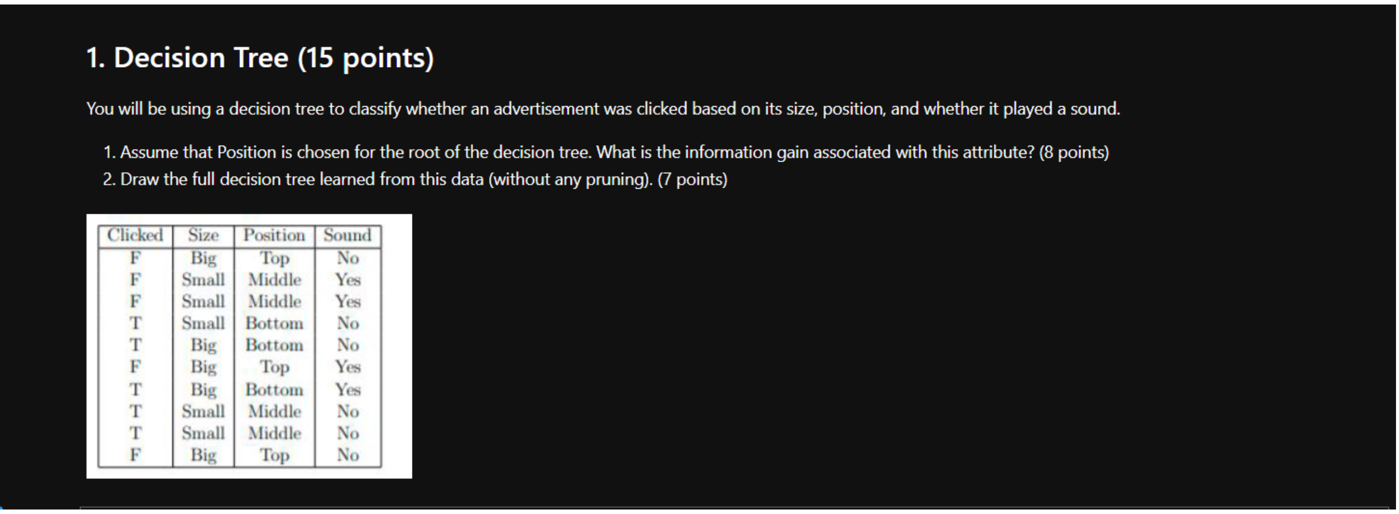
# Theoretical



## 1. Decision Tree (15 points)

You will be using a decision tree to classify whether an advertisement was clicked based on its size, position, and whether it played a sound.

1. Assume that Position is chosen for the root of the decision tree. What is the information gain associated with this attribute? (8 points)
2. Draw the full decision tree learned from this data (without any pruning). (7 points)

| Clicked | Size | Position | Sound |
|---|---|---|---|
| F | Big | Top | No |
| F | Small | Middle | Yes |
| F | Small | Middle | Yes |
| T | Small | Bottom | No |
| T | Big | Bottom | No |
| F | Big | Top | Yes |
| T | Big | Bottom | Yes |
| T | Small | Middle | No |
| T | Small | Middle | No |
| F | Big | Top | No |

1.

$$let\ E(s) \coloneqq Entropy(S)$$
$$Values(A = Position) = \{'Top', 'Middle', Bottom'\}$$
$$S = [5^+, 5^-]$$

− − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −

$$S_{top} = [0^+, 3^-]\ ,\quad |S_{top}| = 3$$
$$S_{Middle} = [2^+, 2^-],\qquad |S_{Middle}| = 4$$
$$S_{Bottom} = [3^+, 3^-],\quad |S_{Bottom}| = 3$$

− − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −

$$Gain(S, A = Potision) = E(s) - \sum_{v \in \{'Top', 'Middle', 'Bottom'\}} \frac{|S_v|}{|S|} E(S_v)$$
$$E(s) = -\frac{1}{2}\log\left(\frac{1}{2}\right) - \frac{1}{2}\log\left(\frac{1}{2}\right) = 1$$
$$E(S_{Top}) = E(S_{Bottom}) = 0$$
$$E(S_{Middle}) = -\frac{1}{2}\log\left(\frac{1}{2}\right) - \frac{1}{2}\log\left(\frac{1}{2}\right) = 1$$
$$Gain(S, A = Position) = 1 - \left[\frac{3}{10} * 0 + \frac{4}{10} * 1 + \frac{3}{10} * 0\right] = \frac{3}{5}$$
$$Gain(S, A = Position) = \frac{3}{5}$$

∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼

*2.*

*First, we'll find the attribute that his information gain is the maximum in order to place him as the Root*

− − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −

$$Values(A = Size) = \{'Small', 'Big'\}$$
$$S_{Small} = [2^+, 3^-],\quad |S_{Small}| = 5$$
$$S_{Big} = [3^+, 2^-],\qquad |S_{Big}| = 5$$

− − − − − − − − − − − − − − − − − − − − − − − − − − − − − − −

$$Gain(S, A = Size) = E(s) - \sum_{v \in \{'Small', 'Big'\}} \frac{|S_v|}{|S|} E(S_v)$$

$$E(S_{Big}) = E(S_{Small}) = -\frac{2}{5}\log\left(\frac{2}{5}\right) - \frac{3}{5}\log\left(\frac{3}{5}\right) \cong 0.97$$

$$Gain(S, A = Size) = 1 - \left[\frac{1}{2} * 0.97 + \frac{1}{2} * 0.97\right] = 0.03$$

$$\underline{Gain(S, A = Size) = 0.03}$$

---

$$Value(Sound) = \{'Yes', 'No'\}$$

$$S_{Yes} = [1^+, 3^-], \qquad |S_{Yes}| = 4$$

$$S_{No} = [4^+, 2^-], \qquad |S_{No}| = 6$$

---

$$Gain(S, A = Sound) = E(s) - \sum_{v \in \{'Yes', 'No'\}} \frac{|S_v|}{|S|} E(S_v)$$

$$E(S_{Yes}) = -\frac{1}{4}\log\left(\frac{1}{4}\right) - \frac{3}{4}\log\left(\frac{3}{4}\right) \cong 0.811$$

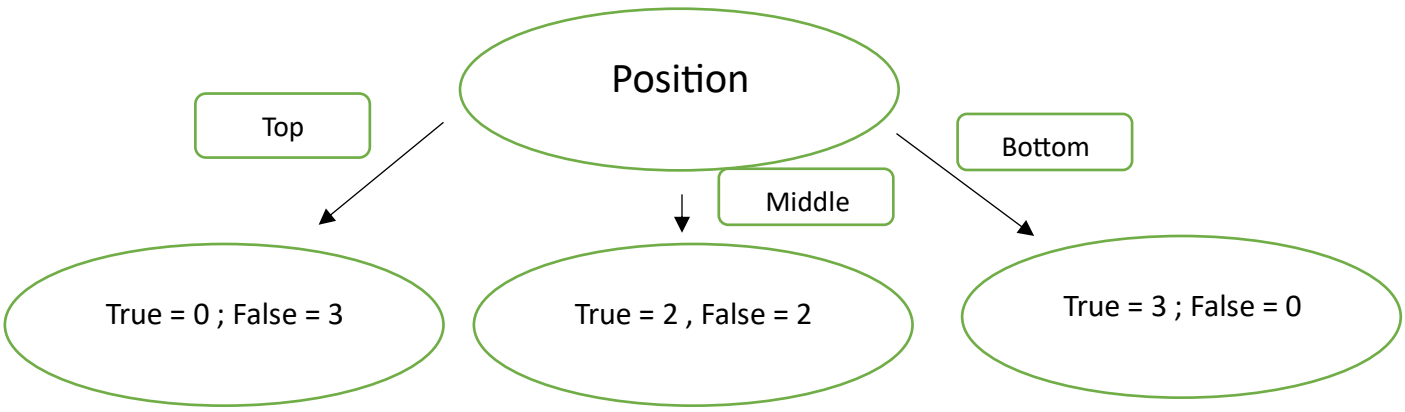
$$E(S_{No}) = -\frac{4}{6}\log\left(\frac{4}{6}\right) - \frac{2}{6}\log\left(\frac{2}{6}\right) \cong 0.918$$

$$Gain(S, A = Sound) = 1 - \left[\frac{4}{10} * 0.811 + \frac{6}{10} * 0.918\right] \cong 0.124$$

$$\underline{Gain(S, A = Sound) \cong 0.124}$$

---

$$Max\ Information\ Gain\ with\ A = Position$$




$$The\ left\ and\ the\ right\ nodes\ balanced. We'll\ keep\ splitting\ the\ middle\ node\ (i.e., 'Middle')$$

