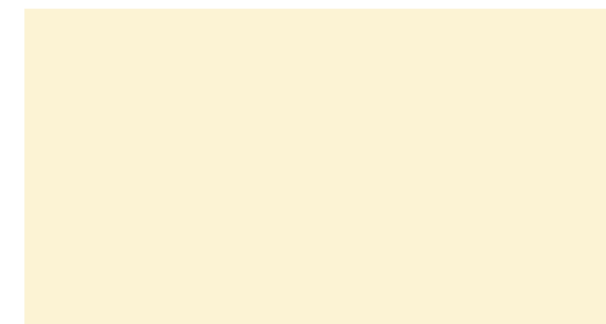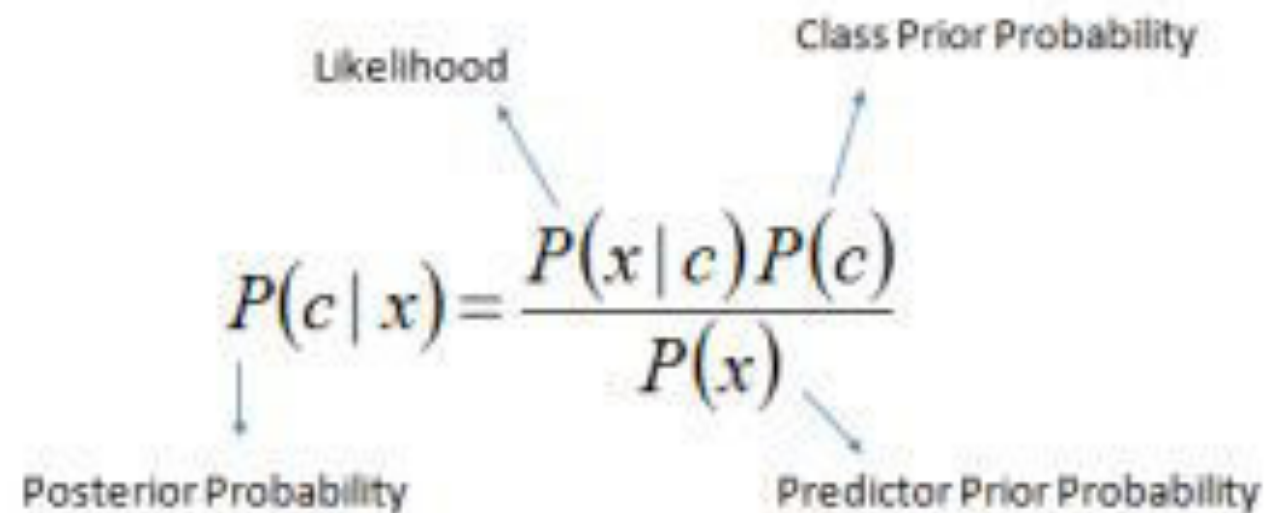# Naive Bayes

Naive Bayes - Introduction

Naive Bayes - (Sometime aka Stupid Bayes :))
- Classification technique based on Bayes' Theorem
- With "naive" assumption of independence among predictors.
- Easy to build
- Particularly useful for very large data sets
- Known to outperform even highly sophisticated classification methods
  a. e.g. Earlier method for spam detection

Introduction - Bayes theorem

- P(c|x) - the posterior probability of class (c, target) given predictor (x, attributes).
- P(c)   - the prior probability of class.
- P(x|c) - is the likelihood which is the probability of predictor given class.
- P(x)   - is the prior probability of predictor.

$$P(c \mid x) = \frac{P(x \mid c) P(c)}{P(x)}$$

Likelihood

Class Prior Probability

Posterior Probability

Predictor Prior Probability

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

How Naive Bayes algorithm works

- We are training data set of weather and corresponding target variable 'Play' (suggesting possibilities of playing).
- Now, we need to classify whether players will play or not based on weather condition.

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

How Naive Bayes algorithm works

1. Convert the data set into a frequency table
2. Create Likelihood table by finding the probabilities like Overcast probability = 0.29 and probability of playing is 0.64.
3. Now, use Naive Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

How Naive Bayes algorithm works

Problem: Players will play if weather is sunny. Is this statement is correct?

