DEFENCE INSTITUTE OF ADVANCED TECHNOLOGY

DEEMED UNIVERSITY

शास्त्रेण रक्षाम्

शस्त्रं प्रकरोति

Machine Learning for Cyber Security (CS-602)

L#10

Naïve Bayes Classifier

By

Dr Sunita Dhavale

Syllabus

- Data Analytics Foundations: R programming, Python Basics -Expressions and Variables, String Operations, Lists and Tuples, Sets, Dictionaries Conditions and Branching, Loops, Functions, Objects and Classes, Reading/Writing files, Hand ling data with Pandas, Scikit Library, Numpy Library, Matplotlib, scikit programming for data analysis, setting up lab environment, study of standard datasets. Introduction to Machine Learning- Applications of Machine Learning, Supervised, unsupervised classification and regression analysis
- Python libraries suitable for Machine Learning Feature Extraction. Data pre-processing, feature analysis etc., Dimensionality Reduction & Feature Selection Methods, Linear Discriminant Analysis and Principal Component Analysis, tackle data class imbalance problem

Syllabus

- Supervised and regression analysis, Regression, Linear Regression, Non-linear Regression, Model evaluation methods, Classification, K-Nearest Neighbor, Naïve Bayes, Decision Trees, Logistic Regression, Support Vector Machines, Artificial Neural Networks, Model Evaluation. Ensemble Learning, Convolutional Neural Networks, Spectral Embedding, Manifold detection and Anomaly Detection
- Unsupervised classification K-Means Clustering, Hierarchical Clustering, Density-Based Clustering, Recommender Systems-Content-based recommender systems, Collaborative Filtering, machine learning techniques for standard dataset, ML applications, Case studies on Cyber Security problems that can be solved using Machine learning like Malware Analysis, Intrusion Detection, Spam detection, Phishing detection, Financial Fraud detection, Denial of Service Detection.

Text/Reference Books

1. Building Machine Learning Systems with Python – Willi Richert, Luis Pedro Coelho

2. Alessandro Parisi, Hands-On Artificial Intelligence for Cybersecurity: Implement smart AI systems for preventing cyber attacks and detecting threats and network anomalies Publication date :Aug 2, 2019, Packt, ISBN-13, 9781789804027

3. Machine Learning: An Algorithmic Perspective – Stephen Marsland

4. Sunita Vikrant Dhavale, "Advanced Image-based Spam Detection and Filtering Techniques", IGI Global, 2017

5. Soma Halder , Sinan Ozdemir, Hands-On Machine Learning for Cybersecurity: Safeguard your system by making your machines intelligent using the Python ecosystem, By Publication date : Dec 31, 2018, Packt, ISBN-13 :9781788992282

1. Stuart Russell, Peter Norvig (2009), "Artificial Intelligence – A Modern Approach", Pearson Elaine Rich & Kevin Knight (1999), "Artificial Intelligence", TMH, 2nd Edition

2. NP Padhy (2010), "Artificial Intelligence & Intelligent System", Oxford

3. ZM Zurada (1992), "Introduction to Artificial Neural Systems", West Publishing Company

4. Research paper for study (if any) – White papers on multimedia from IEEE/ACM/Elsevier/Spinger/ Nvidia sources.

Lab assignments

1	**Python Programming part-1**
2	Python Programming part-2
3	Study and Implement Linear Regression Algorithm for any standard dataset like in cyber security domain
4	Study and Implement KMeans Algorithm for any standard dataset in cyber security domain
5	Study and Implement KNN for any standard dataset in cyber security domain
6	Study and Implement ANN for any standard dataset in cyber security domain
7	Study and Implement PCA for any standard dataset in cyber security domain
8	Case Study: Use of ML along with Fuzzy Logic/GA to solve real world Problem in cyber security domain
9	Mini assignment: Apply ML along with PSO/ACO to solve any real world problem in cyber security domain
10	ML Practice Test – 1 Quiz

