Investigate_a_Dataset

April 10, 2023

1 Project: Investigate a Dataset - [No_Show Appointment]

1.1 Table of Contents

1.1.1 Dataset Description

This dataset contains information from 100k medical appointments in Brazil and is focused on the question of whether or not patients show up for their appointment. It contains 14 columns (variables). Below is a brief description of the columns (variables) and their significance.

1. PatientID - This column represents the identification number of a patient.
2. AppintmentID - This column represents the patient appointment identification number.
3. Gender - This column represents the sex of the patient (i.e Male or Female).
4. ScheduledDay - This column represents the day the patient registered for an appointment with the doctor.
5. AppointmentDay - This column represents the actual day the patient visited the doctor.
6. Age - This column represents the age of the patient or how old the patient is.
7. Neigbourhood - This column represents where the appointment took place.
8. Scholarship - this column gives more information about the patient. It states if the patient's medical bills were or were not sponsored by the Bolsa Famila
9. Hypertension - This column specifies the type of disease the patient has.
10. Diabetes - This column specifies the type of disease the patient has.
11. Alcoholism - This column specifies the type of disease the patient has.
12. Handicap - This column specifies the type of disease the patient has.
13. SMS_received - This column tells us if an sms message was sent to the patient
14. No_show - This column tells us of the attitude of the patient i.e if the patient showed up for an appointment or not.

The dependable variable is the No_show column

The independent variables are: PatientID, AppintmentID, Gender, ScheduledDay, AppointmentDay, Age, Scholarship, Neigbourhood, Hypertension, Diabetes, Alcoholism, Handicap and SMS_received

1.1.2 Question(s) for Analysis

I am concerned with the factors that can help predict if a patient will show up for their appointment or not. To this end I consider the following:

1. Will the patient show-up if they did or did not recieve SMS message?
2. If the medical bill is sponsored will the patient show-up for appointment?

1.2 Importing packages required to analyse dataset

```
In [1]: #Importing necessary packages required to analyse this dataset
        import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        %matplotlib inline
        import seaborn as sns
```

```
In [11]: # Upgrade pandas to use dataframe.explode() function.
         !pip install --upgrade pandas==1.1.5
```

```
Requirement already up-to-date: pandas==1.1.5 in /opt/conda/lib/python3.6/site-packages (1.1.5)
Requirement already satisfied, skipping upgrade: python-dateutil>=2.7.3 in /opt/conda/lib/python
Requirement already satisfied, skipping upgrade: numpy>=1.15.4 in /opt/conda/lib/python3.6/site-
Requirement already satisfied, skipping upgrade: pytz>=2017.2 in /opt/conda/lib/python3.6/site-p
Requirement already satisfied, skipping upgrade: six>=1.5 in /opt/conda/lib/python3.6/site-packa
```

2

2.1 Data Wrangling

2.2 Loading data into Pandas DataFrame from the CSV file

```
In [2]: # Load your data and print out first 10 rows of the dataset
        df = pd.read_csv("Database_No_show_appointments/noshowappointments-kagglev2-may-2016.csv
```

```
In [3]: df.head(10)
```

Out[3]:

	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	
5	9.598513e+13	5626772	F	2016-04-27T08:36:51Z	
6	7.336882e+14	5630279	F	2016-04-27T15:05:12Z	
7	3.449833e+12	5630575	F	2016-04-27T15:39:58Z	
8	5.639473e+13	5638447	F	2016-04-29T08:02:16Z	
9	7.812456e+13	5629123	F	2016-04-27T12:48:25Z	

```
          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
5  2016-04-29T00:00:00Z   76          REPÚBLICA            0             1
6  2016-04-29T00:00:00Z   23         GOIABEIRAS            0             0
7  2016-04-29T00:00:00Z   39         GOIABEIRAS            0             0
8  2016-04-29T00:00:00Z   21         ANDORINHAS            0             0
9  2016-04-29T00:00:00Z   19          CONQUISTA            0             0

   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
5         0           0        0             0      No
6         0           0        0             0     Yes
7         0           0        0             0     Yes
8         0           0        0             0      No
9         0           0        0             0      No
```

```
In [4]: #printing last 10 rows of the dataset
        df.tail(10)
```

```
Out[4]:            PatientId  AppointmentID Gender          ScheduledDay  \
110517  5.574942e+12        5780122      F  2016-06-07T07:38:34Z
110518  7.263315e+13        5630375      F  2016-04-27T15:15:06Z
110519  6.542388e+13        5630447      F  2016-04-27T15:23:14Z
110520  9.969977e+14        5650534      F  2016-05-03T07:51:47Z
110521  3.635534e+13        5651072      F  2016-05-03T08:23:40Z
110522  2.572134e+12        5651768      F  2016-05-03T09:15:35Z
110523  3.596266e+12        5650093      F  2016-05-03T07:27:33Z
110524  1.557663e+13        5630692      F  2016-04-27T16:03:52Z
110525  9.213493e+13        5630323      F  2016-04-27T15:09:23Z
110526  3.775115e+14        5629448      F  2016-04-27T13:30:56Z

                AppointmentDay  Age Neighbourhood  Scholarship  Hipertension  \
110517  2016-06-07T00:00:00Z   19   MARIA ORTIZ            0             0
110518  2016-06-07T00:00:00Z   50   MARIA ORTIZ            0             0
110519  2016-06-07T00:00:00Z   22   MARIA ORTIZ            0             0
110520  2016-06-07T00:00:00Z   42   MARIA ORTIZ            0             0
110521  2016-06-07T00:00:00Z   53   MARIA ORTIZ            0             0
110522  2016-06-07T00:00:00Z   56   MARIA ORTIZ            0             0
110523  2016-06-07T00:00:00Z   51   MARIA ORTIZ            0             0
```

```
110524  2016-06-07T00:00:00Z   21   MARIA ORTIZ            0             0
110525  2016-06-07T00:00:00Z   38   MARIA ORTIZ            0             0
110526  2016-06-07T00:00:00Z   54   MARIA ORTIZ            0             0

        Diabetes  Alcoholism  Handcap  SMS_received No-show
110517         0           0        0             0      No
110518         0           0        0             1      No
110519         0           0        0             1      No
110520         0           0        0             1      No
110521         0           0        0             1      No
110522         0           0        0             1      No
110523         0           0        0             1      No
110524         0           0        0             1      No
110525         0           0        0             1      No
110526         0           0        0             1      No
```

