My Final Project

Moataz Basyouni

February 2020

1 Abstract

This program can be utilized by hospitals to help in preparation for patients ahead of time. This can allow the hospital to use its resources most efficiently, and also mitigate the time for patients to receive treatment.

Contents

List of Figures

2 Introduction

This program uses data from a collection of product-related injury reports in the US from the National Electronic Injury Surveillance System (NEISS), which is a national probability sample of hospitals in the U.S. and its territories. The data includes patient information from each NEISS hospital for every emergency visit involving an injury associated with consumer products. The types of products involved in injuries include: floors, stairs, beds, basketball, bicycles, football, chairs, knives, bathtubs/showers, ceiling/walls.

This program is important because it gives a perspective on the frequencies of injuries caused by different products and the demographics of patients who were injured by them. By determining the demographics of those injured by these products, further studies can be conducted to find out why they are more likely to be injured by them, and how to prevent that from continuing to happen.

Talk about the purpose of the program and what it does. Talk about who can use it and its importance.

3 Methods

3.1 Python code

This is the code I created using Python. It can quickly sort through hospital data and provide important information such as the frequencies of injuries occurring at different times of the year, the most common injuries the patients were diagnosed with, the most common methods of injury, and the most common location at which patients receive their injuries.

```
#!/usr/bin/env python

def openfile(fname):
    import numpy
    import pandas as pd
    tmp_data = pd.read_csv(fname)
    data = tmp_data.to_numpy()
openfile('patientdata.csv')
#defines a function openfile so that whenever the function
#is called with the name of a file, it will open that file

def maxage(fname):
    import numpy
    import pandas as pd
    tmp_data = pd.read_csv(fname)
    data = tmp_data.to_numpy()
    age = data[:,4]
    age_max = numpy.max(age)
    print("The oldest patient is ", age_max, " years old.")
maxage('patientdata.csv')
#this function looks through the fourth column in the file
#and finds the max value. Sifting through the fourth column
#which includes the ages, and finding the max value will help
#find the oldest person in the list

print("\n")

def display(source, numlines = 3):
    print("Here is a preview of the file:")
    with open(source) as f:
        for i, line in enumerate(f):
            print(line.strip())
            if i == numlines:
                break
display('patientdata.csv')
#This function gives a preview of the file with the number of lines
#specified. If number of lines isn't specified, it will show a max
#of 3 lines.

print("\n")

def displaydict(source, numlines = 3):
    print("Here is a preview of the file in a dictionary")
    import csv
    with open(source) as f:
        reader = csv.DictReader(f)
        for i, row in enumerate(reader):
            print(dict(row))
            if i == numlines:
                break
displaydict('patientdata.csv')
#This gives a perview of the file in a dictionary.

print("\n")

def importgenderdata(source, output, gender):
    print("Here is a preview of the female patients only")
    import csv
    assert gender != "Female" or "Male", "Please ensure first letter of gender is capitalize
    with open (source) as fr:
        reader = csv.DictReader(fr)
        header = reader.fieldnames
        with open (output, "w") as fw:
            writer = csv.DictWriter(fw, fieldnames = header, delimiter = ",")
            for row in reader:
                if row["sex"] == gender:
                    writer.writerow(row)
importgenderdata('patientdata.csv', 'femaledata.csv', 'Female')
with open('patientdata.csv') as f:
    for i, line in enumerate(f):
        print(line.strip())
        if i == 3:
            break
#This function imports the female data only into a new file,
#it then previews some of it as done before
print("\n")

def hist(source):
    import numpy
    import pandas as pd
    tmp_data = pd.read_csv(source)
    data = tmp_data.to_numpy()
    import matplotlib.pyplot as plt
    ages = data[:,4]
    x = [ages]
    plt.style.use('ggplot')
    plt.hist(x, bins = 10)
hist('patientdata.csv')
#This functions uses data from column 4 (ages), and builds
#a histogram with it. It can be used as a visual representation
#of all the ages of the patients

print("\n")

def julydates():
    import csv
    import re
    with open("patientdata.csv") as fr:
        dictread = csv.DictReader(fr, delimiter = ',')
        header = dictread.fieldnames
        dates = []
        for row in dictread:
            dates.append(row['treatmentDate'])
    print('Here are the treatment dates which occured in July')
    date_regex = re.compile(r'7/\d*/\d{4}')
    uniqdates = set(dates)
    for date in uniqdates:
        if re.match(date_regex, date):
            print(date)

julydates()
#This function displays only the treatment dates which occured in July of 2015
```

3.2 R code

This is the code I created using R. The first one provides useful information about the different age distributions among patients with recorded injuries. The second code provides an insight on the patients' demographics.

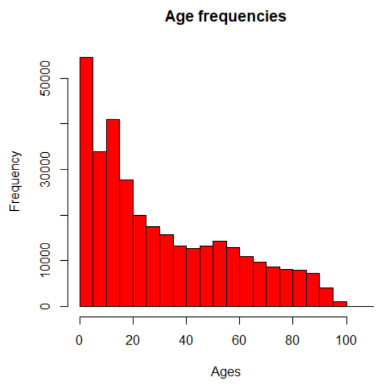
```
setwd("C:\\Users\\Motaz\\")
ptdata <- read.csv("nss15.csv")
hist(ptdata$age, main="Age frequencies", xlab="Ages", border="black",col="red")
#Opens the patient data file and uses it to create a histogram of the patients' ages.
This code also adds a title, axes labels, and colors to the histogram.

setwd("C:\\Users\\Motaz\\")
ptdata <- read.csv("nss15.csv")
ggplot(ptdata, aes(x=as.factor(race), fill=as.factor(race)))+
+ geom_bar()+scale_fill_manual(values=c("NA","red","orange","yellow","green","blue"))+
theme(legend.position="none")
#creates a bar graph based on the data of the patients' races.
This code also adds color to differentiate the bars for every race.
```

4 Results

Talk about the outputs of my program after inputting my dataset.

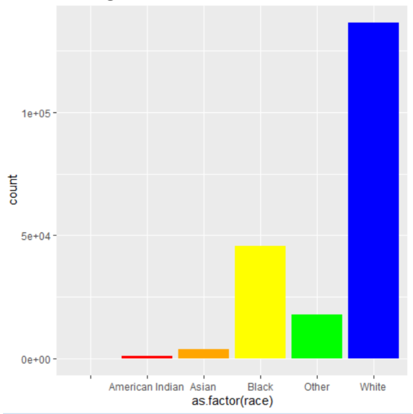
5 Figures

Figure 1: Ages histogram



Histogram of patients' ages.

Figure 2: Patient race



Histogram of patients' recorded race.

6 Discussion

Explain the results that the program produced and their purpose and how they can be used.

7 Conclusion

summarize every section

8 References

Reference stackoverflow and the other sources used