COMP9414: Artificial Intelligence
Tutorial Week 7: Machine Learning

1. Construct a Decision Tree for the following set of examples.

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

   What class is assigned to the instance {D15, Sunny, Hot, High, Weak}?

2. Consider a Naive Bayes classifier for the same set of examples. What class is now assigned to the instance {D15, Sunny, Hot, High, Weak}?

3. **Programmming.** Try out the Naive Bayes classifier from NLTK. Here you can load a set of documents, convert them into features, then train and test the classifier. The example below is for moview reviews. The categories are 'neg' and 'pos' and the document features are True or False depending on whether a word is contained in the document.

```
from nltk import FreqDist, NaiveBayesClassifier
from nltk.corpus import movie_reviews
from nltk.classify import accuracy
import random

documents = [(list(movie_reviews.words(fileid)), category)
      for category in movie_reviews.categories()
      for fileid in movie_reviews.fileids(category)]
random.shuffle(documents) # This line shuffles the order of the documents
all_words = FreqDist(w.lower() for w in movie_reviews.words())
word_features = list(all_words)[:2000]

def document_features(document):
    document_words = set(document)
    features = {}
    for word in word_features:
        features['contains({})'.format(word)] = (word in document_words)
    return features

featuresets = [(document_features(d), c) for (d,c) in documents]
train_set, test_set = featuresets[100:], featuresets[:100] # Split data
classifier = NaiveBayesClassifier.train(train_set)
print(accuracy(classifier, test_set))
```