2.2.1 Data Inspection

```
In [5]: # To display tupule of the dimensions of the dataset
        df.shape

Out[5]: (110527, 14)

In [6]: # print the datatypes of the columns
        df.dtypes

Out[6]: PatientId         float64
        AppointmentID       int64
        Gender             object
        ScheduledDay       object
        AppointmentDay     object
        Age                 int64
        Neighbourhood      object
        Scholarship         int64
        Hipertension        int64
        Diabetes            int64
        Alcoholism          int64
        Handcap             int64
        SMS_received        int64
        No-show            object
        dtype: object

In [10]: # to display a coincise summary of dataframe plus null value in each column
         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 [7]: # to return number of unique values in each column
        df.nunique()
```

```
Out[7]: PatientId          62299
        AppointmentID     110527
        Gender                 2
        ScheduledDay      103549
        AppointmentDay        27
        Age                  104
        Neighbourhood         81
        Scholarship            2
        Hipertension           2
        Diabetes               2
        Alcoholism             2
        Handcap                5
        SMS_received           2
        No-show                2
        dtype: int64
```

```
In [8]: #to return useful descriptive statistics for each column of data
        df.describe()
```

```
Out[8]:           PatientId  AppointmentID            Age    Scholarship  \
        count  1.105270e+05   1.105270e+05  110527.000000  110527.000000
        mean   1.474963e+14   5.675305e+06      37.088874       0.098266
        std    2.560949e+14   7.129575e+04      23.110205       0.297675
        min    3.921784e+04   5.030230e+06      -1.000000       0.000000
        25%    4.172614e+12   5.640286e+06      18.000000       0.000000
        50%    3.173184e+13   5.680573e+06      37.000000       0.000000
        75%    9.439172e+13   5.725524e+06      55.000000       0.000000
        max    9.999816e+14   5.790484e+06     115.000000       1.000000
```

```
        Hipertension       Diabetes     Alcoholism        Handcap  \
count  110527.000000  110527.000000  110527.000000  110527.000000
mean        0.197246       0.071865       0.030400       0.022248
std         0.397921       0.258265       0.171686       0.161543
min         0.000000       0.000000       0.000000       0.000000
25%         0.000000       0.000000       0.000000       0.000000
50%         0.000000       0.000000       0.000000       0.000000
75%         0.000000       0.000000       0.000000       0.000000
max         1.000000       1.000000       1.000000       4.000000

        SMS_received
count  110527.000000
mean        0.321026
std         0.466873
min         0.000000
25%         0.000000
50%         0.000000
75%         1.000000
max         1.000000
```

The result above shows: I) minimum age is a negative number. it shows that the column contains an erroneous value because age can never be negative. II) The Maximum Value in the Handcap column is 5. It is supposed to take Boolean values of 0 or 1. There is the need to print the values in these columns during inspection.

2.2.2 A) inspecting Handicap Column

```
In [6]: #To display values used in the Handap column.
        df.Handcap.unique()
```

```
Out[6]: array([0, 1, 2, 3, 4])
```

2.2.3 B) inspecting Age Column

```
In [10]: # check to see if there are errors with the ages. look out for decimal numbers and nega
         # display the all ages of the patients
         df.Age.unique()
```

```
Out[10]: array([ 62,  56,   8,  76,  23,  39,  21,  19,  30,  29,  22,  28,  54,
                 15,  50,  40,  46,   4,  13,  65,  45,  51,  32,  12,  61,  38,
                 79,  18,  63,  64,  85,  59,  55,  71,  49,  78,  31,  58,  27,
                  6,   2,  11,   7,   0,   3,   1,  69,  68,  60,  67,  36,  10,
                 35,  20,  26,  34,  33,  16,  42,   5,  47,  17,  41,  44,  37,
                 24,  66,  77,  81,  70,  53,  75,  73,  52,  74,  43,  89,  57,
                 14,   9,  48,  83,  72,  25,  80,  87,  88,  84,  82,  90,  94,
                 86,  91,  98,  92,  96,  93,  95,  97, 102, 115, 100,  99,  -1])
```

According to https://en.wikipedia.org/wiki/Child_development#:~:text=Some%20age%2Drelated%20deve some age-related development periods and examples of defined intervals include a) newborn (ages 0–4 weeks);

In [12]:
```
#To display total number of patients aged 0 years in the dataset
df.query('0 == Age').shape[0]
```

Out[12]: 3539

In [13]:
```
#To display total number of patients aged 0 years in the dataset
df.query('1 == Age').shape[0]
```

Out[13]: 2273

In [15]:
```
# To display the kind of diseases patients with age 0 have,
df[(df.Age == 0) & ((df.Hipertension == 1) | (df.Diabetes== 1) | (df.Alcoholism == 1) |
```

Out[15]:

	PatientId	AppointmentID	Gender	ScheduledDay	AppointmentDay	Age	Neighbourhood	Scholarship	Hipertension	Diabetes	Alcoholism	Handcap	SMS_received	No-show
98247	3.647246e+14	5788682	F	2016-06-08T13:18:12Z	2016-06-08T00:00:00Z	0	JABOUR	0	0	0	0	1	0	No

In [16]:
```
#To display total patients aged 100 years and above in the dataset
df.query('100 <= Age').shape[0]
```

Out[16]: 11

In [14]:
```
#To display total patients aged 100 years and above in the dataset with the type of sic
df.query('100 <= Age')
```

Out[14]:

