tu11_re_PandasReview_HW

February 23, 2023

1 Pandas Review Homework

Import pandas

[1]:
```
import pandas as pd
```

1.1 1. Make a data frame from a Python dictionary.

Create a Python dictionary containing

- the names of four of your friends (real or imaginary)
- their ages
- the year they started college
- their majors

[2]:
```
dis_chars = {'Names': ['Christian', 'Noah', 'Aditya','Kim'],
             'Ages': [22, 22, 22,20],
             'Year': [2019 , 2019, 2019, 2021],
            'Majors': ['Biomedical Engineering',␣
 ↪'Psyhcology','Business','Choreography']}
```

Make a pandas data frame from your dictionary.

[3]:
```
list_name = pd.DataFrame(dis_chars)
```

Show your new data frame.

[4]:
```
list_name
```

[4]:

	Names	Ages	Year	Majors
0	Christian	22	2019	Biomedical Engineering
1	Noah	22	2019	Psyhcology
2	Aditya	22	2019	Business
3	Kim	20	2021	Choreography

Fetch the ages of all your friends.

[5]:
```
list_name['Ages']
```

[5]:
```
0    22
1    22
2    22
3    20
Name: Ages, dtype: int64
```

Fetch the name of your fourth friend.

[6]:
```
list_name['Ages'][3]
```

[6]:
```
20
```

Fetch the age of your third friend.

[7]:
```
list_name['Ages'][2]
```

[7]:
```
22
```

Compute and show the average age of your friends.

[8]:
```
list_name['Ages'].mean()
```

[8]:
```
21.5
```

1.2 2. Find a table of data on Wikipedia and import it.

Go to Widepedia and find a table of data. It can be anything you want.

In the cell below, import the data and display it (first and last five rows).

[26]:
```
atx_population = pd.read_clipboard()
```

[29]:
```
atx_population.head(5)
```

[29]:

	Census	Pop.	Note	%±
0	1850	629	NaN	-
1	1860	3,494	NaN	455.5%
2	1870	4,428	NaN	26.7%
3	1880	11,013	NaN	148.7%
4	1890	14,575	NaN	32.3%

[28]:
```
atx_population.tail(5)
```

[28]:

	Census	Pop.	Note	%±
14	1990	465,622	NaN	34.6%
15	2000	656,562	NaN	41.0%
16	2010	790,390	NaN	20.4%
17	2020	961,855	NaN	21.7%
18	2021 (est.)	964,177	NaN	0.2%

1.3 3. Load the RMS titanic data and export a subset of columns

Load the titanic data, make a new `DataFrame` of the fare paid and the survival columns, and export it as a `.csv` file.

[2]:
```
titanic = pd.read_csv("data/titanic.csv")
```

Import your new `.csv` file into a new `DataFrame` and show it (first and last five rows).

[4]:
```
titanic.head(5)
```

[4]:
```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked
0      0         A/5 21171   7.2500   NaN        S
1      0          PC 17599  71.2833   C85        C
2      0  STON/O2. 3101282   7.9250   NaN        S
3      0            113803  53.1000  C123        S
4      0            373450   8.0500   NaN        S
```

[72]:
```
titanic.tail(5)
```

[72]:
```
     PassengerId  Survived  Pclass                                      Name  \
886          887         0       2                     Montvila, Rev. Juozas
887          888         1       1              Graham, Miss. Margaret Edith
888          889         0       3  Johnston, Miss. Catherine Helen "Carrie"
889          890         1       1                     Behr, Mr. Karl Howell
890          891         0       3                       Dooley, Mr. Patrick

        Sex   Age  SibSp  Parch      Ticket   Fare Cabin Embarked
886    male  27.0      0      0      211536  13.00   NaN        S
887  female  19.0      0      0      112053  30.00   B42        S
888  female   NaN      1      2  W./C. 6607  23.45   NaN        S
889    male  26.0      0      0      111369  30.00  C148        C
890    male  32.0      0      0      370376   7.75   NaN        Q
```

1.4 4. Fetch specific rows of data of the titanic data

Fetch all the second class passengers of the titanic data and put them in a new `DataFrame` and show it.

