**Summary**

**Pruning:**

Pruning is an important step in the post-processing of decision trees. Without pruning, a fully-grown decision tree would be overfit. It is necessary to remove some unnecessary nodes to make the decision tree model more generalizable

1. **Compute the entropy of conversion**

P(high)= p(medium)= p(low)=

E (conversion)=-((5/16)\*log2(5/16)+(5/16)\*log2(5/16)+(6/16)\*log(6/16))= 1.579434

1. **Compute the root node**

**2.1 Compute the information gain of demand**

E(demand=heavy)= -((3/7)\*log2(3/7)+(2/7)\*log2(2/7)+(2/7)\*log2(2/7))= 1.556657

E(demand=moderate)= -((1/4)\*log2(1/4)+(1/4)\*log2(1/4)+(2/4)\*log2(2/4))=1.5

E(demand=low)= -(0+(2/5)\*log2(2/5)+(3/5)\*log2(3/5))= 0.9709506

E(demand)= (7/16)\*1.556657+(4/16)\*1.5+(5/16)\*0.9709506= 1.359459

Gain(conversion,demand)= 1.579434-1.359459= 0.219975

**2.2 Compute the information gain of strategic**

E(Strategic=yes)= -((3/9)\*log2(3/9)+(3/9)\*log2(3/9)+(3/9)\*log2(3/9))= 1.584963

E(Strategic=no)= -((2/7)\*log2(2/7)+(2/7)\*log2(2/7)+(3/7)\*log2(3/7))= 1.556657

E(Strategic)= (9/16)\* 1.584963+(7/16)\* 1.556657=1.572579

Gain(conversion,strategic)= 1.579434- 1.572579=0.006855287

* 1. **Compute the information gain of campaign**

E(Campaign=aggressive)=-((4/9)\*log2(4/9)+(4/9)\*log2(4/9)+(1/9)\*log2(1/9))= 1.392147

E(Campaign=lowkey)=-((1/7)\*log2(1/7)+(1/7)\*log2(1/7)+(5/7)\*log2(5/7))= 1.148835

E(Campaign)= (9/16)\* 1.392147 +(7/16)\* 1.148835 =1.285698

Gain(conversion, Campaign)= 1.579434-1.285698=0.2937359

**Because the value of Gain(conversion, Campaign) is the biggest So root node should be campaign.**

1. **Compute the second node**

**3.1 Compute the information gain of demand**

When compaign=aggressive:

E(demand=heavy)= -((3/6)\*log2(3/6)+(2/6)\*log2(2/6)+(1/6)\*log2(1/6))= 1.459148

E(demand=moderate)= -((1/2)\*log2(1/2)+(1/2)\*log2(1/2)+0)=1

E(demand=low)= 0

E(demand)=(6/9)\*1.459148+(2/9)\*1+0= 1.0780281.194988

Gain(conversion,demand)= 1.579434-1.0780281.194988 =0.3844465

When compaign=lowkey:

E(demand=heavy)= - 1 \* log2(1)= 0

E(demand=moderate)= - 1/2 \* log2(1/2) - 1/2 \* log2(1/2) = 1

E(demand=low)= - 1/4 \* log2(1/4) - 3/4 \* log2(3/4) = 0.811

E(demand)= = 1/7 \* 0 + 2/7 \* 1+ 4/7 \* 0.811= 0.749

Gain(conversion,demand)= 1.579 – 0.749= 0.83

**3.2 Compute the information gain of strategic**

When compaign=aggressive:

E(Strategic=yes)= = -((3/7)\*log2(3/7)+(3/7)\*log2(3/7)+(1/7)\*log2(1/7))= 1.448816

E(Strategic=no)= = -((1/2)\*log2(1/2)+(1/2)\*log2(1/2)+0)=1

E(Strategic)= (7/9)\* 1.448816+(2/9)\*1= 1.349079

Gain(conversion,strategic)= 1.579434- 1.349079= 0.2303552

When compaign=lowkey:

E(Strategic=yes)= =- 1 \* log2(1)= 0

E(Strategic=no)= - 1/5 \* log2(1/5) - 1/5 \* log2(1/5) - 3/5 \* log2(3/5)= 1.371

E(Strategic)= 2/7 \* 0+ 5/7 \* 1.371= 0. 979

Gain(conversion,strategic)= = 1.579 – 0.979= 0.6

**Because the value of the information gain of demand is higher, so the second node is demand**