INFORMATICS PROFESSIONALS. LEADING THE WAY.

4D-3: Data Analytics 3 Artificial Intelligence and Machine Learning

Alexis B. Carter, MD

Children's Healthcare of Atlanta

Clinical Informatics Board Review Course

Clinical Informatics Subspecialty Delineation of Practice (CIS DoP)

Domain 1: Fundamental Knowledge and Skills (no Tasks are associated with this Domain which is focused on fundamental knowledge and skills)

Clinical Informatics

K001. The discipline of informatics (e.g., definitions, history, careers, professional organizations)
K002. Fundamental informatics concepts, models, and theories
K003. Core clinical informatics literature (e.g., foundational literature, principle journals, critical analysis of literature, use of evidence to inform practice)
K004. Descriptive and inferential statistics
K005. Health Information Technology (HIT) principles and science
K006. Computer programming fundamentals and computational thinking
K007. Basic systems and network architectures
K008. Basic database structure, data retrieval and analytics techniques and tools
K009. Development and use of interoperability/exchange standards (e.g., Fast Health Interoperability Resources [FHIR], Digital Imaging and Communications in Medicine [DICOM])
K010. Development and use of transaction standards (e.g., American National Standards Institute X12)
K011. Development and use of messaging standards (e.g., Health Level Seven [HL7] v2)
K012. Development and use of ancillary data standards (e.g., imaging and Laboratory Information System[LIS])
K013. Development and use of data model standards
K014. Vocabularies, terminologies, and nomenclatures (e.g., Logical Observation Identifiers Names and Codes [LOINC], Systematized Nomenclature of Medicine --Clinical Terms [SNOMED-CT], RxNorm, International Classification Of Diseases[ICD], Current Procedural Terminology [CPT])
K015. Data taxonomies and ontologies
K016. Security, privacy, and confidentiality requirements and practices
K017. Legal and regulatory issues related to clinical data and information sharing
K018. Technical and non-technical approaches and barriers to interoperability
K019. Ethics and professionalism

The Health System

K020. Primary domains of health, organizational structures, cultures, and processes (e.g., health care delivery, public health, personal health, population health, education of health professionals, clinical research)
K021. Determinants of individual and population health
K022. Forces shaping health care delivery and considerations regarding health care access
K023. Health economics and financing
K024. Policy and regulatory frameworks related to the healthcare system
K025. The flow of data, information, and knowledge within the health system

Domain 2: Improving Care Delivery and Outcomes

K026. Decision science (e.g., Bayes theorem, decision analysis, probability theory, utility and preference assessment, test characteristics)
K027. Clinical decision support standards and processes for development, implementation, evaluation, and maintenance
K028. Five Rights of clinical decision support (i.e., information, person, intervention formats, channel, and point/time in workflow)
K029. Legal, regulatory, and ethical issues regarding clinical decision support
K030. Methods of workflow analysis
K031. Principles of workflow re-engineering
K032. Quality improvement principles and practices (e.g., Six Sigma, Lean, Plan-Do-Study-Act [PDSA] cycle, root cause analysis)
K033. User-centered design principles (e.g., iterative design process)
K034. Usability testing
K035. Definitions of measures (e.g., quality performance, regulatory, pay for performance, public health surveillance)
K036. Measure development and evaluation processes and criteria
K037. Key performance indicators (KPIs)
K038. Claims analytics and benchmarks
K039. Predictive analytic techniques, indications, and limitations
K040. Clinical and financial benchmarking sources (e.g., Gartner, Healthcare Information and Management Systems Society [HIMSS] Analytics, Centers for Medicare and Medicaid Services [CMS], Leapfrog)
K041. Quality standards and measures promulgated by quality organizations (e.g., National Quality Forum [NQF], Centers for Medicare and Medicaid Services [CMS], National Committee for Quality Assurance [NCQA])
K042. Facility accreditation quality and safety standards (e.g., The Joint Commission, Clinical Laboratory Improvement Amendments [CLIA])
K043. Clinical quality standards (e.g., Physician Quality Reporting System [PQRS], Agency for Healthcare Research and Quality [AHRQ], National Surgical Quality Improvement Program [NSQIP], Quality Reporting Document Architecture [QRDA], Health Quality Measure Format [HQMF], Council on Quality and Leadership [CQL], Fast Health Interoperability Resources [FHIR] Clinical Reasoning)
K044. Reporting requirements
K045. Methods to measure and report organizational performance
K046. Adoption metrics (e.g., Electronic Medical Records Adoption Model [EMRAM], Adoption Model for Analytics Maturity [AMAM])
K047. Social determinants of health
K048. Use of patient-generated data
K049. Prediction models
K050. Risk stratification and adjustment
K051. Concepts and tools for care coordination
K052. Care delivery and payment models

Domain 3: Enterprise Information Systems

K053. Health information technology landscape (e.g., innovation strategies, emerging technologies)
K054. Institutional governance of clinical information systems
K055. Information system maintenance requirements
K056. Information needs analysis and information system selection
K057. Information system implementation procedures
K058. Information system evaluation techniques and methods
K059. Information system and integration testing techniques and methodologies
K060. Enterprise architecture (databases, storage, application, interface engine)
K061. Methods of communication between various software components
K062. Network communications infrastructure and protocols between information systems (e.g., Transmission Control Protocol/Internet Protocol [TCP/IP], switches, routers)
K063. Types of settings (e.g., labs, ambulatory, radiology, home) where various systems are used
K064. Clinical system functional requirements
K065. Models and theories of human-computer (machine) interaction (HCI)
K066. HCI evaluation, usability engineering and testing, study design and methods
K067. HCI design standards and design principles
K068. Functionalities of clinical information systems (e.g., Electronic Health Records [EHR], Laboratory Information System [LIS], Picture Archiving and Communication System [PACS], Radiology Information System [RIS] vendor-neutral archive, pharmacy, revenue cycle)
K069. Consumer-facing health informatics applications (e.g., patient portals, mobile health apps and devices, disease management, patient education, behavior modification)
K070. User types and roles, institutional policy and access control
K071. Clinical communication channels and best practices for use (e.g., secure messaging, closed loop communication)
K072. Security threat assessment methods and mitigation strategies
K073. Security standards and safeguards
K074. Clinical impact of scheduled and unscheduled system downtimes
K075. Information system failure modes and downtime mitigation strategies (e.g., replicated data centers, log shipping)
K076. Approaches to knowledge repositories and their implementation and maintenance
K077. Data storage options and their implications
K078. Clinical registries
K079. Health information exchanges
K080. Patient matching strategies
K081. Master patient index
K082. Data reconciliation
K083. Regulated medical devices (e.g., pumps, telemetry monitors) that may be integrated into information systems
K084. Non-regulated medical devices (e.g., consumer devices)
K085. Telehealth workflows and resources (e.g., software, hardware, staff)

Domain 4: Data Governance and Data Analytics

K086. Stewardship of data
K087. Regulations, organizations, and best practice related to data access and sharing agreements, data use, privacy, security, and portability
K088. Metadata and data dictionaries
K089. Data life cycle
K090. Transactional and reporting/research databases
K091. Techniques for the storage of disparate data types
K092. Techniques to extract, transform, and load data
K093. Data associated with workflow processes and clinical context
K094. Data management and validation techniques
K095. Standards related to storage and retrieval from specialized and emerging data sources
K096. Types and uses of specialized and emerging data sources (e.g., imaging, bioinformatics, internet of things (IoT), patient-generated, social determinants)
K097. Issues related to integrating emerging data sources into business and clinical decision making
K098. Information architecture
K099. Query tools and techniques
K100. Flat files, relational and non-relational/NoSQL database structures, distributed file systems
K101. Definitions and appropriate use of descriptive, diagnostic, predictive, and prescriptive analytics
K102. Analytic tools and techniques (e.g., Boolean, Bayesian, statistical/mathematical modeling)
K103. Advanced modeling and algorithms
K104. Artificial intelligence
K105. Machine learning (e.g., neural networks, support vector machines, Bayesian network)
K106. Data visualization (e.g., graphical, geospatial, 3D modeling, dashboards, heat maps)
K107. Natural language processing
K108. Precision medicine (customized treatment plans based on patient-specific data)
K109. Knowledge management and archiving science
K110. Methods for knowledge persistence and sharing
K111. Methods and standards for data sharing across systems (e.g., health information exchanges, public health reporting)

Domain 5: Leadership and Professionalism

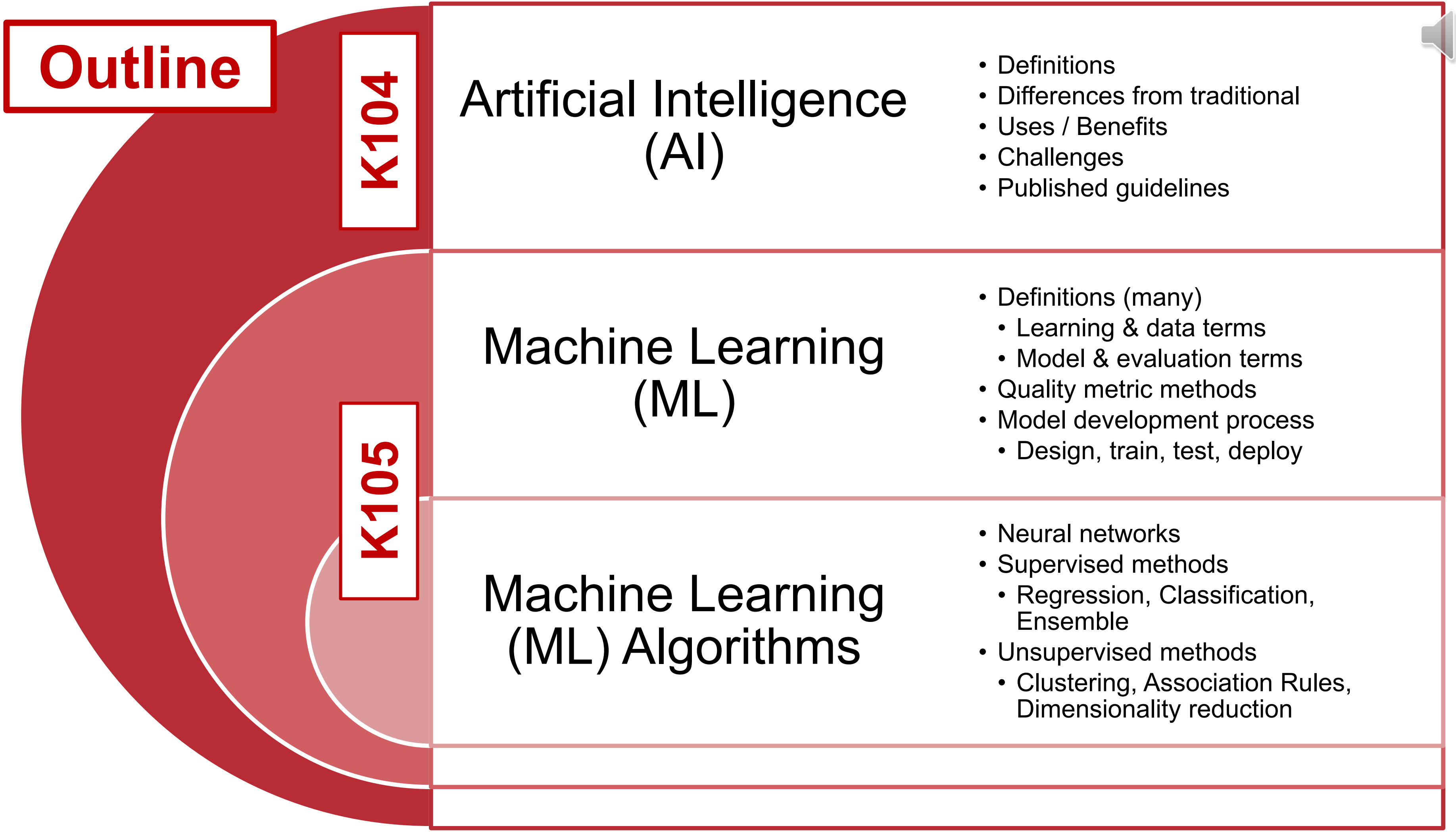
K112. Environmental scanning and assessment methods and techniques
K113. Consensus building, collaboration, and conflict management
K114. Business plan development for informatics projects and activities (e.g., return on investment, business case analysis, pro forma projections)
K115. Basic revenue cycle
K116. Basic managerial/cost accounting principles and concepts
K117. Capital and operating budgeting
K118. Strategy formulation and evaluation
K119. Approaches to establishing Health Information Technology (HIT) mission and objectives
K120. Communication strategies, including one-on-one, presentation to groups, and asynchronous communication
K121. Effective communication programs to support and sustain systems implementation
K122. Writing effectively for various audiences and goals
K123. Negotiation strategies, methods, and techniques
K124. Conflict management strategies, methods, and techniques
K125. Change management principles, models, and methods
K126. Assessment of organizational culture and behavior change theories
K127. Theory and methods for promoting the adoption and effective use of clinical information systems
K128. Motivational strategies, methods, and techniques
K129. Basic principles and practices of project management
K130. Project management tools and techniques
K131. Leadership principles, models, and methods
K132. Intergenerational communication techniques
K133. Coaching, mentoring, championing and cheerleading methods
K134. Adult learning theories, methods, and techniques
K135. Teaching modalities for individuals and groups
K136. Methods to assess the effectiveness of training and competency development
K137. Principles, models, and methods for building and managing effective interdisciplinary teams
K138. Team productivity and effectiveness (e.g., articulating team goals, defining rules of operation, clarifying individual roles, team management, identifying and addressing challenges)
K139. Group management processes (e.g., nominal group, consensus mapping, Delphi method)

Knowledge Statements from the DoP

Artificial Intelligence and Machine Learning

- K104. Artificial intelligence
- K105. Machine learning
  - (e.g., neural networks, support vector machines, Bayesian network)
  - Will cover machine learning versions of…
    - K039. Predictive analytic techniques, indications, and limitations
    - K049. Prediction models

Outline



K104

Artificial Intelligence (AI)

- Definitions
- Differences from traditional
- Uses / Benefits
- Challenges
- Published guidelines

K105

Machine Learning (ML)

- Definitions (many)
  - Learning & data terms
  - Model & evaluation terms
- Quality metric methods
- Model development process
  - Design, train, test, deploy

Machine Learning (ML) Algorithms

- Neural networks
- Supervised methods
  - Regression, Classification, Ensemble
- Unsupervised methods
  - Clustering, Association Rules, Dimensionality reduction

K104. Artificial Intelligence

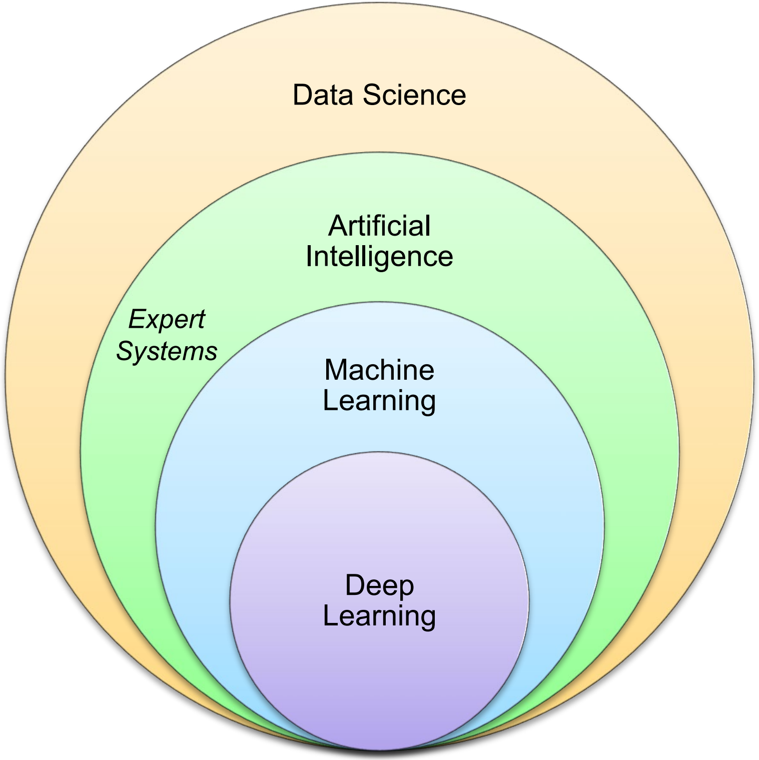
Artificial Intelligence

Definitions

Clinical Informatics
Board Review Course

Definitions

- **Data Science:** Science of organizing / analyzing massive amounts of data (In pathology = computational pathology)
- **Artificial intelligence (AI)**: ability of a computer or computer-controlled robot to perform tasks commonly associated with intelligent beings
    - https://www.britannica.com/technology/artificial-intelligence
- **Expert Systems:** Human knowledge encoded in a knowledgebase
- **Machine Learning (ML):** Algorithms which allow computers to learn with**out** explicit programming
- **Deep Learning**: Specific set of ML tools designed to handle big data (e.g., specific neural networks)



Definitions

- **Narrow AI***
  - The machine can perform a **single** specific task better than a human
- **General AI**
  - The machine can perform **any intellectual task** with the **same** accuracy as a human
- **Strong AI**
  - The machine **outperforms** humans in **many** tasks

* All current AI tools, if successful, are only narrow AI.

