



NOMBRES:

NICOLÁS AÑAZCO

CARRERA:

INGENIERÍA EN SISTEMAS

MATERIA:

SISTEMAS EXPERTOS

FECHA:

22/05/2020

Como caso práctico se implementará un sistema CBR básico para determinar la calidad del vino rojo. Para ello, se trabajará con el corpus **Wine Quality Data Set**.

El corpus se compone de un total de 1599 muestras de vino rojo que contienen información de pruebas fisicoquímicas realizadas en vinos rojos

Parte de las muestras:

fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.8	0.88	0	2.6	0.098	25	67	0.9968	3.2	0.68	9.8	5
7.8	0.76	0.04	2.3	0.092	15	54	0.997	3.26	0.65	9.8	5
11.2	0.28	0.56	1.9	0.075	17	60	0.998	3.16	0.58	9.8	6
7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.4	0.66	0	1.8	0.075	13	40	0.9978	3.51	0.56	9.4	5
7.9	0.6	0.06	1.6	0.069	15	59	0.9964	3.3	0.46	9.4	5
7.3	0.65	0	1.2	0.065	15	21	0.9946	3.39	0.47	10	7
7.8	0.58	0.02	2	0.073	9	18	0.9968	3.36	0.57	9.5	7
7.5	0.5	0.36	6.1	0.071	17	102	0.9978	3.35	0.8	10.5	5
6.7	0.58	0.08	1.8	0.097	15	65	0.9959	3.28	0.54	9.2	5
7.5	0.5	0.36	6.1	0.071	17	102	0.9978	3.35	0.8	10.5	5
5.6	0.615	0	1.6	0.089	16	59	0.9943	3.58	0.52	9.9	5
7.8	0.61	0.29	1.6	0.114	9	29	0.9974	3.26	1.56	9.1	5
8.9	0.62	0.18	3.8	0.176	52	145	0.9986	3.16	0.88	9.2	5
8.9	0.62	0.19	3.9	0.17	51	148	0.9986	3.17	0.93	9.2	5
8.5	0.28	0.56	1.8	0.092	35	103	0.9969	3.3	0.75	10.5	7

Aplicar la técnica de los vecinos más cercanos indicada en clase y empleando la fórmula propuesta por wguillen.

El único resultado proporcionado por RBC será la calidad del vino, dependiendo de los parámetros de entrada.

El cálculo de similitud entre el caso buscado y los casos en la base de conocimiento se realiza utilizando la fórmula:

$$Similitud(A1C1, A12) = 1 - \frac{|A1C2 - A1C1|}{(val\ max - val\ min)}$$

```
def similarity(ce):
    valor=0
    for i in range(len(mini)):
        valor+= weight[i] * (1-((abs(ce[i]-cn[i]))/(maxi[i]-mini[i])))
    return valor/sum(weight)
```

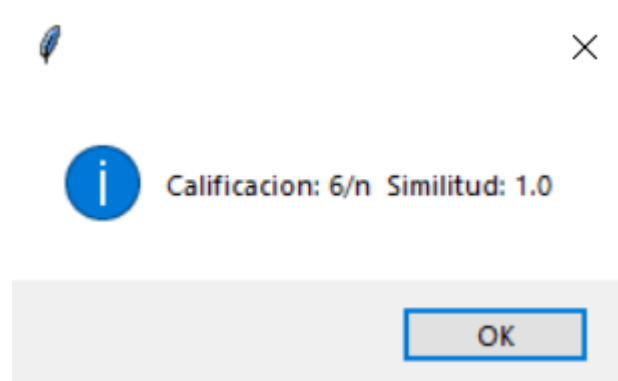
Desarrollar una pequeña interfaz en Python u otro lenguaje donde se coloquen los atributos y el sistema indique la calidad del vino.

ANALISIS DE VINOS

Acidez Fija	4.6	3	Acidez Volatil	0.12	3
Acido Citrico	0.0	3	Azucar residual	0.9	5
Choruros	0.012	1	Dioxido de Azufre Libre	1	1
Total Dióxido de azufre	6	1	Densidad	0.9900	1
PH	2.74	6	Sulfatos	0.33	1
Alcohol	8.4	5			

Procesar

RESULTADOS:



#Vino	Acidez F	Acidez \	Acido C	Azucar f	Chorurc	Dioxido	Total Di	Densitar	PH	Sulfatos	Alcohol	Califica	Similitud
1599	4.6	0.12	0.0	0.9	0.012	1.0	6.0	0.99	2.74	0.33	8.4	6	1.0
1600	4.8	0.12	0.1	0.9	0.012	1.0	6.0	0.99	2.74	0.33	14.9	6	0.822
1332	8.4	0.39	0.1	1.7	0.075	6.0	25.0	0.99581	3.09	0.43	9.7	6	0.81
1470	10.0	0.69	0.11	1.4	0.084	8.0	24.0	0.99577	2.88	0.47	9.7	5	0.809
439	7.0	0.62	0.18	1.5	0.062	7.0	50.0	0.9951	3.08	0.6	9.3	5	0.809
1420	7.8	0.53	0.01	1.6	0.077	3.0	19.0	0.995	3.16	0.46	9.8	5	0.806
1418	7.8	0.53	0.01	1.6	0.077	3.0	19.0	0.995	3.16	0.46	9.8	5	0.806
1392	7.1	0.62	0.06	1.3	0.07	5.0	12.0	0.9942	3.17	0.48	9.8	5	0.806
1338	6.0	0.5	0.0	1.4	0.057	15.0	26.0	0.99447	3.36	0.45	9.5	5	0.801
1337	6.0	0.5	0.0	1.4	0.057	15.0	26.0	0.99447	3.36	0.45	9.5	5	0.801

