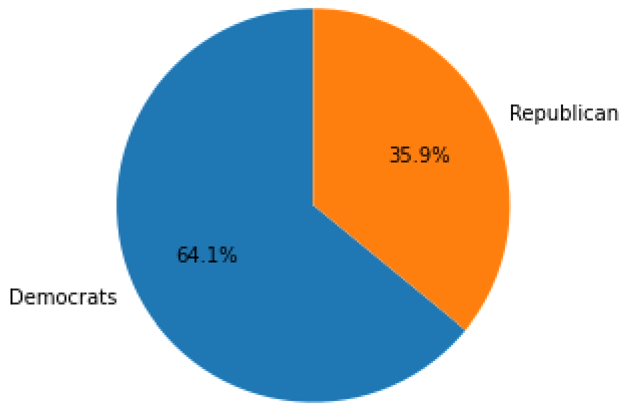
EDA

Univariate analyses

Parties proportion in transactions

In [548…

```
party_trades = merged['Party'].value_counts()
party_trades_d = party_trades[0]
party_trades_r = party_trades[1]

d_perc = party_trades_d / merged['Party'].value_counts().sum()
r_perc = party_trades_r / merged['Party'].value_counts().sum()

# Pie chart, where the slices will be ordered and plotted counter-clockwise:
labels = 'Democrats', 'Republican'
sizes = [d_perc, r_perc]

fig1, ax1 = plt.subplots()
ax1.pie(sizes, labels=labels, autopct='%1.1f%%', startangle=90)
ax1.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.

plt.show()
```



We see that Democrats make up about 64.1 percent of the transactions, while Republicans make up about 35.9 percent.

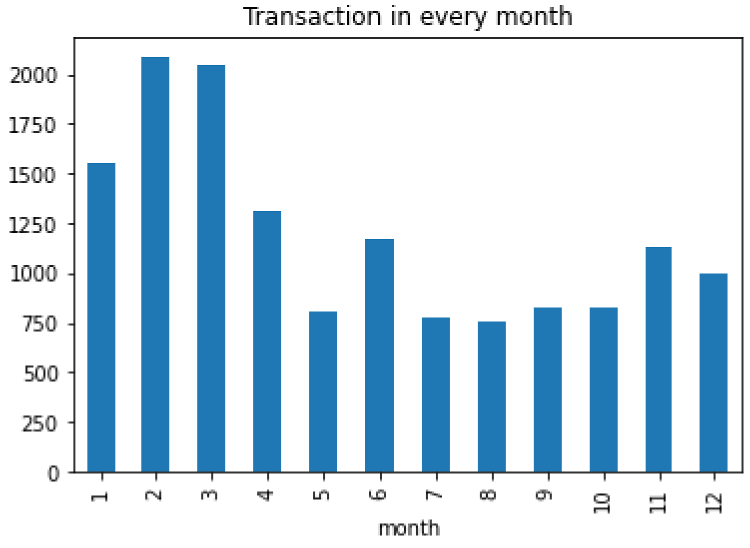
Common Month of transactions

In [549…

```
data_copy = data.copy()
data_copy['month'] = data_copy['transaction_date'].apply(lambda x:x.month)
data_copy.groupby('month').size().plot(kind='bar')
plt.title('Transaction in every month')
```

Out[549…

```
Text(0.5, 1.0, 'Transaction in every month')
```



From the chart, we can see the trends that many of the transactions happened in first six month of the year.

Bivariate analyses

Is there evidence of insider trading?

In [550…

```
merged['avg_amount'] = merged['amount'].apply(lambda x: (float(x.split('$')[1][:-3].replace(',','')) +
                                                          float(x.split('$')[-1].replace(',','')))/2)
merged.groupby(['transaction_date', 'type']).count()['avg_amount'].sort_values(ascending=False).head(10)
```

Out[550…

```
transaction_date  type
2019-06-24        sale_full    204
2020-03-18        purchase     204
2021-02-16        sale_full    157
2020-02-20        sale_full    115
2021-02-11        purchase     109
2020-04-02        purchase      77
2020-11-19        purchase      74
```