- "**AI Effect**" and "**Tesler's theorem**"
  - AI is whatever hasn't been done yet
  - Optical character and voice recognition, automated pap smear and peripheral blood smear readers, bioinformatics pipelines → no longer considered AI
- **Autonomous intelligence**
  - AI is making the decisions (no "human-in-the-loop")
- **Augmented intelligence**
  - AI is used to augment and/or assist humans in their work
  - Maintains "human-in-the-loop"; human ultimately making decisions

Expert Systems vs. Machine Learning

Expert Systems

- Rules, relationships, ontologies explicitly coded or programmed into a knowledgebase
  - Rules engines of expressly programmed IF-THEN statements (e.g., MYCIN, Internist-I, CADUCEUS)
- Handles limited amounts of data compared to machine learning

Machine Learning

- Not based on human knowledgebases or specified rules
- Uses algorithms to learn repetitive data patterns
  - Can discover new patterns, make predictions
- Handles large data sets in complex settings

AMIA
INFORMATICS PROFESSIONALS. LEADING THE WAY.

Artificial Intelligence

Differences

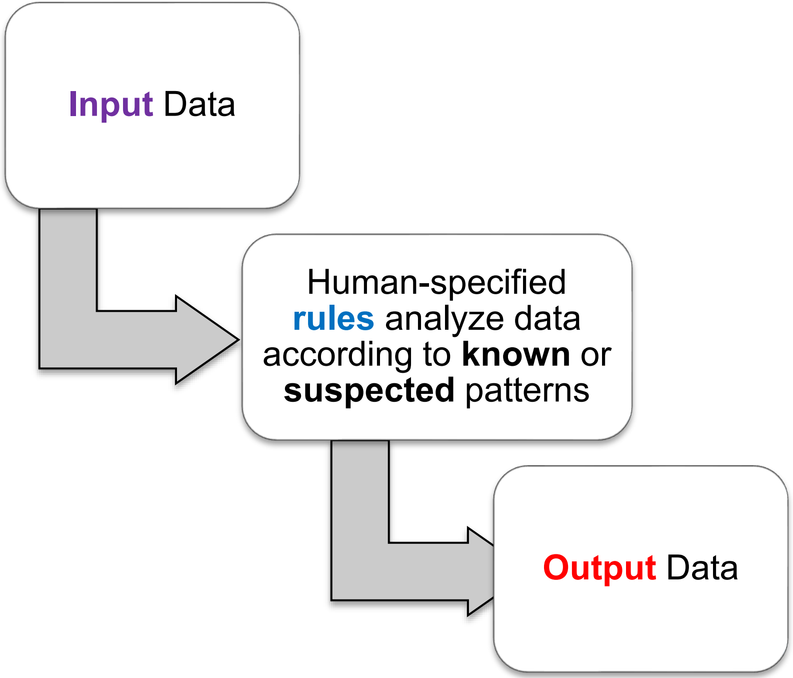
Clinical Informatics
Board Review Course

Machine Learning vs. Traditional Statistics

Function	**Traditional Statistics**	**Machine Learning**
Defines explicit mathematical relationship between inputs and outputs	Yes	Not usually
Makes assumptions about characteristics and distribution of the data fed to it •Parametric vs. Non-parametric •Normal distribution vs. Non-normal distribution	Yes	Not usually
Handles large # input variables	Not usually	Yes
Can use complex multifactorial data	Not usually	Yes
Reason for output is clear and explainable	Yes	Not usually (**black box problem**)

Machine Learning vs. Traditional Programming

Traditional Programming

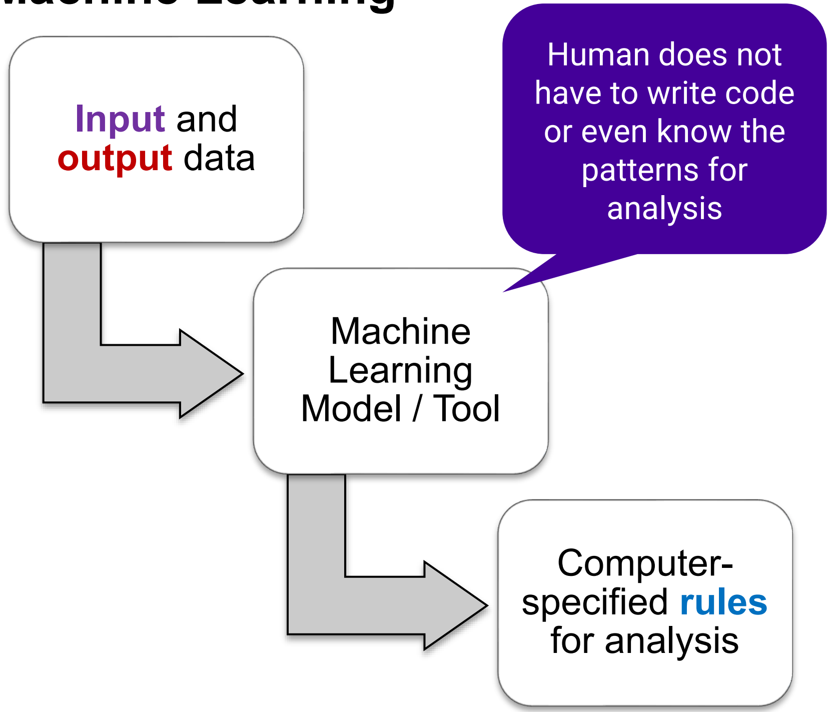


Input Data

Human-specified **rules** analyze data according to **known** or **suspected** patterns

Output Data

Machine Learning



Input and **output** data

Machine Learning Model / Tool

Computer-specified **rules** for analysis

Human does not have to write code or even know the patterns for analysis

Artificial Intelligence

Uses and Benefits

Clinical Informatics
Board Review Course

Uses and Benefits of AI

- Predictions
  - Medical diagnoses and problems (active research area)
  - Predicting patient volumes → adjust staffing (especially during system changes)
  - Predicting optimal future state workflows / functional gaps in process redesign
  - Predicting, detecting and subverting malware attacks
- Classifications
  - Pattern detection (e.g., diagnoses), feature detection (images)
- Decision support
  - Making prior authorization decisions
- Signal conversion
  - E.g., natural language processing, voice recognition, optical character recognition
- Problem-solving
- Anomaly detection
  - Detecting errors in data (e.g., pathology reports…Ye JJ, Tan MR*, J Pathol Inform*, 2019; 10:20)
- Assistance with time-consuming tasks
  - Counting mitoses
  - Medical documentation
  - Prior authorization

Artificial Intelligence

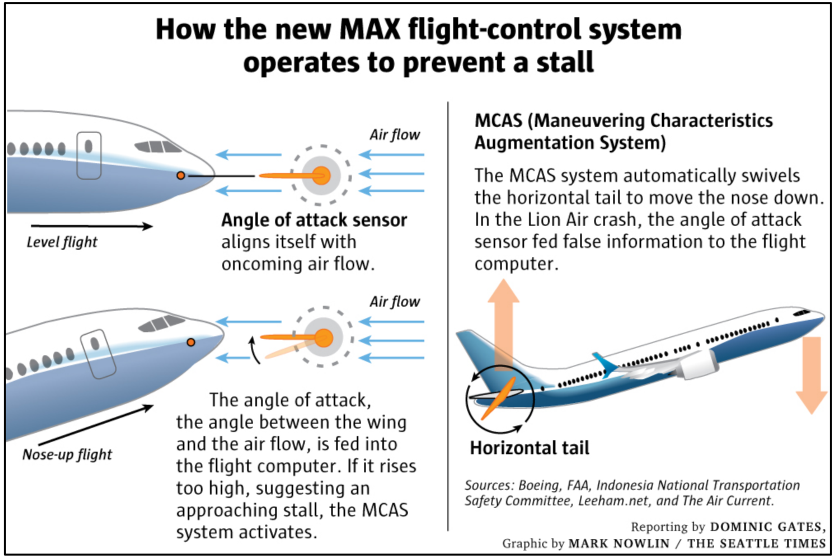
Challenges

AI Challenges

- As with any new technology, there are a lot of challenges
- Challenges similar to other non-AI software
  - Cybersecurity risks
  - Software can be developed with bad data or bad science
  - **Automation bias** – assumption that the computer is right, even when it doesn't make sense [Goddard et al 2012]
  - Inaccurate assumptions about data accuracy and representation
    - IBM Watson, PubMed and Genomics

Artificial Intelligence – Story of Harm

- Boeing 737 MAX flight control system
  - Two plane crashes killing all 346 passengers in Oct 2018, Mar 2019
  - **Faulty** angle-of-attack **sensors** fed bad data to system
  - **No redundant sensors** required to detect when sensor was faulty
  - **No usable human override mechanism**
  - **Default configuration did not show alerts** for mismatched sensor data (when >1 sensor present)
  - **System was not set to disengage** when multiple errors generated at once
  - **Similar errors during simulations not reported to FAA** by Boeing because they were considered “advisory” rather than “critical”
  - FAA, citing lack of funding and resources, over the years had delegated increasing authority to Boeing to assess its own work during certification processes



How the new MAX flight-control system operates to prevent a stall

Angle of attack sensor aligns itself with oncoming air flow.

The angle of attack, the angle between the wing and the air flow, is fed into the flight computer. If it rises too high, suggesting an approaching stall, the MCAS system activates.

MCAS (Maneuvering Characteristics Augmentation System)

The MCAS system automatically swivels the horizontal tail to move the nose down. In the Lion Air crash, the angle of attack sensor fed false information to the flight computer.

Sources: Boeing, FAA, Indonesia National Transportation Safety Committee, Leeham.net, and The Air Current.

Reporting by DOMINIC GATES, Graphic by MARK NOWLIN / THE SEATTLE TIMES

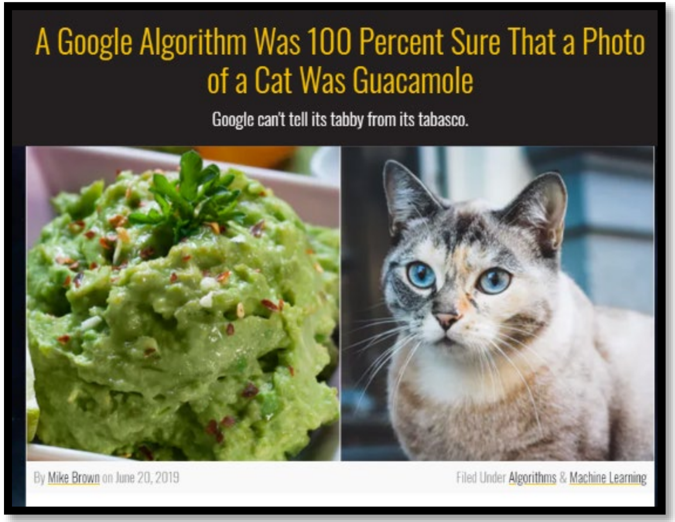
- Image from: [MAX-737-sensor-W - ARFFWG | ARFF Working Group]
- [Washington Post story]
- [FAA Summary]

AI Challenges – Data Quality

- Good quality data is **critical**
  - bad data → bad model
  - Some models need large amount of training data
- Data have insufficient quantity / variability for context
  - Especially problematic for models finding less common patterns (e.g., disease screening, anomaly detection)
  - Underrepresented populations → non-generalizable rules (socioeconomic, gender, race, ethnic and other disparities)
- Data labels represent human bias / false beliefs
  - e.g., court sentences, hiring / firing decisions
  - Can promulgate or exacerbate inequality
- Data have incomplete, inaccurate and/or variable labels
  - Different terms or metrics for same label due to human inconsistency
- Critical input data may be missing
  - **Polanyi's Paradox**:
    - Human decision-making beyond explicit understanding or description
  - Human may not realize which data contributed to human decision
  - Critical inputs may not be represented in AI training data

AI Challenges – Model Problems

- Models can be brittle (unstable)
  - Do not produce consistent output when given similar inputs
  - Unable to see the forest for the trees (double-edged sword)
    - Purpose of models is to analyze details too complex for humans to synthesize, but…
    - Humans better able to factor in general features and/or situational awareness
      - Humans will ignore details in favor of general assessment
      - e.g., cat image classified as guacamole by Google AI when a few critical pixels were changed which were not noticeable to the human eye
  - Small changes to data introduced by hackers (**adversarial examples**) → wrong output [Nature 2019]



A Google Algorithm Was 100 Percent Sure That a Photo of a Cat Was Guacamole

Google can't tell its tabby from its tabasco.

By Mike Brown on June 20, 2019

Filed Under Algorithms & Machine Learning

https://www.inverse.com/article/56914-a-google-algorithm-was-100-percent-sure-that-a-photo-of-a-cat-was-guacamole

AI Challenges – Transparency

- Definitions (multiple)
  - For AI developers: Reasons for model's performance are known and understood
  - For end-users (ethics): Sufficient information is published such that model's performance can be audited [World Health Organization 2021]
- Lack of transparency (**Black box problem**)
  - Rules developed by the AI algorithm
    - May be indecipherable after model is trained, even to the developer(s)
    - May not be able to determine why algorithm generated certain output
    - May generally work well but some output may be inexplicably wrong

AI Challenges – Ethics – Beneficence

- Hot topic because of some noted failures [bias, Wage decisions, Criminal justice decisions]
- 3 main categories: Beneficence, Intelligibility and Accountability
- **Beneficence:** Maximize benefits; minimize risks and harms
  - AI can propagate and exacerbate human bias because…
    - Human bias can infiltrate data used to build models
    - Mitigate through Diversity, Equity and Inclusiveness practices
      - Fair representation of all groups in data
      - Fair and equitable benefits, risks and harms across each group
    - Monitor for and mitigate bias
  - Person should be able to choose whether his/her data used in algorithm
  - Protect human autonomy in decisions (**“human-in-the-loop”**)
    - ACR and RSNA recommendation → do not approve autonomous AI until sufficient human-supervised AI experience obtained
  - Research integrity (good science)
  - Sustainability: Environmental, Workplace (ease of maintenance)

AI Challenges – Ethics – Intelligibility

- **Intelligibility** achieved through **transparency** and **explainability**
- **Transparency** [World Health Organization 2021]
  - Sufficient information **published** before the design or deployment of an AI technology
    - Describes how technology is designed, intended use, data used, etc.
  - Also means that a person knows when AI is being used on them
- **eXplainability** (XAI)
  - Providing the human user an explanation of how the AI tool works

XAI principle [NIST 2020]	Description
Explanation	AI delivers **evidence** or **reasons** for all outputs • **User benefits** • **Societal acceptance -** designed to generate trust and acceptance • **Regulatory and compliance** • **System development -** assists with developing, debugging, improving or maintaining the system • **Owner benefit -** benefits to system operators
Meaningful	Explanations are **understandable** to end-users
Explanation accuracy	Explanation **correctly** reflects system's process for generating output
Knowledge limits	System ONLY operates under conditions for which it was designed or when system reaches sufficient confidence for output

AI Challenges – Ethics

Auditability

- Monitor tool for performance and to ensure ethics are followed
- Formal oversight mechanisms
- **Responsibility**
  - Person(s)/entity(ies) responsible for monitoring AI
- **Responsiveness**
  - Developers and users systematically examine to determine whether it is responding adequately, appropriate and according to expectations AND respond when it is not working (fix or terminate AI program)

Accountability

- Person(s)/entity(ies) accountable when something goes wrong with AI
- Can be personal, organizational or regulatory
- Medicolegal liability
  - AI is not standard of care
  - Regulations not yet developed in US
  - EU paper that discusses that liability is based on physician using standard of care

AI Challenges - Cybersecurity

- AI can be hacked just like any other software
  - Robotic surgical systems (https://www.ncbi.nlm.nih.gov/pubmed/30397993)
- Hacked systems have potential for unauthorized disclosure, patient harm
- Human autonomy (“human-in-the-loop”) may help detect malfunctioning AI

- US national efforts for AI cybersecurity
  - National Security Commission on Artificial Intelligence (NSCAI)
    - Established 2018 by John S. McCain National Defense Authorization Act (Public Law 115-232)

AI Challenges – Legal and Regulatory

- Regulations surrounding AI still in development
  - [Allen 2019, Hernandez-Boussard et al 2021, O'Sullivan et al 2019]
- FDA white paper on AI and machine learning in **Software as a Medical Device** (SaMD)
  - FDA approved first AI device for a very limited use case in April 2018
- CLIA (laboratory testing) → algorithms must be static prior to validation and use
- Medicolegal accountability for decisions made by AI system is still uncertain

Levels of AI

Levels of AI for Society of Automotive Engineers (SAE): https://blog.ansi.org/2018/09/sae-levels-driving-automation-j-3016-2018/#gref
Radiology version: https://www.carjonline.org/article/S0846-5371(19)30006-3/fulltext

