Joscandy Nunez
CPE 695

Q.1:

Bias-variance trade-off is a machine learning model's generalization. This can lead to either underfitting (bias) or overfitting (variance) the predictions to the data, and a bigger measured error.

Bias : wrong assumptions that lead to underfitting, i.e data is linear when it is a polynomial function of degree > 1.
Variance : it uses the patterns in the training data too well, leading to overfitting., i.e too-complex models that does a great job with training data but fails to accurately predict unseen test data

Because complexity cannot be too much or too little without compromising accuracy, this "tug of war" between bias and variance is called the trade-off.

Things that can be done to reduce bias to Increase model complexity:

- Use adequate learning model (linear vs non-linear)
- Increase model size by adding more features to detect more local patterns
- Apply random sampling to make the predictions "fair" or equally given a chance

Things that can be done to reduce variance to decrease model complexity:

- Apply regularization to standardize features/decrease amplitude of coefficients
- Feature selection (drop features not needed for the model prediction)
- Increase size of data to have the model observe more global patterns

Q.2:

TP => 50

FN => 30

FP => 40

TN => 60

precision = TP/(TP + FP) = 50/(50 + 40) = 0.55555555555

recall = TP/(TP + FN) = 50/(50 + 30) = 0.625

F1 = 2 * (0.55555555555 * 0.625)/(0.55555555555 + 0.625)

F1 = 0.58823529411

Joscandy Nunez
CPE 695

Q3.
Do entropy for all classes and use that to obtain gain for each feature :
Root node [PlayTennis] entropy:
probability(Yes) = 6 / 10 = 0.6
probability(No) = 4 / 10 = 0.4
using log base 2 :
E = - (0.6 * log(0.6) + 0.4 * log(0.4)) = - (0.6 * -0.737 + 0.4 * -1.322) = 0.971

Find information gain for wind:
E (weak) = - (5/7 * log(5/7) + 2/7 * log(2/7)) = - (5/7 * -0.4854 + 2/7 * -1.8074) = 0.86311428571
E (strong) = - (1/3 * log(1/3) + 2/3 * log(2/3)) = - (1/3 * -1.585 + 2/3 * -0.585) = 0.91833333333
E = 0.87967999999
I.G = parent E - child E = 0.971 - 7/10 * 0.86311428571 - 3/10 * 0.91833333333
= 0.09132

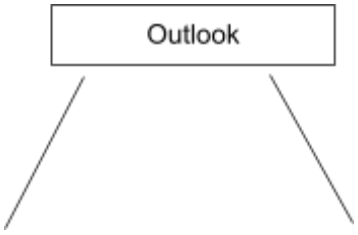
Find information gain for humidity:
E (high) = - (2/5 * log(2/5) + 3/5 * log(3/5)) = - (2/5 * -1.322 + 3/5 * -0.737) = 0.971
E (normal) = - (4/5 * log(4/5) + 1/5 * log(1/5)) = - (4/5 * -0.3219 + 1/5 * -2.322) = 0.72192
E = 0.84646
I.G = parent E - child E = 0.971 - 5/10 * 0.72192 - 5/10 * 0.971
= 0.12454

Find information gain for temperature:
E (hot) = - (1/4 * log(1/4) + 3/4 * log(3/4)) = - (1/4 * -2 + 3/4 *-0.415) = 0.81125
E (cool) = - (3/4 * log(3/4) + 1/4 * log(1/4)) = - (3/4 * -0.415 + 1/4 * -2) = 0.81125
E (mild) = - (⅔ * log(⅔) + ⅓ * log(⅓)) = - (⅔ * -0.585 + ⅓ * -1.585) = 0.91833333333
E = 0.87549999999
I.G = parent E - child E
= 0.971 - 3/10 * 0.91833333333 - 4/10 * 0.81125 - 3/10 * 0.91833333333
= 0.0955

Find information gain for outlook:
E (sunny) = - (1/3 * log(1/3) + 2/3 * log(2/3)) = - (1/3 * -1.585 + 2/3 * -0.585) = 0.91833333333
E (overcast) = - (2/2 * log(2/2) + 0/2 * log(0/2)) = - (0 + 0) = 0
E (rain) = - (1/4 * log(1/4) + 3/4 * log(3/4)) = - (1/4 * -2 + 3/4 * -0.415) = 0.81125
E = 0.69999999999
I.G = parent E - child E = 0.971 - 3/10 * 0.91833333333 - 0 - 4/10 * 0.81125
= 0.371

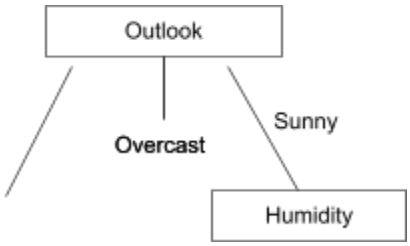
Outlook shows the most gain so it becomes parent node

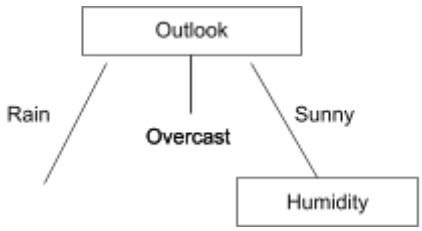
Joscandy Nunez
CPE 695



First, there is overcast entropy = 0 so no gain, and this will be a leaf node.
Now, building child nodes for every attribute in outlook:
Get information gain for child nodes :
{D1, D2, D8, D9}
IG of humidity: Outlook E(Sunny) - Humidity E = 0.91833333333 - 3/4 * 0 - 1/4 * 0 =
= 0.91833333333

For Outlook E(sunny), humidity has the most gain possible. Humidity becomes next child node

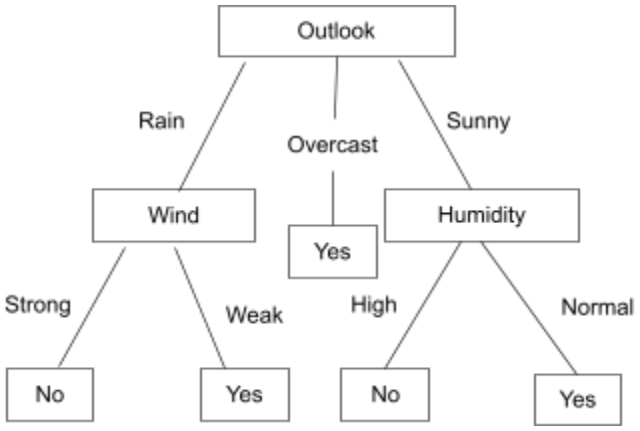




Child node of Rain:
{D4, D5, D6, D10}
IG of wind: Outlook E(Rain) - Wind E = 0.81125 - 3/4 * 0 - 1/4 * 0= 0.81125
The outlook E(Rain) corresponds to having the child node as wind because then it will have the most gain possible.

Joscandy Nunez
CPE 695

Decision Tree:



Q.4:
Class 1: (40/70) * (20/40) * (0/10) = 0
Class 2: (30/70) * (20/40) * (10/10) = **0.21428571428**
So final prediction is class 2