Investigate_a_Dataset

May 25, 2022

> **Tip**: Welcome to the Investigate a Dataset project! You will find tips in quoted sections like this to help organize your approach to your investigation. Once you complete this project, remove these **Tip** sections from your report before submission. First things first, you might want to double-click this Markdown cell and change the title so that it reflects your dataset and investigation.

1 Project: patient Data Analysis

1.1 Table of Contents

## Introduction

1.1.1 Dataset Description

> **Tip**: In this section of the report, provide a brief introduction to the dataset you've selected/downloaded for analysis. Read through the description available on the homepage-links present here. List all column names in each table, and their significance. In case of multiple tables, describe the relationship between tables.

1.1.2 Question(s) for Analysis

> **Tip**: Clearly state one or more questions that you plan on exploring over the course of the report. You will address these questions in the **data analysis** and **conclusion** sections. Try to build your report around the analysis of at least one dependent variable and three independent variables. If you're not sure what questions to ask, then make sure you familiarize yourself with the dataset, its variables and the dataset context for ideas of what to explore.

> **Tip**: Once you start coding, use NumPy arrays, Pandas Series, and DataFrames where appropriate rather than Python lists and dictionaries. Also, **use good coding practices**, such as, define and use functions to avoid repetitive code. Use appropriate comments within the code cells, explanation in the mark-down cells, and meaningful variable names.

```
In [1]: # Use this cell to set up import statements for all of the packages that you
        #   plan to use.
        import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
        % matplotlib inline
        # Remember to include a 'magic word' so that your visualizations are plotted
        #   inline with the notebook. See this page for more:
        #   http://ipython.readthedocs.io/en/stable/interactive/magics.html
```

```
In [ ]: # Upgrade pandas to use dataframe.explode() function.
        !pip install --upgrade pandas==0.25.0
```

```
Collecting pandas==0.25.0
  Downloading https://files.pythonhosted.org/packages/1d/9a/7eb9952f4b4d73fbd75ad1d5d6112f407e69
    100% || 10.5MB 3.4MB/s eta 0:00:01   11% |                            | 1.2MB 26.6MB/s eta 0
Requirement already satisfied, skipping upgrade: python-dateutil>=2.6.1 in /opt/conda/lib/python
Requirement already satisfied, skipping upgrade: pytz>=2017.2 in /opt/conda/lib/python3.6/site-p
Collecting numpy>=1.13.3 (from pandas==0.25.0)
  Downloading https://files.pythonhosted.org/packages/45/b2/6c7545bb7a38754d63048c7696804a0d9473
    100% || 13.4MB 2.7MB/s eta 0:00:01   37% |                    | 5.0MB 26.0MB/s eta 0:00:01
Requirement already satisfied, skipping upgrade: six>=1.5 in /opt/conda/lib/python3.6/site-packa
tensorflow 1.3.0 requires tensorflow-tensorboard<0.2.0,>=0.1.0, which is not installed.
Installing collected packages: numpy, pandas
  Found existing installation: numpy 1.12.1
    Uninstalling numpy-1.12.1:
      Successfully uninstalled numpy-1.12.1
```

## Data Wrangling

Tip: In this section of the report, you will load in the data, check for cleanliness, and then trim and clean your dataset for analysis. Make sure that you **document your data cleaning steps in mark-down cells precisely and justify your cleaning decisions.**