Levels	Society of Automotive Engineers (SAE)	Radiology Version*
0	**No driving automation** The performance by the driver of the entire DDT. Basically, systems under this level are found in conventional automobiles.	**No automation** Interpretation / Intervention is done solely by the radiologist **Liability: Radiologist / Clinician**
1	**Driver Assistance** Sustained and ODD-specific execution of either the lateral or the longitudinal vehicle motion control subtask of the DDT. Does not include the execution of these subtasks simultaneously. Expected that the driver performs the remainder of the DDT.	**Physician Assistance** Interpretation/intervention is done primarily by the radiologist with AI providing secondary oversight (ie, existing CAD software for mammography and lung nodules, worklist prioritization). **Liability: Radiologist / Clinician**
2	**Partial Driving Automation** Similar to Level 1, but characterized by both the lateral and longitudinal vehicle motion control subtasks of the DDT. Expectation that the driver completes the object and event detection and response (OEDR) subtask and supervises the driving automation system.	**Partial automation** Interpretation/intervention is done primarily by the AI with radiologist providing secondary oversight (ie, bone age prediction, chest x-ray pathology detection and report pre-population). **Liability: Radiologist / Clinician**
3	**Conditional Driving Automation** The sustained and ODD-specific performance by an ADS of the entire DDT, with the expectation that the human driver will be ready to respond to a request to intervene when issued by the ADS.	**Conditional Automation** Interpretation/intervention is done solely by the AI for a specific indication with the expectation that radiologist will intervene if the results are positive or indeterminate (ie, automated triaging of normal cases where radiologist is expected to intervene if positive but not if negative). **Liability: Artificial Intelligence AND Radiologist / Clinician**
4	**High Driving Automation** Sustained and ODD-specific ADS performance of the entire DDT is carried out without any expectation that a user will respond to a request to intervene.	**High Automation** Interpretation/intervention is done solely by the AI for a specific indication without the expectation that radiologist will intervene. AI is able to arrive at a differential diagnosis and recommend management autonomously (ie, AI analyzes thyroid ultrasound and recommends and/or performs biopsy for a nodule). **Liability: Artificial Intelligence**
5	**Full Driving Automation** Sustained and unconditional performance by an ADS of the entire DDT without any expectation that a user will respond to a request to intervene. Please note that this performance, since it has no conditions to function, is not ODD-specific.	**Full automation** Interpretation/intervention is done solely by the AI for all indications expected of radiologists. AI is able to arrive at a differential diagnosis and recommend management autonomously (ie, chest x-ray requisition states “r/o pneumonia,” AI reports bone tumor with differential diagnosis and recommendations for further imaging/consultation). **Liability: Artificial Intelligence**

AI Challenges

Personnel

- Medicine lacks sufficient data scientists
- Many data scientists lack expertise in medicine and/or healthcare environment

Organizational

- Lack AI strategies
  - Right tasks
  - Right data
  - Right evidence standard(s)
  - Right approaches for integration
- Deploying models in clinical environments is challenging (patient safety, population differences between locations)

Financial

- Lack of reimbursement mechanisms
- Harder to define returns on investment

Technical

- Lack of adequate computational infrastructure
- Introduces new cybersecurity threats that aren’t yet addressed

Artificial Intelligence

Guidelines

Clinical Informatics
Board Review Course

AI Guidelines

- Guideline for machine learning model development (US, Canada, UK Guideline – Oct 2021)
  - Multidisciplinary expertise throughout
  - Good software/security practices
  - Data representative of intended patient population
  - Training data independent of testing data
  - Reference data is well characterized
  - Model design tailored to available data and reflects intended use
  - Focus on keeping the human in the loop (human AI team)
  - Testing demonstrates performance during clinically relevant conditions
  - Users provided clear essential information for use
  - Deployed models are monitored for performance in the real world
- AI Ethics Guidelines and White Papers
  - WHO Ethics Guidelines for AI World Health Organization 2021
  - UNESCO Report of the Social and Human Sciences Commission (SHS) - UNESCO Digital Library
  - EU guidelines Ethics guidelines for trustworthy AI | Shaping Europe’s digital future (europa.eu)
  - Artificial Intelligence Ethics Framework for the Intelligence Community

K105. Machine learning (e.g., neural networks, support vector machines, Bayesian network)

K039. Predictive Analytic Techniques, indications and Limitations

K049. Prediction Models

Machine Learning

Definitions

Clinical Informatics
Board Review Course

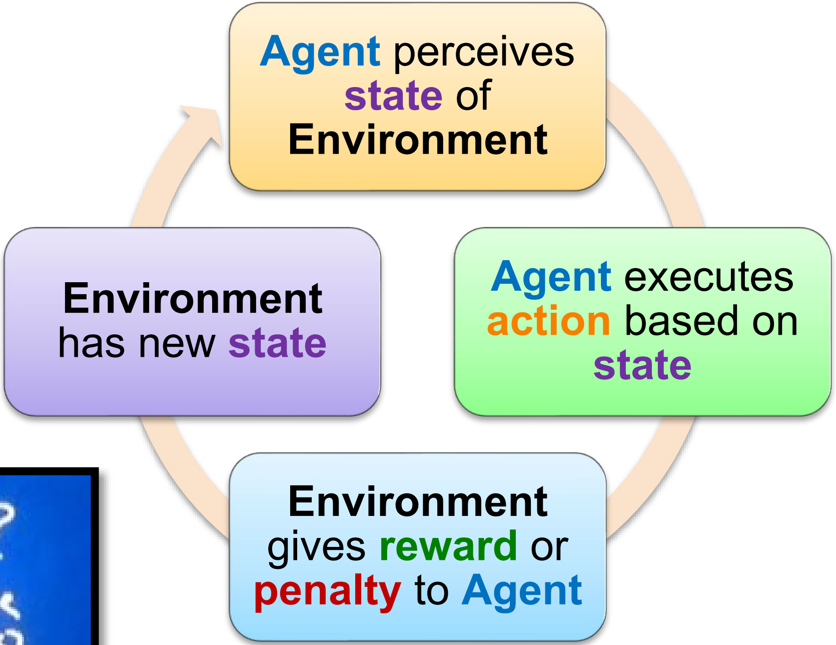
ML Definitions – Types of Learning

Supervised learning	Trains on classified and/or labeled data • Goal → train model to generate **known** answers, patterns or relationships
Fully supervised	All data labeled to same extent (degree of detail)
Semi-supervised	Some data are labeled while other data are not • Unlabeled data may be auto-labeled to match patterns on labeled data
Weakly supervised	Small amount of data have detailed labels; rest of data have fewer labels
Unsupervised learning	Data which have **not** been classified or labeled • Goal → model discovers **new** (previously **unknown**) patterns or relationships

More on these later…

ML Definitions – Types of Learning

- **Reinforcement learning**
  - Used to learn how to reach a (complex) goal
    - Game playing (IBM Watson and Jeopardy)
    - Speech to text, financial trading



Agent perceives **state** of **Environment**

Agent executes **action** based on **state**

Environment gives **reward** or **penalty** to **Agent**

Environment has new **state**

ML Definitions – Types of Learning

- **Reinforcement Learning** (cont.)
  - Goal → learn **policy** (value function) through trial-and-error optimizing long-term reward
    - **Policy:** function which outputs an optimal action to maximize the expected average reward
    - **Value:** future reward received by taking an action in a particular state
  - Different from supervised and unsupervised learning
    - Uses unlabeled data like unsupervised learning but…
    - Uses outcomes to affect model training like supervised learning
  - **Markov Decision Process**
    - Use with a **known** model where states, actions, probabilities of actions generating states, rewards and penalties are evident
    - **Q-learning**: finds optimal action-selection policy for any given finite Markov Decision Process
  - **Monte Carlo method**
    - Use when one or more of the elements are **unknown** (e.g., probabilities of an action → state)
      - Runs through the process many times; each state (s_1, s_2, …, s_n) is reached many times
      - Average outcomes from all previous experiences in a given state
      - Probabilities of action → state are calculated

ML Definitions – Types of Learning

- **Transfer learning**
  - Separate category vs. subtype of supervised learning
  - Data used for training the model are transferred from a different related domain
    - Data were developed for use in a domain different than the one intended for the model
    - Example: Using natural images from ImageNet to train a models for medical images [Alzubaidi et al 2021]
  - Coarse training done on transferred data
  - Fine tune training with smaller data directly related to domain of use
  - Reasons
    - Data is expensive
    - Higher quality and quantity data may be more available, cheaper in another domain

ML Definitions - Data

- **Instance**
  - Single event in a data set
  - # instances required to train a model depends on the problem and model used
  - **Outlier**
    - Instance which is significantly different from the remaining instances in the population
    - Can skew results
    - Different models have different sensitivities to outliers
- **Label** – observed value for a feature of an individual instance
- **Feature**
  - An aspect (variable) of the training data

	Feature 1	**Feature 2**	**Feature 3**
Instance 1	Red	Slow	Yes
Instance 2	Red	Fast	No
Instance 3	Green	Medium	No

Red, Green, Slow, Fast, Medium, Yes and No are all **labels** in this data set.

ML Definitions - Data

- **Feature**
  - a.k.a. **vector, attribute**
  - In clustering and instance-based methods, called a **dimension**
  - Features can be selected columns in the raw data or..
    - Can be calculated or combinations of data from >=1 columns in the raw data
    - e.g., Body mass index is a calculation of mass (body weight) and height
  - **Geocoder**
    - Type of feature that maps a data point to a map so that you know where the event happened

ML Definitions – Models

- **Algorithm**
  - Repeatable process used to train a model from a given set of training data
- **Parameter**
  - Internal values inside machine learning that the model derives based on training data
  - e.g., weights, bias values
- **Model** = algorithm + parameters
  - When a model is used for classification, it is called a **classifier** [Asiri 2021]
  - **Weak learner (weak model):** model whose performance only slightly > random chance
  - Good model: model that **generalizes well** (it performs the same on new data as it did on the training (and test) data)
- **Epoch**
  - 1 epoch = 1 pass through the training data

ML Definitions – Models

- **Hyperparameter**
  - Parameter that is manually set prior to running the algorithm (not set by the model)
  - Manually specify change or limits in input weights that loss function can make per training step
  - These are adjusted during model optimization (**tuning**)
  - Examples of hyperparameters:
    - Limit on total # epochs
    - K in K-means clustering
- **Loss function** (objective, cost)
  - measure of the deviation from the correct output
- **Stochastic gradient descent** method to reduce loss function
  - Mathematically descends the curve of the loss function to a minimum value
  - Effective training method for most models

ML Definitions – Model Evaluation

- Most methods to evaluate models are for **supervised** models
  - Some can be used for both supervised and unsupervised models
  - A few are more often used for unsupervised models
- Unless noted to be for unsupervised models, the methods displayed are primarily (or only) for supervised models
- Not all methods of evaluating a model are shown

ML Definitions – Model Evaluation

- **Signal**
  - The true underlying pattern you are trying to learn from the data
  - Well designed machine learning separates signal from noise
- **Noise**
  - Irrelevant information or randomness in a data set
  - **Irreducible error**
    - Noise that can't be reduced by optimizing algorithms
    - Can sometimes be reduced by better cleaning of data
    - Due to
      - Inherent randomness
      - Misframed problem
      - Incomplete feature set

ML Definitions – Model Evaluation

Bias

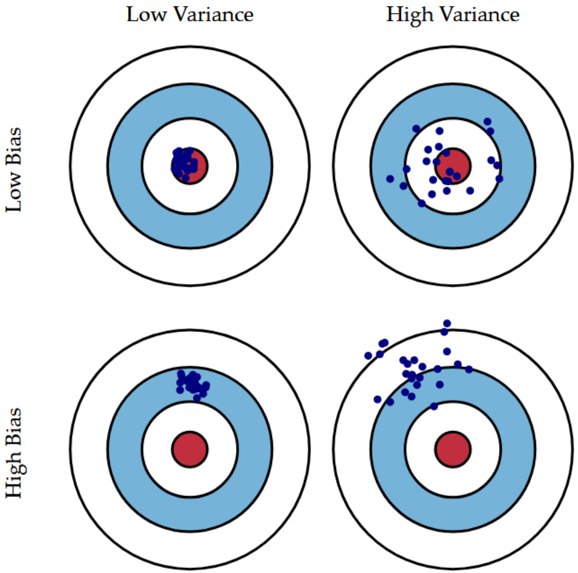
- Measure of inaccuracy
- High bias + low variance ➔ consistently inaccurate results

Variance

- Measure of imprecision (lack of reproducibility)
- High variance + low bias ➔ inconsistently accurate results

Irreducible error

- Noise that cannot be reduced by optimizing algorithms



https://devopedia.org/bias-variance-trade-off

ML Definitions – Model Evaluation

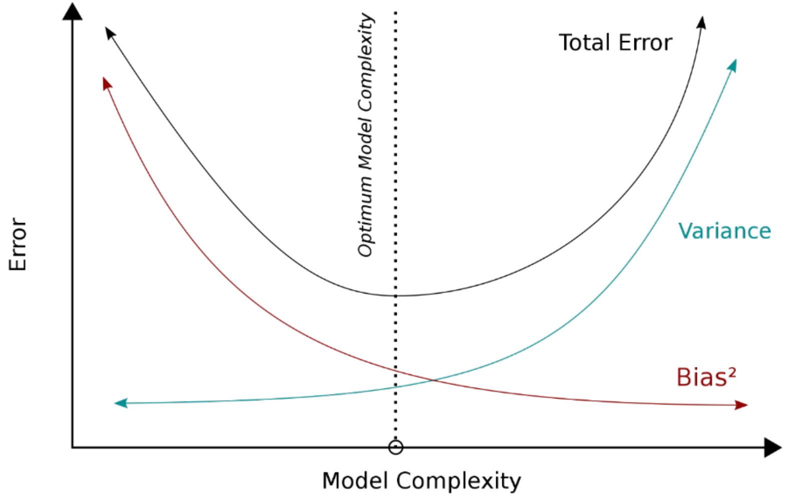
Bias

- *Not just an ethical term…*
- Amount of inaccuracy in the model's performance after training
- High bias → model is inaccurate (underfit)
- Low bias → model is accurate (but may be overfit)

Variance

- Amount of imprecision (square of standard deviation (σ) → σ^2)
- Due to model's sensitivity to small fluctuations in the training set
- High variance → model is imprecise (and likely overfit)
- Low variance → model is precise (but may not be accurate and may be underfit)

ML Definitions – Model Evaluation



- Bias-Variance Trade-Off
  - Things that reduce variance increase bias
  - Things that reduce bias increase variance

$$Total\ error = (bias^2) + variance + irreducible\ error$$

https://en.wikipedia.org/wiki/Bias%E2%80%93variance_tradeoff
https://towardsdatascience.com/understanding-the-bias-variance-tradeoff-165e6942b229

ML Definitions – Model Evaluation

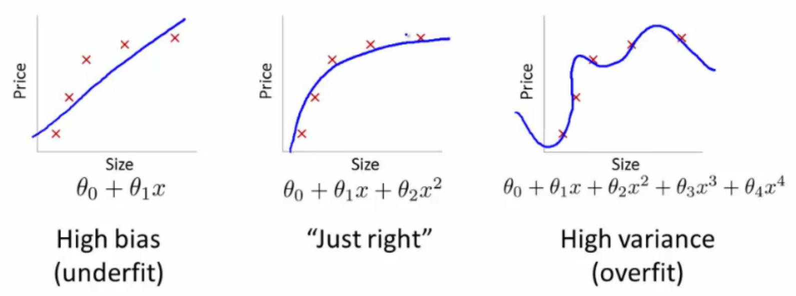
- **Goodness of fit**
  - How closely a model's output values match the observed (true) values
- **Underfitting**
  - Model does not accurately predict output for the data fed to it
    - high bias, low or high variance
  - More common at beginning of model development prior to tuning
  - Causes related to model and its settings
    - Too simple or rigid for the data (e.g., using linear model for complex data)
    - Training is paused too early
    - Hyperparameters are suboptimal
    - Not enough features selected
    - Suboptimal features selected
  - Causes related to data
    - Training data more simple than real-world data (underrepresented populations)

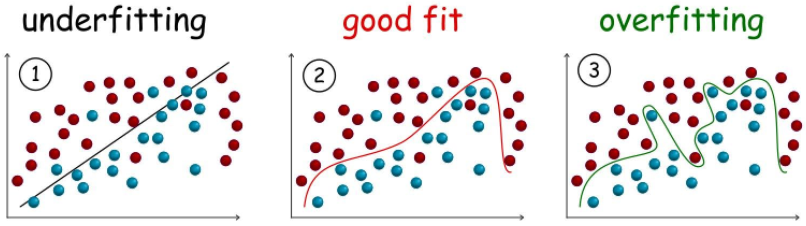
ML Definitions – Model Evaluation

- **Overfitting**
  - Occurs when statistical model exactly fits training data BUT…
    - Does not fit new data well (test or production data)
  - Training set has low error rate but test set has high error rate = high variance
  - **Most common problem** for any statistical model using a training set
  - Causes related to model and its settings
    - Too many features selected for # instances in the data set (most common reason for overfitting)
      - > 1 feature selected per every 10 instances is too many
      - Excessive detail in training data
    - Model selected is designed for data more complex than the data examined
      - e.g., using a neural network for linear relationships
      - too many parameters learned
    - Model trains too long (too many epochs)
  - Causes related to data
    - Training data is more complex than real-world data

Salmasian H. U4M6L1. Introduction to advanced Analytics models. AMIA Health Information Certification Course. Accessed August 30, 2021

ML Definitions – Model Evaluation





https://datascience.stackexchange.com/questions/361/when-is-a-model-underfitted

https://www.pianshen.com/article/57881552343/

Cause of overfitting	Prevention / Correction Technique
Model has too many features (>1 per 10 instances)	• Remove irrelevant input features (avoid underfitting) • **Dimensionality reduction** • Increase total number of instances
Model trains too long	• Early stopping (fewer epochs, cycles)
Training data is more complex than real-world data	• Add instances to avoid sampling error and reduce complexity
Model is too complex for the data	• Choose simpler model • **Regularization**

Salmasian H. U4M6L1. Introduction to advanced Analytics models. AMIA Health Information Certification Course. Accessed August 30, 2021

