Exercise sheet: Decision trees and ensemble methods

Prepared by: Mr Chunchao Ma (PhD Candidate), Mauricio A Álvarez

1. The table below lists a sample of data from a census There are four descriptive features and one target

ID	AGE	EDUCATION	MARITAL STATUS	OCCUPATION	ANNUAL INCOME
1	39	bachelors	never married	transport	25K-50K
2	50	bachelors	married	professional	25K-50K
3	18	high school	never married	agriculture	$\leq$25K
4	28	bachelors	married	professional	25K-50K
5	37	high school	married	agriculture	25K-50K
6	24	high school	never married	armed forces	$\leq$ 25K
7	52	high school	divorced	transport	25K-50K
8	40	doctorate	married	professional	$\geq$ 50K

feature in this dataset: AGE, EDUCATION, MARITAL STATUS and OCCUPATION. The target feature is the ANNUAL INCOME.

   (a) Calculate **information gain** (based on entropy) for the EDUCATION, MARITAL STATUS, and OCCUPATION features.

   (b) Calculate **information gain** using the **Gini index** for the EDUCATION, MARITAL STATUS, and OCCUPATION features.

   (c) When building a decision tree, the easiest way to handle a continuous feature is to define a threshold around which splits will be made. What would be the optimal threshold to split the continuous AGE feature (use information gain based on entropy as the feature selection measure)?

Answers:

(a) Based on the lecture slides, we know that computing information gain (based on entropy) involves the following three equations (we also follow the notation on the lecture slides):

$$H(t, \mathcal{D}) = -\sum_{I \in \text{levels}(t)} (P(t = I) \times \log_2(P(t = I)))$$

$$\text{rem}(d, \mathcal{D}) = \sum_{I \in \text{levels}(d)} \underbrace{\frac{|\mathcal{D}_{d=I}|}{|\mathcal{D}|}}_{weighting} \times \underbrace{H\left(t, \mathcal{D}_{d=I}\right)}_{\substack{\text{entropy of} \\ \text{partition } \mathcal{D}_{d=1}}}$$

$$IG(d, \mathcal{D}) = H(t, \mathcal{D}) - \text{rem}(d, \mathcal{D})$$

Step 1: we calculate the entropy for the target feature in this dataset.

$$
\begin{aligned}
&H(\text{ANNUAL INCOME}, \mathcal{D}) \\
&= - \sum_{l \in \begin{cases} < 25K, \\ 25K - 50K, \\ > 50K \end{cases}} P(\text{AN.INC.} = l) \times \log_2(P(\text{AN.INC.} = l)) \\
&= - \left(\left(\frac{2}{8} \times \log_2 \left(\frac{2}{8} \right) \right) + \left(\frac{5}{8} \times \log_2 \left(\frac{5}{8} \right) \right) + \left(\frac{1}{8} \times \log_2 \left(\frac{1}{8} \right) \right) \right) \\
&= 1.2988 \text{ bits}
\end{aligned}
$$

Step 2: Remainder for each feature: e.g Remainder for EDUCATION feature:

$$
\begin{aligned}
\text{rem (EDUCATION, } \mathcal{D}) &= \left(\frac{|\mathcal{D}_{\text{EDUCATION =high school}}|}{|\mathcal{D}|} \times H\left(t, \mathcal{D}_{\text{EDUCATION= high school }}\right) \right) \\
&+ \left(\frac{|\mathcal{D}_{\text{EDUCATION =bachelors}}|}{|\mathcal{D}|} \times H\left(t, \mathcal{D}_{\text{EDUCATION =bachelors}}\right) \right) \\
&+ \left(\frac{|\mathcal{D}_{\text{EDUCATION =doctorate}}|}{|\mathcal{D}|} \times H\left(t, \mathcal{D}_{\text{EDUCATION =doctorate}}\right) \right) \\
&= \left(\frac{4}{8} \times \left(- \sum_{l \in \begin{cases} <25K, \\ 25K-50K, \\ >50K \end{cases}} P(\text{AN.INC.} = l) \times \log_2(P(\text{AN.INC.} = l)) \right) \right) \\
&+ \left(\frac{3}{8} \times \left(- \sum_{l \in \begin{cases} <25K, \\ 25K-50K, \\ >50K \end{cases}} P(\text{AN.INC.} = l) \times \log_2(P(\text{AN.INC.} = l)) \right) \right) \\
&+ \left(\frac{1}{8} \times \left(- \sum_{l \in \begin{cases} <25K, \\ 25K-50K, \\ >50K \end{cases}} P(\text{AN.INC.} = l) \times \log_2(P(\text{AN.INC.} = l)) \right) \right) \\
&= -\frac{4}{8} \times \left(\left(\frac{2}{4} \times \log_2\left(\frac{2}{4}\right) \right) + \left(\frac{2}{4} \times \log_2\left(\frac{2}{4}\right) \right) + \left(0 \times \log_2\left(\frac{0}{4}\right) \right) \right) \\
&- \frac{3}{8} \times \left(\left(1 \times \log_2\left(\frac{3}{3}\right) \right) + \left(0 \times \log_2\left(\frac{0}{3}\right) \right) + \left(0 \times \log_2\left(\frac{0}{3}\right) \right) \right) \\
&- \frac{1}{8} \times \left(\left(0 \times \log_2\left(\frac{0}{1}\right) \right) + \left(0 \times \log_2\left(\frac{0}{1}\right) \right) + \left(1 \times \log_2\left(1\right) \right) \right) = 0.5
\end{aligned}
$$

Remainder for other features follow the same idea above.

Step 3: Compute the information gain, E.g with regard to the EDUCATION feature

$$
\begin{aligned}
IG(\text{ EDUCATION } \mathcal{D}) &= H(\text{ANNUAL INCOME}, \mathcal{D}) - \text{rem(EDUCATION }, \mathcal{D}) \\
&= 1.2988 - 0.5 = 0.7988 \quad \text{bits}
\end{aligned}
$$

To compute the information gain for the other features, we follow the same ideas than above.

The table below lists the rest of the calculations for the information gain for the EDUCATION , MARITAL STATUS , and OCUPATION features.

Split by Feature	Level	Instances	Rem.	Info. Gain
EDUCATION	high school	$\mathbf{d}_3, \mathbf{d}_5, \mathbf{d}_6, \mathbf{d}_7$	0.5	0.7988
	bachelors	$\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3$		
	doctorate	$\mathbf{d}_8$		
MARITAL STATUS	never married	$\mathbf{d}_1, \mathbf{d}_3, \mathbf{d}_6$	0.75	0.5488
	married	$\mathbf{d}_2, \mathbf{d}_4, \mathbf{d}_5, \mathbf{d}_8$		
	divorced	$\mathbf{d}_7$		
OCCUPATION	transport	$\mathbf{d}_1, \mathbf{d}_7$	0.5944	0.7044
	professional	$\mathbf{d}_2, \mathbf{d}_4, \mathbf{d}_8$		
	agriculture	$\mathbf{d}_3, \mathbf{d}_5$		
	armed forces	$\mathbf{d}_6$		

(b) Based on the Lecture slides, we know that information gain can be calculated using the **Gini index** by replacing the entropy measure with the **Gini index**.

The Gini index is

$$\text{Gini } (t, \mathcal{D}) = 1 - \sum_{I \in \text{ levels } (t)} P(t = I)^2$$

We use the same idea that in part (a) by replacing the entropy measure with the **Gini index**.

Step 1: We calculate the **Gini index** for the target feature in this dataset.

$$\begin{aligned}
&\text{Gini (ANNUAL INCOME, } \mathcal{D}) \\
&= 1 - \sum_{l \in \begin{cases} < 25K, \\ 25K - 50K, \\ > 50K \end{cases}} P(\text{AN.INC.} = l)^2 \\
&= 1 - \left(\left(\frac{2}{8}\right)^2 + \left(\frac{5}{8}\right)^2 + \left(\frac{1}{8}\right)^2 \right) = 0.5313
\end{aligned}$$

