

CSCE 580: Introduction to AI *CSCE 581: Trusted AI*

Lecture 18: Explanation, Machine Learning – Unsupervised

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

26TH OCT 2023

**Carolinian Creed: “I will practice personal and
academic integrity.”**

Credits: Copyrights of all material reused acknowledged

Organization of Lecture 18

- Introduction Segment
 - Recap of Lecture 17
- Main Segment
 - Trust/ Explanations, LIME - Recap
 - Unsupervised ML
 - Algorithms
- Concluding Segment
 - Course Project Discussion
 - About Next Lecture – Lecture 19
 - Ask me anything

Introduction Section

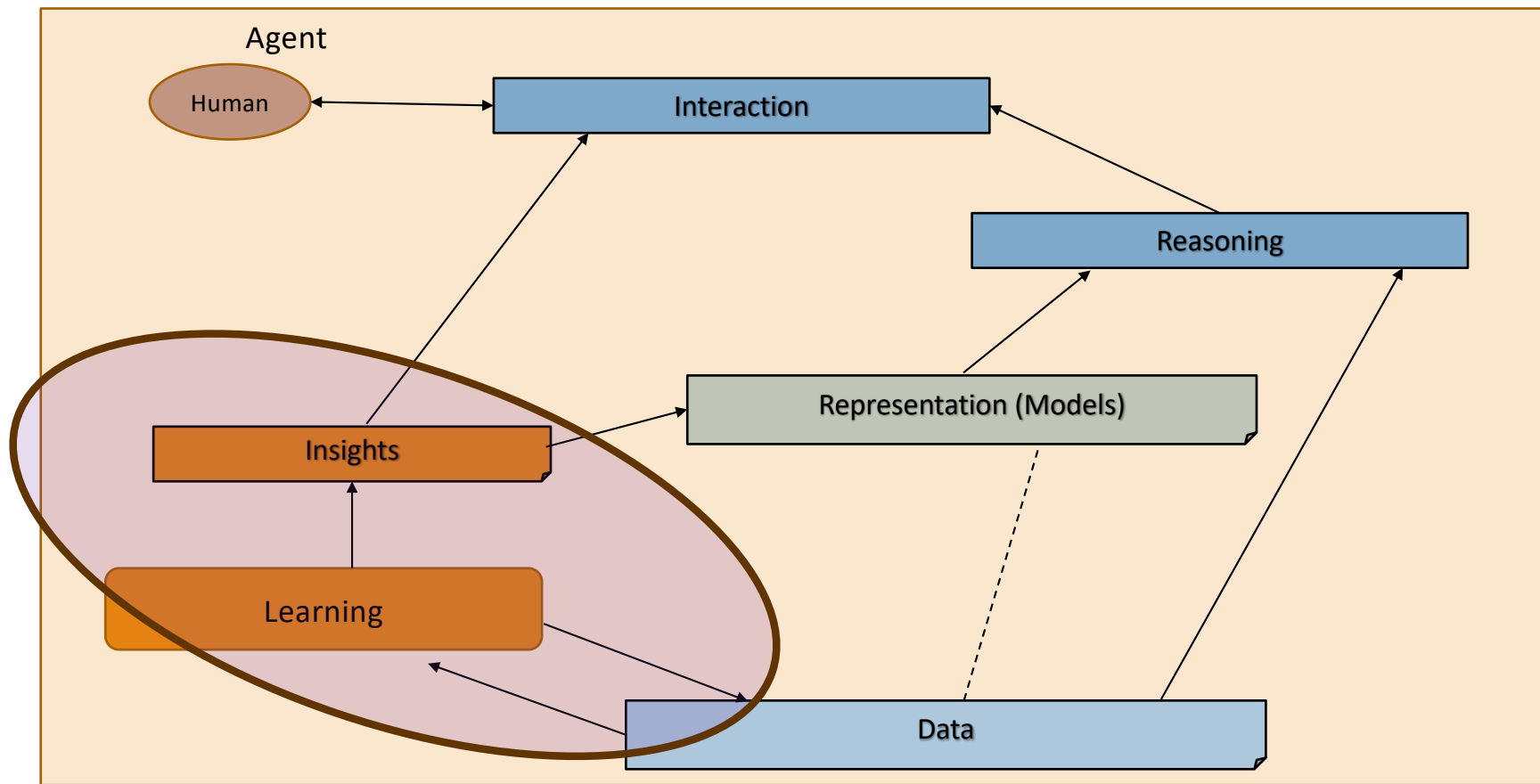
Recap of Lecture 17

- Topic discussed
 - Building Chatbots
 - Rasa
 - SafeChat Framework

Intelligent Agent Model



Relationship Between Main AI Topics



Where We Are in the Course

CSCE 580/ 581 – In This Course

- Week 1: Introduction, Aim: Chatbot / Intelligence Agent
- Weeks 2-3: Data: Formats, Representation and the Trust Problem
- Week 4-5: Search, Heuristics - Decision Making
- Week 6: Constraints, Optimization – Decision Making
- Week 7: Classical Machine Learning – Decision Making, Explanation
- Week 8: Machine Learning - Classification
- Week 9: Machine Learning - Classification – Trust Issues and

Mitigation Methods

- Topic 10: Learning neural network, deep learning, Adversarial attacks
- Week 11: Large Language Models – Representation, Issues
- Topic 12: Markov Decision Processes, Hidden Markov models -
Decision making
- Topic 13: Planning, Reinforcement Learning – Sequential decision
making
- Week 14: AI for Real World: Tools, Emerging Standards and Laws;
Safe AI/ Chatbots