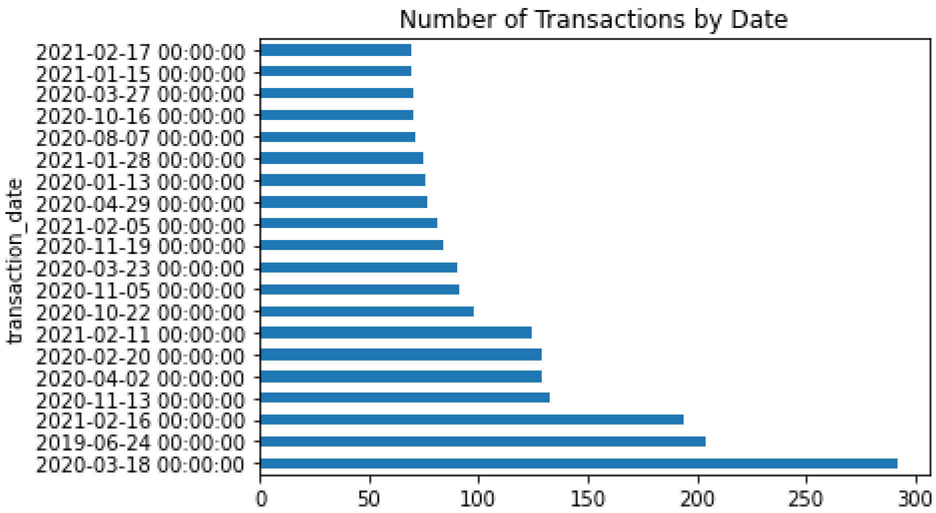
```
2020-03-23        sale_full     69
2020-11-13        purchase      66
2021-02-05        purchase      64
Name: avg_amount, dtype: int64
```

In [551…

```
merged.groupby('transaction_date').count()['avg_amount'].sort_values(ascending=False) \
    .head(20).plot(kind='barh', title='Number of Transactions by Date')
```

Out[551…

```
<AxesSubplot:title={'center':'Number of Transactions by Date'}, ylabel='transaction_date'>
```



We can see most of the transactions gathered around year 2020 and 2021. To dig in further, we recalled that 2020 was the year when Covid started, so we looked into early 2020 for some more insights.

In [552…

```
merged.loc[(merged['transaction_date'] == '2020-02-20')]['type'].value_counts()
```

Out[552…

```
sale_full       115
purchase          9
sale_partial      5
Name: type, dtype: int64
```

In [553…

```
merged.loc[(merged['transaction_date'] == '2020-03-18')]['type'].value_counts()
```

Out[553…

```
purchase        204
sale_full        64
```

```
sale_partial     24
Name: type, dtype: int64
```

On 2020-02-20, right before the index collapsed due to Covid, there are 113 full sale transactions. The date was right before the time of stock market crash. Therefore, we have the reason to believe there was insider trading where some congress people might be able to sale all their stocks before encountering significant losses since they got signal of market dump ahead.

Moreover, 2020-03-18 was the time when market reached its bottom and started to recover fastly. Even with the most professional stock market analysts, the chance people could predict the bottom that accurately is very small. The considerable amount of purchase on the date gave us more confidence that there was insider trading.

