MAYO CLINIC

# How I Learned to Stop Worrying and Love the Medical AI Hype

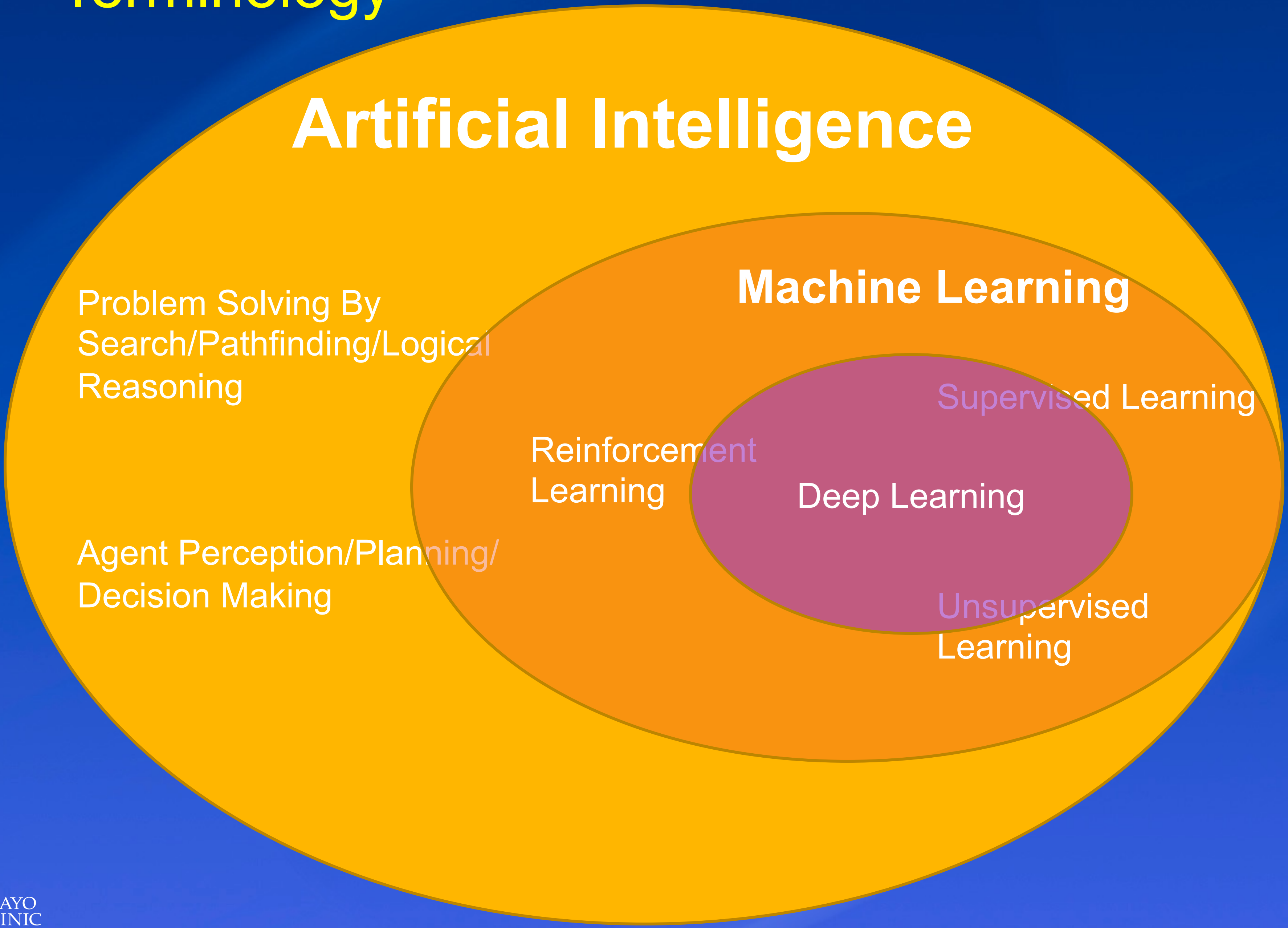
Garrett Jenkinson, PhD
Data Scientist, Assistant Professor of Biomedical Informatics
Division of Computational Biology
Department of Quantitative Health Sciences
Mayo Clinic

## Outline

- AI/ML 101
  - Terminology
  - Successes and Excellent Reasons for Hype
- Lessons Along the Way for ML in Medical Practice
  - Garbage in/Garbage Out
  - Use Prior Knowledge
  - Curse of Dimensionality
  - Data Leakage
  - Interpretability and Complex Decision Boundaries
  - Objective Function Misalignment, Class Imbalance
  - Association versus Causation
  - Use decision-theoretic thinking
  - Fairness and Calibration
  - Out of Distribution Predictions

