Solutions to Assignment 2: Regression and Decision Trees

Question 1

Consider the following data on experience (in years) and salary (in thousands of dollars) of a sample of employees:

Experience (Years)	Salary ($1000)
1.2	42
2.5	46
3.1	51
3.9	58
5.2	62

a) Build a model that can help you automatically predict the salary of an employee given their years of experience. Write down the equation of your model's hypothesis.

First, we assign x = Experience, so the linear regression hypothesis is defined as:

$$h_w(x) = w_0 + w_1 \cdot x$$

Since this is a univariate regression problem, then we can use the Ordinary Least Squares (OLS) equations to find the intercept coefficient and the coefficient associated with the experience. The OLS equations are:

$$w_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}, \text{and,}$$

$$w_0 = \bar{y} - w_1\bar{x}.$$

The mean of x and y are:

$$\bar{x} = \frac{1.2 + 2.5 + 3.1 + 3.9 + 5.2}{5} = 3.18$$

$$\bar{y} = \frac{42 + 46 + 51 + 58 + 62}{5} = 51.8$$

Let's find w_1 first,

$$
\begin{aligned}
w_1 &= \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \\
&= \frac{(-1.98)(-9.8) + (-0.68)(-5.8) + (-0.08)(-0.8) + (0.72)(6.2) + (2.02)(10.2)}{(-1.98)^2 + (-0.68)^2 + (-0.08)^2 + (0.72)^2 + (2.02)^2} \\
&= \frac{19.404 + 3.944 + 0.064 + 4.464 + 20.604}{3.9204 + 0.4624 + 0.0064 + 0.5184 + 4.0804} \\
&= \frac{48.48}{8.988} \\
&= 5.39
\end{aligned}
$$

Next, we will calculate w_0,

$$
\begin{aligned}
w_0 &= \bar{y} - w_1\bar{x} \\
&= 51.8 - 5.39 \cdot 3.18 \quad = 34.66
\end{aligned}
$$

Accordingly, the hypothesis equation is:

$$h_w(x) = 34.66 + 5.39 \cdot x$$

b) Use your hypothesis to predict the salary of an employee with 4.5 years of experience.

For an employee with 4.5 years of experience, their salary is predicted as:

$$h_w(x) = 34.66 + 5.39 \cdot 4.5 = \$58,915$$

c) Interpret the coefficients of your hypothesis.

w_0: When x is 0, meaning no experience, the predicted salary is \$34,660.

w_1: For every unit increase in x or additional year of experience, the predicted salary increases by \$5,390.

Question 2

You have the following data on size (in square feet) and number of bedrooms of 3 houses and their respective prices (in thousands of dollars):

House Number	Size (Sq. Ft.)	Bedrooms	Price ($1000)
1	900	1	200
2	1600	3	330
3	1875	4	400

You want to build a model to automatically predict the price of a house given its size (in sq. ft.) and number of bedrooms. You decide to build your model analytically.

a) Define your matrix X and your vector y.

First, we assign $x_1 =$ size, and $x_2 =$ number of bedrooms,

$$X = \begin{bmatrix} 1 & 900 & 1 \\ 1 & 1600 & 3 \\ 1 & 1875 & 4 \end{bmatrix}, \quad y = \begin{bmatrix} 200 \\ 330 \\ 400 \end{bmatrix}$$

b) Solve the normal equation and write the equation of the resulting hypothesis.

The normal equation for linear regression is:

$$w = (X^T X)^{-1} X^T y$$

$$X^T X = \begin{bmatrix} 3 & 4375 & 8 \\ 4375 & 6966250 & 14125 \\ 8 & 14125 & 29 \end{bmatrix}$$

$$X^T y = \begin{bmatrix} 930 \\ 1558250 \\ 3150 \end{bmatrix}$$

$$(X^T X)^{-1} = \begin{bmatrix} 212.72 & -0.3622 & 118.44 \\ -0.3622 & 0.000622 & -0.2044 \\ 118.44 & -0.2044 & 67.389 \end{bmatrix}$$

$$
\begin{aligned}
w &= (X^T X)^{-1} X^T y \\
&= \begin{bmatrix} 212.72 & -0.3622 & 118.44 \\ -0.3622 & 0.000622 & -0.2044 \\ 118.44 & -0.2044 & 67.389 \end{bmatrix} \begin{bmatrix} 930 \\ 1558250 \\ 3150 \end{bmatrix} \\
&= \begin{bmatrix} 171.66 \\ -0.0667 \\ 88.33 \end{bmatrix}
\end{aligned}
$$

Therefore, the hypothesis is given by:

$$h_w(x) = 171.66 - 0.0667 \cdot x_1 + 88.33 \cdot x_2$$

c) Predict the price of a house that is 1500 sq. ft. and has 3 bedrooms.

