

## *CSCE 580: Introduction to AI*

### Lecture 12: Machine Learning

---

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

26<sup>TH</sup> SEP 2024

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

**Credits: Copyrights of all material reused acknowledged**

# Organization of Lecture 12

---

- Introduction Segment
  - Recap of Lecture 11
- Main Segment
  - Problem Settings
  - Data preparation and feature engineering
  - Solving classification problems
  - Quiz 2
- Concluding Segment
  - Course Project Discussion
  - About Next Lecture – Lecture 13
  - Ask me anything

# Introduction Section

---

# Recap of Lecture 11

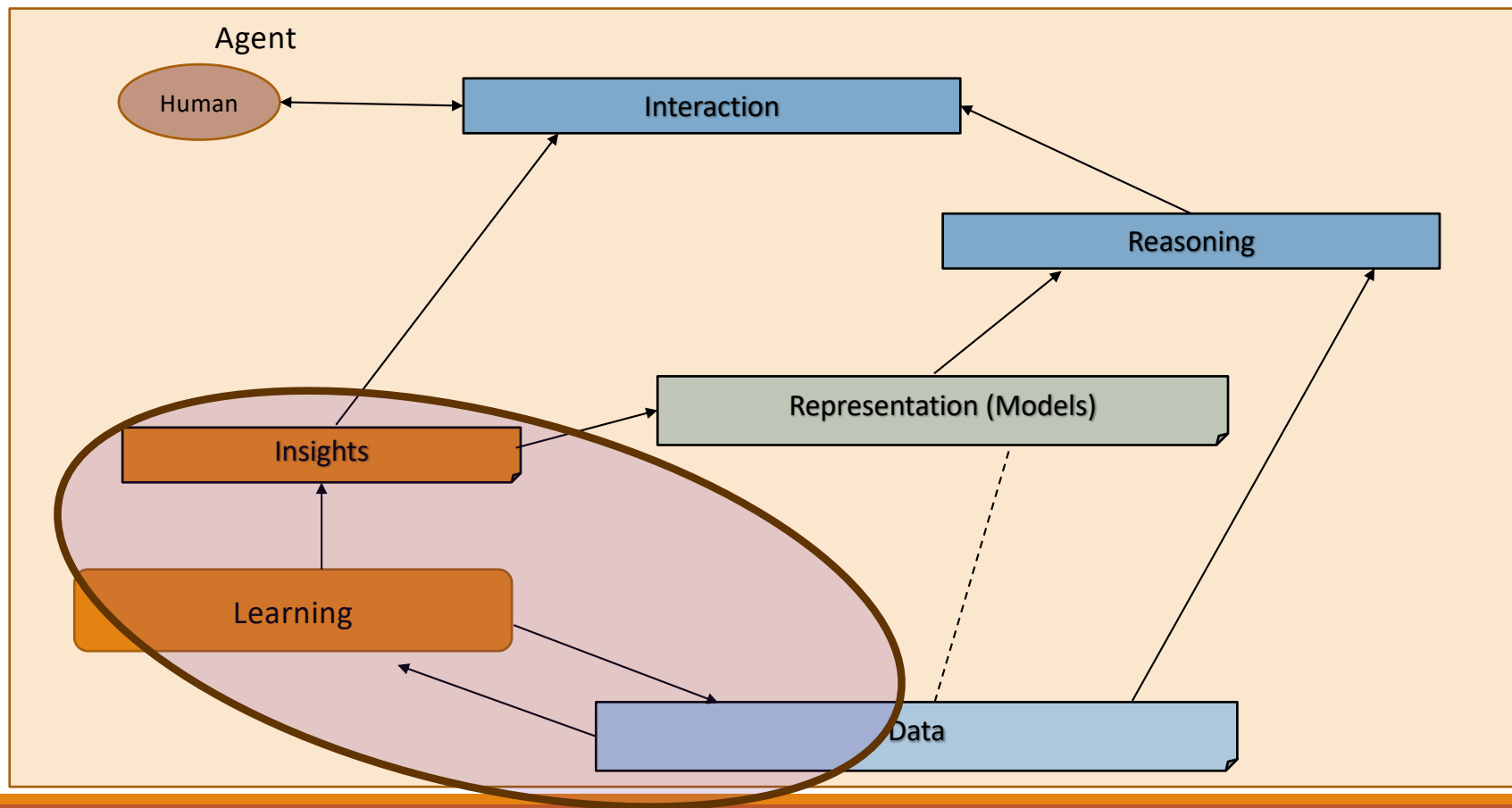
---

- Constraint Satisfaction Problem
- Optimization Problems

# 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



# 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

# Nomenclature

---

Column, Attribute, Feature

Row, Item

1	PID	ST_NUM	ST_NAME	OWN_OCCUPIED	NUM_BEDROOMS	NUM_BATH	SQ_FT
2	100001000	104	PUTNAM	Y	3	1	1000
3	100002000	197	LEXINGTON	N	3	1.5	--
4	100003000		LEXINGTON	N	n/a	1	850
5	100004000	201	BERKELEY	12	1	NaN	700
6		203	BERKELEY	Y	3	2	1600
7	100006000	207	BERKELEY	Y	NA	1	800
8	100007000	NA	WASHINGTON		2	HURLEY	950
9	100008000	213	TREMONT	Y	1	1	
10	100009000	215	TREMONT	Y	na	2	1800

# Types of Attributes/ Columns

- Numeric: has number as value in computational sense; all mathematical functions are valid.
  - Example: SQ\_FT
- Categorical: has distinct values
  - Nominal: each value is incomparable with other
    - Example: OWN\_OCCUPIED, ST\_NAME
  - Ordinal: the values can be ordered
    - Example: ST\_NUM, NUM\_BEDS
- Comment:
  - Q: what type is a binary variable?