---

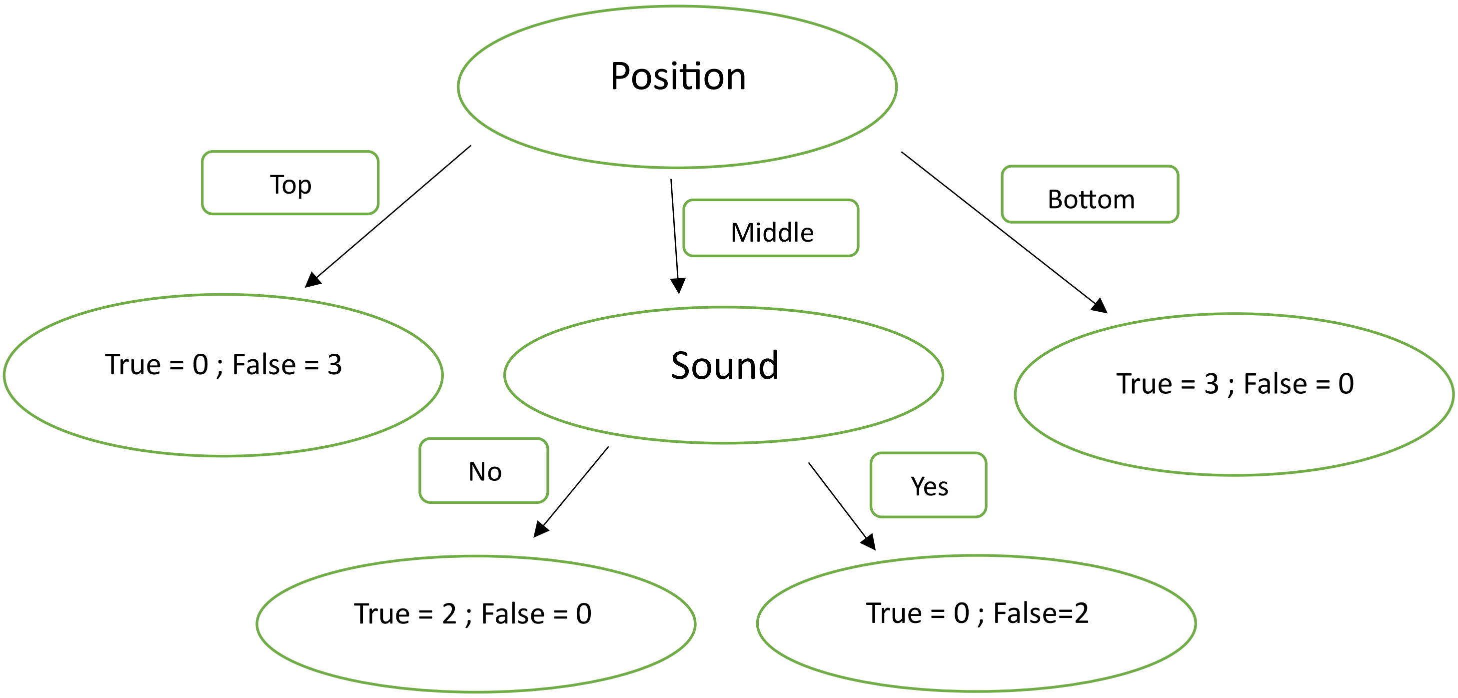
$$V(Sound) = \{'Yes', 'No'\}$$

$$S_{Yes} = [0^+, 2^-], \qquad |S_{yes}| = 2$$

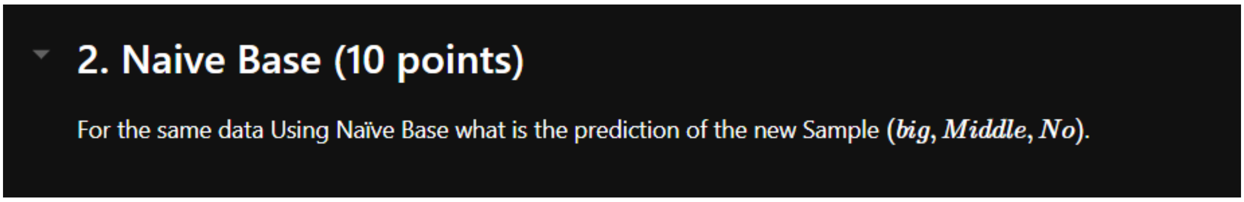
$$S_{No} = [2^+, 0^-], \qquad |S_{No}| = 2$$

$$For\ A = Sound, we'll\ get\ Minimun\ impurity\ entropy\ (i.e., 0), thus\ we'll$$
$$get\ the\ Maximun\ information\ gain\ with\ A = Sound\ for\ node\ 'Middle'$$

*Therefore, we'll choose $A = Sound$ as our split node.*



*The resulting Tree is shown above ↑*



## 2. Naive Base (10 points)

For the same data Using Naïve Base what is the prediction of the new Sample $(\boldsymbol{big}, \boldsymbol{Middle}, \boldsymbol{No})$.

