ATTENTION

I fail to run my code in gradescope, so please run it for me manually. Thank you!
codes based on python 3.6
Strongly recommend to use Pycharm .
Run Steps:

(1) Put NaiveBayesClassifier.py , training.txt , testing.txt into one document.
After

```
training_trans("training.txt")
testing_trans("testing.txt")
write_raw_index("training.csv")
write_raw_index("testing.csv")
```

There will be training.csv and testing.csv in the same document.

(2) Change the data path in line 124 and 125. (below)

```
train_data = pd.read_csv
("C:\\Users\\13703\\Documents\\learning\\exchange\\cs165a
\\mp1\\training.csv")
# Load training data
test_data = pd.read_csv
("C:\\Users\\13703\\Documents\\learning\\exchange\\cs165a
\\mp1\\testing.csv")
# Load testing data
```

Change the data path to where the training.csv and testing.csv locate.

(3) switch the test data from training.txt and testing.txt(line 120)

```
testing_trans("testing.txt")
# choose a testing data.
# if use training data as test data, it should be "training.txt"
```

(4) Run NaiveBayesClassifier.py

Architecture

(1) Prepare for the dataset

```
# translate .txt into .csv
def training_trans(txt): #translate training.txt
  return training.csv
```

```
def testing_trans(txt): # translate texting.txt
  return training.csv
```

```
def write_raw_index(file) #add index on the top of .csv
```

(2) Calculate Probabilities
Prior probabilities

```
def prior_prob(train_data): #Calculate Prior probabilities
                            #when Rain Tomorrow is Yes and No.
```

Continuous data probability

```
def calculate_prob(mean, std, x): # Gaussian distribution
```

Discrete data probability

```
 def calculate_discrete_variable_prob(attr, val, tot_rows):
 # discrete data probability
```

(3) Predict

```
def predict(test_data, prior_yes_prob, prior_no_prob):
```

Preprocessing

(1) Translate .txt into .csv

(2) Add index at the top of .csv

Model Building

(1) Decide whether the attribute is continuous or discrete.

(2) If the attribute is continuous, calculate the probability separately for different attributes and different "yes" or "no" in Rain Tomorrow by Gaussian Probability formula.

(3) If the attribute is discrete, everything is the same with when the attribute is continuous, expect the calculate formula. The Probability formula for discrete attributes is $\frac{the number of selected data}{the number of total data}$

(4) Multiply all the probabilities, and the conclusion should also multiply the prior probability
$P(RainTomorrow = Yes)$and $P(RainTomorrow = No)$

(5) If the probability of $P(RainTomorrow = Yes) > P(RainTomorrow = No)$ the predict label is 1, which means tomorrow will rain. Else it is 0, which means tomorrow will not rain.

Results

When test data is testing.txt:

(1) Accuracy:
0.7807723250201126

(2) Running time:
118.49517893791199 s

When test data is training.txt:

(1) Accuracy:
0.7695905844482447

(2) Running time:
912.9175615310669 s
There is too much data so that the time is much longer. The running time is beyond 10 minutes.

Challenges

(1) The training data and the testing data are both in the type .txt . In order to classify them, we need to transform them into .csv . What's more we need to add a top as the first line in order to show what attribute they belong with.

(2) Among the data set, there are continuous attributes and discrete attributes. The ways to calculate the probabilities are different,so we need to separate the continuous attributes and discrete attributes.

Weaknesses

There is no smoothing process. So when there is a data that appears in the testing data but not in the training data, the probability of this data is 0. To overcome it, we can add a Laplacian smoothing so that every data has it's probability.