

Final Project

Esteban Mendoza

March 2020

1 Abstract

Contents

1	Abstract	1
2	Introduction	3
3	Materials & Methods	3
3.1	Script	3
4	Results	6
5	Discussion	6

2 Introduction

A disease is a condition that negatively affects the structure and hinders the homeostatic functions of an organism. Disease is characterized by specific symptoms exhibited by the affected organism. As medical technology has grown, so has the access to treatments that relieve the symptoms and repair the malfunctions caused by disease. Unfortunately, the costs to develop and employ these innovations have proven to be expensive and unaffordable for most people seeking treatment. In a society where resources to expend on healthcare are scarce, it is important to question how we can effectively allocate limited resources to maintain the health of the general public. Diseases can be categorized into different classifications, such as communicable and non-communicable. Additionally, these diseases can be categorized as infectious, deficiency, hereditary, and physiological diseases. However, assigning “disease” to a condition is a subjective topic. Studies have shown that different factors account for whether people believe themselves to be ill. Some of these factors include class, gender, ethnic group “and less obvious factors such as proximity to support from family members” [1]. Additionally, as expectations of health change throughout time, so does the classification of something as a disease.

For example, osteoporosis was “officially recognized as a disease by the WHO in 1994” [1]. This classification changed osteoporosis from a “normal part of aging” to a recognized pathological condition. Homosexuality has also had a history in the classification of disease. In the early 20th century, homosexuality was considered an endocrine disorder, then later classified as a mental disorder, and then finally “de-pathologized” in 1974. It’s important to properly identify a condition as a disease in order to properly allocate resources for treatment, while also being conscious of the weight and stigmatization that the label “disease” might carry.

In the analysis of this dataset, I attempt to identify what different members of society classify as a disease, and how much public funding should go into their management. The data used to inform this analysis was collected in a survey form. The survey was sent out to Finnish laypeople, doctors, nurses, and parliament members. The purpose of the survey was to collect opinions on different states of being and identify how these people classified them.

3 Materials & Methods

3.1 Script

The following script works from the shell. It asks the user to select a “state of being.” Then the user is asked to select a rank from 1-5. The rank they choose will produce a graph that highlights the percentage of people (doctors, nurses, laypeople, and parliament members) that considered the selected “state of being” as a disease (1-2: not a disease; 3: neutral; 4-5: yes a disease).

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

```

# coding: utf-8

# In[ ]:

#!/home/eebc177student/anaconda3/bin/python3
#written script will work anywhere in shell

import numpy as np
import pandas
import os
datapath = "/home/eebc177student/Developer/repos/eeb-c177-project/analyses"
directory = '/home/eebc177student/Developer/repos/eeb-c177-project/analyses'
os.chdir(directory)
#set working directory in analyses directory bc that's where the csv file is

import csv
import re
data = pandas.read_csv('final_data.csv')
data = data.rename(columns={'Restless_Legs_SyndromeA.1': 'Restless_Legs_SyndromeB'})
columns = list(data.columns) #turn the columns into a list
columns = columns[4:-1] #only use columns from these indexes bc they are the con

#A = is X a disease (rank from 1-5)
regex = re.compile(r'[\w\s]*[^A]A{1}$')
columnsA = list(filter(regex.match, columns))

#B = should public funding be used for X (rank 1-5)
regex = re.compile(r'[\w\s]*[^B]B{1}$')
columnsB = list(filter(regex.match, columns))

dis = str(input("what_disease_do_you_want_to_compare?:_")) #ask user what disease
dis = dis.upper() #make input uppercase bc column names uppercase
dis = dis.split()
error = False #for/if statement corrects for incorrect name input
for word in dis:
    if word in columnsA:
        pass
    else:
        error = True
if error: print('Please_enter_a_valid_condition')
else:
    def listtostring(s): #
        str1 = ""

```

```

        return(str1.join(s))
dis = listtostring(dis)
df = pandas.DataFrame(data)
df = df[['Group', dis]] #extract data columns for people surveyed, and disea

from collections import Counter

classify = str(input('Percent of participants that considered {} as a disease
#find percent of people who classified X as a disease on a scale from 1-5
#choose a number on the scale
error = False #for/if statement corrects for incorrect rank input
number = [str(i) for i in range(1,6)]
for rank in classify:
    if rank in number:
        pass
    else:
        error = True
if error: print('please enter a rank from 1-5')
else:
    classify = float(classify)

def profession(data, person):
    person_HArank = df.values.tolist() #make the people and classification
    HA5_people = person_HArank.count([person, classify]) #count list item
    person_HArank = [tuple(i) for i in person_HArank] #tuple instead of list
    counts = Counter(x[0] for x in person_HArank) #count number of tuple
    total_people = counts[person] #count all people in the survey
    percentage = HA5_people/total_people*100 #calculate percentage of people
    return percentage

#different groups of people from data
layperson = profession(data, 'Layperson')
nurse = profession(data, 'Nurse')
doctor = profession(data, 'Doctor')
parliament = profession(data, 'Parliament')

import matplotlib.pyplot as plt
##matplotlib inline
plt.style.use('ggplot')
#make a bar graph
def plot_percentage_person(layperson, doctor, nurse, parliament):
    #function to plot people and percentages
    x = ['Layperson', 'Doctor', 'Nurse', 'Parliament'] #people on the x axis
    percent = [layperson, doctor, nurse, parliament] #percentages to be plotted
    x_pos = [i for i, _ in enumerate(x)] #add groups of people

```

```

#bar graph settings
plt.bar(x_pos, percent, color='green')
plt.xlabel("Person")
plt.ylabel("Percent")
plt.title("Percent of professionals surveyed who classify {} as a

plt.xticks(x_pos, x)

#plot the graph
plt.show()
return
plot_percentage_person(layperson, doctor, nurse, parliament)
#use the function

```

4 Results

5 Discussion

References

- [1] Scully JL. What is a disease? *EMBO reports*, 5(7):650–653, 2004.