  - A: depends on the semantics – nominal (gender), ordinal (number basements).

1	PID	ST_NUM	ST_NAME	OWN_OCCUPIED	NUM_BEDROOMS	NUM_BATH	SQ_FT
2	100001000	104	PUTNAM	Y	3	1	1000
3	100002000	197	LEXINGTON	N	3	1.5	--
4	100003000		LEXINGTON	N	n/a	1	850
5	100004000	201	BERKELEY	12	1	NaN	700
6		203	BERKELEY	Y	3	2	1600
7	100006000	207	BERKELEY	Y	NA	1	800
8	100007000	NA	WASHINGTON		2	HURLEY	950
9	100008000	213	TREMONT	Y	1	1	
10	100009000	215	TREMONT	Y	na	2	1800

# Why is Type of Variable Important

---

- Handling of missing values
- Distance between
  - Values
  - Data items
- Used for measuring accuracy, error
- Guiding the learning process
  - Selection of algorithms

# Concepts

---

- **Input data:** data available
  - **Training data:** used for training a learning algorithm and get a model
    - [Optional] **Validation data:** used to tune parameters
  - **Test data:** used to test a learning model
- **Classification problem**
  - Separating data into classes (also called labels, categorical types)
  - One of the attributes is the class label we are trying to learn
  - Class label is the **supervision**
- **Clustering problem**
  - We are trying to learn grouping of data
  - There is no attribute indicating membership in the groups (hence, **unsupervised**)
- **Prediction problem**
  - Learning value of a continuous variable

Reference: <https://machinelearningmastery.com/difference-test-validation-datasets/>  
<https://www2.seas.gwu.edu/~bell/csci243/lectures/classification.pdf>

# Sample Learning Task

---

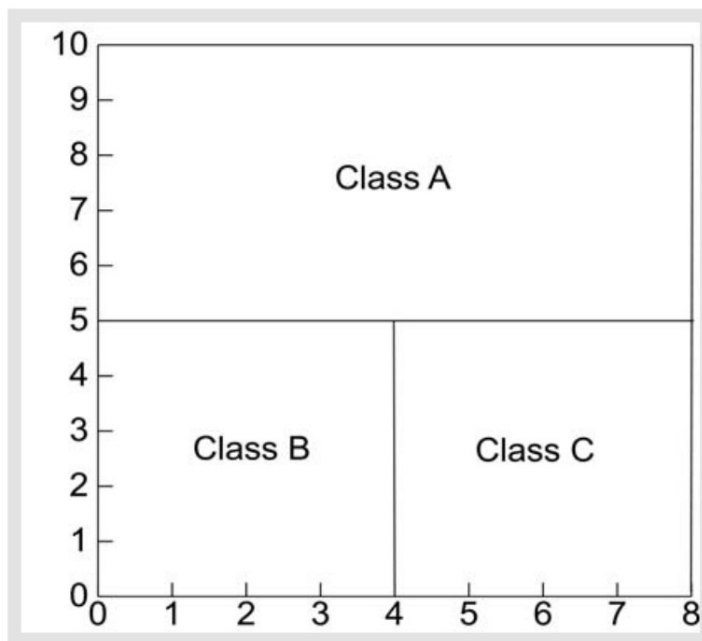
- COVID-19 data

Notebook: <https://github.com/biplav-s/course-d2d-ai/blob/main/sample-code/l6-l7-l8-supervised-ml/Supervised-Regression-Classification.ipynb>