ML Definitions – Model Evaluation

- **Confusion matrix – Single Class**
  - Assessment of model's *"confusion"* in analyzing instances (i.e., assigning instances to the wrong class or outcome)
  - Evaluated for each outcome / class the model produces (e.g., classification)
  - For *binary* outcome → simple 2 x 2 table
  - # columns, # rows = # output classes

Has Hodgkin Lymphoma (HL) or not?	**Predicted HL**	**Predicted not HL**	
Actual HL	TP = 9	FN = 10	FN+TP = 19
Actual not HL	FP = 1	TN = 80	TN+FP = 81
	FP+TP = 10	TN+FN = 90	Total = 100

Measure	**Formula**	**Example Result**
Accuracy or Correct Classification Rate	$\frac{TP + TN}{Total}$	$\frac{9 + 80}{100} = 0.89$
Misclassification or error rate (1 – Accuracy)	$\frac{FP + FN}{Total}$	$\frac{1 + 10}{100} = 0.11$
Sensitivity **(Recall, True Positive Rate)**	$\frac{TP}{FN + TP}$	$\frac{9}{10 + 9} = 0.47$
Specificity **(True Negative Rate)**	$\frac{TN}{TN + FP}$	$\frac{80}{80 + 1} = 0.99$
Precision **(Positive Predictive Value)**	$\frac{TP}{TP + FP}$	$\frac{9}{9 + 1} = 0.90$

ML Definitions – Model Evaluation

- **Confusion matrix – Multi-class**
  - Evaluated for each outcome / class the model produces (e.g., classification)
  - # columns, # rows = # output classes

  - Confusion matrix with five possible outcome classes
  - TP for each class in green

		Predicted Hodgkin Lymphoma Type				
		NLP	LR	NS	MC	LD
Actual	NLP	60	2	3	4	3
	LR	1	10	3	2	1
	NS	0	4	60	0	6
	MC	6	3	3	20	0
	LD	1	2	1	0	5

ML Definitions – Model Evaluation

- **Confusion matrix – Multi-class**
  - Evaluated for each outcome / class the model produces (e.g., classification)
  - # columns, # rows = # output classes
  - Confusion matrix with five possible outcome classes

Predicted Hodgkin Lymphoma Type

		NLP	LR	NS	MC	LD
Actual	NLP	60	**2**	**3**	**4**	**3**
	LR	**1**	10	3	2	1
	NS	**0**	4	60	0	6
	MC	**6**	3	3	20	0
	LD	**1**	2	1	0	5

- Calculating ONLY for **Nodular Lymphocyte Predominant (NLP) Hodgkin Lymphoma**:
  - TP_{NLP} = 60
  - **FP_{NLP} = sum of predicted NLP which were not NLP = 1 + 0 + 6 + 1 = 8**
  - **FN_{NLP} = sum of actual NLP not predicted as NLP = 2 + 3 + 4 + 3 = 12**
  - TN_{NLP} = all remaining items in matrix = 120
  - Total Hodgkin Lymphoma cases = 200

For NLP Hodgkin Lymphoma only:

Measure	Formula	Example Result
Accuracy or Correct Classification Rate	$\frac{TP+TN}{Total}$	$\frac{60+120}{200}=0.9$
Misclassification or error rate (1 – Accuracy)	$\frac{FP+FN}{Total}$	$\frac{8+12}{200}=\mathbf{0.1}$
Sensitivity (Recall, True Positive Rate)	$\frac{TP}{FN+TP}$	$\frac{60}{12+60}=0.83$
Specificity (True Negative Rate)	$\frac{TN}{TN+FP}$	$\frac{120}{120+8}=0.94$
Precision (Positive Predictive Value)	$\frac{TP}{TP+FP}$	$\frac{60}{60+8}=0.88$

ML Definitions – Model Evaluation

- **Confusion matrix – Multi-class**
  - Evaluated for each outcome / class the model produces (e.g., classification)
  - # columns, # rows = # output classes
  - Confusion matrix with five possible outcome classes

Predicted Hodgkin Lymphoma Type

Actual

	NLP	LR	NS	MC	LD
NLP	60	2	3	4	3
LR	1	10	3	2	1
NS	0	4	60	0	6
MC	6	3	3	20	0
LD	1	2	1	0	5

- Calculating ONLY for **Lymphocyte-Rich Hodgkin Lymphoma**:
  - TP_{NLP} = 10
  - **FP_{NLP} = sum of predicted NLP which were not NLP = 2 + 4 + 3 + 2 = 11**
  - **FN_{NLP} = sum of actual NLP not predicted as NLP = 1 + 3 + 2 + 1 = 7**
  - TN_{NLP} = all remaining items in matrix = 172
  - Total Hodgkin Lymphoma cases = 200

For NLP Hodgkin Lymphoma only:

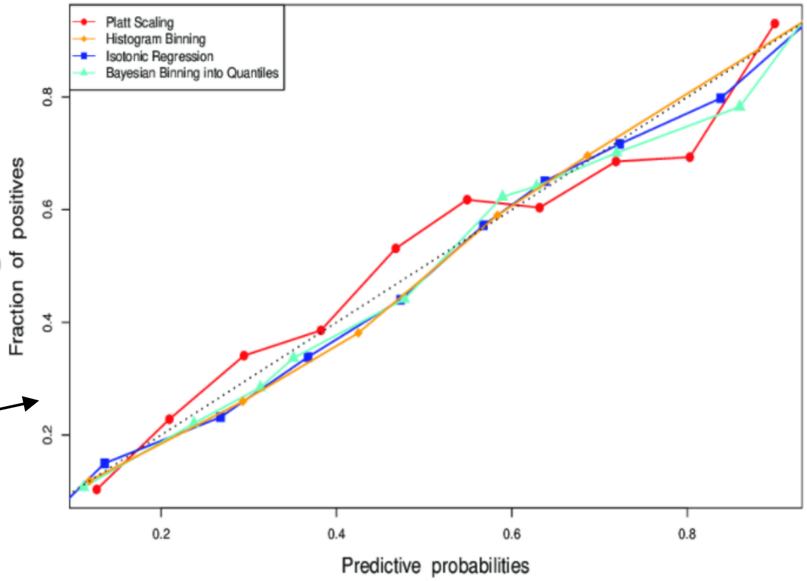
Measure	Formula	Example Result
Accuracy or Correct Classification Rate	$\frac{TP+TN}{Total}$	$\frac{10+172}{200} = 0.91$
Misclassification or error rate (1 – Accuracy)	$\frac{FP+FN}{Total}$	$\frac{11+7}{200} = \mathbf{0.09}$
Sensitivity **(Recall, True Positive Rate)**	$\frac{TP}{FN+TP}$	$\frac{10}{7+10} = \mathbf{0.59}$
Specificity **(True Negative Rate)**	$\frac{TN}{TN+FP}$	$\frac{172}{172+11} = 0.94$
Precision **(Positive Predictive Value)**	$\frac{TP}{TP+FP}$	$\frac{10}{10+11} = \mathbf{0.48}$

ML Definitions – Model Evaluation

- **Null error rate**
  - For classification methods, rate of being wrong if you ALWAYS pick the majority class
  - If the majority class has 105 instances out of 165 total instances
    - Null error rate = (165 – 105)/165 = 36%
  - **Accuracy paradox**
    - Best classifier for the intended use may have a higher error rate than the null error rate
    - Occurs when condition or outcome is very low percentage of overall data set (e.g., 1%)
    - Model can correctly predict absence of the condition in 99% of cases – hooray! BUT…
    - May completely fail to detect the condition being sought
      - 100% failure of detecting the condition (but null error rate is only 1%)
    - Take home point → Use different statistical methods when trying to screen for low incidence conditions
      - **F_β score** (see supplemental material)
      - **Matthews Correlation Coefficient**
      - **Stratified K-fold cross-validation**

ML Definitions – Model Evaluation

- For classification methods (categorical output)
  - **Cohen's kappa**
    - Measures how well the classifier performs as compared to random chance
  - **Calibration (probabilities)**
    - Observed vs. expected probability of class membership
    - Must be good if the model is to be used for prognostic models
    - Assess with **reliability diagrams** [Dimitriadis et al. 2021]
      - Observed event frequency plotted against predictive probability



[Huang et al. 2020]

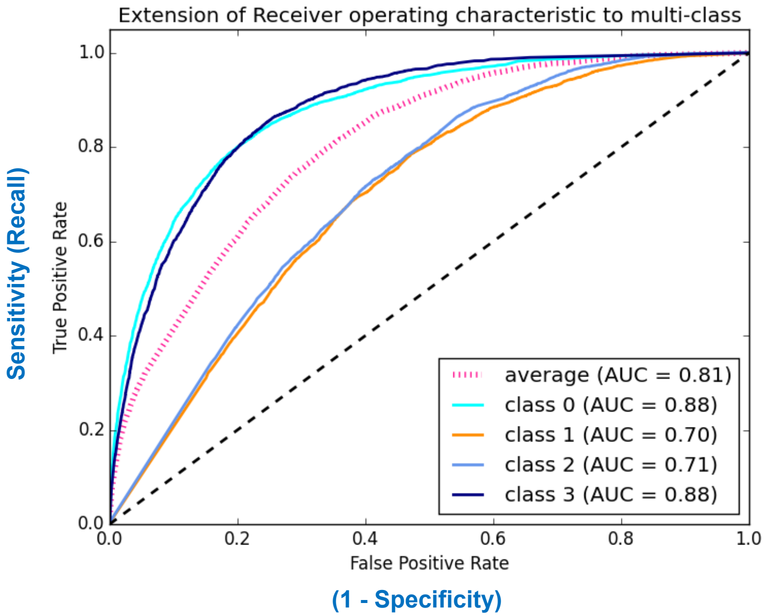
ML Definitions – Model Evaluation

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{x_i - x_{\text{true}}}{x_{\text{true}}}\right)^2}$$

- For regression methods (numerical output)
  - **Root mean squared error (RMSE)**
    - **n**: number of measurements; **x_i**: observed measurement; **x_{true}**: actual measurement
    - Nearly guaranteed to be higher on test data than training data
    - The lower the result, the better the fit (the more the observed values match expected values)
    - If RMSE is higher on test data than on training data → model has overfit the training data
  - **Correlation (r) and coefficient of determination (r^2)**
    - **Correlation coefficient (r):** indicates strength of relationship between x and y on a scatter plot
      - Perfectly positively correlated model: r = 1
      - No correlation whatsoever: r = 0
      - Perfectly negatively correlated model: r = -1
      - Correlation does **not** imply causation
    - **Coefficient of determination (r^2)**
      - A.k.a. goodness-of-fit
      - Values between 0 and 1, expressed as percent (%); 100% is a perfectly fit model
      - Represents percent variation in y that is *not* explained by variation in x
      - Proportion of the variance in the predicted variable accounted for by the model

ML Definitions – Model Evaluation

- For both classification (supervised) and unsupervised models - **Discrimination capability**
  - Measures model's ability to discriminate between groups, classes or clusters
  - **Receiver Operated Characteristic (ROC) curve**
    - Plot of sensitivity against (1-specificity)
  - **Area Under the Curve (AUC)**
    - Area under the ROC curve
    - a.k.a. **concordance (C) statistic**
    - AUC of 1 = perfect discrimination between groups
    - Threshold for acceptable performance
      - Model to replace human → very high (near 1)
      - Model to assist human → 0.7-0.9 may be ok



https://stackoverflow.com/questions/43537579/roc-curve-in-un-balanced-data

AMIA

INFORMATICS PROFESSIONALS. LEADING THE WAY.

Machine Learning

Model Development Process

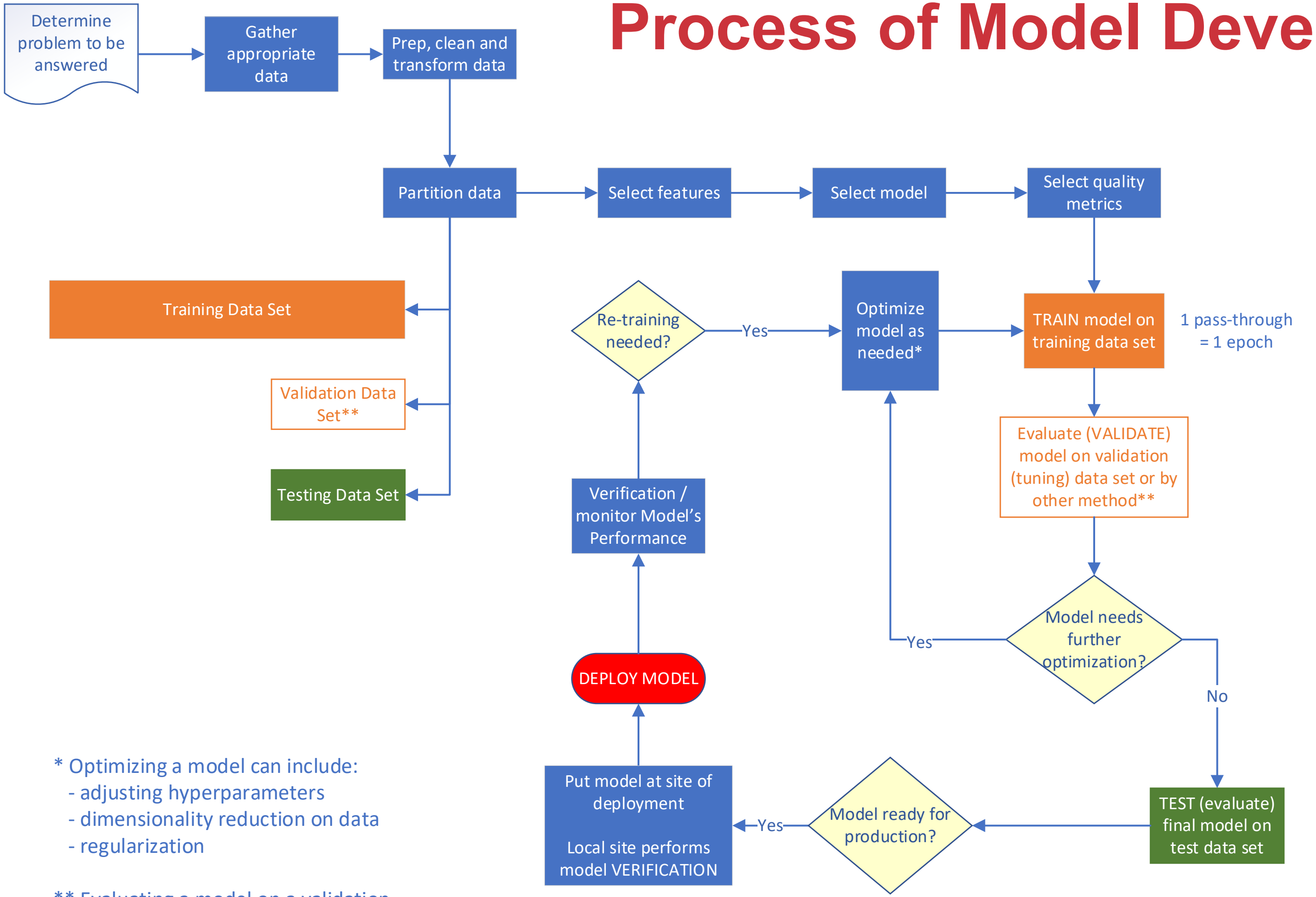
Clinical Informatics
Board Review Course

Process of Model Development

- Many ways that a model can be trained → tested → deployed
    - Depends on model, amount of data, and other factors
- Phases of model development have variable nomenclature between authors
    - E.g., learning phase, inference phase
- A few definitions to resolve possible confusion

	What it means in machine learning…	**What it means in a hospital laboratory…**
Validation	Evaluating preliminary (non-final) *model* • Results of evaluation lead to tweaking (tuning) the model	Final evaluation of a *laboratory test* where no further changes to the test procedure are expected
Testing	Final evaluation of a *machine learning model* where no further changes to the model are expected	Evaluating preliminary (non-final) *laboratory test* OR Performing live clinical testing

Process of Model Development



Determine problem to be answered

Gather appropriate data

Prep, clean and transform data

Partition data

Training Data Set

Validation Data Set**

Testing Data Set

Select features

Select model

Select quality metrics

TRAIN model on training data set

1 pass-through = 1 epoch

Evaluate (VALIDATE) model on validation (tuning) data set or by other method**

Model needs further optimization?

Yes

Optimize model as needed*

No

TEST (evaluate) final model on test data set

Model ready for production?

Yes

Put model at site of deployment

Local site performs model VERIFICATION

DEPLOY MODEL

Verification / monitor Model's Performance

Re-training needed?