[26]:
```
# Grabbing all Pclass
titanic_pclass = titanic[['PassengerId','Pclass']]
```

[33]:
```
# Grabbing only second class
titanic_pclass_2 = titanic_pclass[titanic_pclass['Pclass'] == 2]
titanic_pclass_2
```

[33]:
```
     PassengerId  Pclass
9             10       2
15            16       2
17            18       2
20            21       2
21            22       2
..           …     …
866          867       2
874          875       2
880          881       2
883          884       2
886          887       2

[184 rows x 2 columns]
```

Fetch all the first and third class passengers, put them in a new `DataFrame`, and show it.

[35]:
```
# Grabbing all classs except the second class
titanic_pclass_1_3 = titanic_pclass[titanic_pclass['Pclass'] != 2]
titanic_pclass_1_3
```

[35]:
```
     PassengerId  Pclass
0             1       3
1             2       1
2             3       3
3             4       1
4             5       3
..           …     …
885          886       3
887          888       1
888          889       3
889          890       1
890          891       3

[707 rows x 2 columns]
```

1.5 5. Plot some Titanic data

First, import `matplotlib`

[3]:
```
import matplotlib as plt
import matplotlib.pyplot as plt
```

1.5.1 5.a - Scatter plot

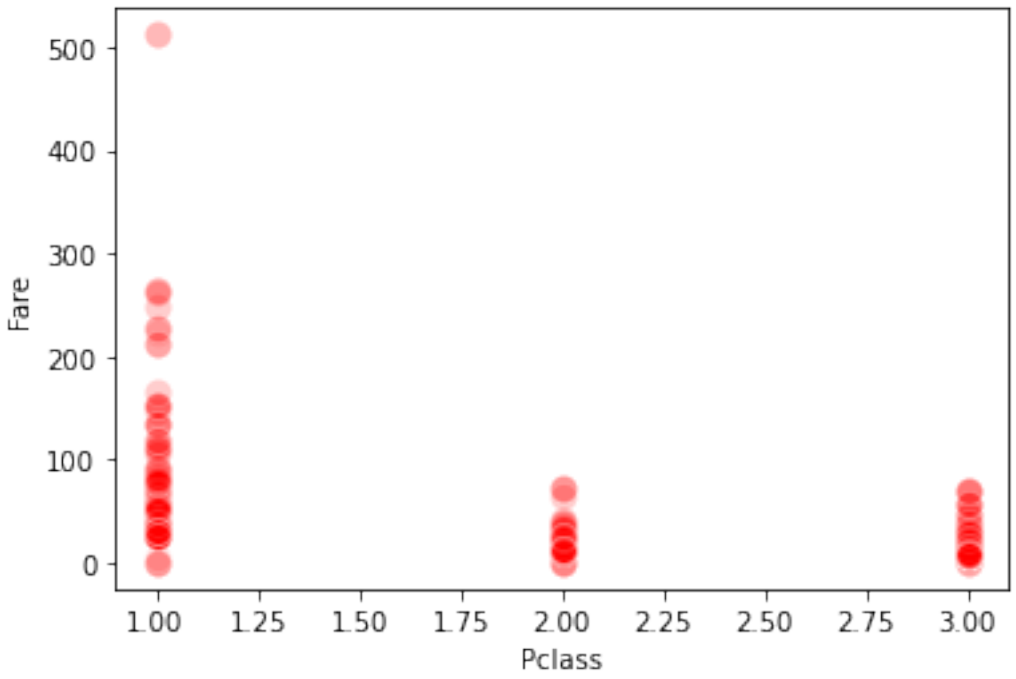
Make a scatter plot of fare vs. cabin class (seems like these should be perfectly related).

[37]:
```
titanic.head()
```

[37]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

[5]:
```
titanic.plot.scatter(x = 'Pclass', y = 'Fare', color = 'red', edgecolor =␣
↪'white', alpha = 0.1, s = 100);
```



1.5.2 5.b - Distribution plot (challenging!)

Plot the distributions of fare paid for survivors and deceased in a way that makes for a good visual comparison.

