Zeru-Zhou-project4

February 8, 2022

1 Project 4 – Zeru Zhou

TA Help: NA

Collaboration: NA

- Got help from Dr. Ward's videos

1.1 Question 1

[7]:
```
import pandas as pd
```

[8]:
```
my_df = pd.read_csv("/depot/datamine/data/stackoverflow/unprocessed/2021.csv")
```

[3]:
```
from block_timer.timer import Timer
```

[10]:
```
with Timer(title="csv") as t1:
    my_df.to_csv("/scratch/brown/zhou902/2021.csv", index = False)
with Timer(title="parquet") as t2:
    my_df.to_parquet("/scratch/brown/zhou902/2021.parquet", index = False)
with Timer(title="feather") as t3:
    my_df.to_feather("/scratch/brown/zhou902/2021.feather")

print(t1.elapsed)
print(t2.elapsed)
print(t3.elapsed)
print(f'Parquet: {t2.elapsed/t1.elapsed:.1%}')
print(f'Feather: {t3.elapsed/t1.elapsed:.1%}')
```

```
[csv] Total time 2.25361 seconds.
[parquet] Total time 0.41709 seconds.

2.253605325706303
0.41709266416728497
0.33891418669372797
Parquet: 18.5%
Feather: 15.0%

[feather] Total time 0.33891 seconds.
```

[12]:
```
with Timer(title="csv") as t1:
    pd.read_csv("/scratch/brown/zhou902/2021.csv")
with Timer(title="parquet") as t2:
    pd.read_parquet("/scratch/brown/zhou902/2021.parquet")
with Timer(title="feather") as t3:
    pd.read_feather("/scratch/brown/zhou902/2021.feather")

print(t1.elapsed)
print(t2.elapsed)
print(t3.elapsed)
print(f'Parquet: {t2.elapsed/t1.elapsed:.1%}')
print(f'Feather: {t3.elapsed/t1.elapsed:.1%}')
```

```
[csv] Total time 0.95159 seconds.

0.9515923364087939
0.34658367838710546
0.17277581617236137
Parquet: 36.4%
Feather: 18.2%

[parquet] Total time 0.34658 seconds.
[feather] Total time 0.17278 seconds.
```

[13]:
```
from pathlib import Path
```

[16]:
```
print(f'csv: {Path("/scratch/brown/zhou902/2021.csv").stat().st_size/1000000}')
print(f'parquet: {Path("/scratch/brown/zhou902/2021.parquet").stat().st_size/
 ↪1000000}')
print(f'feather: {Path("/scratch/brown/zhou902/2021.feather").stat().st_size/
 ↪1000000}')
```

```
csv: 79.910042
parquet: 5.414069
feather: 25.78445
```

In writting, parquet is 18.5% of csv and feather is 15.0% of csv. In reading, parquet is 36.4% of csv and feather is 18.2% of csv. The sizes in MB are displayed above.