	PatientId	AppointmentID	Gender	ScheduledDay	AppointmentDay	Age	Neighbourhood	Scholarship	Hipertension
58014	9.762948e+14	5651757	F	2016-05-03T09:14:53Z	2016-05-03T00:00:00Z	102	CONQUISTA	0	0
63912	3.196321e+13	5700278	F	2016-05-16T09:17:44Z	2016-05-19T00:00:00Z	115	ANDORINHAS	0	0
63915	3.196321e+13	5700279	F	2016-05-16T09:17:44Z	2016-05-19T00:00:00Z	115	ANDORINHAS	0	0
68127	3.196321e+13	5562812	F	2016-04-08T14:29:17Z	2016-05-16T00:00:00Z	115	ANDORINHAS	0	0
76284	3.196321e+13	5744037	F	2016-05-30T09:44:51Z	2016-05-30T00:00:00Z	115	ANDORINHAS	0	0
79270	9.739430e+12	5747809	M	2016-05-30T16:21:56Z	2016-05-30T00:00:00Z	115	ANDORINHAS	0	0
79272	9.739430e+12	5747808	M	2016-05-30T16:21:56Z	2016-05-30T00:00:00Z	115	ANDORINHAS	0	0
90372	2.342836e+11	5751563	F	2016-05-31T10:19:49Z	2016-05-31T00:00:00Z	115	ANDORINHAS	0	0
92084	5.578313e+13	5670914	F	2016-05-06T14:55:36Z	2016-05-06T00:00:00Z	115	ANDORINHAS	0	0
97666	7.482346e+14	5717451	F	2016-05-19T07:57:56Z	2016-05-19T00:00:00Z	115	ANDORINHAS	0	0
108506	3.939642e+11	5721152	F	2016-05-19T15:32:09Z	2016-05-19T00:00:00Z	115	ANDORINHAS	0	0

```
79270   2016-05-31T00:00:00Z  100       TABUAZEIRO            0             0
79272   2016-05-31T00:00:00Z  100       TABUAZEIRO            0             0
90372   2016-06-02T00:00:00Z  102      MARIA ORTIZ            0             0
92084   2016-06-03T00:00:00Z  100  ANTÔNIO HONÓRIO            0             0
97666   2016-06-03T00:00:00Z  115         SÃO JOSÉ            0             1
108506  2016-06-01T00:00:00Z  100          MARUÍPE            0             0

        Diabetes  Alcoholism  Handcap  SMS_received No-show
58014          0           0        0             0      No
63912          0           0        1             0     Yes
63915          0           0        1             0     Yes
68127          0           0        1             0     Yes
76284          0           0        1             0      No
79270          0           0        1             0      No
79272          0           0        1             0      No
90372          0           0        0             0      No
92084          0           0        0             1      No
97666          0           0        0             1      No
108506         0           0        0             0      No
```

2.2.4 c) inspecting spelling errors

```
In [15]: # To check for spelling errors with column names
         name_of_column = list(df.columns)
         # To print the column names as a list
         df.columns

Out[15]: Index(['PatientId', 'AppointmentID', 'Gender', 'ScheduledDay',
                'AppointmentDay', 'Age', 'Neighbourhood', 'Scholarship', 'Hipertension',
                'Diabetes', 'Alcoholism', 'Handcap', 'SMS_received', 'No-show'],
               dtype='object')
```

2.2.5 d) Check for duplicates

```
In [16]: df.duplicated().value_counts()

Out[16]: False    110527
         dtype: int64

In [21]: sum(df.duplicated())

Out[21]: 0
```

2.3 Comprehensive summary of observation and solutions to the problems found

After inspection, the following were observed:

1) There are 110527 rows in the dataset and 14 columns.
2) There are 3539 patients aged 0 years.

3) Patients aged 0 years are handicapped. They do not suffer from diabetics, hypertension and neither are they alcoholic
4) Some column names were spelt wrongly eg handcap.
5) The handicap column did not have boolean values. A patient who is handicapped is assigned a value of 1 and if not handicapped a value 0. In other to correct this, any column that contains a number greater than 0 will be assigned 1
6) The age column contained an erronous value of minus one. This might be due to typographic error. After checking to see the total number of rows with age of 1 years we found that there are 2273 patients that are a year old. It is safe to drop the row with errors since it will not have significant effect on the result.
7) There are eleven patients whose ages are greater than or equals to 100 years. Majority of them are handicapped.
8) No missing values was found.
9) There are no duplicate files in the dtaset
10) It was also observed that the ScheduledDay and AppointmentDay columns were read as objects (strings) instead of Date Time object we wont be converting it since we are not performing time difference.
11) from the Pdf file and the dataset it is seen that in the No-show column if the patient showed up to their appointment the person is assigned 'No and 'Yes' if they did not show up.

3 Data Cleaning

3.0.1 Renaming column

```
In [9]: #user-defined function to rename column

        def rename_column(df, old_columns, new_columns):
            if len(old_columns) !=len(new_columns):
                return print('Error!!! Number of old_columns must be equal to number of new_colu
            else:
                for i in range(len(old_columns)):
                    df.rename(columns = { old_columns[i] : new_columns[i] }, inplace = True)
                return df

In [10]: #renaming columns if user-defined function is called
         old_columns = ['ScheduledDay', 'PatientId', 'AppointmentDay', 'AppointmentID', 'Handcap
         new_columns = ['Scheduled_Day', 'Patient_Id', 'Appointment_Day', 'Appointment_ID', 'Han
         rename_column(df, old_columns, new_columns)

Out[10]:          Patient_Id  Appointment_ID Gender         Scheduled_Day  \
         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
         5       9.598513e+13         5626772      F  2016-04-27T08:36:51Z
         6       7.336882e+14         5630279      F  2016-04-27T15:05:12Z
         7       3.449833e+12         5630575      F  2016-04-27T15:39:58Z
```