[4]:
```
# Create Str label for survive/decreased
titanic['Survived_Label'] = titanic['Survived'].
 ↪replace([0,1],['Decreased','Survived'])
titanic.head()
```

[4]:
```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked Survived_Label
```

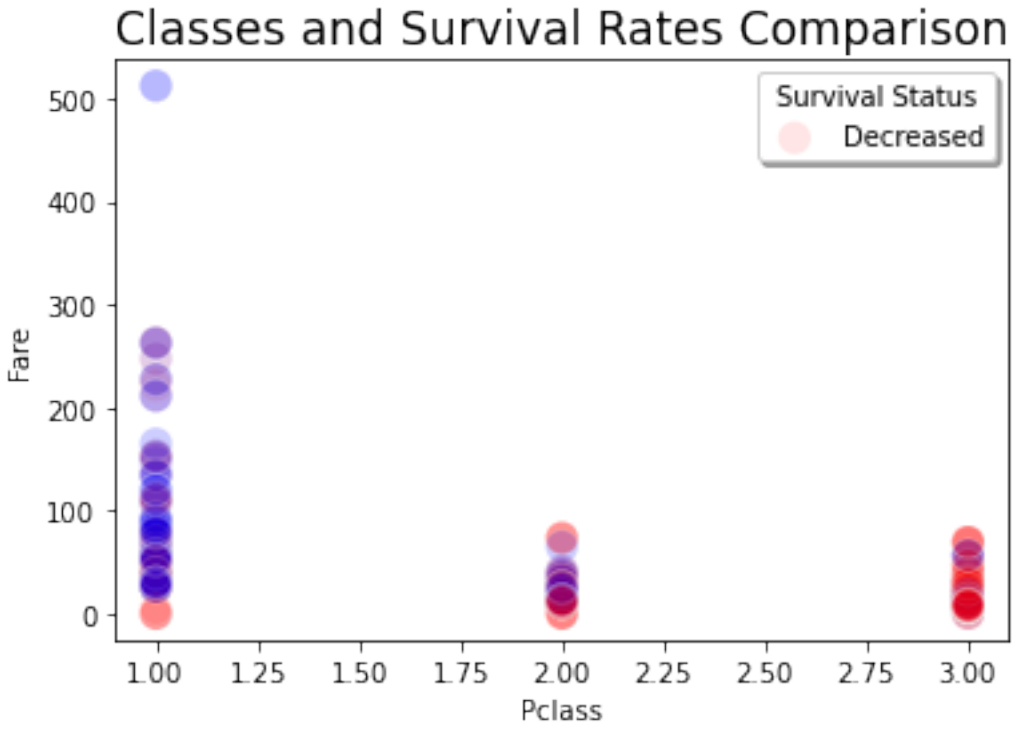
```
0      0         A/5 21171   7.2500   NaN        S      Decreased
1      0          PC 17599  71.2833   C85        C       Survived
2      0  STON/O2. 3101282   7.9250   NaN        S       Survived
3      0            113803  53.1000  C123        S       Survived
4      0            373450   8.0500   NaN        S      Decreased
```

[22]:
```
# Creating Color Map for Survived and Decreased
c_map = {'Survived': 'blue', 'Decreased': 'red'}

# Creating Graph
titanic.plot.scatter(x = 'Pclass', y = 'Fare',
                     color = titanic['Survived_Label'].map(c_map),
                     edgecolor = 'white',
                     alpha = 0.1,
                     s = 150);

plt.title('Classes and Survival Rates Comparison', fontsize = 17);

plt.legend(labels = titanic['Survived_Label'].tolist(), title = 'Survival␣
 ↪Status', fancybox=True, shadow=True);
```



^ There is an issue with the legend(). Thus, I will now created two scatter plot for each survival status, then combined it together. ^

[24]:
```
# DataFrame for `Decreased` Passenger
titanic_decreased = titanic[['Pclass','Survived_Label','Fare']]
titanic_decreased = titanic_decreased[titanic_decreased['Survived_Label'] ==␣
 ↪'Decreased']
titanic_decreased.head()
```

[24]:

	Pclass	Survived_Label	Fare
0	3	Decreased	7.2500
4	3	Decreased	8.0500
5	3	Decreased	8.4583
6	1	Decreased	51.8625
7	3	Decreased	21.0750

[25]:
```
# DataFrame for `Survived` Passenger
titanic_survived = titanic[['Pclass','Survived_Label','Fare']]
titanic_survived = titanic_survived[titanic_survived['Survived_Label'] ==␣
 ↪'Survived']
titanic_survived.head()
```

[25]:

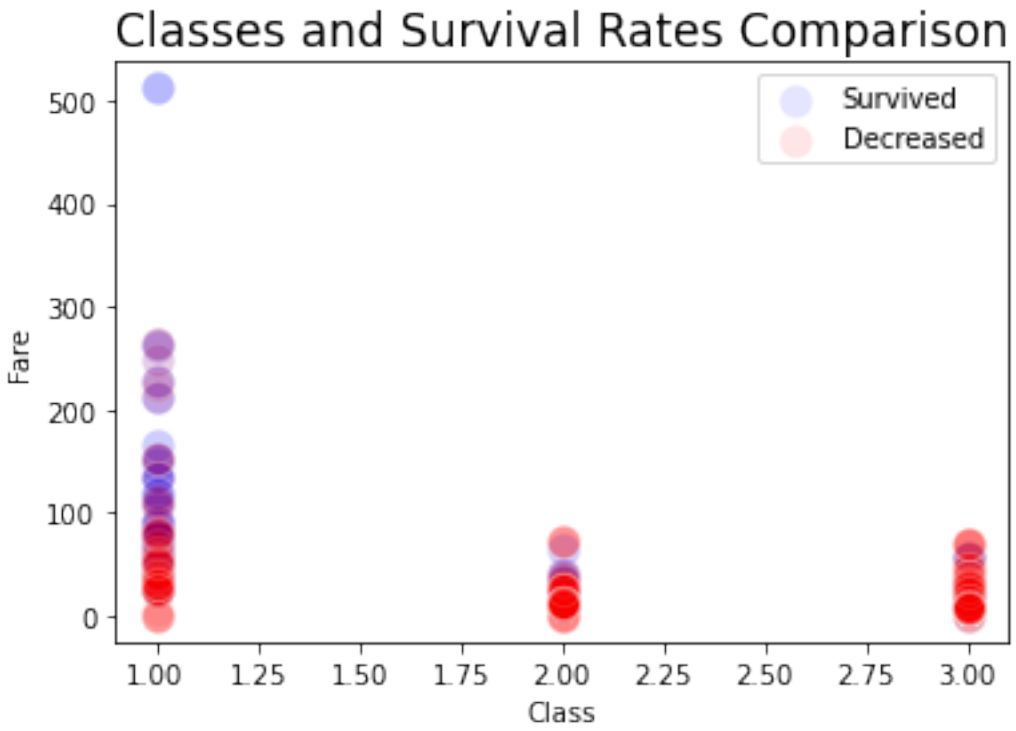
	Pclass	Survived_Label	Fare
1	1	Survived	71.2833
2	3	Survived	7.9250
3	1	Survived	53.1000
8	3	Survived	11.1333
9	2	Survived	30.0708

[40]:
```
# Scatter Plot for Survived Passenger
plt.scatter(x = titanic_survived['Pclass'], y = titanic_survived['Fare'], color␣
 ↪= 'blue',
           alpha = 0.1, edgecolor = 'white', label = 'Survived', s = 150);

# Scatter Plot for Decreased Passenger
plt.scatter(x = titanic_decreased['Pclass'], y = titanic_decreased['Fare'],␣
 ↪color = 'red',
           alpha = 0.1, edgecolor = 'white', label = 'Decreased', s = 150);

plt.title('Classes and Survival Rates Comparison', fontsize = 17)
plt.xlabel('Class')
plt.ylabel('Fare')
plt.legend()
```

[40]:
```
<matplotlib.legend.Legend at 0x7fa6846151f0>
```



1.6 6. Calculate new columns

1.6.1 6.a - Compute total number of relatives

Create a new column in your titanic `DataFrame` quantifying the total number of relatives on board (siblings + parents – the number of siblings are in `SibSp` and the number of parents are in `Parch`).

[31]:
```
titanic_sib_parent = titanic
titanic_sib_parent['Relatives'] = titanic_sib_parent['SibSp'] +␣
↪titanic_sib_parent['Parch']
```

[33]:
```
titanic_sib_parent.head()
```

[33]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0
3	4	1	1				
4	5	0	3				

```
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                            Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked Survived_Label  Relatives
0      0         A/5 21171   7.2500   NaN        S      Decreased          1
1      0          PC 17599  71.2833   C85        C       Survived          1
2      0  STON/O2. 3101282   7.9250   NaN        S       Survived          0
3      0            113803  53.1000  C123        S       Survived          1
4      0            373450   8.0500   NaN        S      Decreased          0
```

1.6.2 6.b - Did a person have any relatives on board?

