Investigate_a_Dataset

October 13, 2018

1 Project: Analysis of No-Show Medical Appointment Data From Brazil

1.1 Table of Contents

Introduction

The data shows the demographics for 110,527 no-show medical applointments. No-show is defined as a person, who has a scheduled appointment, does not show up for that appointment. Data came from Kaggle https://www.kaggle.com/joniarroba/noshowappointments. There are 14 columns in the dataframe. The following analysis explores which factors included in this dataset, if any affect the ability of patients to keep medical appointments. A description of the data is as follows:

PatientId - Identification of a patient
AppointmentID - Identification of each appointment
Gender = Male or Female
Appointment Day = The day of the appointment
Scheduled Day = The day someone called or registered the appointment, this is before appointment of course
Age
Neighbourhood = Where the appointment takes place
Scholarship = True or False
Hypertension (column name = hipertension) = True or False
Diabetes = True or False
Alcoholism = True or False
Handicap (column name = Handcap) = True or False
SMS_received = 1 or more messages sent to the patient
No-show = True or False

```
In [72]: ##Import numpy, pandas, matplotlib

         import numpy as np
```

```
import pandas as pd
import matplotlib.pyplot as plt
% matplotlib inline
```

Data Wrangling

1.1.1 General Properties

Load the data from the Kaggle website. Take an initial look at the information contained in the dataframe.

```
In [73]: ## Load the dataset and briefly look at the contents
         df = pd.read_csv('Appointment Data.csv')
         df.head(5)
```

```
Out[73]:       PatientId  AppointmentID Gender          ScheduledDay  \
         0  2.987250e+13        5642903      F  2016-04-29T18:38:08Z
         1  5.589980e+14        5642503      M  2016-04-29T16:08:27Z
         2  4.262960e+12        5642549      F  2016-04-29T16:19:04Z
         3  8.679510e+11        5642828      F  2016-04-29T17:29:31Z
         4  8.841190e+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
```

Further analysis on the dataframe indicates that there are 110527 rows of data with no null values. A statistical description of the data gives an overview. Initial inspection shows that the maximum age of all patients is 115.

```
In [74]: ## Show size of dataframe
         df.shape
```

```
Out[74]: (110527, 14)
```

```
In [75]: ## Show statistical description
         df.describe()
```

Out[75]:

```
          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    3.920000e+04   5.030230e+06      -1.000000       0.000000
25%    4.172615e+12   5.640286e+06      18.000000       0.000000
50%    3.173180e+13   5.680573e+06      37.000000       0.000000
75%    9.439170e+13   5.725524e+06      55.000000       0.000000
max    9.999820e+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
```

In [76]:

```
## Show information about the data; in this case, look for null values
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
```

There are 14 columns included in the dataframe. Of those, PatientID, AppointmentID, and ScheduledDay are unlikely to yield insight into why patients skip healthcare appointments or to explain how to increase compliance in showing up for appointments. Remove these columns.

```
In [77]: ## Drop columns that are not necessary for analysis
         df.drop(['PatientId', 'AppointmentID', 'ScheduledDay'], axis = 1, inplace = True)
```

```
In [78]: ## Check to see that the dataframe contains only the columns for analysis
         df.head(5)
```

```
Out[78]:   Gender        AppointmentDay  Age      Neighbourhood  Scholarship  \
         0      F  2016-04-29T00:00:00Z   62    JARDIM DA PENHA            0
         1      M  2016-04-29T00:00:00Z   56    JARDIM DA PENHA            0
         2      F  2016-04-29T00:00:00Z   62      MATA DA PRAIA            0
         3      F  2016-04-29T00:00:00Z    8  PONTAL DE CAMBURI            0
         4      F  2016-04-29T00:00:00Z   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
```

1.1.2 Data Cleaning: Making the Data Easier to Analyze

Change text data to numerical data and extract the month and day of week from the appointment date to form two new columns.

```
In [79]: ## Import datetime, which allows analyst to separate elements of the date, ie day of we
         import datetime as dt
```

```
In [80]: ## Create columns for day of week and month of appointment
         df['AppointmentDay'] = pd.to_datetime(df['AppointmentDay'])
         df['Month'] = df['AppointmentDay'].dt.month
         df['WeekDay'] = df['AppointmentDay'].dt.weekday
```

```
In [81]: ## Change No_show and Gender columns to integer data points for analysis
         df.rename(columns={'No-show': 'No_show'}, inplace=True)
         df.No_show.replace(['Yes', 'No'], [1, 0], inplace=True)
         df.Gender.replace(['F', 'M'], [1, 0], inplace = True)
```

