Session #24,25

CMSC 409: Artificial Intelligence

http://www.people.vcu.edu/~mmanic/

Virginia Commonwealth University, Fall 2023, Dr. Milos Manic (mmanic@vcu. edu)

1

CMSC 409: Artificial Intelligence

Session # 24,25

Topics for today

- Announcements
- Lessons learned on project 04
- Probabilistic classifier
  - *Univariate and multivariate classification, examples*
  - *General Bayes classifier, examples*
- Learning from Neighbors
  - Eager vs. lazy learners
  - Lazy learners
    - *K-nearest-neighbor classifier*
    - *Case based reasoning (CBR)*
    - *Distance measures*
      - *Euclidian space, coding theory, fuzzy space*
      - *Euclidian, Manhattan, Chebyshev, angle distance*

2

CMSC 409: Artificial Intelligence

Announcements Session # 24,25

- Canvas
  - *Prev. session slides updated*
- TAs
  - *Victor Cobilean <cobileanv@vcu.edu>, Harindra Sandun Mavikumbure mavikumbureh@vcu.edu; TA office hours: Thursdays, 3:30 - 4:30pm (Zoom)*
- Project #4
  - *graded, review week after deadline (deadline was Nov. 10)*
- **Final exam**
  - **Dec. 12 (take home, 48hr open book exam)**
  - Prep examples posted
- Paper (optional)
  - *The 4th draft (final submission) due Nov. 28*
    - *In addition to previous draft, it should contain a technique (or selection thereof), you plan on using to solve the selected problem (check out the class paper instructions for the 4th draft)*
- Interest in PhD or MSc program?
- Subject line and signature
  - *Please use [CMSC 409] Last_Name Question*

3

Lessons learned

Project 04

4

Class Statistics

STATISTICS

COUNT	24
Minimum Value	0
Maximum Value	15
Range	15
Average	13
Median	15
Standard Deviation	4.33
Variance	18.78

GRADE DISTRIBUTION

Greater than 100	0
90 - 100	18
80 - 89	1
70 - 79	2
60 - 69	1
50 - 59	0
40 - 49	0
30 - 39	0
20 - 29	0
10 - 19	0
0 - 9	2
Not submitted	0

Project 4 review

- **Pr. 4.1**

Pr.4.

1. Create the feature vector by writing a program that applies the following text mining techniques to this set of paragraphs. (4 pts)

Download and unzip "Project4_code.zip" files. A set of paragraphs is given in the file "Project4_paragraphs*.txt*". Proceed with the following steps.

A. Tokenize paragraphs
B. Remove punctuation and special characters (including html tags)
C. Remove numbers
D. Convert upper-case to lower-case
E. Remove stop words. A set of stop words is provided in the file "Project4_*stop_words.txt*"
F. Perform stemming. Use the Porter stemming code provided in the file "*Porter_Stemmer_X.txt*"
G. Combine stemmed words (in case Porter stemming did not catch some words and you would need to stem them by yourself).
H. Extract the most frequent words (i.e. words for the feature vector, or most characteristic, distinct words).

I. Provide the feature vector in your report.

Project 4 review

- **Feature vector**
  - The final feature vector should list most frequent (characteristic, or distinct) words.
    - for ex., it should not contain words that appear only once in a document.
  - it should only contain stemmed words (for example "movi", "comedi", "film", "pretti", etc.)

{'jimmi': 0, 'mario': 1, 'drink': 2, 'game': 3, 'book': 4, 'film': 5, 'woman': 6, 'movi': 7, 'murder': 8, 'pretti': 9, 'comedi': 10, 'on': 11, 'read': 12, 'star': 13, 'take': 14}

Project 4 review

- **Pr.4.1**

- **Threshold 3 ():**

- **Threshold 7 ():**

Project 4 review

- **Pr.4.1**

  - **Threshold 10 ():**


```
{'jimmi': 0, 'mario': 1, 'azumi': 2, 'hamlet': 3, 'zombi': 4, 'drink': 5, 'game': 6, 'book': 7, 'film': 8, 'woman': 9, 'movi': 10, 'danc': 11, 'murder': 12, 'pretti': 13, 'comedi': 14, 'father': 15, 'shock': 16, 'on': 17, 'read': 18, 'young': 19, 'star': 20, 'take': 21, 'wa
i': 22, 'noth': 23, 'make': 24, 'mr': 25, 'old': 26, 'bad': 27, 'work': 28, 'show': 29, 'life': 30, 'year': 31, 'new': 32, 'music': 33, 'go': 34, 'simpli': 35, 'look': 36, 'charact': 37, 'plot': 38, 'even': 39, 'word': 40, 'stori': 41, 'realli': 42, 'dont': 43, 'two': 44, '
kill': 45, 'part': 46, 'peopl': 47, 'differ': 48, 'great': 49, 'out': 50, 'mani': 51, 'there': 52, 'main': 53, 'girl': 54, 'himself': 55, 'veri': 56, 'watch': 57, 'friend': 58, 'same': 59, 'audienc': 60, 'up': 61, 'complet': 62, 'such': 63, 'time': 64, 'love': 65, 'seem': 6
6, 'see': 67, 'scene': 68, 'actor': 69, 'doesnt': 70, 'line': 71, 'man': 72, 'gui': 73, 'enough': 74, 'sound': 75, 'end': 76, 'first': 77, 'be': 78, 'far': 79, 'over': 80, 'without': 81, 'well': 82, 'world': 83, 'im': 84, 'that': 85, 'think': 86, 'come': 87, 'through': 88,
'good': 89, 'littl': 90, 'made': 91, 'chang': 92, 'big': 93, 'walk': 94, 'quit': 95, 'leav': 96, 'enjoi': 97, 'more': 98, 'back': 99, 'final': 100, 'seen': 101, 'both': 102, 'much': 103, 'set': 104, 'classic': 105, 'although': 106, 'those': 107, 'us': 108, 'plai': 109, 'her
e': 110, 'still': 111, 'best': 112, 'feel': 113, 'done': 114, 'laugh': 115, 'now': 116, 'point': 117, 'give': 118, 'down': 119, 'perform': 120, 'actual': 121, 'act': 122, 'turn': 123, 'real': 124, 'try': 125, 'thing': 126, 'know': 127, 'find': 128, 'begin': 129, 'whole': 13
0, 'name': 131, 'few': 132, 'wasnt': 133, 'idea': 134, 'director': 135, 'someth': 136, 'sens': 137, 'cast': 138, 'sure': 139, 'fan': 140, 'screen': 141, 'keep': 142, 'awai': 143, 'anoth': 144, 'place': 145, 's': 146, 'iv': 147, 'help': 148, 'role': 149, 'mention': 150, 'rig
ht': 151, 'never': 152, 'probabl': 153, 'want': 154, 'cant': 155, 'befor': 156, 'worth': 157, 'attempt': 158, 'tell': 159, 'meet': 160, 'need': 161, 'effect': 162, 'start': 163, 'saw': 164, 'becom': 165, 'interest': 166, 'get': 167, 'develop': 168, 'better': 169, 'happen':
170, 'rest': 171, 'last': 172}
```


