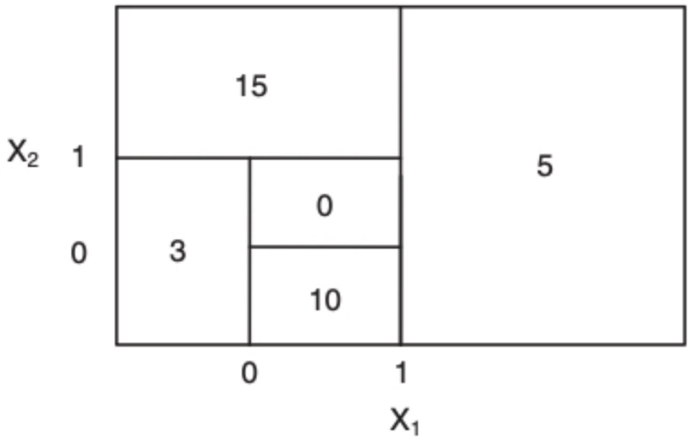
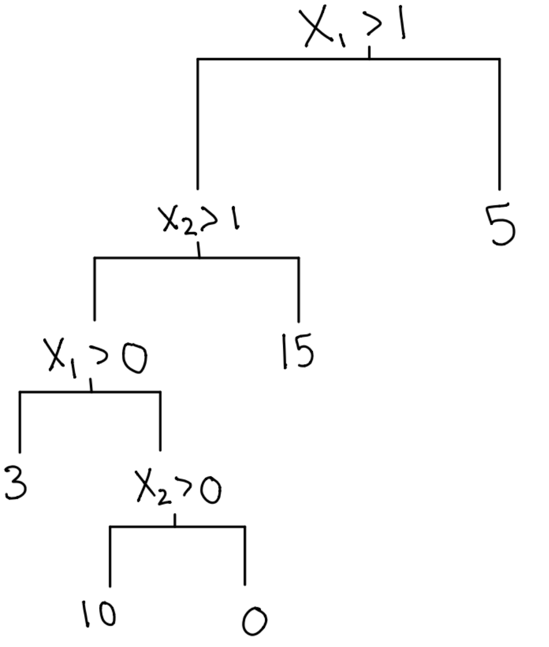
ESE 541 Week 9 & 10 Practice Problems - Solutions

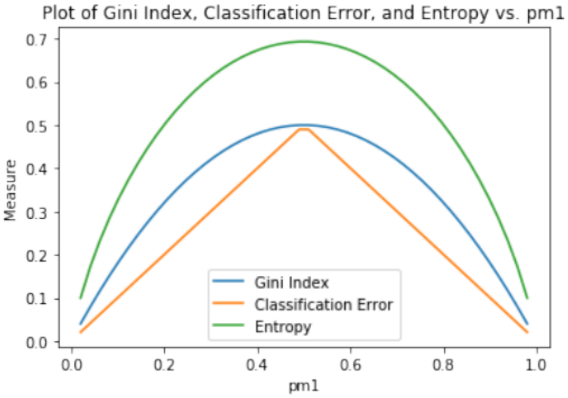
**Problem 1**. Draw an example (of your own invention) of a partition of two-dimensional feature space that could result from recursive binary splitting. Your example should contain at least five regions. Draw a decision tree corresponding to this partition. Be sure to label all aspects of your figures, including the regions, the cutpoints, and so forth.





**Problem 2**. Consider the Gini index, classification error, and entropy in a simple classification setting with two classes. Create a single plot that displays each of these quantities as a function of $\hat{p}_{m1}$. The $x$-axis should display $\hat{p}_{m1}$, ranging from 0 to 1, and the $y$-axis should display the value of the Gini index, classification error, and entropy.

```
import numpy as np
import matplotlib.pyplot as plt

pm1s =  np.linspace(start=0, stop=1) #Default length = 50

# Remove 0 and 1 to compute logarithms in entropy
pm1s = pm1s[1:-1] # Length is now 48
pm0s = 1 - pm1s

# Calculate Gini indexes
# We multiply by 2 since both terms in the gini index over the k classes
# are identical due to there only being two classes
gini_indexes = [2*(pm1s[i] * pm0s[i]) for i in range(48)]

# Calulcate classification error
class_errors = [(1- max(pm1s[i], pm0s[i])) for i in range(48)]

# Calculate entropy
entropies = [(-pm1s[i] * np.log(pm1s[i]) - pm0s[i] * np.log(pm0s[i])) for i in␣
 ↪range(48)]

plt.plot(pm1s, gini_indexes, pm1s, class_errors, pm1s, entropies)
plt.legend(['Gini Index', 'Classification Error', 'Entropy'])
plt.title('Plot of Gini Index, Classification Error, and Entropy vs. pm1')
plt.xlabel('pm1')
plt.ylabel('Measure')
```



**Problem 3**. Answer the following questions (and make sure you justify your answers):

1. *True* or *False*: It is not possible to avoid overfitting in a Random Forest by pruning each one of the trees in the forest.