1.1.3 General Properties

Tip: You should *not* perform too many operations in each cell. Create cells freely to explore your data. One option that you can take with this project is to do a lot of explorations in an initial notebook. These don't have to be organized, but make sure you use enough comments to understand the purpose of each code cell. Then, after you're done with your analysis, create a duplicate notebook where you will trim the excess and organize your steps so that you have a flowing, cohesive report.

```
In [2]: # Load your data and print out a few lines. Perform operations to inspect data
        #   types and look for instances of missing or possibly errant data.
        df = pd.read_csv("noshowappointments-kagglev2-may-2016.csv")
        df.head()
```

```
Out[2]:       PatientId  AppointmentID Gender          ScheduledDay  \
        0  2.987250e+13        5642903      F  2016-04-29T18:38:08Z
        1  5.589978e+14        5642503      M  2016-04-29T16:08:27Z
        2  4.262962e+12        5642549      F  2016-04-29T16:19:04Z
        3  8.679512e+11        5642828      F  2016-04-29T17:29:31Z
        4  8.841186e+12        5642494      F  2016-04-29T16:07:23Z

                 AppointmentDay  Age      Neighbourhood  Scholarship  Hipertension  \
        0  2016-04-29T00:00:00Z   62    JARDIM DA PENHA            0             1
        1  2016-04-29T00:00:00Z   56    JARDIM DA PENHA            0             0
        2  2016-04-29T00:00:00Z   62      MATA DA PRAIA            0             0
        3  2016-04-29T00:00:00Z    8  PONTAL DE CAMBURI            0             0
        4  2016-04-29T00:00:00Z   56    JARDIM DA PENHA            0             1

           Diabetes  Alcoholism  Handcap  SMS_received No-show
        0         0           0        0             0      No
        1         0           0        0             0      No
        2         0           0        0             0      No
        3         0           0        0             0      No
        4         1           0        0             0      No
```

```
In [25]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 110527 entries, 0 to 110526
Data columns (total 14 columns):
PatientId         110527 non-null float64
AppointmentID     110527 non-null int64
Gender            110527 non-null object
ScheduledDay      110527 non-null object
AppointmentDay    110527 non-null object
Age               110527 non-null int64
Neighbourhood     110527 non-null object
Scholarship       110527 non-null int64
Hipertension      110527 non-null int64
Diabetes          110527 non-null int64
Alcoholism        110527 non-null int64
Handcap           110527 non-null int64
SMS_received      110527 non-null int64
No-show           110527 non-null object
dtypes: float64(1), int64(8), object(5)
memory usage: 11.8+ MB
```

```
In [12]: df.duplicated()
```

```
Out[12]: 0          False
         1          False
         2          False
```

```
3         False
4         False
          ...
110522    False
110523    False
110524    False
110525    False
110526    False
Length: 110527, dtype: bool
```

```
In [6]: df.drop(['AppointmentDay','ScheduledDay'],axis=1 , inplace=True)
```

```
In [7]: df.head()
```

```
Out[7]:       PatientId  AppointmentID Gender  Age      Neighbourhood  Scholarship  \
        0  2.987250e+13        5642903      F   62    JARDIM DA PENHA            0
        1  5.589978e+14        5642503      M   56    JARDIM DA PENHA            0
        2  4.262962e+12        5642549      F   62      MATA DA PRAIA            0
        3  8.679512e+11        5642828      F    8  PONTAL DE CAMBURI            0
        4  8.841186e+12        5642494      F   56    JARDIM DA PENHA            0

           Hipertension  Diabetes  Alcoholism  Handcap  SMS_received No-show
        0             1         0           0        0             0      No
        1             0         0           0        0             0      No
        2             0         0           0        0             0      No
        3             0         0           0        0             0      No
        4             1         1           0        0             0      No
```

```
In [8]: df.drop(['AppointmentID','PatientId'],axis=1 , inplace=True)
```

```
In [9]: df.head()
```

```
Out[9]:   Gender  Age      Neighbourhood  Scholarship  Hipertension  Diabetes  \
        0      F   62    JARDIM DA PENHA            0             1         0
        1      M   56    JARDIM DA PENHA            0             0         0
        2      F   62      MATA DA PRAIA            0             0         0
        3      F    8  PONTAL DE CAMBURI            0             0         0
        4      F   56    JARDIM DA PENHA            0             1         1

           Alcoholism  Handcap  SMS_received No-show
        0           0        0             0      No
        1           0        0             0      No
        2           0        0             0      No
        3           0        0             0      No
        4           0        0             0      No
```

```
In [10]: df.describe()
```

```
Out[10]:                  Age    Scholarship   Hipertension       Diabetes  \
         count  110527.000000  110527.000000  110527.000000  110527.000000
         mean       37.088874      0.098266      0.197246      0.071865
         std        23.110205      0.297675      0.397921      0.258265
         min        -1.000000      0.000000      0.000000      0.000000
         25%        18.000000      0.000000      0.000000      0.000000
         50%        37.000000      0.000000      0.000000      0.000000
         75%        55.000000      0.000000      0.000000      0.000000
         max       115.000000      1.000000      1.000000      1.000000

                   Alcoholism        Handcap   SMS_received
         count  110527.000000  110527.000000  110527.000000
         mean        0.030400       0.022248       0.321026
         std         0.171686       0.161543       0.466873
         min         0.000000       0.000000       0.000000
         25%         0.000000       0.000000       0.000000
         50%         0.000000       0.000000       0.000000
         75%         0.000000       0.000000       1.000000
         max         1.000000       4.000000       1.000000
```

```
In [9]: age=df.query('Age=="-1"')
        age
```

```
Out[9]:           PatientId  AppointmentID Gender          ScheduledDay  \
        99832  4.659432e+14        5775010      F  2016-06-06T08:58:13Z

                     AppointmentDay  Age Neighbourhood  Scholarship  Hipertension   \
        99832  2016-06-06T00:00:00Z   -1         ROMÃO            0             0

                     Diabetes  Alcoholism  Handcap  SMS_received No-show
        99832         0           0        0             0      No
```

