# Naive Bayes in sci-kit learn

Dr. Ashish Tendulkar

**IIT Madras** 

**Machine Learning Practice** 

# Naive Bayes Classifier

## **Naive Bayes classifier**

 Naive Bayes classifier applies Bayes' theorem with the "naive" assumption of conditional independence between every pair of features given the value of the class variable.

For a given class variable y and dependent feature vector  $x_1$  through  $x_m$ ,

the naive conditional independence assumption is given by:

$$P(x_i|y,x_1,...,x_{i-1},x_{i+1},...,x_m) = P(x_i|y)$$

Naive Bayes learners and classifiers can be extremely fast compared to more sophisticated methods.

#### List of NB Classifiers

Implemented in sklearn.naive\_bayes module

GaussianNB

BernoulliNB CategoricalNB

MultinomialNB ComplementNB

- Implements fit method to estimate parameters of NB classifier with feature matrix and labels as inputs.
- The prediction is performed using predict method.

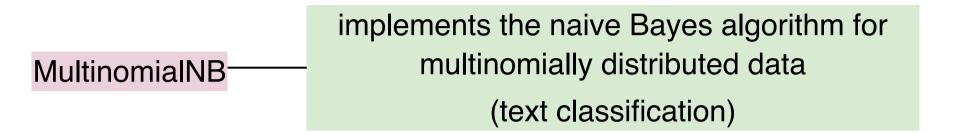
## Which NB to use if data is only numerical?

```
GaussianNB — implements the Gaussian Naive Bayes algorithm for classification
```

Instantiate a GaussianNBClassifer estimator and then call fit method using X\_train and y\_train.

```
1 from sklearn.naive_bayes import GaussianNB
2 gnb = GaussianNB()
3 gnb.fit(X_train, y_train)
```

#### Which NB to use if data is multinomially distributed?



Instantiate a MultinomialNBClassifer estimator and then call fit method using X\_train and y\_train.

```
1 from sklearn.naive_bayes import MultinomialNB
2 mnb = MultinomialNB()
3 mnb.fit(X_train, y_train)
```

#### What to do if data is imbalanced?

ComplementNB implements the complement naive Bayes (CNB) algorithm.

Instantiate a ComplementNBClassifer estimator and then call fit method using X\_train and y\_train.

```
1 from sklearn.naive_bayes import ComplementNB
2 cnb = ComplementNB()
3 cnb.fit(X_train, y_train)
```

CNB regularly outperforms MNB (often by a considerable margin) on text classification tasks.

# What to do if data has multivariate Bernoulli distributions?

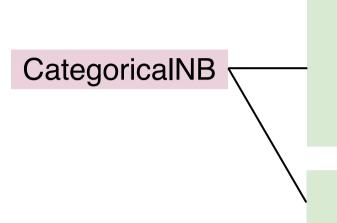
#### BernoulliNB

- implements the naive Bayes algorithm for data that is distributed according to multivariate Bernoulli distributions
- each feature is assumed to be a binaryvalued (Bernoulli, boolean) variable

Instantiate a BernoulliNBClassifer estimator and then call fit method using X\_train and y\_train.

```
1 from sklearn.naive_bayes import BernoulliNB
2 bnb = BernoulliNB()
3 bnb.fit(X_train, y_train)
```

## What to do if data is categorical?



implements the categorical naive Bayes algorithm suitable for classification with discrete features that are categorically distributed

assumes that each feature, which is described by the index i, has its own categorical distribution.

Instantiate a CategoricalNBClassifer estimator and then call fit method using X\_train and y\_train.

```
1 from sklearn.naive_bayes import CategoricalNB
2 canb = CategoricalNB()
3 canb.fit(X_train, y_train)
```