# **Decision Tree-Fuzzy Decisions**

Krishna Kumar Veeraputhiran

Grand Canyon University

DSC 540 – O500 – Machine Learning for Data Science

Dr. Aiman Darwiche

September 8,2021

# **Decision Tree-Fuzzy Decisions**

In this report we will see how the decision tree we explained in the previous report can be converted to a Fuzzy Decision tree. Decision tree have a wider application in real world, however according to Gopal (2019), there are some glaring drawbacks in using them. They are highly unstable and with a minor change to the training dataset and also performance wise they are not great. Fuzzy decision trees on the other hand is able to fix these issues. We will see how the input attributes will be fuzzified in this document.

**Fuzzify Attributes**

The dataset that we will use here is the same dataset used in deriving the information gain. The attributes that we will fuzzify are Temperature (x1), Wind (x2) and Traffic-Jam (x3). We will first see how x1 can be fuzzified here. Fkx1 represent the fuzzy set for x1 where K1 = 2 with membership values; x1 = mild between x1=hot and x1=cool suitable values of μ1(x1(i)) and μ2(x1(i)) are assumed for x1 = mild. Below is the table depicting the membership value of x1.

|  |  |  |  |
| --- | --- | --- | --- |
| s(i) | x1 (Temperature) | μ1(x1(i)) (Hot) | μ2(x1(i)) (Cool) |
| s(1) | Hot | 1 | 0 |
| s(2) | Hot | 1 | 0 |
| s(3) | Hot | 1 | 0 |
| s(4) | Mild | 0.3 | 0.6 |
| s(5) | Cool | 0 | 1 |
| s(6) | Cool | 0 | 1 |
| s(7) | Cool | 0 | 1 |
| s(8) | Mild | 0.3 | 0.6 |
| s(9) | Cool | 0 | 1 |
| s(10) | Mild | 0.3 | 0.6 |
| s(11) | Mild | 0.3 | 0.6 |
| s(12) | Mild | 0.3 | 0.6 |
| s(13) | Hot | 1 | 0 |
| s(14) | Mild | 0.3 | 0.6 |

Fkx2 represent the fuzzy set for x2 where K2 = 2 with membership values; x2=weak and x2=strong suitable values of μ1(x2(i)) and μ2(x2(i)) are assumed. Below is the table depicting the membership value of x2.

|  |  |  |  |
| --- | --- | --- | --- |
| s(i) | x2 (Wind) | μ1(x2(i)) (Weak) | μ2(x2(i)) (Strong) |
| s(1) | Weak | 1 | 0 |
| s(2) | Strong | 0 | 1 |
| s(3) | Weak | 1 | 0 |
| s(4) | Weak | 1 | 0 |
| s(5) | Weak | 1 | 0 |
| s(6) | Strong | 0 | 1 |
| s(7) | Strong | 0 | 1 |
| s(8) | Weak | 1 | 0 |
| s(9) | Weak | 1 | 0 |
| s(10) | Weak | 1 | 0 |
| s(11) | Strong | 0 | 1 |
| s(12) | Strong | 0 | 1 |
| s(13) | Weak | 1 | 0 |
| s(14) | Strong | 0 | 1 |

Fkx3 represent the fuzzy set for x3 where K3 = 2 with membership values; x3=long and x3=short suitable values of μ1(x3(i)) and μ2(x3(i)) are assumed. Below is the table depicting the membership value of x3.

|  |  |  |  |
| --- | --- | --- | --- |
| s(i) | x3 (Traffic-Jam) | μ1(x3(i)) (Long) | μ2(x3(i)) (Short) |
| s(1) | Long | 1 | 0 |
| s(2) | Long | 1 | 0 |
| s(3) | Long | 1 | 0 |
| s(4) | Long | 1 | 0 |
| s(5) | Short | 0 | 1 |
| s(6) | Short | 0 | 1 |
| s(7) | Short | 0 | 1 |
| s(8) | Long | 1 | 0 |
| s(9) | Short | 0 | 1 |
| s(10) | Short | 0 | 1 |
| s(11) | Short | 0 | 1 |
| s(12) | Long | 1 | 0 |
| s(13) | Short | 0 | 1 |
| s(14) | Long | 1 | 0 |