In [82]:
```python
## Insert row numbers
df.insert(0, 'Row_num', range(1, 1 + len(df)))
df.head()
```

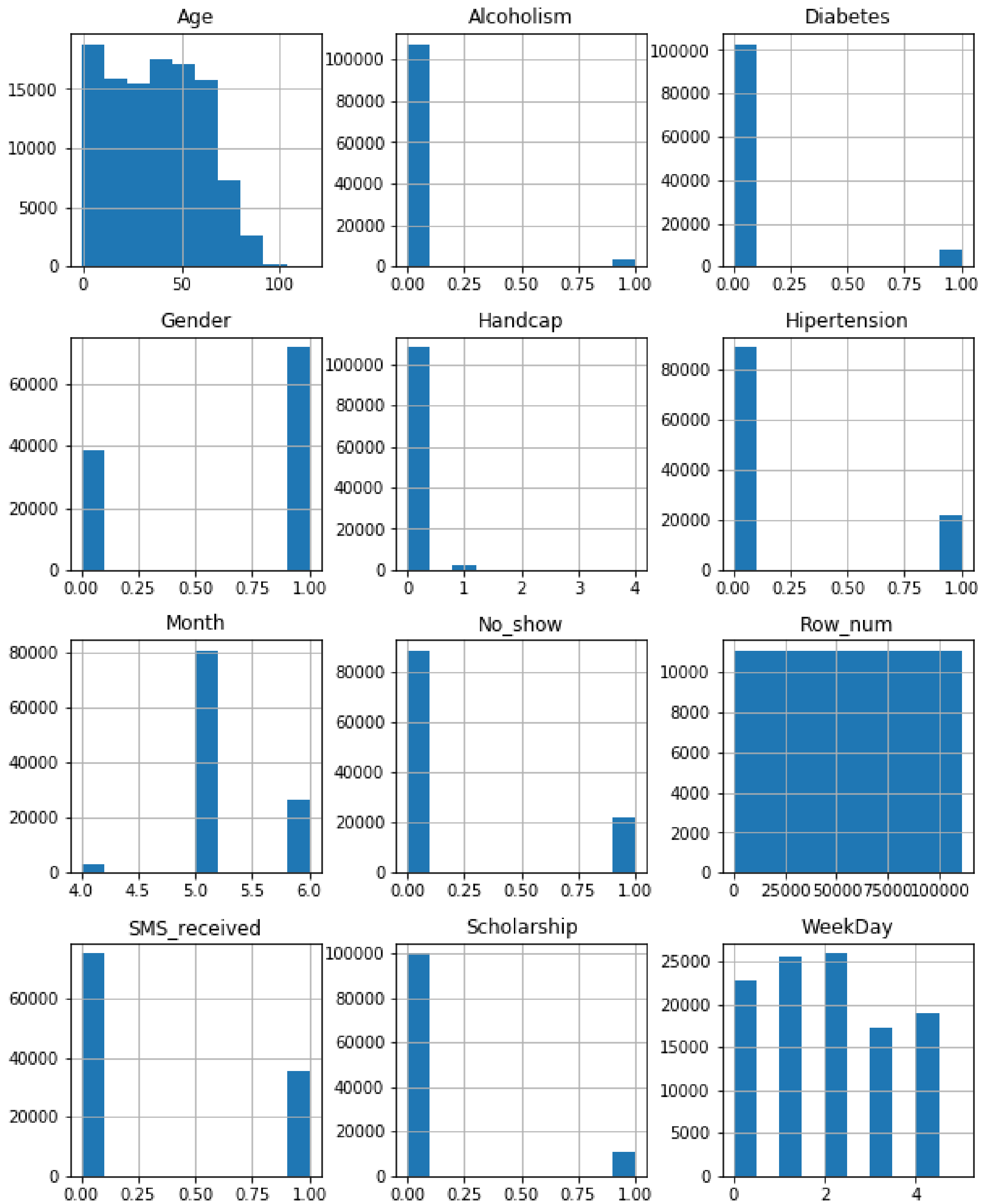
Out[82]:
```
   Row_num  Gender AppointmentDay  Age      Neighbourhood  Scholarship  \
0        1       1     2016-04-29   62    JARDIM DA PENHA            0
1        2       0     2016-04-29   56    JARDIM DA PENHA            0
2        3       1     2016-04-29   62      MATA DA PRAIA            0
3        4       1     2016-04-29    8  PONTAL DE CAMBURI            0
4        5       1     2016-04-29   56    JARDIM DA PENHA            0

   Hipertension  Diabetes  Alcoholism  Handcap  SMS_received  No_show  Month  \
0             1         0           0        0             0        0      4
1             0         0           0        0             0        0      4
2             0         0           0        0             0        0      4
3             0         0           0        0             0        0      4
4             1         1           0        0             0        0      4

   WeekDay
0        4
1        4
2        4
3        4
4        4
```