Yes

* Optimizing a model can include:
- adjusting hyperparameters
- dimensionality reduction on data
- regularization

** Evaluating a model on a validation data set may not always be needed.

Process of Model Development

1. Determine the problem to be answered

- Foundational to rest of the steps in model development

2. Gather appropriate data (instances)

- **Most important, time-consuming and expensive part of the process**
- Data set = collection of instances
- Collect data from multiple sources to simulate real-word data for intended use
    - Genders, races, ethnicities, socioeconomic statuses, etc.
- Exception: **Transfer learning**
    - Bulk of data obtained from a different domain because more of it is available
        - Bulk of training done with this data
    - Use smaller set of real-world data from intended domain for tuning

Process of Model Development

3. Prepare, Transform and Cleanse Data

- **Assess and/or create labels (supervised models only)**
  - Degree and extent of labeled data determines type of learning → informs model selection
  - Features should have similar/same scales of measure or labels; Labels should have similar/same names
  - **Manual labeling**
    - Performed by subject matter experts, error prone, can introduce bias, terms may not be standardized
  - **Automated labeling**
    - Can be performed using human-labeled data used as a template
    - Completely automated: No human-labeled data as a template
      - **Multiple-instance learning**: Automatically learns important details of instances grouped under broad labels
- **Cleanse data**
  - May need to normalize normally distributed data to standard deviation units OR…
  - Normalize to percentile span of data range if not normally distributed data
  - Correct missing/erroneous data
  - Remove outliers if possible without introducing bias

Process of Model Development

4. Partition data

- Randomly split data with no overlap between data sets
  - Potential for sampling errors between sets
  - Small data sets may not have much data to split
- Method of partitioning data may differ when data sets are smaller (see later)

Data Set Name	Typical % of Data	Description
Training Data Set	67-80%	• Data used to train the model
Validation (Tuning) Data Set *(not always used)*	10-15%	• Data used to evaluate *preliminary* trained model • Model is run on the data but is NOT trained on it • Model's performance on these data used to optimize model before re-training on training data set again • May not be used if no hyperparameters or if insufficient data in general
Testing Data Set	10-33%	• Data used to evaluate the *final* trained model • Model is run on the data but is NOT trained on it • Model's performance on these data used as its final evaluation

Process of Model Development

5. Select features (independent variables)

- General rule: select is <=1 feature for each 10 instances in the development data set
  - Higher number of features → overfitting
- Each feature should be selected based on its likely association and/or ability to explain the dependent variable (outcome, condition)
- Manual selection by a human
  - Subject to human bias → constrains model
  - Features selected not always relevant or optimal
- Automatic selection by a computer algorithm (e.g., deep learning)
  - Some ML models automatically select features during primary model training

Process of Model Development

5. Select features (independent variables)

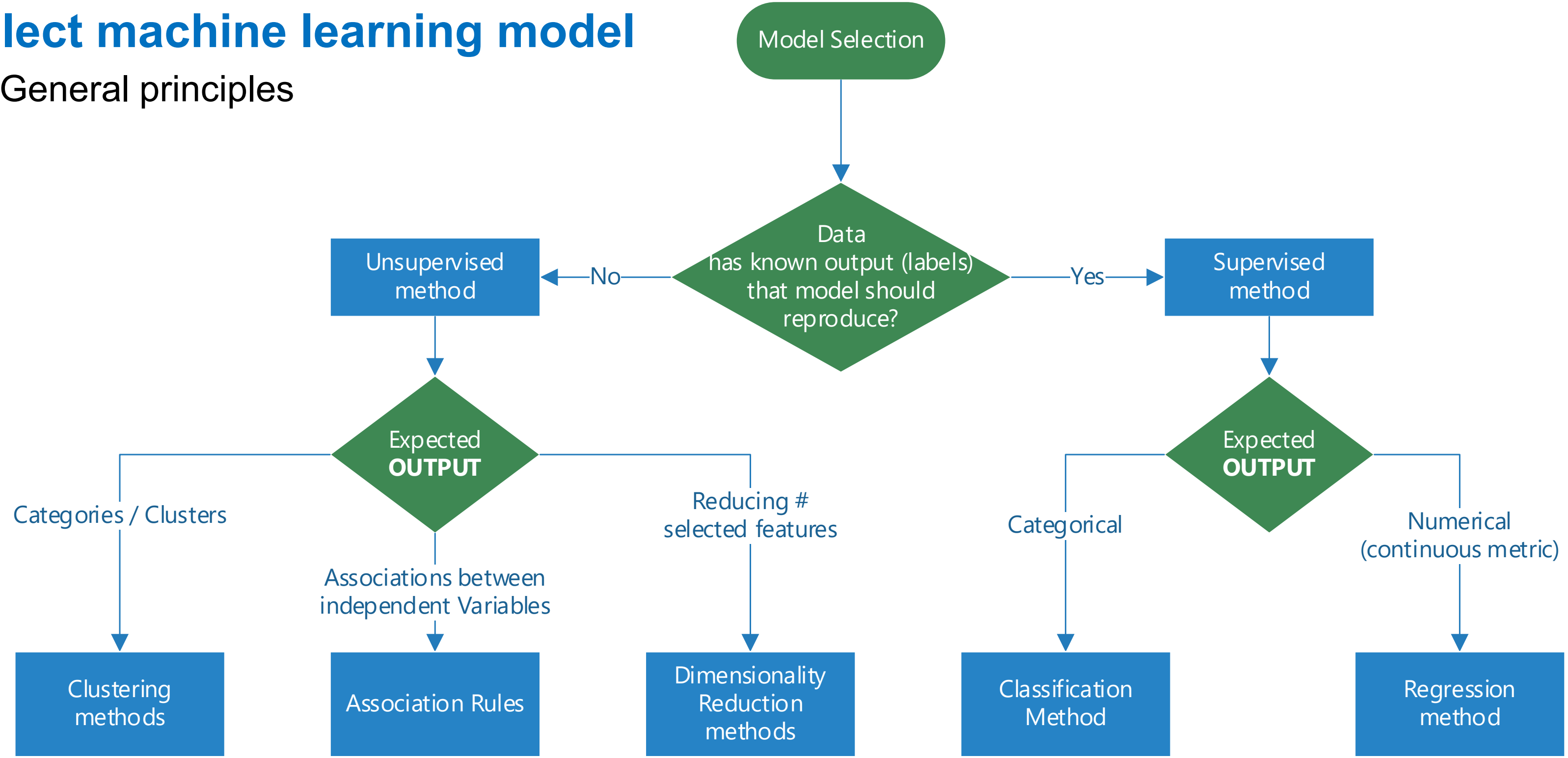
Methods of automatic feature selection	
Forward selection	• Iterative inclusion of new feature as long as it makes a contribution that explains the variation in the dependent variable • Stops when no additional contributing variables (features) found
Backward selection	• For regression models • Remove feature if it worsens the strength of the association • Iteratively remove variables (features) until removal worsens the strength of the association
Stepwise selection	• Combination of forward and backward selection • Forward feature added, then all included variables are evaluated for backward selection to validate the addition of the forward feature
Forced inclusion	• Features are selected based on known association in prior studies

- When a model is overfit, consider reducing the number of features
- **Dimensionality reduction methods**
  - Unsupervised machine learning model
  - Unique because used as an adjunct to another model
  - Goal: reduce the number of selected features (dimensions) by determining most important ones
  - See unsupervised ML methods below

Process of Model Development

6. Select machine learning model

General principles



Process of Model Development

7. Choose quality metrics based on model

- Should be appropriate for machine learning method AND purpose of the model
  - e.g., looking for majority classes vs. minority classes (screening for low incidence conditions)

8. Train model on training data set

- 1 pass through of training the model = 1 epoch
- After model has been optimized, it may train again on the training data set

Process of Model Development

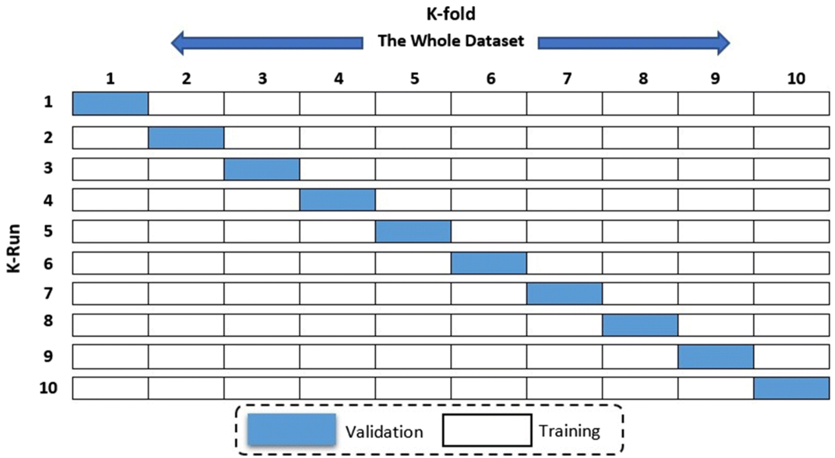
9. Evaluate (validate) model's performance

- If adequate data present
  - Run model on separate "hold out" validation (tuning) data set
  - Check performance of the preliminary model using methods described earlier
  - May need different metrics for low incidence classes
  - Determine what, if any, changes need to be made to
    - Hyperparameter settings
    - Feature selection
    - Model complexity
  - Some models will integrate checking performance and adjusting model
- If amount of data (instances) is suboptimal
  - Data is split into training data set and testing data set (no up-front split of validation data set)
  - Use cross-validation, bootstrapping or other equivalent method to train the model *and* evaluate it

Machine Learning Validation Methods

K-fold cross-validation

- Method of choice for smaller data sets
  - Some say to use it for larger data sets as well
- Training data randomly partitioned into equal # (k) subsamples (folds)
  - k is typically set between 5 and 10
  - If k = 10, then hold out 1 fold as a validation data set and use the remaining 9 folds collectively as the training data set
  - Train the model on the collective 9 folds together then validate and calculate performance metrics on the remaining validation fold
  - Process is repeated until all combinations of training data and validation data are evaluated
  - Quality metrics performed on each validation fold are averaged across each k-run



https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-019-0990-x

- Pros: decreases risk of sampling error, overfitting and underfitting
- Cons: Requires training and validating the model a total of k times (i.e., if k=5, total # training epochs = 5, total # validations = 5)

Machine Learning Validation Methods

- **Leave One Out Cross-Validation (LOOCV)**
  - Extreme case of k-fold where k = n (number of instances)
  - Use when you have very small data sets where standard K-fold cross-validation will not work
  - Can focus on noise unique to that dataset, so this method is NOT preferred

Machine Learning Validation Methods

- **Bootstrapping** (a.k.a. **bootstrap sampling**; **resampling cross-validation**)
  - Start with data set of n instances (original population) and create a new population of n instances (bootstrapped population) by:
    1. Randomly drawing a single instance out of the original population and place it in the “bag”; chance of selection = 1/n.
    2. Randomly draw another instance out of the original population as if the first instance had never been removed (or had been *replaced* in the original population); chance of selection still = 1/n per instance. Selected instance may be the same or different as one previously selected.
    3. Repeat process until n draws have been completed.
  - Instances in the “bag” → **bootstrapped sample**.
    - Instances never selected → **“out-of-bag” (OOB) sample**. (see supplemental material)
  - Because you are always drawing from the entire original population, it is likely that the *same instance may be selected more than once.*

Machine Learning Validation Methods

- **Bootstrapping** *(continued)*
  - As n increases:
    - On average, 0.632 (63.2%) of instances are selected for the bag
    - On average, 0.368 (36.8%) of instances are never chosen and are out-of-bag.
  - Train the model on the bagged instances
  - Validate or test the model on the out-of-bag (OOB) instances
  - Repeat the bootstrapping process many times and calculate performance metrics for each collection of out-of-bag (OOB) instances and then average them
  - Advantages
    - Works well on small data sets; handles outliers well
    - Makes no assumptions about distribution of data (i.e., no assumptions about data being normally distributed)
    - Allows for calculation of standard errors and confidence intervals
  - Limitations
    - Can require long computation time; will have margin of error

Process of Model Development

10.Optimize Model (a.k.a. tuning)

- Goal: minimize deviation from the correct output (minimize the loss function)
- Methods of optimization
  - Tune (adjust) hyperparameters
  - Dimensionality reduction (see unsupervised algorithms)
  - Regularization (see below)
  - Consider a different model if the current one is simply not working

Process of Model Development

10. Optimize Model (continued)

- **Regularization**
  - Category of methods that artificially force algorithm to build a less complex model (i.e., more generalizable → less likely to overfit data)
    - Places constraints on the model's variability in the setting of noisy data, similar to curve smoothing
    - Can perform feature selection by adjusting weights of some features to zero → feature elimination
    - Prevents parameters from getting too large → slightly higher bias (inaccuracy) but significantly less variance (more precise)
    - Minimizes the loss function for data
  - Mathematical methods (see supplemental material)
    - **Least Absolute Shrinkage and Selection Operator (LASSO) (L1) regularization**
    - **Ridge (L2) regularization**
    - **Elastic Net regularization (combo of L1 and L2)**
    - Special regularization for neural networks
  - Non-mathematical methods (e.g., early stopping, pruning decision trees)

Process of Model Development

11.Test model on testing data set

- Performed by model developers on *final* model *after* all training and validation (tuning) is completed
- Performance should be approximate to that of the validation data set(s)
  - Differences can be due to bias or variance
  - Check for…
    - Overfitting (low bias, high variance)
    - Inadequate sample size during training and validation
    - Correct methods used for evaluation of performance
    - Correct model used for data and problem

Process of Model Development

12. Deploy model

- Remove model from training environment
  - Put model into the device or software where it will be used
- Types of deployed models
  - **Static model:**
    - Develop model via training --> stop training --> use model in static manner
    - Most common model used in medicine
  - **Incremental or continuous model:**
    - Incrementally or continuously retrained after deployment
    - Cannot be used in certain medical environments under federal law (e.g., laboratories)
    - Requires special monitoring and controlling to ensure that model is still performing accurately
- **Verification**
  - Performed by the local site using the final model
    - Check to ensure performance after transit and deployment in a new setting
  - Similar to verification of FDA-approved systems after they are received

Process of Model Development

13.Post-Deployment Monitoring

- Look for shifts and trends in model output that may represent increase in bias or variance
- **Model stability**
  - Ability of a model to produce similar output over a range of similar inputs, including inputs not previously seen but which are not substantially different from prior inputs
  - A stable model is a **robust model**
- **Unstable (brittle) models**
  - Models that do not produce consistent output when given substantially similar inputs

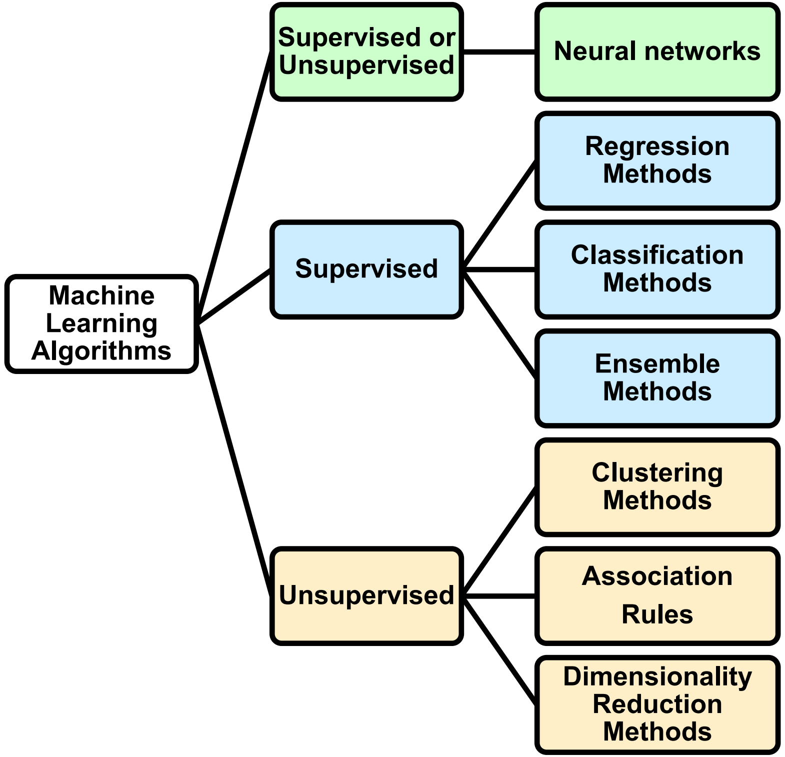
Machine Learning

Algorithms

Clinical Informatics
Board Review Course

Machine Learning Algorithms

- Each category has algorithms that are primarily used for that purpose
- However, classification algorithms may sometimes be used for regression and vice versa
- Unsupervised algorithms may sometimes be used with supervised learning



Machine Learning

Supervised or Unsupervised Algorithms

Artificial Neural Networks

Clinical Informatics
Board Review Course

Artificial neural networks (ANNs)

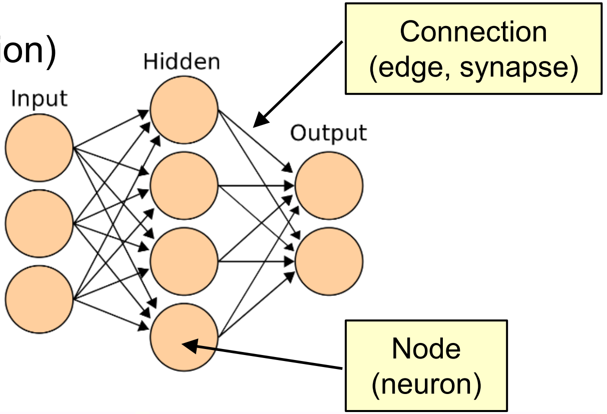
- **Artificial neural networks (ANNs)** - a.k.a. connectionist systems
  - Goal is to solve problems the way that a human brain would
  - Operate via flow of signals through nets of connections, akin to biological networks
    - Network trained to *optimize signals of each node-to-node connection* via adjusting weights (w) and biases (b) over activation (a) functions

$$example = \sigma(w_0 a_0 + w_1 a_1 + \ldots + w_{n-1} a_{n-1} + b)$$

  - Do *not* separate memory and processing
  - Advantages
    - Handles large amounts of complex data
  - Limitations / Disadvantages
    - Computationally intensive
      - Back-propagation and better processing technology have helped reduce the impact of this
    - Unraveling the pathways after training is completed can be difficult to impossible → **Black Box Problem**

