Missing Medical Appointments

in this analysis we will explore the reasons behind people not showing up for their medical appointments by examining the data set provided by various medical facilities in Brazil, Rio De Janeiro.

First step : loading and cleaning the data

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [2]:
```
noshowappointments_df = pd.read_csv(r"C:\Users\SARA\noshowappointments-kagglev2-may-2016.csv")
```

the original data included a negative number in the age column, assuming it's an entry error it is corrected below.

In [3]:
```
mask = noshowappointments_df['Age'] < 0
noshowappointments_df.ix[mask, 'Age'] = noshowappointments_df.ix[mask, 'Age'] * (-1)
```

changing the data type of the "No-show" column to boolean values for easier handling.

In [4]:
```
def show_or_noshow(string):
    if string == "Yes" or string =="yes":
        return True
    elif string == "No" or string == "no":
        return False
    else:
        return None

new_Noshow = noshowappointments_df["No-show"].apply(show_or_noshow)
noshowappointments_df["No-show"] = new_Noshow
```

dropping columns that will not be used in this analysis

In [5]:
```
noshowappointments_df = noshowappointments_df.drop(noshowappointments_df.columns[[0,1,3,6,7,8,
9,10,11,12]], axis=1,inplace=False)
```

In [6]:
```
noshowappointments_df.head()
```

Out[6]:

	Gender	AppointmentDay	Age	No-show
0	F	2016-04-29T00:00:00Z	62	False
1	M	2016-04-29T00:00:00Z	56	False
2	F	2016-04-29T00:00:00Z	62	False
3	F	2016-04-29T00:00:00Z	8	False
4	F	2016-04-29T00:00:00Z	56	False

changing the "AppointmentDay" column to just the day of the week on which the Appointment was scheduled.

In [7]:
```
from datetime import datetime as dt

def parse_date(date):
    if date == '':
        return None
    else:
        date = dt.strptime(date, '%Y-%m-%dT%H:%M:%SZ')
        return dt.strftime(date,"%a")


noshowappointments_df["AppointmentDay"]= noshowappointments_df["AppointmentDay"].apply(parse_da
te)
noshowappointments_df.columns=['Gender', 'Appointment Day', 'Age', 'No-show']
```

In [8]:
```
noshowappointments_df.groupby("Appointment Day").sum()
```

Out[8]:

	Age	No-show
Appointment Day		
Fri	704744	4037.0
Mon	836502	4690.0
Sat	2090	9.0
Thu	642282	3338.0
Tue	955330	5152.0
Wed	958376	5093.0

A General look at the data

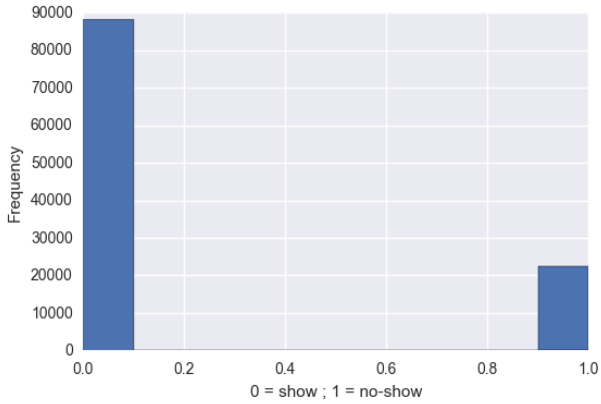
In [9]:
```
patient_count_byShow = noshowappointments_df.groupby("No-show").count()
patient_count_byShow
```

Out[9]:

	Gender	Appointment Day	Age
No-show			
False	88208	88208	88208
True	22319	22319	22319

In [19]:
```
noShow_dist_plot = noshowappointments_df["No-show"].plot(kind="hist")

noShow_dist_plot.set_xlabel("0 = show ; 1 = no-show")
```

Out[19]:
```
<matplotlib.text.Text at 0xfb40f60>
```



where 1 refers to True, that is, the no-show patients; and 0 refers to False, ie. the patients who did show up.
It can be observed that approximately 20% of all patients fail to show up for their appointments

Does the gender of the patient play a role in missing an appointment?

since the count method will give the same value for all columns, choosing "Appointment Day" arbitraraly just to get the count of patients. Then caculating the percentage of patients who didn't show up to the total number of patients of that gender

In [11]:
```
gender_grouped = noshowappointments_df.groupby("Gender")
gender_grouped.groups
female_patiencount =gender_grouped.count()["Appointment Day"]["F"]
female_noshowsum = gender_grouped.sum()["No-show"]["F"]
female_noshowperc = female_noshowsum/female_patiencount
female_noshowperc
```

Out[11]:
```
0.20314587973273943
```

In [12]:
```
male_patiencount =gender_grouped.count()["Appointment Day"]["M"]
male_noshowsum = gender_grouped.sum()["No-show"]["M"]
male_noshowperc = male_noshowsum/male_patiencount
male_noshowperc
```

Out[12]:
```
0.19967947889471915
```

about 20% of both male and female patients do not show up to their appointments; suggesting no correlation between the gender of the patient and the likelihood of missing an appointment.

