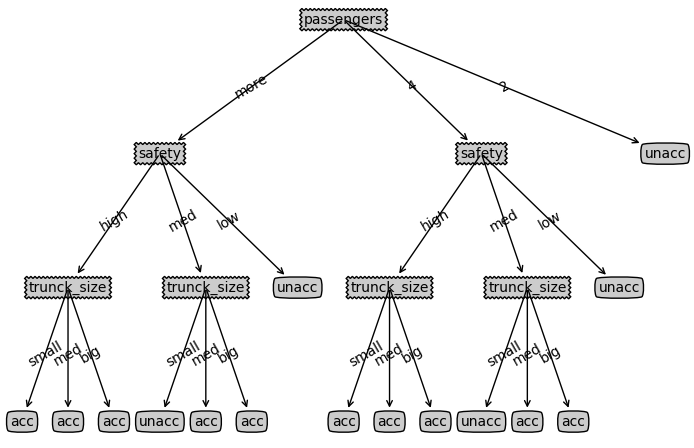
ECE 49595 Assignment 3

Tree Drawn from Car Dataset



chooseBestFeature Code

def gini(labels):

    \_, count = np.unique(labels, return\_counts=True)

    probs = count / len(labels)

    gini\_idx = 1 - np.sum(probs \*\* 2)

    return gini\_idx

def chooseBestFeature(dataSet):

    '''

    choose best feature to split based on Gini index

    Parameters

    -----------------

    dataSet: 2-D list

        [n\_sampels, m\_features + 1]

        the last column is class label

    Returns

    ------------------

    bestFeatId: int

        index of the best feature

    '''

    best\_gain = 0

    bestFeatId = -1

    num\_features = len(dataSet[0]) - 1

    total\_samples = len(dataSet)

    for ID in range(num\_features):

        gini\_idx = gini([x[-1] for x in dataSet])

        unique\_vals = set([x[ID] for x in dataSet])

        for val in unique\_vals:

            subset = [x for x in dataSet if x[ID] == val]

            gini\_idx\_subset = gini([x[-1] for x in subset])

            gain = gini\_idx - (len(subset) / total\_samples) \* gini\_idx\_subset

            if gain > best\_gain:

                best\_gain = gain

                bestFeatId = ID

    return bestFeatId

stopCriteria Code

def stopCriteria(dataSet):

    '''

    Criteria to stop splitting:

    1) if all the classe labels are the same, then return the class label;

    2) if there are no more features to split, then return the majority label of the subset.

    Parameters

    -----------------

    dataSet: 2-D list

        [n\_sampels, m\_features + 1]

        the last column is class label

    Returns

    ------------------

    assignedLabel: string

        if satisfying stop criteria, assignedLabel is the assigned class label;

        else, assignedLabel is None

    '''

    assignedLabel = None

    labels = [x[-1] for x in dataSet]

    unique\_labels = np.unique(labels)

    if len(unique\_labels) == 1:

        assignedLabel = unique\_labels[0]

    elif len(dataSet[0]) == 1:

        assignedLabel = max(set(labels), key=labels.count)

    return assignedLabel