  - **Threshold 14 ():**


```
{'jimmi': 0, 'mario': 1, 'drink': 2, 'game': 3, 'book': 4, 'film': 5, 'woman': 6, 'movi': 7, 'murder': 8, 'pretti': 9, 'comedi': 10, 'on': 11, 'read': 12, 'star': 13, 'take': 14, 'wai': 15, 'noth': 16, 'make': 17, 'bad': 18, 'work': 19, 'show': 20, 'life': 21, 'year': 22, '
new': 23, 'music': 24, 'go': 25, 'simpli': 26, 'look': 27, 'charact': 28, 'plot': 29, 'even': 30, 'stori': 31, 'realli': 32, 'dont': 33, 'two': 34, 'kill': 35, 'part': 36, 'peopl': 37, 'great': 38, 'out': 39, 'mani': 40, 'there': 41, 'girl': 42, 'veri': 43, 'watch': 44, 'au
dienc': 45, 'up': 46, 'complet': 47, 'such': 48, 'time': 49, 'love': 50, 'seem': 51, 'see': 52, 'scene': 53, 'actor': 54, 'doesnt': 55, 'line': 56, 'man': 57, 'end': 58, 'first': 59, 'be': 60, 'far': 61, 'over': 62, 'without': 63, 'well': 64, 'world': 65, 'im': 66, 'that':
67, 'think': 68, 'come': 69, 'through': 70, 'good': 71, 'littl': 72, 'made': 73, 'big': 74, 'more': 75, 'back': 76, 'final': 77, 'seen': 78, 'both': 79, 'much': 80, 'set': 81, 'us': 82, 'plai': 83, 'here': 84, 'still': 85, 'best': 86, 'feel': 87, 'point': 88, 'give': 89, 'd
own': 90, 'perform': 91, 'actual': 92, 'act': 93, 'thing': 94, 'know': 95, 'find': 96, 'begin': 97, 'whole': 98, 'name': 99, 'director': 100, 'someth': 101, 'mention': 102, 'right': 103, 'never': 104, 'probabl': 105, 'want': 106, 'cant': 107, 'befor': 108}
```


Project 4 review

- **Pr. 4.2**

2. Using the feature vector generated in the first task, write a program that generates the Term Document Matrix (TDM) for ALL of the paragraphs in "Project4_paragraphs.*txt*", similar to TDM below. (5 pts)

Example TDM

Keyword set	review	watch	scene	**…**
Paragraph 1	1	4		…
Paragraph 2	2	0	1	…
…..	…	…	…	…
Paragraph 20	2	0	0	…

a) **Provide the TDM in your report.** (3 pts)

b) For each of the text mining steps (A to H), explain the purpose of each step and what sort of information is lost while applying each of those text-mining steps. (2 pts)

Project 4 review

Threshold = 14

Term Document Matrix (TDM)

- Use the Feature vector to generate the TDM
- Dimensionality (m x n) of the TDM is driven by number of paragraphs (*m*) in the document and the number of words (*n*) in the feature vector.
- Dimensionality of the TDM can be reduced by making the feature vector smaller (by applying a higher frequency threshold to obtain the most frequent words).

paragraph	jimmi	mario	drink	game	book	film	woman	movi	murder	pretti	comedi	on	read	star	take
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0
2	0	0	0	0	0	0	1	0	0	0	3	1	0	0	0
3	0	0	0	0	0	0	0	1	2	1	0	0	0	1	0
4	0	0	0	0	0	2	0	1	0	0	0	6	0	0	0
5	0	0	0	0	0	1	0	2	4	1	0	2	0	0	0
6	0	0	0	0	0	0	0	9	0	0	0	4	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
8	21	0	0	0	0	0	0	1	0	0	0	0	0	0	1
9	0	0	0	0	0	1	0	4	0	0	0	0	0	1	1
10	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0
11	0	0	0	1	0	2	0	5	0	2	0	1	0	0	1
12	0	0	0	0	0	1	0	6	0	0	0	4	0	2	0
13	0	0	0	0	0	1	0	1	0	0	0	3	0	0	0
14	0	0	0	0	4	6	0	4	0	4	0	1	0	0	0
15	0	18	0	12	0	0	1	0	0	1	0	0	0	6	2
16	0	0	0	0	0	2	0	0	0	0	0	0	0	4	0
17	0	0	0	0	0	12	2	1	0	0	0	2	0	0	0
18	0	0	0	0	0	7	0	3	0	0	0	1	2	1	0
19	0	0	2	0	4	4	0	2	0	1	0	0	3	0	0
20	0	0	0	0	1	7	2	0	0	0	2	1	2	0	1
21	0	0	0	0	0	3	0	5	0	0	0	0	0	0	0
22	0	0	0	0	0	1	0	5	0	0	2	0	0	0	0
23	0	0	0	0	0	2	0	0	1	1	0	0	0	0	0
24	0	0	0	0	0	9	0	3	0	0	0	1	0	0	0
25	0	0	0	0	0	0	0	1	0	2	0	0	0	0	2
26	0	0	0	0	0	2	0	1	0	0	0	1	0	0	0
27	0	0	0	0	0	7	0	0	0	0	0	0	0	1	0
28	0	0	0	0	0	0	0	4	0	0	0	1	0	0	0
29	0	0	0	0	0	4	0	0	0	0	0	2	0	0	0
30	0	0	0	0	0	1	0	1	0	0	0	2	1	0	0
31	0	0	0	0	0	0	0	0	10	0	0	1	0	0	0
32	0	0	0	0	0	1	0	3	0	0	0	0	0	0	2
33	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
34	0	0	0	0	0	0	0	2	0	1	0	2	0	0	1
35	0	0	0	0	0	10	0	11	0	0	0	4	1	0	2
36	0	0	0	0	0	5	0	0	0	2	1	1	0	0	0
37	0	0	0	0	0	2	1	2	0	0	0	0	0	1	0
38	0	0	0	0	0	2	0	3	0	0	0	0	0	0	2
39	0	0	0	0	9	3	0	1	0	0	0	3	4	0	0
40	0	0	0	0	0	6	0	0	0	0	0	0	3	2	0
41	0	0	0	0	0	4	0	0	0	0	2	2	0	0	0
42	0	0	0	0	0	1	0	2	0	1	0	0	0	0	0
43	0	0	0	0	0	3	0	1	0	0	0	0	0	0	2
44	0	0	0	1	0	16	0	8	0	0	6	1	0	1	1
45	0	0	0	0	2	3	1	5	2	0	3	2	1	3	4
46	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0
47	0	0	14	4	0	2	0	3	0	0	0	6	2	0	5
48	0	0	0	0	0	3	0	5	0	0	0	0	0	0	0
49	0	0	0	0	0	6	0	2	0	0	0	3	0	0	0
50	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0
51	0	0	0	0	0	6	17	1	0	16	0	3	0	0	0
52	0	0	0	0	0	2	0	3	0	0	0	0	0	0	0
53	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
55	0	0	0	0	0	3	0	1	1	0	0	0	0	0	0

Tokenizing
- Extracts the unique words from the text.
- We may lose information about the context and the relationship between the words

Removing punctuation and special characters
- They occur frequently and assumed to hold no context.

Remove numbers
- Not a good feature to use to summarize a text

Converting upper case letters to lowercase
- Makes all words uniform. easy to isolate unique words.

Stop words
- Removed because they are so common but provide little information to the text.

Stemming
- To reduce words down to their root. This allows for combining words that are essentially the same information.

Extracting the frequency of the stemmed words from each paragraph will give us a pattern for each paragraph that we can use to cluster. This will be used to construct the TDM.

The feature vector is the final list of unique stemmed words in the text that occur more frequently than a threshold that we can manipulate.

