

## *CSCE 581: Introduction to Trusted AI*

### Lectures 9 and 10: Quiz 1, (Supervised) ML - Trust

---

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

11<sup>TH</sup> AND 13<sup>TH</sup> FEB, 2025

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

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

# Organization of Lectures 9, 10

---

- Introduction Section
  - Recap from Week 4 (Lectures 7 and 8)
  - Announcements and News
- Main Section
  - L9: Supervised ML, Quiz 1
  - L10: Trust issues in (Supervised) ML
- Concluding Section
  - About next week – Lectures 11, 12
  - Ask me anything

# Introduction Section

---

# Recap from Week 4 (Lectures 7, 8 )

---

- We looked at
  - Supervised ML algorithms, ML tools
  - Deep-dive into German credit
- Project descriptions finalized

# AI News

---

- SuperBowl and AI
  - Google faked Gemini AI output in Super Bowl ad, <https://www.theverge.com/news/608188/google-fake-gemini-ai-output-super-bowl>
  - Super Bowl 2025: AI Wars Bring ChatGPT, Google To Game, Watch Top Ads, <https://www.forbes.com/sites/martineparis/2025/02/06/chatgpt-to-debut-super-bowl-ad-google-edits-ai-watch-42-top-spots/>

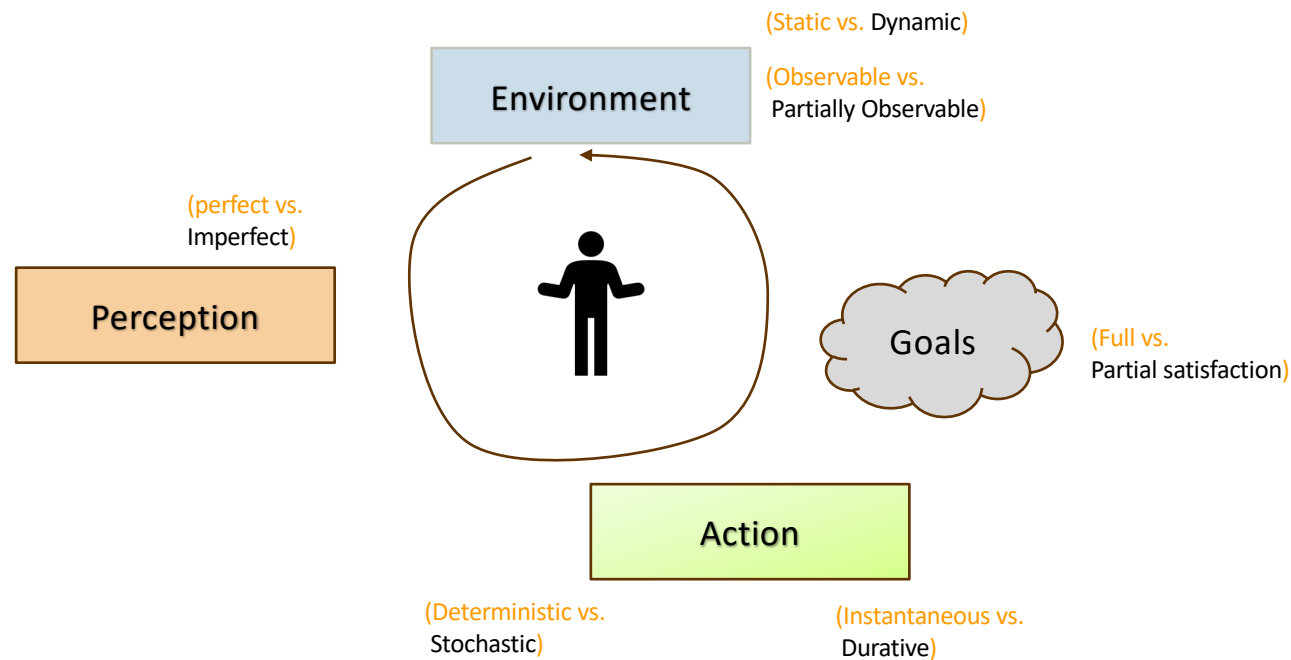
## Announcement: Change to Student Assessment

A = [920-1000]  
B+ = [870-919]  
B = [820-869]  
C+ = [770-819]  
C = [720-769]  
D+ = [670-719]  
D = [600-669]  
F = [0-599]

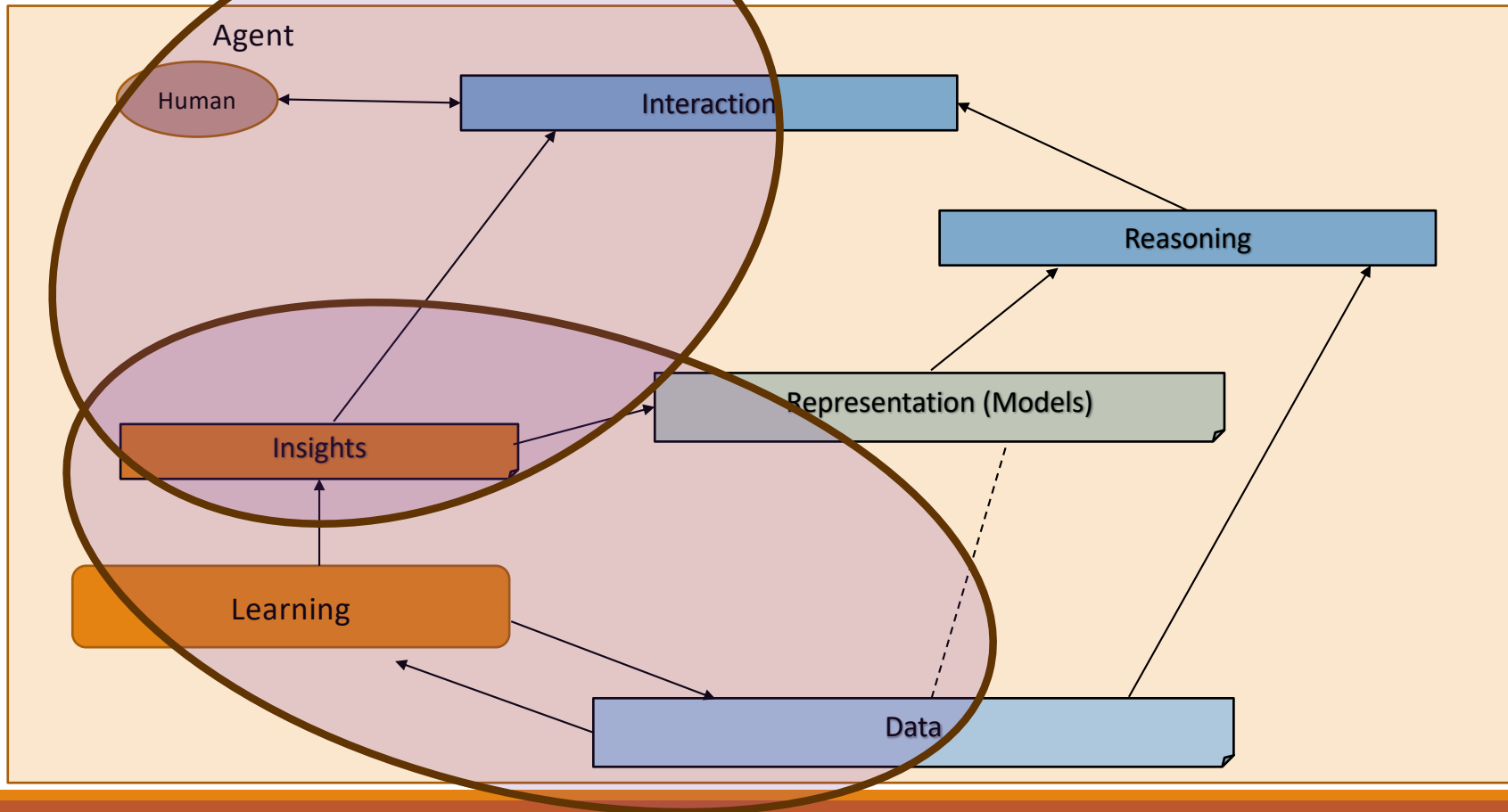
Tests	Undergrad	Grad
Course Project – report, in-class presentation	600	600
Quiz – 2 quizzes	200	200
Final Exam	200	100
Additional Final Exam – Paper summary, in-class presentation		100
Total	1000 points	1000 points