The membership value of all three attributes are given below. Let us also assume the dataset is crisp, meaning μD(s(i)) =1 and hence all membership values are multiplied by 1.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| s(i) | μD(s(i)) | x1 | μ1(x1(i)) | μ2(x1(i)) | x2 | μ1(x2(i)) | μ2(x2(i)) | x3 | μ1(x3(i)) | μ2(x3(i)) |
| s(1) | 1 | Hot | 1 | 0 | Weak | 1 | 0 | Long | 1 | 0 |
| s(2) | 1 | Hot | 1 | 0 | Strong | 0 | 1 | Long | 1 | 0 |
| s(3) | 1 | Hot | 1 | 0 | Weak | 1 | 0 | Long | 1 | 0 |
| s(4) | 1 | Mild | 0.3 | 0.6 | Weak | 1 | 0 | Long | 1 | 0 |
| s(5) | 1 | Cool | 0 | 1 | Weak | 1 | 0 | Short | 0 | 1 |
| s(6) | 1 | Cool | 0 | 1 | Strong | 0 | 1 | Short | 0 | 1 |
| s(7) | 1 | Cool | 0 | 1 | Strong | 0 | 1 | Short | 0 | 1 |
| s(8) | 1 | Mild | 0.3 | 0.6 | Weak | 1 | 0 | Long | 1 | 0 |
| s(9) | 1 | Cool | 0 | 1 | Weak | 1 | 0 | Short | 0 | 1 |
| s(10) | 1 | Mild | 0.3 | 0.6 | Weak | 1 | 0 | Short | 0 | 1 |
| s(11) | 1 | Mild | 0.3 | 0.6 | Strong | 0 | 1 | Short | 0 | 1 |
| s(12) | 1 | Mild | 0.3 | 0.6 | Strong | 0 | 1 | Long | 1 | 0 |
| s(13) | 1 | Hot | 1 | 0 | Weak | 1 | 0 | Short | 0 | 1 |
| s(14) | 1 | Mild | 0.3 | 0.6 | Strong | 0 | 1 | Long | 1 | 0 |

**Fuzzy Entropy and Information Gain**

The βqk , membership concerning the qth class induced by the fuzzy set Fkxj can be calculated as how below in the following steps,

β1 = ∑14i=1  min { μD(s(i)), μ1(y(i))} / ∑14i=1  μD(s(i))

β1 = 0+0+1+1+1+0+1+0+1+1+1+1+1+0 / (1+1+1+1+1+1+1+1+1+1+1+1+1+1) = 9/14 = 0.643

β2 = 1+1+0+0+0+1+0+1+0+0+0+0+0+1/(1+1+1+1+1+1+1+1+1+1+1+1+1+1) = 5/14 = 0.357

Entropy(D) = - (β1log2 β1 + β2log2 β2) = -(0.643\* log2 (0.643) + 0.357\* log2 (0.357) )

= -(0.643\*(-0.637) + 0.357\*(-1.486))

= -( -0.41-0.53) = ***0.94***

*Entropy after partitioning D on x1*

β11 = ∑14i=1  min { μ1(x1(i)), μ1(y(i))} / ∑14i=1  μ1(x1(i))

= (0+0+1+0.3+0+0+0+0+0+0.3+0.3+0.3+1+0)/(1+1+1+0.3+0+0+0+0.3+0+0.3+0.3+0.3+1+0.3)

= 3.2/5.8 = ***0.552***

β 21 = (1+1+0+0+0+0+0+0.3+0+0+0+0+0+0.3)/(1+1+1+0.3+0+0+0+0.3+0+0.3+0.3+0.3+1+0.3)

=2.6/5.8 = ***0.448***

Entropy(D1) = - β11log2 β11 – β21log2 β21

= -0.552 log20.552 - 0.448 log20.448 = -0.552(-0.857) – 0.448(-1.158) =0.473+0.519 = ***0.992***

β12=(0+0+0+0.6+1+0+1+0+1+0.6+0.6+0.6+0+0)/(0+0+0+0.6+1+1+1+0.6+1+0.6+0.6+0.6+0+0.6) = 5.4/7.6 = ***0.71***

β22= (0+0+0+0+0+1+0+0.6+0+0+0+0+0+0.6)/(0+0+0+0.6+1+1+1+0.6+1+0.6+0.6+0.6+0+0.6)