Project 4 review

- **Pr. 4.3**

3. Write a program implementing the clustering algorithm of your choice (Kohonen WTA or FCAN). Apply that algorithm to TDM to group similar paragraphs together. (6 pts)

a) How many clusters/topics have you identified? (2 pts)
b) What drives the dimensionality of TDM? What can you do to reduce that dimensionality? Does the order of data being fed to the algorithm matter? (2 pts)
c) Show and comment on the results. (2 pts)

 Session 24,25, Updated on 11/14/23 7:23:38 AM

13

Project 4 review

Pr.4.3

- **WTA/ FCAN**
  - Clustering should be applied to the TDM.
  - Once clustering completed, specify paragraphs for each cluster.
- **Discussion**
  - Results of the clustering should be discussed (why was a specific number of clusters obtained)
  - What were the choices you made for clustering (radius, learning rate)
  - Order of data fed to the algorithm may affect the result of clustering (affect the weights and the order in which the clusters are discovered).

 Session 24,25, Updated on 11/14/23 7:23:38 AM

14

Probabilistic Classifiers

- Probabilistic classifiers
  - Model the probabilistic distribution $P(C_i \mid x)$
  - the probability of belonging to class *Ci* given prior knowledge of *x*
  - make predictions based on probabilistic inference on this model.

- Bayesian decision theory: general framework for modeling $P(C_i \mid x)$ distribution using a Bayesian network

Probabilistic Classifiers

- Different probabilistic classifiers
  - Naïve Bayes:
    - application of Bayes theorem with **(naïve) assumption of independence** of features
    - we assume features independent from each other, samples are independent and identically distributed
  - Hidden Markov Models (HMM):
    - Instances in a sample are **not independent** and the **data is composed by sequences** generated by a parametric random process.
    - **Hidden refers to the state sequence** through which the model passes (not model parameters);
    - Viewed as simplest form of *dynamic* Bayesian network.
  - Dynamic Bayesian networks:
    - Generalization of hidden Markov models and Kalman filters
  - Kalman filters:
    - Linear quadratic estimation, applied in time series analysis

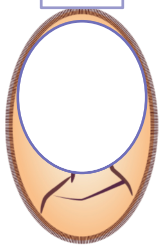
Simple example

(an example taking from link below)

or

Drew could be a name for a Male or a Female

Drew:



Is Drew a Male or a Female?
We can apply Bayes rule...

Training Dataset

Name	Sex
Drew	Male
Claudia	Female
Drew	Female
Drew	Female
Alberto	Male
Karin	Female
Nina	Female
Sergio	Male

$$P(male \mid drew) = \frac{P(drew \mid male)P(male)}{p(drew)}$$

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

From: http://www.cs.ucr.edu/~eamonn/CE/Bayesian%20Classification%20withInsect_examples.pdf

Session 24,25, Updated on 11/14/23 7:23:38 AM

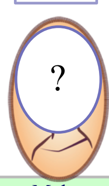
17

Simple example

$$P(Male \mid Drew) = \frac{P(Drew \mid Male)P(Male)}{p(Drew)}$$

Drew:

P person is Male, given person's name is Drew= 1/3 (likelihood) x 3/8 (prior) / marginal likelihood=
= three males, one of them is Drew x 3 Males out of 8 people / probability that is Drew (3 "Drews" in 8 people)
P(Female | Drew= 2/5 (2 Drews that are females) x…



Is Drew a Male or a Female?

$$P(Male \mid Drew) = \frac{(1/3)(3/8)}{3/8} = \frac{0.125}{3/8}$$

Drew is more likely to be a Female

$$P(Female \mid Drew) = \frac{(2/5)(5/8)}{3/8} = \frac{0.250}{3/8}$$

Training Dataset

Name	Sex
Drew	Male
Claudia	Female
Drew	Female
Drew	Female
Alberto	Male
Karin	Female
Nina	Female
Sergio	Male

$$P(Female \mid Drew) = \frac{P(Drew \mid Female)P(Female)}{p(Drew)}$$

Denominator the same (can be ignored)

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

Session 24,25, Updated on 11/14/23 7:23:38 AM

18

Simple example

Drew:

Drew is more likely to be a Female!

$$P(Male \mid Drew) = \frac{P(Drew \mid Male)P(Male)}{p(Drew)}$$

$$P(Female \mid Drew) = \frac{P(Drew \mid Female)P(Female)}{p(Drew)}$$

Training Dataset

Name	Sex
Drew	Male
Claudia	Female
Drew	Female
Drew	Female
Alberto	Male
Karin	Female
Nina	Female
Sergio	Male

$$P(Male \mid Drew) = \frac{(1/3)(3/8)}{3/8} = \frac{0.125}{3/8}$$

$$P(Female \mid Drew) = \frac{(2/5)(5/8)}{3/8} = \frac{0.250}{3/8}$$

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

19

Refresher…

The most common types of logarithms are **common logarithms, where the base is 10, binary logarithms, where the base is 2, and natural logarithms**, where the base is e ≈ 2.71828.

How do you calculate logarithms?

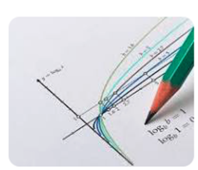
The natural logarithm and the common logarithm. How to calculate logarithm with an arbitrary base? Log base 2: an example.
...
Applying logarithms to arithmetic computations.

Rule or special case	Formula
Product	$\ln(x * y) = \ln(x) + \ln(y)$
Quotient	$\ln(x/y) = \ln(x) - \ln(y)$
Log of power	$\ln(x^y) = y * \ln(x)$
Log of e	$\ln(e) = 1$

What is logarithm used for?

A logarithm (or log) is the mathematical expression used **to answer the question: How many times must one "base" number be multiplied by itself to get some other particular number?** For instance, how many times must a base of 10 be multiplied by itself to get 1,000? The answer is 3 (1,000 = 10 × 10 × 10).
Aug 12, 2020

https://www.snexplores.org › article › explainer-what-are-...



What is…

log(100) =
log(10,000) =
log(10,000) =
log(0.0001)

20

Refresher…

What are logarithms again?

First, let's all just remind ourselves what a logarithm is (those who remember their logs well can skip to the next page!). The logarithm of a number is the exponent you raise above 10 to get that number. This is best seen by examples.

log(100) = 2 (why? because $10^2 = 100$)

log(10,000) = 4 (why? because $10^4 = 10{,}000$)

An easy rule that works for multiples of 10 is that the log is equal to the number of zeros trailing the one (go ahead and count the zeros!):

log (10,000,000) = 7

log (1,000,000,000,000) = 12

These multiples of 10 are always easy, but you can take the log of any number (in this case, we suggest you use your calculator- just type in the number, then hit the "log" button).

log(3,462) = 3.539327 (why? because $10^{3.539327} = 3{,}462$)

Logs can also be figured for numbers less than one. When a number is a fraction (less than one), then the log is always negative.

log(0.01) = -2 (why? because $10^{-2} = 0.01$)

Why does this work? Because 10^{-2} is the same as $1/10^2$, which equals 1/100, which equals 0.01!

log(0.0001) = -4 (why? because $10^{-4} = 1/10^4 = 1/10{,}000 = 0.0001$)

An easy rule that works for decimals that are multiples of 0.1 is that the log is equal to the number of zeros trailing the decimal plus the "1" (go ahead and count those zeros again!):

log (0.0000001) = -7

log (0.000000000001) = -12

Again, these multiples of 0.1 are always easy, but you can take the log of any positive decimal (but again, we suggest using your calculator!):

log(0.3462) = -0.4607 (why? because $10^{-0.4607} = 0.3462$)

https://mathbench.umd.edu/modules/misc_scaling/page08.htm

Multivariate Classification

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

- Probability that previously unseen sample x belongs to class Ci :

Marginal likelihood is:

$$P(e) = P(R=0,e) + P(R=1,e) + \cdots$$
$$= \sum_{i=0}^{r} P(e \mid R=i)\, P(R=i)$$

$$P(C_i \mid x) = \frac{P(x \mid C_i)P(C_i)}{P(x)} = \frac{P(x \mid C_i)P(C_i)}{\sum_{k=1}^{K} P(x \mid C_k)P(C_k)}$$

where $\sum_{k=1}^{K} P(x \mid C_k)P(C_k)$ is used for normalization (*again, marginal likelihood*).