Analysis between parties and transactions amount

In [554…

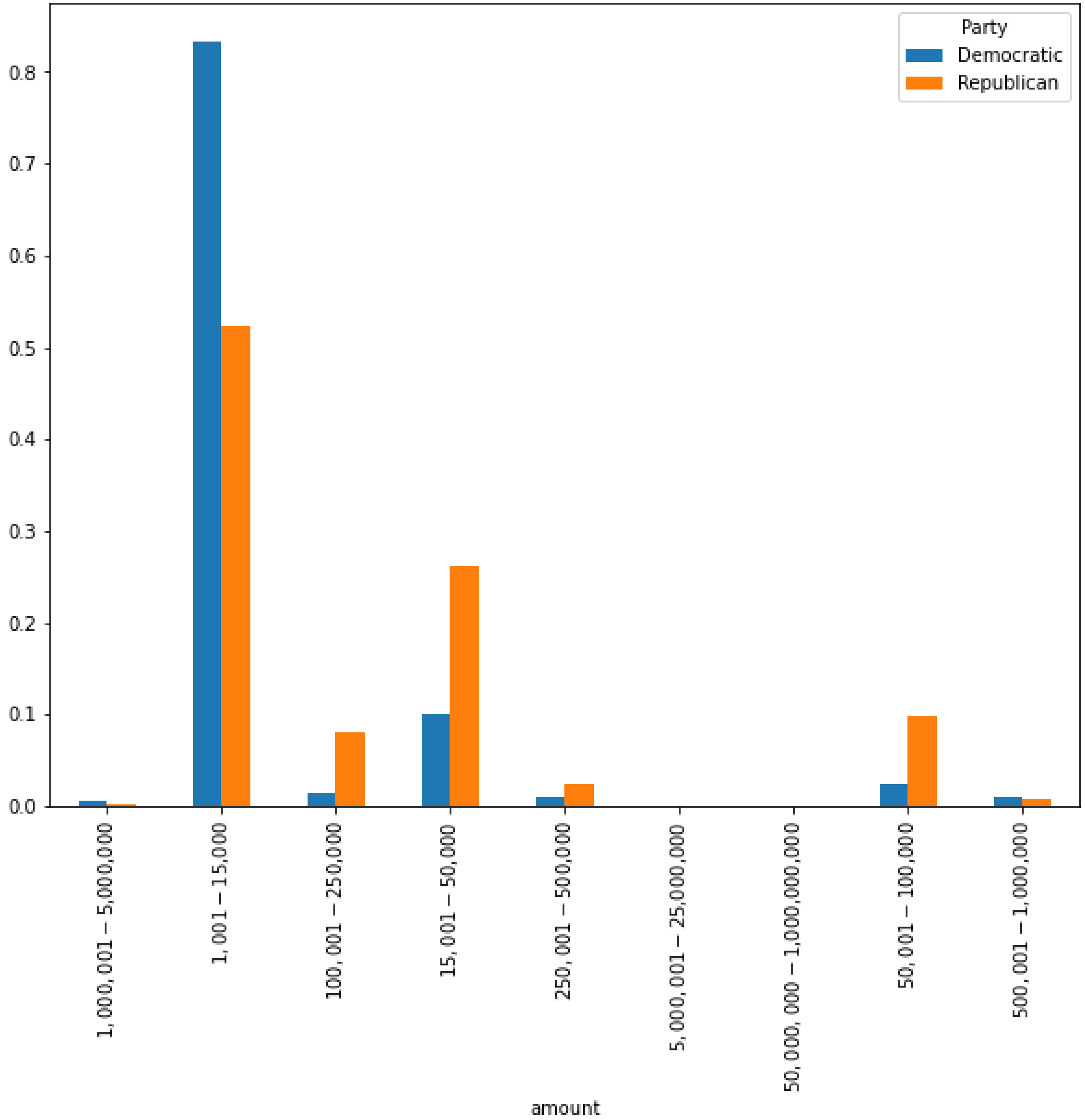
```
pivoted = (
    merged
        .pivot_table(index='amount', columns='Party', aggfunc='size')
        .apply(lambda x: x / x.sum())
)

pivoted
```

Out[554…

Party	Democratic	Republican
amount		
$1,000,001-$**5,000,000**	0.006362	0.001565
$1,001-$**15,000**	0.833608	0.523474
$100,001-$**250,000**	0.013711	0.080986
$15,001-$**50,000**	0.100801	0.261541
$250,001-$**500,000**	0.010639	0.024844
$5,000,001-$**25,000,000**	0.000877	0.000196
$50,000,000-$**1,000,000,000**	NaN	0.000196
$50,001-$**100,000**	0.023692	0.098396
$500,001-$**1,000,000**	0.010310	0.008803

In [555…
```
pivoted.plot.bar(stacked=False,figsize=(10,8))
```

Out[555… <AxesSubplot:xlabel='amount'>



From the pivot table and the bar chart, we discovered that Democrats usually make smaller transactions under 15,000 dollars. On the other hand, Republicans usually make larger transactions above 15,000 dollars.

Interesting Aggregations

In [556…

```
merged.groupby('Party')['transaction_date'].aggregate(['max','min'])
```

Out[556…

Party	max	min
Democratic	2022-05-04	2012-06-19
Republican	2022-04-29	2018-09-08

Through groupbying Parties and aggregating transactions date with min and max functions, we discovered that Democrats started investing way earlier than Republicans in 2012.