**Change:** 4 quizzes to 2; no best of 3

# Intelligent Agent Model



## Relationship Between Main AI Topics (Covered in Course)





# High Level Semester Plan (Adapted, Approximate)

## CSCE 581 –

- Week 1: Introduction
- Week 2: Background: AI - Common Methods
- Week 3: The Trust Problem
- Week 4: Machine Learning (Structured data) - Classification
- Week 5: Machine Learning (Structured data) - Classification – Trust Issues
- Week 6: Machine Learning (Structured data) – Classification – Mitigation Methods
- Week 7: Machine Learning (Structured data) – Classification – Explanation Methods
- Week 8: Machine Learning (Text data, **vision**) – Classification,

### **Large Language Models**

- Week 9: Machine Learning (Text data) - Classification – Trust Issues, LLMs
- Week 10: Machine Learning (Text data) – Classification – Mitigation Methods
- Week 11: Machine Learning (Text data) – Classification – Explanation Methods
- Week 12: Emerging Standards and Laws, **Real world applications**
- Week 13: Project presentations
- Week 14: Project presentations, Conclusion

AI/ ML topics and with a  
focus on fairness, explanation,  
Data privacy, reliability

# Main Section

---

# Supervised ML – Methods Continued

---

# Random Forest

---

- An ensemble method
- Credits
  - Ideas introduced by Tin Kam Ho in 1995, [https://en.wikipedia.org/wiki/Tin\\_Kam\\_Ho](https://en.wikipedia.org/wiki/Tin_Kam_Ho)
  - Matured by Leo Breiman and Adele Cutler at Berkeley ([https://www.stat.berkeley.edu/~breiman/RandomForests/cc\\_home.htm#intro](https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#intro))
  - History: Khaled Fawagreh, Mohamed Medhat Gaber & Eyad Elyan (2014) Random forests: from early developments to recent advancements, Systems Science & Control Engineering, 2:1, 602-609, DOI: [10.1080/21642583.2014.956265](https://doi.org/10.1080/21642583.2014.956265)
- Main steps (Input: data, N= number of trees)
  - If the number of cases in the training set is N, sample N cases at random - but *with replacement*, from the original data. This sample will be the training set for growing the tree.
  - If there are M input variables, a number  $m \ll M$  is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing.
  - Each tree is grown to the largest extent possible. There is no pruning.

**Slide Courtesy:** Leo Breiman and Adele Cutler website

# Random Forest in Action

---

- Code examples:
  - <https://github.com/biplav-s/course-d2d-ai/blob/main/sample-code/l6-l7-supervised-ml/Supervised-RandomForest-Classification.ipynb>
- Scikit Library: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

# Activity: Try Weka and Classifiers

---

- Naïve Bayes Method
- Gradient Tree Boosting
- Neural Network – MLP

# Naïve Bayes Classifier

## Notation:

Class variable  $y$  and  
dependent feature vector  $x_1$  through  $x_n$

Using the naive conditional independence assumption that

$$P(y | x_1, \dots, x_n) = \frac{P(y)P(x_1, \dots, x_n | y)}{P(x_1, \dots, x_n)}$$

$$P(x_i | y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i | y),$$

for all  $i$ , this relationship is simplified to

$$P(y | x_1, \dots, x_n) = \frac{P(y) \prod_{i=1}^n P(x_i | y)}{P(x_1, \dots, x_n)}$$

**Bayes assumption:** given the  
value of the class variable,  
every pair of features are  
conditionally independent

Since  $P(x_1, \dots, x_n)$  is constant given the input, we can use the following classification rule:

$$P(y | x_1, \dots, x_n) \propto P(y) \prod_{i=1}^n P(x_i | y)$$

↓

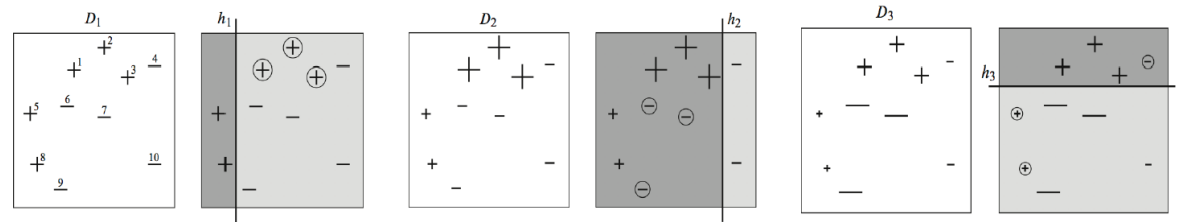
$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^n P(x_i | y),$$

**Source:** [https://scikit-learn.org/stable/modules/naive\\_bayes.html](https://scikit-learn.org/stable/modules/naive_bayes.html)

# Boosting Methods

- Concepts

- **Weak learner:** a classifier that is only slightly correlated with the true classification
  - label examples better than random guessing
- **Strong learner:** a classifier that is (arbitrarily) well-correlated with the true classification.