• Multiplying (a lot of) probabilities can result in floating point underflow. Since:

$$\log(xy) = \log x + \log y$$

then for predicting $P(C_i \mid x)$, i.e. belonging of x to class Ci , we can focus on the numerator (log of it) :

$$g_i(x) = \log P(x \mid C_i) + \log P(C_i)$$

Note: the values of $g_i(x)$ are **negative**

*(**log** of a number **less** than **1** is negative: Since 0.01 = 10^{-2}, we must have **log** 0.01 = **-2**. ... **log 1** = 0 and **log** 10 = **1**, Cuny)*

Multivariate Classification

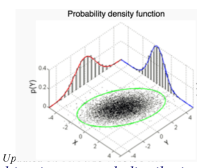
$$g_i(x) = \log P(x \mid C_i) + \log P(C_i)$$

- Maximum of $g_i(x)$ indicates (predicts) the class C_i of x sample (of i values, i.e. for all C_i classes) .
- One approach to find $P(x \mid C_i)$ is to assume that it is drawn from a Gaussian distribution, i.e:

$$P(x \mid C_i) \sim N_d(\mu_i, \Sigma_i)$$

$$P(x \mid C_i) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1} (x-\mu_i) \right]$$

- Where:
  - N_d *is the Multivariate Normal Distribution (MVN)*
  - d is the dimension of the data
  - Σ_i is the covariance matrix of the Gaussian distribution
  - μ_i is the mean of the Gaussian distribution



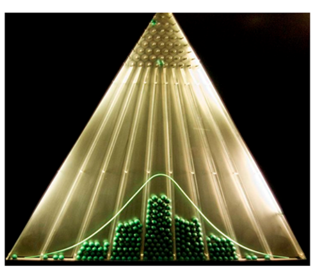
https://en.wikipedia.org/wiki/Multivariate_normal_distribution

23

Why is Gaussian the King of all distributions?

Significance of Gaussian distribution

Vidhi Chugh Aug 16, 2020 · 5 min read



Source: The bean machine is called the first generator of normal random variables

Where do we find the existence of Gaussian distribution?

ML practitioners or not, almost all of us have heard of this most popular form of distribution somewhere or the other. Everywhere we look around us, majority of the processes follow approximate Gaussian form, for e.g. age, height, IQ, memory, etc.

On a lighter note, there is one well-known example of Gaussian lurking around all of us i.e. 'bell curve' during appraisal time 😊

Yes, Gaussian distribution resonates with bell curve quite often and its probability density function is represented by the following mathematical formula:

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

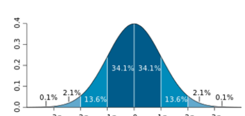
probability density function of Gaussian distribution

The mode is the number in a set of numbers that appears the most often. The mean of a set of numbers is the sum of all the numbers divided by the number of values in the set. The mean is also known as the average. May 19, 2022

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Gaussian Distribution and its key characteristics:

- Gaussian distribution is a continuous probability distribution with symmetrical sides around its center.
- Its mean, median and mode are equal.
- Its shape looks like below with most of the data points clustered around the mean with asymptotic tails.



Interpretation:

- ~68% of the values drawn from normal distribution lie within 1σ
- ~95% of the values drawn from normal distribution lie within 2σ
- ~99.7% of the values drawn from normal distribution lie within 3σ

https://towardsdatascience.com/why-is-gaussian-the-king-of-all-distributions-c45e0fe8a6e5

24

Multivariate Classification

- Having a training dataset $X^{(i)} \in R^{N_i x d}$ we can estimate the values of Σ_i, μ_i and $P(C_i)$ using the following equations:

$$P(C_i) = \frac{N_i}{N}$$

N_i is the number of samples that belong to class i
N is the number of all samples

$$\mu_i = \frac{\sum_{k=1}^{N_i} x_k^{(i)}}{N_i}, \textit{where } x_k^{(i)} \textit{ is the sample } k \textit{ that belongs to the class } Ci$$

$$\boxed{\Sigma_i = \frac{\sum_k (x_k^{(i)} - \mu_i)(x_k^{(i)} - \mu_i)^T}{N_i}} \rightarrow \boxed{\text{For multi dimensional data}}$$

$$\boxed{\Sigma_i = \sigma_i^2 = \frac{\sum_k (x_k^{(i)} - \mu_i)^2}{N_i}} \rightarrow \boxed{\text{For 1D data}}$$

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

Multivariate Classification

$$P(C_i \mid x) = \frac{P(x \mid C_i)P(C_i)}{P(x)} = \frac{P(x \mid C_i)P(C_i)}{\sum_{k=1}^{K} P(x \mid C_k)P(C_k)}$$

- In summary, first calculate Σ_i, μ_i, and $P(C_i)$ using the training dataset;
- then, to predict the class of a new sample x, evaluate $P(x \mid C_i)$ for all classes Ci, and evaluate the following equation:

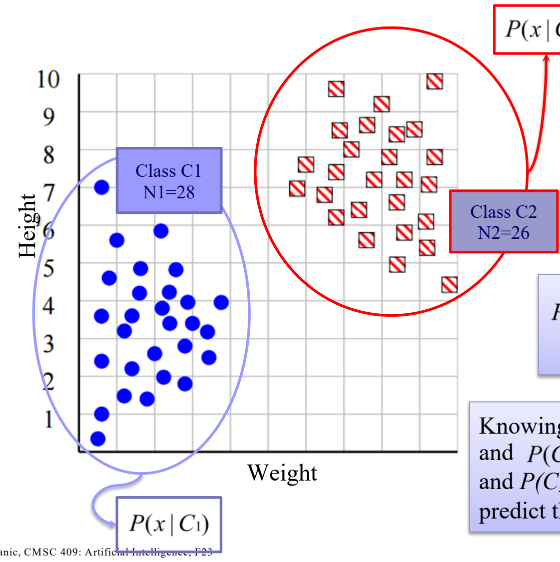
$$g_i(x) = \log P(x \mid C_i) + \log P(C_i)$$

- then we predict that the sample x belongs to the class i (class C_i corresponding to the maximum $g_i(x)$)
- which in turn is equivalent to the maximum of

$$P(x \mid C_i)P(C_i)$$

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

Multivariate Classification



N= N1+N2= 54

P(C1)=N1/N=0.52

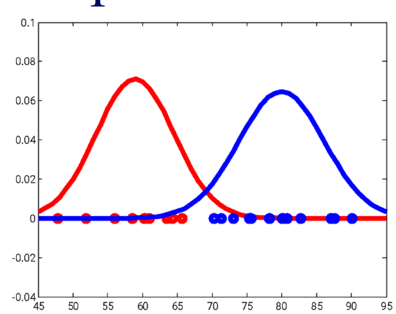
P(C2)=N2/N=0.48

$P(x \mid C_1)$ and $P(x \mid C_2)$ can be modeled as Gaussians.