Step 2: Remainder for each feature: e.g Remainder for EDUCATION feature:

$$
\begin{aligned}
\text{rem (EDUCATION, } \mathcal{D}) &= \left(\frac{|\mathcal{D}_{\text{EDUCATION =high school}}|}{|\mathcal{D}|} \times \text{ Gini } (t, \mathcal{D}_{\text{EDUCATION =high school}}) \right) \\
&+ \left(\frac{|\mathcal{D}_{\text{EDUCATION =bachelors}}|}{|\mathcal{D}|} \times \text{ Gini } (t, \mathcal{D}_{\text{EDUCATION =bachelors}}) \right) \\
&+ \left(\frac{|\mathcal{D}_{\text{EDUCATION =doctorate}}|}{|\mathcal{D}|} \times \text{ Gini } (t, \mathcal{D}_{\text{EDUCATION =doctorate}}) \right) \\
&= \left(\frac{4}{8} \times \left(1 - \sum_{l \in \begin{cases} <25K, \\ 25K-50K, \\ >50K \end{cases}} P(\text{AN.INC.} = l)^2 \right) \right) \\
&\quad + \left(\frac{3}{8} \times \left(1 - \sum_{l \in \begin{cases} <25K, \\ 25K-50K, \\ >50K \end{cases}} P(\text{AN.INC.} = l)^2 \right) \right) \\
&\quad + \left(\frac{1}{8} \times \left(1 - \sum_{l \in \begin{cases} <25K, \\ 25K-50K, \\ >50K \end{cases}} P(\text{AN.INC.} = l)^2 \right) \right) \\
&= \frac{4}{8} \times \left(1 - \left(\left(\frac{2}{4} \right)^2 + \left(\frac{2}{4} \right)^2 + 0^2 \right) \right) \\
&\quad + \frac{3}{8} \times \left(1 - \left(1^2 + 0^2 + 0^2 \right) \right) \\
&\quad + \frac{1}{8} \times \left(1 - \left(0^2 + 0^2 + 1^2 \right) \right) = 0.25
\end{aligned}
$$

Remainder for other features follow the same idea above.

Step 3: Compute the information gain, E.g with regard to the EDUCATION feature

$$
\begin{aligned}
IG(\text{ EDUCATION } \mathcal{D}) = &\text{ Gini (ANNUAL INCOME, } \mathcal{D}) - \text{rem(EDUCATION , } \mathcal{D}) \\
&= 0.5313 - 0.25 = 0.2813
\end{aligned}
$$

We follow the same procedure to compute the information gain with respect to the other features.

The table below lists the rest of the calculations of information gain for the EDUCATION, MARITAL STATUS, and OCCUPATION features.

Split by Feature	Level	Instances	Partition Gini Index	Rem.	Info. Gain
EDUCATION	high school	$\mathbf{d}_3, \mathbf{d}_5, \mathbf{d}_6, \mathbf{d}_7$	0.5	0.25	0.2813
	bachelors	$\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3$	0		
	doctorate	$\mathbf{d}_8$	0		
MARITAL STATUS	never married	$\mathbf{d}_1, \mathbf{d}_3, \mathbf{d}_6$	0.4444	0.3542	0.1771
	married	$\mathbf{d}_2, \mathbf{d}_4, \mathbf{d}_5, \mathbf{d}_8$	0.375		
	divorced	$\mathbf{d}_7$	0		
OCCUPATION	transport	$\mathbf{d}_1, \mathbf{d}_7$	0	0.2917	0.2396
	professional	$\mathbf{d}_2, \mathbf{d}_4, \mathbf{d}_8$	0.4444		
	agriculture	$\mathbf{d}_3, \mathbf{d}_5$	0.5		
	armed forces	$\mathbf{d}_6$	0		

(c) First sort the instances in the dataset according to the AGE feature, as shown in the following table.

ID	AGE	ANNUAL INCOME
3	18	$< 25K$
6	24	$< 25K$
4	28	$25K - 50K$
5	37	$25K - 50K$
1	39	$25K - 50K$
8	40	$> 50K$
2	50	$25K - 50K$
7	52	$25K - 50K$

Based on this ordering, the mid-points in the AGE values of instances that are adjacent in the new ordering but that have different target levels define the possible threshold points. These points are 26, 39.5, and 45.

