**Question 1**

"self.tokenize = nltk.tokenize.word\_tokenize"

training accuracy: 0.7075333333333333

validation accuracy: 0.6943333333333334

testing accuracy: 0.70555

"self.tokenize = nltk.tokenize.WhitespaceTokenizer().tokenize "

training accuracy: 0.7212

validation accuracy: 0.7112666666666667

testing accuracy: 0.70925

"self.tokenize = nltk.tokenize.SpaceTokenizer().tokenize"

training accuracy: 0.7211333333333333

validation accuracy: 0.7112666666666667

testing accuracy: 0.7092

(training set: 0.2, validation set: 0.3, testing set: 0.5)

training accuracy: 0.723

validation accuracy: 0.7086

testing accuracy: 0.7102

(training set: 0.3, validation set: 0.3, testing set: 0.4)

training accuracy: 0.7212

validation accuracy: 0.7112666666666667

testing accuracy: 0.70925

(training set: 0.4, validation set: 0.3, testing set: 0.3)

training accuracy: 0.72045

validation accuracy: 0.7080666666666666

testing accuracy: 0.7135333333333334

(training set: 0.5, validation set: 0.3, testing set: 0.2)

training accuracy: 0.71796

validation accuracy: 0.7099333333333333

testing accuracy: 0.7168

(training set: 0.6, validation set: 0.3, testing set: 0.3)

training accuracy: 0.7177333333333333

validation accuracy: 0.7078666666666666

testing accuracy: 0.7262

(training set: 0.3, validation set: 0.2, testing set: 0.5)

training accuracy: 0.7212

validation accuracy: 0.7103

testing accuracy: 0.71004

(training set: 0.3, validation set: 0.3, testing set: 0.4

model = LogisticRegression(solver=' lbfgs')

training accuracy: 0.723

validation accuracy: 0.7086

testing accuracy: 0.7102

(training set: 0.3, validation set: 0.3, testing set: 0.4)

model = LogisticRegression(solver='newton-cg')

training accuracy: 0.7212

validation accuracy: 0.7113

testing accuracy: 0.7093

(training set: 0.3, validation set: 0.3, testing set: 0.4)

model = LogisticRegression(solver='liblinear')

training accuracy: 0.7212

validation accuracy: 0.7112

testing accuracy: 0.70905

(training set: 0.3, validation set: 0.3, testing set: 0.4)

model = LogisticRegression(solver='sag')

training accuracy: 0.7212

validation accuracy: 0.7112

testing accuracy: 0.7093

(training set: 0.3, validation set: 0.3, testing set: 0.4)

model = LogisticRegression(solver='saga')

training accuracy: 0.7213

validation accuracy: 0.7107

testing accuracy: 0.709

model = LogisticRegression(penalty='elasticnet', solver='saga', l1\_ratio=0.1)

training accuracy: 0.7210666666666666

validation accuracy: 0.7106

testing accuracy: 0.70935

model = LogisticRegression(penalty='elasticnet', solver='saga', l1\_ratio=0.3)

training accuracy: 0.721

validation accuracy: 0.7108666666666666

testing accuracy: 0.70975

model = LogisticRegression(penalty='elasticnet', solver='saga', l1\_ratio=0.5)

training accuracy: 0.7208666666666667

validation accuracy: 0.7106666666666667

testing accuracy: 0.7098

model = LogisticRegression(penalty='elasticnet', solver='saga', l1\_ratio=0.8)

training accuracy: 0.7212

validation accuracy: 0.7101333333333333

testing accuracy: 0.709

model = LogisticRegression(penalty='elasticnet', solver='saga', l1\_ratio=1.0)

training accuracy: 0.7210666666666666

validation accuracy: 0.7108

testing accuracy: 0.7088

Q2