Assessment of Missingness

NMAR

We believe that the column 'ticker' is not missing at random. The first reason could be those tickers are not publicly traded or listed on the NASDAQ. Therefore, the information would not be disclosed. The other reason might be that there is insider trading. So if there is the possibiolity for one representative to manipulate specific ticker's price movement, we believe he/she would not show the detail of this transactions to the public.

MAR: Permutation test

In [557…

```
# function to calculate pivot table to compute tvd
def calculate_pivot_table(missing_col, other_col):
    df = (
        data
            .assign(isna=data[missing_col].isna())
            .pivot_table(index=other_col, columns='isna', aggfunc='size')
    )
    df = df / df.sum()
    return df
```

In [558…

```
# function to perform permutation test
```

```
def permutation(missing_col, other_col, N=500):
    df = calculate_pivot_table(missing_col, other_col)
    obs_tvd = df.diff(axis=1).iloc[:, -1].abs().sum() / 2

    shuffled = data.copy()
    shuffled['isna'] = shuffled[missing_col].isna()
    n_repetitions = N
    tvds = []
    for _ in range(n_repetitions):

        # Shuffling the column
        shuffled['shuffled'] = np.random.permutation(shuffled[other_col])

        # Computing and storing TVD
        pivoted = (
            shuffled
                .pivot_table(index='isna', columns='shuffled', aggfunc='size')
                .apply(lambda x: x / x.sum(), axis=1)
        )

        tvd = pivoted.diff().iloc[:, -1].abs().sum() / 2
        tvds.append(tvd)

    pval = np.mean(tvds >= obs_tvd)
    print("the p-value is {}".format(pval))
    pd.Series(tvds).plot(kind='hist', density=True, ec='w', bins=10,
                         title=f'p-value: {pval}', label='Simulated TVDs')
    plt.axvline(x=obs_tvd, color='red', linewidth=4, label='Observed TVD')
    plt.legend()
```

In [559…

```
merged.isnull().sum()
```

Out[559…

```
disclosure_year           0
disclosure_date           0
transaction_date          0
owner                  6669
ticker                 1147
asset_description         4
type                      0
amount                    0
representative            0
district                  0
```

```
ptr_link                     0
cap_gains_over_200_usd       0
Party                       44
avg_amount                   0
dtype: int64
```

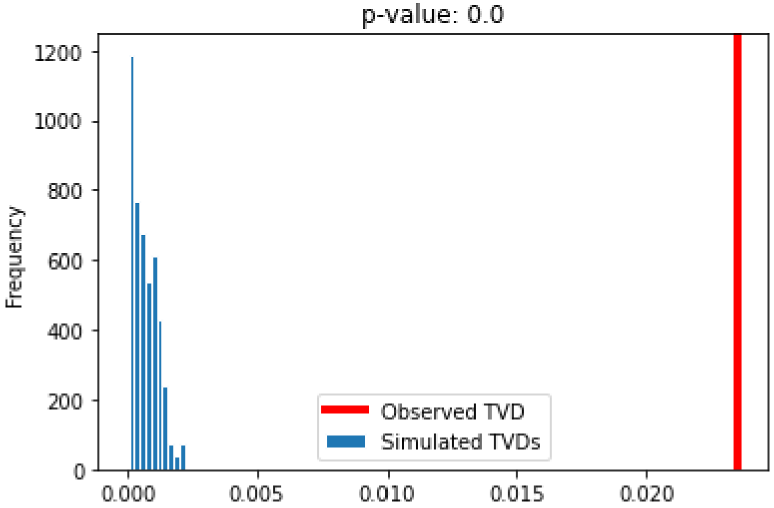
For the column 'owner', there are 6669 missing values. So we try to find the relationship between 'owner' and other columns to determine whether the missingness is Missing At Random or Missing Completely At Random. We will perform permutation tests on 'owner' and other columns.

First we try to see the relationship between owner and the trasaction amount. We hold the belief that the owner of transaction is missing might because the owner did not want others know they did large amount transactions.

