



Universidad Autónoma del Estado de México

Ingeniería en Computación

Materia: Redes Neuronales

Axel Valenzuela Juárez
25 de Septiembre del 2019

1. Clasificador Bayesiano

Antes de empezar a programar fue necesario comprender el algoritmo bayesiano para eso hice uso de los apuntes vistos en clase y las diapositivas subidas por el profesor. Para predecir la clase o categoria, se calculan todas las probabilidades de cada clase y se toma la mayor

$$prediccion = \max(P(A_i|B))$$

Para obtener la probabilidad de clase se debe calcular la proporción de cada clase.

$$P(clase = Ci) = (instanciasconclase = ci) / (totaldeinstancias)$$

La probabilidad condicional es la probabilidad de que ocurra cierto valor de una variable dada una clase.

$$P(v = vi | clase = Ci) = (InstanciasconV = vi \& Al mismo Tiempoclase = Ci) / InstanciaconClase = Ci$$

2. Codificación

El Primer paso en mi caso fue analizar el conjunto de datos otorgado, un archivo llamado datos.csv . En este caso el conjunto de datos se componia de 5 columnas y 9 filas de datos.

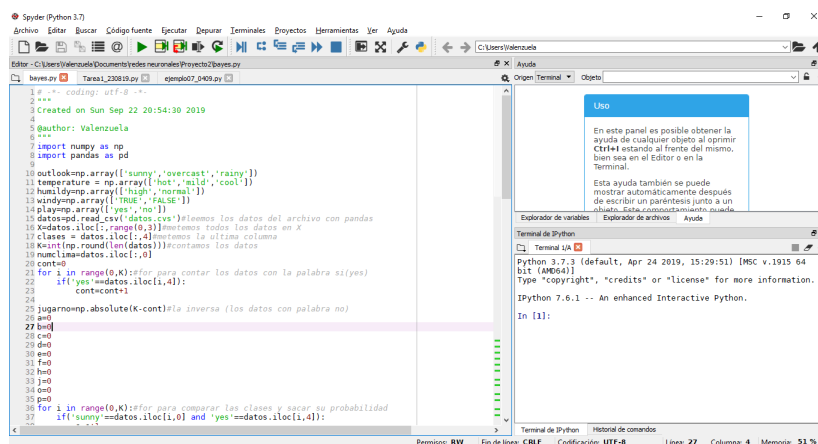


Figura 1: Primer fragmento de código.

Así que una de las primeras cosas a elaborar fue la importación de las bibliotecas Numpy y Pandas para tratar el manejo de archivos con la extensión .csv . Se ocupó el método `iloc` para meter todos los datos del archivo csv en una variable llamada `X`, posteriormente decidí meter la última columna de los datos en otra variable para así siempre tener los resultados a la mano, la variable que utilicé fue llamada `clases`. Decidí contar las filas de datos ya que gracias a eso sabría cuántas iteraciones hacer en mi código.

Realice un for para contar las veces que se repetía la palabra zes.^{esto con la finalidad de obtener la probabilidad de clase esto se realizó simplemente comparando con iloc y la palabra zes} todo esto dentro de un if.

```

33 j=0
34 c=0
35 p=0
36 for i in range(0,K):#for para comparar las clases y sacar su probabilidad
37     if('sunny'==datos.iloc[i,0] and 'yes'==datos.iloc[i,4]):
38         a=a+1
39     if('overcast'==datos.iloc[i,0] and 'yes'==datos.iloc[i,4]):
40         b=b+1
41     if('rainy'==datos.iloc[i,0] and 'yes'==datos.iloc[i,4]):
42         c=c+1
43     if('hot'==datos.iloc[i,1] and 'yes'==datos.iloc[i,4]):
44         d=d+1
45     if('mild'==datos.iloc[i,1] and 'yes'==datos.iloc[i,4]):
46         e=e+1
47     if('cool'==datos.iloc[i,1] and 'yes'==datos.iloc[i,4]):
48         f=f+1
49     if('high'==datos.iloc[i,2] and 'yes'==datos.iloc[i,4]):
50         h=h+1
51     if('normal'==datos.iloc[i,2] and 'yes'==datos.iloc[i,4]):
52         j=j+1
53     if(False==datos.iloc[i,3] and 'yes'==datos.iloc[i,4]):
54         o=o+1
55     if(True==datos.iloc[i,3] and 'yes'==datos.iloc[i,4]):
56         p=p+1
57     #diferentes probabilidades
58     precS=a/cont
59     precO=b/cont
60     precR=c/cont
61     precHot=d/cont
62     precH=e/cont
63     precC=f/cont
64     precH=h/cont
65     precN=j/cont
66     preco=o/cont
67     precp=p/cont
68     probclase=cont/K#la probabilidad de la clase play