- Boosting

- “Convert weak learners to strong learners”
- Adapt[at]ive Resampling and Combining algorithm

Figure: AdaBoost. Source: Figure 1.1 of [Schapire and Freund, 2012]

**Source:** [https://en.wikipedia.org/wiki/Boosting\\_\(machine\\_learning\)](https://en.wikipedia.org/wiki/Boosting_(machine_learning))

Image Courtesy: Prof. Cheng Li



# Boosting Methods

**Gradient Boosting = Gradient Descent + Boosting**

Adaboost

$$H(x) = \sum_t \rho_t h_t(x)$$

- ▶ Fit an additive model (ensemble)  $\sum_t \rho_t h_t(x)$  in a forward stage-wise manner.
- ▶ In each stage, introduce a weak learner to compensate the shortcomings of existing weak learners.
- ▶ In Adaboost, “shortcomings” are identified by high-weight data points.

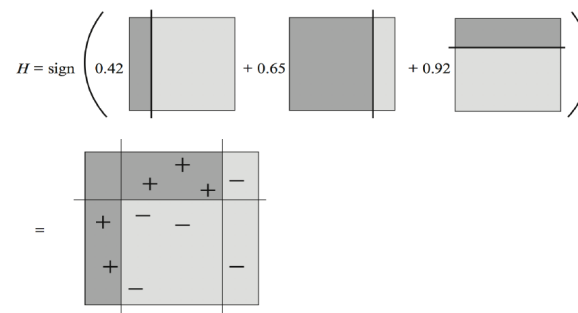


Figure: AdaBoost. Source: Figure 1.2 of [Schapire and Freund, 2012]

Content and Image Courtesy: Prof. Cheng Li

[https://www.ccs.neu.edu/home/vip/teach/MLcourse/4\\_boosting/slides/gradient\\_boosting.pdf](https://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/gradient_boosting.pdf)

# Boosting Methods

Gradient Boosting = Gradient Descent + Boosting

Adaboost

## AdaBoost,

*Illustration: for binary classification, images*

1. Form a large set of simple features
2. Initialize weights for training images
3. For T rounds
  1. Normalize the weights
  2. For available features from the set, train a classifier using a single feature and evaluate the training error
  3. Choose the classifier with the lowest error
  4. Update the weights of the training images: increase if classified wrongly by this classifier, decrease if correctly
4. Form the final strong classifier as the linear combination of the T classifiers (coefficient larger if training error is small)

**Source:** [https://en.wikipedia.org/wiki/Boosting\\_\(machine\\_learning\)](https://en.wikipedia.org/wiki/Boosting_(machine_learning))

$$H(x) = \sum_t \rho_t h_t(x)$$

$$H = \text{sign} \left( 0.42 \begin{array}{|c|c|c|} \hline \text{dark} & \text{light} & \text{light} \\ \hline \end{array} + 0.65 \begin{array}{|c|c|c|} \hline \text{dark} & \text{dark} & \text{light} \\ \hline \end{array} + 0.92 \begin{array}{|c|c|c|} \hline \text{light} & \text{light} & \text{light} \\ \hline \end{array} \right)$$
  
$$= \begin{array}{|c|c|c|} \hline + & + & - \\ \hline + & - & - \\ \hline + & - & - \\ \hline \end{array}$$

Figure: AdaBoost. Source: Figure 1.2 of [Schapire and Freund, 2012]

Image Courtesy: Prof. Cheng Li

[https://www.ccs.neu.edu/home/vip/teach/MLcourse/4\\_boosting/slides/gradient\\_boosting.pdf](https://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/gradient_boosting.pdf)

# Boosting Methods

**Gradient Boosting = Gradient Descent + Boosting**  
Adaboost

**Gradient Boosting = Gradient Descent + Boosting**  
Gradient Boosting

- ▶ Fit an additive model (ensemble)  $\sum_t \rho_t h_t(x)$  in a forward stage-wise manner.
- ▶ In each stage, introduce a weak learner to compensate the shortcomings of existing weak learners.
- ▶ In Gradient Boosting, “shortcomings” are identified by gradients.
- ▶ Recall that, in Adaboost, “shortcomings” are identified by high-weight data points.
- ▶ Both high-weight data points and gradients tell us how to improve our model.

$$H(x) = \sum_t \rho_t h_t(x)$$

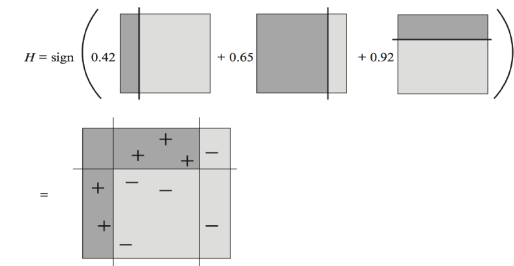


Figure: AdaBoost. Source: Figure 1.2 of [Schapire and Freund, 2012]

Content and Image Courtesy: Prof. Cheng Li  
[https://www.ccs.neu.edu/home/vip/teach/MLcourse/4\\_boosting/slides/gradient\\_boosting.pdf](https://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/gradient_boosting.pdf)

# NBC and Boosting in Action

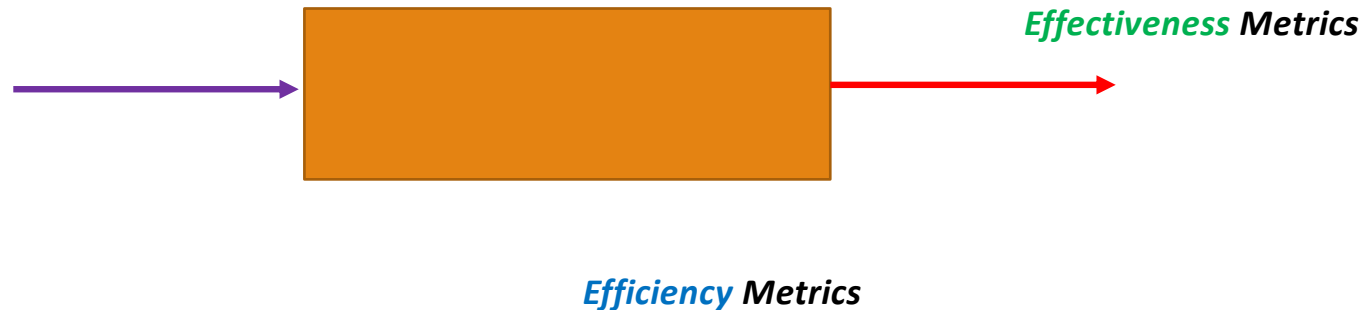
---

- Code examples:
  - <https://github.com/biplav-s/course-d2d-ai/blob/main/sample-code/l6-l7-l8-supervised-ml/Supervised-NaiveBayes-GradientBoost-NN-Classification.ipynb>
- Scikit Library:
  - NBC: [https://scikit-learn.org/stable/modules/naive\\_bayes.html](https://scikit-learn.org/stable/modules/naive_bayes.html)
  - GradientBoost: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html>