1.1.4 Data Cleaning

> **Tip**: Make sure that you keep your reader informed on the steps that you are taking in your investigation. Follow every code cell, or every set of related code cells, with a markdown cell to describe to the reader what was found in the preceding cell(s). Try to make it so that the reader can then understand what they will be seeing in the following cell(s).

```
In [10]: df.drop(index=99832, inplace=True)
```

```
In [13]: df.head()
```

```
Out[13]:       PatientId  AppointmentID Gender          ScheduledDay  \
         0  2.987250e+13        5642903      F  2016-04-29T18:38:08Z
         1  5.589978e+14        5642503      M  2016-04-29T16:08:27Z
         2  4.262962e+12        5642549      F  2016-04-29T16:19:04Z
         3  8.679512e+11        5642828      F  2016-04-29T17:29:31Z
         4  8.841186e+12        5642494      F  2016-04-29T16:07:23Z
         AppointmentDay  Age      Neighbourhood  Scholarship  Hipertension  \
0  2016-04-29T00:00:00Z   62    JARDIM DA PENHA            0             1
1  2016-04-29T00:00:00Z   56    JARDIM DA PENHA            0             0
2  2016-04-29T00:00:00Z   62      MATA DA PRAIA            0             0
3  2016-04-29T00:00:00Z    8  PONTAL DE CAMBURI            0             0
4  2016-04-29T00:00:00Z   56    JARDIM DA PENHA            0             1

   Diabetes  Alcoholism  Handcap  SMS_received No-show
0         0           0        0             0      No
1         0           0        0             0      No
2         0           0        0             0      No
3         0           0        0             0      No
4         1           0        0             0      No
```

```
In [23]: df['No-show'].value_counts()
```

```
Out[23]: No     88208
         Yes    22319
         Name: No-show, dtype: int64
```

```
In [13]: df.rename(columns={'No-show':'Noshow'},inplace=True)
```

```
In [14]: df.head()
```

```
Out[14]:       PatientId  AppointmentID Gender          ScheduledDay  \
         0  2.987250e+13        5642903      F  2016-04-29T18:38:08Z
         1  5.589978e+14        5642503      M  2016-04-29T16:08:27Z
         2  4.262962e+12        5642549      F  2016-04-29T16:19:04Z
         3  8.679512e+11        5642828      F  2016-04-29T17:29:31Z
         4  8.841186e+12        5642494      F  2016-04-29T16:07:23Z

                  AppointmentDay  Age      Neighbourhood  Scholarship  Hipertension  \
         0  2016-04-29T00:00:00Z   62    JARDIM DA PENHA            0             1
         1  2016-04-29T00:00:00Z   56    JARDIM DA PENHA            0             0
         2  2016-04-29T00:00:00Z   62      MATA DA PRAIA            0             0
         3  2016-04-29T00:00:00Z    8  PONTAL DE CAMBURI            0             0
         4  2016-04-29T00:00:00Z   56    JARDIM DA PENHA            0             1

            Diabetes  Alcoholism  Handcap  SMS_received Noshow
         0         0           0        0             0     No
         1         0           0        0             0     No
         2         0           0        0             0     No
         3         0           0        0             0     No
         4         1           0        0             0     No
```

```
In [ ]:
```