For a house with 1500 sq.ft. and 3 bedrooms:

$$
\begin{aligned}
h_w(x) &= 171.66 - 0.0667 \cdot x_1 + 88.33 \cdot x_2 \\
&= 171.66 - 0.0667 \cdot 1500 + 88.33 \cdot 3 \\
&= \$336,600
\end{aligned}
$$

d) Interpret the coefficients of your hypothesis.

w_0: When x_1 and x_2 are 0, meaning no additional information is available about a house, we predict a starting price of \$171,660.

w_1: For every unit increase in x_1 or for every increase in square footage of the house, the predicted price decreases by \$66.7, given that x_2 or the number of bedrooms is constant.

w_2: For every unit increase in x_2 or for every additional bedroom, the predicted price increases by \$88,330, given that x_1 or the square footage of the house is constant.

Question 3

You have the following data on the income (in thousands of dollars) and age of 6 individuals and whether they were approved for a credit card (1 = approved, 0 = not approved):

Individual	Income ($1000)	Age (Years)	Approved (0/1)
1	45	25	0
2	60	35	1
3	75	40	1
4	50	28	0
5	90	50	1
6	100	60	1

You want to build a model to predict whether a new applicant will be approved for a credit card given their income and age.

a) Define a suitable cost function that you will target to optimize for this problem.

First we assign,

- x_1 = income, and,
- x_2 = age

This is a binary classification problem, so a suitable cost function is:

$$J(w) = -\frac{1}{m}\sum_{i=1}^{m}\Big(y^{(i)}\log(h_w(x^{(i)})) + (1-y^{(i)})\log(1-h_w(x^{(i)}))\Big)$$

where $h_w(x^{(i)})$ is:

$$h_w(x^{(i)}) = \frac{1}{1+e^{-(w_0+w_1\cdot x_1+w_2\cdot x_2)}},$$

and for $x_0 = 1$,

$$h_w(x^{(i)}) = \frac{1}{1+e^{-w^T x}},$$

b) How many coefficients will you need to optimize in this case?

We will need to optimize three coefficients: w_0, w_1, and w_2.

c) Using the cost function you defined in part (a), derive the update rule for each coefficient that needs to be optimized. Show your work by deriving the expression for the derivative of the cost function with respect to each coefficient.

The update rule for each coefficient is given by:

$$w_j := w_j - \alpha \frac{\partial J(w)}{\partial w_j}$$

We want to find the partial derivative of $J(w)$ with respect to w_j,

$$\frac{\partial J(w)}{\partial w_j} = -\frac{1}{m} \sum_{i=1}^{m} \left(\frac{y(i)}{h_w(x^{(i)})} - \frac{1 - y^{(i)}}{1 - h_w(x^{(i)})} \right) \cdot \frac{\partial h_w(x^{(i)})}{\partial w_j} \tag{1}$$

So we need to find $\frac{\partial h_w(x^{(i)})}{\partial w_j}$ to find the $\frac{\partial J(w)}{\partial w_j}$.

$$\begin{aligned}
\frac{\partial h_w(x^{(i)})}{\partial w_j} &= \frac{\partial}{\partial w_j}\left(\frac{1}{1 + e^{-w^T x}}\right) \\
&= \frac{\partial}{\partial w_j}\left(\frac{1}{1 + e^{-(w_0 + w_1 \cdot x_1 + w_2 \cdot x_2)}}\right) \\
&= \frac{\partial}{\partial w_j}(1 + e^{-(w_0 + w_1 \cdot x_1 + w_2 \cdot x_2)})^{-1} \\
&= -1 \cdot (1 + e^{-w^T x})^{-2} \cdot e^{-w^T x} \cdot (-x_j) \\
&= (1 + e^{-w^T x})^{-2} \cdot e^{-w^T x} \cdot x_j \\
&= (h_w(x^{(i)}))^2 \cdot e^{-w^T x} \cdot x_j
\end{aligned} \tag{2}$$