# Metric Types

---

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



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

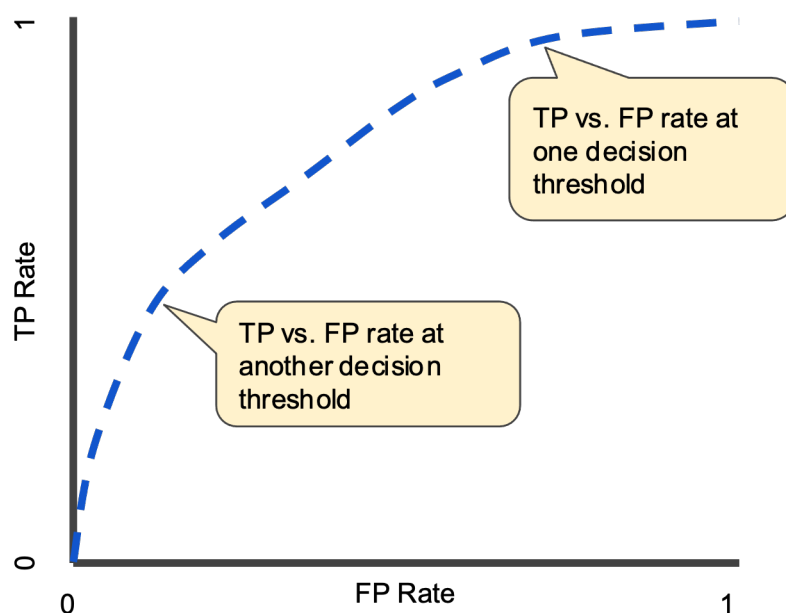
**Recall** =  
$$\frac{(TP)}{(TP+FN)}$$

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

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

# ROC – Receiver Operating Characteristic curve

An ROC curve plots TPR vs. FPR at different classification thresholds



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

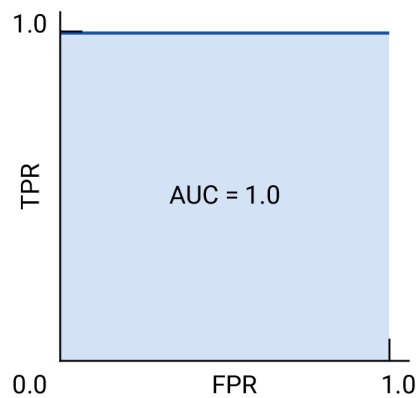
**False Positive Rate =**  
$$\frac{(FP)}{(FP+TN)}$$

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

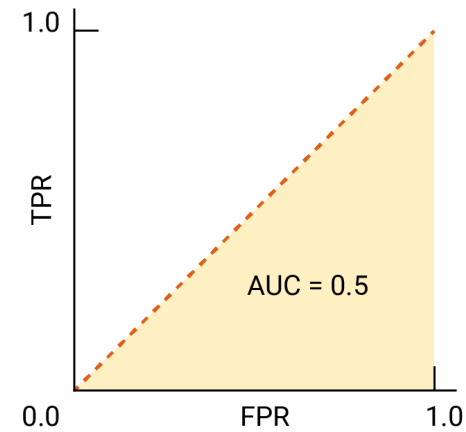
Source: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>

# AUC/ ROC Examples

ROC and AUC of a perfect system



ROC and AUC of completely random guesses



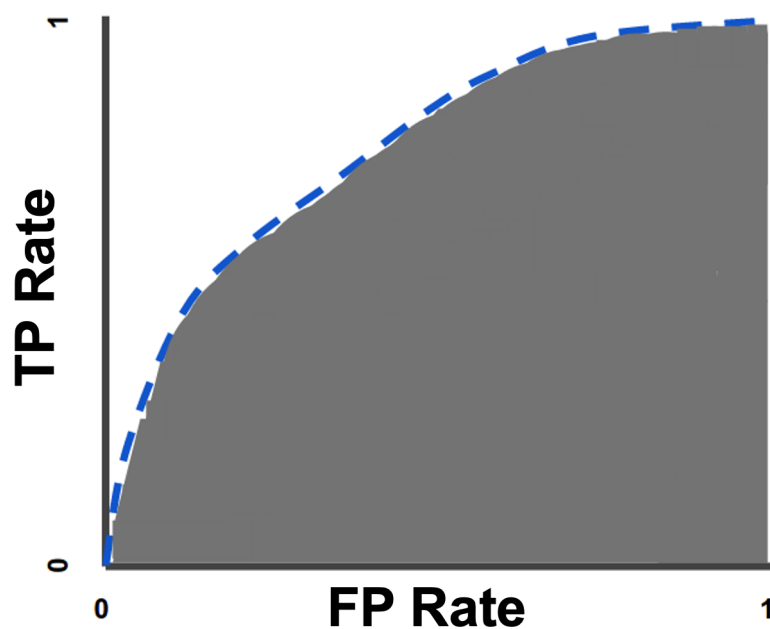
The AUC is 0.5, representing a 50% probability of correctly ranking a random positive and negative example

**Source:** <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>



# AUC – Area Under the ROC Curve

---



- Aggregate measure of performance across all possible classification thresholds.
- Interpretation: probability that the model ranks a random positive example more highly than a random negative example

Not helpful when the cost of false negatives vs. false positives are asymmetric

Source: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>

# Exercise and References

- Google:  
<https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>
  - Take quiz
- Blogs: <https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/>