ANN – Definitions

- Definitions
  - **Nodes** (akin to neurons) → transfer functions
  - **Connections** (akin to synapses, a.k.a. edges)
  - **Back-propagation** (nice YouTube video)
    - Process where ANN learns whether it made a mistake or not based on output
      - Adjusts internal parameters of transfer functions (nodes) using loss functions and stochastic gradient descent functions in waves *propagating backwards from the output nodes to the input nodes*
    - Helps speed up processing
  - Layers (nodes in each layer *usually* have same activation function)
    - **Input layer**: # nodes = # features selected in data
    - **Output layer**: # nodes = # output categories of data
    - **Hidden layer(s):**
      - **Shallow networks** usually have 1
      - **Deep networks** have >3



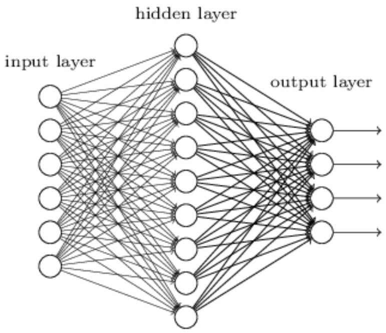
Connection (edge, synapse)

Node (neuron)

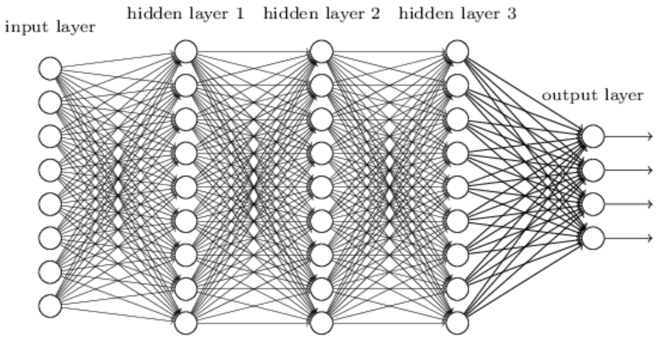
ANN – Definitions

- **Deep Learning** (a.k.a. deep networks; deep nets)
  - Goal: *imitate the human brain* in processing data and decision-making patterns
  - Usually multiple (Some say > 1 to >3 to hundreds to thousands) of hidden layers
    - Thousands to millions of interconnections; large number non-linear computations
  - Means more in-depth processing, *not* more in-depth knowledge

"Non-deep" feedforward neural network



Deep neural network



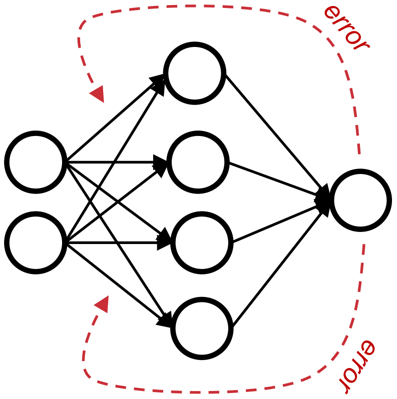
https://stats.stackexchange.com/questions/182734/what-is-the-difference-between-a-neural-network-and-a-deep-neural-network-and-w

ANN – Shallow vs. Deep Learning

	Shallow ANN	Deep ANN
Number of hidden layers	Usually 1	Usually >3
Amount of data on which the ANN performs well	Small to medium	Very large
Ability to accept variable input data	+	+++
Input data needs to be understood, labeled and have features selected	Yes	Not as much Can do automated feature extraction
Execution Time	Few minutes to hours	Hours to weeks
Amount of training data required	Less	Extensive and diverse (>10,000 instances)
Hardware	Can run on low-end machine	Requires powerful machine(s)
Interpretability	Easy to impossible	Difficult to impossible

ANN – Types

- Many, many types
- **Feed-forward network**
  - Refers to any ANN (or portion of ANN) which is unidirectional from input to output
  - Oldest type of ANN
- **Multilayer Perceptron (MLP)** (a.k.a. vanilla neural network)
  - **Perceptron**: iterative algorithm that determines best values for the coefficient vector
  - Typical example of a shallow feed-forward network
  - *Fully connected* multi-layer neural network
    - May have 1 or several hidden layers
  - Steps
    - Starting with the input layer, propagate data forward to output layer
    - Calculate error of output (observed vs. expected)
    - Backpropagate the error to adjust weights and biases to minimize it
  - Repeat steps over multiple epochs to learn ideal weights and biases



ANN – Convolutional Neural Network (CNN)

- Common applications
  - Useful for recognizing subpatterns and motifs in unstructured data (e.g., images)
  - **Image analysis** (extracting features, image classification), classifying time-sequence or gene-sequence data
- Input: > 10,000 instances ideal
- Output
  - Image classification and/or image feature selection
  - **Saliency map**
    - Feature (activation) map directly overlaid on the original image
    - Shows which features considered more relevant, but doesn't tell you why
- Advantages: Less "black box" than other neural networks
- Limitations / Disadvantages
  - Still don't know *why* features were chosen

- How it works (high level image analysis)
  - Deep learning network
  - **Convolution**: mathematical operation on two functions to produce a 3rd function that describes how shape of one function is modified by the other
  - **Convolution (filter) layer**
    - Data (pixels for images) that match the pattern of weights are amplified (creates hot spots)
  - **Pooling Layer**
    - Masks data except for the amplified values (hot spots)
    - Creates a **feature (activation) map**
  - Cycles of convolutions to pooled layers repeated until a defined set of criteria are reached THEN data sent to feed-forward network to get an overall classification of the data (image)

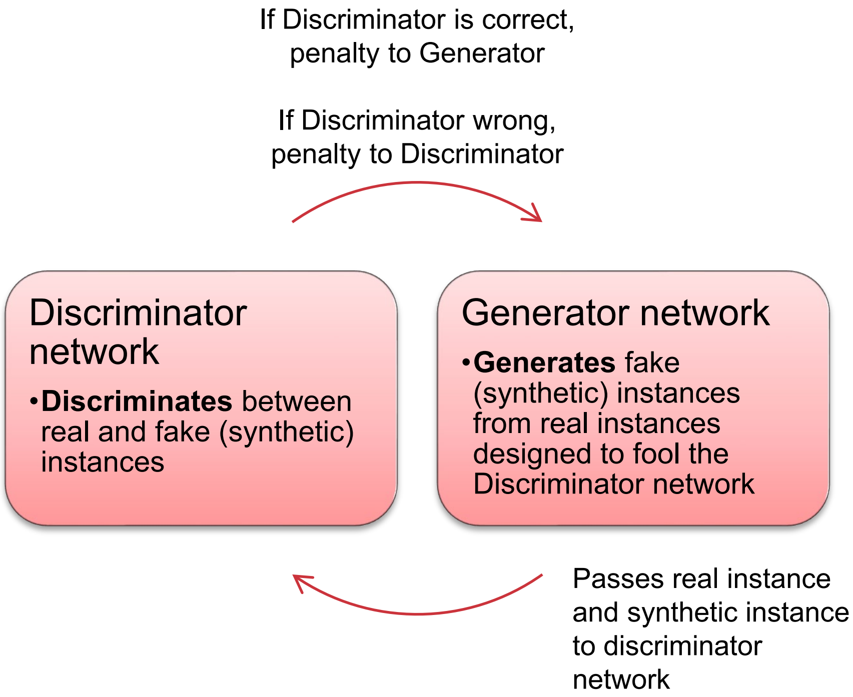
ANN – Recurrent Neural Network (RNN)

- Common applications
  - Commonly used in text classification (**natural language processing**)
  - Handles time-series data well (e.g., audio recordings, text feeds)
  - Anomaly detection in quality control data
- Many variations of this type
  - **Long Short-Term Memory (LSTM)**
    - Data storage units called **gated nodes**
    - Can flexibly represent short-term or long-term data
- Input: > 10,000 instances ideal

- How it works (high level)
  - Deep learning network
  - Output *recurrently feeds back on itself* to inform the next prediction (analyzes current and past data)
  - Data from prior instances to be used as input for subsequent instances
  - Network nodes can accumulate historical data → called **context nodes**
    - Have their own weights which are adjusted via backpropagation during training

ANN – Generative Adversarial network (GAN)

- Common applications (recent 2014)
  - Used to generate DeepFake images
  - Used to simulate cat-and-mouse fraud schemes
- How it works (high level)
  - Pairs of deep learning neural networks trained in tandem repeatedly
  - **Discriminator network**
    - With each iteration, increases ability to discriminate between synthetic instances from generator network and real instances
  - **Generator network**
    - With each iteration, increases its ability to fool the discriminator network
  - Training stops when Discriminator network can no longer discriminate real from fake instances (probability of real vs. fake is 50%)



If Discriminator is correct, penalty to Generator

If Discriminator wrong, penalty to Discriminator

Discriminator network

- **Discriminates** between real and fake (synthetic) instances

Generator network

- **Generates** fake (synthetic) instances from real instances designed to fool the Discriminator network

Passes real instance and synthetic instance to discriminator network

INFORMATICS PROFESSIONALS. LEADING THE WAY.

Machine Learning

Supervised Algorithms

Regression Methods

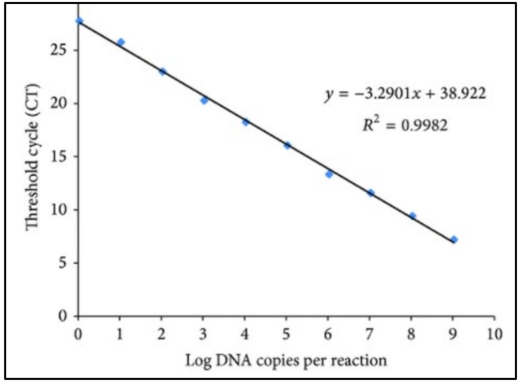
Clinical Informatics
Board Review Course

Regression Methods

- Used when expected output (dependent variable) is continuous (numerical)
- Requires the least number of training instances
- Describes, estimates or predicts the linear relationship between >=2 numerical variables
- e.g., estimating life expectancy, staffing shortages, population growth prediction, pandemic spread

Simple Linear Regression

- A.k.a. **univariate regression**
- Common uses
  - Model finds the line of best fit (calibration) which can then be used for prediction of y based on x
- How it works (high level)
  - Assumes linear relationship between dependent variable (y) and the independent variable (x)
  - Uses method of least squares to find the best line through the data
    - Resulting function: $y = \beta_0 + \beta_1 x + \varepsilon$
    - β_1 : slope (**regression coefficient**)
    - β_0 : y-intercept; ε = error
  - Goal is to minimize the loss function
- Input: *single* input variable (x; independent variable)
- Output
  - Single output variable (y; dependent variable)
  - Output is a continuous metric (numeric) variable



[Qiu et al 2014]

Multiple Linear Regression

- Common uses:
  - Determine strength of relationship between >=2 independent variables and a dependent variable
  - Value of dependent variable at certain independent variables
- How it works (high level)
  - $y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_i x_i + \varepsilon_i$
- Input: >1 input independent variable (covariate; x_1, x_2, etc.); no collinearity between them
- Output: Still a single output variable (y_i)

Polynomial Regression

- Common uses:
  - Models non-linear exponential data such as growth rates, progression of pandemics, etc.
- How it works (high level)
  - $y_i = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_i x^i + \varepsilon_i$
- Input: Uses exponentials of *single* input variable (independent variable x)
- Output: single output variable (y_i); dependent variable
- Advantages
  - Better fit to non-linear exponential data
- Limitations / Disadvantages
  - Higher risk of overfitting because it is sensitive to noise

INFORMATICS PROFESSIONALS. LEADING THE WAY.

Machine Learning

Supervised Algorithms

Classification Methods

Clinical Informatics
Board Review Course

Classification Methods

- Output is a categorical (class) variable
- Used for predictions, classification, categorization
- Predicts probabilities that an outcome (dependent variable) belongs to a particular class (independent variable)
- e.g., automated diagnosis, image classification

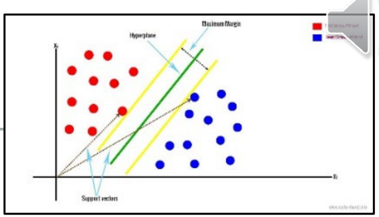
Logistic regression

- a.k.a. **logit regression**
- NOT a regression model (output is categorical → classification method)
  - Called regression because mathematical formula similar to regression methods
- Common application: Sepsis prediction models
- How it works:
  - Uses simple logistic function to fit data to sigmoid output
  - Goal is to maximize the **maximum likelihood estimation** (contrast regression methods)
- Input: Can be multiple independent variables (nominal, ordinal or metric)
- Output:
  - **Binary classification**: Probability that an instance belongs or does not belong to a single class
  - **Multiclass classification**: Probability that an instance belongs to one of n classes
- Advantages: widely used, easy to understand
- Limitations:
  - Cannot handle large numbers of features (input variables)
  - Cannot handle situations where relationship between input and output variables is not constant
  - Best for binary outcomes; may overfit if insufficient training data

Naïve Bayes Classifier

- Discussed elsewhere in the course in more detail
- Uses Bayes' Theorem to perform classification based on probability
  - Assumes independence between features (hence naïve)
  - Vector x representing some n features (independent variables)
  - Assigns current instance probabilities for every K-cluster of potential outcomes
  - Can predict binary, categorical and numerical data
- **Gaussian Naïve bayes**
  - Use for numerical data with Gaussian distribution

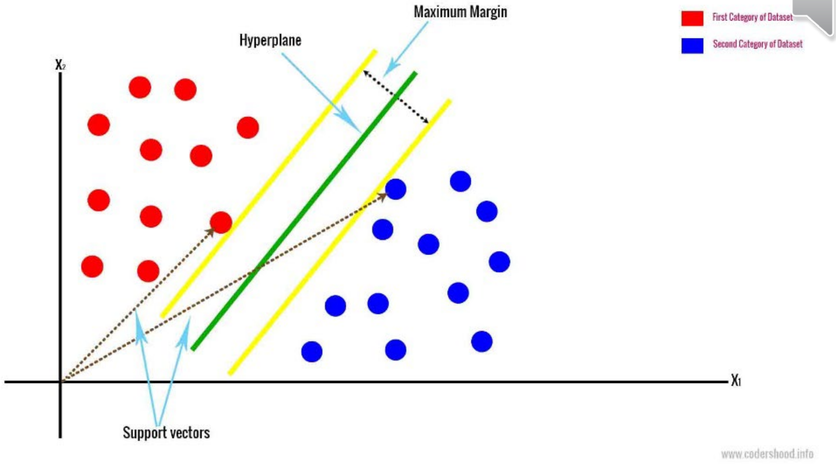
Support Vector Machine



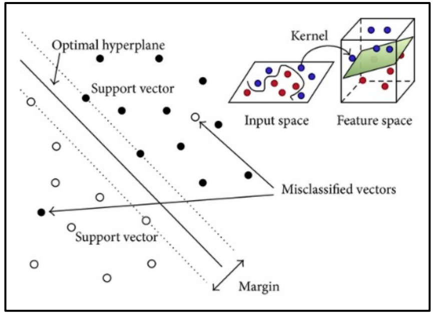
- Common uses: Fetal aneuploidy screening, Prediction of metastasis from gene profiles, Autoverification of GC/MS in the lab
- How it works (high level)
  - Determines optimal boundary between two classes in multidimensional (n-dimensional) space
    - 2 features, boundary is a line
    - 3 features, boundary is a plane (surface)
    - >3 features, boundary is a **hyperplane** and cannot be visualized
  - **Support vector** = data point closest to the boundary (hardest to classify)
    - Direct bearing on optimum location of the decision plane
- Training mechanism
  - Optimize boundary by maximizing margin between support vectors of different classes
  - Focus on borderlines is UNIQUE (outliers automatically ignored)
- Input
  - Categorical variables must be converted to numeric (see feature engineering in supplemental material)
  - Kernel and penalty hyperparameters
- Output
  - Binary or multiclass classification
  - New instances are classified based on location with respect to the hyperplane

Support Vector Machine

- Advantages
  - Efficient use of computer memory
  - Works well with high dimensional data
  - Less prone to overfitting
- Limitations / Disadvantages
  - # features cannot exceed # instances
  - Uses only numeric or dummy-coded categorical data
  - Decision boundary can be hard to interpret
  - Can be outperformed by random forest and gradient boosting machines
- Guide to Support Vector Machines



https://www.codershood.info/2019/01/10/support-vector-machine-machine-learning-algorithm-with-example-and-code/



https://www.hindawi.com/journals/ijbi/2013/323268/

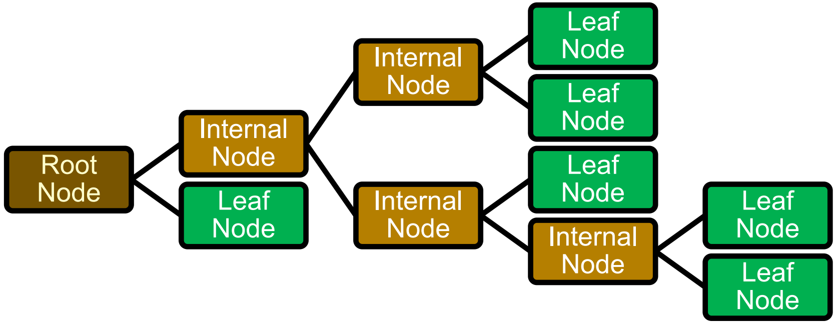
k-Nearest Neighbor

- Not the same as K-means clustering
- Common uses
  - Derive tumor infiltrating lymphocyte density
  - To predict treatment response
- How it works (high level)
  - Plots instances in multi-dimensional space
  - NO MODEL (**Instance-based method**)
  - Knowledge is stored in the structure of the mapped data (training data not discarded)