## Exploratory Data Analysis

Tip: Now that you've trimmed and cleaned your data, you're ready to move on to exploration. **Compute statistics** and **create visualizations** with the goal of addressing the research questions that you posed in the Introduction section. You should compute the relevant statistics throughout the analysis when an inference is made about the data. Note that at least two or more kinds of plots should be created as part of the exploration, and you must compare and show trends in the varied visualizations.

Tip: - Investigate the stated question(s) from multiple angles. It is recommended that you be systematic with your approach. Look at one variable at a time, and then follow it up by looking at relationships between variables. You should explore at least three variables in relation to the primary question. This can be an exploratory relationship between three variables of interest, or looking at how two independent variables relate to a single dependent variable of interest. Lastly, you should perform both single-variable (1d) and multiple-variable (2d) explorations.

1.1.5 Research Question 1 (Replace this header name!)

```
In [18]: # Use this, and more code cells, to explore your data. Don't forget to add
         #   Markdown cells to document your observations and findings.
         plt.hist('Age')
         plt.hist('Noshow')
         plt.xlabel('Age'
         plt.ylabel('patients')
         plt.title('age effect')
         plt.lagend()
         plt.show()
```

```
        ---------------------------------------------------------------------------

        TypeError                                 Traceback (most recent call last)

        <ipython-input-18-558dc8e9bbce> in <module>()
          1 # Use this, and more code cells, to explore your data. Don't forget to add
          2 #   Markdown cells to document your observations and findings.
    ----> 3 plt.hist['Age']
          4 plt.hist['Noshow']
          5 plt.xlabel('Age')


        TypeError: 'function' object is not subscriptable
```

1.1.6 Research Question 2 (Replace this header name!)

```
In [12]: # Continue to explore the data to address your additional research
         #   questions. Add more headers as needed if you have more questions to
         #   investigate.
         df['Scholarship'].value_counts()
```

```
Out[12]: 0    99666
         1    10861
         Name: Scholarship, dtype: int64
```

