

```
In [1]: import pandas as pd
```

```
In [2]: df = pd.read_csv("SMSSpamCollection", sep="\t", names=['label', 'message'])  
df.head()
```

Out[2]:

	label	message
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

## Text Cleaning

```
In [3]: import nltk  
import re  
from nltk.corpus import stopwords  
from nltk.stem import PorterStemmer  
ps = PorterStemmer()
```

```
In [4]: corpus=[]  
for i in range(len(df)):  
    rp = re.sub('[^a-zA-Z]', " ", df['message'][i])  
    rp = rp.lower()  
    rp = rp.split()  
    rp = [ps.stem(word) for word in rp if not word in set(stopwords.words('english'))]  
    rp = " ".join(rp)  
    corpus.append(rp)
```

## Vectorization

```
In [5]: from sklearn.feature_extraction.text import CountVectorizer  
cv = CountVectorizer()  
X = cv.fit_transform(corpus).toarray()
```

```
In [6]: y=pd.get_dummies(df['label'],drop_first=True)
```

### Train-Test Split

```
In [7]: from sklearn.model_selection import train_test_split  
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.3,random_state=0)
```

## Modeling

## Navie Bayes Classifier with default parameters

```
In [8]: from sklearn.naive_bayes import MultinomialNB  
model=MultinomialNB()  
model.fit(X_train,y_train)
```

Out[8]: MultinomialNB()

### Predictions

```
In [9]: ypred_test = model.predict(X_test)  
ypred_train = model.predict(X_train)
```

### Evaluation

```
In [10]: from sklearn.metrics import accuracy_score  
print("Train Accuracy:",accuracy_score(y_train,ypred_train))  
print("Test Accuracy:",accuracy_score(y_test,ypred_test))
```

Train Accuracy: 0.9912820512820513

Test Accuracy: 0.9796650717703349