We calculate the information gain for each of these possible threshold points using the entropy value we calculated in part (a) of this question (1.2988 bits) as follows:

Split by Feature	Partition	Instances	Partition Entropy	Rem.	Info. Gain
> 26	$\mathcal{D}_1$	$\mathbf{d}_3, \mathbf{d}_6$	0	0.4875	0.8113
	$\mathcal{D}_2$	$\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_4, \mathbf{d}_5, \mathbf{d}_7, \mathbf{d}_8$	0.6500		
> 39.5	$\mathcal{D}_3$	$\mathbf{d}_1, \mathbf{d}_3, \mathbf{d}_4, \mathbf{d}_5, \mathbf{d}_6$	0.9710	0.9456	0.3532
	$\mathcal{D}_4$	$\mathbf{d}_2, \mathbf{d}_7, \mathbf{d}_8$	0.9033		
> 45	$\mathcal{D}_5$	$\mathbf{d}_1, \mathbf{d}_3, \mathbf{d}_4, \mathbf{d}_5, \mathbf{d}_6, \mathbf{d}_8$	1.4591	1.0944	0.2044
	$\mathcal{D}_6$	$\mathbf{d}_2, \mathbf{d}_7$	0		

We select the threshold as 26 since the point 26 has the highest information gain from the thresholds.

2. The following table lists a dataset of the scores students achieved on an exam described in terms of whether the student studied for the exam (STUDIED) and the energy level of the lecturer when grading the student's exam (ENERGY). Which of the two descriptive features should we use as the testing

ID	STUDIED	ENERGY	SCORE
1	yes	tired	65
2	no	alert	20
3	yes	alert	90
4	yes	tired	70
5	no	tired	40
6	yes	alert	85
7	no	tired	35

criterion at the root node of a decision tree to predict students' scores?

Answer:

The target feature in this question (SCORE) is continuous. When a decision tree is predicting a continuous target, we choose as the descriptive feature to use at each node in the tree, the one that results in the minimum weighted variance after the dataset has been split based on that feature.
Following the notation on the Lecture slides, the variance at a node can be calculated using the following equation:

$$\text{var}(t, \mathcal{D}) = \frac{\sum_{i=1}^{n} (t_i - \bar{t})^2}{n-1}$$

If the dataset is split by STUDIED, we have two partition domains $\mathcal{D}_1$ and $\mathcal{D}_2$ (see the table below). $\mathcal{D}_1$ and $\mathcal{D}_2$ have four $(\mathbf{d}_1, \mathbf{d}_3, \mathbf{d}_4, \mathbf{d}_6)$ and three instances $(\mathbf{d}_4, \mathbf{d}_5, \mathbf{d}_7)$ separately. Based on the values of SCORE, we obtain $\text{var}(t, \mathcal{D}_1) = 141\frac{2}{3}$ and $\text{var}(t, \mathcal{D}_1) = 108\frac{1}{3}$. Similarly, we can obtain the variance when the data is split by the feature ENERGY (see the table below).

We choose the feature that minimises the weighted variance across the resulting partitions:

$$\mathbf{d}[best] = \underset{d \in \mathbf{d}}{\text{argmin}} \sum_{I \in \text{levels}(d)} \frac{|\mathcal{D}_{d=l}|}{|\mathcal{D}|} \times \text{var}\left(t, \mathcal{D}_{d=I}\right)$$

The table below shows the calculation of the weighted variance for each of the descriptive features in this domain.

Split by Feature	Level	Partition	Instances	$P(d = l)$	$\text{var}(t, \mathcal{D})$	Weighted Variance
STUDIED	yes	$\mathcal{D}_1$	$\mathbf{d}_1, \mathbf{d}_3, \mathbf{d}_4, \mathbf{d}_6$	$\frac{4}{7}$	$141\frac{2}{3}$	127.3810
	no	$\mathcal{D}_2$	$\mathbf{d}_4, \mathbf{d}_5, \mathbf{d}_7$	$\frac{3}{7}$	$108\frac{1}{3}$	
ENERGY	alert	$\mathcal{D}_5$	$\mathbf{d}_2, \mathbf{d}_3, \mathbf{d}_6$	$\frac{3}{7}$	1525	829.7619
	tired	$\mathcal{D}_6$	$\mathbf{d}_1, \mathbf{d}_4, \mathbf{d}_5, \mathbf{d}_7$	$\frac{4}{7}$	$308\frac{1}{3}$	

From these calculations we can see that splitting the dataset using the STUDIED feature results in the lowest weighted variance. Consequently, we should use the STUDIED feature at the root node of the tree.