MAYO CLINIC

## Terminology

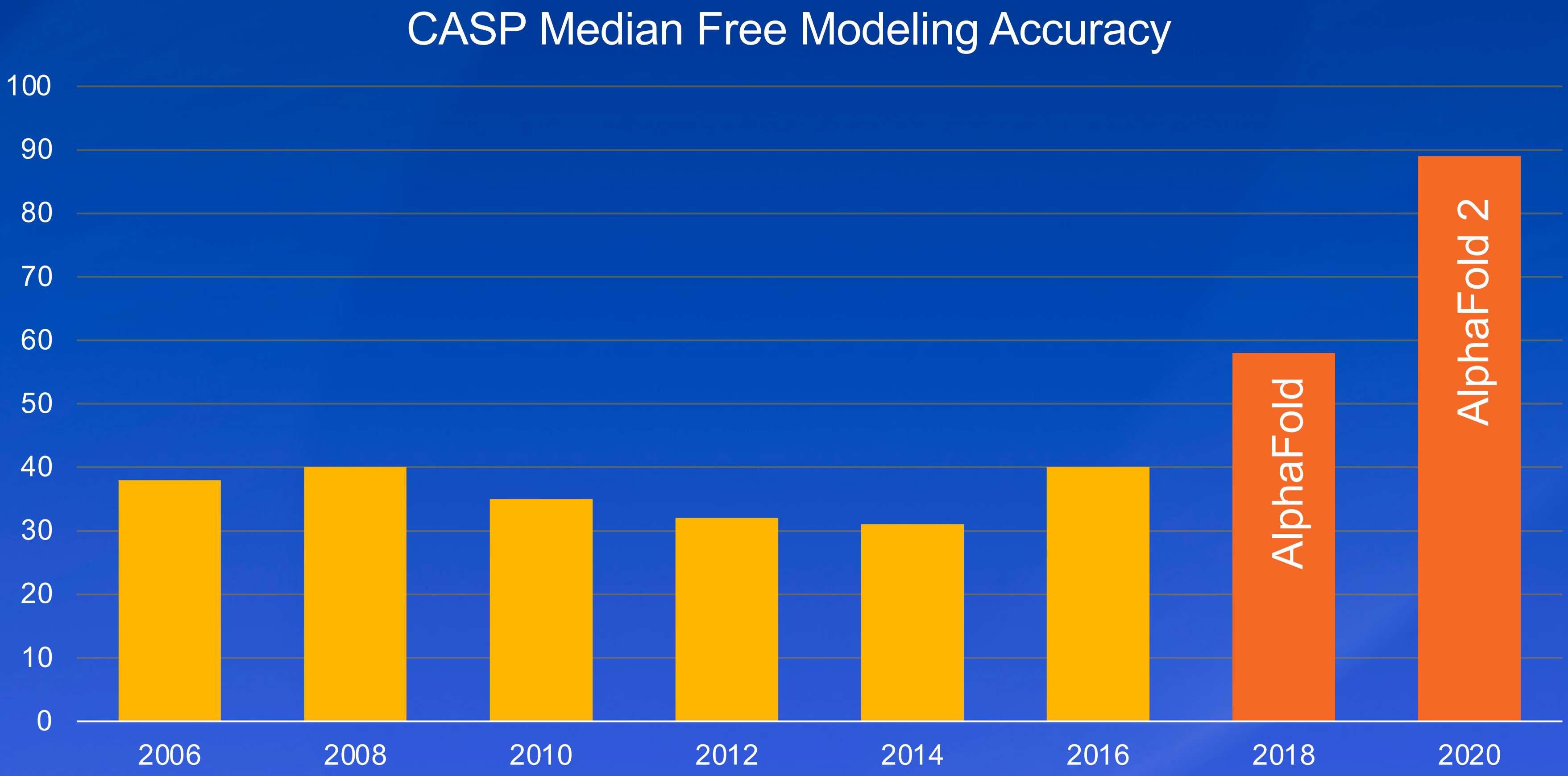


MAYO CLINIC

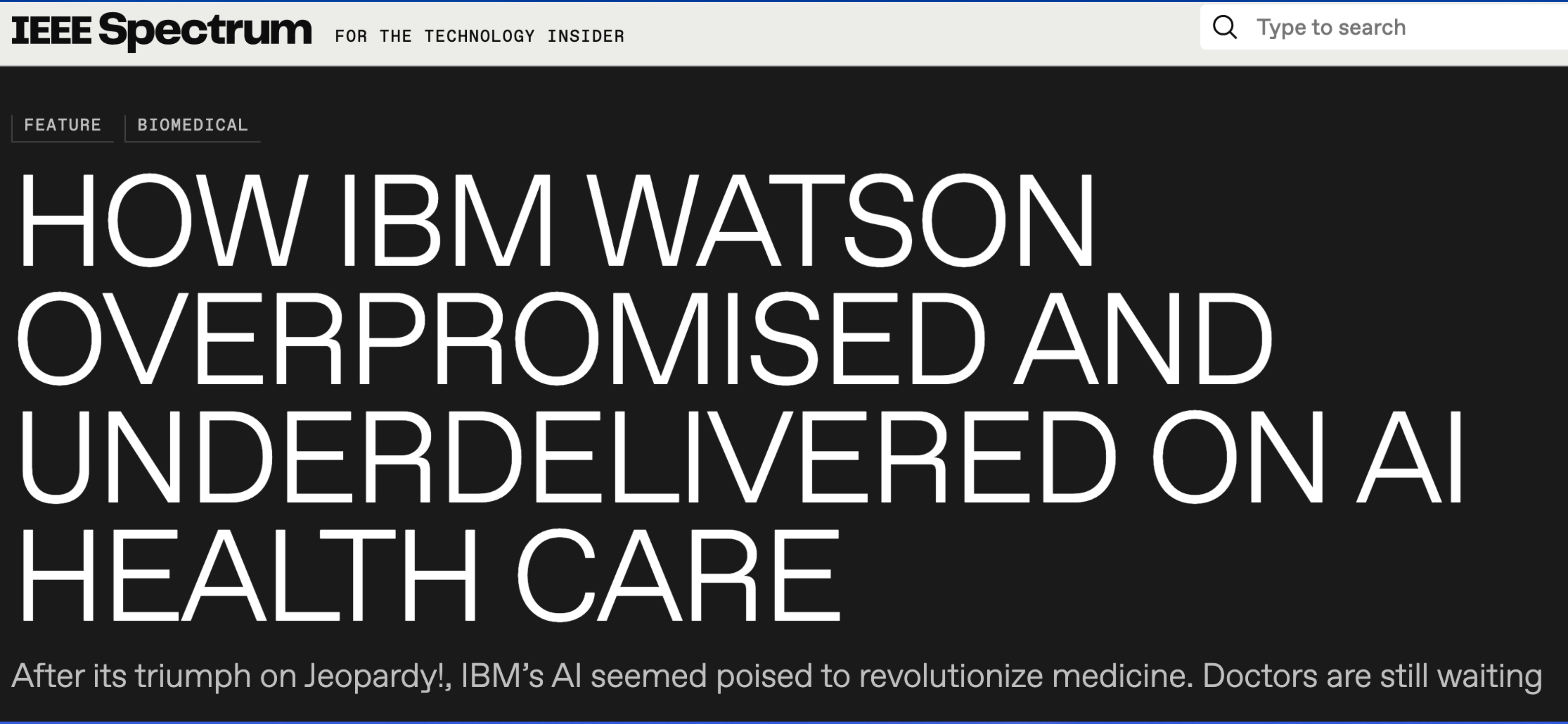
## Excellent Reasons For Hype

- High profile super-human performance systems
  - IBM DeepBlue Chess, 1997
  - IBM Watson Jeopardy, 2011
  - Hinton, ImageNet Classification 2012-
  - AlphaGO, 2016
  - Poker, Pac-Man, Quake3, Dota2, StarCraft2, Atari, speech recognition, skin cancer detection, prostate cancer detection, diabetic retinopathy, machine translation
- Self-driving cars, factory robots
- Natural language processing, GPT3, codex

## AlphaFold achieves near experimental accuracy



## Care, diligence and acknowledgement of unique challenges are required


IEEE Spectrum FOR THE TECHNOLOGY INSIDER

Type to search

FEATURE | BIOMEDICAL

HOW IBM WATSON OVERPROMISED AND UNDERDELIVERED ON AI HEALTH CARE

After its triumph on Jeopardy!, IBM's AI seemed poised to revolutionize medicine. Doctors are still waiting

MAYO CLINIC

## Probabilistic View of Supervised Learning

- We have input features $\boldsymbol{x} \in \mathbb{R}^d$ (numerical descriptions of examples)
  - $\boldsymbol{x}$ = [185, 70], representing weight/height
- We have output labels $y \in \mathbb{R}$ (numerical outcomes, sometimes multidimensional as well)
  - $y$ =0 or 1, representing diabetes diagnosis
  - $y$ = A1c measurement
- We seek a function $\mathrm{f}(\cdot)$ such that $\mathrm{f}(\boldsymbol{x}) \simeq y$
  - Machine Learning/Deep Learning are tools to find this function
  - It is helpful to view this as coming from estimating $p(y|x)$ and then picking $y$

## Example feature vectors $x$

- a greyscale 256x256 image where each pixel takes value between 0 and 255, and d = 256 × 256 = 65,536
- a color image with 256x256 pixels and r, g, b "channels" making 256x256x3 array of numbers between 0 and 255 and
  d = 256 × 256 × 3 = 196,608
- a non-negative count vector of 10,000 genes measured by RNA-seq in blood with d=10,000
- a vector of estimated probabilities in the range [0, 1] of methylation at d = 750,000 CpG sites in the genome

## Example labels

- Whether the patient is healthy (0) or has cancer (1)
- Whether this DNA variant causes outlier expression (1) or not (0)