8	5.639473e+13	5638447	F	2016-04-29T08:02:16Z
9	7.812456e+13	5629123	F	2016-04-27T12:48:25Z
10	7.345362e+14	5630213	F	2016-04-27T14:58:11Z
11	7.542951e+12	5620163	M	2016-04-26T08:44:12Z
12	5.666548e+14	5634718	F	2016-04-28T11:33:51Z
13	9.113946e+14	5636249	M	2016-04-28T14:52:07Z
14	9.988472e+13	5633951	F	2016-04-28T10:06:24Z
15	9.994839e+10	5620206	F	2016-04-26T08:47:27Z
16	8.457439e+13	5633121	M	2016-04-28T08:51:47Z
17	1.479497e+13	5633460	F	2016-04-28T09:28:57Z
18	1.713538e+13	5621836	F	2016-04-26T10:54:18Z
19	7.223289e+12	5640433	F	2016-04-29T10:43:14Z
20	6.222575e+14	5626083	F	2016-04-27T07:51:14Z
21	1.215484e+13	5628338	F	2016-04-27T10:50:45Z
22	8.632298e+14	5616091	M	2016-04-25T13:29:16Z
23	2.137540e+14	5634142	F	2016-04-28T10:27:05Z
24	8.734858e+12	5641780	F	2016-04-29T14:19:19Z
25	5.819370e+12	5624020	M	2016-04-26T15:04:17Z
26	2.578785e+10	5641781	F	2016-04-29T14:19:42Z
27	1.215484e+13	5628345	F	2016-04-27T10:51:45Z
28	5.926172e+12	5642400	M	2016-04-29T15:48:02Z
29	1.225776e+12	5642186	F	2016-04-29T15:16:29Z
...	...	...	...	...
110497	7.935892e+14	5757745	M	2016-06-01T09:46:33Z
110498	9.433654e+13	5787655	F	2016-06-08T10:21:14Z
110499	8.219692e+14	5757697	F	2016-06-01T09:42:56Z
110500	4.434384e+14	5787233	F	2016-06-08T09:35:13Z
110501	4.544252e+11	5758133	M	2016-06-01T10:19:12Z
110502	7.316229e+14	5787937	F	2016-06-08T10:50:42Z
110503	2.362182e+13	5759473	F	2016-06-01T13:00:36Z
110504	9.947983e+12	5788052	F	2016-06-08T11:06:21Z
110505	5.667344e+13	5758455	F	2016-06-01T10:45:50Z
110506	8.973883e+11	5758779	M	2016-06-01T11:09:20Z
110507	4.769462e+14	5786918	F	2016-06-08T09:04:18Z
110508	9.433654e+13	5757656	F	2016-06-01T09:41:00Z
110509	4.952968e+14	5786750	M	2016-06-08T08:50:51Z
110510	2.362182e+13	5757587	F	2016-06-01T09:35:48Z
110511	8.235996e+11	5786742	F	2016-06-08T08:50:20Z
110512	9.876246e+13	5786368	F	2016-06-08T08:20:01Z
110513	8.674778e+13	5785964	M	2016-06-08T07:52:55Z
110514	2.695685e+12	5786567	F	2016-06-08T08:35:31Z
110515	6.456342e+14	5778621	M	2016-06-06T15:58:05Z
110516	6.923772e+13	5780205	F	2016-06-07T07:45:16Z
110517	5.574942e+12	5780122	F	2016-06-07T07:38:34Z
110518	7.263315e+13	5630375	F	2016-04-27T15:15:06Z
110519	6.542388e+13	5630447	F	2016-04-27T15:23:14Z
110520	9.969977e+14	5650534	F	2016-05-03T07:51:47Z
110521	3.635534e+13	5651072	F	2016-05-03T08:23:40Z

```
110522  2.572134e+12         5651768      F  2016-05-03T09:15:35Z
110523  3.596266e+12         5650093      F  2016-05-03T07:27:33Z
110524  1.557663e+13         5630692      F  2016-04-27T16:03:52Z
110525  9.213493e+13         5630323      F  2016-04-27T15:09:23Z
110526  3.775115e+14         5629448      F  2016-04-27T13:30:56Z

            Appointment_Day  Age      Neighbourhood  Scholarship  \
0       2016-04-29T00:00:00Z   62    JARDIM DA PENHA            0
1       2016-04-29T00:00:00Z   56    JARDIM DA PENHA            0
2       2016-04-29T00:00:00Z   62      MATA DA PRAIA            0
3       2016-04-29T00:00:00Z    8  PONTAL DE CAMBURI            0
4       2016-04-29T00:00:00Z   56    JARDIM DA PENHA            0
5       2016-04-29T00:00:00Z   76          REPÚBLICA            0
6       2016-04-29T00:00:00Z   23         GOIABEIRAS            0
7       2016-04-29T00:00:00Z   39         GOIABEIRAS            0
8       2016-04-29T00:00:00Z   21         ANDORINHAS            0
9       2016-04-29T00:00:00Z   19          CONQUISTA            0
10      2016-04-29T00:00:00Z   30     NOVA PALESTINA            0
11      2016-04-29T00:00:00Z   29     NOVA PALESTINA            0
12      2016-04-29T00:00:00Z   22     NOVA PALESTINA            1
13      2016-04-29T00:00:00Z   28     NOVA PALESTINA            0
14      2016-04-29T00:00:00Z   54     NOVA PALESTINA            0
15      2016-04-29T00:00:00Z   15     NOVA PALESTINA            0
16      2016-04-29T00:00:00Z   50     NOVA PALESTINA            0
17      2016-04-29T00:00:00Z   40          CONQUISTA            1
18      2016-04-29T00:00:00Z   30     NOVA PALESTINA            1
19      2016-04-29T00:00:00Z   46           DA PENHA            0
20      2016-04-29T00:00:00Z   30     NOVA PALESTINA            0
21      2016-04-29T00:00:00Z    4          CONQUISTA            0
22      2016-04-29T00:00:00Z   13          CONQUISTA            0
23      2016-04-29T00:00:00Z   46          CONQUISTA            0
24      2016-04-29T00:00:00Z   65         TABUAZEIRO            0
25      2016-04-29T00:00:00Z   46          CONQUISTA            0
26      2016-04-29T00:00:00Z   45     BENTO FERREIRA            0
27      2016-04-29T00:00:00Z    4          CONQUISTA            0
28      2016-04-29T00:00:00Z   51          SÃO PEDRO            0
29      2016-04-29T00:00:00Z   32       SANTA MARTHA            0
...                      ...  ...                ...          ...
110497  2016-06-01T00:00:00Z   76        MARIA ORTIZ            0
110498  2016-06-08T00:00:00Z   59        MARIA ORTIZ            0
110499  2016-06-01T00:00:00Z   66        MARIA ORTIZ            0
110500  2016-06-08T00:00:00Z   59        MARIA ORTIZ            0
110501  2016-06-01T00:00:00Z   44        MARIA ORTIZ            0
110502  2016-06-08T00:00:00Z   22         GOIABEIRAS            0
110503  2016-06-01T00:00:00Z   64       SOLON BORGES            0
110504  2016-06-08T00:00:00Z    4        MARIA ORTIZ            0
110505  2016-06-01T00:00:00Z   55        MARIA ORTIZ            0
110506  2016-06-01T00:00:00Z    5        MARIA ORTIZ            0
```