Exploratory Data Analysis

1.1.3 Which factors affect the ability of patients to keep an appointment?

A quick look at the visual data by column does not show anything unusual.

In [83]:
```python
## Visual check of the data presented to see if anything stands out
df.hist(figsize = (10, 13));
```



Calcualting correlation across the data shows weak correlations between all data points except for SMS_received and No_show. This correlation is moderate.

```
In [84]: ## Create variables for show and no_show appointments
         show = df.No_show == 0
         no_show = df.No_show == 1
```

```
In [85]: ## Check the correlation coefficients to see if there is a strong relationship in any c
         df.corr()
```

Out[85]:

	Row_num	Gender	Age	Scholarship	Hipertension	Diabetes	Alcoholism	Handcap	SMS_received	No_show	Month	WeekDay
Row_num	1.000000	0.017935	0.015960	0.000771	0.004828	0.013588	-0.025579	0.000184	0.069934	-0.017192	0.769393	-0.038182
Gender	0.017935	1.000000	0.106440	0.114293	0.055718	0.032554	-0.106167	-0.022814	0.046298	0.004119	0.006051	-0.003916
Age	0.015960	0.106440	1.000000	-0.092457	0.504586	0.292391	0.095811	0.078033	0.012643	-0.060319	0.014547	0.003088
Scholarship	0.000771	0.114293	-0.092457	1.000000	-0.019729	-0.024894	0.035022	-0.008586	0.001194	0.029135	-0.002588	-0.000673
Hipertension	0.004828	0.055718	0.504586	-0.019729	1.000000	0.433086	0.087971	0.080083	-0.006267	-0.035701	0.003779	0.003455
Diabetes	0.013588	0.032554	0.292391	-0.024894	0.433086	1.000000	0.018474	0.057530	-0.014550	-0.015180	0.003741	0.006614
Alcoholism	-0.025579	-0.106167	0.095811	0.035022	0.087971	0.018474	1.000000	0.004648	-0.026147	-0.000196	0.003920	0.002701
Handcap	0.000184	-0.022814	0.078033	-0.008586	0.080083	0.057530	0.004648	1.000000	-0.024161	-0.006076	-0.001479	0.004352
SMS_received	0.069934	0.046298	0.012643	0.001194	-0.006267	-0.014550	-0.026147	-0.024161	1.000000	0.126431	0.108070	-0.089858
No_show	-0.017192	0.004119	-0.060319	0.029135	-0.035701	-0.015180	-0.000196	-0.006076	0.126431	1.000000	-0.020886	0.001165
Month	0.769393	0.006051	0.014547	-0.002588	0.003779	0.003741	0.003920	-0.001479	0.108070	-0.020886	1.000000	-0.062496
WeekDay	-0.038182	-0.003916	0.003088	-0.000673	0.003455	0.006614	0.002701	0.004352	-0.089858	0.001165	-0.062496	1.000000

Looking at the correlations between No-show appointments and all of the types of observations resulted in the following: There is a 20% rate of no-show appointments across all appointments. The percentage of appointments missed by females and those missed by males was the same, with females missing 20.31% of all appointments scheduled by women and males missing 19.97% of all appointments scheduled by males. This includes patients at all ages.

```
In [86]: ## Number of show (0) vs. no-show (1) appointments
         df.No_show.value_counts()
```

```
Out[86]: 0    88208
         1    22319
         Name: No_show, dtype: int64
```