Knowing $P(x \mid C_1)$, $P(x \mid C_2)$, $P(C_1)$, and $P(C_2)$, we can predict $P(C_1 \mid x)$ and $P(C_2 \mid x)$ using the Bayes rule to predict the class of new samples.

Example

Female(C1)	Male(C2)
56.0307	73.0330
64.2989	87.1268
60.3343	75.5307
51.8031	80.1888
47.8763	78.1822
58.5809	80.7478
65.7290	70.2774
60.9058	87.6195
60.2713	82.7291
63.4388	90.0496
	87.0834
	80.0574
	75.3048
	71.3055
	80.0849



When dealing with continuous data, we frequently assume that values follow Gaussian distribution.

For. ex., training data contains continuous attribute x. We can segment data by class, then compute mean & variance for each class. Probability distribution of some value v given a class C1 is: $P(x = v \mid C_1)$

For 1D data, the multivariate Gaussian reduces to:

$$P(x = v \mid C_1) = \frac{1}{\sigma_1\sqrt{2\pi}} \exp\left[-\frac{(v-\mu_1)^2}{2(\sigma_1)^2}\right]$$

Example

Assuming that samples were drawn from Gaussian *pdf* (probability density distribution)

Female(C1)	Male(C2)
56.0307	73.0330
64.2989	87.1268
60.3343	75.5307
51.8031	80.1888
47.8763	78.1822
58.5809	80.7478
65.7290	70.2774
60.9058	87.6195
60.2713	82.7291
63.4388	90.0496
	87.0834
	80.0574
	75.3048
	71.3055
	80.0849

The ***mean*** calculated from the data set:

For Female	For Male
$\mu_1 = 58.92$	$\mu_2 = 79.95$

Standard deviation calculated from the dataset

For Female	For Male
$\sigma_1 = 5.623$	$\sigma_2 = 6.164$

$$\sigma = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(x_k-\mu)^2}$$

$$P(x \mid C_1) = \frac{1}{\sigma_1\sqrt{2\pi}}\exp\left[-\frac{(x-\mu_1)^2}{2(\sigma_1)^2}\right]$$

$$P(x \mid C_2) = \frac{1}{\sigma_2\sqrt{2\pi}}\exp\left[-\frac{(x-\mu_2)^2}{2(\sigma_2)^2}\right]$$

$$P(C_1) = \frac{10}{25}$$

$$P(C_2) = \frac{15}{25}$$

To predict the class of a new sample x, we just evaluate *P(x|C1)P(C1)* and *P(x|C2)P(C2)*;

The one that is larger, corresponds to the predicted class

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

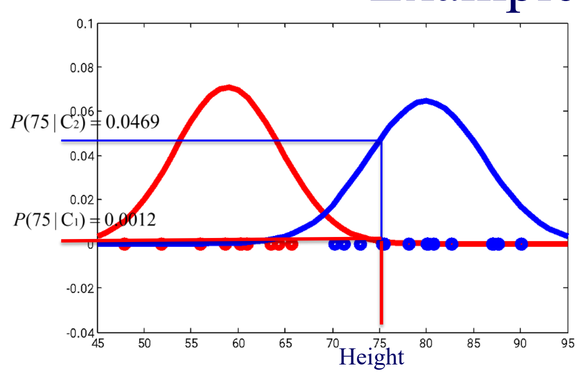
© M. Manic, CMSC 409: Artificial Intelligence, F23

Session 24,25, Updated on 11/14/23 7:23:38 AM

29

Example

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



$$P(x \mid C_1) = \frac{1}{\sigma_1\sqrt{2\pi}}\exp\left[-\frac{(x-\mu_1)^2}{2(\sigma_1)^2}\right]$$

$$P(x \mid C_2) = \frac{1}{\sigma_2\sqrt{2\pi}}\exp\left[-\frac{(x-\mu_2)^2}{2(\sigma_2)^2}\right]$$

$$P(C_1) = \frac{10}{25} \qquad P(C_2) = \frac{15}{25}$$

Input: 75

$P(75|C_1) = 0.0012$

$P(75|C_2) = 0.0468$

$P(75 \mid C_1)P(C_1) = 4.8000\text{e-}04$

$P(75 \mid C_2)P(C_2) = 0.0281$

We predict that the subject with height 75 belongs to class C2, because

$P(75 \mid C_1)P(C_1) < P(75 \mid C_2)P(C_2)$

$$g_1(75) = \log(P(x \mid C_1)) + \log(P(C_1))$$

$$g_1(75) = \log 0.0012 + \log\frac{10}{25} = -7.6417$$

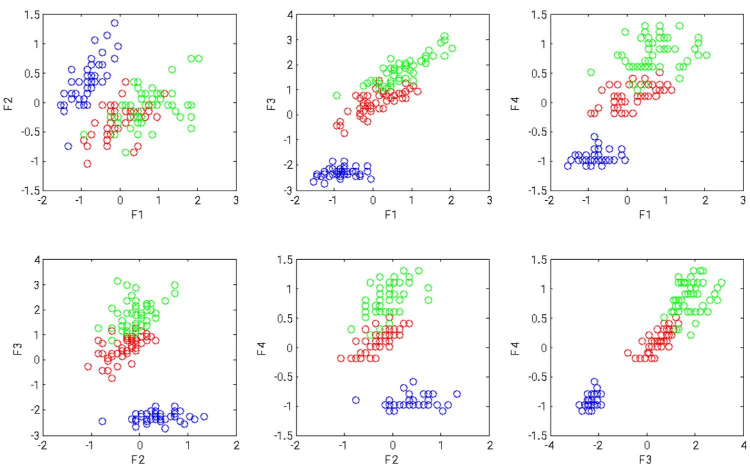
$$g_2(75) = \log(P(x \mid C_2)) + \log(P(C_2))$$

$$g_2(75) = \log 0.0469 + \log\frac{15}{25} = -3.5706$$

© M. Manic, CMSC 409: Artificial Intelligence, F23

30

Example #1: Iris data set



https://archive.ics.uci.edu/ml/datasets/Iris

31

Learning from Neighbors

- Eager vs. lazy learners
- Lazy learners
  - *K-nearest-neighbor classifier*
  - *Case based reasoning (CBR)*
  - *Distance measures*
    - *Euclidian space, coding theory, fuzzy space*
    - *Euclidian, Manhattan, Chebyshev, angle distance*

32

Lazy Learners – Learning from Neighbors

Eager vs. lazy learners

- Eager learners
  - *Decision trees, ANNs, SVMs, association rules*
  - *The model is defined before unseen patterns arrive (eager to classify new patterns)*
  - *Essential part of the work done in training phase*

- Lazy learners
  - *Store training pattern and waits until testing pattern arrives to cluster/predict…*
  - *Storage/computation expensive, good fit for parallel execution*
  - *Incremental learning, learning by analogy*
  - *Essential part of the work done essentially in testing phase*

Lazy Learners – k-Nearest-Neighbor Classifier

k-Nearest-Neighbor Classifier

Algorithm

- Searches for *k* training patterns most similar to testing pattern
- For *k*=1: unknown pattern assigned to the closest single pattern's class
- For *k*=n: classifies unknown pattern as belonging to
  - a major class of neighbors
  - average of *k* similar patterns
- Both classification and prediction

Similarity

- *Similarity based on certain similarity measure or distance metrics*
- *Various distance measures*
  - *Hamming, Euclidean, Manhatan*