110507	2016-06-08T00:00:00Z	0	MARIA ORTIZ	0		
110508	2016-06-01T00:00:00Z	59	MARIA ORTIZ	0		
110509	2016-06-08T00:00:00Z	33	MARIA ORTIZ	0		
110510	2016-06-01T00:00:00Z	64	SOLON BORGES	0		
110511	2016-06-08T00:00:00Z	14	MARIA ORTIZ	0		
110512	2016-06-08T00:00:00Z	41	MARIA ORTIZ	0		
110513	2016-06-08T00:00:00Z	2	ANTÔNIO HONÓRIO	0		
110514	2016-06-08T00:00:00Z	58	MARIA ORTIZ	0		
110515	2016-06-08T00:00:00Z	33	MARIA ORTIZ	0		
110516	2016-06-08T00:00:00Z	37	MARIA ORTIZ	0		
110517	2016-06-07T00:00:00Z	19	MARIA ORTIZ	0		
110518	2016-06-07T00:00:00Z	50	MARIA ORTIZ	0		
110519	2016-06-07T00:00:00Z	22	MARIA ORTIZ	0		
110520	2016-06-07T00:00:00Z	42	MARIA ORTIZ	0		
110521	2016-06-07T00:00:00Z	53	MARIA ORTIZ	0		
110522	2016-06-07T00:00:00Z	56	MARIA ORTIZ	0		
110523	2016-06-07T00:00:00Z	51	MARIA ORTIZ	0		
110524	2016-06-07T00:00:00Z	21	MARIA ORTIZ	0		
110525	2016-06-07T00:00:00Z	38	MARIA ORTIZ	0		
110526	2016-06-07T00:00:00Z	54	MARIA ORTIZ	0		

	Hypertension	Diabetes	Alcoholism	Handicap	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
5	1	0	0	0	0	No
6	0	0	0	0	0	Yes
7	0	0	0	0	0	Yes
8	0	0	0	0	0	No
9	0	0	0	0	0	No
10	0	0	0	0	0	No
11	0	0	0	0	1	Yes
12	0	0	0	0	0	No
13	0	0	0	0	0	No
14	0	0	0	0	0	No
15	0	0	0	0	1	No
16	0	0	0	0	0	No
17	0	0	0	0	0	Yes
18	0	0	0	0	1	No
19	0	0	0	0	0	No
20	0	0	0	0	0	Yes
21	0	0	0	0	0	Yes
22	0	0	0	0	1	Yes
23	0	0	0	0	0	No
24	0	0	0	0	0	No
25	1	0	0	0	1	No

```
26                1          0             0          0              0      No
27                0          0             0          0              0      No
28                0          0             0          0              0      No
29                0          0             0          0              0      No
...             ...        ...           ...        ...            ...     ...
110497            0          0             0          0              0      No
110498            0          0             0          0              0      No
110499            1          1             0          0              0      No
110500            0          0             0          0              0      No
110501            0          0             0          0              0      No
110502            0          0             0          0              0      No
110503            0          0             0          0              0      No
110504            0          0             0          0              0      No
110505            0          0             0          0              0      No
110506            0          0             0          0              0      No
110507            0          0             0          0              0      No
110508            0          0             0          0              0      No
110509            0          0             0          0              0      No
110510            0          0             0          0              0      No
110511            0          0             0          0              0      No
110512            0          0             0          0              0      No
110513            0          0             0          0              0      No
110514            0          0             0          0              0      No
110515            1          0             0          0              0     Yes
110516            0          0             0          0              0     Yes
110517            0          0             0          0              0      No
110518            0          0             0          0              1      No
110519            0          0             0          0              1      No
110520            0          0             0          0              1      No
110521            0          0             0          0              1      No
110522            0          0             0          0              1      No
110523            0          0             0          0              1      No
110524            0          0             0          0              1      No
110525            0          0             0          0              1      No
110526            0          0             0          0              1      No

[110527 rows x 14 columns]
```

In [11]:
```
# display first 3 rows of dataset to view changes made
df.head(3)
```

Out[11]:
```
     Patient_Id  Appointment_ID Gender         Scheduled_Day  \
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

         Appointment_Day  Age    Neighbourhood  Scholarship  Hypertension  \
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

   Diabetes  Alcoholism  Handicap  SMS_received No_show
0         0           0         0             0      No
1         0           0         0             0      No
2         0           0         0             0      No
```

3.0.2 Dropping rows that contain erronous values

```
In [12]: # To drop the row of the patient with age -1
         df.drop(df[df['Age'] == -1].index, axis = 0, inplace = True)

In [13]: # check to confirm that the changes has been effected.
         df.Age.unique()

Out[13]: array([ 62,  56,   8,  76,  23,  39,  21,  19,  30,  29,  22,  28,  54,
                 15,  50,  40,  46,   4,  13,  65,  45,  51,  32,  12,  61,  38,
                 79,  18,  63,  64,  85,  59,  55,  71,  49,  78,  31,  58,  27,
                  6,   2,  11,   7,   0,   3,   1,  69,  68,  60,  67,  36,  10,
                 35,  20,  26,  34,  33,  16,  42,   5,  47,  17,  41,  44,  37,
                 24,  66,  77,  81,  70,  53,  75,  73,  52,  74,  43,  89,  57,
                 14,   9,  48,  83,  72,  25,  80,  87,  88,  84,  82,  90,  94,
                 86,  91,  98,  92,  96,  93,  95,  97, 102, 115, 100,  99])

In [14]: df['Handicap'] = np.where(df['Handicap']>0, 1, 0)

In [26]: #check to confirm that the changes has been effected.
         df.Handicap.unique()

Out[26]: array([0, 1])

In [15]: # check if the data types of columns are in good shape
         df.dtypes

Out[15]: Patient_Id         float64
         Appointment_ID       int64
         Gender              object
         Scheduled_Day       object
         Appointment_Day     object
         Age                  int64
         Neighbourhood       object
         Scholarship          int64
         Hypertension         int64
         Diabetes             int64
         Alcoholism           int64
         Handicap             int64
         SMS_received         int64
         No_show             object
         dtype: object
```

```
In [16]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 110526 entries, 0 to 110526
Data columns (total 14 columns):
Patient_Id         110526 non-null float64
Appointment_ID     110526 non-null int64
Gender             110526 non-null object
Scheduled_Day      110526 non-null object
Appointment_Day    110526 non-null object
Age                110526 non-null int64
Neighbourhood      110526 non-null object
Scholarship        110526 non-null int64
Hypertension       110526 non-null int64
Diabetes           110526 non-null int64
Alcoholism         110526 non-null int64
Handicap           110526 non-null int64
SMS_received       110526 non-null int64
No_show            110526 non-null object
dtypes: float64(1), int64(8), object(5)
memory usage: 12.6+ MB
```

```
In [17]: df.head(4)
```

Out[17]:

	Patient_Id	Appointment_ID	Gender	Scheduled_Day	\
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	

	Appointment_Day	Age	Neighbourhood	Scholarship	Hypertension	\
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	