```

Figura 2: Se muestra la comparacion de clases.

Una vez hecho esto procedí a comparar la probabilidad de cada una de las clases , en total 5, otra vez apoyándome de la ayuda de iloc y comparando con sus respectivos nombres, el resultado se dividió con el total de clases sacado anteriormente y así ya tenía lista la precisión de cada una de las clases, solo me hacia falta leer los nuevos datos y multiplicar las precisiones para obtener la precisión total.

Para leer los datos de prueba cree otro archivo llamado datosprueba.csv al igual que el anterior saque los datos gracias a iloc y obtuve los datos de la ultima clase, conté el número de filas y lo siguiente fue calcular la probabilidad de las clases para después poder multiplicarlas y obtener su valor.

```

70 datos2=pd.read_csv('datosprueba.csv')#se lee los datos a verificar
71 X=datos2.iloc[:,range(0,3)]
72 clases = datos2.iloc[:,4]
73 K1=int(np.round(len(datos2)))#contamos el numero de datos
74 for i in range(0,K1):#for para calcular la probabilidad
75     resultados=
76     r2=1
77     r3=1
78     r4=1
79     r5=1
80     var1 = datos2.iloc[i,0]
81     var2=datos2.iloc[i,1]
82     var3=datos2.iloc[i,2]
83     var4=datos2.iloc[i,3]
84     var5=datos2.iloc[i,4]
85     if(var1=='sunny'):#si encuentra sunny en la columna 0 mete l
86         resultado=precS
87     if(var1=='overcast'):#si encuentra overcast en la columna 6
88         resultado=precO
89     if(var1=='rainy'):#
90         resultado=precR
91     if(var2=='hot'):#
92         r2=precHot
93     if(var2=='mild'):#si encuentra mild en la columna 0 mete la
94         r2=precM
95     if(var2=='cool'):#
96         r2=precC
97
98     if(var3=='high'):#
99         r3=precH
100     if(var3=='normal'):#si encuentra normal en la columna 0 met
101         r3=precN
102
103     if(var4==False):
104         r4=preco
105     if(var4==True):
106         r4=precp

```

Figura 3: Comparacion de datos por medio de los arreglos.

Guardé en diferentes arreglos los datos por columna para ir comparando los datos por filas por medio de un for e if, simplemente comparando el valor de los arreglos por sus respectivos posibles valores.

Para finalizar multiplique todos los resultados de las probabilidades encontradas dando como resultado la probabilidad de salir a jugar o no salir.

3. Conclusion

El clasificador bayesiano es fácil de implementar y entender, al funcionar por medio de probabilidades podemos tener un muy buen entendimiento de este algoritmo, pudiendo calcular algunas series de datos para comprobar que el algoritmo funciona de manera correcta. En conclusión el clasificador bayesiano es muy bueno a la hora de predecir sucesos.

4. Referencias

http : //www.ai.org.mx/ai/images/sitio/2014/05/ingresos/less/trabajfinal_r.sucar.pdf Luis Enrique Sucar, Investigador Titular, Instituto Nacional de Astrofísica, Recuperado el 22 de Septiembre del 2019