   False. Although it is not computationally efficient, it is possible – in theory– to avoid overfitting by prunning the trees in a random forest.

2. *True* or *False*: The trees used in Bagging are more flexible than the trees used in Random Forests.

   True. The trees in RF are more constrained in their construction.

**Problem 4**. Consider a dataset with two input features, **GPA** and **Gender**, and one output variable **Salary** after graduation (in thousands of dollars). The dataset has five points represented in the table below:

| Obs. | Gender | GPA | Salary |
|---|---|---|---|
| 1 | Male | 3.0 | 82 |
| 2 | Female | 4.0 | 88 |
| 3 | Male | 4.0 | 90 |
| 4 | Female | 3.0 | 84 |
| 5 | Male | 3.0 | 80 |

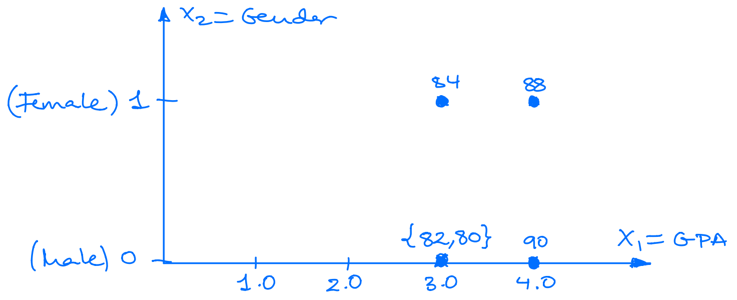
Define the quantitative variables $X_1$ = GPA and the qualitative variable

$$X_2 = \begin{cases} 0 & \text{for Males,} \\ 1 & \text{for Females.} \end{cases}$$

Answer the questions below:

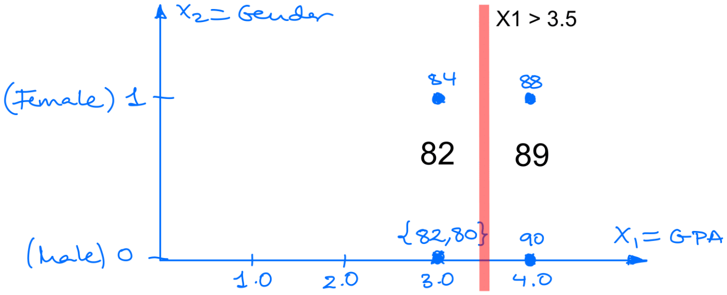
1. Sketch the dataset on the plane $X_1/X_2$.

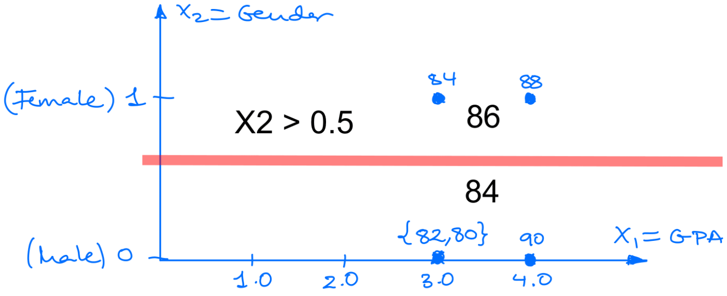
   Sketch below:



2. On your sketch, add two possible decision rules (partitions). For each partition, calculate the optimal regression value for each region.

Below are two possible partitions. We can calculate the optimal regression values by finding the average of the datapoints within the resulting regions.





3. If our goal is to achieve the minimum training RSS, how many leaves should our regression tree have?

We can achieve a training RSS of 2 if we use **four** leaves (one for each unique [Gender, GPA] observation). In general, a tree with $n_\mathrm{u}$ leaves (where $n_\mathrm{u}$ is the number of unique observations) will achieve the minimal training RSS if trained properly.