	Diabetes	Alcoholism	Handicap	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

```
In [32]: # comprehensive summary of dataset after cleaning
         df.info ()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 110526 entries, 0 to 110526
Data columns (total 14 columns):
```

```
Patient_Id          110526 non-null float64
Appointment_ID      110526 non-null int64
Gender              110526 non-null object
Scheduled_Day       110526 non-null object
Appointment_Day     110526 non-null object
Age                 110526 non-null int64
Neighbourhood       110526 non-null object
Scholarship         110526 non-null int64
Hypertension        110526 non-null int64
Diabetes            110526 non-null int64
Alcoholism          110526 non-null int64
Handicap            110526 non-null int64
SMS_Received        110526 non-null int64
No_Show             110526 non-null object
dtypes: float64(1), int64(8), object(5)
memory usage: 12.6+ MB
```

After cleaning: 1) The rows in the dataset reduced from 110527 to 110526 because we dropped the row where the age was equal to -1. 2) The columns in the dataset is still 14 in number 3) The datatypes of the columns are in the correct order

```
In [18]: df.head(2)
```

```
Out[18]:      Patient_Id  Appointment_ID Gender         Scheduled_Day  \
         0  2.987250e+13         5642903      F  2016-04-29T18:38:08Z
         1  5.589978e+14         5642503      M  2016-04-29T16:08:27Z

                Appointment_Day  Age    Neighbourhood  Scholarship  Hypertension  \
         0  2016-04-29T00:00:00Z   62  JARDIM DA PENHA            0             1
         1  2016-04-29T00:00:00Z   56  JARDIM DA PENHA            0             0

            Diabetes  Alcoholism  Handicap  SMS_received No_show
         0         0           0         0             0      No
         1         0           0         0             0      No
```

```
In [19]: #code to change 'No' to 'showed' in the 'No_show' column
         df.loc [df['No_show'] == 'No', 'No_show'] = 'Showed'

         #code to change 'Yes' to 'Missed' in the 'No_show' column
         df.loc [df['No_show'] == 'Yes', 'No_show'] = 'Missed'
```

```
In [20]: #code to view if changes has been effected.
         df.head(4)
```

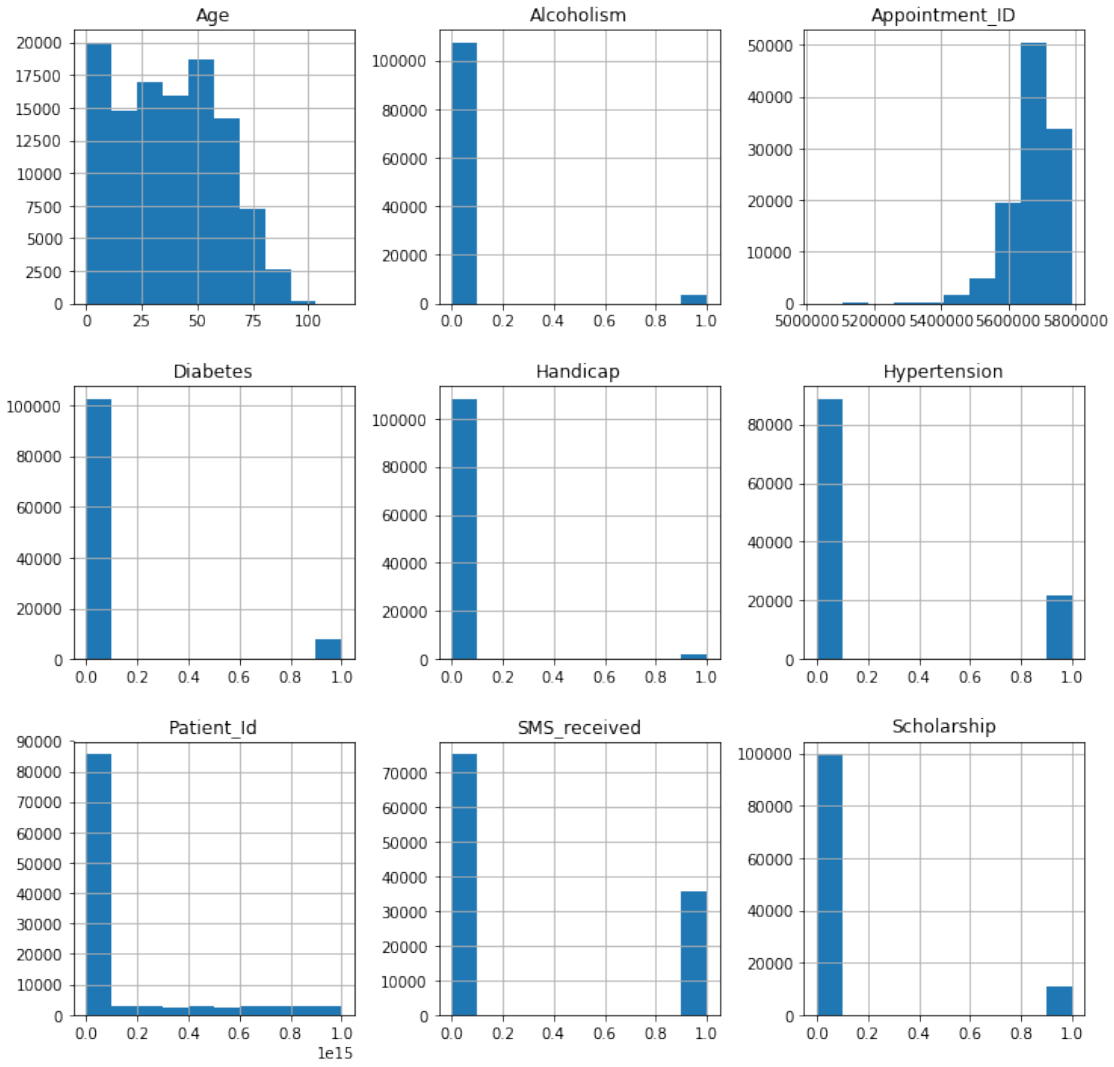
```
Out[20]:      Patient_Id  Appointment_ID Gender         Scheduled_Day  \
         0  2.987250e+13         5642903      F  2016-04-29T18:38:08Z
         1  5.589978e+14         5642503      M  2016-04-29T16:08:27Z
         2  4.262962e+12         5642549      F  2016-04-29T16:19:04Z
```

```
3  8.679512e+11         5642828      F  2016-04-29T17:29:31Z

         Appointment_Day  Age      Neighbourhood  Scholarship  Hypertension  \
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

   Diabetes  Alcoholism  Handicap  SMS_received No_show
0         0           0         0             0  Showed
1         0           0         0             0  Showed
2         0           0         0             0  Showed
3         0           0         0             0  Showed
```

The 'yes' and 'No' entries in the 'no_show' column is confusing. For clarity, I changed the 'No' to 'Showed' to represent patients that showed up to their appointments and 'Yes' to 'Missed' indicating those that didnt show up for their appointment.