- Whether this patient will have a severe reaction (1) or not to COVID (0)
- The number of COVID patients entering the ER tomorrow
- Variant pathogenicity {B, LB, VUS, LP, P}
- Pixel position of LL and UR corners of bounding box around a tumor in a chest x-ray

## Example functions

- If weight/height > 3, predict diabetes (1), else predict no diabetes (0)
  - This is example of a decision tree (using an augmented/crossed feature)
- Deep neural networks, random forests, logistic regression, KNN, SVM, etc, are all just algorithms to take training data in and produce concrete calculation representing $\mathrm{f}(\boldsymbol{x}) \simeq y$
  - Conceptually no different from above decision tree

MAYO CLINIC

## Bayes Rule & Bayes Error

- The Bayes Rule is the best function $\mathrm{f}(\cdot)$ possible which has performance of Bayes Error
- "Best" means we need a loss function to evaluate if $\mathrm{f}(\boldsymbol{x}) \simeq y$ numerically
  - 0/1 loss for classification
  - $(\mathrm{f}(\boldsymbol{x}) - y)^2$ for regression
- The Bayes rule for 0/1 loss in binary classes:

$$f(\boldsymbol{x}) = \begin{cases} 1, & \text{if } p(y = 1|\boldsymbol{x}) > \frac{1}{2} \\ 0, & \text{otherwise} \end{cases}$$

## Lesson: Garbage in, Garbage Out

- The Bayes Error tells us the best we can do predicting y from measurements x
- If x is number of ChrY copies in each cell of fetus, Bayes Error for sexing fetus is near 0
    - $p(y=1|x=0) \simeq 0$, $p(y=1|x>0) \simeq 1$
- If x is WGS of mother and father, Bayes error for sexing fetus is near 0.5
    - $p(y=1|x) \simeq 0.5$ for all x
- Human-level performance can be rough proxy in some cases for Bayes Error
    - If human case is hopeless, think hard first

## Lesson: Focus More on the Data

- Often data cleaning, collection, representation will improve your performance much faster than overly focusing on ML algorithm/methods
  - AutoML tools like AutoGluon rapidly test and combine many cutting-edge tools
  - If AutoML fails catastrophically, perhaps Bayes Error is high or data integrity/representation is poor
- If no opportunity to improve data, then focus on modeling assumptions and selecting best suited algorithms