$$P(Clicked = True) = \frac{5}{10}$$
$$P(Clicked = False) = \frac{5}{10}$$

| Sound | True | False |
|---|---|---|
| Yes | 1/5 | 3/5 |
| No | 4/5 | 2/5 |

| Size | True | False |
|---|---|---|
| Big | 2/5 | 3/5 |
| Small | 3/5 | 2/5 |

| Position | True | False |
|---|---|---|
| Top | 0 | 3/5 |
| Middle | 2/5 | 2/5 |
| Bottom | 3/5 | 0 |

$$Prediciton(< Size = Big, Position = Middle, Sound = No >) = ?$$

$$V_{nb} = argmax_{v_j \in \{True, False\}}(P(v_j)) \prod_i P(a_i|v_j)$$

$$V_{nb}(True) = P(True)P(Big|True)P(Middle|True)P(No|True) = \left(\frac{5}{10}\right)\left(\frac{2}{5}\right)\left(\frac{2}{5}\right)\left(\frac{4}{5}\right) = 0.064$$

$$V_{nb}(False) = P(False)P(Big|False)P(Middle|False)P(No|False) = \left(\frac{5}{10}\right)\left(\frac{3}{5}\right)\left(\frac{2}{5}\right)\left(\frac{2}{5}\right) = 0.048$$

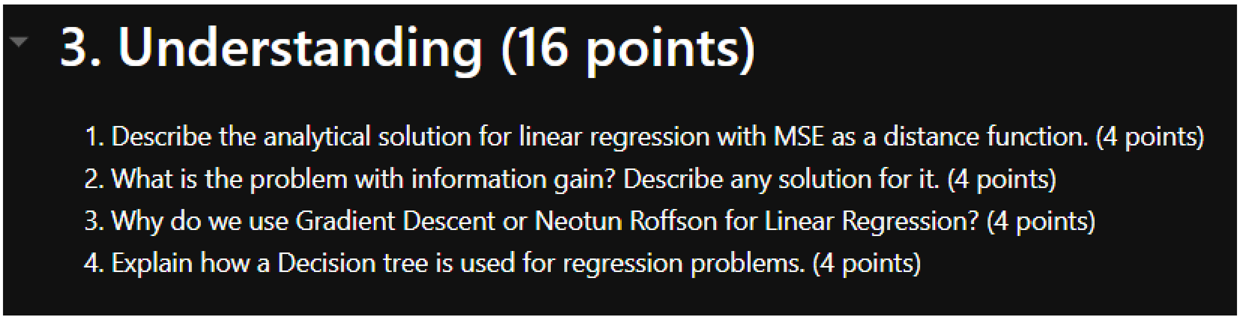
$$Normalize\ the\ values:$$

$$V_{nb}\widetilde{(True)} = \frac{0.064}{0.064 + 0.048} = 0.571\ ; V_{nb}\widetilde{(False)} = \frac{0.048}{0.064 + 0.048} = 0.428$$

$$V_{nb}\widetilde{(True)} > V_{nb}\widetilde{(False)}, thus:$$

$$Prediciton(< Size = Big, Position = Middle, Sound = No >) = True$$

∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼∼



## 3. Understanding (16 points)

1. Describe the analytical solution for linear regression with MSE as a distance function. (4 points)
2. What is the problem with information gain? Describe any solution for it. (4 points)
3. Why do we use Gradient Descent or Neotun Roffson for Linear Regression? (4 points)
4. Explain how a Decision tree is used for regression problems. (4 points)

1. Linear regression is a technique used to understand the relationship between two variables, often by fitting a line to the data. When employing Mean Squared Error (MSE) as the distance function, our goal is to find the line that best predicts the outcome variable based on the input variable.
   The analytical solution involves finding the slope (m) and y-intercept (b) of this line by minimizing the MSE.

In simple, we want to minimize the average squared difference between the predicted values and the actual values in our dataset.

2. The issue with information gain is that it tends to favour features with lots of categories. This means if a feature has many options, it might look more useful for splitting data, even if it's not actually that helpful.
   One solution is using something called "Information Gain Ratio" This adjusts the information gain to be fairer across all features, no matter how many categories they have. It basically makes sure each feature gets a fair shot at showing how useful it is for making decisions in the model.

3. We use Gradient Descent or Newton-Raphson for Linear Regression because they help us find the best-fitting line for our data.
   Gradient Descent is slowly adjusting the parameters of the line to minimize the error. It's similar to walking down a hill step by step until we find the lowest point, which corresponds to the best parameters for our line.
   Newton-Raphson, on the other hand, calculates the best parameters directly using derivatives. It's like taking a shortcut if we already have a good guess of the parameters.

4. A decision tree for regression predicts numbers instead of categories. It works by splitting the data based on features, finding the average value at each split, and using that to make predictions. It keeps splitting until it creates a tree that best fits the data. Then, to predict a new number, it follows the splits down the tree to find the average value associated with the features of the new data point. This gives us the predicted number.