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

How Naive Bayes algorithm works

Problem: Players will play if weather is sunny. Is this statement is correct?

```
P(Yes | Sunny) = P( Sunny | Yes) * P(Yes) / P (Sunny)
```

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

How Naive Bayes algorithm works

Problem: Players will play if weather is sunny. Is this statement is correct?

```
P(Yes | Sunny) = P( Sunny | Yes) * P(Yes) / P (Sunny)
Here we have
    P (Sunny |Yes) = 3/9 = 0.33,
    P(Sunny) = 5/14 = 0.36,
    P( Yes)= 9/14 = 0.64
```

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

How Naive Bayes algorithm works

Problem: Players will play if weather is sunny. Is this statement is correct?

```
P(Yes | Sunny) = P( Sunny | Yes) * P(Yes) / P (Sunny)
Here we have
   P (Sunny |Yes) = 3/9 = 0.33,
   P(Sunny) = 5/14 = 0.36,
   P( Yes)= 9/14 = 0.64
Now, P (Yes | Sunny) = 0.33 * 0.64 / 0.36 = 0.60, (high probability)
```

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

How Naive Bayes algorithm works

Problem: Players will play if weather is sunny. Is this statement is correct?

```
P(No | Sunny) = P( Sunny | No) * P(No) / P (Sunny)
Here we have
   P (Sunny |No) = 2/5 = 0.4,
   P(Sunny) = 5/14 = 0.36,
   P( No)= 5/14 = 0.36
Now, P (No | Sunny) = 0.4 * 0.36 / 0.36 = 0.40, (low probability)
```

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

How Naive Bayes algorithm works

For Outlook = Rainy, Temp = Mild, Humidity = Normal, Windy = True, YES or No?

Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Problem2:   Rainy   Mild   Normal   True   ???

How Naive Bayes algorithm works

Lets prepare frequency tables of each feature

Frequency Tables

Outlook		Play Golf Yes	Play Golf No
		Yes	No
	Sunny	3 3/9	2 2/5
	Overcast	4 4/9	0 0/5
	Rainy	2 2/9	3 3/5

Temp.		Play Golf Yes	Play Golf No
		Yes	No
	Hot	2 2/9	2 2/5
	Mild	4 4/9	2 2/5
	Cool	3 3/9	1 1/5

Humidity		Play Golf Yes	Play Golf No
		Yes	No
	High	3 3/9	4 4/9
	Normal	6 6/9	1 1/5

Windy		Play Golf Yes	Play Golf No
		Yes	No
	False	6 6/9	2 2/5
	True	3 3/9	3 3/5

How Naive Bayes algorithm works

Outlook	★	Play Golf	
		Yes	No
	Sunny	3 3/9	2 2/5
Outlook	Overcast	4 4/9	0 0/5
	Rainy	2 2/9	3 3/5

Temp.		Play Golf	
		Yes	No
	Hot	2 2/9	2 2/5
Temp.	Mild	4 4/9	2 2/5
	Cool	3 3/9	1 1/5

Humidity		Play Golf	
		Yes	No
	High	3 3/9	4 4/9
Humidity	Normal	6 6/9	1 1/5

Windy		Play Golf	
		Yes	No
	False	6 6/9	2 2/5
Windy	True	3 3/9	3 3/5

Problem2: For Outlook = Rainy, Temp = Mild, Humidity = Normal, Windy = True, YES or No?

How Naive Bayes algorithm works

	⭐	Play Golf	
		Yes	No
Outlook	Sunny	3 3/9	2 2/5
	Overcast	4 4/9	0 0/5
	Rainy	2 2/9	3 3/5

		Play Golf	
		Yes	No
Temp.	Hot	2 2/9	2 2/5
	Mild	4 4/9	2 2/5
	Cool	3 3/9	1 1/5

		Play Golf	
		Yes	No
Humidity	High	3 3/9	4 4/9
	Normal	6 6/9	1 1/5

		Play Golf	
		Yes	No
Windy	False	6 6/9	2 2/5
	True	3 3/9	3 3/5

Problem2: For Outlook = Rainy, Temp = Mild, Humidity = Normal, Windy = True, YES or No?