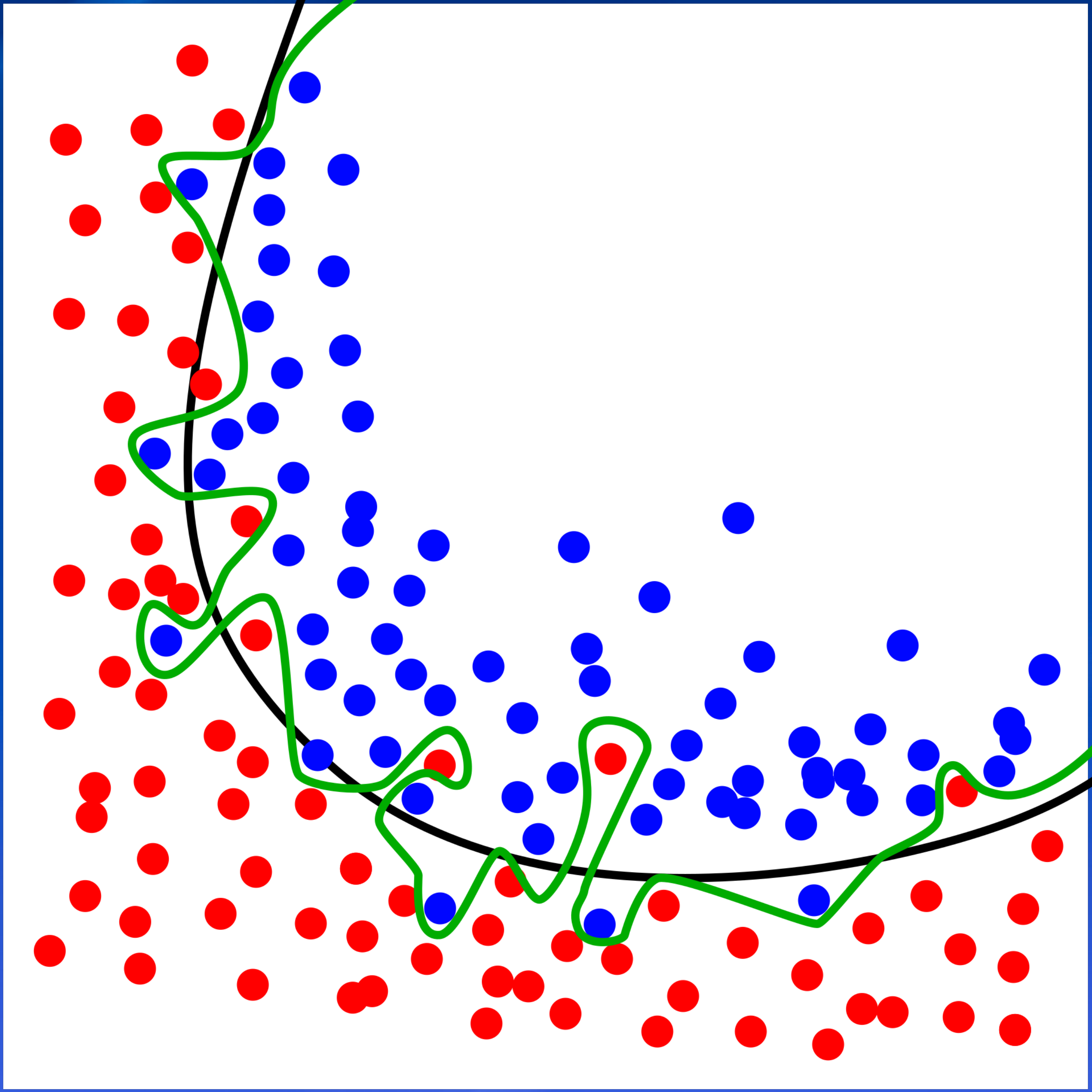
MAYO CLINIC

## We learn from limited examples

- We have data pairs $(\boldsymbol{x}_n, y_n), n = 1, \ldots, N$
- From these we need to estimate $p(y|\boldsymbol{x})$ or $f(\boldsymbol{x})$
- If Bayes rule $f(\boldsymbol{x})$ is simple, we can use lower *N*
  - One male and one female example could train the Bayes classifier in sex pred from ChrY per cell count
  - If $\boldsymbol{x}$ is all sensor measurements in car and $y$ is gas/brake pressure and steering angle, much larger *N* needed