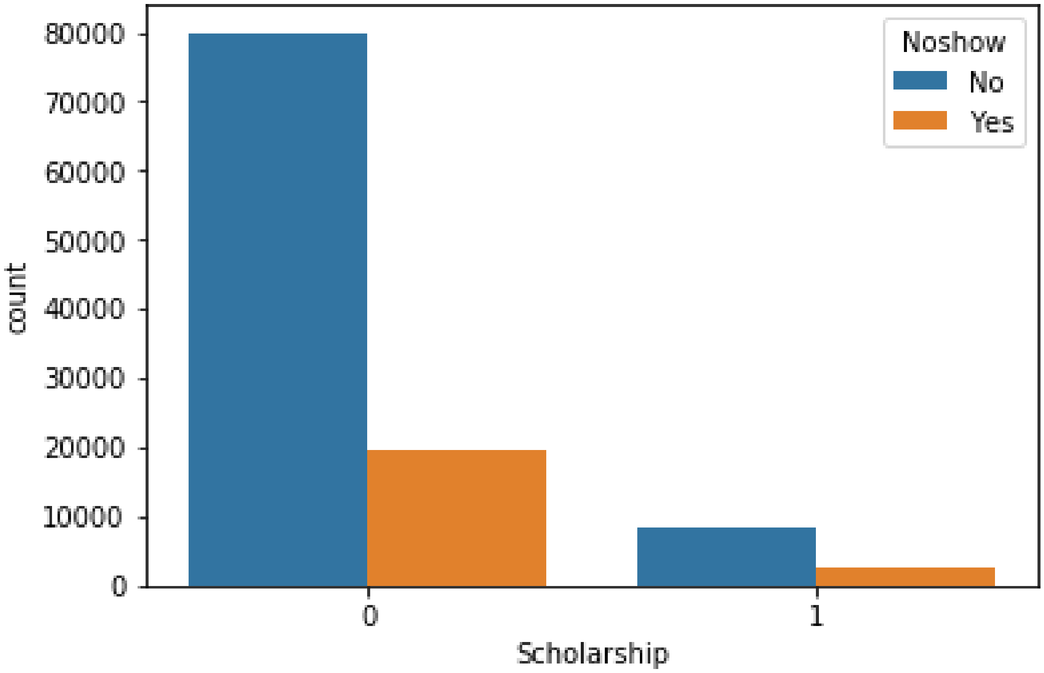
```
In [27]: x = sns.countplot(x=df.Scholarship , hue = df.Noshow,data=df)
         ax.set_tital('Show/Noshow for Scholarship')
         x_ticks_labels=['NoScholarship','Scholarship']
         ax.set_xticklabels(x_ticks_labals)
         plt.show()
```

```
        ---------------------------------------------------------------------------

        AttributeError                            Traceback (most recent call last)

        <ipython-input-27-3bc4e3bca2b4> in <module>()
          1 x = sns.countplot(x=df.Scholarship , hue = df.Noshow,data=df)
    ----> 2 ax.set_tital('Show/Noshow for Scholarship')
          3 x_ticks_labels=['NoScholarship','Scholarship']
          4 ax.set_xticklabels(x_ticks_labals)
          5 plt.show()


        AttributeError: 'AxesSubplot' object has no attribute 'set_tital'
```



```
In [3]: df['Gender'].value_counts()
```

```
Out[3]: F    71840
        M    38687
        Name: Gender, dtype: int64
```

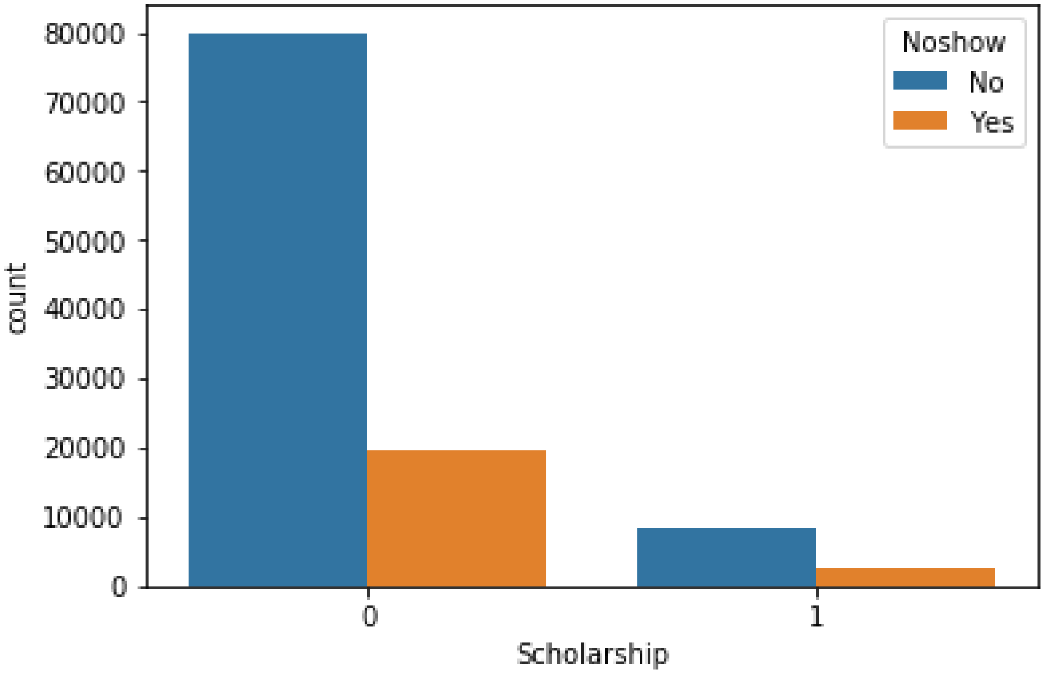
```
In [7]: x = sns.countplot(x=df.Scholarship , hue = df.Noshow,data=df)
        ax.set_tital('Show/Noshow for Female and Males ')
        x_ticks_labels=['Male','Female']
        ax.set_xticklabels(x_ticks_labals)
        plt.show()
```

```
        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-7-18121ce414bd> in <module>()
          1 x = sns.countplot(x=df.Scholarship , hue = df.Noshow,data=df)
    ----> 2 ax.set_tital('Show/Noshow for Female and Males ')
          3 x_ticks_labels=['Male','Female']
          4 ax.set_xticklabels(x_ticks_labals)
          5 plt.show()


        NameError: name 'ax' is not defined
```