In [560…

```
permutation('owner','amount')
```

```
the p-value is 0.0
```



In [561…

```
calculate_pivot_table('owner', 'type')
```

Out[561…

isna	False	True
type		
exchange	0.007628	0.010496

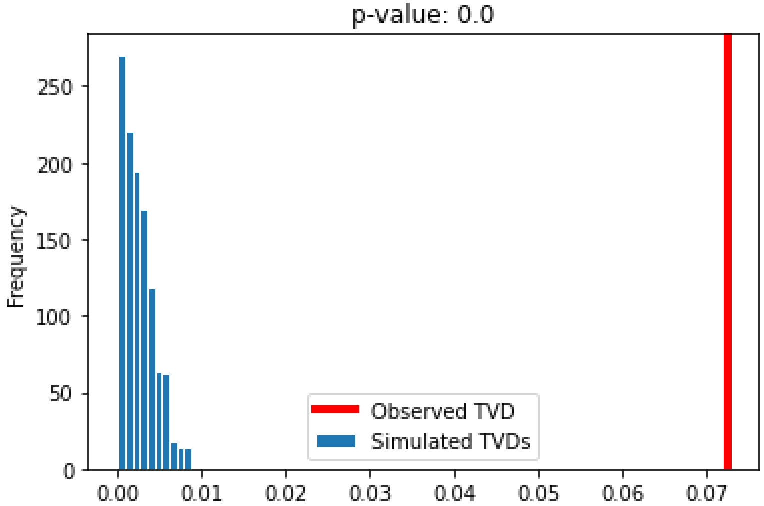
isna	False	True
type		
purchase	0.548527	0.488379
sale_full	0.287086	0.356725
sale_partial	0.156760	0.144399

Then we do permutation test on 'owner' and 'type' columns. From above chart, it seems more likely that when the type is 'purchase' or 'sale_full', the owner is likely to be missing. We will do permutation tests to check.

In [562…

```
permutation('owner','type')
```

```
the p-value is 0.0
```



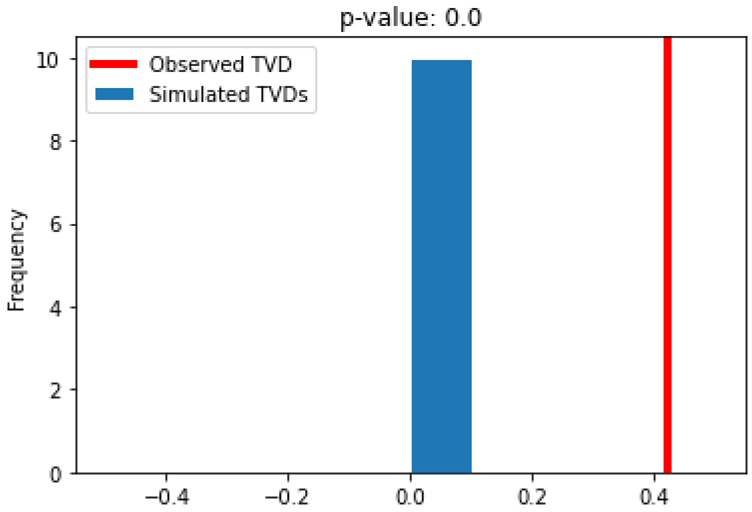
Since the p-value is 0, we can conclude that the missingenss of 'owner' is depend on 'type'

Last, we do permutation test on 'owner' and 'trasaction_date'. The value of owner is missing mighe be affected by the trasaction date. Maybe there are some insider tradings at some date so the owner are not willing to provide there names when trading.

In [563…

```
permutation('owner', 'transaction_date')
```

```
the p-value is 0.0
```



From the permutation test above, since the pvalue is 0, we can conclude that the missingness of column of owner is depent on the column of transaction date.

Hypothesis Testing

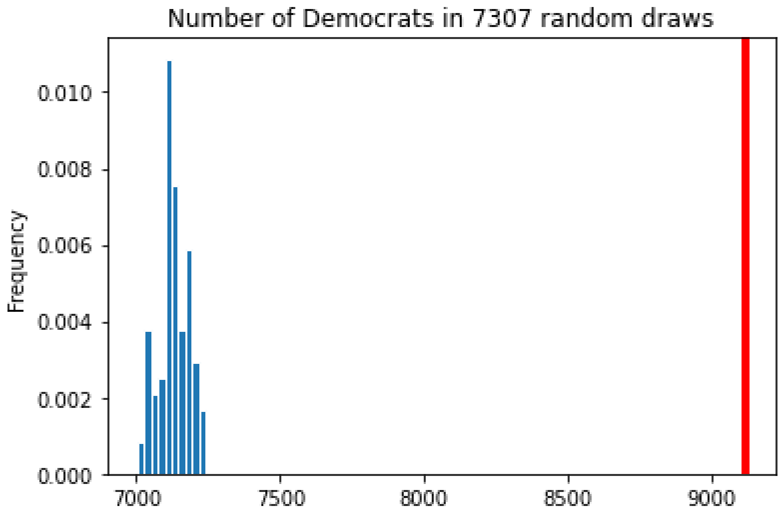
1. Does one party trade more often?