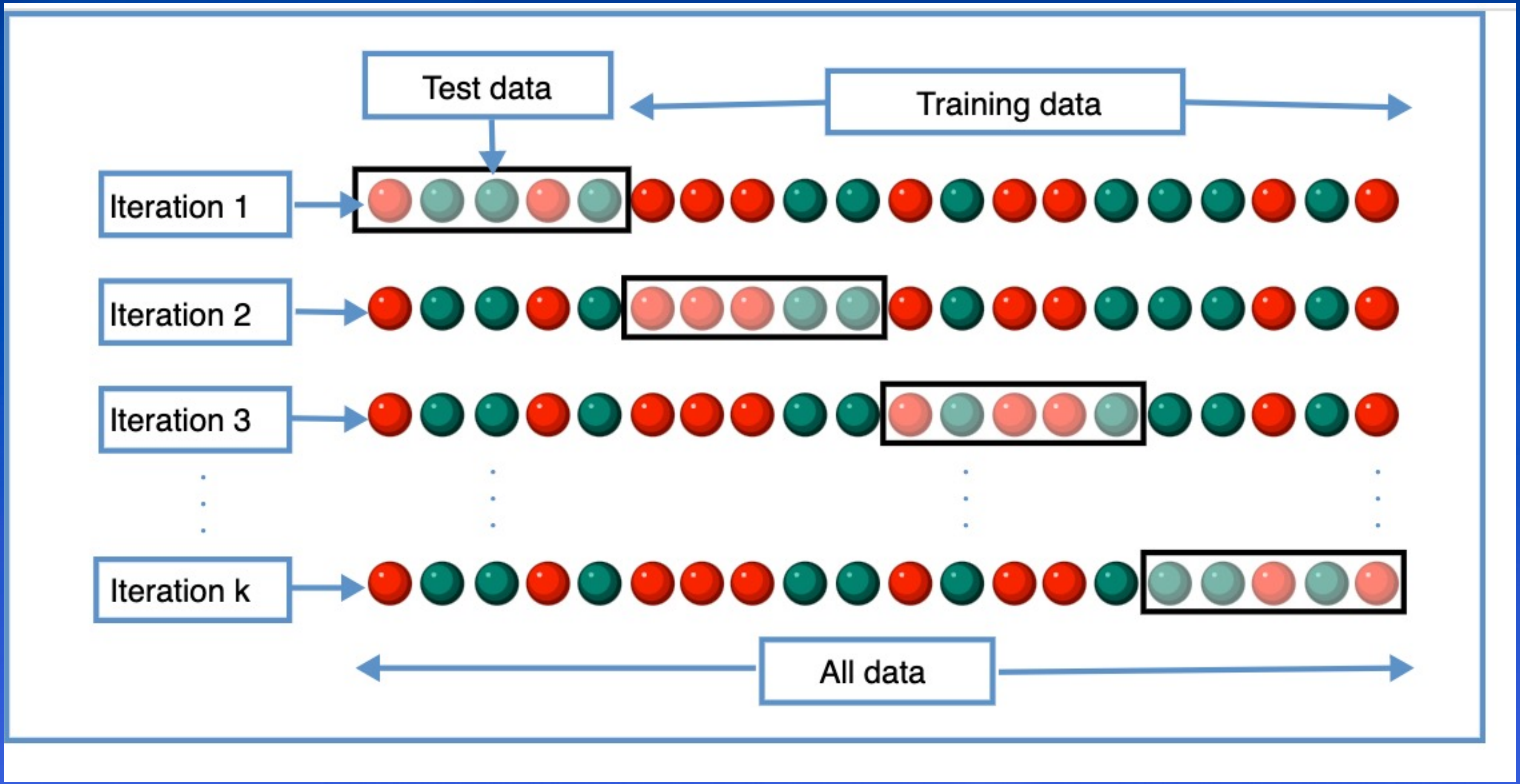
## Overfitting

- Complex decision boundaries fit “noise”
- Higher error on unseen test data



https://en.wikipedia.org/wiki/Overfitting#/media/File:Overfitting.svg

## Cross-validation



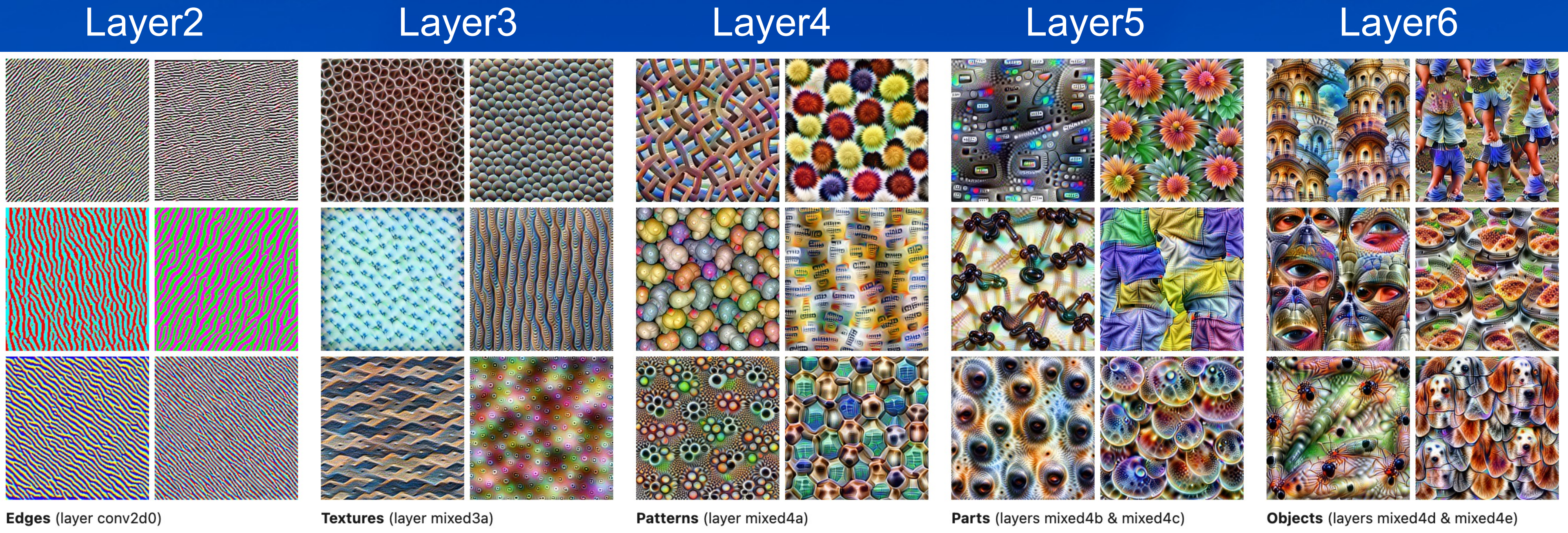
https://upload.wikimedia.org/wikipedia/commons/b/b5/K-fold_cross_validation_EN.svg