Let's find $e^{-w^T x}$ in terms of $h_w(x^{(i)})$,

$$\begin{aligned}
h_w(x^{(i)}) &= \frac{1}{1 + e^{-w^T x}} \\
(1 + e^{-w^T x}) \cdot h_w(x^{(i)}) &= 1 \\
h_w(x^{(i)}) + e^{-w^T x} \cdot h_w(x^{(i)}) &= 1 \\
e^{-w^T x} \cdot h_w(x^{(i)}) &= 1 - h_w(x^{(i)}) \\
e^{-w^T x} &= \frac{1 - h_w(x^{(i)})}{h_w(x^{(i)})}
\end{aligned} \tag{3}$$

Substituting 3 in 2, we get,

$$\begin{aligned}\frac{\partial h_w(x^{(i)})}{\partial w_j} &= (h_w(x^{(i)}))^2 \cdot \frac{1 - h_w(x^{(i)})}{h_w(x^{(i)})} \cdot x_j \\ &= (h_w(x^{(i)}))^2 \cdot \frac{1 - h_w(x^{(i)})}{h_w(x^{(i)})} \cdot x_j \\ &= h_w(x^{(i)}) \cdot (1 - h_w(x^{(i)})) \cdot x_j \end{aligned} \tag{4}$$

Substituting 4 in 1, we get,

$$\begin{aligned}\frac{\partial J(w)}{\partial w_j} &= -\frac{1}{m}\sum_{i=1}^{m}\left(\frac{y^{(i)}}{h_w(x^{(i)})} - \frac{1 - y^{(i)}}{1 - h_w(x^{(i)})}\right) \cdot h_w(x^{(i)}) \cdot (1 - h_w(x^{(i)})) \cdot x_j \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left(\frac{y^{(i)}}{h_w(x^{(i)})} \cdot h_w(x^{(i)}) \cdot (1 - h_w(x^{(i)})) - \frac{1 - y^{(i)}}{1 - h_w(x^{(i)})} \cdot h_w(x^{(i)}) \cdot (1 - h_w(x^{(i)}))\right) \cdot x_j \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)}(1 - h_w(x^{(i)})) - (1 - y^{(i)})h_w(x^{(i)})\right) \cdot x_j \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} - y^{(i)}h_w(x^{(i)}) - h_w(x^{(i)}) + y^{(i)}h_w(x^{(i)})\right) \cdot x_j \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} - h_w(x^{(i)})\right) \cdot x_j \\ &= \frac{1}{m}\sum_{i=1}^{m}\left(h_w(x^{(i)}) - y^{(i)}\right) \cdot x_j \end{aligned} \tag{5}$$

So the update rule for each coefficient is:

$$w_j := w_j - \alpha\frac{1}{m}\sum_{i=1}^{m}\left(h_w(x^{(i)}) - y^{(i)}\right) \cdot x_j$$

d) Assuming an initial value of 1 for first the coefficient, 2 for the second coefficient, 3 for the third coefficient, 4 for the fourth coefficient, and so on, run one iteration of gradient descent using a learning rate of 0.01.

$w_0 = 1$, $w_1 = 2$, and $w_3 = 3$,

$$w^T x^{(1)} = w_0 + w_1 \cdot x_1^{(1)} + w_2 \cdot x_2^{(1)} = 1 + 2 \cdot 45 + 3 \cdot 25 = 166$$

$$h_w(x^{(1)}) = \frac{1}{1+e^{-w^T x^{(1)}}} = \frac{1}{1+e^{-166}} = 1$$

$$w^T x^{(2)} = w_0 + w_1 \cdot x_1^{(2)} + w_2 \cdot x_2^{(2)} = 1 + 2 \cdot 60 + 3 \cdot 35 = 226$$

$$h_w(x^{(2)}) = \frac{1}{1+e^{-w^T x^{(2)}}} = \frac{1}{1+e^{-226}} = 1$$

$$w^T x^{(3)} = w_0 + w_1 \cdot x_1^{(3)} + w_2 \cdot x_2^{(3)} = 1 + 2 \cdot 75 + 3 \cdot 40 = 271$$