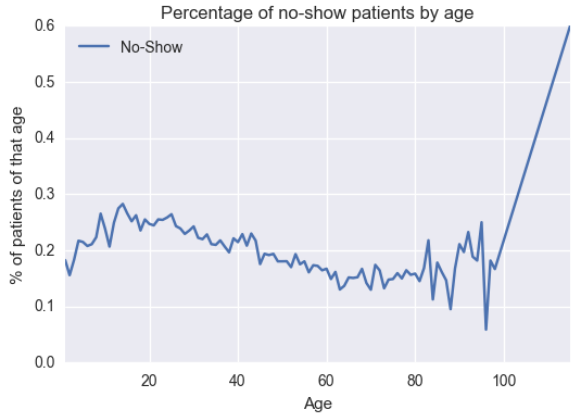
Do people tend to miss more or less appointments as they get older?

In [11]:
```
age_total_count = noshowappointments_df.groupby("Age").count()["Gender"]
```

by multiplying by the "No-show" column we are left with the rows that correspond to the True value. the rows that correspond to False will equal 0 and are deleted from the data frame afterwards.

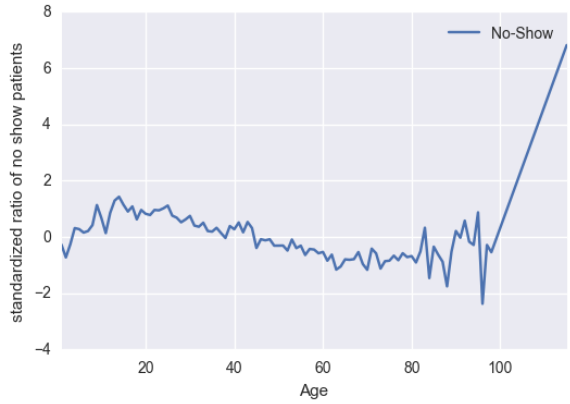
In [18]:
```
age_True = noshowappointments_df["Age"]*noshowappointments_df["No-show"]
age_No_show = age_True.to_frame(name = "Age; no show")

age_No_show = age_No_show[age_No_show["Age; no show"] != 0]
age_No_show = age_No_show.join(noshowappointments_df["Gender"])
age_No_show = age_No_show.groupby("Age; no show").count()
age_No_show = age_No_show["Gender"]/age_total_count
age_No_show = age_No_show.dropna()
age_No_show =age_No_show.to_frame()
age_No_show.columns=["No-Show"]
ageNoshow_plot = age_No_show.plot(title = "Percentage of no-show patients by age")
ageNoshow_plot.set_xlabel( "Age")
ageNoshow_plot.set_ylabel("% of patients of that age")
```

Out[18]:
```
<matplotlib.text.Text at 0xdce3f98>
```



In [23]:
```
age_std_plot = (age_No_show-age_No_show.mean())/age_No_show.std(ddof=0)
stdAgeNoShow_plot = age_std_plot.plot()
stdAgeNoShow_plot.set_xlabel("Age")
stdAgeNoShow_plot.set_ylabel("standardized ratio of no show patients")
```

Out[23]:
```
<matplotlib.text.Text at 0xfd00e10>
```



There seems to be a slight fluctioation of data along the age axis. the relationship does not seem to be linear.the ratio of no-show patients starts to decrease after around 20 years of age. of course the data after age 80 is not as reliable as there are fewer data points to rely on.

Does the no-show rate vary for different days of the week?

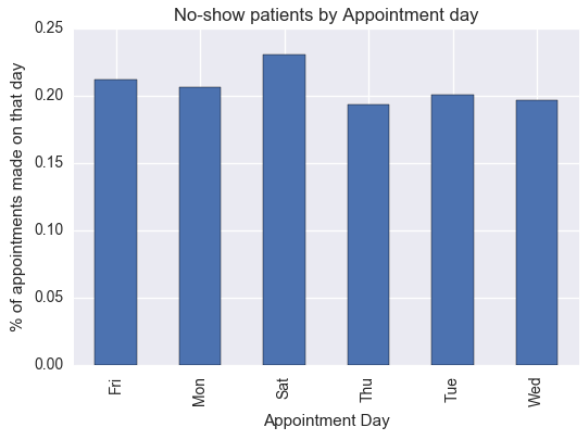
In [24]:
```
weekday_attendance = noshowappointments_df.groupby("Appointment Day")
weekday_attendance_df= weekday_attendance.sum()["No-show"]
```

In [25]:
```
total_patient_count = noshowappointments_df.groupby("Appointment Day").count()["No-show"]
total_patient_count
```

Out[25]:
```
Appointment Day
Fri    19019
Mon    22715
Sat       39
Thu    17247
Tue    25640
Wed    25867
Name: No-show, dtype: int64
```

In [26]:
```
weekday_plot = (weekday_attendance_df/total_patient_count).plot(kind = "bar", title = "No-show
patients by Appointment day")
weekday_plot.set_ylabel("% of appointments made on that day")
```

Out[26]:
```
<matplotlib.text.Text at 0xc4c2dd8>
```



There is seemingly no correlation between the day of week and the percentage of people missing thier appointments

Conclusion : Perhaps people just don't show up sometimes.

Of the 3 variables examined none of them seem to have a strong relationship with the rate of no-shows.
Although it seems that in all cases 20% of the patients do miss their appointments.

However it's important to note that the dataset had erroneous data points such patients over the age of 80 (only 5 patients 115 years old). As well as other dataset limitation, listed below:

- the data used spans over aproximately 1.5 months of patient appointment records, which might not have been enough.
- data was collected from only one city in Brazil.

Acknowledgement

recources that helped with the code used in this analysis:
links

https://www.tutorialspoint.com/python/time_strptime.htm
https://stackoverflow.com/questions/16766643/convert-date-string-to-day-of-week