## Lesson: Regularization/Occam's Razor

- Simple answers require less evidence/data
- Complicated answers require greater evidence
- L1&L2 penalties/Dropout/Shrinkage/Bayesian methods can help
  - Cross-validation commonly used to evaluate over-fitting and tune regularization hyperparams/priors
- Data Augmentation (e.g., jittering bootstrap, image manipulation, language rearrangements) can add robustness to common but irrelevant differences

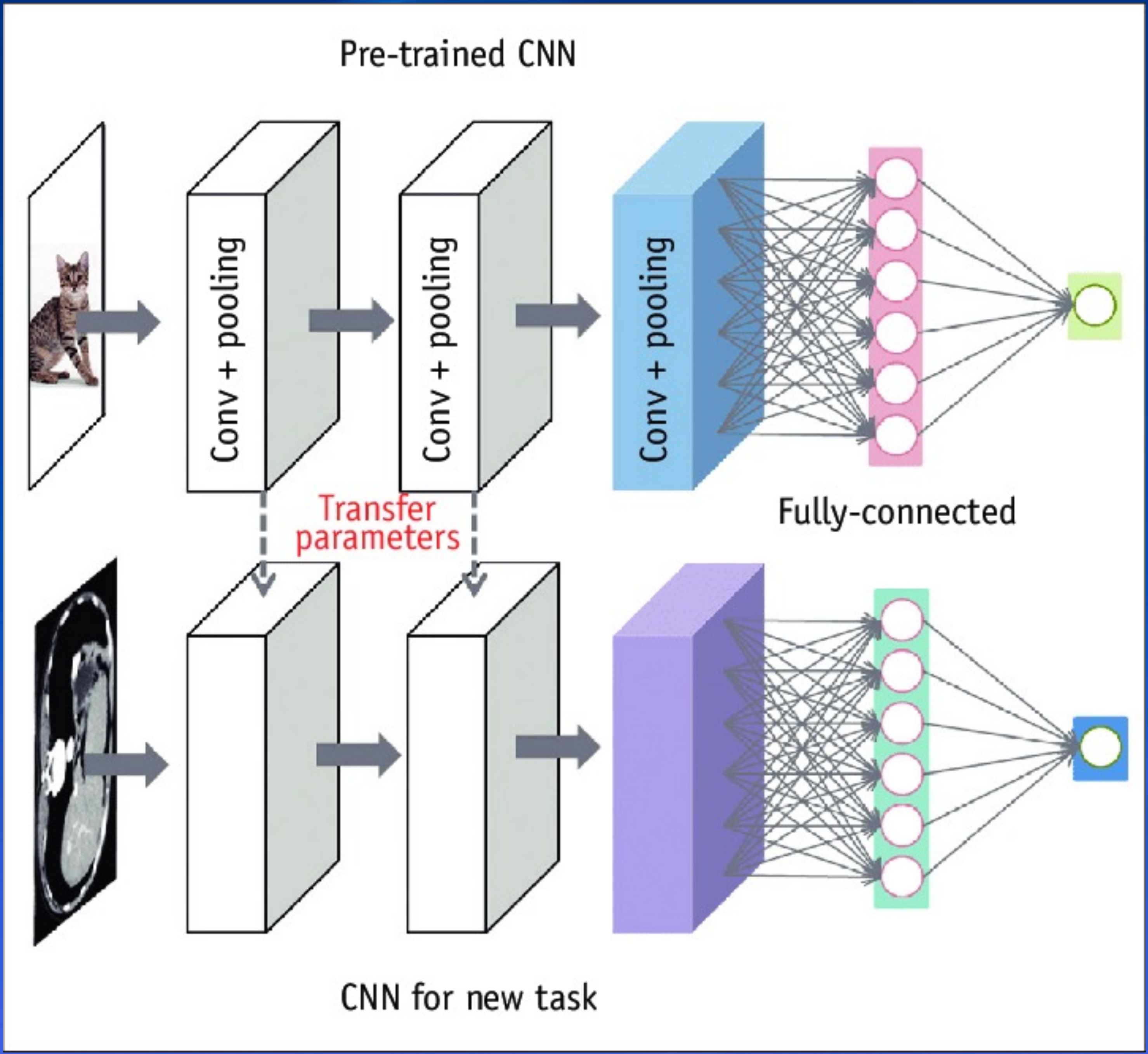
# Why "Deep" Learning

- Each layer is its own ensemble of learners
- Intuitively layers extract features, then combine them in increasingly sophisticated ways
- Greater abstraction deeper in network



Layer2
Layer3
Layer4
Layer5
Layer6
**Edges** (layer conv2d0)
**Textures** (layer mixed3a)
**Patterns** (layer mixed4a)
**Parts** (layers mixed4b & mixed4c)
**Objects** (layers mixed4d & mixed4e)

Figure from Olah, et al., "Feature Visualization", Distill, 2017.



https://doi.org/10.3348/kjr.2019.0312

## Lesson: Use prior knowledge

- Transfer learning
  - Great for NLP or images where huge datasets available for pretraining
  - Don’t train a huge CNN from scratch with 100’s of medical images
  - Don’t train a huge transformer from scratch in 10K clinical notes
- Bayesian methods can be even more rigorous when good prior data is available, e.g., in lab tests