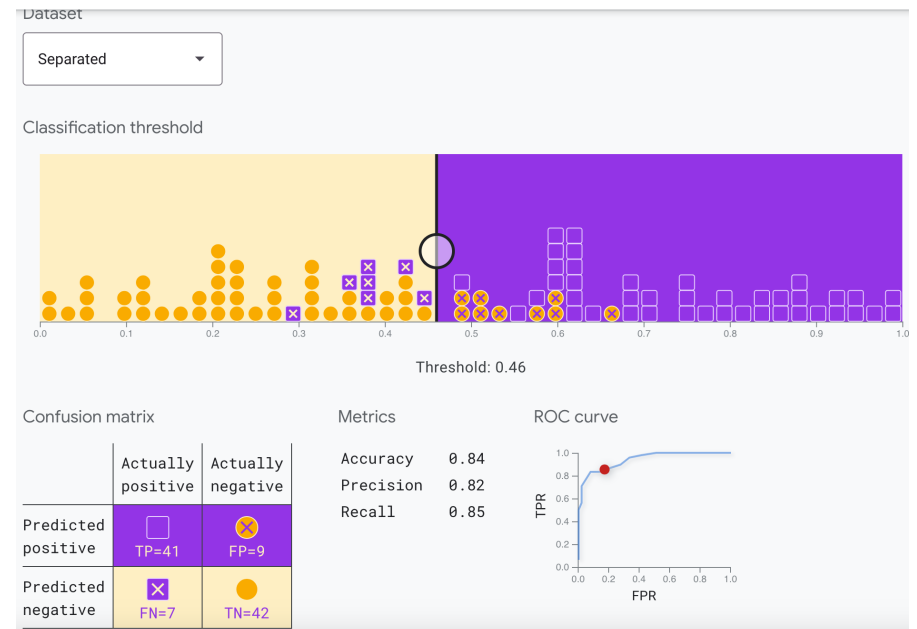


Image credit: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>

# Which ML Classification Method to Choose?

---

- 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>

# Discussion: 10 Tips Paper

---

- Access: <https://biodatamining.biomedcentral.com/articles/10.1186/s13040-017-0155-3>
- Chicco, D. Ten quick tips for machine learning in computational biology. *BioData Mining* **10**, 35 (2017). <https://doi.org/10.1186/s13040-017-0155-3>

# The Tips

- Tip 1: Check and arrange your input dataset properly
- Tip 2: Split your input dataset into three independent subsets (training set, validation set, test set), and use the test set only once you complete training and optimization phases
- Tip 3: Frame your biological problem into the right algorithm category
- Tip 4: Which algorithm should you choose to start? The simplest one!
- Tip 5: Take care of the imbalanced data problem
- Tip 6: Optimize each hyper-parameter
- Tip 7: Minimize overfitting
- Tip 8: Evaluate your algorithm performance with the Matthews correlation coefficient (MCC) or the Precision-Recall curve
- Tip 9: Program your software with open source code and platforms
- Tip 10: Ask for feedback and help to computer science experts, or to collaborative Q&A online communities

# Lecture 7: Quiz 1

---

-

# New Exercise: COMPASS and Recidivism

---

-

# Recall Exercise: German Credit

---

-



# Datasets

---

- UCI Dataset:
  - Weka: <https://www.ics.uci.edu/~mlearn/MLRepository.html> (e.g., download: <https://prdownloads.sourceforge.net/weka/uci-20070111.tar.gz>)
  - Check in UCI – variants:
    - <https://archive.ics.uci.edu/dataset/573/south+german+credit+update>
- Weka
  - Direct link: <https://github.com/Waikato/weka-3.8/blob/master/wekadocs/data/credit-g.arff>
  - As part of development packages
    - like DataHub, <https://datahub.io/machine-learning/credit-g#python>

# German Credit Data

<https://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29>

- Dataset that classifies people's credit risk based on their individual attributes such as Age, Income, Gender, etc.
  - 1000 rows of data, each with 20 attributes to check bias against
- Each entry represents an individual who takes credit from a bank
- Each entry is classified as *Good* or *Bad* credit risk based on their profile

## Example Instance:

A11 6 A34 A43 1169 A65 A75 4 A93 A101 4 A121 67 A143 A152 2  
A173 1 A192 A201 1

1. Credit amount (numerical);
2. Credit duration (numerical);
3. Credit purpose (categorical);
4. Status of existing checking account (categorical);
5. Status of savings accounts and bonds (categorical);
6. Number of existing credits (numerical);
7. Credit history (categorical);
8. Installment plans (categorical);
9. Installment rate (numerical);
10. Property (categorical);
11. Residence (categorical);
12. Period of present residency (numerical);
13. Telephone (binary);
14. Employment (categorical);
15. Employment length (categorical);
16. Personal status and gender (categorical);
17. Age (numerical);
18. Foreign worker (binary);
19. Dependents (numerical);
20. Other debtors (categorical);
21. Credit score (binary)

**Example record:** Alice is requesting a loan amount of 1567 DM for a duration of 12 months for the purpose of purchasing a television, with a positive checking account balance that is smaller than 200 DM, having less than 100 DM in savings account, and having one existing credit at this bank. She duly paid existing credits at the bank till now and has no other installment plan. She possesses a car and owns a house, has been living at the present residence for one year and has a registered telephone. She is a skilled employee, working in the present employment for past four years. She is a 22-year-old married female and is a German citizen. She has one dependent and no guarantors. The recorded outcome for Alice (attribute #21) is a good credit score.

Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml], Irvine, CA: University of California, School of Information and Computer Science

VERMA, S., AND RUBIN, J. 2018. FAIRNESS DEFINITIONS EXPLAINED. IN PROCEEDINGS OF THE INTERNATIONAL WORKSHOP ON SOFTWARE FAIRNESS, FAIRWARE '18, 1–7. NEW YORK, NY, USA: ASSOCIATION FOR COMPUTING MACHINERY, [HTTPS://WWW.ECE.UBC.CA/~MJULIA/PUBLICATIONS/FAIRNESS\\_DEFINITIONS\\_EXPLAINED\\_2018.PDF](https://www.ece.ubc.ca/~mJulia/publications/fairness_definitions_explained_2018.pdf)

# German Credit Data

<https://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29>

- Dataset that classifies people's credit risk based on their individual attributes such as Age, Income, Gender, etc.
  - 1000 rows of data, each with 20 attributes to check bias against
- Each entry represents an individual who takes credit from a bank
- Each entry is classified as *Good* or *Bad* credit risk based on their profile
  - It is **worse** to class a **customer as good when they are bad**, than it is to class a **customer as bad when they are good**.

1. Credit amount (numerical);
2. Credit duration (numerical);
3. Credit purpose (categorical);
4. Status of existing checking account(categorical);
5. Status of savings accounts and bonds (categorical);
6. Number of existing credits (numerical);
7. Credit history(categorical);
8. Installment plans (categorical);
9. Installment rate(numerical);
10. Property (categorical);
11. Residence (categorical);
12. Period of present residency (numerical);
13. Telephone (binary);
14. Employment (categorical);
15. Employment length (categorical);
16. Personal status and gender (categorical); 1
17. Age (numerical);
18. Foreign worker (binary);
19. Dependents (numerical);
20. Other debtors (categorical);
21. Credit score (binary)

Review detailed data exploration at:

<https://www.kaggle.com/sanyalush/predicting-credit-risk>

**Example record:** Alice is requesting a loan amount of 1567 DM for a duration of 12 months for the purpose of purchasing a television, with a positive checking account balance that is smaller than 200 DM, having less than 100 DM in savings account, and having one existing credit at this bank. She duly paid existing credits at the bank till now and has no other installment plan. She possesses a car and owns a house, has been living at the present residence for one year and has a registered telephone. She is a skilled employee, working in the present employment for past four years. She is a 22-year-old married female and is a German citizen. She has one dependent and no guarantors. The recorded outcome for Alice (attribute #21) is a good credit score.

Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml], Irvine, CA: University of California, School of Information and Computer Science

VERMA, S., AND RUBIN, J. 2018. FAIRNESS DEFINITIONS EXPLAINED. IN PROCEEDINGS OF THE INTERNATIONAL WORKSHOP ON SOFTWARE FAIRNESS, FAIRWARE '18, 1–7. NEW YORK, NY, USA: ASSOCIATION FOR COMPUTING MACHINERY, [HTTPS://WWW.ECE.UBC.CA/~MJULIA/PUBLICATIONS/FAIRNESS\\_DEFINITIONS\\_EXPLAINED\\_2018.PDF](https://www.ece.ubc.ca/~m Julia/publications/fairness_definitions_explained_2018.pdf)

# Trust Issues

---

# Context: German Credit Data's Analysis

---

- Review detailed data exploration at:  
<https://www.kaggle.com/sanyalush/predicting-credit-risk>
- Notice issues in lending ?
  - No single female
  - Discrimination by gender, age, ... ?

# Discussion on Reading Material - 1

---

“Biases in AI Systems”, Ramya Srinivasan, Ajay Chander  
Communications of the ACM, August 2021, Vol. 64 No. 8, Pages 44-49  
10.1145/3464903

<https://cacm.acm.org/magazines/2021/8/254310-biases-in-ai-systems/fulltext>

# Taxonomy of Biases

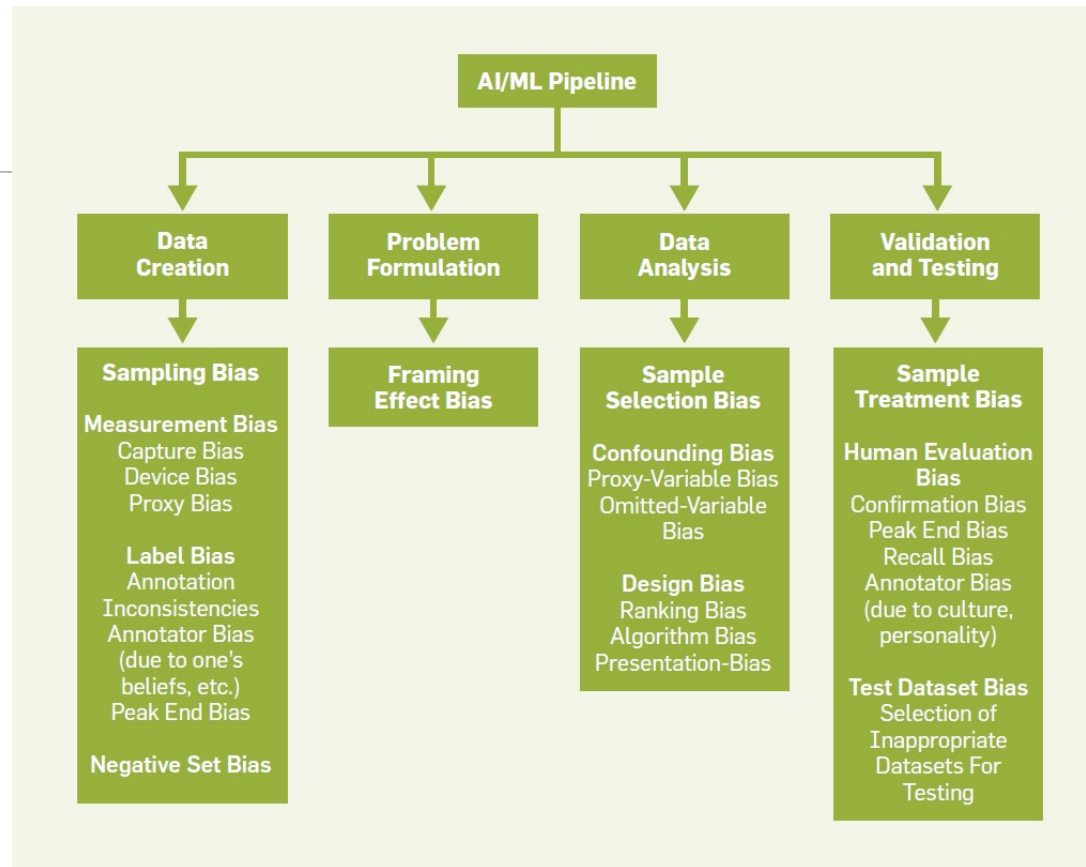


Figure courtesy: "Biases in AI Systems", Ramya Srinivasan, Ajay Chander, CACM 2021

# Data Creation Biases

- Sampling bias

- Selecting specific types of instances more than others
- Examples:
  - Twitter for finding traffic on a road (assumptions: one social media platform, driver v/s riders)
  - FaceTime for recording accidents (assumptions: a company product, having smart phone)
- Measurement bias
  - Proxies used instead of true values (arrests for crime, hospital visits for health)
- Label bias
  - Inconsistent usage of terms, Subjectivity, peak effect – memory based bias from psychology
  - Serious bottleneck for supervised methods
- Negative set bias
  - Not having enough data for negative classification

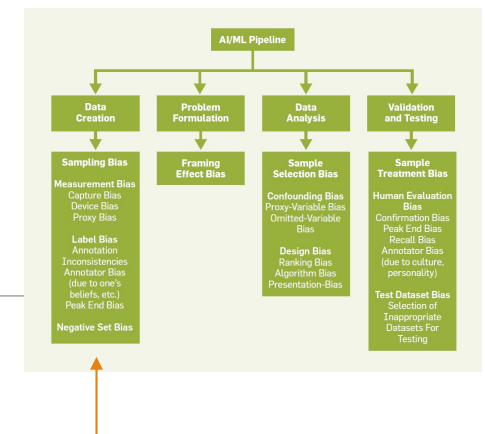
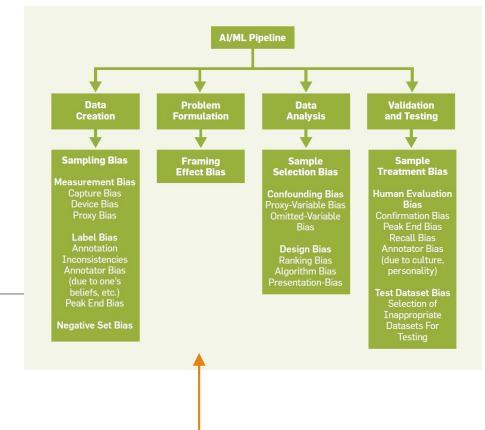