- Input:
  - Training instances > 100 up to 100,000 ideal
  - Distance function and k are hyperparameters
- Output
  - For each new instance (x), algorithm finds k training instances with closest distance to x and returns majority result
    - Class for classification problems
    - Mean for regression problems

Decision Trees

- Most common supervised classification method
  - Goal: classify a set of training instances accurately
  - Examples: Classification & Regression Trees (CART), C4.5, ID3
- How it works (high level)
  - Root node --> (internal node)n --> leaf node
  - Each node is a feature that is examined
    - If value of feature is below a specified threshold, left branch is followed; otherwise, right branch
    - Threshold splits population by purest possible subsets of classes
    - Each internal node = 1 feature (independent variable)
    - Each leaf node = outcome class (dependent variable)
  - Repeats recursively until purity cannot be improved
  - “Growing the tree” = training



- Specialized accuracy, error or quality checking for this method
  - **Gini impurity** = 0 with single class populations
  - **Entropy** = high when large number of evenly mixed classes

Decision Trees

- Input: Training instances > 100 up to 1,000,000 ideal
- Output
  - Probability of class membership of the new instance
- Advantages
  - Each node easily *explainable* as if-then cutoffs
- Limitations / Disadvantages
  - Susceptible to overfitting (high variance) because small variations in data can cause branches to be created that are not useful
  - Mitigate by limited number of nodes allowed in a branch OR…
  - By preventing nodes from being added unless they produce a statistically significant increase in purity (i.e., **"pruning" the decision tree**)

Machine Learning

Supervised Algorithms

Ensemble Methods

Clinical Informatics
Board Review Course

Ensemble methods

- Models that are groups (ensembles) of > 1 model to improve performance
- Common uses
  - Used on output of high variance (imprecision) and low bias (low inaccuracy)
  - Group of simple models may outperform a single complex model
  - Can be trained in parallel or in sequence (series)
- Models should be different from one another for best advantage
- Can be used for classification or regression
  - For **classification**, each model produces probability (vote) of a class then votes are counted
  - For **regression**, each model produces metric (numeric) variable averaged across models

Parallel ensembles

- Training different models in parallel with each other
- Methods to create diversity in parallel ensembles (helps decrease overfitting)
- **Bagging**
  - Train each model against *random subset of the training data*, i.e., **b**ootstrap **agg**regat**ing**
  - Attempts to reduce chance of overfitting complex models
  - Trains large number of relatively unconstrained models (strong learners) in parallel
  - Combines all models together to smooth out their predictions
  - Bootstrap each model in the ensemble of models then base outcomes on
    - Total # outcomes OR
    - Aggregate probabilities of each prediction
- **Random subspaces**
  - Train each model with entire training data set *but* use *random subsets of features* per model
  - Useful when number of features is large

Parallel ensembles – Random Forest

- Common uses: 30-day hospital readmission algorithms
- How it works (high level)
  - Ensemble of randomly selected decision trees (to make a "forest") run in parallel
  - Performance is better when each decision tree in the forest is different (**uncorrelated**)
  - Can be used for classification or regression
  - Uses <u>**both**</u> bagging and random subspaces as metrics
- Input
  - Training instances > 100 up to 1,000,000 ideal
  - 3 main hyperparameters need to be set before running the model
    - Node size, # trees allowed, # features sampled
      - Trees are kept short (limited # nodes in branches)
  - Data
    - Each sample population is a randomly bootstrapped sample with 2/3 used for training and 1/3 used for testing (**bagging**)
    - Each tree limits splitting strategy to random subset of features (**random subspaces** method)

Parallel ensembles – Random Forest

- Output
  - When used for classification
    - Output of each model aggregated by "voting"
    - Each vote weighted *equally* (i.e., not weighted, unlike boosting)
      - **Hard voting**: most frequent class selected is voted for
      - **Soft voting**: averaging probabilities for each class, selected then calculating average probability per class then selecting class with highest average probability
- Advantages
  - Can determine relative importance of individual features for classification tasks by calculating weighted average of decreased class impurity (increased purity) produced by all nodes of the forest that use each feature

Sequential (Series) Ensembles

- Each model in the ensemble is run sequentially (in series)
  - Uses multiple short decision trees as weak models
  - Each iteration focuses on learning from the mistakes of the one before it
  - Constructed using **boosting** (not bagging) methods
- **Boosting methods**
  - Attempts to improve predictive flexibility of models
  - Trains large # of constrained (weak learner) models (e.g., decision trees with limited depth)
  - Combines all constrained models into a single strong learner
  - Uses *weighted* voting (unlike parallel ensembles)
  - Advantages
    - No data preprocessing required; handles missing data; very powerful method with good performance
    - Helps reduce high bias (inaccuracy)…more likely to see bias with shallow decision trees
  - Limitations
    - Controversy over whether this method reduces or increases overfitting
    - Requires intense computation

Sequential (Series) Ensembles

AdaBoost (Adaptive boosting)

- Misclassified data from each algorithm have weights INCREASED
  - Increases chance that the next algorithm/model will classify them correctly
  - Well-classified data may have weights decreased
- Developed for binary classification
  - Also multiclass classification with soft voting
- Advantages
  - Quick and easy to use
  - Works well with large data sets
  - Uses several algorithms to improve accuracy
- Limitations
  - Takes longer to train
  - Can be impossible to interpret

Gradient Boosting

- Operates similarly to AdaBoost EXCEPT model trains on the residual errors of the previously run model
- Residual errors calculated using gradient descent algorithm to reduce the loss function

Others: CatBoost

Machine Learning

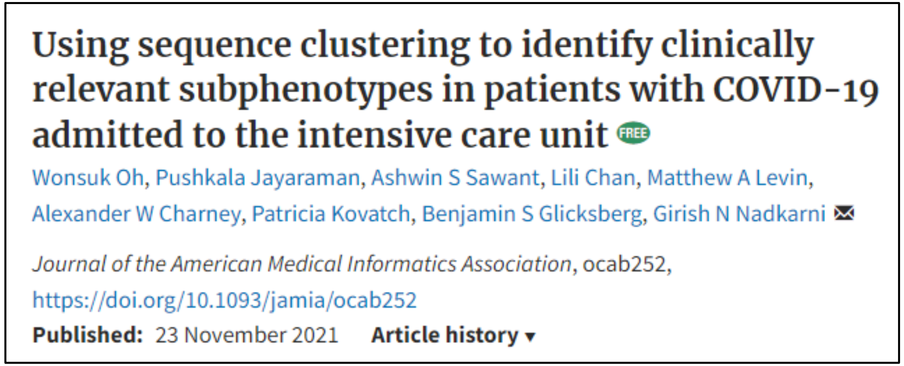
Unsupervised Algorithms

Clustering Methods

Clinical Informatics
Board Review Course

Clustering Methods

- Usually unsupervised
- Goal: *discover* patterns or relationships (dimensions) between data instances in a population
- Uses
  - Create hypotheses about data structure and associations
  - Simplify data while preserving features
  - Oh et al 2021
- Hard vs soft clustering
  - **Exclusive (hard) clustering**
    - An instance can only belong to 1 cluster
    - E.g., K-means clustering
  - **Fuzzy (soft) clustering**
    - An instance can have more than 1 cluster assignment



Using sequence clustering to identify clinically relevant subphenotypes in patients with COVID-19 admitted to the intensive care unit FREE

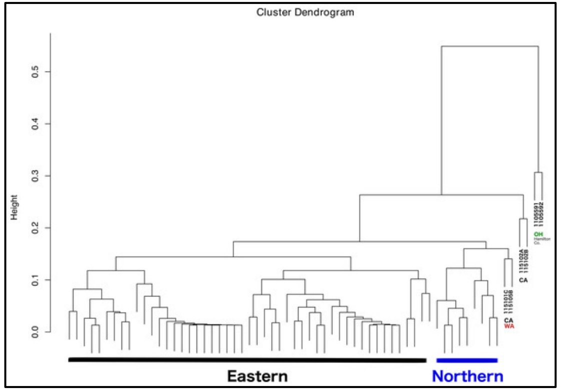
Wonsuk Oh, Pushkala Jayaraman, Ashwin S Sawant, Lili Chan, Matthew A Levin, Alexander W Charney, Patricia Kovatch, Benjamin S Glicksberg, Girish N Nadkarni ✉

Journal of the American Medical Informatics Association, ocab252, https://doi.org/10.1093/jamia/ocab252

Published: 23 November 2021 **Article history** ▾

Clustering Methods – Hierarchical Clustering

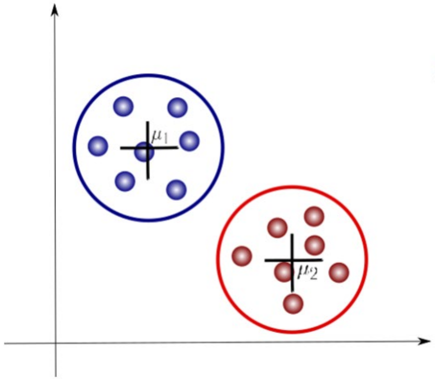
- Common uses
  - Bioinformatics and genetics --> relationships or classes of samples based on their genetic profiles, treatments, outcomes
- How it works (high level)
  - Instances are grouped based on similarities and differences
  - **Agglomerative clustering**
    - Bottom up approach; most common
    - Each instance is a cluster and successively merged with most similar other cluster
    - Each feature of an instance population = dimension
  - **Divisive clustering**
    - Top down approach; NOT common
    - Entire data is a cluster then divided into two clusters based on similarities and differences
    - Cycle is repeated until all instances are in separate clusters
- Output: **Dendrogram** (tree diagram)

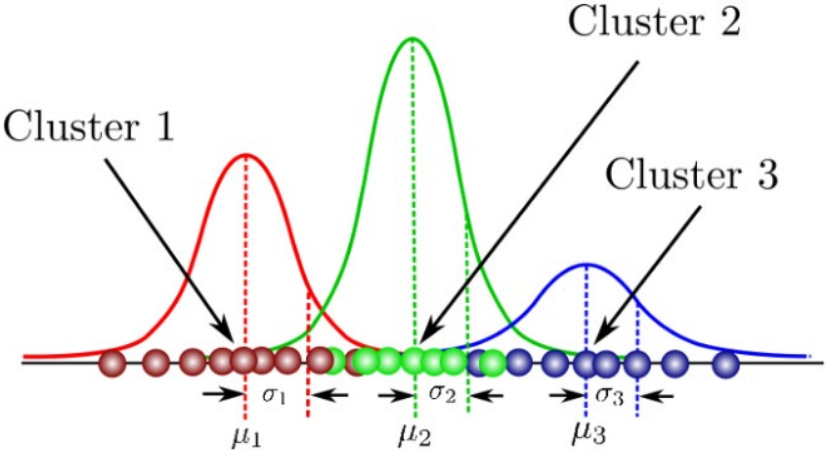


[Lado et al 2020]

Clustering Methods – Probabilistic Clustering

- Instances clustered based on probability that they belong to a particular distribution
- **Gaussian Mixture Model (GMM)**
  - Leveraged to determine which gaussian distribution an instance belongs to
- [Carrasco 2021]





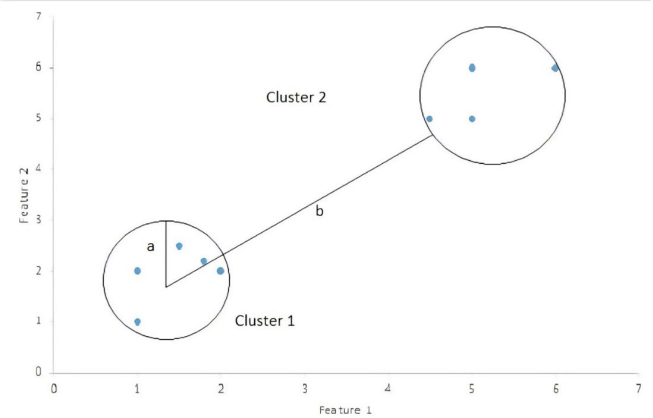
Clustering Methods – K-Means Clustering

- How it works (high level)
  - Instances assigned to manually defined number (**K**) of clusters based on their similarity
    - Randomly assign instances to clusters
    - Compute distances of all instances in the cluster from the **centroid** (center of the cluster) using a defined distance metric
    - Move instances closest to centroid of the K-cluster to that K-cluster
    - Recalculate the centroid position
    - REPEAT from above
  - Cycle stops when
    - Centroids stop moving location (no more cluster moves)
  - Higher k means more granularity; lower k means less granularity

- Not the same as k-Nearest Neighbor
- Common uses
  - Detection of cervical intraepithelial neoplasia
  - Works better with numerical data than other clustering analyses
- Limitations / Disadvantages
  - Ideal grouping will not occur when K(#) does not fit population characteristics
  - Values of some dimensions (features) may be correlated (redundant)
    - These can increase noise without improving clustering
    - Identify and remove these as part of data cleanup prior to training

Clustering Methods – K-Means Clustering

- Mitigate this by using good estimation of K
  - Compare compactness of groups for range of K #
  - Sum of squared error (SSE) for all members of clusters indicating compactness - change K and recalculate until optimal K defined
  - **Silhouette coefficient** →
    - Calculate ratio of cluster SSE & cluster separation



a: average intracluster distance
b: average intERcluster distance
Silhouette coefficient (S)

- $S = \frac{(b-a)}{\max(a,b)}$

Results are from -1 to +1

- 1: clusters well apart
- 0: clusters indifferent
- -1:clusters assigned incorrectly

Max(a,b): maximum distance of ALL distances between a and b for ALL cluster pairs

Machine Learning

Unsupervised Algorithms

Association Rules

Clinical Informatics
Board Review Course

Association Rules

- Rule-based method to find relationships (association rules) between independent variables
- Generates associations, *not* causation
- Rated in terms of support and confidence

Support	% total transactions from a transaction database that the rule satisfies
Confidence	Degree of certainty of an association

- Data must be converted to categorical data to use this algorithm type
  - Can be done through **discretization** = converting continuous data into bins

Association Rules

- **Market basket analysis**
  - Customers who bought X also bought Y
  - X and Y are independent variables
  - E.g., customers who bought dog leashes also bought fireplace logs
- **Apriori algorithm**
  - Uses a hash tree to count item sets navigating through data set in breadth first manner
  - Common uses
    - Detecting adverse drug reactions

Machine Learning

Unsupervised Algorithms

Dimensionality Reduction Methods

Clinical Informatics
Board Review Course

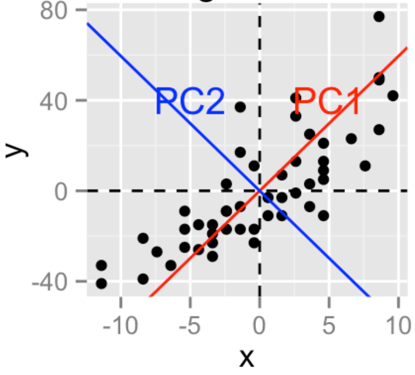
Dimensionality Reduction Methods

- Common uses
  - Typically used in the pre-processing stage rather than as a model on its own
    - prepares data for other models
- How it works (high level)
  - Methods that rank the importance of dimensions (features)
  - Goal → reduce the overall number of features to the most important ones
    - Many data sets have redundancy (**multi-collinearity**), which can cause problems with data
    - e.g., redundant attributes such as cancer diagnosis, problem list (which likely includes the cancer diagnosis) and ICD-10 codes (which also likely contain the cancer diagnosis and some of the other diagnoses)

Dimensionality Reduction Methods

- **Principle Components Analysis (PCA)**
  - Common uses
    - Use to identify unimportant dimensions OR…
    - Select first several dimension components (highest ranked) for features selection only
  - How it works (high level)
    - **Principle component**: axis through the data that is a function of contribution of variability in a population
    - Aligns with maximum variance in a population AND
    - Each principle component has to be **orthogonal** (at right angles) with all other principle components
    - >3 dimensions cannot be visualized
    - 1 principle component per dimension
  - Input
    - Usually requires continuous metric dimensions (features)
    - Variations can use categorical variables

Figure 1A



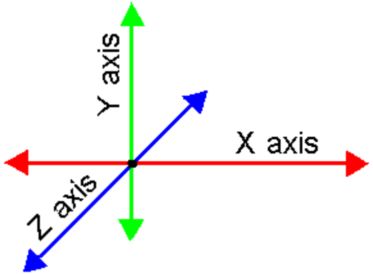


Image Source

Dimensionality Reduction Methods

- **Singular value decomposition (SVD)**
  - A = USVT
    - A: Matrix
    - U and V are orthogonal matrices
    - S is the diagonal matrix
- **Autoencoders**
  - Use neural networks to compress data then recreate data as output
  - part of an artificial neural network that uses unsupervised learning to reduce data dimensions (reduce noise)

That’s a wrap!