Null Hypothesis: Two parties trade equally often Alternative Hypothesis: One party trades more often than the other significance level: 0.05

In [564…

```
party_trades = merged['Party'].value_counts()
party_trades_d = party_trades[0]
party_trades_r = party_trades[1]

N = 100

simulation = pd.DataFrame(np.random.choice(['R', 'D'], p=[0.5, 0.5], size=(N, len(merged))))
results = (simulation == 'D').sum(axis=1)

pd.Series(results).plot(kind='hist',
                        density=True,
                        ec='w',
                        title='Number of Democrats in 7307 random draws')
plt.axvline(x=party_trades_d, color='red', linewidth=4)
p_value = (results > party_trades_d).mean()
p_value
```

Out[564… 0.0



Conclusion: Based on the p-value of 0.0, we reject the null hypothesis. Democrats trade more often than Republicans.

2. Does one party make larger trades?

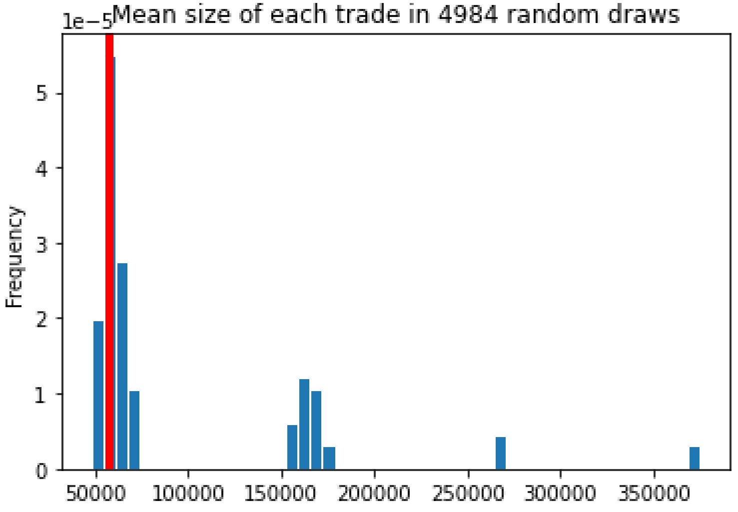
Null Hypothesis: Two parties make equally large trades Alternative Hypothesis: One party makes larger trades than the other significance level: 0.05

In [565…

```
# 4984 Democrats
avg_amount = merged.groupby('Party')['avg_amount'].mean()
d_amount = avg_amount[0]

N = 100

simulation = pd.DataFrame(np.random.choice(merged['avg_amount'], size=(N, 4984)))
results = simulation.mean(axis=1)

pd.Series(results).plot(kind='hist',
                        density=True,
                        bins=50,
                        ec='w',
                        title='Mean size of each trade in 4984 random draws')
plt.axvline(x=d_amount, color='red', linewidth=4)
```

```
p_value = (results < d_amount).mean()
p_value
```

Out [565... 0.3



Conclusion: Based on the p-value around 0.47, we fail to reject the Null. Two parties make equally large trades

3. Do the two parties invest in different stocks or sectors? For instance, do Democrats invest in Tesla more than Republicans?

