|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  | 20-12-2022**Project Report - Real Time Analytics** *Jathin Krishnan (117463)* |
|  | Guided by Prof Sebastian Zając |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | AIM To create a model and host the model on an app to predict the chance of cancer based on lifestyle data DATASETData used for creating model comes from <https://data.world/cancerdatahp/lung-cancer-data/workspace/file?filename=cancer+patient+level%20data+sets.xlsx>. Context  Many people's lives are cut short due to cancer. However, with big data we are able to combat this malicious disease through data driven predictions.  Content  In this dataset we will find info about hundreds of cancer patients about their lifestyles. | |  |
|  | Following Columns were taken from the dataset in order to create a model   1. Age 2. Dry Cough. 3. Smoking. 4. Balanced Diet 5. Alcohol 6. Obesity 7. Cancer Outcome |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | Approach | 1. Load and Check data 2. Create multiple models 3. Select the best model 4. Create an App and host it on local server 5. Allow user to input and let the model make the predictions for the new data |  |
|  | Code: *Cancer\_prediction\_Model.py* Intention of this file is   1. Load the data from Cancer.csv 2. Split it into training and test set (70/30 ratio) 3. Build multiple models 4. Check the stats of each model, select one with a best parameter, here ‘Accuracy’ was chosen 5. Load the model onto CancerPredictonModel.pkl   #!/usr/bin/env python import pandas as pd  from sklearn.model\_selection import train\_test\_split from sklearn.linear\_model import LogisticRegression from sklearn.metrics import classification\_report from sklearn.metrics import confusion\_matrix from sklearn.tree import DecisionTreeClassifier from sklearn.naive\_bayes import MultinomialNB import pickle filename='Cancer.csv' names =['Age','DryCough','Smoking','BalancedDiet','Alcohol','Obesity','Outcome'] dataframe = pd.read\_csv(filename,names = names) array = dataframe.values x=array[1:,:6] print(x) y=array[1:,6] print(y) test\_size=0.3 seed = 8 x\_train,x\_test,y\_train,y\_test = train\_test\_split(x ,y ,test\_size = test\_size,random\_state=seed) model =LogisticRegression(solver='liblinear') model.fit(x\_train,y\_train) result = model.score(x\_test,y\_test) print("Accuracy - Model 1: %.3f%%" % (result\*100.0)) from sklearn.svm import SVC model\_2 = SVC(kernel='linear') model\_2.fit(x\_train,y\_train) result = model\_2.score(x\_test,y\_test) print("Accuracy - Model 2: %.3f%%" % (result\*100.0)) model\_3 = DecisionTreeClassifier() model\_3.fit(x\_train,y\_train) result = model\_3.score(x\_test,y\_test) print("Accuracy - Model 3: %.3f%%" % (result\*100.0)) model\_4 = MultinomialNB() model\_4.fit(x\_train,y\_train) result = model\_4.score(x\_test,y\_test) print("Accuracy - Model 4: %.3f%%" % (result\*100.0)) #Rule out 4th model as accuracy is the least predicted\_1 = model.predict(x\_test) predicted\_2 = model\_2.predict(x\_test) predicted\_3 = model\_3.predict(x\_test) report1 = classification\_report(y\_test,predicted\_1) print(report1) report2 = classification\_report(y\_test,predicted\_2) print(report2) report3 = classification\_report(y\_test,predicted\_3) print(report3) matrix1 = confusion\_matrix(y\_test,predicted\_1) print(matrix1) matrix2 = confusion\_matrix(y\_test,predicted\_2) print(matrix2) matrix3 = confusion\_matrix(y\_test,predicted\_3) print(matrix3) with open('CancerPredictonModel.pkl','wb') as f:  pickle.dump(model\_3,f) with open ('CancerPredictonModel.pkl','rb') as f:  loaded\_model = pickle.load(f) test\_set =[[35,7,8,6,5,7]] print(loaded\_model.predict(test\_set)) test\_set =[[34,7,7,7,7,7]] print(loaded\_model.predict(test\_set)) Code: *app.py* Intention of this file is   1. Load the model 2. Allow User to input new data. Use streamlit package. Streamlit is an open-source Python library that makes it easy to create and share custom web apps for machine learning and data science. We can build and deploy powerful data apps quickly. 3. Feed the new data to model 4. Let the model predict the outcome on the chance of Cancer   import os import streamlit as st import pickle import numpy as np  mydir = 'C:/Users/jathi/PycharmProjects/RTA/Cancer-Prediction-Model-main' myfile = 'CancerPredictonModel.pkl' filename = os.path.join(mydir, myfile) model = pickle.load(open(filename,'rb'))  def predict\_diabetes(age,drycough,smoking,balanceddiet,alcohol,obesity):  input = np.array([[age,drycough,smoking,balanceddiet,alcohol,obesity]]).astype(np.float64)  prediction = model.predict(input)  return float(prediction)  def main():  #st.title("Cancer Prediction ")  html\_temp = """  <div style="background-color= #AEEEE ;padding:10px">  <h2 style="color:black;text-align:center;">Cancer Prediction Application</h2>  </div>  """  st.markdown(html\_temp, unsafe\_allow\_html=True)   age = st.text\_input("Age", " ")  drycough = st.text\_input("Dry Cough level", " ")  fatigue = st.text\_input("Smoking Level", " ")  balanceddiet = st.text\_input("Balanced Diet level", " ")  alcohol = st.text\_input("Alcohol Consumption", " ")  obesity = st.text\_input("Obesity", " ")  safe\_html = """  <div style="background-color:#EEEE3F;padding:10px >  <h2 style="color:white;text-align:center;">You are safe.Lead a healthy Lifestyle!!</h2>  </div>  """  danger\_html= """  <div style="background-color:#A00000;padding:10px >  <h2 style="color:white;text-align:center;">You are not safe,get checked OUT!!</h2>  </div>  """   if st.button("Predict"):  output=predict\_diabetes(age,drycough,fatigue,balanceddiet,alcohol,obesity)  if not output >0.5:  a ="Low probability for Cancer"  else:  a="High probability for Cancer"  st.success("{}".format(a))   if output <=0.7:  st.markdown(safe\_html,unsafe\_allow\_html=True)  else:  st.markdown(danger\_html, unsafe\_allow\_html=True)  if \_\_name\_\_ == '\_\_main\_\_':  main() | |  |

|  |  |  |
| --- | --- | --- |
|  | Results  1. App is hosted 2. New input is collected and loaded into model 3. Prediction is displayed |  |
|  |
| Git:  https://github.com/jk117463/RTA/tree/master/Cancer-Prediction-Model-main |