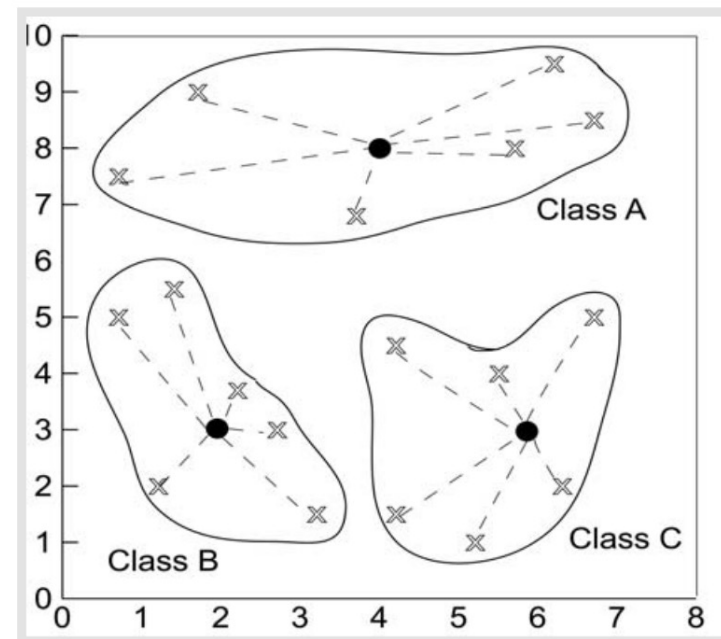
-

# Methods for Classification

**Partitioning Based**



**Distance Based**

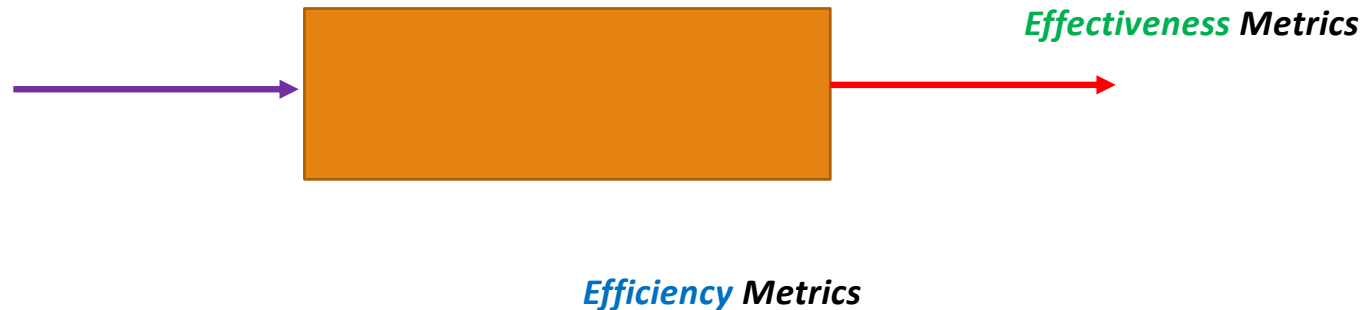


Source: <https://www2.seas.gwu.edu/~bell/csci243/lectures/classification.pdf>

# Metric Types

---

- **Effectiveness**: what the user of a system sees, primarily cares about
- **Efficiency**: what the executor in a system sees, primarily cares about





# Example: Predicting COVID cases

---

- **Effectiveness**: what the user of a system sees, primarily cares about
  - *How accurate (high) is the prediction?*
  - *How low is the error?*
- **Efficiency**: what the executor in a system sees, primarily cares about
  - *How low is the error?*
  - *How fast was prediction made?*
  - *How stable is the prediction to change in data?*

# Example: Detecting Spam in Email

---

- **Effectiveness**: what the user of a system sees, primarily cares about
  - *How many spams identified?*
  - *How many spams missed?*
- **Efficiency**: what the executor in a system sees, primarily cares about
  - *How fast were spams detected?*
  - *How much memory was used per million emails processed ?*