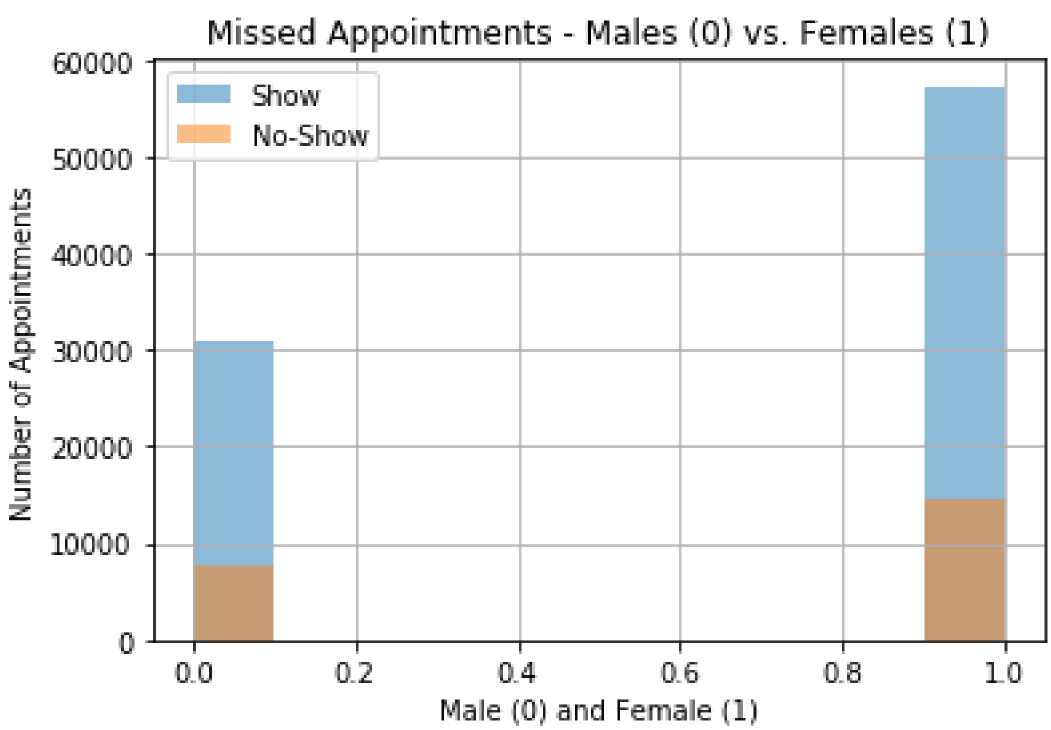
```
In [87]: ## Overall percentage of no-show appointments
         oa = df.No_show[no_show].count() / df.No_show.count() * 100
         oa
```

```
Out[87]: 20.193255946510806
```

```
In [88]: ## Number of females (1) and males (0) with appointments in this dataset
         df.Gender.value_counts()
```

```
Out[88]: 1    71840
         0    38687
         Name: Gender, dtype: int64
```

```
In [89]: ## Visualization of gender data
         df.Gender[show].hist(alpha = 0.5, label = 'Show')
         df.Gender[no_show].hist(alpha = 0.5,label = 'No-Show')
         plt.title('Missed Appointments - Males (0) vs. Females (1)')
         plt.xlabel('Male (0) and Female (1)')
         plt.ylabel('Number of Appointments')
         plt.legend();
```



The visualization shows that females schedule more appointments than males and also miss more appointments than males. A closer look at the data will give a more complete picture as to whether or not gender should be a consideration when determining how to keep patients from missing appointments.

```
In [90]: ## Create variables for gender data
         male = df.Gender == 0
         female = df.Gender == 1
```

```
In [91]: ## Overall numbers of females (1) and males (0) with missed appointments
         df.Gender[no_show].value_counts()
```

```
Out[91]: 1    14594
         0     7725
         Name: Gender, dtype: int64
```

```
In [92]: ## Percent of female no-shows across all females
         female_no_show = 14594/71840
         female_no_show
```

```
Out[92]: 0.20314587973273943
```

```
In [93]: ## Percent of male no-shows across all females
         male_no_show = 7725/38687
         male_no_show
```