Likelihood of yes = P(Outlook = Rainy|Yes) * P(Temp = Mild| Yes) * P(Humidity = Normal|Yes)*P(Windy = True|Yes)*P(Yes)
= ?

How Naive Bayes algorithm works

		Play Golf	
	★	Yes	No
Outlook	Sunny	3 3/9	2 2/5
	Overcast	4 4/9	0 0/5
	Rainy	2 2/9	3 3/5

		Play Golf	
		Yes	No
Temp.	Hot	2 2/9	2 2/5
	Mild	4 4/9	2 2/5
	Cool	3 3/9	1 1/5

		Play Golf	
		Yes	No
Humidity	High	3 3/9	4 4/9
	Normal	6 6/9	1 1/5

		Play Golf	
		Yes	No
Windy	False	6 6/9	2 2/5
	True	3 3/9	3 3/5

Problem2: For Outlook = Rainy, Temp = Mild, Humidity = Normal, Windy = True, YES or No?

Likelihood of yes = P(Outlook = Rainy|Yes) * P(Temp = Mild| Yes) * P(Humidity = Normal|Yes)*P(Windy = True|Yes)*P(Yes)

= 2/9 * 4/9 * 6/9 * 3/9 * 9/14

= **0.0141**

How Naive Bayes algorithm works

	★	Play Golf	
		Yes	No
Outlook	Sunny	3 3/9	2 2/5
	Overcast	4 4/9	0 0/5
	Rainy	2 2/9	3 3/5

		Play Golf	
		Yes	No
Temp.	Hot	2 2/9	2 2/5
	Mild	4 4/9	2 2/5
	Cool	3 3/9	1 1/5

		Play Golf	
		Yes	No
Humidity	High	3 3/9	4 4/9
	Normal	6 6/9	1 1/5

		Play Golf	
		Yes	No
Windy	False	6 6/9	2 2/5
	True	3 3/9	3 3/5

Problem2: For Outlook = Rainy, Temp = Mild, Humidity = Normal, Windy = True, YES or No?

Likelihood of yes = P(Outlook = Rainy|Yes) * P(Temp = Mild| Yes) * P(Humidity = Normal|Yes)*P(Windy = True|Yes)*P(Yes)

$\quad$ = $\quad$ 2/9 $\quad$ * $\quad$ 4/9 $\quad$ * $\quad$ 6/9 $\quad$ * 3/9 $\quad$ * 9/14

$\quad$ = **0.0141**

Likelihood of No = P(Outlook = Rainy|No) * P(Temp = Mild| No) * P(Humidity = Normal|No)*P(Windy = True|No)*P(No)

$\quad$ = $\quad$ 3/5 $\quad$ * $\quad$ 2/5 $\quad$ * $\quad$ 1/5 $\quad$ * 3/5 $\quad$ * 5/14

$\quad$ = 0.0103

How Naive Bayes algorithm works

	★	Play Golf	
		Yes	No
Outlook	Sunny	3 3/9	2 2/5
	Overcast	4 4/9	0 0/5
	Rainy	2 2/9	3 3/5

		Play Golf	
		Yes	No
Temp.	Hot	2 2/9	2 2/5
	Mild	4 4/9	2 2/5
	Cool	3 3/9	1 1/5

		Play Golf	
		Yes	No
Humidity	High	3 3/9	4 4/9
	Normal	6 6/9	1 1/5

		Play Golf	
		Yes	No
Windy	False	6 6/9	2 2/5
	True	3 3/9	3 3/5

Problem2: For Outlook = Rainy, Temp = Mild, Humidity = Normal, Windy = True, YES or No?