Add another column – a Boolean column – indicating whether each person had any relatives on board.

[37]:
```
titanic_sib_parent['Relatives_Status'] = titanic_sib_parent['Relatives'] > 0
titanic_sib_parent.head()
```

[37]:
```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                            Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked Survived_Label  Relatives  \
0      0         A/5 21171   7.2500   NaN        S      Decreased          1
1      0          PC 17599  71.2833   C85        C       Survived          1
2      0  STON/O2. 3101282   7.9250   NaN        S       Survived          0
3      0            113803  53.1000  C123        S       Survived          1
4      0            373450   8.0500   NaN        S      Decreased          0

   Relatives_Status
0              True
1              True
2             False
3              True
4             False
```

1.7 7. Computing descriptive statistics

1.7.1 7.a - Compute a mean for a column

Compute the proportion of survivors of the RMS Titanic. **Hint**: the coding of `Survival` as 0 or 1 really works to our advantage here: the proportion of survivors in any group is easily computed using a common statistical function. The 7.a section header should also give you a big clue!

[41]:
```
# Mean of Survival Rates
titanic['Survived'].mean()
```

[41]:
```
0.3838383838383838
```

[42]:
```
# Mean of Survival Rate In Class
titanic[['Survived','Pclass']].groupby('Pclass').mean()
```

[42]:

Pclass	Survived
1	0.629630
2	0.472826
3	0.242363

1.7.2 7.a - Compute a mean for a subset of data

Compute the proportion of survivors for the females on the RMS Titanic (you can do this in one go, or two steps, using an intermediate object containing just the female data).

[57]:
```
# Get female list
titanic_female = titanic[['Survived','Sex']]
titanic_female = titanic_female[titanic_female['Sex'] == 'female']
titanic_female
```

[57]:

	Survived	Sex
1	1	female
2	1	female
3	1	female
8	1	female
9	1	female
..	…	…
880	1	female
882	0	female
885	0	female
887	1	female
888	0	female

[314 rows x 2 columns]

[58]:
```
titanic_female['Survived'].mean()
```

[58]: 0.7420382165605095

1.7.3 7.b - Compute statistics by group

Compute the proportion of female vs. male survivors of the RMS Titanic.

[68]:
```
# Mean of Survival Rate By Sex
titanic_mean = titanic[['Survived','Sex']].groupby('Sex').mean()
titanic_mean
```

[68]:
```
        Survived
Sex
female  0.742038
male    0.188908
```

Now compute the proportion of female vs. male survivors of the RMS Titanic, *along with the **s**tandard **e**rror of the **m**ean*. The **bold** type should give you a hint about the name of the method to compute the standard error. To do this, you'll need to combine the `groupby()` and `agg()` methods!

[69]:
```
# Sem of Survival Rate By Sex
titanic_sem = titanic[['Survived','Sex']].groupby('Sex').sem()
titanic_sem
```

[69]:
```
        Survived
Sex
female   0.02473
male     0.01631
```

[89]:
```
titanic_sex = titanic[['Survived','Sex']].groupby('Sex')
titanic_sex.agg(['mean','sem'])
```

[89]:
```
        Survived
            mean      sem
Sex
female  0.742038  0.02473
male    0.188908  0.01631
```

What does this tell you about gender roles when the RMS Titanic was sunk?

- `Women had a higher chance of survival than men.`

Compute the proportion of survivors by cabin class and their standard error.

[4]:
```
titanic_class = titanic[['Survived','Pclass']].groupby('Pclass')
titanic_class.agg(['mean','sem'])
```

[4]:
```
        Survived
            mean       sem
```

```
Pclass
1       0.629630  0.032934
2       0.472826  0.036906
3       0.242363  0.019358
```

What does this tell you about socio-economic status when the RMS Titanic was sunk?

- There were higher chance of surviving if passenger were wealthy. As the data suggested, there were approximately 60 percent of first class passenger surviving the incident. Meamwhile, there were only approximately 20 percent of third class passenger surviving the incident.