# Comparing Classification Methods

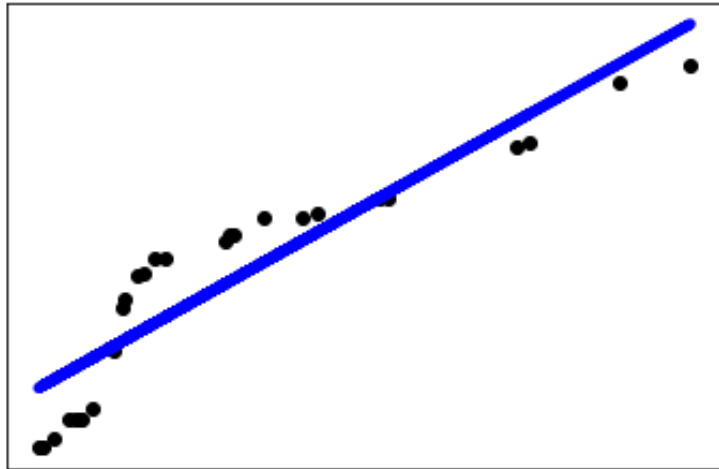
---

- Predictive accuracy
- Interpretability: providing insight
- Robustness: handling noisy data
- Speed
- Scalability: large volume of data

Source: Data Mining: Concepts and Techniques, by Jiawei Han and Micheline Kamber

# Linear Regression

---



Notebook: <https://github.com/biplav-s/course-d2d-ai/blob/main/sample-code/l6-l7-l8-supervised-ml/Supervised-Regression.ipynb>

# Metrics: Accuracy, Precision, Recall

Actual Class	Predicted class		
		Class = Yes	Class = No
	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

**Accuracy** =  
$$\frac{(TP+TN)}{(TP+FP+FN+TN)}$$

**Precision** =  
$$\frac{(TP)}{(TP+FP)}$$

**Recall** =  
$$\frac{(TP)}{(TP+FN)}$$

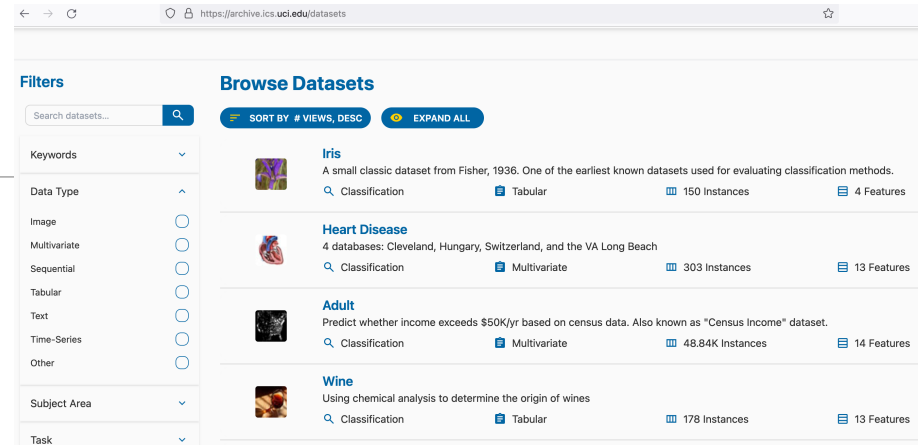
**F1 Score: Harmonic Mean**  
$$\frac{1}{F1} = \frac{1}{Precision} + \frac{1}{Recall}$$

$$F1 = \frac{2 * (Recall * Precision)}{(Recall + Precision)}$$

# Reference and Demo

- Data: UCI Datasets

- <https://archive.ics.uci.edu/datasets>
- Browse or search



## Weka 3: Machine Learning Software in Java

Weka is a collection of machine learning algorithms for data mining tasks. It contains tools for data preparation, classification, regression, clustering, association rules mining, and visualization.

Found only on the islands of New Zealand, the Weka is a flightless bird with an inquisitive nature. The name is pronounced like this, and the bird sounds like this.

Weka is open source software issued under the GNU General Public License.

We have put together several free online courses that teach machine learning and data mining using Weka. The videos for the courses are available on Youtube.

Weka supports deep learning!