Defence Institute of Advanced Technology

School of Computer Engineering & Mathematical Sciences

SEMESTER-I TIME TABLE (AUTUMN 2024)$

PROGRAMMES: (I) CS [M.TECH IN CYBER SECURITY] (II) AI [M.TECH CSE (ARTIFICIAL INTELLIGENCE)] BATCH: 2024-2026

Lecture / Day	L1 0900-1000	L2 1000-1100	L3 1100-1200	L4 1200-1300		L4 1400-1500	L4 1500-1600	L4 1600-1700	L4 1700-1800
Monday	CE-602 (AI) CS-602 (CS)	CE-604 (AI) CS-603 (CS)	CE-601 (AI) CS-604 (CS)	CE-601 (AI) LAB CS-603 (CS)	Lunch Break 1300-1400	LAB CE-601 (AI) LAB CS-602 (CS)		AM607	
Tuesday	CE-603 (AI) LAB CS-603 (CS)	CE-602 (AI) CS-602 (CS)	CE-601 (AI) CS-605 (CS)	CE-604 (AI) CS-604 (CS)		PGC 601		AM607	
Wednesday	CS-605 (CS)	CE-603 (AI) CS-602 (CS)	CE-602 (AI) CS-603 (CS)	CE-604 (AI) CS-604 (CS)		CE-605(AI) LAB CS-605 (CS)	LAB CS-605 (CS)	AM607	
Thursday	LAB CE-604 (AI) CS-603 (CS)	LAB CE-604 (AI) CS-605 (CS)	LAB CE-602 (AI) CS-601 (CS)	CE-603 (AI) CS-601 (CS)		PGC 601		AM607	
Friday	LAB CE-603 (AI) LAB CS-601 (CS)		LAB CE-602 (AI) CS-601 (CS)	LAB CS-604 (CS)		CE-605(AI) LAB CS-604 (CS)	CE-605(AI)	LAB CE-605(AI)	

COURSE CODE & COURSE NAME		FACULTY
Programme: CS [M.Tech in Cyber Security] Classroom: Arjun	Programme: AI [M.Tech CSE (Artificial Intelligence)] Classroom: Kaveri	
CS-601 Data Security & Privacy	CE-601 Responsible Artificial Intelligence;	MJN: Dr. Manisha J. Nene
CS-602 ML for Cyber Security	CE-604 Practical Machine Learning;	SVD: Dr. Sunita V. Dhavale
CS-605 Network and Cloud Security	CE-602 Intelligent Algorithms	CRS: Prof. CRS Kumar
CS-604 Advanced System Security	---------	DVV: Dr. Deepti V. Vidyarthi
CS-603 Applied Cryptography	---------	AM: Dr. Arun Mishra
---------	CE-603 Deep Neural Network;	US: Dr. Upasna Singh
---------	CE-605 Mathematics for ML;	Unit-2: Dr Upasna, Unit 4: Dr Sunita, Unit3:MJN, Unit 1: Faculty To be Nominated
AM-607 Mathematics for Engineers	AM-607 Mathematics for Engineers	OO/DS/DP: Dr Odelly O., Dr Dasari S., Dr. Debasis P.
PGC-601 Research Methodology	PGC-601 Research Methodology	Common Subject for All

$ TENTATIVE T.T. SUBJECT TO CHANGE

Program Coordinator,
M.Tech (CS & AI), Batch 2024-26

Director, SoCE&MS

Syllabus

- Data Analytics Foundations: R programming, Python Basics -Expressions and Variables, String Operations, Lists and Tuples, Sets, Dictionaries Conditions and Branching, Loops, Functions, Objects and Classes, Reading/Writing files, Hand ling data with Pandas, Scikit Library, Numpy Library, Matplotlib, scikit programming for data analysis, setting up lab environment, study of standard datasets. Introduction to Machine Learning- Applications of Machine Learning, Supervised, unsupervised classification and regression analysis
- Python libraries suitable for Machine Learning Feature Extraction. Data pre-processing, feature analysis etc., Dimensionality Reduction & Feature Selection Methods, Linear Discriminant Analysis and Principal Component Analysis, tackle data class imbalance problem

Syllabus

- Supervised and regression analysis, Regression, Linear Regression, Non-linear Regression, Model evaluation methods, Classification, K-Nearest Neighbor, Naïve Bayes, Decision Trees, Logistic Regression, Support Vector Machines, Artificial Neural Networks, Model Evaluation. Ensemble Learning, Convolutional Neural Networks, Spectral Embedding, Manifold detection and Anomaly Detection
- Unsupervised classification K-Means Clustering, Hierarchical Clustering, Density-Based Clustering, Recommender Systems- Content-based recommender systems, Collaborative Filtering, machine learning techniques for standard dataset, ML applications, Case Study: Image spam detection.

