0.1 Decision Tree

0.1.1 Graph

1. A graph consists of nodes (circles) and edges (lines) connecting the nodes.
2. A walk is a sequence of edges which joins a sequence of nodes
3. A trail is a walk where all edges are distinct
4. A cycle is a trail in which the only repeated nodes are the first and last nodes
5. An acyclic graph has no cycles

0.1.2 Tree

1. A tree is an acyclic graph.
2. A rooted tree has a root node.
3. Depth of node in a rooted tree = distance of node from root node
   (a) depth of root node = 0

0.1.3 Decision Tree

1. Is a rooted tree

0.1.4 Entropy

Given a outcome variable Y, with possible outcomes $y_1, y_2, \ldots, y_n$ which occur with purity / probability $P(y_1), P(y_2), \ldots, P(y_n)$, the entropy of Y is defined as:

$$D(Y) = -\sum_{i=1}^{n} P(y_i) \log_2 P(y_i)$$

0.1.5 Conditional Entropy

Given a feature variable X, with split outcome x_1, x_2 which occur with probability $P(x_1), P(x_2)$, the conditional entropy of Y given X is defined as:

$$D(Y|X) = \sum_{i=1}^{2} P(x_i) D(Y|X = x_i)$$

0.1.6 Decision Tree Algorithm: Entropy

1. Start at root node
2. Check for termination conditions, if any, e.g.:
   (a) Minimum purity threshold reached
   (b) Tree cannot be further split with the preset minimum purity threshold.
   (c) Any other stopping criterion is satisfied (such as the maximum depth of the tree).
3. Calculate entropy for current node (base entropy)
4. For each feature variable, for each split outcome, calculate conditional entropy.
5. Choose the feature variable and split outcome with the highest entropy reduction = base entropy - conditional entropy. Branch the current node by this choice.
6. Repeat Step 2-5 for each of the two branched nodes.

0.1.7 Gini Index

Given a outcome variable Y, with possible outcomes $y_1, y_2, \ldots, y_n$ which occur with probability $P(y_1), P(y_2), \ldots, P(y_n)$, the Gini index of Y is defined as:

$$G(Y) = \sum_{i=1}^{n} P(y_i)(1 - P(y_i))$$

0.1.8 Conditional Gini Index

Given a feature variable X, with split outcome x_1, x_2 which occur with probability $P(x_1), P(x_2)$, the conditional Gini index of Y given X is defined as:

$$G(Y|X) = \sum_{i=1}^{2} P(x_i)G(Y|X = x_i)$$

0.1.9 Decision Tree Algorithm: Gini Index

1. Same as Decision Tree Algorithm for Entropy but replace Entropy with Gini Index.

0.1.10 Complexity Parameter C_p

1. Smaller values of C_p correspond to decision trees of larger sizes
2. Larger values of C_p correspond to decision trees of smaller sizes

0.1.11 Prediction Surface

1. Rectangular surfaces
2. Can only be axis-aligned

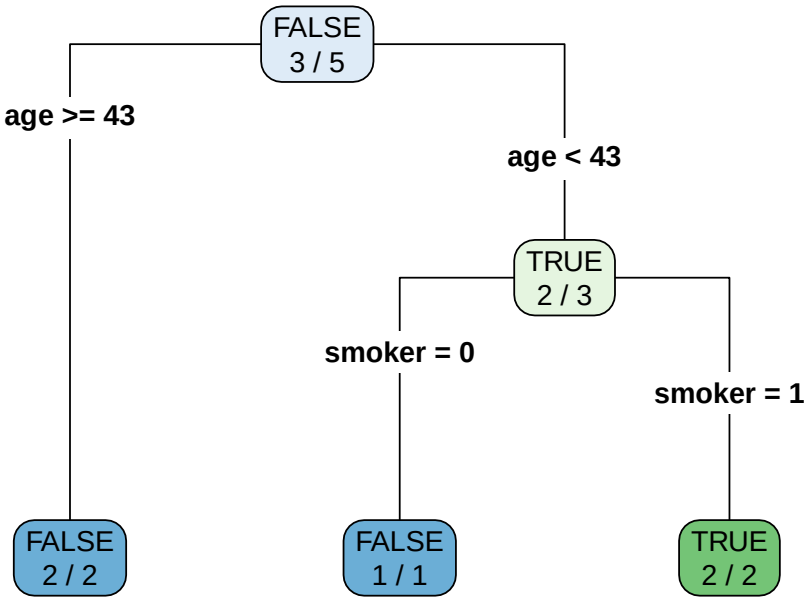
0.1.12 R Implementation

```
library(rpart)
library(rpart.plot)
data <- data.frame(
  id = 1:5,
  gender = c('M', 'M', 'F', 'M', 'F'),
  age = c(21, 33, 40, 60, 45),
  smoker = c(TRUE, FALSE, TRUE, TRUE, FALSE),
  bmi = c(22, 25, 28, 24, 26),
  diabetes = c(TRUE, FALSE, TRUE, FALSE, FALSE),
  stringsAsFactors = TRUE
)
data
```

```
##   id gender age smoker bmi diabetes
## 1  1      M  21   TRUE  22     TRUE
## 2  2      M  33  FALSE  25    FALSE
## 3  3      F  40   TRUE  28     TRUE
## 4  4      M  60   TRUE  24    FALSE
## 5  5      F  45  FALSE  26    FALSE
```

```
fit <- rpart(
  diabetes ~ gender + age + smoker + bmi,
  method = 'class',
  data = data,
  control = rpart.control(minsplit=1),
  parms = list(split = 'information')
)

rpart.plot(fit, type = 4, extra = 2, clip.right.labs = FALSE, varlen = 0,
           faclen = 0)
```



0.1.13 Calculation Intensive Exam Questions & Solutions

Entropy involving n outcomes

Adapted from Midterm Q2. Let X be the outcome variable with $n = 2$ possible outcomes, which occur with purity c(0.5, 0.5). Calculate the entropy of X.

Solution.

1. Copy paste the following code

```
entropy <- function(prob) {
  sum <- 0
  for (p in prob) {
    sum <- sum + p * log2(p)
```

```
  }
  return (-sum)
}
```

2. Calculate entropy

```
entropy(c(0.5, 0.5))

## [1] 1
```

Gini Index involving n outcomes

Adapted from Midterm Q28. Let X be the outcome variable with $n = 2$ possible outcomes, which occur with purity c(1490/2201, 1-1490/2201). Calculate the Gini index of X.

Solution.

1. Copy paste the following code

```
gini_index <- function(prob) {
  sum <- 0
  for (p in prob) {
    sum <- sum + p * (1-p)
  }
  return (sum)
}
```

2. Calculate Gini index

```
gini_index(c(1490/2201, 1-1490/2201))

## [1] 0.4373668
```