### Getting started

- Requirements
- Download
- Documentation
- FAQ
- Getting Help

### Further information

- Citing Weka
- Datasets
- Related Projects
- Miscellaneous Code
- Other Literature

### Developers

- Development
- History
- Subversion
- Contributors
- Commercial licenses

- Tools:

- Weka - <https://www.cs.waikato.ac.nz/ml/weka/>
- Download tool and dataset

- Libraries

- Scikit - <https://scikit-learn.org/stable/>

# Reference and Demo

- Data: UCI Datasets
  - <https://archive.ics.uci.edu/datasets>
  - Browse or search
- Tools:
  - Weka - <https://www.cs.waikato.ac.nz/ml/weka/>
  - Download tool and dataset
- Libraries
  - Scikit - <https://scikit-learn.org/stable/>

The screenshot shows the scikit-learn website homepage. The header includes the scikit-learn logo and navigation links: Install, User Guide, API, Examples, Community, and More. The main banner features the text "scikit-learn" and "Machine Learning in Python", along with buttons for "Getting Started", "Release Highlights for 1.3", and "GitHub". To the right of the banner, a list of features is displayed: "Simple and efficient tools for predictive data analysis", "Accessible to everybody, and reusable in various contexts", "Built on NumPy, SciPy, and matplotlib", and "Open source, commercially usable - BSD license". Below the banner, three main sections are visible: "Classification" (describing object categorization with applications like spam detection and algorithms like gradient boosting), "Regression" (describing continuous-valued attribute prediction with applications like drug response and algorithms like gradient boosting), and "Clustering" (describing automatic grouping of objects with applications like customer segmentation and algorithms like k-means).

# Exercise: German Credit

---

- Check in UCI
- Look at variants
  - <https://archive.ics.uci.edu/dataset/573/south+german+credit+update>



# Lecture 12: Summary

---

- We talked about
  - Problem Settings
  - Data preparation and feature engineering
  - Solving classification problems
- Quiz 2

# Course Project

---

# Discussion: Projects

---

- New: two projects
  - Project 1: model assignment
  - Project 2: single problem/ llm based solving / fine-tuning/ presenting result

# Project Discussion

---

1. Go to Google spreadsheet against your name
2. Enter model assignment name and link from (<http://modelai.gettysburg.edu/> )

1. Create a private Github repository called “CSCE58x-Fall2024-<studentname>-Repo”. Share with Instructor (biplav-s) and TA (vishalpallagani)
2. Create Google folder called “CSCE58x-Fall2024-<studentname>-SharedInfo”. Share with Instructor ([prof.biplav@gmail.com](mailto:prof.biplav@gmail.com)) and TA ([vishal.pallagani@gmail.com](mailto:vishal.pallagani@gmail.com))
3. Create a Google doc in your Google repo called “Project Plan” and have the following by next class (Sep 5, 2024)

## Timeline

1. Title:
2. Key idea: (2-3 lines)
3. Data need:
4. Methods:
5. Evaluation:
6. Milestones
  1. // Create your own
7. Oct 3, 2024

# Reference: Project 1 Rubric (30% of Course)

## Assume total for Project-1 as 100

- **Project results** – 60%
  - Working system ? – 30%
  - Evaluation with results superior to baseline? – 20%
  - Went through project tasks completely ? – 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 your milestones policy (right) - up to 30%

## Milestones and Penalties

- Project plan due by Sep 5, 2024 [-10%]
- Project deliverables due by Oct 3, 2024 [-10%]
- Project presentation on Oct 8, 2024 [-10%]

# About Next Lecture – Lecture 13

---

# Lecture 13: Machine Learning

- Structured Data: Supervised Methods
  - Decision trees/ random forest
  - The variety of methods
  - Choosing a method that works
- Reading material:
  - “Which ML to Use” with title: Data-driven advice for applying machine learning to bioinformatics problems  
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5890912/>
  - “10 tips with title”: Ten quick tips for machine learning in computational biology  
<https://biodatamining.biomedcentral.com/articles/10.1186/s13040-017-0155-3>

7	Sep 10 (Tu)	Search - Uninformed
8	Sep 12 (Th)	Search - Informed; Heuristics
9	Sep 17 (Tu)	Local search
10	Sep 19 (Th)	Adversarial games and search
11	Sep 24 (Tu)	Constraints & optimization
12	Sep 26 (Th)	Machine Learning - Basics
13	Oct 1 (Tu)	Machine Learning – Classification – Decision Trees, Random Forest
14	Oct 3 (Th)	Machine Learning – Classification – NBC, Gradient Boosting, ML- Text