Text/Reference Books

1. Building Machine Learning Systems with Python – Willi Richert, Luis Pedro Coelho
2. Learning scikit-learn: Machine Learning in Python – Raúl Garreta, Guillermo Moncecchi
3. Machine Learning: An Algorithmic Perspective – Stephen Marsland
4. Sunita Vikrant Dhavale, "Advanced Image-based Spam Detection and Filtering Techniques", IGI Global, 2017
5. Trevor Hastie, Robert Tibshirani, Jerome Friedman – The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Second Edition. February 2009

1. Stuart Russell, Peter Norvig (2009), "Artificial Intelligence – A Modern Approach", Pearson Elaine Rich & Kevin Knight (1999), "Artificial Intelligence", TMH, 2nd Edition
2. NP Padhy (2010), "Artificial Intelligence & Intelligent System", Oxford
3. ZM Zurada (1992), "Introduction to Artificial Neural Systems", West Publishing Company
4. Research paper for study (if any) – White papers on multimedia from IEEE/ACM/Elsevier/Spinger/ Nvidia sources.

Lab assignments

1	**Python Programming part-1**
2	Python Programming part-2
3	Study and Implement Linear Regression Algorithm for any standard dataset like Iris
4	Study and Implement KMeans Algorithm for any standard dataset like Iris
5	Study and Implement KNN for any standard dataset like Iris
6	Study and Implement ANN for any standard dataset like Iris
7	Study and Implement PCA for any standard dataset like Iris
8	Case Study: Use of ML along with Fuzzy Logic/ GA/PSO/ACO to solve real world Problem
9	Mini assignment: Apply ML to solve any real world problem
10	ML Practice Test – 1 Quiz

Defence Institute of Advanced Technology

School of Computer Engineering & Mathematical Sciences

SEMESTER-I TIME TABLE (AUTUMN 2024)$

PROGRAMMES: (I) CS [M.TECH IN CYBER SECURITY] (II) AI [M.TECH CSE (ARTIFICIAL INTELLIGENCE)] BATCH: 2024-2026

Lecture / Day	L1 0900-1000	L2 1000-1100	L3 1100-1200	L4 1200-1300		L4 1400-1500	L4 1500-1600	L4 1600-1700	L4 1700-1800
Monday	CE-602 (AI) CS-602 (CS)	CE-604 (AI) CS-603 (CS)	CE-601 (AI) CS-604 (CS)	CE-601 (AI) LAB CS-603 (CS)	Lunch Break 1300-1400	LAB CE-601 (AI) LAB CS-602 (CS)		AM607	
Tuesday	CE-603 (AI) LAB CS-603 (CS)	CE-602 (AI) CS-602 (CS)	CE-601 (AI) CS-605 (CS)	CE-604 (AI) CS-604 (CS)		PGC 601		AM607	
Wednesday	CS-605 (CS)	CE-603 (AI) CS-602 (CS)	CE-602 (AI) CS-603 (CS)	CE-604 (AI) CS-604 (CS)		CE-605(AI) LAB CS-605 (CS)	LAB CS-605 (CS)	AM607	
Thursday	LAB CE-604 (AI) CS-603 (CS)	LAB CE-604 (AI) CS-605 (CS)	LAB CE-602 (AI) CS-601 (CS)	CE-603 (AI) CS-601 (CS)		PGC 601		AM607	
Friday	LAB CE-603 (AI) LAB CS-601 (CS)		LAB CE-602 (AI) CS-601 (CS)	LAB CS-604 (CS)		CE-605(AI) LAB CS-604 (CS)	CE-605(AI)	LAB CE-605(AI)	

COURSE CODE & COURSE NAME		FACULTY
Programme: CS [M.Tech in Cyber Security] Classroom: Arjun	Programme: AI [M.Tech CSE (Artificial Intelligence)] Classroom: Kaveri	
CS-601 Data Security & Privacy	CE-601 Responsible Artificial Intelligence;	MJN: Dr. Manisha J. Nene
CS-602 ML for Cyber Security	CE-604 Practical Machine Learning;	SVD: Dr. Sunita V. Dhavale
CS-605 Network and Cloud Security	CE-602 Intelligent Algorithms	CRS: Prof. CRS Kumar
CS-604 Advanced System Security	---------	DVV: Dr. Deepti V. Vidyarthi
CS-603 Applied Cryptography	---------	AM: Dr. Arun Mishra
---------	CE-603 Deep Neural Network;	US: Dr. Upasna Singh
---------	CE-605 Mathematics for ML;	Unit-2: Dr Upasna, Unit 4: Dr Sunita, Unit3:MJN, Unit 1: Faculty To be Nominated
AM-607 Mathematics for Engineers	AM-607 Mathematics for Engineers	OO/DS/DP: Dr Odellu O., Dr Dasari S., Dr. Debasis P.
PGC-601 Research Methodology	PGC-601 Research Methodology	Common Subject for All