Exploratory Data Analysis

```
In [21]: #Printing to explore the data
         df.hist(figsize = (12,12));
```

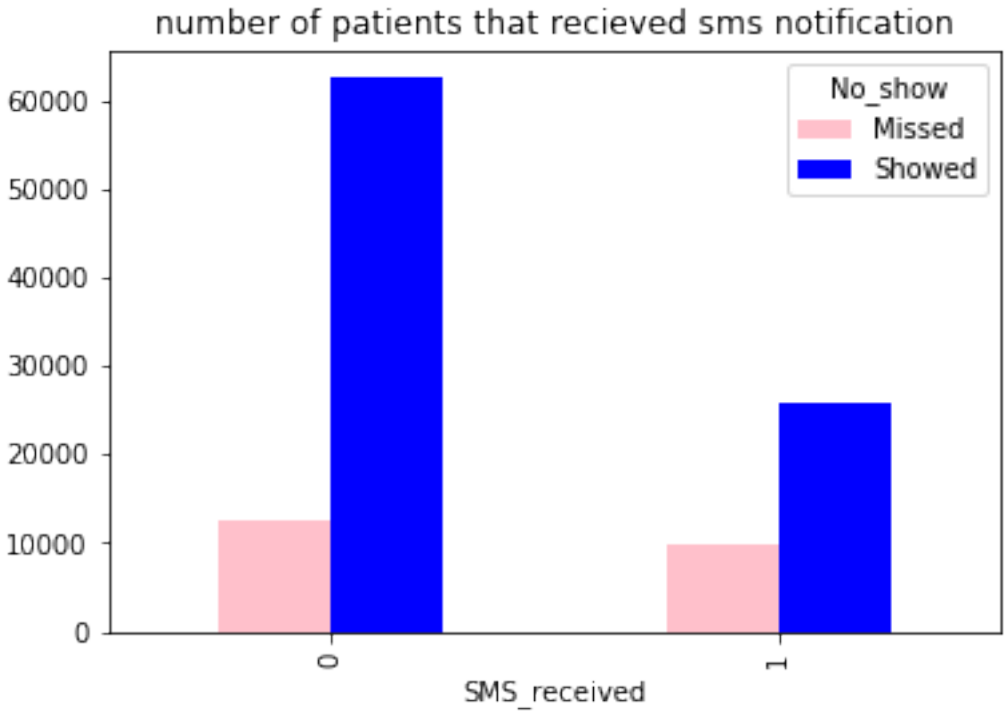


3.0.3 Research Question 1 (Will the patient show-up if they did or did not recieve SMS message?)

```
In [23]: # code to group  patients that recieved sms into show_up or missed and storing. this wi
         sms = df.groupby('SMS_received').No_show.value_counts()

In [25]: #view the grouping of people that recieved sms
         sms

Out[25]: SMS_received  No_show
         0             Showed     62509
                       Missed     12535
         1             Showed     25698
                       Missed      9784
         Name: No_show, dtype: int64
```

In [26]:
```
#code to create a dictinary that maps 1 and 0 to 'received' and 'not recieved'
sms_status = {1:'received', 0:'Not received'}

#plotting a chart that shows the number of patients that recieved an sms reminder
sms_reminder = sms_reminder.unstack()
sms_reminder.plot(kind='bar', color = ['pink', 'blue'], title = 'number of patients tha
print ('Patients that did not recieve sms = 0 and Patients that recieved sms = 1' )
```

```
Patients that did not recieve sms = 0 and Patients that recieved sms = 1
```