$$h_w(x^{(3)}) = \frac{1}{1+e^{-w^T x^{(3)}}} = \frac{1}{1+e^{-271}} = 1$$

$$w^T x^{(4)} = w_0 + w_1 \cdot x_1^{(4)} + w_2 \cdot x_2^{(4)} = 1 + 2 \cdot 50 + 3 \cdot 28 = 185$$

$$h_w(x^{(4)}) = \frac{1}{1+e^{-w^T x^{(4)}}} = \frac{1}{1+e^{-185}} = 1$$

$$w^T x^{(5)} = w_0 + w_1 \cdot x_1^{(5)} + w_2 \cdot x_2^{(5)} = 1 + 2 \cdot 90 + 3 \cdot 50 = 331$$

$$h_w(x^{(5)}) = \frac{1}{1+e^{-w^T x^{(5)}}} = \frac{1}{1+e^{-331}} = 1$$

$$w^T x^{(6)} = w_0 + w_1 \cdot x_1^{(6)} + w_2 \cdot x_2^{(6)} = 1 + 2 \cdot 100 + 3 \cdot 60 = 381$$

$$h_w(x^{(6}) = \frac{1}{1+e^{-w^T x^{(6)}}} = \frac{1}{1+e^{-381}} = 1$$

$$
\begin{aligned}
w_0 &= w_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)} \right) \\
&= 1 - (0.01) \frac{1}{6} \sum_{i=1}^{6} \left(h_w(x^{(i)}) - y^{(i)} \right) \\
&= 1 - (0.01) \frac{1}{6} \left((1-0) + (1-1) + (1-1) + (1-0) + (1-1) + (1-1) \right) \\
&= 0.997
\end{aligned}
$$

$$
\begin{aligned}
w_1 &= w_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)} \right) \\
&= 2 - (0.01) \frac{1}{6} \sum_{i=1}^{6} \left(h_w(x^{(i)}) - y^{(i)} \right) \cdot x_1^{(i)} \\
&= 2 - (0.01) \frac{1}{6} \left((1-0) \cdot 45 + (1-1) \cdot 60 + (1-1) \cdot 75 + (1-0) \cdot 50 + (1-1) \cdot 90 + (1-1) \cdot 100 \right) \\
&= 1.842
\end{aligned}
$$

$$
\begin{aligned}
w_2 &= w_2 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_w(x^{(i)}) - y^{(i)} \right) \\
&= 3 - (0.01) \frac{1}{6} \sum_{i=1}^{6} \left(h_w(x^{(i)}) - y^{(i)} \right) \cdot x_2^{(i)} \\
&= 3 - (0.01) \frac{1}{6} \left((1-0) \cdot 25 + (1-1) \cdot 35 + (1-1) \cdot 40 + (1-0) \cdot 28 + (1-1) \cdot 50 + (1-1) \cdot 60 \right) \\
&= 2.912
\end{aligned}
$$

e) Calculate the model accuracy using the updated coefficients.

Using the updated coefficients, we will predict $y = 1$ if $w^T x^{(i)} \geq 0$.

$$w^T x^{(1)} = w_0 + w_1 \cdot x_1^{(1)} + w_2 \cdot x_2^{(1)} = 0.997 + 1.842 \cdot 45 + 2.912 \cdot 25 = 156.69 \geq 0$$

$$w^T x^{(2)} = w_0 + w_1 \cdot x_1^{(2)} + w_2 \cdot x_2^{(2)} = 0.997 + 1.842 \cdot 60 + 2.912 \cdot 35 = 213.44 \geq 0$$

$$w^T x^{(3)} = w_0 + w_1 \cdot x_1^{(3)} + w_2 \cdot x_2^{(3)} = 0.997 + 1.842 \cdot 75 + 2.912 \cdot 40 = 255.63 \geq 0$$

$$w^T x^{(4)} = w_0 + w_1 \cdot x_1^{(4)} + w_2 \cdot x_2^{(4)} = 0.997 + 1.842 \cdot 50 + 2.912 \cdot 28 = 174.63 \geq 0$$

$$w^T x^{(5)} = w_0 + w_1 \cdot x_1^{(5)} + w_2 \cdot x_2^{(5)} = 0.997 + 1.842 \cdot 90 + 2.912 \cdot 50 = 312.37 \geq 0$$

$$w^T x^{(6)} = w_0 + w_1 \cdot x_1^{(6)} + w_2 \cdot x_2^{(6)} = 0.997 + 1.842 \cdot 100 + 2.912 \cdot 60 = 359.92 \geq 0$$