3. The following table lists a dataset containing the details of five participants in a heart disease study, and a target feature RISK which describes their risk of heart disease. Each patient is described in terms of four descriptive features: EXERCISE (how regularly do they exercise?), SMOKER (do they smoke?), OBESE (are they overweight?) FAMILY (did any of their parents or siblings suffer from heart disease?).

ID	EXERCISE	SMOKER	OBESE	FAMILY	RISK
1	daily	false	false	yes	low
2	weekly	true	false	yes	high
3	daily	false	false	no	low
4	rarely	true	true	yes	high
5	rarely	true	true	no	high

(a) As part of the study researchers have decided to create a predictive model to screen participants based on their risk of heart disease. You have been asked to implement this screening model using a **random forest**. The three tables below list three bootstrap samples that have been generated from the above dataset. Using these bootstrap samples create the decision trees that will be in the random forest model (use entropy based information gain as the feature selection criterion).

ID	EXERCISE	FAMILY	RISK
1	daily	yes	low
2	weekly	yes	high
2	weekly	yes	high
5	rarely	no	high
5	rarely	no	high

Boostrap Sample A

ID	SMOKER	OBESE	RISK
1	false	false	low
2	true	false	high
2	true	false	high
4	true	true	high
5	true	true	high

Boostrap Sample B

ID	OBESE	FAMILY	RISK
1	false	yes	low
1	false	yes	low
2	false	yes	high
4	true	yes	high
5	true	no	high

Boostrap Sample C

(b) Assuming the random forest model you have created uses majority voting, what prediction will it return for the following query:

EXERCISE=rarely, SMOKER=false, OBESE=true, FAMILY=yes.

Answers:

(a) The entropy calculation for Bootstrap Sample A is:

$$\begin{aligned}
&\mathbf{H} \quad \text{(RISK, Boostrap Sample A)} \\
&= - \sum_{l \in \left\{ \begin{array}{l} low, \\ high \end{array} \right.} P(\text{RISK} = l) \times \log_2(P(\text{RISK} = l)) \\
&= - \left(\left(\frac{1}{5} \times \log_2 \left(\frac{1}{5} \right) \right) + \left(\frac{4}{5} \times \log_2 \left(\frac{4}{5} \right) \right) \right) = 0.7219 \quad \text{bits}
\end{aligned}$$

The information gain for each of the features in Bootstrap Sample A is as follows:

Split by Feature	Level	Instances	Partition Entropy	Rem.	Info. Gain
EXERCISE	daily	$\mathbf{d}_1$	0	0	0.7219
	weekly	$\mathbf{d}_2, \mathbf{d}_2$	0		
	rarely	$\mathbf{d}_5, \mathbf{d}_5$	0		
FAMILY	yes	$\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_2$	0.9183	0.5510	0.1709
	no	$\mathbf{d}_5, \mathbf{d}_5$	0		

These calculations show that the EXERCISE feature has the highest information gain of the descriptive features in Bootstrap Sample A and should be added as the root node of the decision tree generated from Bootstrap Sample A. What is more, splitting on EXERCISE generates pure sets. So, the decision tree does not need to be expanded beyond this initial test. The final tree generated for Bootstrap Sample A: if EXERCISE Level is **daily**, the RISK is **low**; if EXERCISE Level is **weekly** or **rarely**, the RISK is **high**.

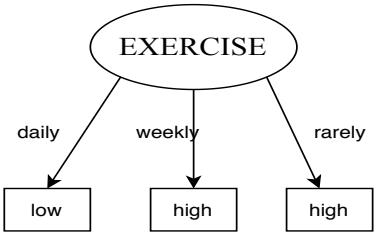

Figure 1: Decision Tree for Boostrap Sample A

By chance, Bootstrap Sample B has the same distribution of target feature values as Bootstrap Sample A, so the entropy calculation for Bootstrap Sample B is the same as the calculation for Bootstrap Sample A:

$$
\begin{aligned}
&\mathbf{H} \quad \text{(RISK, Boostrap Sample B)} \\
&= - \sum_{l \in \begin{cases} low, \\ high \end{cases}} P(\text{RISK} = l) \times \log_2(P(\text{RISK} = l)) \\
&= - \left(\left(\frac{1}{5} \times \log_2 \left(\frac{1}{5} \right) \right) + \left(\frac{4}{5} \times \log_2 \left(\frac{4}{5} \right) \right) \right) = 0.7219 \quad \text{bits}
\end{aligned}
$$