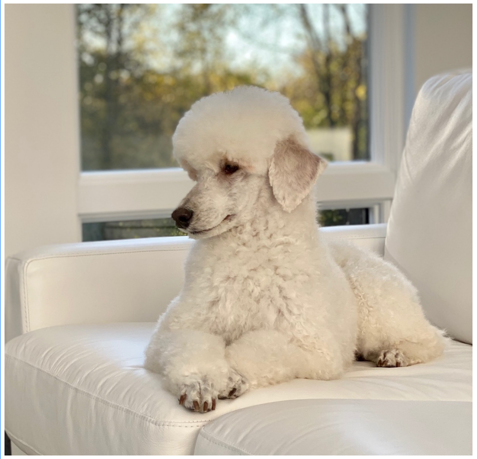
MAYO CLINIC

## Curse of dimensionality

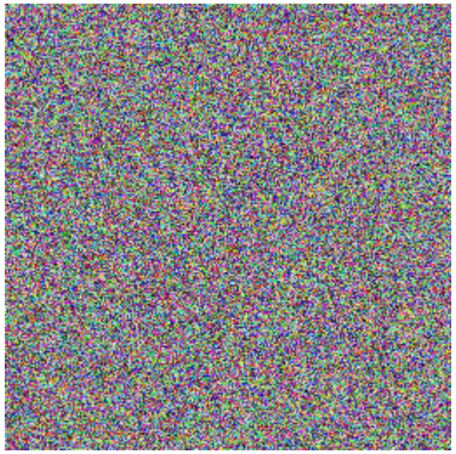
- Volume of d-dimensional box grows exponentially
  - Observing 10-point grid requires $10^d$ observations, e.g. to fit $p(y|\boldsymbol{x})$ over all $\boldsymbol{x}$
- High-dimensions not intuitive
  - Volume almost entirely on outer-shell/veneer
  - You are an extremist, in high enough dim
- Data not uniform in feature space
  - Lives on low-dimensional “manifold”
  - Most randomly generated images look like TV fuzz, not kittens

MAYO CLINIC

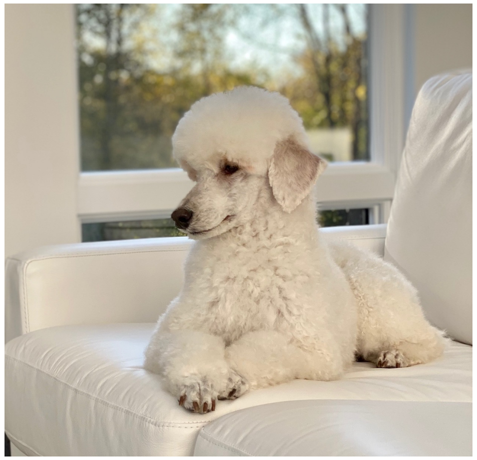
## Lesson: Model complexity/dim and lack of interpretability can hide overfitting



+ 0.005 x



=



"Poodle"
72% confidence

"nematode"
4% confidence

"tennis ball"
98% confidence

- Interpretability a thorny subject
  - Use "simple" models where possible
  - Understand risks of black boxes, SHAP/etc not global explainer

## Data Leakage

- We mentioned cross-validation, but generally we also want hold out test set
    - Evaluates generalization performance
- Data leakage refers to information from a test or validation set entering the model fitting procedure
    - If $\mathrm{f}(x)$ was developed with any knowledge from test set, the evaluation is optimistic/corrupted

## Examples of data leakage

- You plotted/inspected all your data before model building/fitting
- You scaled your features before you did the data split
- You produced a “Table 1” prior to model building
- You did PCA/umap/etc for dimensionality reduction or manifold learning using all data
- You collected some data, worked with it, collected more and re-split randomly
- Your DNA variant impacts same codon but you split by DNA base position

## Lesson: Take ML study design seriously

- Very first step of ML project is designing your train, validation, test splits up front
- Stratified split your test set off and zip the data
  - Only unzip when **FINAL** model selected and tuned
  - You get 1 shot only. Cannot go back and tweak/tune hyperparams, try another model, etc
- Using cross-validation in non-test set data costs computations but can be very effective for model selection and hyper parameter tuning

MAYO CLINIC

## Model training and evaluation require objectives

- ML does not know what you want, only the loss function you provide it to minimize
- The training loss function is not the only metric you should look at
- Think of ML as a cursed monkey paw that grants wishes in easiest and often worst way possible
  - “I wish to be richest person on earth”
  - ML: “Done. I have killed all other people”
- Objective function misalignment is the core of sci-fi AI gone wrong, but is very real problem

MAYO CLINIC

## Lesson: Beware of misalignment

- Made up example: train a model for maximal accuracy in Sickle Cell Disease
- Take last 50k patients seen in Midwest clinic and train model with some lab measures
- Model gets 99.9% accuracy!
- Always says no SCD because only 50 patients had illness and labs mostly imputed/uninformative
- Add extensive EHR data, use AUROC and recall to evaluate to ensure catching of cases
- Get 0.97 AUROC, .96 recall with logistic regression

| | Not Black or African American | Black or African American |
|---|---|---|
| No SCD | 49,000 | 1,000 |
| Has SCD | 2 | 48 |

MAYO CLINIC

## Label/feature leakage is prominent form of misalignment

- Pathology slides labeled pathogenic with high accuracy, precision, recall, AUROC
  - ML learned that pathologist put arrows pointing to malignant features and just looks for arrows
- Radiology DICOM (images and metadata/demographics) and it diagnoses case/control accurately
  - Metadata contains information related to case versus control and was not stripped from dataset
- CNN knows race from chest xray…last slide revisited

## Lesson: Do an error analysis

- Generate a confusion matrix

| Truth\Prediction | Class 0 | Class1 |
|---|---|---|
| Class 0 | # TN | # FP |
| Class 1 | # FN | # TP |