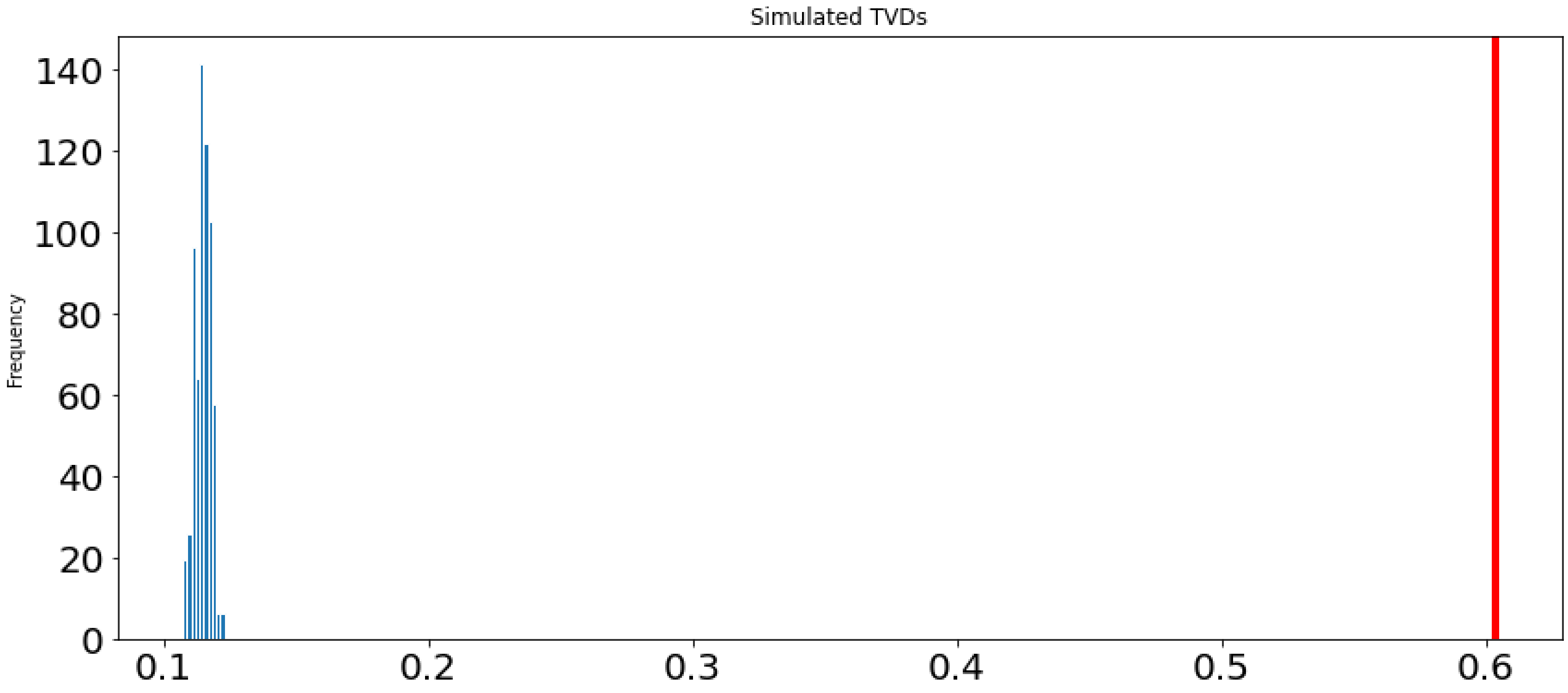
Null Hypothesis: Two parties invest in the same stocks or sectors Alternative Hypothesis: One party invests stocks or sectors different from the other significance level: 0.05

In [566...

```
party_stocks = merged.groupby('Party')['ticker'].value_counts(normalize=True)
party_stocks = party_stocks.unstack().T
party_stocks = party_stocks.fillna(0)
# party_stocks.plot(kind='barh', title='Proportion of Investments by Party', figsize=(30,30))

def total_variation_distance(dist1, dist2):
    return np.sum(np.abs(dist1 - dist2)) / 2

observed_distance = total_variation_distance(party_stocks['Democratic'], party_stocks['Republican'])

N = 100
```

```
ps_draw = np.random.multinomial(10000, party_stocks['Republican'], size=N) / 10000
tvd_draw = np.sum(np.abs(ps_draw - party_stocks['Republican'].to_numpy()), axis=1) / 2
pd.Series(tvd_draw).plot(kind='hist',
                    figsize=(14,6),
                     density=True,
                     ec='w',
                     title='Simulated TVDs',
                         fontsize=20)
plt.axvline(x=observed_distance, color='red', linewidth=4)
p_value = (tvd_draw > observed_distance).mean()
p_value
```

Out[566… 0.0



Conclusion: Based on the p-value of 0.0, we reject the Null. Two parties invest in different stocks or sectors