= 2.2/7.6 = ***0.29***

Entropy(D2) = - β12log2 β12 – β22log2 β22

= -0.71 log20.71 - 0.29 log20.29 = -0.71(-0.494) – 0.29 (-1.786) = 0.351 + 0.518 = ***0.869***

w1 = ∑14i=1  μ1(x1(i)) / ∑2k=1  (∑14i=1  μk(x1(i))) = 5.8/ (5.8+7.6) = ***0.433***

w2 = 7.6/(5.8+7.6) = ***0.567***

***Entropy(D,x1) = w1 \* Entropy(D1) + w2\* Entropy(D2) = 0.433\*0.992 + 0.567\*0.869 = 0.92***

***Gain(D,x1) = Entropy(D) – Entropy(D,x1) = 0.94-0.92 = 0.02 🡪Eqn 1***

*Entropy after partitioning D on x2*

β11 = (0+0+1+1+1+0+0+0+1+1+0+0+1+0)/(1+0+1+1+1+0+0+1+1+1+0+0+1+0) = 6/8 = ***0.75***

β 21 = 2/8 = ***0.25***

Entropy(D1) = - β11log2 β11 – β21log2 β21

= -0.75 log2 0.75 – 0.25 log2 0.25 = -0.75(-0.415) – 0.25(-2) = 0.311+0.5 = ***0.811***

β12 = (0+0+0+0+0+0+1+0+0+0+1+1+0+))/(0+1+0+0+0+1+1+0+0+0+1+1+0+1)= 3/6 =***0.5***

β22= 3/6 = ***0.5***

Entropy(D2) = - β12log2 β12 – β22log2 β22

= -0.5 log20.5 -0.5 log20.5 =***1.0***

w1 = 8/(8+6) = **0.57**

w2 = 6/(8+6) = **0.43**

***Entropy(D,x2) = w1 \* Entropy(D1) + w2\* Entropy(D2) = 0.57\*0.811 + 0.43\*1= 0.89***

***Gain(D,x2) = Entropy(D) – Entropy(D,x2) =0.94 – 0.89 = 0.05 🡪Eqn 2***

*Entropy after partitioning D on x3*

β11 = (0+0+1+1+0+0+0+0+0+0+0+1+0+0)/(1+1+1+1+0+0+0+1+0+0+0+1+0+1) =3/7 = ***0.43***

β 21 = 4/7 = ***0.57***

Entropy(D1) = - β11log2 β11 – β21log2 β21

= -(0.43\* log20.43 +0.57 log20.57) = -(0.43\*(-1.217) + 0.57\*(-0.811)) = 0.523 + 0.462 = ***0.985***

β12 = (0+0+0+0+1+0+1+0+1+1+1+0+1+0)/(0+0+0+0+1+1+1+0+1+1+1+0+1+0) = 6/7 = ***0.86***

β22= 1/7 = 0.14

Entropy(D2) = - β12log2 β12 – β22log2 β22

= -0.86 log20.86 -0.14 log20.14 = -0.86 (-0.218) – 0.14(-2.84) = 0.187 + 0.397 = ***0.584***

w1 = 7/7+7 =0.5

w2 = 7/7+7 =0.5

***Entropy(D,x3) = w1 \* Entropy(D1) + w2\* Entropy(D2) = 0.5\*0.985 + 0.5\*0.584 = 0.785***

***Gain(D,x3) = Entropy(D) – Entropy(D,x3) =0.94 – 0.785 = 0.155 🡪 Eqn 3***

***From Eqn 1, 2 & 3 we can see that the Gain is greatest for x3 (Traffic-Jam) and hence the Root node need to be Traffic-Jam in our case.***

**Plot Fuzzy Decision Tree**

Long Short

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| μD1 | x1 | x2 | y |  | μD2 | x1 | x2 | y |
| 1 | Hot | Weak | No | 1 | Cool | Weak | Yes |
| 1 | Hot | Strong | No | 1 | Cool | Strong | No |
| 1 | Hot | Weak | Yes | 1 | Cool | Strong | Yes |
| 1 | Mild | Weak | Yes | 1 | Cool | Weak | Yes |
| 1 | Mild | Weak | No | 1 | Mild | Weak | Yes |
| 1 | Mild | Weak | Yes | 1 | Mild | Strong | Yes |
| 1 | Mild | Strong | No | 1 | Hot | Weak | Yes |

{s1,s2,s3,s4,s8,s12,s14} {s5,s6,s7,s9,s10,s11,s13}

Similarly, we can continue subsequent nodes to grow the Fuzzy Decision tree. The node based on which the split need to be made need to depend on the Gain of the attributes. Attributes with higher gain value is selected as a node for split.

# **References**

Gopal, M. (2019). *Applied machine learning*. McGraw-Hill Education.