# -\*- coding: utf-8 -\*-

"""

Created on Mon Oct 24 16:46:02 2022

@author: vivas

"""

# Importamos las librerias que vamos a utilizar

import numpy as np

import matplotlib.pyplot as plt

import matplotlib as mpl

import pandas as pd

# Variables

# Funciones

def contador(bodypart):

    cont\_cabeza = cont\_espalda = cont\_abdomen = cont\_ala = cont\_frente = 0

    if bodypart == "head":

        cont\_cabeza += 1

        return cont\_cabeza, 0

    elif bodypart == "front":

        cont\_frente += 1

        return cont\_frente,1

    elif bodypart == "back":

        cont\_espalda += 1

        return cont\_espalda,2

    elif bodypart == "wing":

        cont\_ala += 1

        return cont\_ala,3

    elif bodypart == "abdomen":

        cont\_abdomen += 1

        return cont\_abdomen,4

    else:

        return 0, 0

def matriz(subjects,bodypart):

    dur = 0

    matrizb\_dur = []

    matriza\_dur = []

    c = 17

    f = len(subjects[:,0])

    col = []

    for i in range(1,c):

        fil = [0,0,0,0,0]

        matrizb\_dur=["","","","",""]

        for k in range(1,f):

            if subjects[k-1,0] == i:

                cont,posicion=(contador(bodypart[k-1,4].lower()))

                dur = bodypart[k-1,3]

                if cont!=0 or posicion!=0 or dur != 0:

                    fil[posicion]+=cont

                    matrizb\_dur[posicion] += (f" {dur} ;")

        col.append(fil)

        matriza\_dur.append(matrizb\_dur)

    return col,matriza\_dur

# Leemos la data

dataset = (pd.read\_csv('wCS10 Grooming -food 60-65 for ethogram.csv'))

# Limpiamos la data

#newdataset = newdataset.dropna()

newdataset = dataset.iloc[:, 2:].values

# Creamos un vector de unos

unos = np.ones((len(newdataset),), dtype=int)

# Creamos las tuplas para la duracion y los sujetos

duration = list(zip(newdataset[:,1], newdataset[:,3]))

subjects = list(zip(newdataset[:,0]-1,unos))

# Creamos un diccionario para asignar un color a una parte del cuerpo

dic = {

       'head' : '#FFC1C1',

       'back' : '#CDCDB4',

       'abdomen' : '#872657',

       'wing' : '#C76114',

       'front': '#CD0000',

       'blank': 'white'

       }

# Creamos el grafico

mpl.use('svg')

new\_rc\_params = {'text.usetex': False,

"svg.fonttype": 'none'

}

mpl.rcParams.update(new\_rc\_params)

fig, ax = plt.subplots()

# Utilizamos un ciclo para recorrer la data

for i in range(len(newdataset)):

    # Creamos un color por defecto en caso de que no consiga alguno en el diccionario

    color = 'black'

    # leemos las partes del cuerpo y las ponemos todas en minusculas

    bodypart = newdataset[i,4].lower()

    #Hacemos un condicional para conseguir el color segun la parte del cuerpo leida

    if bodypart in dic.keys():

         color = dic[bodypart]

    # graficampos

    ax.broken\_barh([duration[i]], subjects[i], facecolors=color)

# Organizamos la tabla

ax.set\_xlabel('Time in frames per second')

ax.set\_yticks([1,2,3,4,5,6,7,8,9,10

               ,11,12,13,14,15,16],

              labels=[

                  'Subject 1', 'Subject 2', 'Subject 3', 'Subject 4',

                  'Subject 5', 'Subject 6', 'Subject 7','Subject 8',

                  'Subject 9','Subject 10','Subject 11','Subject 12',

                  'Subject 13','Subject 14','Subject 15','Subject 16'])

# Exportar archivo

plt.show()

np.savetxt("data.csv", newdataset, delimiter=",",fmt='%s')

plt.savefig('wCS10 Grooming -food 150-155 for ethogram1.svg', format='svg')

# Calculamos cuantas veces hicieron las determinadas acciones

matriz\_boutcount, matriz\_duration = matriz(newdataset,newdataset)

orden1 = [['Head','Front','Back','Wing','abdomen']]

orden1.extend(matriz\_boutcount)

orden2 = [['Head','Front','Back','Wing','abdomen']]

orden2.extend(matriz\_duration)

orden1.extend(orden2)

# Exportamos el archivo

np.savetxt("data.csv", orden1 , delimiter=",",fmt='%s')