Key Readings

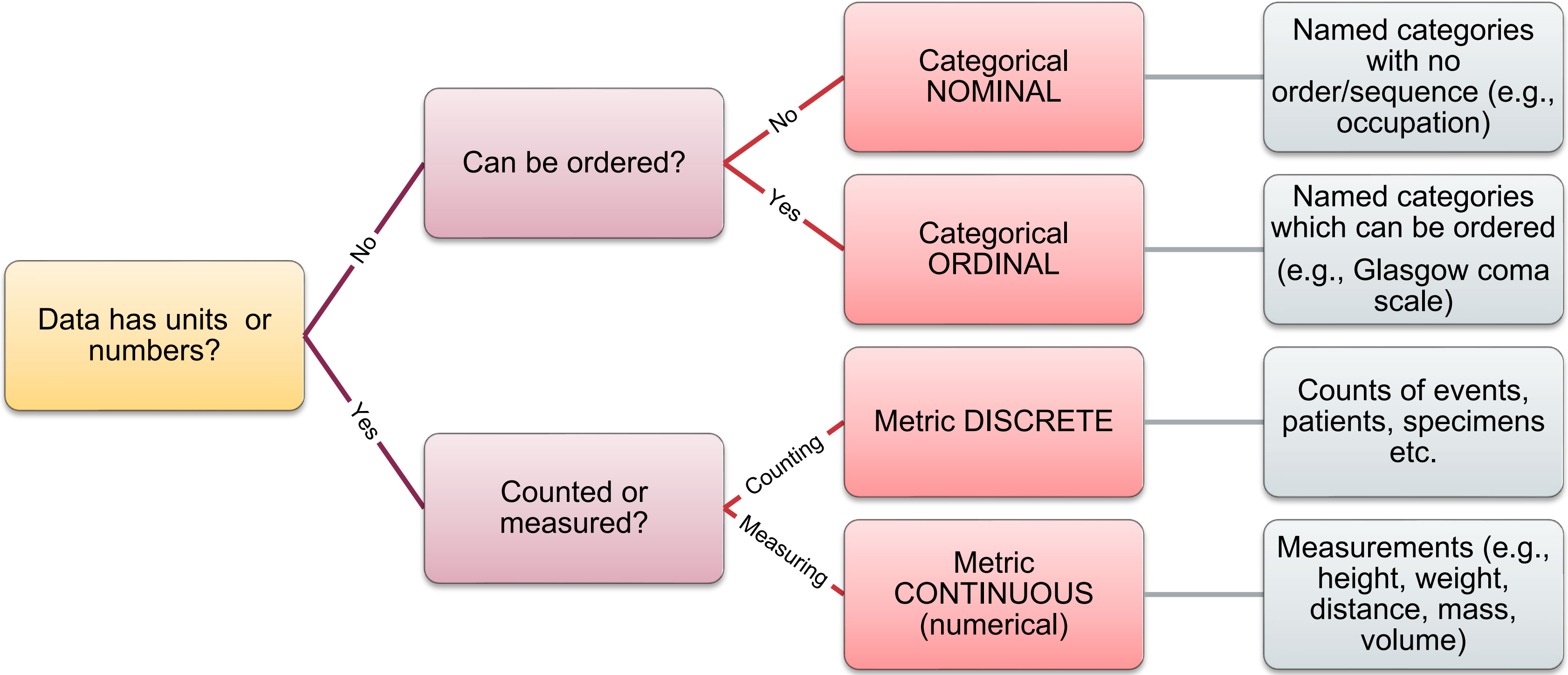
- Hoyt R, Muenchen R. *Introduction to Biomedical Data Science*. Lulu.com, 2019. https://www.informaticseducation.org/about-the-textbook.
- Bowers D. *Medical Statistics from Scratch: an Introduction for Health Professionals.* Hoboken NJ: Wiley Blackwell, 2014.
- Callaway J. *Machine Learning: the Ultimate Guide.* Columbia, SC: 2021. ISBN 9798750222902.
- Burkov A. *The Hundred-Page Machine Learning Book.* 2019. ISBN 978-1-9995795-0-0.

Key Readings

- https://www.nist.gov/project-category/materials-genome-initiative-mgi/machine-learning-ai
- https://www.nist.gov/artificial-intelligence
- https://towardsdatascience.com/which-machine-learning-model-to-use-db5fdf37f3dd
- https://blogs.sas.com/content/subconsciousmusings/2020/12/09/machine-learning-algorithm-use/

Supplemental Material

Statistics – variables



Data has units or numbers?

No

Can be ordered?

No

Categorical NOMINAL

Named categories with no order/sequence (e.g., occupation)

Yes

Categorical ORDINAL

Named categories which can be ordered

(e.g., Glasgow coma scale)

Yes

Counted or measured?

Counting

Metric DISCRETE

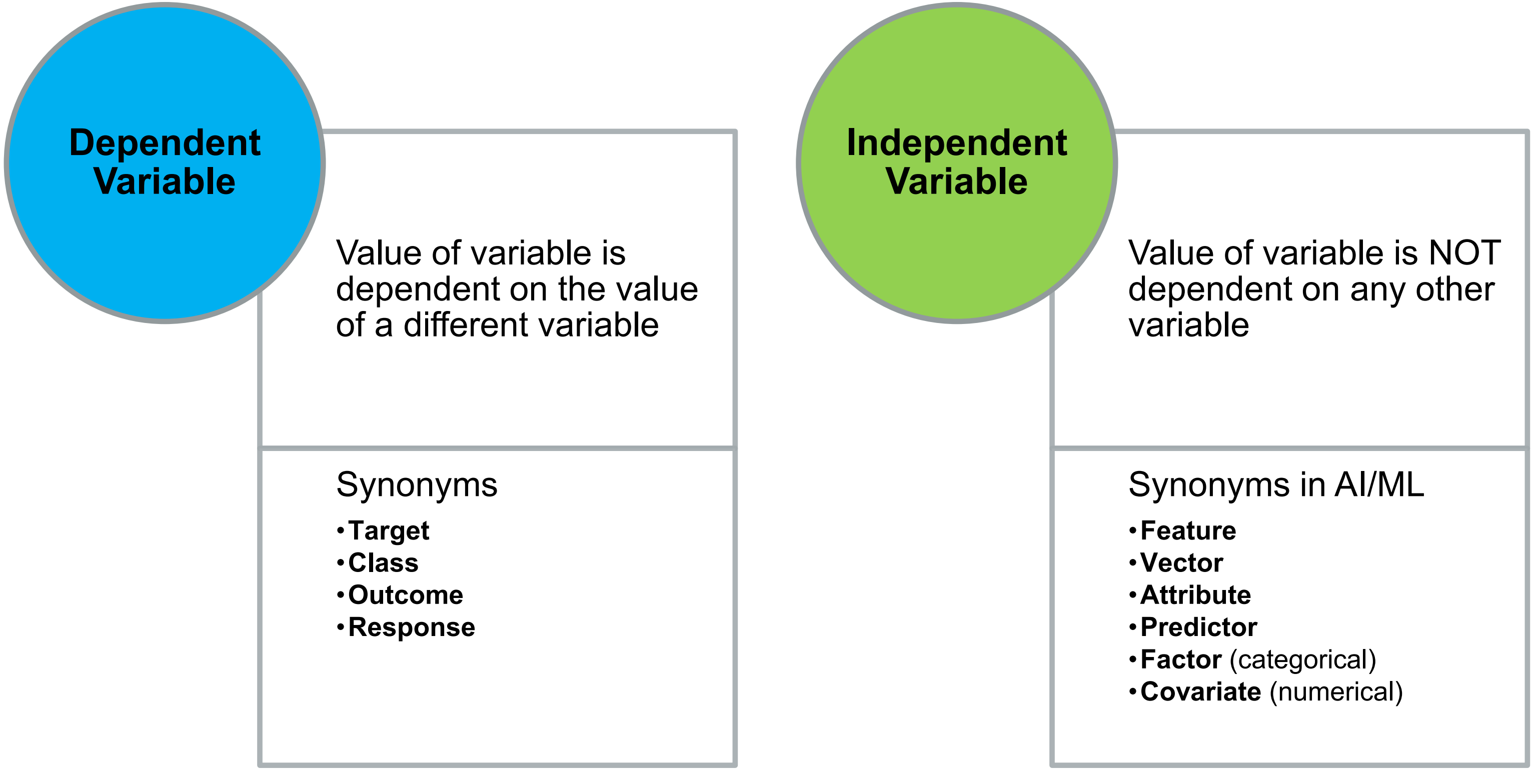
Counts of events, patients, specimens etc.

Measuring

Metric CONTINUOUS (numerical)

Measurements (e.g., height, weight, distance, mass, volume)

ML Definitions – Data



Dependent Variable

Value of variable is dependent on the value of a different variable

Synonyms

- **Target**
- **Class**
- **Outcome**
- **Response**

Independent Variable

Value of variable is NOT dependent on any other variable

Synonyms in AI/ML

- **Feature**
- **Vector**
- **Attribute**
- **Predictor**
- **Factor** (categorical)
- **Covariate** (numerical)

Statistics – causation vs. association

Causation

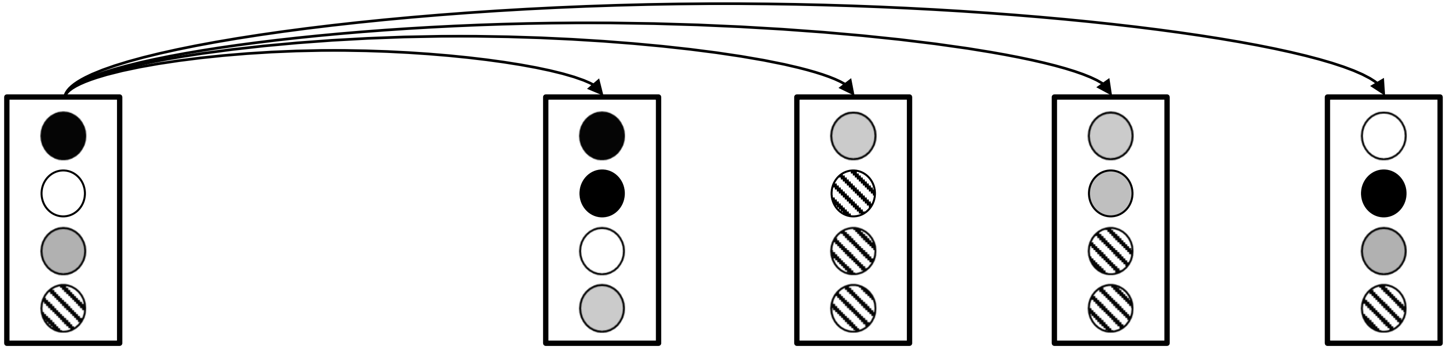
- Cause-and-effect relationship
  - One thing causes another
  - Independent variable causes a particular value in a dependent variable
  - Criteria to satisfy if relationship is causal

Criteria for causation	Description
Chronology (temporality)	Effect has to occur AFTER the cause
Dose-response relationship	Increasing exposure increases the risk
Consistency	Effect is consistent when results are replicated in different settings using different methods
Plausibility	Effect agrees with accepted understanding of medical processes
Coherence	Compatible with existing theory and knowledge (which may be wrong)

Association

- Link between two independent variables

Bootstrapping (random sampling with replacement)

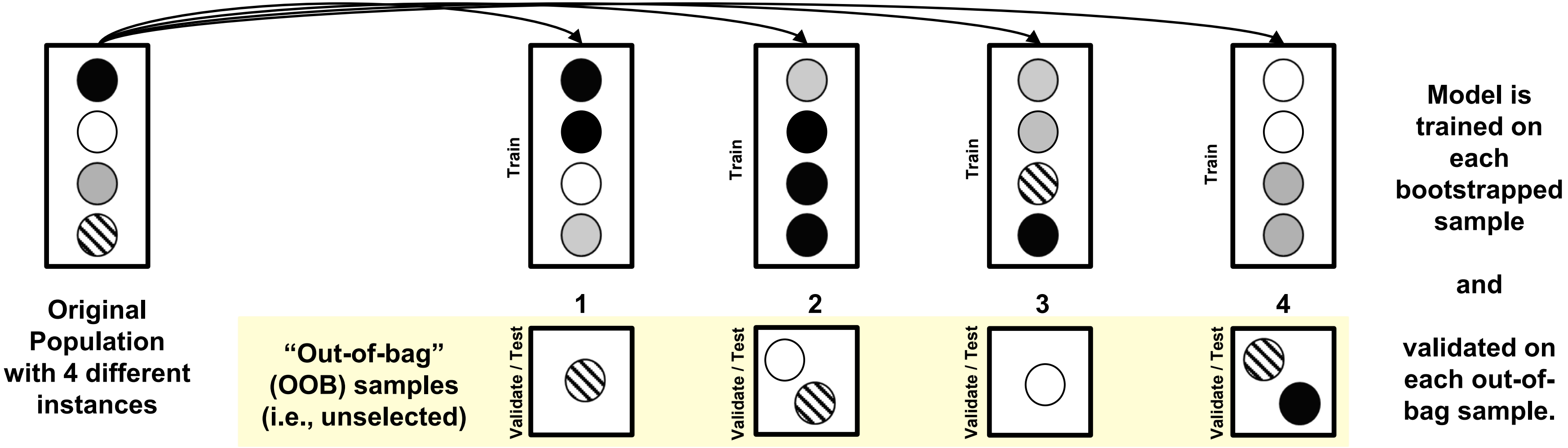


Original Population with 4 different instances

Bootstrapped Samples

- Best used for data where the number of instances is low
- Each sample is drawn and put “in the bag” as if it never left the original population (or as if it was *replaced* each time it was drawn from the hat before drawing again). n = number of instances.
- Chance that an instance is drawn into a bootstrapped sample is 1/n each time an instance is drawn.
    - **Sampling without replacement** (not bootstrapping)**:** instances are removed after sampling and not available to be selected again in the same bootstrapped sample (no duplicates; statistics are different).

Bootstrapping (random sampling with replacement)



Original Population with 4 different instances

"Out-of-bag" (OOB) samples (i.e., unselected)

Model is trained on each bootstrapped sample

and

validated on each out-of-bag sample.

If each instance is unique in the population, then chance that an instance will ***not*** be selected for a single draw is $\frac{(n-1)}{n}$ (in this case, 3/4 or 0.75 or 75% chance).

- Chance that an instance will not be selected in one bootstrapped sample = $\left(\frac{(n-1)}{n}\right)^n$; in this case, $(0.75)^4$ = 0.32 (or 32%).
- As n increases, the chance that a unique instance is not chosen in the bootstrapped sample approaches $\frac{1}{e}$ = 0.368
- Translates to approximately 36.8% instances not chosen for the bootstrapped sample and available for validation or testing.

ML Definitions – Model Evaluation

$$F\ Score = 2\ x\ \frac{(Precision\ x\ Recall)}{(Precision + Recall)}$$

- **F Score**
  - Harmonic mean of precision and recall
  - a.k.a. **F1 score**
  - For binary classifier with no weights on importance of precision vs. recall
    - Perfect F score = 1
    - Increasing precision decreases recall and vice versa
  - **F_β score**: allows weighting of importance of precision vs. recall
    - If recall is 2x as important as precision, then $\beta = 2$
    - F_β score = F score when $\beta = 1$

$$F_\beta\ Score = (1 + \text{⍰}2)\ x\ \frac{(Precision\ x\ Recall)}{(\beta^2\ x\ Precision) + Recall)}$$

  - For multi-class evaluations
    - Compute the F1 score for each class alone then…
    - **Macro F1**: average all the per-class precisions and average all the per-class recalls and then use averages to compute overall F1 score
    - **Weighted F1 score**: Same as macro F1 except that each class is factored by a weight
  - Often used to compare classifiers, but some believe this is often used incorrectly

Regularization Methods

- Mathematical methods
  - **Least Absolute Shrinkage and Selection Operator (LASSO) (L1) regularization**
    - Modifies objective function by adding penalty hyperparameter (e.g., C)
    - If C=0, then hyperparameter has *no effect* on the algorithm
    - As value of C increases above zero, more parameters forced to be set to zero to minimize the objective
      - Can eliminate less essential features and produce a sparse model
      - Helps to increase model explainability by showing which features are essential
  - **Ridge (L2) regularization**
    - Does *not* reduce features but gives some features less weight
    - Better than L1 for maximizing performance of model on hold-out data
    - Is differentiable, so gradient descent can be used for optimizing the objective function
  - **Elastic Net regularization (combo of L1 and L2)**
  - Special regularization for neural networks (Weight decay, Dropout, Batch normalization)
- Non-mathematical methods (Early stopping, Computational data augmentation, Pruning decision trees)

Feature Engineering

- Process of transforming raw data into a data set
- Transform categorical variables into metric continuous (numeric)
  - **One-hot encoding**
    - Transform categories into an array of binary switches, one item per categories
    - Adds dimensionality to features (complexity)
  - Map ordinal values to numbers
  - Map categorical value to its statistic (e.g., mean)
- Transform continuous metric (numeric) variables to categorical
  - **Discretization** (a.k.a. **binning** / **bucketing**)
    - Typically done by converting data based on numeric range into differently named bins or buckets
- **Normalization**: convert actual range into normalized range
- **Standardization**: rescale feature values to a normal distribution

Melanoma = [1,0,0]
Dysplastic nevus = [0,1,0]
Benign nevus = [0,0,1]

Clustering Methods – Hierarchical Quality Metrics

- **Ward method (ward linkage)**
  - Pair of clusters merged when smallest increase in within-cluster variability
  - Measured by…
    - Determining cluster center (centroid) then…
    - Computing squared distances between each cluster and cluster center
    - Subtracting squared distance of cluster A from squared distance of cluster B
    - Clusters with lowest sum of squared errors (SSE) are paired
- **Average linkage**
- **Complete linkage**
  - similarity of the furthest pair
  - Outliers may cause merging of close groups later than is optimal
- **Single linkage**
  - Similarity of the closest pair
  - Can cause premature merging
- **Centroid similarity**
  - Each iteration mergers clusters with the foremost similar point