1.2 Question 2

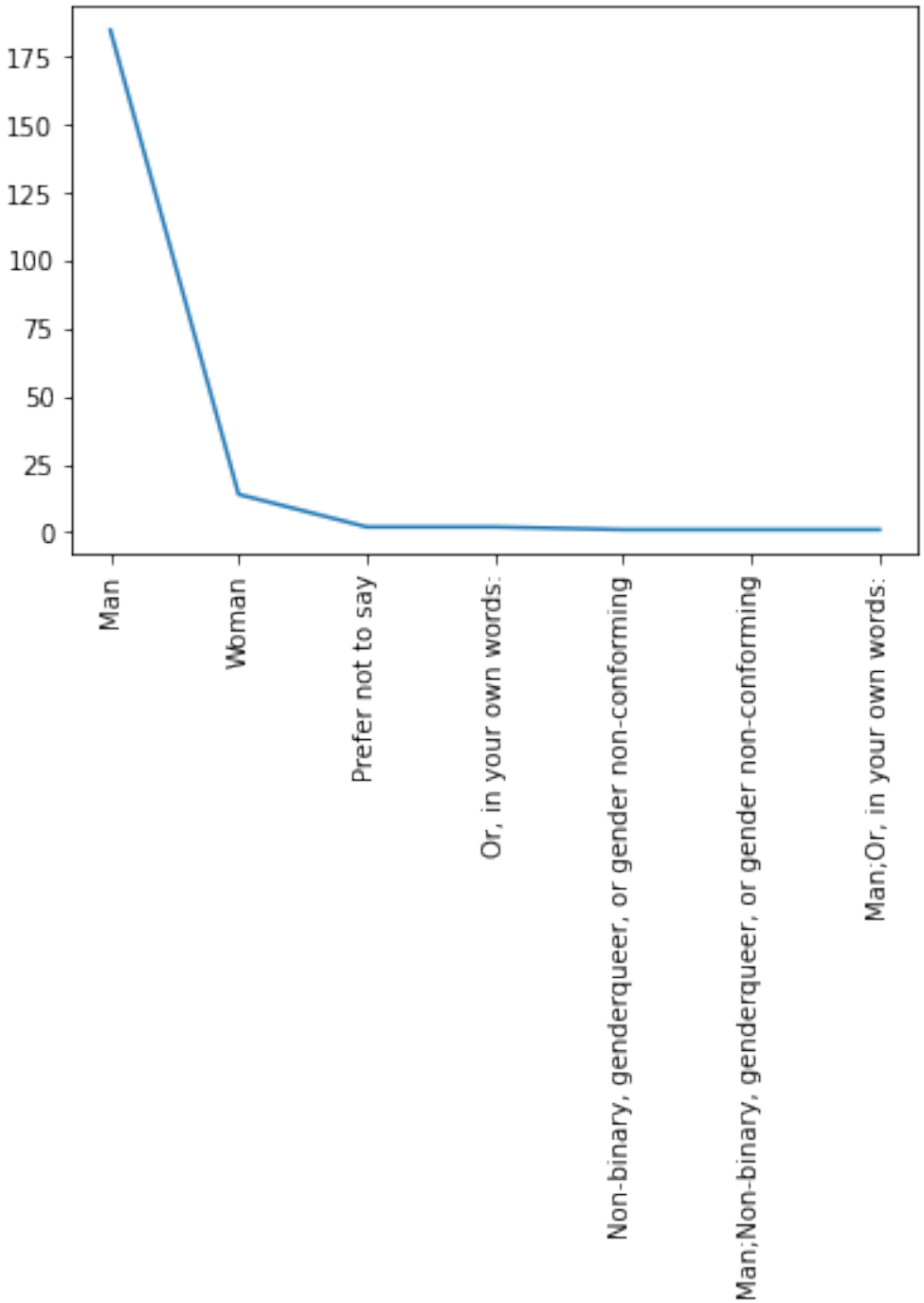
[31]:
```
my_df.loc[my_df['US_State']=="Indiana", "Gender"].value_counts()
```

[31]:
```
Man                                                    185
Woman                                                   14
Prefer not to say                                        2
Or, in your own words:                                   2
Non-binary, genderqueer, or gender non-conforming        1
Man;Non-binary, genderqueer, or gender non-conforming    1
```

```
Man;Or, in your own words:                                    1
Name: Gender, dtype: int64
```

[34]:
```
my_df.loc[my_df['US_State']=="Indiana", "Gender"].value_counts().plot(rot=90)
```

[34]: <AxesSubplot:>



Value_counts is applied and plot is made.

1.3 Question 3

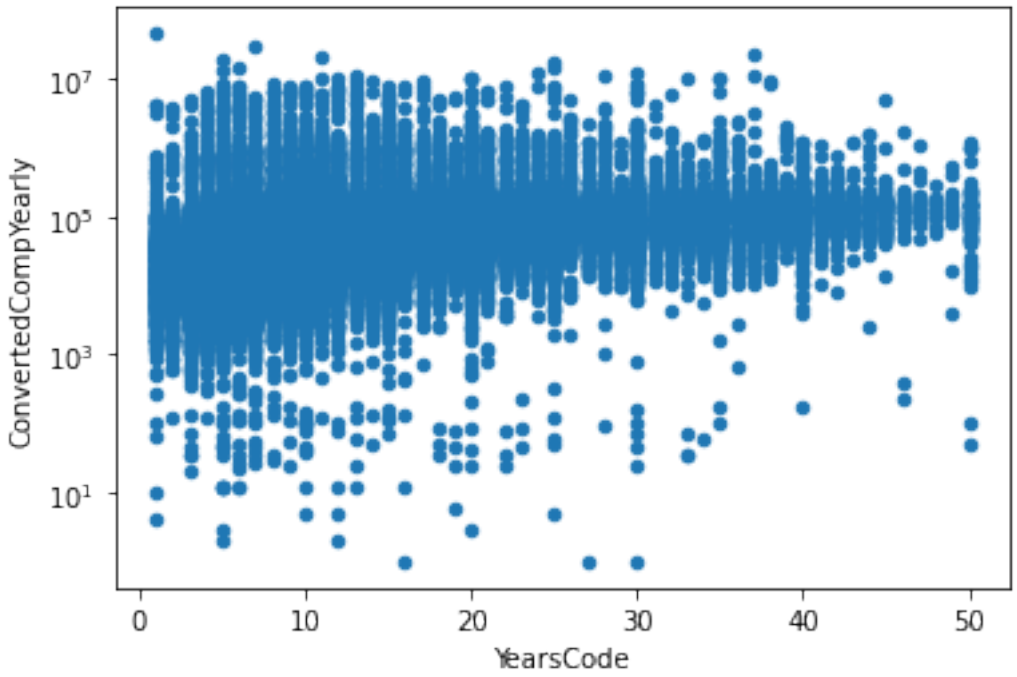
[3]:
```
my_df["ConvertedCompYearly"].unique()
```