Figure courtesy: "Biases in AI Systems", Ramya Srinivasan, Ajay Chander, CACM 2021



# Problem Formulation Bias

- “What cannot be measured cannot be controlled”



- Framing Effect Bias
  - Compas: predicting recidivism – criminals re-offending
    - Pro-republica claim: biased against black defendants as the group was associated with a higher false-positive rate (**equalized odds** and **equality of opportunity fairness** criteria)
      - **Equalized odds**: if protected and unprotected groups have equal true positive rate and false positive rate
    - Northpointe: scores satisfied fairness from the viewpoint of **predictive rate parity**
      - **Predictive rate parity**: precision, the number of the true positives divided by the total number of examples predicted positive within a group, same within groups

Figure courtesy: “Biases in AI Systems”, Ramya Srinivasan, Ajay Chander, CACM 2021

# Data Analysis Bias

*Due to algorithmic properties or computational limitations*

- Sample Selection Bias
  - “selection of individuals, groups, or data for analysis in such a way that the samples are not representative of the population intended to be analyzed”
  - Example – *“In analyzing the effect of motherhood on wages, if the study is restricted to women who are already employed, then the measured effect will be biased as a result of conditioning on employed women”*
- Confounding Bias
  - Common causes that affect both inputs and outputs, but are not accounted for. Example – *“admissions to a graduate school are based on the person's previous grade point average. There might be other factors, however, such as ability to get coaching, which in turn may be dependent on sensitive attributes such as race; and these factors may determine the grade point average and admission rates”*
  - Proxy variables: there are proxies for protected variables being omitted; indirect bias
  - Omitted variables: model does not consider variables

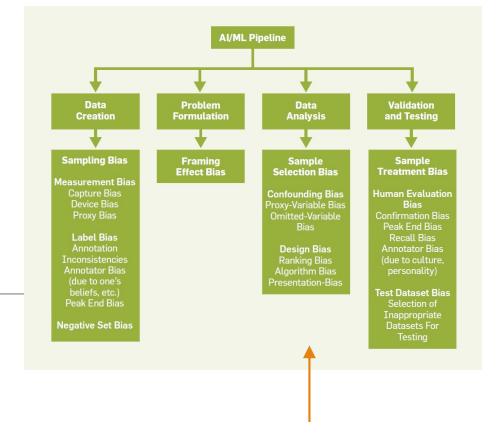


Figure courtesy: “Biases in AI Systems”, Ramya Srinivasan, Ajay Chander, CACM 2021

# Data Analysis Bias

*Due to algorithmic properties or computational limitations*

- Design Bias
  - Ranking Bias: Which top results to be shown? What should be the cut-off?
- Presentation Bias: where and how the result is presented?

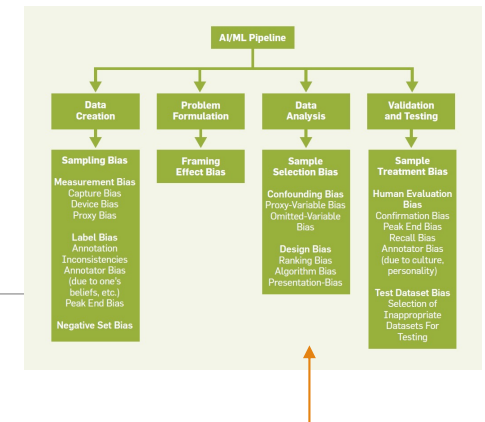


Figure courtesy: "Biases in AI Systems", Ramya Srinivasan, Ajay Chander, CACM 2021

# Validation and Testing Bias

- Human Evaluation Bias
  - Human's biases: confirmation bias, peak end effect, and prior beliefs
- Test Dataset Bias
  - Faces issues similar to (training) dataset selection
  - Further, is it appropriate for application in hand and up-to-date over time?

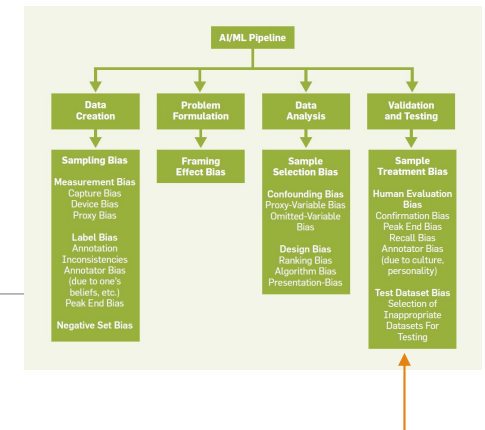


Figure courtesy: "Biases in AI Systems", Ramya Srinivasan, Ajay Chander, CACM 2021

# Guidelines for Developers

---

- Draw a structural diagram illustrating various features of interest and their interdependencies
  - To avoid bias, all features that associated with the target feature of interest is important
- Understand which features of the data are sensitive based on the application
  - Understand the laws: protected by law
  - Best practices that are good to have
- Datasets used for analysis should be representative of the true population under consideration
- Standardize rules for annotating data
- Validate model with representative population, not a subset

# Discussion on Reading Material - 2

---

“Data science and AI in the age of COVID-19 - Reflections on the response of the UK’s data science and AI community to the COVID-19 pandemic”, Alan Turing Institute, 2021

- Approach of report
  - A series of workshops in academic circles in Nov and Dec 2020
  - Discussions and findings distilled into the report

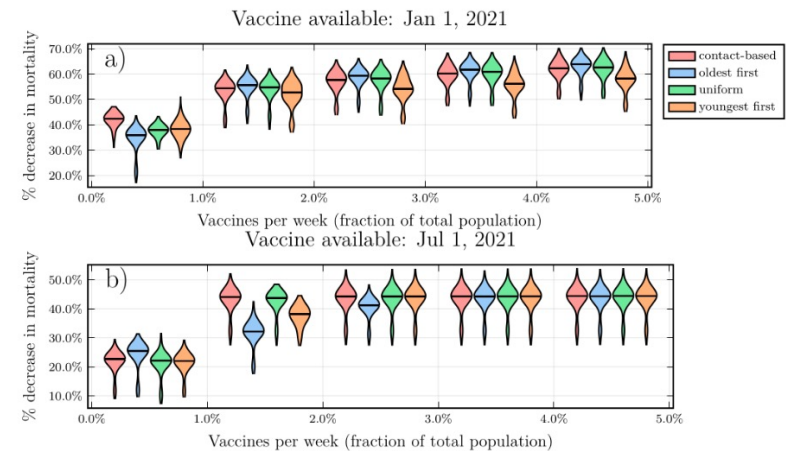
# Example AI Usage: Distribution of Vaccines