$ TENTATIVE T.T. SUBJECT TO CHANGE

Program Coordinator,
M.Tech (CS & AI), Batch 2024-26

Director, SoCE&MS

Naïve Bayes Classifier

Probabilities

- Have two dice h_1 and h_2
- The probability of rolling an *i* given die h_1 is denoted $P(i|h_1)$. This is a *conditional probability*
- Pick a die at random with probability $P(h_j)$, j=1 or 2. The probability for picking die h_j *and* rolling an i with it is called *joint probability* and is $P(i, h_j)=P(h_j)P(i| h_j)$.
- For any events X and Y, $P(X,Y)=P(X|Y)P(Y)$
- If we know $P(X,Y)$, then the so-called *marginal probability* $P(X)$ can be computed as

$$P(X) = \sum_Y P(X,Y)$$

- Probabilities sum to 1.

Given two events A and B, if $P(B) > 0$, then the conditional probability that A occurs given that B occurs is defined to be

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Essentially, since event B has occurred, it becomes the new sample space.

Example

Q. A fair coin is tossed twice. What is the probability that both tosses result in heads given that at least one of the tosses resulted in a heads?

Example

Q. A fair coin is tossed twice. What is the probability that both tosses result in heads given that at least one of the tosses resulted in a heads?

Sol. $\Omega = \{HH, TT, HT, TH\}$
$\mathcal{P}(HH) = \mathcal{P}(TT) = \mathcal{P}(HT) = \mathcal{P}(TH) = 1/4$

$$\begin{aligned}
&\mathcal{P}(HH|\text{at least one toss heads}) \\
&\quad = \mathcal{P}(HH|HT \cup TH \cup HH) \\
&\quad = \frac{\mathcal{P}(HH \cap (HT \cup TH \cup HH))}{\mathcal{P}(HT \cup TH \cup HH)} \\
&\quad = \frac{\mathcal{P}(HH)}{\mathcal{P}(HT \cup TH \cup HH)} \\
&\quad = \frac{1}{3}
\end{aligned}$$

Bayes Rule

$$\mathcal{P}(A|B) = \frac{\mathcal{P}(A \cap B)}{\mathcal{P}(B)}$$

$$\mathcal{P}(A \cap B) = \mathcal{P}(A|B)\mathcal{P}(B)$$

$$\mathcal{P}(A \cap B) = \mathcal{P}(B|A)\mathcal{P}(A)$$

$$\mathcal{P}(A|B)\mathcal{P}(B) = \mathcal{P}(B|A)\mathcal{P}(A)$$

$$\mathcal{P}(A|B) = \frac{\mathcal{P}(B|A)\mathcal{P}(A)}{\mathcal{P}(B)} \text{ (Bayes' Rule)}$$

Bayes Theorem

- P(x) is the normalizing factor/evidence. So, it has no role in classification. So, only we have to consider the likelihood and the prior.
- If x is Feature vector, determine the class.
- Objective of Bayes Decision Making - probability of obtaining a particular class given the Feature vector

$$P(\omega_j \mid x) = \frac{p(x \mid \omega_j) P(\omega_j)}{P(x)}$$

$$posterior = \frac{likelihood \times prior}{evidence}$$

In the case of two categories

$$P(x) = \sum_{j=1}^{j=2} P(x \mid \omega_j) P(\omega_j)$$

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B|A)P(A)}{P(B)}$$

Does patient have cancer or not?

- A patient takes a lab test and the result comes back positive. It is known that the test returns a correct positive result in only 98% of the cases and a correct negative result in only 97% of the cases. Furthermore, only 0.008 of the entire population has this disease.

1. What is the probability that this patient has cancer?
2. What is the probability that he does not have cancer?
3. What is the diagnosis?

$$\left.\begin{array}{r} hypothesis\,1: 'cancer' \\ hypothesis\,2: '\neg cancer' \end{array}\right\} \quad hypothesis \;\; space \;\; H$$

$$- data : '+'$$

$$1. P(cancer \mid +) = \frac{P(+ \mid cancer)P(cancer)}{P(+)} = \frac{\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots}{\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots} = \ldots\ldots\ldots$$

$$P(+ \mid cancer) = 0.98$$

$$P(cancer) = 0.008$$

$$P(+) = P(+ \mid cancer)P(cancer) + P(+ \mid \neg cancer)P(\neg cancer)$$

$$= \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$P(+ \mid \neg cancer) = 0.03$$

$$P(\neg cancer) = \ldots\ldots\ldots\ldots$$

$$2. P(\neg cancer \mid +) = \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$3. Diagnosis\ ??$$