Main Section

Credit: Retrieved from internet

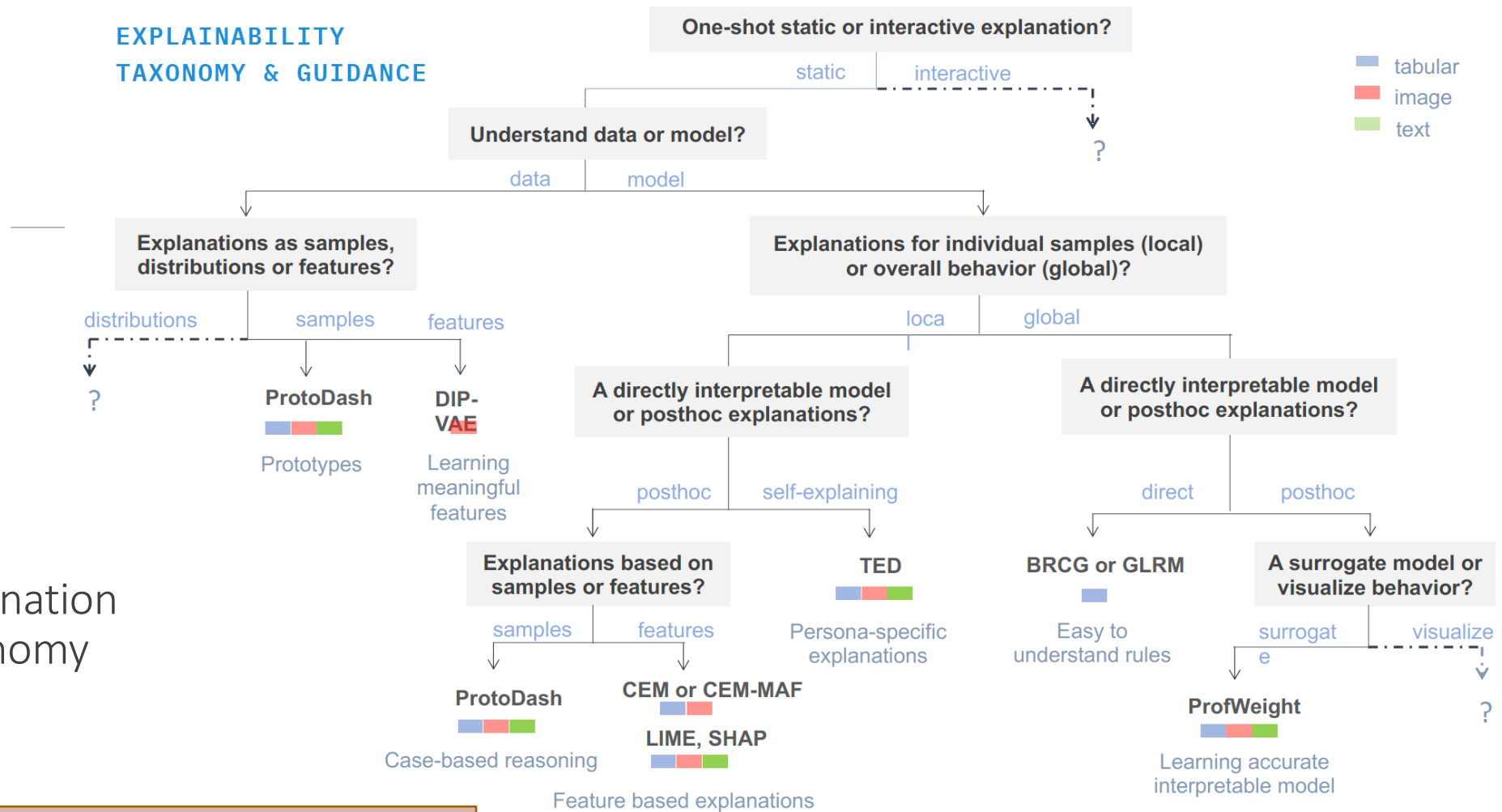
Recap: AI Trust/ Explanations

Types of Explanations

- **Feature-based**: from the features of the data, which feature(s) were most important for given decision output
 - Example: For a loan, is it income or the person's age ?
- **Sample-based**: from data in training, which data points were important for given test point; helps understand sampling and its representation in wider population
 - Example: For a loan, what instances similar to the loan application would have gotten the loan ?
- **Counter-factual**: what-ifs – what do you change about the input to change the decision output
 - Example: For a loan, does getting an additional borrower insurance increase chance of getting the loan?
- Natural language

Source: Explainable Machine Learning in Deployment, FAT* 2020

EXPLAINABILITY TAXONOMY & GUIDANCE



Explanation Taxonomy

Figure Credit: Diptikalyan Saha and Vijay Arya, Oct 2021

Machine Learning – Insights from Data

- Descriptive analysis
 - Describe a past phenomenon
 - **Methods:** classification (feedback from label), clustering, dimensionality reduction, anomaly detection, neural methods, reinforcement learning (feedback from hint/ reward)
- Predictive analysis
 - Predict about a new situation
 - **Methods:** time-series, neural networks
- Prescriptive analysis
 - What an agent should do
 - **Methods:** simulation, reinforcement learning, reasoning
- New areas
 - Counterfactual analysis
 - Causal Inferencing
 - Scenario planning

Unsupervised Machine Learning

- Group data into clusters/ classes without supervision
 - Limited supervision
- What is a good cluster ?
 - Samples within a cluster should be “**near**” to each other (**cohesiveness**)
 - Samples in a cluster should be “**far**” from other samples in other clusters. (**distinctiveness**)

Data Representation

- Data matrix representation
 - N objects (data rows) x p attributes (columns)
 - Similar to classification
- Dissimilarity matrix
 - Object x Object structure
 - $D(i, j)$ is difference or dissimilarity between (i, j) , 0 means similar and 1 means dissimilar

Clustering for Data Understanding and Applications

- Biology: taxonomy of living things: kingdom, phylum, class, order, family, genus and species
- Information retrieval: document clustering
- Land use: Identification of areas of similar land use in an earth observation database
- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- City-planning: Identifying groups of houses according to their house type, value, and geographical location
- Earth-quake studies: Observed earth quake epicenters should be clustered along continent faults
- Climate: understanding earth climate, find patterns of atmospheric and ocean
- Economic Science: market resarch

Content: Jiawei Han, Micheline Kamber and Jian Pei
Data Mining: Concepts and Techniques, 3rd ed.