---

- **Problem:** Limited supply, larger demand, many technologies, recipient hesitancy; How do distribute equitably, fairly and efficiently
- Possible (automated) solutions
  - Random: pick receiver based on random choice
    - **Benefit:** Easy to implement
    - **Problems:** Equitable but not fair, receiver may not be at risk or not want it, others wanting it may not get it
    - **Question:** assumes we can give vaccine quickly to the selected person
  - Prioritized random: make a prioritized list of groups, assign randomly in each group
    - **Benefit:** identifies affected groups
    - **Problems:** receiver may not want the vaccine
    - **Question:** who comes up with groups?, is it rewarding groups who have not been taking precautions ? Assumes we can give vaccine quickly to the selected person
  - ...
  - Benefit-cost: based on contribution to economy
    - **Benefit:** efficient

# Impact of Decisions in Vaccine Distribution

- Article: [‘The Pandemic Is a Prisoner’s Dilemma Game’](#)
- [Prioritising COVID-19 vaccination in changing social and epidemiological landscapes](#), Sep 2020.
- Choices
  - impact of vaccinating 60+ year-olds first;
  - <20 year-olds first;
  - uniformly by age; and
  - a novel contact-based strategy
- Insights
  - Vaccination reduces median deaths by 32%-77% (22%-63%) for January (July) availability, depending on the scenario.
  - Vaccinating 60+ year-olds first prevents more deaths (up to 8% more) than transmission-interrupting strategies





# Discussion on Reading Material

---

“Data science and AI in the age of COVID-19 - Reflections on the response of the UK’s data science and AI community to the COVID-19 pandemic”, Alan Turing Institute, 2021

- Findings

1. Researchers responded to COVID need with enthusiasm leading to a large number of projects
  1. **Word-wide\* (for context)**: protein to aid disease detection and treatment (**molecular scale**), the analysis of patient data like images and conditions to improve patient care (**clinical scale**) and analysis of cases and social media to predict disease severity, understand mis-information and communicate effectively (**societal scale**).
  2. **UK specific examples**: model disease spread, navigate lockdown
2. Major hurdle was lack of “robust and timely data”, especially access and standardization
  1. Develop protocols for collecting and managing protected data
  2. Develop protocols for generating anonymized and synthetic data

\* Joseph Bullock, Alexandra Luccioni, Katherine Hoffmann Pham, Cynthia Sin Nga Lam, and Miguel Luengo-Oroz. Mapping the landscape of artificial intelligence applications against covid-19. In Journal of Artificial Intelligence Research 69, 807-845, 2020.

# Discussion on Reading Material

---

“Data science and AI in the age of COVID-19 - Reflections on the response of the UK’s data science and AI community to the COVID-19 pandemic”, Alan Turing Institute, 2021

- Findings

3. Concern over inequality and exclusion slowed progress. Inadequate representation and engagement from some groups
4. Challenge in communicating research findings to policy makers and public. Specifically, timeliness, accuracy and clarity.
  - Communication among experts
  - Communication among researchers and policy makers
  - Communication among researchers and public

\* Joseph Bullock, Alexandra Luccioni, Katherine Hoffmann Pham, Cynthia Sin Nga Lam, and Miguel Luengo-Oroz. Mapping the landscape of artificial intelligence applications against covid-19. In Journal of Artificial Intelligence Research 69, 807-845, 2020.

# Project Discussion

---

# Course Project

---

- **Framework**

1. (Problem) Think of a problem whose solution may benefit people (e.g., health, water, air, traffic, safety)
2. (User) Consider how the primary user (e.g., patient, traveler) may be solving the problem today
3. (AI Method) Think of what the solution will do to help the primary user
  1. Solution => ML task (e.g. classification), recommendation, text summarization, ...
  2. Use a foundation model (e.g., LLM-based) solution as the baseline
4. (Data) Explore the data for a solution to work
5. (Reliability: Testing) Think of the evaluation metric we should employ to establish that the solution will work? (e.g., 20% reduction in patient deaths)
6. (Holding Human Values) Discuss if there are fairness/bias, privacy issues?
7. (Human-AI) Finally, elaborate how you will explain the primary user that your solution is trustable to be used by them

# Project Discussion: What to Focus on ?

---

- Problem: you should care about it
- Data: should be available
- Method: you need to be comfortable with it. Have at least two – one serves as baseline
- Trust issue
  - Due to Users
    - Diverse demographics
    - Diverse abilities
    - Multiple human languages
  - Or other impacts
- What one does to mitigate trust issue

# Rubric for Evaluation of Course Project

---

## Project

- Project plan along framework introduced (7 points)
- Challenging nature of project
- Actual achievement
- Report
- Sharing of code

## Presentation

- Motivation
- Coverage of related work
- Results and significance
- Handling of questions

# Project Discussion

1. Create a private Github repository called “CSCE581-Spring2025-<studentname>-Repo”. Share with Instructor (biplav-s)
2. Create a folder called “Project”. Inside, create a text file called “ProjectPlan.md” (or “ProjectPlan.txt”) and have details by the next class (Jan 30, 2025)

1. Title:
2. Key idea: (2-3 lines)
3. Who will care when done:
4. Data need:
5. Methods:
6. Evaluation:
7. Users:
8. Trust issue:

# Concluding Section

---



# Week 4 (L7 and L8): Concluding Comments

---

- We looked at
  - Supervised ML algorithms, ML tools
  - Deep-dive into German credit
- Project descriptions finalized

# About Next Week – Lectures 11, 12

---

# Lectures 11, 12: Trust and Mitigation

---

- ML/ Classification: Trust Issues
- ML/ Classification: Mitigation

5	Jan 28 (Tu)	Common AI methods: ML Landscape
6	Jan 30 (Th)	AI - Structured: Analysis – Supervised ML
7	Feb 4 (Tu)	AI - Structured: Analysis – Supervised ML
8	Feb 6 (Th)	Project discussion (1)
9	Feb 11 (Tu)	Quiz 1
10	Feb 13 (Th)	AI - Structured: Analysis – Supervised ML – Trust Issues
11	Feb 18 (Tu)	AI - Structured: Analysis – Supervised ML – Trust Issues
12	Feb 20 (Th)	AI - Structured: Analysis – Supervised ML – Mitigation Methods