Normalization

- *If pattern attributes of significantly different ranges – normalize*

Lazy Learners – k-Nearest-Neighbor Classifier

k-Nearest-Neighbor Classifier

Distance for categorical attributes

- Such as color (e.g. distance between blue and green, black and white)
- Hamming distance (1 or 0), or grade (black & white maps to [0,1] range)

Missing attributes

- If both comparable attribute from two patterns are missing , *dist.*=1
- If one missing, then *dist.*=|*attrib*-1|

Determining *k*

- *Heuristics*
- *k=1, 2, 3,…until satisfies error criterion (min error)*
- *Cases:*

$$N_{patterns} \to \infty, \;\; k = 1$$

$$N_{patterns} \to \infty, \;\; k = \infty$$

Lazy Learners – k-Nearest-Neighbor Classifier

k-Nearest-Neighbor Classifier

Attribute weighting

- *Each attribute carries same importance*
- *Better - lower weighting of irrelevant attributes*
- *Pruning of irrelevant patterns*

Complexity

- For a DB of *D* patterns and *k*=1, *O(D)*
- If patterns organized in search tree, then *O(log(D)),*
  - *growth of a decision tree is O(log(n)) when there are n leaves*
- Parallelization reduces *O* (up to *O*(1))

Partial distance

- Distance between *n* attributes only (if these prove to be above threshold, remaining attributes are not checked)

Editing

- Pruning of irrelevant, redundant training patterns

Lazy Learners – Case-Based Reasoning

Case-Based Reasoning (CBR)

Algorithm

- Based on a DB of problem solutions (cases)
- (in *k*-nearest-neighbors, patterns are stored)
- E.g. case based law, medical case based treatments and diagnosis, engineering diagnostic problems (tech help)
- When unseen case is to be classified, a DB of similar cases is searched
- If identical training case is found, the accompanying solution is returned
- If no identical case is found,
  - the closest (neighboring) solution is returned
  - E.g. for solutions as graphs – subgraphs that are similar are searched for
- Problems
  - More training cases
  - Accuracy vs. efficiency

Lazy Learners – Distance Measures

Distances - Similarity measures

- Similarity based on certain similarity or distance metrics
- Various distance measures
  - **Euclidian space**
    - *Manhattan (1-norm)*
    - *Euclidean (2-norm)*
    - *Minkowski (p-norm)*
    - *Infinity-norm*
  - **Coding theory**
    - *Hamming*
  - **Fuzzy space**
    - *Fuzzy measures*

$$Dist_{Manhattan(p=1)} = \sum_{i=1}^{n} |x_i - y_i|$$

$$Dist_{Euclidian(p=2)} = \sqrt[2]{\sum_{i=1}^{n} |x_i - y_i|^2}$$

$$l^p(X,Y) = \sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p}$$

$$\lim_{p\to\infty}\left(l^p(X,Y)\right) = \lim_{p\to\infty}\left(\sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p}\right) =$$

$$= \max(|x_1 - y_1|, |x_2 - y_2|, \ldots, |x_n - y_n|)$$

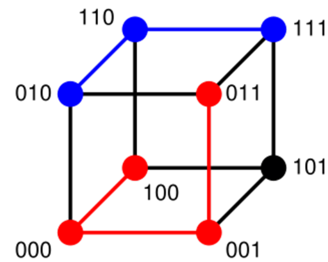
$$Dist_{Hamming(p=1)} = \left(\sum_{i=1}^{n} x_i \cdot y_i \;|\; x_i, y_i \in (0,1)\right)$$

Lazy Learners – Distance Measures

Distances - *Hamming distance*

- *For two **strings** of equal length, HD is the number of positions for which the corresponding symbols are **different**.*
- *For binary strings: metric space for n-length binary strings - Hamming cube; HD= number of ones in a xor b*

- *E.g.*
  - **100->011 has distance 3 (red path)**
  - **010->111 has distance 2 (blue path)**
  - **HD=2**
    - **1011101**
    - **1001001**
  - **HD=3**
    - **2143896**
    - **2233796**
  - **HD=3**
    - **"toned“**
    - **"roses"**

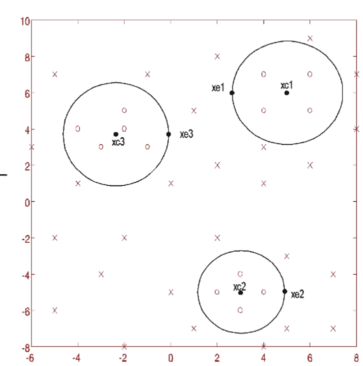


Lazy Learners – Distance Measures

Distances – *Eucledian distance*

- ***Euclidian (Pythagorean) geometry**, considered by the Greek mathematician Euclid (300 BC)*
- *“ordinary” distance that can be measured by ruler*
- *Based on Pythagorean theorem*
- *2-norm distance*

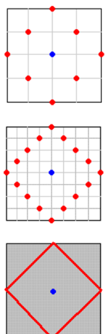
$$Dist_{Euclidian(p=2)} = \sqrt[2]{\sum_{i=1}^{n} |x_i - y_i|^2}$$



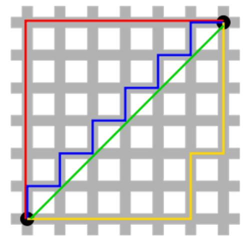
Lazy Learners – Distance Measures

Distances – ***Manhattan or Citiblock distance***

- ***Taxicab geometry****, considered by Hermann Minkowski in the 19th century*
- *Unlike Euclidean D., distances not squared (large difference in one dimension less likely to dominate the total distance)*



- *E.g.*
  - **red, blue, yellow = 12**
  - **green line** $\sqrt{6^2 + 6^2} \approx 8.48$



L_1 metric (norm)

	Taxicab geometry	Euclidian geom.
shape	Square	Circle
one side length	r*sqr(2)	/
Circumference (area)	4*r*sqr(2) (area= 2*r^2)	2*r*pi (area= r^2*pi)

Figures taken from
http://en.wikipedia.org/wiki/Taxicab_geometry

What about taxicab ellipse, hyperbola, parabola, equilateral triangles?

Lazy Learners – Distance Measures

Distances – ***Chebyshev (Tchebychev) distance***

- *Russian mathematician (18th century)*
- *Greatest distance of differences between two vectors along any coordinate dimension.*
- *In 2-dim space -* ***chessboard distance*** *(for a king)*
- *Infinity-norm distance*

$$Dist_{Chebyshev} = \max_i(|x_i - y_i|) = \lim_{k\to\infty} \sqrt[k]{\sum_{i=1}^{n} |x_i - y_i|^k}$$

(L_∞ metric)

number of moves a king requires

Figure taken from
http://en.wikipedia.org/wiki/Chebyshev_distance

Lazy Learners – Distance Measures

Distances in chess

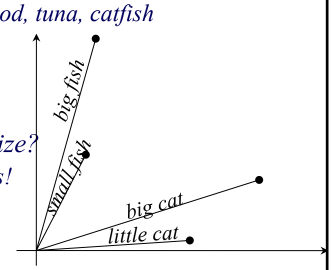
- *Distance between squares on the chessboard*
  - *To reach from one square to another, only kings require the number of moves equal to the distance; rooks, queens and bishops require one or two moves*
  - *For **rooks & bishops** (same color only) measured in Manhattan distance;*
  - *For **kings and queens** in Chebyshev distance*

pawn, rook, knight, bishop, queen, and king

Lazy Learners – Distance Measures

Distances – ***Angle distance***

- *Similarities in the way the fields within each record are related*
- E.g.
  - ***Species***
    - ***Sardines** should go with salmon, sardines, cod, tuna, catfish*
    - ***Kitten** should go with lions, tigers, cougars*
  - ***Size** (kitten with sardines)*
  - ***Whiskers** (kitten with catfish)*
- *How about the length of tail, body length, claw size?*
  - ***Single points** vs. **ratios of lengths** in each species!*



Angle

- ***Sine angle** rather than **magnitude***
- *Sine – relation*
  - *(0 & 180 different by constant factor -1);*
- *Cosine – correlation*
  - *(0 for orthogonal, 1 for parallel vectors)*

Example: EEG epilepsy interictal spike detection….