Solution

$P(cancer) = .008$ $P(\neg\ cancer) = .992$

$P(+ve|cancer) = .98$ $P(-ve|cancer) = .02$

$P(+ve|\neg\ cancer) = .03$ $P(-ve|\neg\ cancer) = .97$

Using Bayes Formula:

$P(cancer|+ve) = P(+ve|cancer) x P(cancer) / P(+ve)$

$= 0.98 \times 0.008 / P(+ve) = .00784 / P(+ve)$

$P(\neg\ cancer|+ve) = P(+ve|\neg\ cancer) x P(\neg\ cancer) / P(+ve)$

$= 0.03 \times 0.992 / P(+ve) = .0298 / P(+ve)$

So, the patient most likely does not have cancer.

Naïve Bayes algorithm

- It uses Bayes theorem for classification problems.
- It is highly used in text classification.
- In text classification tasks, data contains high dimension (as each word represent one feature in the data).
- It is used in spam filtering, sentiment detection, rating classification etc.
- This model predicts the probability of an instance belongs to a class with a given set of feature value.
- It is a probabilistic classifier.
- It is called naïve because it assumes that one feature in the model is independent of existence of another feature. i.e. each feature contributes to the predictions with no relation between each other.
- In real world, this condition satisfies rarely.

Classification with the Naive-Bayes principle

The example to be classified is described by $\mathbf{x} = (x_1, \ldots, x_n)$.

1. For each x_i, and for each class c_j, calculate the conditional probability, $P(x_i|c_j)$, as the relative frequency of x_i among those training examples that belong to c_j.
2. For each class, c_j, carry out the following two steps:

   i) estimate $P(c_j)$ as the relative frequency of this class in the training set;

   ii) calculate the conditional probability, $P(\mathbf{x}|c_j)$, using the "naive" assumption of mutually independent attributes:

$$P(\mathbf{x}|c_j) = \prod_{i=1}^{n} P(x_i|c_j)$$

3. Choose the class with the highest value of $P(c_j) \cdot \prod_{i=1}^{n} P(x_i|c_j)$.

To Handle Continuous Attributes

- attributes (such as age, price or weight) acquire values from continuous domains.
- discretize.
- e.g. age with the boolean attribute old that is true for age > 60 and false otherwise.
- part of the available information gets lost.
- The loss will be mitigated if we divide the original domain into not two, but several intervals, say, .0; 10; : : : .90; 100.
- dividing the original domain into shorter—and thus more numerous—intervals, provided that the number of values in each bin is sufficient

Example. ‘Play Tennis’ data

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
Day1	Sunny	Hot	High	Weak	*No*
Day2	Sunny	Hot	High	Strong	*No*
Day3	Overcast	Hot	High	Weak	Yes
Day4	Rain	Mild	High	Weak	Yes
Day5	Rain	Cool	Normal	Weak	Yes
Day6	Rain	Cool	Normal	Strong	*No*
Day7	Overcast	Cool	Normal	Strong	Yes
Day8	Sunny	Mild	High	Weak	*No*
Day9	Sunny	Cool	Normal	Weak	Yes
Day10	Rain	Mild	Normal	Weak	Yes
Day11	Sunny	Mild	Normal	Strong	Yes
Day12	Overcast	Mild	High	Strong	Yes
Day13	Overcast	Hot	Normal	Weak	Yes
Day14	Rain	Mild	High	Strong	*No*

Based on the examples in the table, classify the following datum ***x***:

x=(Outlook=Sunny, Temp=Cool, Hum=High, Wind=strong)

- That means: Play tennis or not?

$$h_{NB} = \underset{h\in[yes,no]}{\arg\max}\, P(h)P(\mathbf{x}\,|\,h) = \underset{h\in[yes,no]}{\arg\max}\, P(h)\prod_t P(a_t\,|\,h)$$

$$= \underset{h\in[yes,no]}{\arg\max}\, P(h)P(Outlook = sunny\,|\,h)P(Temp = cool\,|\,h)P(Humidity = high\,|\,h)P(Wind = strong\,|\,h)$$

Example

Learning Phase

Outlook	Play=*Yes*	Play=*No*
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Temperature	Play=*Yes*	Play=*No*
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