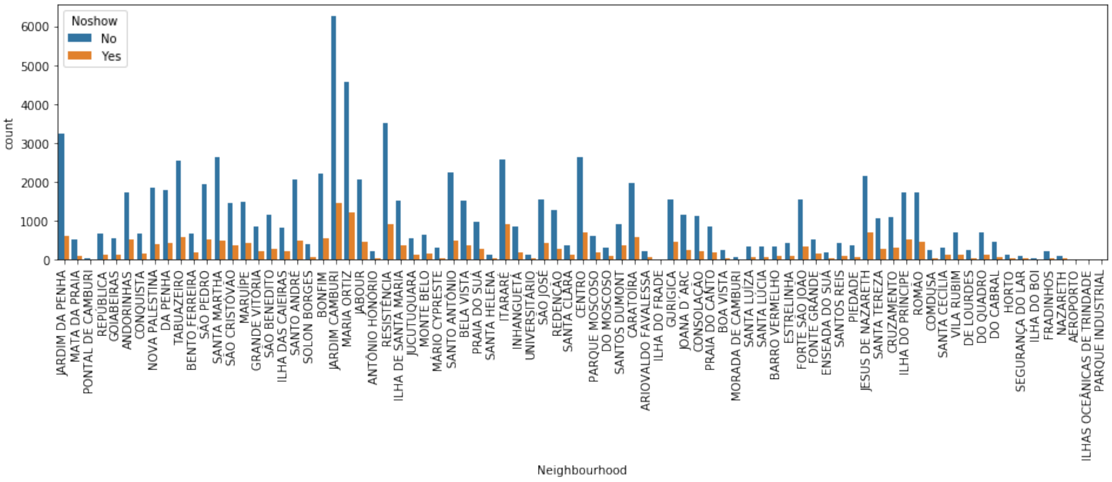
```
In [8]: plt.figure(figsize=(16,4))
        plt.xticks(rotation=90)
        ax=sns.countplot(x=df.Neighbourhood,hue=df.Noshow)
        ax_set_title("Show/Noshow by Neighbourhood")
        plt.show()
```

```
        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-8-73c6bd504702> in <module>()
          2 plt.xticks(rotation=90)
          3 ax=sns.countplot(x=df.Neighbourhood,hue=df.Noshow)
    ----> 4 ax_set_title("Show/Noshow by Neighbourhood")
          5 plt.show()


        NameError: name 'ax_set_title' is not defined
```



```
In [11]: df['Handcap'].value_counts()

Out[11]: 0    108286
         1      2042
         2       183
         3        13
         4         3
         Name: Handcap, dtype: int64
```