Clustering as a Preprocessing Tool (Utility)

- Summarization:
 - Preprocessing for regression, PCA, classification, and association analysis
- Compression:
 - Image processing: vector quantization
- Finding K-nearest Neighbors
 - Localizing search to one or a small number of clusters
- Outlier detection
 - Outliers are often viewed as those “far away” from any cluster

Content: Jiawei Han, Micheline Kamber and Jian Pei
Data Mining: Concepts and Techniques, 3rd ed.

Considerations for a Clustering Algorithm

- Need a distance measure for *far* and *near*
- Be able to explain what a cluster means
- Handle different types of attributes: numeric, categorical (nominal, ordinal), binary
- Detect different shapes of clusters
- Handle noisy data
- Scale
 - Size
 - Dimensions

Major Clustering Approaches (I)

Partitioning approach:

- Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors
- Typical methods: **k-means**, k-medoids, CLARANS

Hierarchical approach:

- Create a hierarchical decomposition of the set of data (or objects) using some criterion
- Typical methods: Diana, Agnes, **BIRCH**, CAMELEON

Density-based approach:

- Based on connectivity and density functions
- Typical methods: **DBSCAN**, OPTICS, DenClue

Grid-based approach:

- based on a multiple-level granularity structure
- Typical methods: STING, WaveCluster, CLIQUE

Content: Jiawei Han, Micheline Kamber and Jian Pei
Data Mining: Concepts and Techniques, 3rd ed.

Major Clustering Approaches (II)

Model-based:

- A model is hypothesized for each of the clusters and tries to find the best fit of that model to each other
- Typical methods: **EM**, SOM, COBWEB

Frequent pattern-based:

- Based on the analysis of frequent patterns
- Typical methods: p-Cluster

User-guided or constraint-based:

- Clustering by considering user-specified or application-specific constraints
- Typical methods: COD (obstacles), constrained clustering

Link-based clustering:

- Objects are often linked together in various ways
- Massive links can be used to cluster objects: **SimRank**, LinkClus

Content: Jiawei Han, Micheline Kamber and Jian Pei
Data Mining: Concepts and Techniques, 3rd ed.

Partitioning Algorithms: Basic Concept

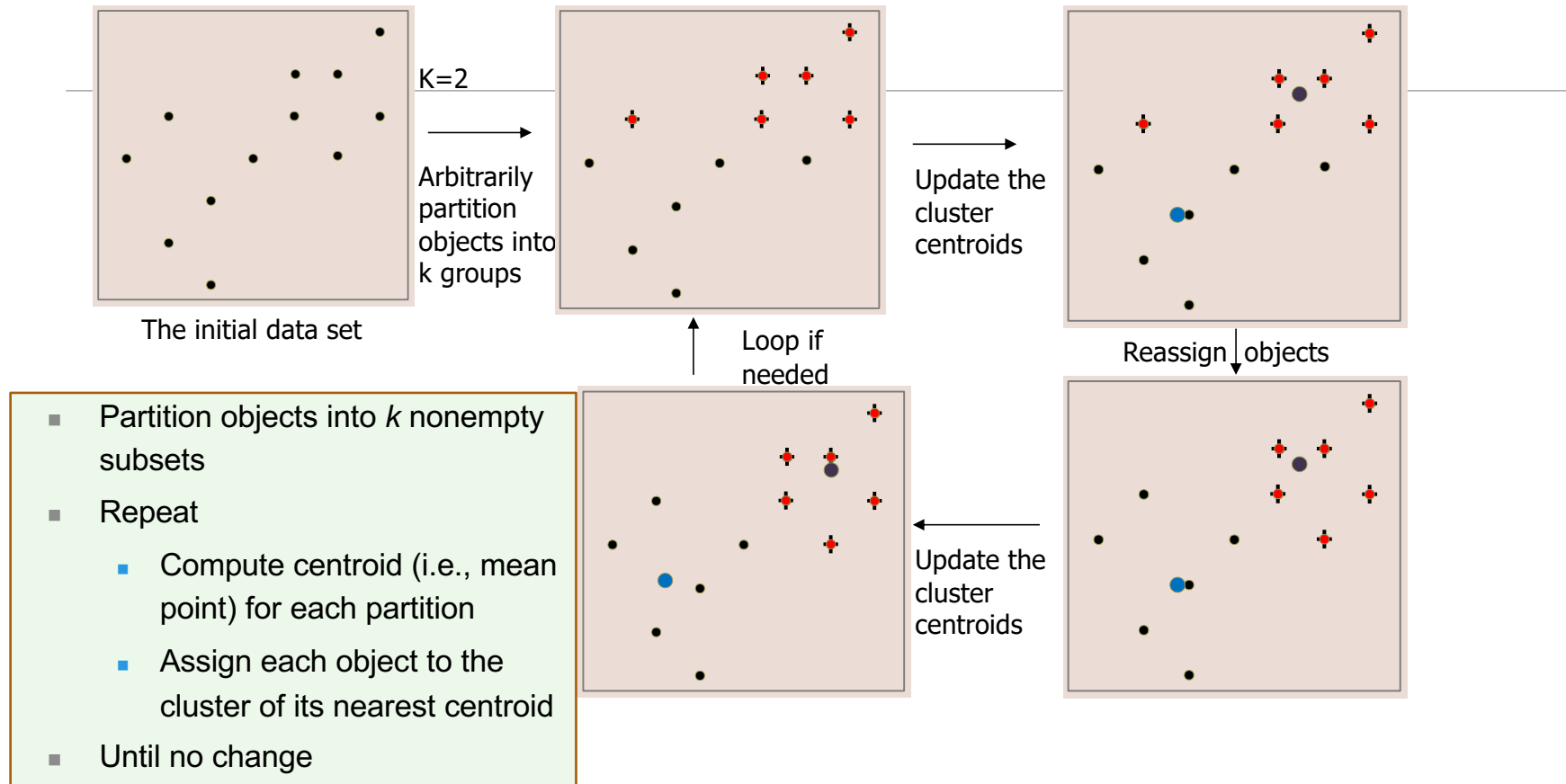
Partitioning method: Partitioning a database ***D*** of ***n*** objects into a set of ***k*** clusters, such that the sum of squared distances is minimized (where c_i is the centroid or medoid of cluster C_i)