New Trends:

Can you explain AI?…

Explainable Artificial Intelligence (XAI)

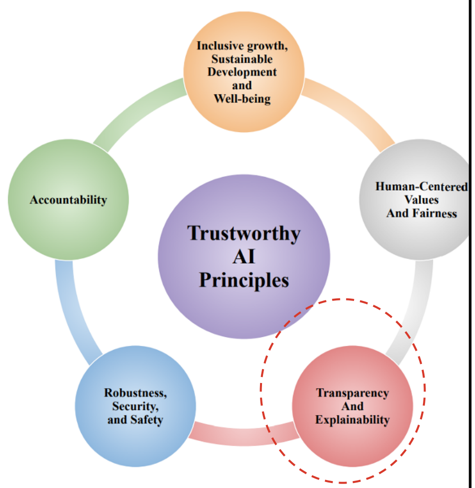
DARPA DEFENSE ADVANCED RESEARCH PROJECTS AGENCY

- Machines trusted today?
  - Today, numerous applications of AI exist in everyday lives
  - Industry, gaming, military
  - Gunning, DARPA*:* Can you understand, trust, manage *"artificially intelligent machine partners"?*
- DARPA XAI:
  - Produce explainable models while maintaining performance (prediction accuracy)
  - Enable users to understand, trust, and effectively manage…*machine partners*...
  - New ML systems that can explain their rationale, convey understanding of "*how they will behave in the future*"
  - Need to be combined with new "human computer interface" techniques
    - translate models into "understandable" and useful explanations

http://www.darpa.mil/program/explainable-artificial-intelligence

Interpretable AI/ Explainable AI (XAI)

- Tremendous success of AI in many domains
  - Medical Diagnosis, Fraud detection, Anomaly detection, Image recognition, Intelligent Transportation
- Humans are still hesitant to develop, deploy, and use AI methods
  - Humans can not understand the internal decision making process of ML methods -> **lack of trust**
- Specially, AI systems in safety-critical domains
  - Black-box models leading to a lack of trust
- **Interpretable AI/ Explainable AI (XAI)**
  - We need to crack open the black-box, models need to explain themselves to us
  - Allowing humans to fully utilize the benefits of AI



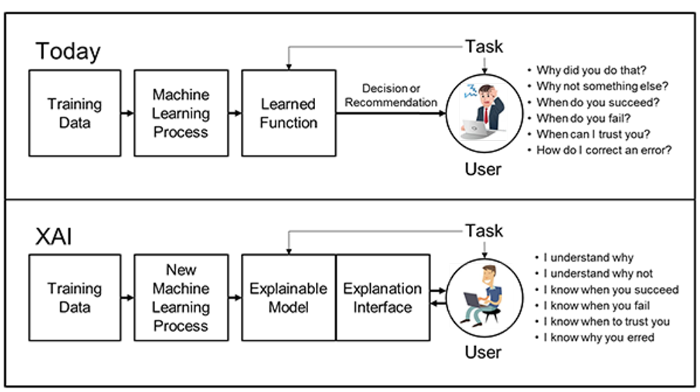
C. Wickramasinghe, D. Marino, J. Grandio, and M. Manic, "Trustworthy AI Development Guidelines for Human System Interaction," in Proc. 13th International Conference on Human System Interaction, IEEE HSI 2020, Tokyo, Japan, June 6-8. 2020

Explainable Artificial Intelligence (XAI)

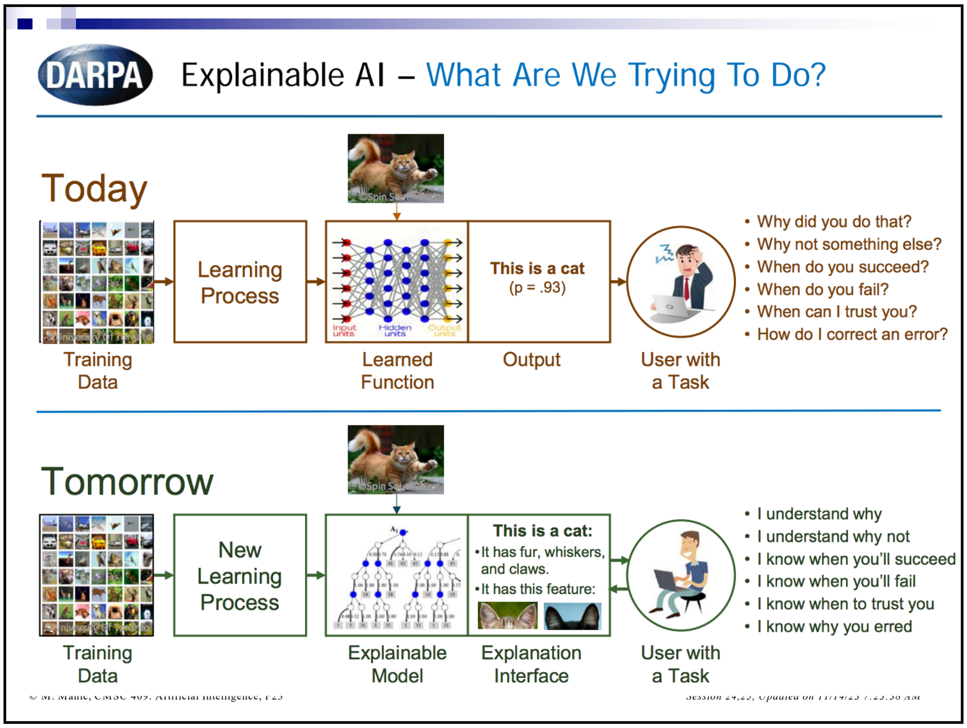
DARPA DEFENSE ADVANCED RESEARCH PROJECTS AGENCY

- DARPA XAI Challenges:
  1. ML problems for classification (heterogeneous, multimedia data)
  2. ML problems for decision policies for autonomous systems