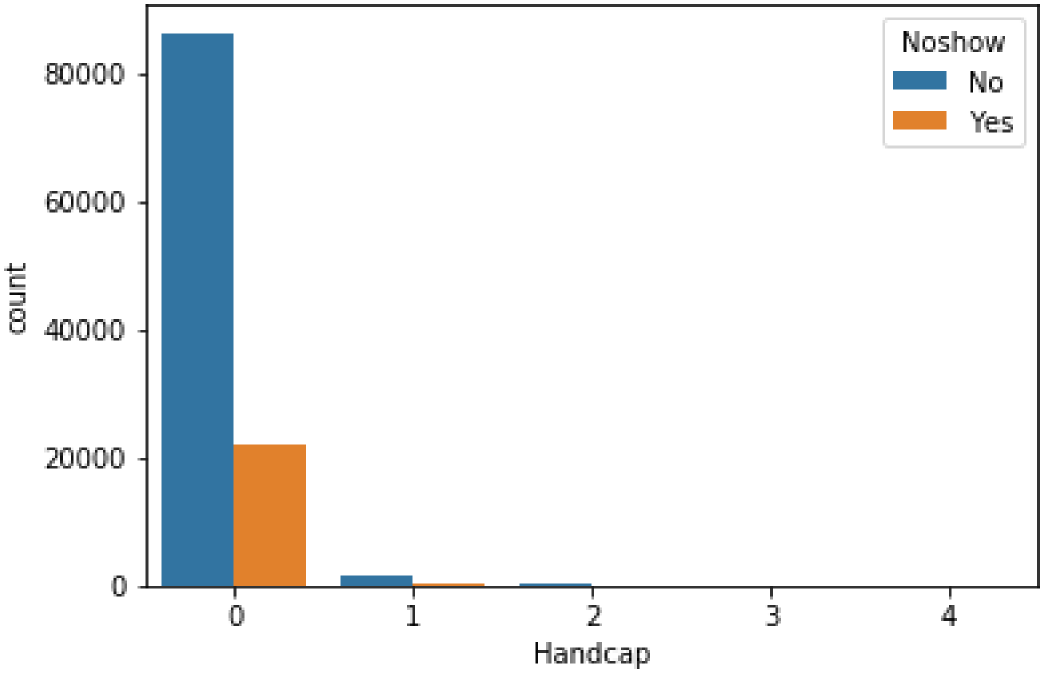
```
In [12]: x = sns.countplot(x=df.Handcap , hue = df.Noshow,data=df)
         ax_set_title("Show/Noshow by Handcap")
```

```
        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-12-7acf60d9da2c> in <module>()
          1 x = sns.countplot(x=df.Handcap , hue = df.Noshow,data=df)
    ----> 2 ax_set_title("Show/Noshow by Handcap")


        NameError: name 'ax_set_title' is not defined
```



```
In [13]: df['SMS_received'].value_counts()
```

```
Out[13]: 0    75045
         1    35482
         Name: SMS_received, dtype: int64
```