Likelihood of yes = **0.0141**
Likelihood of No = **0.0103**

After Normalizaion,
    Yes = 0.0141 / (0.0141 + 0.0103) = **0.58**
    No = 0.0103 / (0.0141 + 0.0103) = **0.42**

How Naive Bayes algorithm works

		Play Golf	
	⭐	Yes	No
Outlook	Sunny	3 3/9	2 2/5
	Overcast	4 4/9	0 0/5
	Rainy	2 2/9	3 3/5

		Play Golf	
		Yes	No
Temp.	Hot	2 2/9	2 2/5
	Mild	4 4/9	2 2/5
	Cool	3 3/9	1 1/5

		Play Golf	
		Yes	No
Humidity	High	3 3/9	4 4/9
	Normal	6 6/9	1 1/5

		Play Golf	
		Yes	No
Windy	False	6 6/9	2 2/5
	True	3 3/9	3 3/5

Problem2: For Outlook = Rainy, Temp = Mild, Humidity = Normal, Windy = True, YES or No?

YES

Zero Frequency Problem

What if any of the count is 0?

- ?

Zero Frequency Problem

What if any of the count is 0?
- Add 1 to all counts
- It is a form of Laplace smoothing
- See https://en.wikipedia.org/wiki/Additive_smoothing

Using Python

```
#Import Library of Gaussian Naive Bayes model
from sklearn.naive_bayes import GaussianNB
import numpy as np

#assigning predictor and target variables
X= np.array([[-3,7],[1,5], [1,2], [-2,0], [2,3], [-4,0], [-1,1], [1,1], [-2,2], [2,7], [-4,1],
[-2,7]])
y = np.array([3, 3, 3, 3, 4, 3, 3, 4, 3, 4, 4, 4])
#Create a Gaussian Classifier
model = GaussianNB()

# Train the model using the training sets
model.fit(X, y)

#Predict Output
predicted= model.predict([[1,2],[3,4]])
print(predicted)

See the Jupyter Notebook.
```

Tips to improve the Naive Bayes Model

- If continuous features do not have normal distribution,
  - we should use transformation or different methods to convert
- If test data set has zero frequency issue,
  - apply smoothing techniques "Laplace smoothing"
- Remove correlated features,
  - as the highly correlated features are voted twice in the model
  - and it can lead to over inflating importance.
- Naive Bayes classifier has limited options for parameter tuning
- Can't be ensembled - because there is no variance to reduce

Variants

Gaussian:
- It is used in classification and it assumes that features follow a normal distribution.

Multinomial:
It is used for discrete counts.
Implements the naive Bayes algorithm for multinomially distributed data
It is one of the two classic naive Bayes variants used in text classification

Bernoulli:
The binomial model is useful if your feature vectors are binary (i.e. zeros and ones).
One application would be text classification with 'bag of words' model
    where the 1s & 0s are "word occurs in the document"
    and "word does not occur in the document" respectively.

http://scikit-learn.org/stable/modules/naive_bayes.html

Applications

1. **Real time Prediction:**
   - Naive Bayes is an eager learning classifier and it is sure fast. Thus, it could be used for making predictions in real time.
2. **Multi class Prediction:**
   - Well known for multi class prediction feature.

Applications

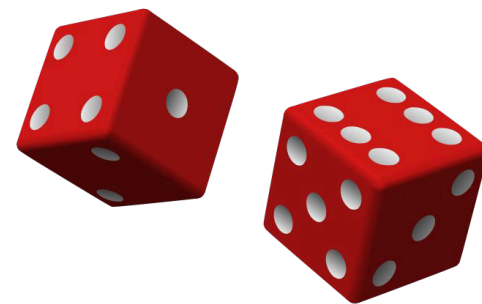1. **Text classification/ Spam Filtering/ Sentiment Analysis:**
   - Mostly used in text classification
   - Have higher success rate as compared to other algorithms.
   - Widely used in Spam filtering (identify spam e-mail)
   - and Sentiment Analysis
2. **Recommendation System:**
   - Naive Bayes Classifier and Collaborative Filtering together builds a Recommendation System that uses machine learning and data mining techniques to filter unseen information and predict whether a user would like a given resource or not

Further references

- http://scikit-learn.org/stable/modules/naive_bayes.html
- https://en.wikipedia.org/wiki/Bayes%27_theorem
- https://en.wikipedia.org/wiki/Naive_Bayes_classifier

Naive Bayes