$$E = \sum_{i=1}^k \sum_{p \in C_i} (p - c_i)^2$$

Given k , find a partition of k clusters that optimizes the chosen partitioning criterion

- Global optimal: exhaustively enumerate all partitions
- Heuristic methods: *k-means* and *k-medoids* algorithms
- *k-means* (MacQueen'67, Lloyd'57/'82): Each cluster is represented by the center of the cluster
- *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

An Example of *K-Means* Clustering



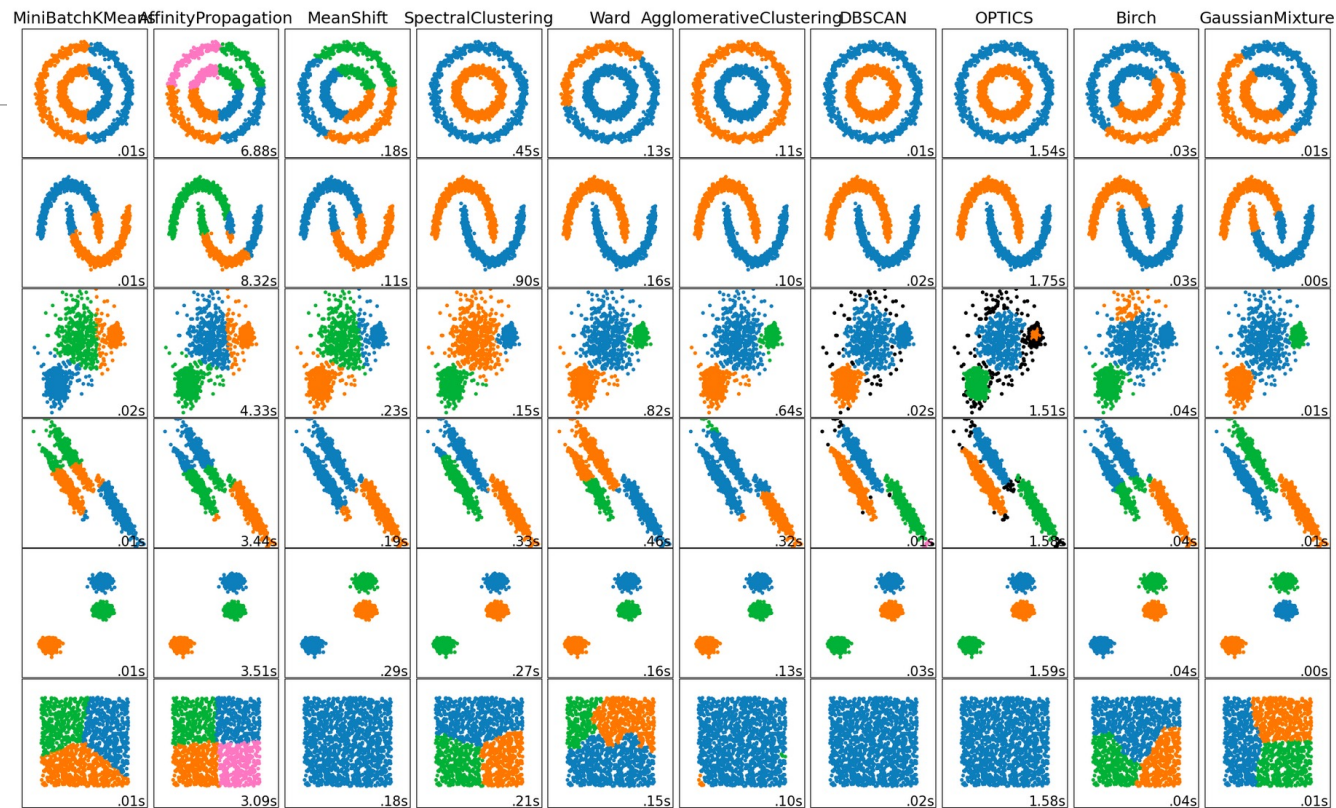
Comments on the *K-Means* Method

- **Strength:** *Efficient:* $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, $k, t \ll n$.
 - Comparing: PAM: $O(k(n-k)^2)$, CLARA: $O(ks^2 + k(n-k))$
- **Comment:** Often terminates at a *local optimal*.
- **Weakness**
 - Applicable only to objects in a continuous n -dimensional space
 - Using the k-modes method for categorical data
 - In comparison, k-medoids can be applied to a wide range of data
 - Need to specify k , the *number* of clusters, in advance (there are ways to automatically determine the best k (see Hastie et al., 2009)
 - Sensitive to noisy data and *outliers*
 - Not suitable to discover clusters with *non-convex shapes*

Exercise: Weka

- Use K-means on weather.arff
- Vary k

Snapshot of Clustering Methods



A comparison of the clustering algorithms in scikit-learn

Course Project

Project Discussion: What Problem Fascinates You ?

- Data
 - Water
 - Finance
 - ...
- Analytics
 - Search, Optimization, Learning, Planning, ...
- Application
 - Building chatbot
- Users
 - Diverse demographics
 - Diverse abilities
 - Multiple human languages

Project execution in sprints

- Sprint 1: (Sep 12 – Oct 5)
 - **Solving**: Choose a decision problem, identify data, work on solution methods
 - **Human interaction**: Develop a basic chatbot (no AI), no problem focus
- Sprint 2: (Oct 10 – Nov 9)
 - **Solving**: Evaluate your solution on problem
 - **Human interaction**: Integrated your choice of chatbot (rule-based or learning-based) and methods
- Sprint 3: (Nov 14 – 30)
 - **Evaluation**: Comparison of your solver chatbot with an LLM-based alternative, like ChatGPT

Project Discussion: Dates and Deliverables

Project execution in sprints

- Sprint 1: (Sep 12 – Oct 5)
 - **Solving**: Choose a decision problem, identify data, work on solution methods
 - **Human interaction**: Develop a basic chatbot (no AI), no problem focus
- Sprint 2: (Oct 10 – Nov 9)
 - **Solving**: Evaluate your solution on problem
 - **Human interaction**: Integrated your choice of chatbot (rule-based or learning-based) and methods
- Sprint 3: (Nov 14 – 30)
 - **Evaluation**: Comparison of your solver chatbot with an LLM-based alternative, like ChatGPT