The model predicts $y = 1$ for all datapoints. Accordingly,

$$\text{accuracy} = \frac{4}{6} * 100 = 66.67\%$$

Question 4

You built a model that can predict whether a customer would churn (i.e., leave the company) (1 = churn, 0 = no churn) using a number of features, with one of them being x_1 (salary – in thousands of dollars). In the dataset you leveraged to build the model, only 10% of the customers churned. The remaining 90% did not churn. The confusion matrix for applying the model to a test set is as follows:

	Predicted No Churn	Predicted Churn
Actual No Churn	180	20
Actual Churn	70	30

a) Calculate the accuracy, precision, recall, and F1 score for the model.

$$\text{Accuracy} = \frac{180 + 30}{300} = 0.7$$

$$\text{Precision} = \frac{30}{30 + 20} = 0.6$$

$$\text{Recall} = \frac{30}{30 + 70} = 0.3$$

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = 0.4$$

b) Interpret the F1 score and explain when it is more useful than accuracy.

While the model's accuracy of 70% seems high, the F1 score of 40% reveals limitations in capturing positive cases. The F1 score is a more reliable metric than accuracy in this case since we are clearly dealing with a case of an imbalanced dataset.

c) Suppose the coefficient corresponding to the feature "salary" in the model you built is -0.6. For an employee with a salary of $30,000, the probability of churning is 0.63. What would the probability of churning be if the company were to increase this employee's salary to $31,000?

We know that with logistic regression, the linear combination of the weights and the inputs is the $log(odds)$.

An increase in salary to $30,000 is a unit increase in salary. Accordingly, the $log(odds)$ would decrease by 0.6.

To get the odds of churning with the current salary, we will use the given probability of 0.63.

$$odds = \frac{p(y=1)}{1-p(y=1)} = \frac{0.63}{0.37} = 1.703$$

And so the $log(odds)$ is:

$$log(odds) = log(1.703) = 0.5322$$

A decrease in the $log(odds)$ by 0.6 means the new $log(odds)$ is -0.0678. Accordingly, the new odds are:

$$odds = e^{(-0.0678)} = 0.934$$

So, the new probability of churning is:

$$p(y=1) = \frac{0.934}{1+0.934} = 0.483$$

The probability of the employee churning would drop to 0.483 if the salary were to be increased to \$31,000.

Question 5

You want to automatically decide whether today would be a good day to play tennis or not based on the day's outlook, temperature, and humidity. You have some recorded historical data, as shown below:

Outlook	Temperature	Humidity	Play Tennis (Y/N)
Sunny	*Hot*	*High*	*No*
Sunny	*Hot*	*Normal*	*Yes*
Overcast	*Hot*	*High*	*Yes*
Overcast	*Mild*	*High*	*Yes*
Sunny	*Mild*	*Normal*	*No*

a) Using entropy as the measure of impurity at each node, build a decision tree classifier that can predict whether today would be a good day to play tennis or not based on outlook, temperature, and humidity. Recall that entropy is calculated as,

$$H(p_1) = -\sum_{i=1}^{n} p_i \cdot log_2(p_i)$$

where p_1 is the fraction of datapoints belonging to class 1 (Yes to Playing Tennis).

$$p_1^{root} = \frac{3}{5} = 0.6$$

Information gain for splitting on Outlook:

$$\begin{aligned}
\text{gain}_{\text{outlook}} &= H(p_1^{root}) - (w^{left} H(p_1^{left}) + w^{right} H(p_1^{right})) \\
&= H(0.6) - (\frac{3}{5} H(\frac{1}{3}) + \frac{2}{5} H(\frac{2}{2})) \\
&= 0.42
\end{aligned}$$