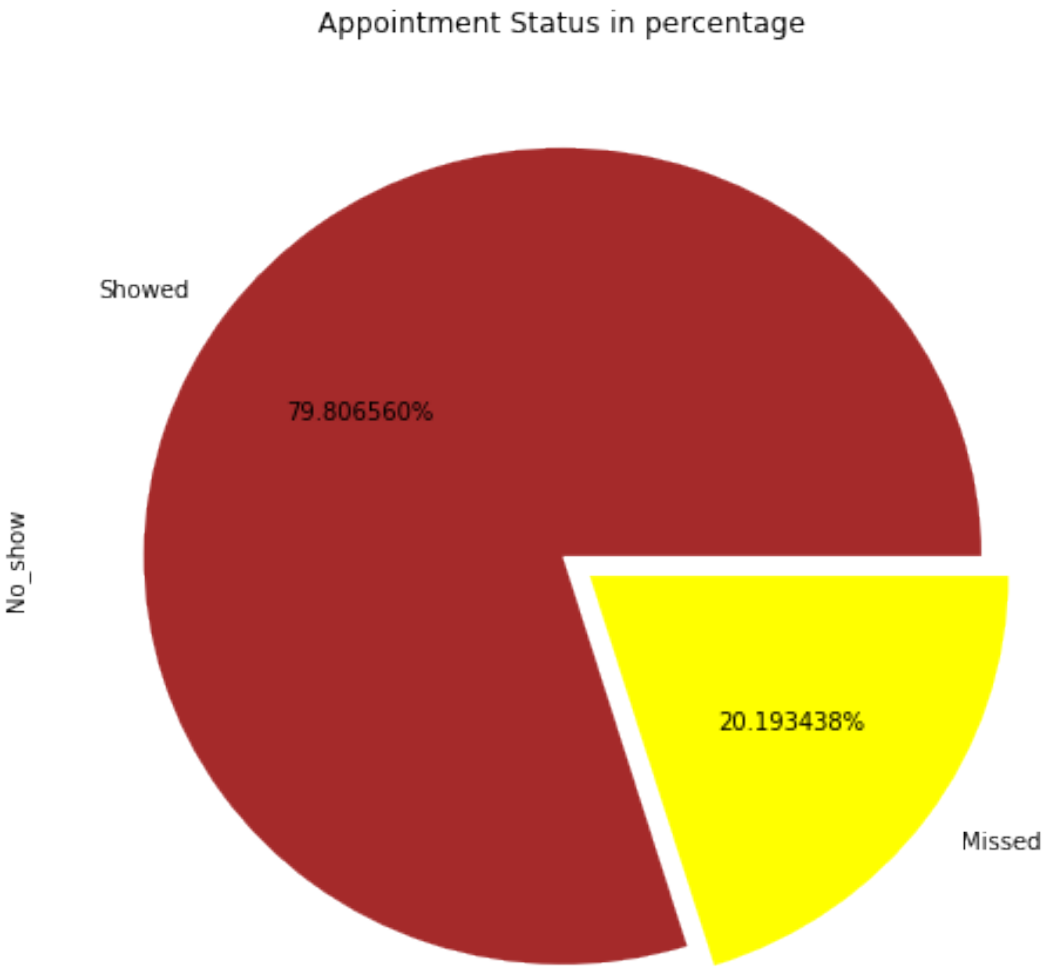


3.0.4 Observation

It is seen that the patients who didn't receive SMS showed up more than those who received SMS.

3.0.5 Research Question 2 (What was the Percentage of Patients that Showed Up for Appointment?)

In [27]:
```
#this code will plot a piechart to show the percentage of patients that those who showe
df.No_show.value_counts().plot.pie(figsize=(8,8), colors = ['brown', 'yellow'], title =
plt.show()
```



3.1 Research Question 3: What gender show up more for appointment?

```
In [28]: # code to show total count of patients that missed their appointment vs those who showe
         df[["Gender", "No_show"]].groupby("No_show").count()
```

```
Out[28]:          Gender
         No_show
         Missed    22319
         Showed    88207
```

it is seen that 22316 patients missed their appointment and 88205 patients showed up for their appointents.

```
In [29]: # code to show the number of females and males that showed up or missed their appointme
         df.groupby('Gender').No_show.value_counts()
```

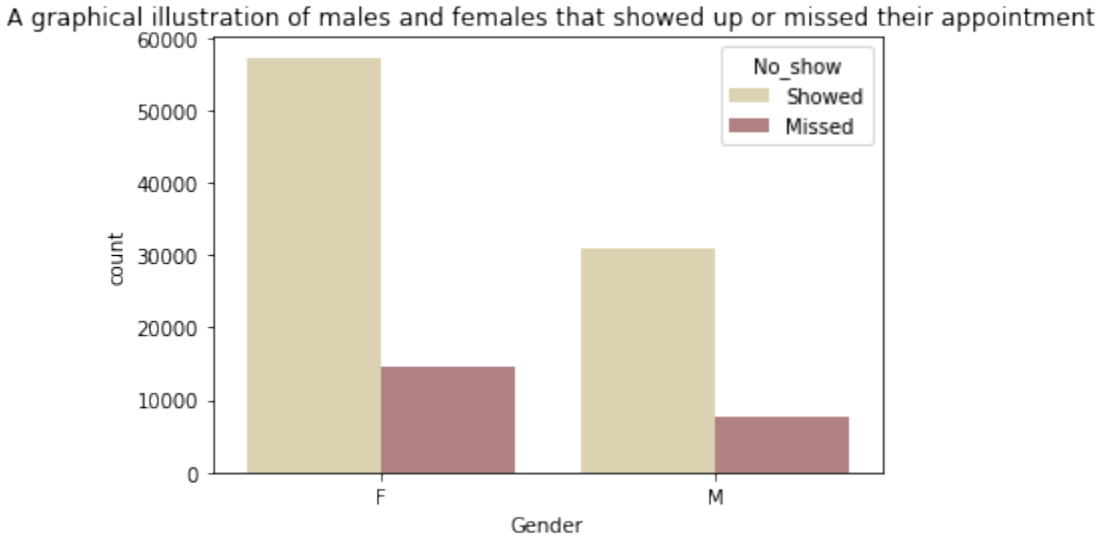
```
Out[29]: Gender  No_show
         F       Showed     57245
                 Missed     14594
         M       Showed     30962
                 Missed      7725
         Name: No_show, dtype: int64
```

```
In [32]: #code that will get row that contains female and male that missed their appointment and

         Appointmentmissed_female = len(df.query('No_show == "Missed" and Gender == "F"'))
         Appointmentmissed_male = len(df.loc[(df['Gender'] == "M") & (df['No_show'] == "Missed")
```

```
In [35]: #Code that will get rows of all female appointment and male appointment
         female_appointment = len(df.loc[df['Gender'] == "F"])
         male_appointment= len(df.loc[df['Gender'] == "M"])
```

```
In [36]: # code that calculates ratio of missed appointment to total appointment for male and fe
         ratio_female = int(round(Appointmentmissed_female/female_appointment*100))
         ratio_male = int(round(Appointmentmissed_male/male_appointment*100))
```

```
In [39]: #Code to plot graph that shows the comparison between number of females and males that

         ax = sns.countplot(x=df.Gender, hue=df.No_show, data=df, palette = 'pink_r')
         ax.set_title(" A graphical illustration of males and females that showed up or missed t
         x_ticks_labels=['female', 'male']
         plt.show();
```



3.1.1 Observation

Conclusions After analysis, it is seen that: 1) recieving SMS did not really have significant impact on showing up for appointment. in other words, from this dataset there was was no way

to ascertain that it was becausepatients did not recieve sms that was why they did not show up for appointment 2) age is an important factor that can influence a patient from not showing up to a medical appointment. 3) 78.81% of patients showed up for their appointment and 20.19% missed their appointment. Furthermore, the female gender showed up more to medical appointments than males.

3.1.2 LIMITATION:

One major limitation was the interpretability of the 'No_show' column. There was a need to make modifications to the No_show column.

3.1.3 SUGGESTION:

I humbly suggest that the dataset should include the following columns: proximity of patients house to the hospital and the means of transporation that can be used. This will help us to know if distance

It could have been helpful to see if the distance from the patient's home to the hospital is a factor that influence whether or not a patient would show up for an appointment.

```
In [ ]: ## Submitting your Project

        > **Tip**: Before you submit your project, you need to create a .html or .pdf version of

        > **Tip**: Alternatively, you can download this report as .html via the **File** > **Dow

        > **Tip**: Once you've done this, you can submit your project by clicking on the "Submit
```

```
In [442]: from subprocess import call
          call(['python', '-m', 'nbconvert', 'Investigate_a_Dataset.ipynb'])
```

```
Out[442]: 0
```

```
In [ ]:
```