[3]:
```
array([  62268.,      nan,   51552., …, 4300000.,  160500.,  816816.])
```

[4]:
```
my_df['YearsCode']=my_df['YearsCode'].astype('str')
my_df['YearsCode']=my_df['YearsCode'].replace("[^0-9]", "", regex = True)
my_df['YearsCode']=pd.to_numeric(my_df['YearsCode'])
```

[6]:
```
my_df.plot(x='YearsCode', y="ConvertedCompYearly", kind = 'scatter', logy =␣
↪True)
```

[6]:
```
<AxesSubplot:xlabel='YearsCode', ylabel='ConvertedCompYearly'>
```



As the YearsCode becomes larger, the range of “ConvertedCompYearly” becomes smaller.

1.4 Question 4

[11]:
```
my_df["LanguageHaveWorkedWith"].unique()
```

[11]:
```
array(['C++;HTML/CSS;JavaScript;Objective-C;PHP;Swift',
       'JavaScript;Python', 'Assembly;C;Python;R;Rust', …,
       'Java;JavaScript;Kotlin;Objective-C;TypeScript',
       'Clojure;Kotlin;SQL', 'Delphi;Elixir;HTML/CSS;Java;JavaScript'],
      dtype=object)
```

[13]:
```
my_df["LanguageHaveWorkedWith"] = my_df["LanguageHaveWorkedWith"].astype(str)
```

[14]:
```
def flatten (List):
    return (item for sublist in List for item in sublist)
```

[16]:
```
pd.Series(flatten(my_df["LanguageHaveWorkedWith"].str.split(";"))).
↪value_counts()
```

[16]:
```
JavaScript     53587
HTML/CSS       46259
Python         39792
SQL            38835
Java           29162
Node.js        27975
TypeScript     24909
C#             22984
Bash/Shell     22385
C++            20057
PHP            18130
C              17329
PowerShell      8871
Go              7879
Kotlin          6866
Rust            5799
Ruby            5569
Dart            4965
Assembly        4632
Swift           4204
R               4185
VBA             3847
Matlab          3846
Groovy          2479
Objective-C     2310
Scala           2148
Perl            2028
Haskell         1749
Delphi          1731
Clojure         1552
Elixir          1438
LISP            1096
nan             1082
```

```
Julia           1068
F#               804
Erlang           651
APL              536
Crystal          466
COBOL            437
dtype: int64
```

Times listed above. I worked with R, python, SQL, and matlab.

1.5 Question 5

[17]:
```
my_df.head()
```

[17]:
```
   ResponseId                                      MainBranch  \
0           1                   I am a developer by profession
1           2           I am a student who is learning to code
2           3  I am not primarily a developer, but I write co…
3           4                   I am a developer by profession
4           5                   I am a developer by profession

                                       Employment  \
0  Independent contractor, freelancer, or self-em…
1                              Student, full-time
2                              Student, full-time
3                              Employed full-time
4  Independent contractor, freelancer, or self-em…

                                         Country US_State UK_Country  \
0                                       Slovakia      NaN        NaN
1                                    Netherlands      NaN        NaN
2                              Russian Federation      NaN        NaN
3                                        Austria      NaN        NaN
4  United Kingdom of Great Britain and Northern I…      NaN    England

                                         EdLevel     Age1stCode  \
0  Secondary school (e.g. American high school, G…  18 - 24 years
1       Bachelor’s degree (B.A., B.S., B.Eng., etc.)  11 - 17 years
2       Bachelor’s degree (B.A., B.S., B.Eng., etc.)  11 - 17 years
3    Master’s degree (M.A., M.S., M.Eng., MBA, etc.)  11 - 17 years
4    Master’s degree (M.A., M.S., M.Eng., MBA, etc.)   5 - 10 years

                                      LearnCode YearsCode  …  \
0  Coding Bootcamp;Other online resources (ex: vi…       NaN  …
1  Other online resources (ex: videos, blogs, etc…         7  …
2  Other online resources (ex: videos, blogs, etc…       NaN  …
3                                           NaN       NaN  …
```

```
4                          Friend or family member        17  …

               Age Gender Trans                Sexuality  \
0  25-34 years old    Man    No  Straight / Heterosexual
1  18-24 years old    Man    No  Straight / Heterosexual
2  18-24 years old    Man    No        Prefer not to say
3  35-44 years old    Man    No  Straight / Heterosexual
4  25-34 years old    Man    No                      NaN

                      Ethnicity               Accessibility  \
0  White or of European descent            None of the above
1  White or of European descent            None of the above
2             Prefer not to say            None of the above
3  White or of European descent  I am deaf / hard of hearing
4  White or of European descent            None of the above

        MentalHealth           SurveyLength                   SurveyEase  \
0  None of the above  Appropriate in length                         Easy
1  None of the above  Appropriate in length                         Easy
2  None of the above  Appropriate in length                         Easy
3                NaN  Appropriate in length  Neither easy nor difficult
4                NaN  Appropriate in length                         Easy

  ConvertedCompYearly
0             62268.0
1                 NaN
2                 NaN
3                 NaN
4                 NaN

[5 rows x 48 columns]
```