Information gain for splitting on Humidity:

$$\begin{aligned}
\text{gain}_{\text{humidity}} &= H(p_1^{root}) - (w^{left} H(p_1^{left}) + w^{right} H(p_1^{right})) \\
&= H(0.6) - (\frac{3}{5} H(\frac{2}{3}) + \frac{2}{5} H(\frac{1}{2})) \\
&= 0.02
\end{aligned}$$

Information gain for splitting on Temperature:

$$\begin{aligned}\text{gain}_{\text{temperature}} &= H(p_1^{root}) - (w^{left}H(p_1^{left}) + w^{right}H(p_1^{right})) \\ &= H(0.6) - (\frac{3}{5}H(\frac{2}{3}) + \frac{2}{5}H(\frac{1}{2})) \\ &= 0.02\end{aligned}$$

Since splitting on **Outlook** results in the highest information gain, we split on **Outlook** in the root node. The **Overcast** branch results in a pure leaf node with value "Yes".

We continue building the **Sunny** branch. We repeat the same process as earlier with a new root node with 3 datapoints.

$$p_1^{root} = \frac{1}{3}$$

Information gain for splitting on Humidity:

$$\begin{aligned}\text{gain}_{\text{humidity}} &= H(p_1^{root}) - (w^{left}H(p_1^{left}) + w^{right}H(p_1^{right})) \\ &= H(\frac{1}{3}) - (\frac{2}{3}H(\frac{1}{2}) + \frac{1}{3}H(\frac{0}{1})) \\ &= 0.252\end{aligned}$$

Information gain for splitting on Temperature:

$$\begin{aligned}\text{gain}_{\text{temperature}} &= H(p_1^{root}) - (w^{left}H(p_1^{left}) + w^{right}H(p_1^{right})) \\ &= H(0.6) - (\frac{1}{3}H(\frac{0}{1}) + \frac{2}{3}H(\frac{1}{2})) \\ &= 0.252\end{aligned}$$

Splitting on **Humidity** or **Temperature** results in the same information gain, so it does not make a difference what we split on in this case. Let's split on **Humidity**. The **High** branch results in a pure leaf node with value "Yes".

We continue building the **Normal** branch. The **Outlook** feature is Sunny for both datapoints, so splitting on **Sunny** won't result in any information gain. We are only left with one feature, which is **Temperature**.

Splitting on **Temperature** results in two pure leaf nodes, one with value "No" following the **Mild** branch and another one with value "Yes" following the **Hot** branch. This concludes the process of building the decision tree.

The resulting decision tree looks like the following:

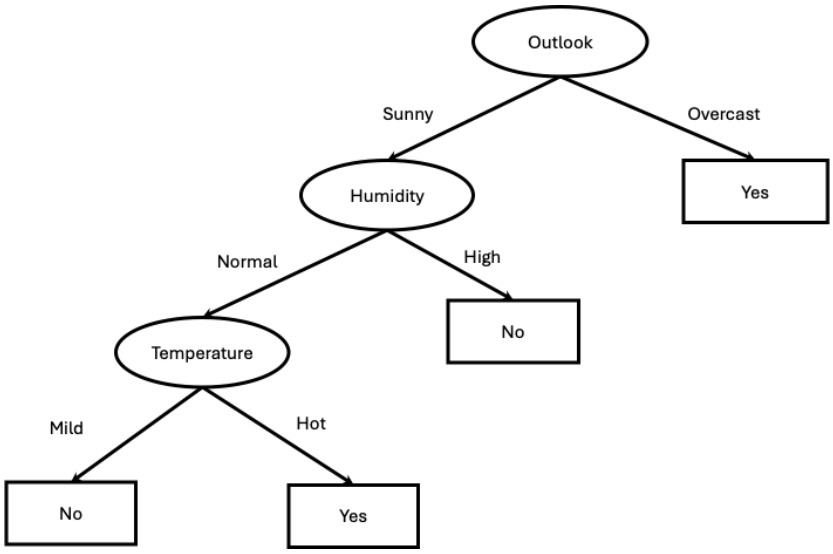

Figure 1: Resulting Decision Tree

b) Use the decision tree classifier to predict whether to play tennis or not for a day with the following features:

- **Outlook: Overcast**
- **Temperature: Hot**
- **Humidity: Normal**

Following the **Overcast** branch at the root node, we hit a pure leaf node with value "Yes". So the decision tree's prediction in this case is "Yes, play tennis".