## # # implementation and Comparing Multinomial and Gaussian Naive Bayes

## **#Naive Bayes using multinomial distribution**

## #**#Naive Bayes using Gaussian distribution**

## # scikit-learn documentation: [MultinomialNB](http://scikit-#learn.org/stable/modules/generated/sklearn.naive\_bayes.MultinomialNB.html) and #[GaussianNB](http://scikit-#learn.org/stable/modules/generated/sklearn.naive\_bayes.GaussianNB.html)

## #

## # Dataset: [Pima Indians #Diabetes](https://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes) from the #UCI Machine Learning Repository

## # read the data

## import pandas as pd

## url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/pima-indians-diabetes/pima-indians-diabetes.data'

## col\_names = ['pregnant', 'glucose', 'bp', 'skin', 'insulin', 'bmi', 'pedigree', 'age', 'label']

## pima = pd.read\_csv(url, header=None, names=col\_names)

## # notice that all features are continuous

## pima.head()

## # create X and y

## X = pima.drop('label', axis=1)

## y = pima.label

## # split into training and testing sets

## from sklearn.cross\_validation import train\_test\_split

## X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, random\_state=1)

## # import both Multinomial and Gaussian Naive Bayes

## from sklearn.naive\_bayes import MultinomialNB, GaussianNB

## from sklearn import metrics

## **#Naive Bayes using multinomial distribution**

## # testing accuracy of Multinomial Naive Bayes

## mnb = MultinomialNB()

## mnb.fit(X\_train, y\_train)

## y\_pred\_class = mnb.predict(X\_test)

## print metrics.accuracy\_score(y\_test, y\_pred\_class)

## **#Naive Bayes using Gaussian distribution**

## # testing accuracy of Gaussian Naive Bayes

## gnb = GaussianNB()

## gnb.fit(X\_train, y\_train)

## y\_pred\_class = gnb.predict(X\_test)

## print metrics.accuracy\_score(y\_test, y\_pred\_class)

## # \*\*Conclusion:\*\* When applying Naive Bayes classification to a dataset with \*\*continuous features\*\*, it is better to use Gaussian Naive Bayes than Multinomial Naive Bayes. The latter is suitable for datasets containing \*\*discrete features\*\* (e.g., word counts).

## #

## # Wikipedia has a short [description](https://en.wikipedia.org/wiki/Naive\_Bayes\_classifier#Gaussian\_naive\_Bayes) of Gaussian Naive Bayes, as well as an excellent [example](https://en.wikipedia.org/wiki/Naive\_Bayes\_classifier#Sex\_classification) of its usage.