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

Wind	Play=*Yes*	Play=*No*
Strong	3/9	3/5
Weak	6/9	2/5

$P(\text{Play}=Yes) = 9/14$ $\quad$ $P(\text{Play}=No) = 5/14$

Based on the examples in the table, classify the following datum ***x***:

x=(Outlook=Sunny, Temp=Cool, Hum=High, Wind=strong)

- That means: Play tennis or not?

$$h_{NB} = \underset{h \in [yes, no]}{\arg\max}\, P(h)P(\mathbf{x} \mid h) = \underset{h \in [yes, no]}{\arg\max}\, P(h)\prod_t P(a_t \mid h)$$

$$= \underset{h \in [yes, no]}{\arg\max}\, P(h)P(Outlook = sunny \mid h)P(Temp = cool \mid h)P(Humidity = high \mid h)P(Wind = strong \mid h)$$

Based on the examples in the table, classify the following datum ***x***:

x=(Outl=Sunny, Temp=Cool, Hum=High, Wind=strong)

- That means: Play tennis or not?

$$h_{NB} = \arg\max_{h\in[yes,no]} P(h)P(\mathbf{x} \mid h) = \arg\max_{h\in[yes,no]} P(h)\prod_t P(a_t \mid h)$$

$$= \arg\max_{h\in[yes,no]} P(h)P(Outlook = sunny \mid h)P(Temp = cool \mid h)P(Humidity = high \mid h)P(Wind = strong \mid h)$$

- Working:

$$P(PlayTennis = yes) = 9/14 = 0.64$$

$$P(PlayTennis = no) = 5/14 = 0.36$$

$$P(Wind = strong \mid PlayTennis = yes) = 3/9 = 0.33$$

$$P(Wind = strong \mid PlayTennis = no) = 3/5 = 0.60$$

$$etc.$$

$$P(yes)P(sunny \mid yes)P(cool \mid yes)P(high \mid yes)P(strong \mid yes) = 0.0053$$

$$P(no)P(sunny \mid no)P(cool \mid no)P(high \mid no)P(strong \mid no) = \mathbf{0.0206}$$

$$\Rightarrow answer: PlayTennis(x) = no$$

RCode

```
# install the package if you don’t already have it
install.packages("e1071")
#load package
library(e1071)
# use the naïveBayes classifier on the iris data
m <- naiveBayes(iris[,1:4], iris[,5])
table(predict(m, iris[,1:4]), iris[,5])
```

Python Code

- from sklearn.datasets import load_iris
- from sklearn.model_selection import train_test_split
- from sklearn.naive_bayes import GaussianNB
- X, y = load_iris(return_X_y=True)
- X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.5, random_state=0)
- gnb = GaussianNB()
- y_pred = gnb.fit(X_train, y_train).predict(X_test)
- print("Number of mislabeled points out of a total %d points : %d" % (X_test.shape[0], (y_test != y_pred).sum()))

Python Code

- naive_bayes = GaussianNB()
- naive_bayes.fit(X_train,Y_train)
- #Check on test data:
- Y_pred = naive_bayes.predict(X_test)
- Y_pred_prob = naive_bayes.predict_proba(X_test)
- print(accuracy_score(Y_test,Y_pred))
- #Learning Curve:
- scikitplot.estimators.plot_learning_curve(naive_bayes, X_train,Y_train)
- # Confusion matrix:
- scikitplot.metrics.plot_confusion_matrix(Y_test, Y_pred, normalize=True)
- # ROC Curve:
- Y_pred_prob = naive_bayes.predict_proba(X_test)
- class_1_prob = list()
- for i in Y_pred_prob:
-     class_1_prob.append(i[1])
- print(roc_auc_score(Y_test,class_1_prob))
- model_result['Naive Bayes'] = roc_auc_score(Y_test, class_1_prob)
- scikitplot.metrics.plot_roc_curve(Y_test, Y_pred_prob, curves=['each_class'])

Conclusions

- Naïve Bayes based on the independence assumption
  - Training is very easy and fast; just requiring considering each attribute in each class separately
  - Test is straightforward; just looking up tables or calculating conditional probabilities with normal distributions
  - Many successful applications, e.g., spam mail filtering
  - A good candidate of a base learner in ensemble learning

Thank You

- **Any Questions???**

DEFENCE INSTITUTE OF ADVANCED TECHNOLOGY

DEEMED UNIVERSITY

शास्त्रेण शस्त्रं प्रकरोति रक्षाम्