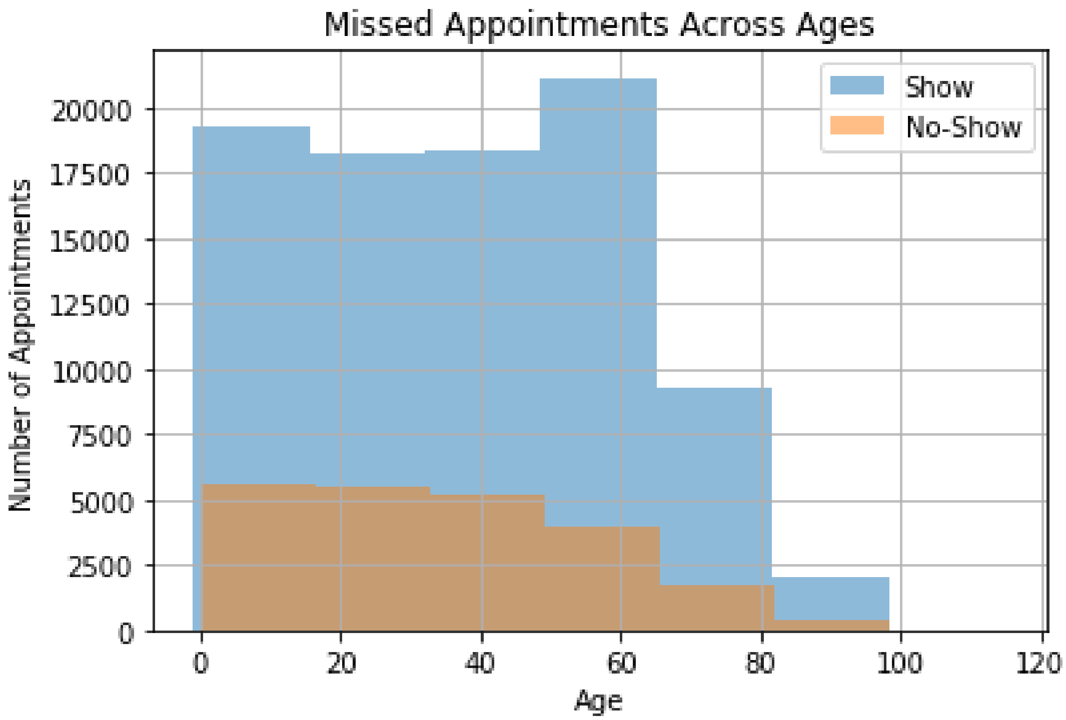
```
Out[93]: 0.19967947889471915
```

```
In [94]: df.Gender[no_show].value_counts() / df.Gender.count() * 100
```

```
Out[94]: 1    13.204013
         0     6.989242
         Name: Gender, dtype: float64
```

A graph of the age data shows that most missed appointments are for ages birth through about 30-years-old, where the no-show rate begins to taper slightly. There is another drop in missed appointments at age 50 and a significant drop about age 65. This could account for the increase in medical issues with age or a more serious approach to health with aging. Additionally, younger adults have more responsibilities and less personal time, possibly accounting for some of the no-show appointments.

```
In [95]: ## Plot of missed appointments across ages
         df.Age[show].hist(alpha = 0.5, bins = 7, label = 'Show')
         df.Age[no_show].hist(alpha = 0.5, bins = 7, label = 'No-Show')
         plt.title('Missed Appointments Across Ages')
         plt.xlabel('Age')
         plt.ylabel('Number of Appointments')
         plt.legend();
```




```
In [99]: ##Create dataframe to look at only no-show data
         df2 = df.query("No_show == '1'")
         df2.head()
```

Out[99]:

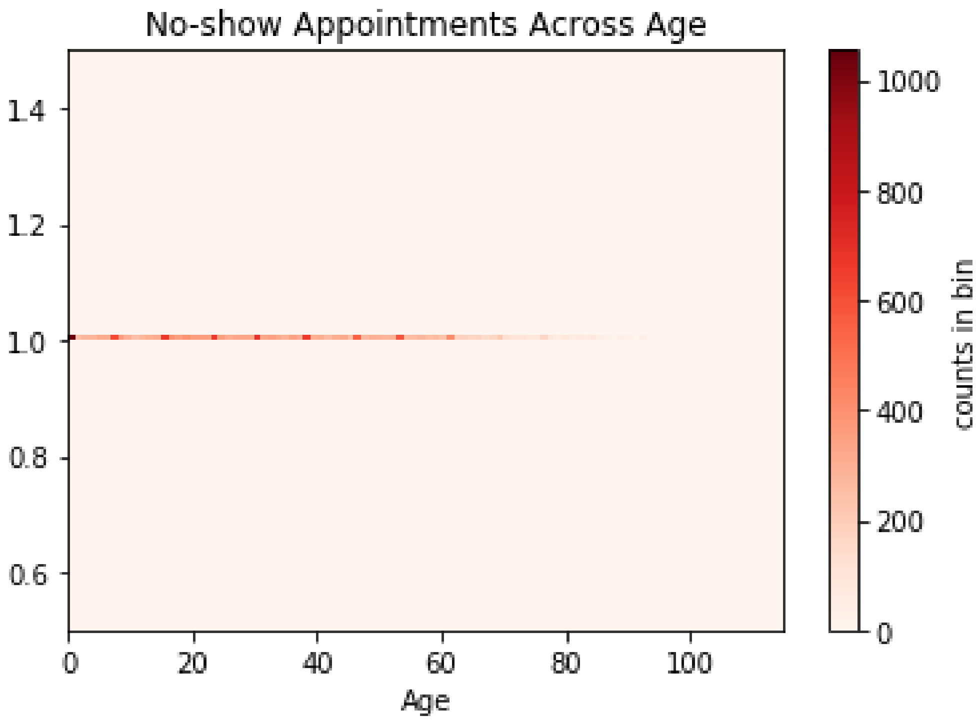
	Row_num	Gender	AppointmentDay	Age	Neighbourhood	Scholarship
6	7	1	2016-04-29	23	GOIABEIRAS	0
7	8	1	2016-04-29	39	GOIABEIRAS	0
11	12	0	2016-04-29	29	NOVA PALESTINA	0
17	18	1	2016-04-29	40	CONQUISTA	1
20	21	1	2016-04-29	30	NOVA PALESTINA	0

	Hipertension	Diabetes	Alcoholism	Handcap	SMS_received	No_show	Month
6	0	0	0	0	0	1	4
7	0	0	0	0	0	1	4
11	0	0	0	0	1	1	4
17	0	0	0	0	0	1	4
20	0	0	0	0	0	1	4

	WeekDay
6	4
7	4
11	4
17	4
20	4

The plot below gives a visualization of the decrease in no-show appointments with age. The darker the color, the more no-show appointments.

```
In [100]: ## Plot age visualization
          plt.hist2d(df2.Age, df2.No_show, bins = 100, cmap = 'Reds')
          cb = plt.colorbar()
          cb.set_label('counts in bin')
          plt.title('No-show Appointments Across Age')
          plt.xlabel('Age');
```



The percentages of no-show appointments for patients with serious chronic medical conditions such as alcoholism, diabetes, and handicaps is low, although diabetic patients account for a more than 6% rate of missed appointments. Patients with hypertension miss appointments at a rate of 16.9%. This could possibly be explained by the rate of hypertension in the population. Patients on 'scholarship' miss about 11.5% of their appointments. Economic factors such as inability to pay, inability to leave work, or lack of transportation may explain a portion of these missed appointments.