The information gain for each of the features in Bootstrap Sample B is as follows:

Split by Feature	Level	Instances	Partition Entropy	Rem.	Info. Gain
SMOKER	true	$\mathbf{d}_2, \mathbf{d}_2, \mathbf{d}_4, \mathbf{d}_5$	0	0	0.7219
	false	$\mathbf{d}_1$	0		
OBESE	true	$\mathbf{d}_4, \mathbf{d}_5$	0	0.5510	0.1709
	false	$\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_2$	0.9183		

These calculations show that the SMOKER feature has the highest information gain of the descriptive features in Bootstrap Sample B and should be added as the root node of the decision tree generated

from Bootstrap Sample B. What is more, splitting on SMOKER generates pure sets, So the decision tree does not need to be expanded beyond this initial test. The final tree generated for Bootstrap Sample B: if SMOKER Level is **true**, the RISK is **high**; if SMOKER Level is **false**, the RISK is **low**.

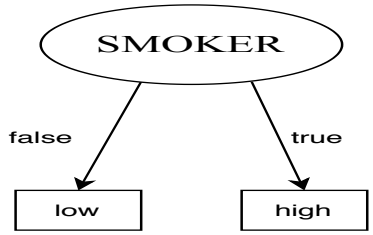

Figure 2: Decision Tree for Boostrap Sample B

The entropy calculation for Bootstrap Sample C is:

$$\begin{aligned}
&\mathbf{H} \quad \text{(RISK, Boostrap Sample C)} \\
&= - \sum_{l \in \begin{cases} low, \\ high \end{cases}} P(\text{RISK} = l) \times \log_2(P(\text{RISK} = l)) \\
&= -\left(\left(\frac{2}{5} \times \log_2\left(\frac{2}{5}\right)\right) + \left(\frac{3}{5} \times \log_2\left(\frac{3}{5}\right)\right)\right) = 0.9710 \quad \text{bits}
\end{aligned}$$

The information gain for each of the features in Bootstrap Sample C is as follows:

Split by Feature	Level	Instances	Partition Entropy	Rem.	Info. Gain
OBESE	true	$\mathbf{d}_4, \mathbf{d}_5$	0	0.5510	0.4200
	false	$\mathbf{d}_1, \mathbf{d}_1, \mathbf{d}_2$	0.9183		
FAMILY	true	$\mathbf{d}_1, \mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_4$	1.0	0.8	0.1709
	false	$\mathbf{d}_5$	0		

These calculations show that the OBESE feature has the highest information gain of the descriptive features in Bootstrap Sample C and should be added as the root node of the decision tree generated from Bootstrap Sample C. Splitting Bootstrap Sample C creates one pure partition for OBESE = true ($\mathbf{d}_4$, $\mathbf{d}_5$) where all the instances have RISK = high, and an impure partition for OBESE = false where two instances ($\mathbf{d}_1$, $\mathbf{d}_1$) have RISK = low and for one instance ($\mathbf{d}_2$) RISK = high. Normally this would mean that we would continue to split the impure partition to create pure sets. However, in this instance there is only one feature that we can still use to split this partition, the FAMILY feature, and all the instances in this partition have the same level for this feature FAMILY = yes. Consequently, instead of splitting this partition further we simply create a leaf node with the majority target level within the partition: RISK =low.So, the final tree generated for Bootstrap Sample C: if OBESE Level is **true**, the RISK is **high**, if OBESE Level is **false**, the RISK is **low**.

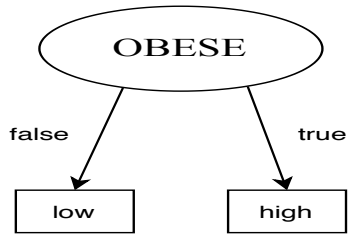

Figure 3: Decision Tree for Boostrap Sample C

(b) Each of the trees in the ensemble will vote as follows:

- Tree 1: EXERCISE=rarely → RISK=high

- Tree 2: SMOKER=false → RISK=low

- Tree 3: OBESE=true → RISK=high

So, the majority vote is for RISK=high, and this is the prediction the model will return for this query.