- Randomly examine ~dozens of cases from each quadrant TP, FP, FN, TP
- Assume the algorithm is a cheater, and try to find out how it cheated
- Use local explanation and/or counterfactual algorithms to try to understand why each case landed in their quadrant of the confusion matrix

## Causality is hard

- Most ML systems will be doing associational predictions and not causal ones
- This can lead to cheating as discussed before (camels are on sand, cows on grass)
- This can lead to poor decision making based on model outputs
  - COVID-19 mortality model has NPV 99.8% and PPV 70%, how to use?
  - NPV is high, triage and send them home!
  - This is causal inference assertion. NPV that high only *when getting full clinical support.* Not same as if *removing* support

MAYO CLINIC

## Lesson: use causal judgements with care

- Increasing literature around causal ML algorithms
  - All observational causal inference is based on assumptions that may not hold
- Where possible use ML models (even “causal” ones) in way that would be safe under associational interpretations
  - In COVID-19 model, consider alert system that only adds oversight/care and does not remove it

MAYO CLINIC

## Levels of difficulty/data requirements increase as outputs become complex

- Roughly, in increasing orders of "difficulty":
  - Binary classification
  - Multiple ordinal classes (think: low, medium, high)
  - Multiple categorical classes (think: lung, liver, spleen)
  - Univariate regression (think: A1c levels)
  - Multivariate regression (think: transcriptome expression levels)
  - Univariate functional/density estimation
  - Multivariate functional/density estimation

MAYO CLINIC

## Taking a step back and see big picture

- Easy to get caught up in building a great ML tool, but think about how it fits into process
- Example: “We built sophisticated regression to determine amount of contamination in sample”
  - How will lab use? Probably only a few choices such as: pass sample, or fail sample and re-analyze.
  - Would binary classification make more sense?
  - Will it be automated or human-in-the-loop?
  - What are practical costs/benefits of deciding to pass versus fail? How to tune algorithm?

MAYO CLINIC

## Decision theoretic framework

- Choice among actions $a \in \mathcal{A}$ from an action space, e.g.:
  - $\mathcal{A}$ = {"re-sequence", "proceed with current data"}
  - $\mathcal{A}$ = {"give chemo", "do surgery/radiation", "wait and see"}
- (Unknown) state of nature: $\theta \in \Theta$
  - $\Theta$ = {"sample contaminated", "sample uncontaminated"}
  - $\Theta$ = {"aggressive tumor", "benign tumor"}
- Loss/utility function- $l(a, \theta) \geq 0$

## Decision Theory, Continued

- Decision procedure/rule from features to actions: $\mathrm{f}(\boldsymbol{x}) \in \mathcal{A}$
- Risk function $R(\theta, \mathrm{f}) = E_\theta\, l(\theta, \mathrm{f}(\boldsymbol{x}))$
- Good rules $\mathrm{f}(\cdot)$ will minimize the risk function
- Need to quantify your loss table (hard!), and combine with confusion matrix (easy!) to assess operating points/decision rules:

| **Loss Table. a = cols $\theta$ = rows** | **Use current data** | **Re-sequence samples** |
|---|---|---|
| No contamination | 0 | \$ to resequence + \$-value of delayed results |
| Contamination | \$-value of potential medical error from contamination | \$ to resequence + \$-value of delayed results |

MAYO CLINIC

## Lesson: use decision theoretic thinking

- Often you will not be able to “pin down” a loss
- Still a valuable exercise to get stakeholders thinking about overall process
  - Thinking about an “action space” can help frame ML task (e.g., regression vs classification)
  - Loss/cost differential of FP versus FN
  - Often highly imbalanced in medicine, need to consider because operating point on ROC or precision/recall curves usually will not be at FP=FN
- If human-in-loop, action can be “flag for manual review”…may not have automated positive actions

MAYO CLINIC

## Lesson: Use Model Calibration or Fairness Modifications to Complex Models

- When a complex model predicts class probabilities they are often uncalibrated
  - The probability is not accurate except in selecting class with highest probability
  - Often probabilities will all be very near 0 or 1, making models appear “over confident”
- If used in human-in-loop decision support context, important to calibrate a model after training it
- “Fairness measures” are at odds with a well-calibrated model so one may need to choose between them [Pleiss *et al* 2017]

MAYO CLINIC

# Production data may look different from training (and testing!) data

- "Out of distribution" refers to systematic changes to $p(\boldsymbol{x})$ or $p(y|\boldsymbol{x})$ in the real setting
- $p(\boldsymbol{x})$ would change if demographics at deployed hospital different than hospital model trained at
- $p(y|\boldsymbol{x})$ could change if, e.g., a new variant made COVID more lethal, perhaps in specific cohort, as compared to time of training data
- Out of distribution detection might label individual $\boldsymbol{x}$ as outliers from training data, which could warn about model uncertainty
- Be especially careful with synthetic/augmented data

MAYO CLINIC

## Lesson: Quantify Uncertainty and Monitor Models Deployed

- If you train a neural network with dropout for regularization (a good idea!) you should use “Monte Carlo Dropout”
  - Easy to implement (~1 line of code), increased accuracy, built in uncertainty estimates
- Otherwise consider more sophisticated Bayesian method or other tools that can let you know when they are unsure
- Monitor live models for “drift” (e.g., increased calling of positive class compared to train set)

MAYO CLINIC

## Review and Bringing It All Together

- Garbage in/Garbage Out
- Use Prior Knowledge
- Curse of Dimensionality
- Data Leakage
- Interpretability and Complex Decision Boundaries
- Objective Function Misalignment, Class Imbalance
- Association versus Causation
- Use decision-theoretic thinking
- Fairness and Calibration
- Out of Distribution Predictions

MAYO CLINIC

## Questions?

MAYO CLINIC