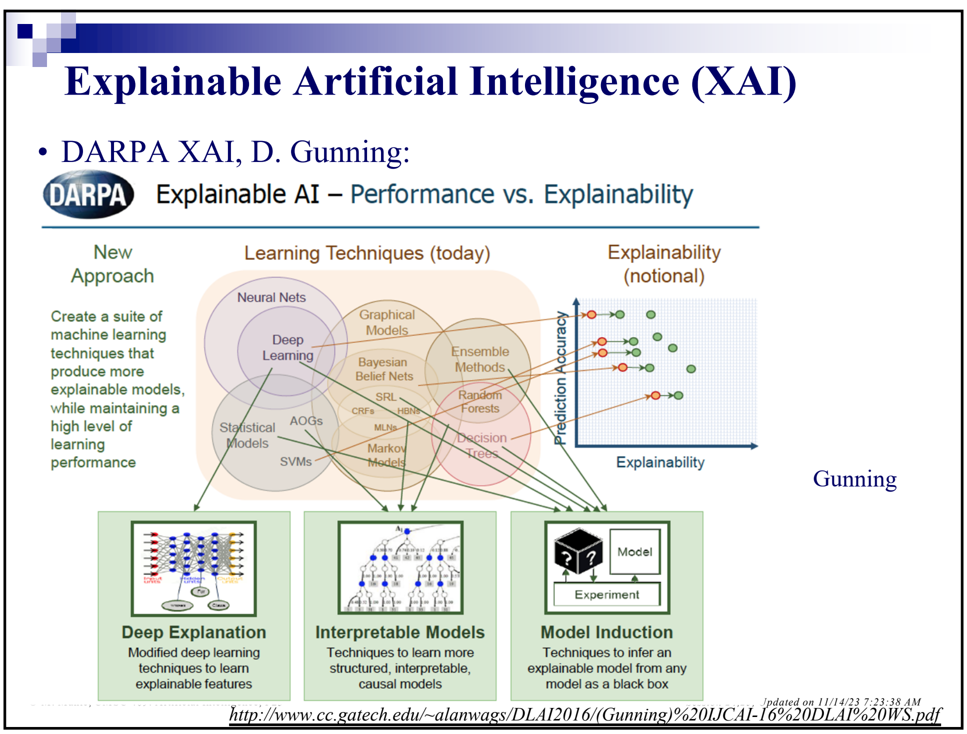
Focus: Classification and Reinforcement Learning



http://www.darpa.mil/program/explainable-artificial-intelligence


DARPA
Explainable AI – What Are We Trying To Do?
Today
Training Data
Learning Process
Learned Function
This is a cat (p = .93)
Output
User with a Task
• Why did you do that?
• Why not something else?
• When do you succeed?
• When do you fail?
• When can I trust you?
• How do I correct an error?
Tomorrow
Training Data
New Learning Process
Explainable Model
This is a cat:
•It has fur, whiskers, and claws.
•It has this feature:
Explanation Interface
User with a Task
• I understand why
• I understand why not
• I know when you'll succeed
• I know when you'll fail
• I know when to trust you
• I know why you erred

49


Explainable Artificial Intelligence (XAI)
• DARPA XAI, D. Gunning:
DARPA
Explainable AI – Performance vs. Explainability
New Approach
Create a suite of machine learning techniques that produce more explainable models, while maintaining a high level of learning performance
Learning Techniques (today)
Neural Nets
Deep Learning
Graphical Models
Bayesian Belief Nets
Ensemble Methods
Random Forests
SRL
CRFs
HBNs
MLNs
Statistical Models
AOGs
SVMs
Markov Models
Decision Trees
Explainability (notional)
Prediction Accuracy
Explainability
Gunning
Deep Explanation
Modified deep learning techniques to learn explainable features
Interpretable Models
Techniques to learn more structured, interpretable, causal models
Model
Experiment
Model Induction
Techniques to infer an explainable model from any model as a black box
http://www.cc.gatech.edu/~alanwags/DLAI2016/(Gunning)%20IJCAI-16%20DLAI%20WS.pdf

50

XAI for Cyber Physical Systems

- Cyber-Physical Systems (CPSs) play a vital role in our modern **critical infrastructures**
- CPSs produce **massive amounts of data,**
  - **Opportunities to use predictive ML models**: performance monitoring and optimization, preventive maintenance, and threat detection
  - **Data is usually unlabelled** -> relying on **supervised learning alone is not sufficient**
  - **Interpretable models** are **essential** for high-risk environments where the model outcomes can result in severe consequences.

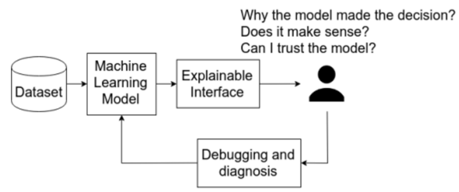


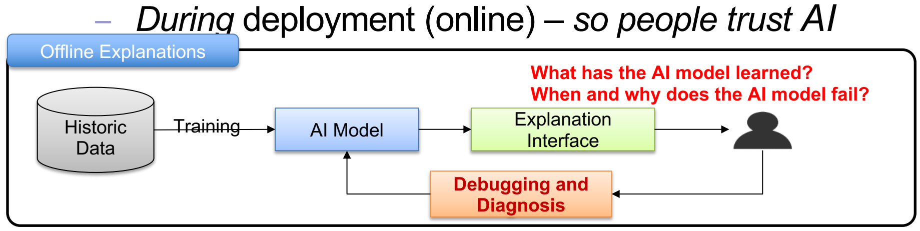
Figure 25: Explainable AI in Anomaly Detection Systems

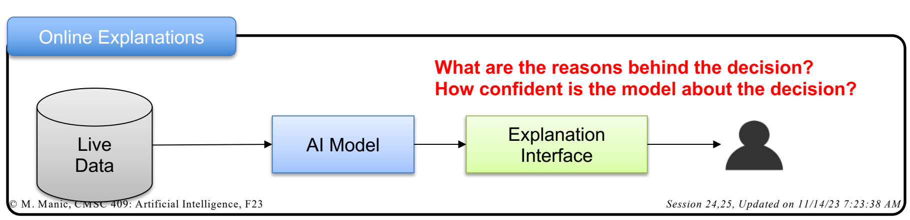
D. Marino, C. Wickramasinghe, M. Manic, "An Adversarial Approach for Explainable AI in Intrusion Detection Systems" in Proc. 44th Annual Conference of the IEEE Industrial Electronics Society, IECON 2018, Washington DC, USA, Oct. 21-23, 2018

- For these reasons, it is necessary to have ML models that are **explainable** and **unsupervised**
- Explainable ML models, supervised and unsupervised
  - Towards **human-in-the-loop** AI systems
  - Towards **AI** and **human** working **together**

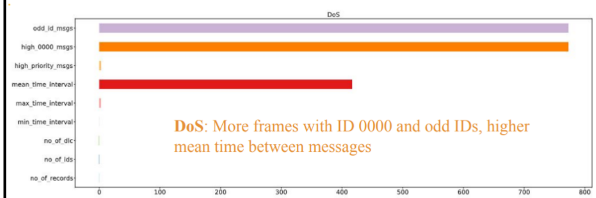
XAI for CPSs

- Explaining AI models at two stages
  - *Before* deployment (offline) – *so people will use AI*
  - *During* deployment (online) – *so people trust AI*

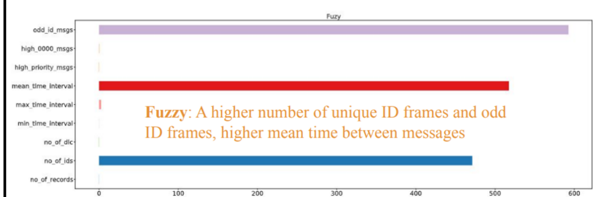




Interpretability



(a) DoS



(b) Fuzzy

Compared to baseline

- **DoS**: More frames with ID 0000 and odd IDs, higher mean time between messages.
- **Fuzzy**: A higher number of unique ID frames and odd ID frames, higher mean time between messages.

Distinguishing two attacks

- **DoS**: Higher number of odd ID frames that are coming from frames with ID 0000
- **Fuzzy**:
  - A highest mean time (530) compared to DoS (420).
  - Number of unique IDs is very high compared to DoS

C. S. Wickramasinghe *et al.*, "RX-ADS: Interpretable Anomaly Detection Using Adversarial ML for Electric Vehicle CAN Data," in *IEEE Transactions on Intelligent Transportation Systems*, doi: 10.1109/TITS.2023.3294349.

53

EXTRA SLIDES

54