Codigo:

```
from tkinter import * # Carga módulo tk (widgets estándar)
from tkinter import ttk # Carga ttk (para widgets nuevos 8.5+)
from tkinter import messagebox
import pandas as pd
import operator
import csv
import os

ventana = Tk()

def analizar():
    newWindows=Tk()
    newWindows.title("Tabla de Similitud")
    df = pd.read_csv('winequality-red-Copy1.csv')
    vinos = [list(row) for row in df.values]
    similares={}

    cn=[float(tfa.get()),float(tva.get()),float(tca.get()),float(trs.get()),float(tc.get()),float(tfsd.get())
    ),
        float(ttsd.get()),float(td.get()),float(tph.get()),float(ts.get()),float(ta.get())]
    minimo=[4.6,0.12,0,0.9,0.012,1,6,0.99,2.74,0.33,8.4]
    maximo=[15.9,1.58,1.0,13.9,0.611,72.0,289.0,1.0,4.01,2.0,14.9]

    weight=[float(ctfa.get()),float(ctva.get()),float(ctca.get()),float(ctrs.get()),float(ctc.get()),float
    (ctfsd.get()),
        float(cttsd.get()),float(ctd.get()),float(ctph.get()),float(cts.get()),float(cta.get())]
    def similitud(ce):
        valor=0
        for i in range(len(minimo)):
            valor+= weight[i] * (1-((abs(ce[i]-cn[i]))/(maximo[i]-minimo[i])))
        return valor/sum(weight)

    for i in range(len(vinos)):
        fila=[]
        fila=vinos[i]
        x = similitud(fila)
        similares.update({str(i):round(x,3)})

    order = dict(sorted(similares.items(), key=operator.itemgetter(1)))
    cols=("#Vino", "Acidez Fija", "Acidez Volatil", "Acido Citrico", "Azucar
    Residual", "Choruros", "Dioxido de Azufre Libre", "Total Dióxido de
    azufre", "Densidad", "PH", "Sulfatos", "Alcohol", "Calificacion", "Similitud")
    tree = ttk.Treeview(newWindows,columns=cols,show='headings')
    vsb = ttk.Scrollbar(newWindows, orient="vertical", command=tree.yview)
```

```
vsb.pack(side=RIGHT, fill=BOTH)
```

```
tree.configure(yscrollcommand=vsb.set)
```

```
for i in range(len(cols)):
```

```
    tree.heading(cols[i],text=cols[i])
```

```
    tree.column(cols[i], minwidth=0, width=50)
```

```
tree.pack(expand=YES, fill=BOTH)
```

```
tam=len(orden)
```

```
for i in range(tam):
```

```
    pos=int(list(orden.items())[i][0])
```

```
    c1=vinos[int(pos)][0]
```

```
    c2=vinos[int(pos)][1]
```

```
    c3=vinos[int(pos)][2]
```

```
    c4=vinos[int(pos)][3]
```

```
    c5=vinos[int(pos)][4]
```

```
    c6=vinos[int(pos)][5]
```

```
    c7=vinos[int(pos)][6]
```

```
    c8=vinos[int(pos)][7]
```

```
    c9=vinos[int(pos)][8]
```

```
    c10=vinos[int(pos)][9]
```

```
    c11=vinos[int(pos)][10]
```

```
    c12=vinos[int(pos)][11]
```

```
    sim=str(list(orden.items())[i][1])
```

```
    tree.insert("",0,i,values=(str(pos),c1,c2,c3,c4,c5,c6,c7,c8,c9,c10,c11,c12,sim))
```

```
#print("Item Mas Similar")
```

```
fpos=list(orden.items())[tam-1][0]
```

```
fval=list(orden.items())[tam-1][1]
```

```
res=vinos[int(fpos)][11]
```

```
li=[cn[0],cn[1],cn[2],cn[3],cn[4],cn[5],cn[6],cn[7],cn[8],cn[9],cn[10],res]
```

```
if li in vinos:
```

```
    messagebox.showinfo(message="Calificacion: " + res + "/n "+" Similitud: "+str(fval))
```

```
else:
```

```
    with open('winequality-red-Copy1.csv','a') as f:
```

```
        writer = csv.writer(f)
```

```
        writer.writerow((li))
```

```
    messagebox.showinfo(message="Calificacion: " + res + "/n "+"Similitud: "+str(fval))
```

```
lista =[0,1,2,3,4,5,6,7,8,9,10]
```

```
#INTERFAZ DE LA COMPARACION
```

```
ventana.geometry('600x225') # anchura x altura
```

```
ventana.config(bg= "brown")
```

```
ventana.title('Calificador de Vinos')
```

```
Label(ventana,text="ANALISIS DE VINOS").place(x=200,y=0)
```

```
Label(ventana,text="Acidez Fija").place(x=0,y=25)
tfa = Spinbox(ventana, from_=4.6, to=15.9 , width=5,increment=0.1,font='Helvetica 12')
tfa.place(x=150,y=25)
ctfa = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
ctfa.place(x=225, y=25)
ctfa.current(3)
```

```
Label(ventana,text="Acidez Volatil").place(x= 308,y=25)
tva = Spinbox(ventana, from_=0.12, to=1.58 , width=5,increment=0.01,font='Helvetica 12')
tva.place(x=450 ,y=25)
ctva = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
ctva.place(x=525, y=25)
ctva.current(3)
```

```
Label(ventana,text="Acido Citrico").place(x=0,y=50)
tca = Spinbox(ventana, from_=0.0, to=1.0 , width=5,increment=0.1,font='Helvetica 12')
tca.place(x=150 ,y=50)
ctca = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
ctca.place(x=225, y=50)
ctca.current(3)
```

```
Label(ventana,text="Azucar residual").place(x= 308,y=50)
trs = Spinbox(ventana, from_=0.9, to=13.9 , width=5,increment=0.1,font='Helvetica 12')
trs.place(x=450 ,y=50)
ctrs = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
ctrs.place(x=525, y=50)
ctrs.current(5)
```

```
Label(ventana,text="Choruros").place(x=0,y=75)
tc = Spinbox(ventana, from_=0.012, to=0.611 , width=5,increment=0.001,font='Helvetica 12')
tc.place(x=150 ,y=75)
ctc = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
ctc.place(x=225, y=75)
ctc.current(1)
```

```
Label(ventana,text="Dioxido de Azufre Libre").place(x= 308,y=75)
tfds = Spinbox(ventana, from_=1.0, to=72.0 , width=5,increment=1.0,font='Helvetica 12')
tfds.place(x=450 ,y=75)
ctfsd = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
ctfsd.place(x=525, y=75)
ctfsd.current(1)
```

```
Label(ventana,text="Total Dióxido de azufre").place(x=0,y=100)
ttsd = Spinbox(ventana, from_=6.0, to=289.0 , width=5,increment=1,font='Helvetica 12')
ttsd.place(x=150 ,y=100)
```

```
cttsd = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
cttsd.place(x=225, y=100)
cttsd.current(1)
```

```
Label(ventana,text="Densidad").place(x= 308,y=100)
td= Spinbox(ventana, from_=0.9900, to=1.0000 , width=6,increment=0.0001,font='Helvetica 12')
td.place(x=450 ,y=100)
ctd = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
ctd.place(x=525, y=100)
ctd.current(1)
```

```
Label(ventana,text="PH").place(x=0,y=125)
tph = Spinbox(ventana, from_=2.74, to=4.01 , width=5,increment=0.01,font='Helvetica 12')
tph.place(x=150 ,y=125)
ctph = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
ctph.place(x=225, y=125)
ctph.current(6)
```

```
Label(ventana,text="Sulfatos").place(x= 308,y=125)
ts= Spinbox(ventana, from_=0.33, to=2.0 , width=5,increment=0.01,font='Helvetica 12')
ts.place(x=450 ,y=125)
cts = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
cts.place(x=525, y=125)
cts.current(1)
```

```
Label(ventana,text="Alcohol").place(x=0,y=150)
ta = Spinbox(ventana, from_=8.4, to=14.9 , width=5,increment=0.1,font='Helvetica 12')
ta.place(x=150 ,y=150)
cta = ttk.Combobox(ventana,values=lista,width=5,font='Helvetica 12')
cta.place(x=225, y=150)
cta.current(5)
```

```
ttk.Button(ventana, text='Procesar', command=analizar).place(x=275,y=200)
ventana.mainloop()
```