[18]:
```
my_df["SurveyEase"].unique()
```

[18]:
```
array(['Easy', 'Neither easy nor difficult', nan, 'Difficult'],
      dtype=object)
```

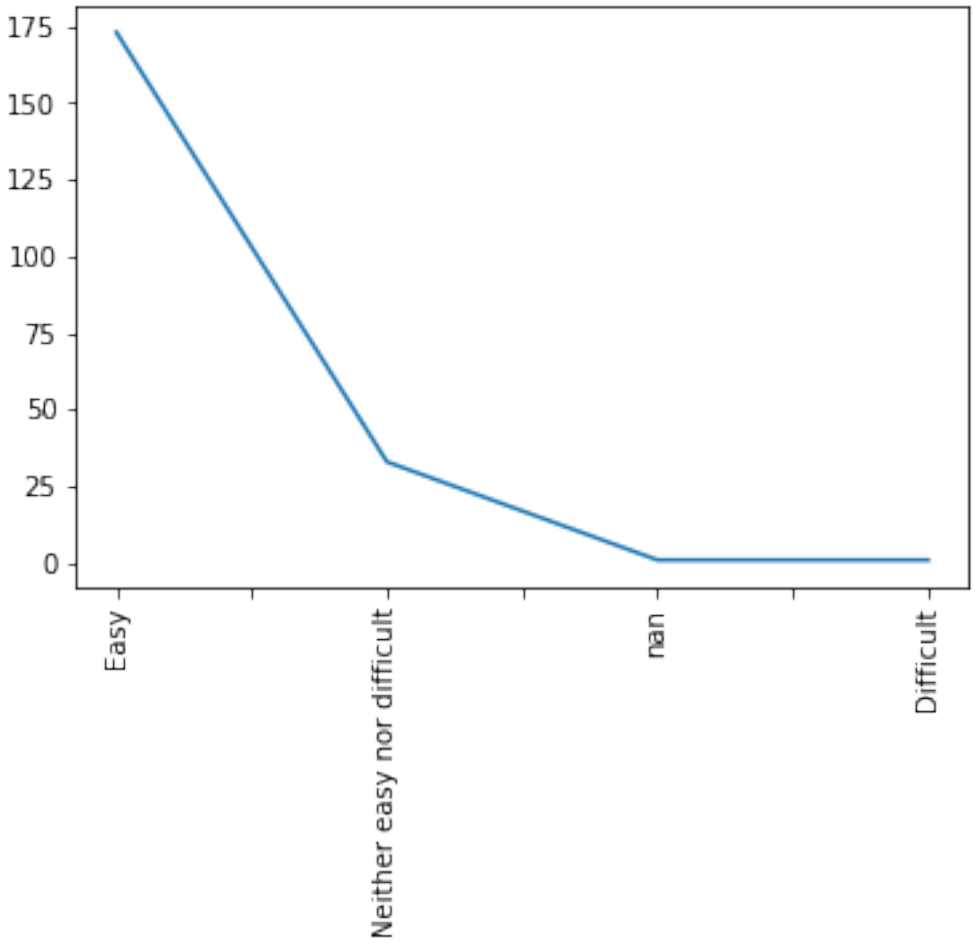
[19]:
```
my_df["SurveyEase"] = my_df["SurveyEase"].astype(str)
```

[22]:
```
my_df.loc[my_df["US_State"]=="Indiana", "SurveyEase"].value_counts()
```

[22]:
```
Easy                          173
Neither easy nor difficult     33
nan                             1
Difficult                       1
Name: SurveyEase, dtype: int64
```

[23]:
```
my_df.loc[my_df["US_State"]=="Indiana", "SurveyEase"].value_counts().plot(rot =␣
↪90)
```

[23]: <AxesSubplot:>



Here is how Indiana people reacted about the Survey Ease. Most people regard it as "Easy", according to the plot.

1.6 Pledge

By submitting this work I hereby pledge that this is my own, personal work. I've acknowledged in the designated place at the top of this file all sources that I used to complete said work, including but not limited to: online resources, books, and electronic communications. I've noted all collaboration with fellow students and/or TA's. I did not copy or plagiarize another's work.

> As a Boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together – We are Purdue.