A quick look at the locations of the appointments indicate that location may play a part in patients missing appointments. However, there is not sufficient data to research a trend.

```
In [103]: ## Percentage appointments missed by patients with alcohol dependence
          alc= df.Alcoholism[no_show].value_counts() / df.Alcoholism[no_show].count() * 100
          alc

Out[103]: 0    96.96671
          1     3.03329
          Name: Alcoholism, dtype: float64
```

```
In [104]: ## Appointments missed by patients with diabetes
          db = df.Diabetes[no_show].value_counts() / df.Diabetes[no_show].count() * 100
          db

Out[104]: 0    93.592903
          1     6.407097
          Name: Diabetes, dtype: float64

In [105]: ## Appointments missed by patients who require financial assistance
          df.Scholarship[no_show].value_counts() / df.Scholarship[no_show].count() * 100

Out[105]: 0    88.449303
          1    11.550697
          Name: Scholarship, dtype: float64

In [106]: ## Counts of patients who show for or miss appointments based on location of the appoi
          df.groupby('Neighbourhood')['No_show'].value_counts()

Out[106]: Neighbourhood        No_show
          AEROPORTO            0             7
                               1             1
          ANDORINHAS           0          1741
                               1           521
          ANTÔNIO HONÓRIO      0           221
                               1            50
          ARIOVALDO FAVALESSA  0           220
                               1            62
          BARRO VERMELHO       0           332
                               1            91
          BELA VISTA           0          1523
                               1           384
          BENTO FERREIRA       0           665
                               1           193
          BOA VISTA            0           254
                               1            58
          BONFIM               0          2223
                               1           550
          CARATOÍRA            0          1974
                               1           591
          CENTRO               0          2631
                               1           703
          COMDUSA              0           254
                               1            56
          CONQUISTA            0           689
                               1           160
          CONSOLAÇÃO           0          1139
                               1           237
          CRUZAMENTO           0          1094
                               1           304
```

```
                                 ...
SANTA MARTHA         0          2635
                     1           496
SANTA TEREZA         0          1060
                     1           272
SANTO ANDRÉ          0          2063
                     1           508
SANTO ANTÔNIO        0          2262
                     1           484
SANTOS DUMONT        0           907
                     1           369
SANTOS REIS          0           435
                     1           112
SEGURANÇA DO LAR     0           117
                     1            28
SOLON BORGES         0           400
                     1            69
SÃO BENEDITO         0          1152
                     1           287
SÃO CRISTÓVÃO        0          1473
                     1           363
SÃO JOSÉ             0          1549
                     1           428
SÃO PEDRO            0          1933
                     1           515
TABUAZEIRO           0          2559
                     1           573
UNIVERSITÁRIO        0           120
                     1            32
VILA RUBIM           0           710
                     1           141
Name: No_show, Length: 160, dtype: int64
```

In [107]:
```
##Missed appointments by patients who have hypertension
hyp = df.Hipertension[no_show].value_counts() / df.Hipertension[no_show].count() * 100
hyp
```

Out[107]:
```
0    83.099601
1    16.900399
Name: Hipertension, dtype: float64
```

In [108]:
```
## Missed appointments by patients who have a handicap
hcap = df.Handcap[no_show].value_counts() / df.Handcap[no_show].count() * 100
hcap
```

Out[108]:
```
0    98.176442
1     1.639858
2     0.165778
3     0.013441
```

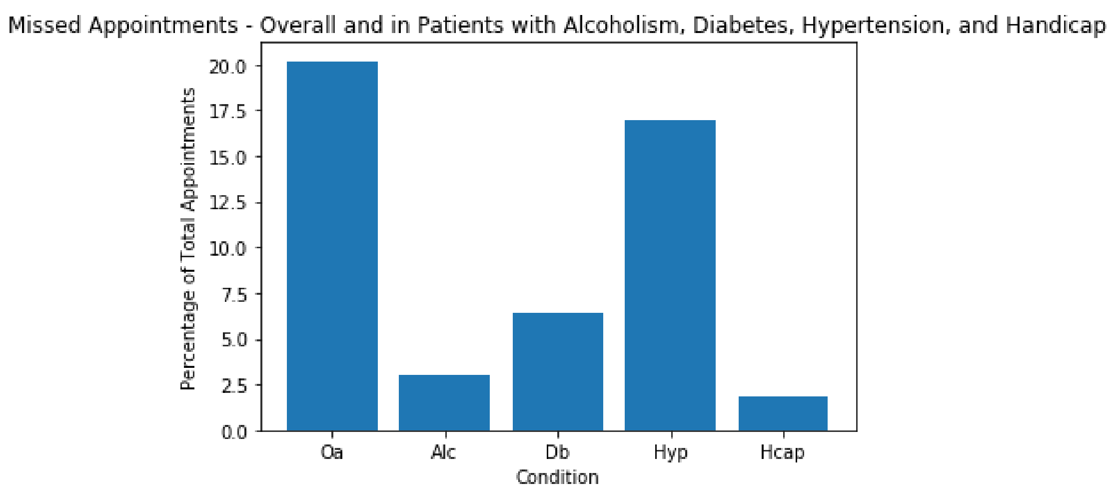
```
4     0.004480
Name: Handcap, dtype: float64
```

In [109]:
```
df.Hipertension.value_counts()/df.Row_num.count() * 100
```

Out[109]:
```
0    80.275408
1    19.724592
Name: Hipertension, dtype: float64
```

The plot below shows the percentage of patients who missed appointments in this dataset with the patients who missed appointments and also have the chronic conditions alcoholism, diabetes, hypertension, and handicap. Patients who have alcoholism, diabetes, and a handicap condition are much less likely to miss appointments than are patients who do not have these conditions. The percentage of patients who missed appointments and have hypertension is 16.9%. Overall, 19.7% of the population of this dataset have the condition hypertension.