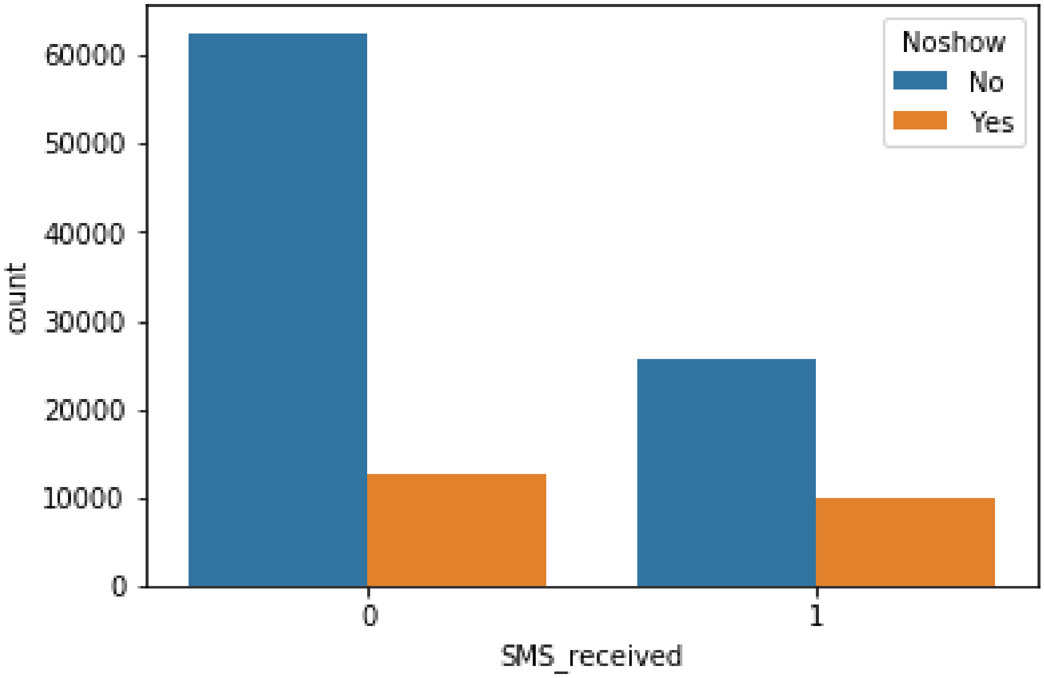
```
In [14]: x = sns.countplot(x=df.SMS_received , hue = df.Noshow,data=df)
         ax.set_tital('Show/Noshow for SMS_received ')
         x_ticks_labels=['NoSMS_received','SMS_received']
         ax.set_xticklabels(x_ticks_labals)
         plt.show()
```

```
    ---------------------------------------------------------------------------

    AttributeError                            Traceback (most recent call last)

    <ipython-input-14-6e708b758d9b> in <module>()
      1 x = sns.countplot(x=df.SMS_received , hue = df.Noshow,data=df)
----> 2 ax.set_tital('Show/Noshow for SMS_received ')
      3 x_ticks_labels=['NoSMS_received','SMS_received']
      4 ax.set_xticklabels(x_ticks_labals)
      5 plt.show()


    AttributeError: 'AxesSubplot' object has no attribute 'set_tital'
```



## Conclusions

Tip: Finally, summarize your findings and the results that have been performed in relation to the question(s) provided at the beginning of the analysis. Summarize the results accurately, and point out where additional research can be done or where additional information could be useful.

Tip: If you haven't done any statistical tests, do not imply any statistical conclusions. And make sure you avoid implying causation from correlation!

1.1.7 Limitations

Tip: Make sure that you are clear with regards to the limitations of your exploration. You should have at least 1 limitation explained clearly.

Tip: Once you are satisfied with your work here, check over your report to make sure that it is satisfies all the areas of the rubric (found on the project submission page at the end of the lesson). You should also probably remove all of the "Tips" like this one so that the presentation is as polished as possible.

1.2 Submitting your Project

Tip: Before you submit your project, you need to create a .html or .pdf version of this notebook in the workspace here. To do that, run the code cell below. If it worked correctly, you should get a return code of 0, and you should see the generated .html file in the workspace directory (click on the orange Jupyter icon in the upper left).

Tip: Alternatively, you can download this report as .html via the **File** > **Download as** submenu, and then manually upload it into the workspace directory by clicking on the orange Jupyter icon in the upper left, then using the Upload button.

Tip: Once you've done this, you can submit your project by clicking on the "Submit Project" button in the lower right here. This will create and submit a zip file with this .ipynb doc and the .html or .pdf version you created. Congratulations!

```
In [ ]: from subprocess import call
        call(['python', '-m', 'nbconvert', 'Investigate_a_Dataset.ipynb'])
```