- Oct 12, 2023
 - Project checkpoint
 - In-class presentation
- Nov 30, 2023
 - Project report due
- Dec 5 / 7, 2023
 - In-class presentation

Skeleton: A Basic Chatbot

- Run in an infinite loop until the user wants to quit
- Handle any user response
 - User can quit by typing “Quit” or “quit” or just “q”
 - User can enter any other text and the program has to handle it. The program should write back what the user entered and say – “I do not know this information”.
- Handle known user query types // Depends on your project
 - “Tell me about N-queens”, “What is N ?”
 - “Solve for N=4?”
 - “Why is this a solution? ”
- Handle chitchat // Support at least 5, extensible from a file
 - “Hi” => “Hello”
 - ...
- *Store session details in a file*

Illustrative Project

1. **Title:** Solve and explain solving of n-queens puzzle
2. **Key idea:** Show students how a course project will look like
3. **Who will care when done:** students of the course, prospective AI students and teachers
4. **Data need:** n: the size of game; interaction
5. **Methods:** search
6. **Evaluation:** correctness of solution, quality of explanation, appropriateness of chat
7. **Users:** with and without AI background; with and without chess background
8. **Trust issue:** user may not believe in the solution, may find interaction offensive (why queens, not kings? ...)

Project Discussion: Illustration

1. Create a private Github repository called “CSCE58x-Fall2023-<studentname>-Repo”. Share with Instructor (biplav-s) and TA (kausik-l)
2. Create Google folder called “CSCE58x-Fall2023-<studentname>-SharedInfo”. Share with Instructor (prof.biplav@gmail.com) and TA (lakkarajukausik90@gmail.com)
3. Create a Google doc in your Google repo called “Project Plan” and have the following by next class (Sep 5, 2023)

1. **Title:** Solve and explain solving of n-queens puzzle
2. **Key idea:** Show students how a course project will look like
3. **Who will care when done:** students of the course, prospective AI students and teachers
4. **Data need:** n: the size of game; interaction
5. **Methods:** search
6. **Evaluation:** correctness of solution, quality of explanation, appropriateness of chat
7. **Users:** with and without AI background; with and without chess background
8. **Trust issue:** user may not believe in the solution, may find interaction offensive (why queens, not kings? ...)

Project Illustration: N-Queens

- Sprint 1: (Sep 12 – Oct 5)
 - **Solving**: Choose a decision problem, identify data, work on solution methods
 - Method 1: Random solution
 - Method 2: Search – BFS
 - Method 3: Search - ...
 - **Human interaction**: Develop a basic chatbot (no AI) as outlined
 - Deliverable
 - Code structure in Github
 - ./data
 - ./code
 - ./docs
 - ./test
 - Presentation: Make sprint presentation on Oct 12, 2023

Reference: Project Rubric

- **Project results – 60%**
 - Working system ? – 30%
 - Evaluation with results superior to baseline? – 20%
 - Considered related work? – 10%
- **Project efforts – 40%**
 - Project report – 20%
 - Project presentation (updates, final) – 20%
- **Bonus**
 - Challenge level of problem – 10%
 - Instructor discretion – 10%
- **Penalty**
 - Lack of timeliness as per announced policy (right) - up to 30%

Milestones and Penalties

- Oct 12, 2023
 - Project checkpoint
 - In-class presentation
 - **Penalty: presentation not ready by Oct 10, 2023 [-10%]**
- Nov 30, 2023
 - Project report due
 - **Project report not ready by date [-10%]**
- Dec 5 / 7, 2023
 - In-class presentation
 - **Project presentations not ready by Dec 4, 2023 [-10%]**

Lecture 18: Summary

- We talked about
 - Building Chatbots
 - Rasa
 - SafeChat Framework

Concluding Section

About Next Lecture – Lecture 19

Lecture 19: Machine Learning - NN, Deep Learning

- Neural Networks
- Deep Learning
- Trust Issues