In [110]:
```
##Plot percentage of missed appointments in patients with conditions
plt.bar([1,2,3,4,5], [20.19, 3.03, 6.41, 16.90, 1.82]);
plt.xticks([1,2,3,4,5], ['Oa', 'Alc', 'Db', 'Hyp', 'Hcap']);
plt.title('Missed Appointments - Overall and in Patients with Alcoholism, Diabetes, Hy
plt.xlabel('Condition')
plt.ylabel('Percentage of Total Appointments');
```



The most unexpected value in the dataset is the 43.8% of missed appointments for patients who received a text message. Healthcare providers are increasingly using these types of messages to remind patients of their appointments, often multiple times prior to the date of service.

The data from the date of appointment is as expected, with appointments missed mostly on Mondays, Tuesdays, and Sundays, respectively. The data for the month of service is calculated. However, because appointment data was only reported for three months of the year, a conclusion cannot be drawn from it.

In [111]:
```
## Missed appointments by patients who received test message appointment reminders
df.SMS_received[no_show].value_counts() / df.SMS_received[no_show].count() * 100
```

```
Out[111]: 0    56.162911
          1    43.837089
          Name: SMS_received, dtype: float64
```

```
In [112]: ## Count of patients who received test message reminders
          df.SMS_received.value_counts()
```

```
Out[112]: 0    75045
          1    35482
          Name: SMS_received, dtype: int64
```

```
In [113]: ## Percentage of no_show appointments by month
          df.Month[no_show].value_counts() / df.Month[no_show].count() * 100
```

```
Out[113]: 5    75.290112
          6    21.873740
          4     2.836149
          Name: Month, dtype: float64
```

```
In [114]: ## Counts of number of appointments by month
          df.Month.value_counts()
```

```
Out[114]: 5    80841
          6    26451
          4     3235
          Name: Month, dtype: int64
```

```
In [115]: ## Percentage of missed appointments by day of week with 0 = Sunday
          df.WeekDay[no_show].value_counts() / df.WeekDay[no_show].count() * 100
```

```
Out[115]: 1    23.083471
          2    22.819123
          0    21.013486
          4    18.087728
          3    14.955867
          5     0.040324
          Name: WeekDay, dtype: float64
```

Conclusions

A comprehensive look at the data included in this dataset indicates that the factors most affecting patient compliance with appointments are age, income level, hypertension, and day of the week. Inititially, it appeared that gender was a significant factor in whether or not patients kept their appointments because of the greater number of appointments missed by females over males. A closer look at the data shows that females and males miss appointments at about the same rate (20.31% and 19.97%), but that females schedule a greater number of appointments.

Patients with serious chronic health problems are not very likely to miss appointments. One exception is patients with hypertension. Hypertension is the medical term for abnormally high blood pressure. It can be caused by stress. Individuals who have hypertension may be overwhelmed and may not be able to make time for medical appointments or may feel that other responsibilities are more important, ie, not losing a portion of a paycheck to attend a medical appointment.

More than 10% of low-income patients miss appointments.

Most missed appointments, around 67%, are missed at the beginning of the week.

Of note is the fact that test message reminders do not improve the attendance rate of patients. Only 56% of patients who received reminders kept their appointments. The no-show rate for patients who receive reminders is more than double the amount of no-show appointments in the dataset. This may be an indication that attempts at changing patient behavior may be less successful in reducing no-show appointments than changing the way healthcare services are delivered to meet the needs of patients.

While this analysis of missed appointment data is limited in scope, it offers some information regarding where improvements can be made in health services. Initiatives could include offering appointments in different locations, offering services at lower cost in some areas, and encouraging employers to allow paid personal time for health appointments.

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

Resources used or viewed: Course materials

Stack Overflow

Python/Numpy/Pandas/Matplotlib Libraries

Matplotlib Cookbook

NIH study on missed appointments for reference and background.

Information on Kaggle regarding dataset.

In [ ]: