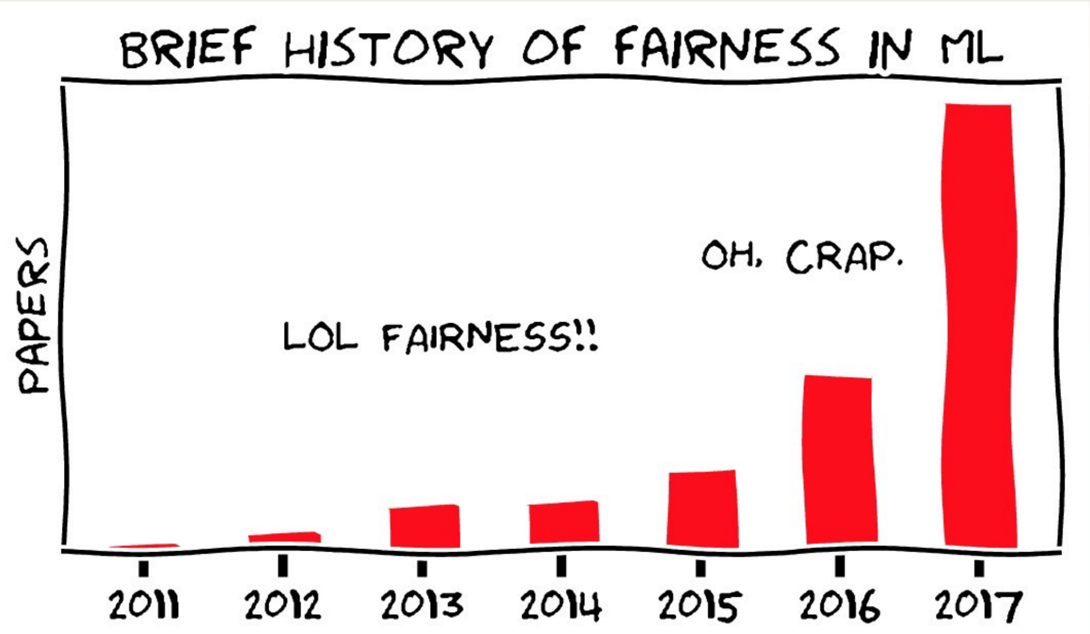
Evaluation: Understanding Bias, Fairness and Error in the Context of Machine Learning

Eric Lofgren MSPH, PhD
Associate Professor
Paul G. Allen School for Global Health
Washington State University

Disclaimer

- Bias and Fairness in Machine Learning is a rapidly evolving and active field – things can and will change
- There are whole long, graduate level classes on this
- Inherently, some of the things I will say here are colored by my own bias
- If you're interested in this topic more, I highly recommend the works of Kristian Lum (Twitter: @KLdivergence) as a starting point



What Do We Mean By "Unbiased"

- There is some process in the world whereby p(O | **Z**) = X, likely with some random noise around it
- A truly unbiased model would, given X and **Z**, predict O with no error that isn't random
- Many people, when they talk about unbiased algorithms, are actually talking about p(O | **Z**) = X *in the data* and the algorithm predicting the relationship that exists *in the data* (more on this later)

What Do We Mean by "Fair"

- There is some process in the world whereby $p(O \mid \mathbf{Z}) = X$, likely with some random noise around it
- Within **Z**, there is some factor, which we'll call A, that a society/government/researchers/field/etc. don't want to have weight in our predictions
- That is, $p(O \mid \mathbf{Z}, A=0) = p(O \mid \mathbf{Z}, A=1)$ for a binary variable
- These variables can be a lot of things, but most commonly we're talking about things that would be considered "protected classes" in other contexts – race, gender, sexual orientation, etc.
- Defining "Fair" can be *hard*
  - It is both *domain* and *feature* specific

Where These Intersect

- It is *very* difficult for a biased algorithm to also be a fair one
- It is entirely possible for an unbiased algorithm to still be unfair
  - p(O | **Z**, A=0) =/= p(O | **Z**, A=1)
- Fairness can be arrived at in several ways, one of which is introducing bias
- This may be okay – the purpose of many ML systems is not raw, unfettered predictive accuracy, even though that's often what we focus on
- These discussions often parallel those we're having in epidemiology, about the appropriateness of certain characteristics being treated as biological variables, modifiability, etc.

The Appeal of "Unbiased" Algorithms

- The internal algorithms people use are terrible, biased, unfair, and difficult to quantify and evaluate
- Formalized procedures can limit how an individual's biases impact decision making
- Couldn't we have a computer, who doesn't care about these things, sort this out for us?
- ML as a peak formalism
- As with everything in Epidemiology, the answer is "It depends…"

Unbiased, Algorithm-based Policing

The Problem

- There's still people
- People who collect the data, evaluate algorithmic performance, implement the models in real-world settings, etc.

Phases of Bias

Data Collection → The Algorithm → Production

The Really Dangerous Part

The First Rule of Bias in ML (According to Me)

- The time to start thinking about bias is as early as possible
  - Ideally before the data is being collected
  - It's definitely not five minutes after you fire up RStudio and start writing code
  - But that's better than five minutes after you click "Submit" in Editorial Manager
- Rare is the algorithmic tool that will let you use math to dig yourself out of a hole you dug with data

Data Collection

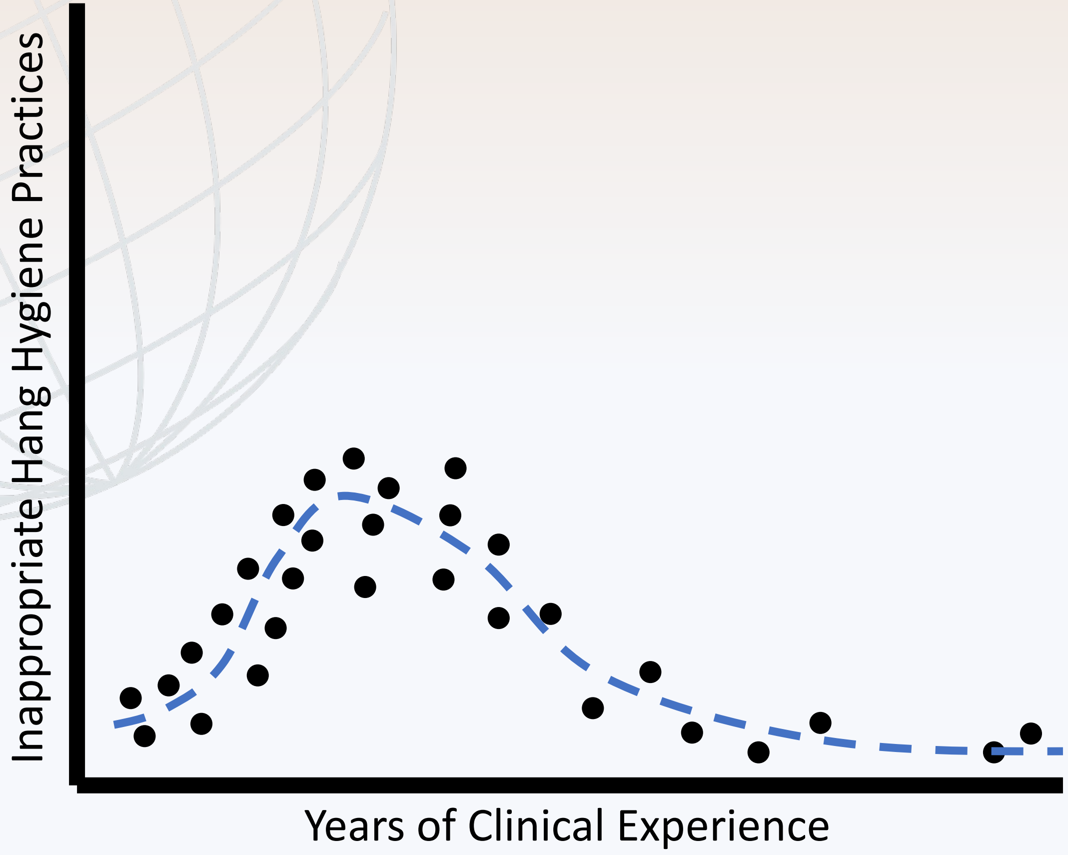
- Sampling and Selection Bias
- Tainted samples
- Limited features and model misspecification
- Inappropriate proxy variables
- Sample size disparity

Example

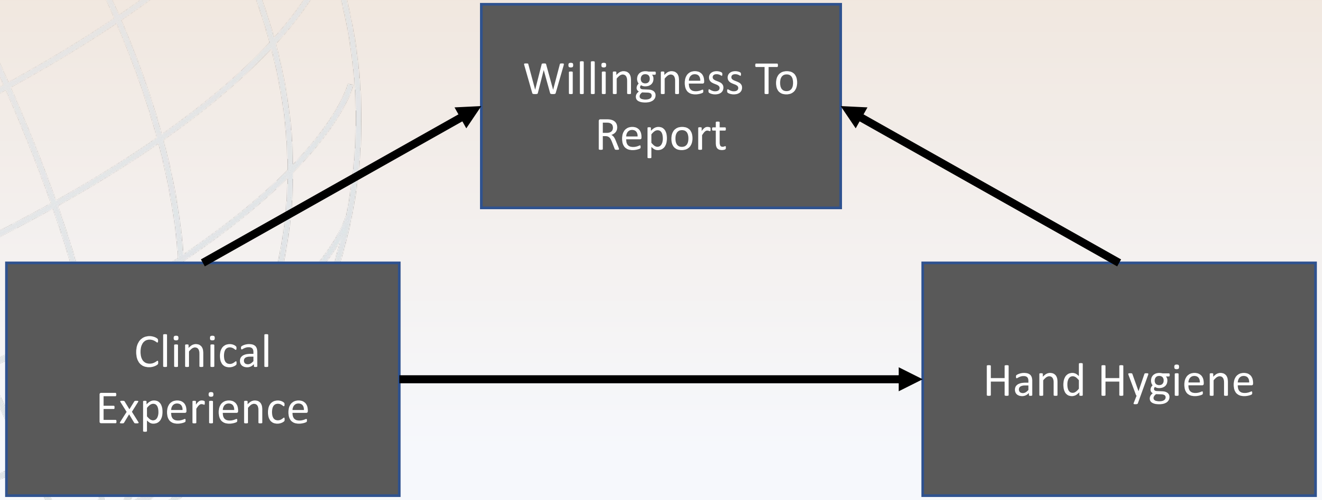
- You’re working on predicting the rate of inappropriate hand hygiene practices based on an anonymous reporting “If You See Something, Say Something”-esq program
- This work will be used to help inform clinician education programs, help infection teams target interventions, etc.

Sampling and Selection Bias

Sampling and Selection Bias



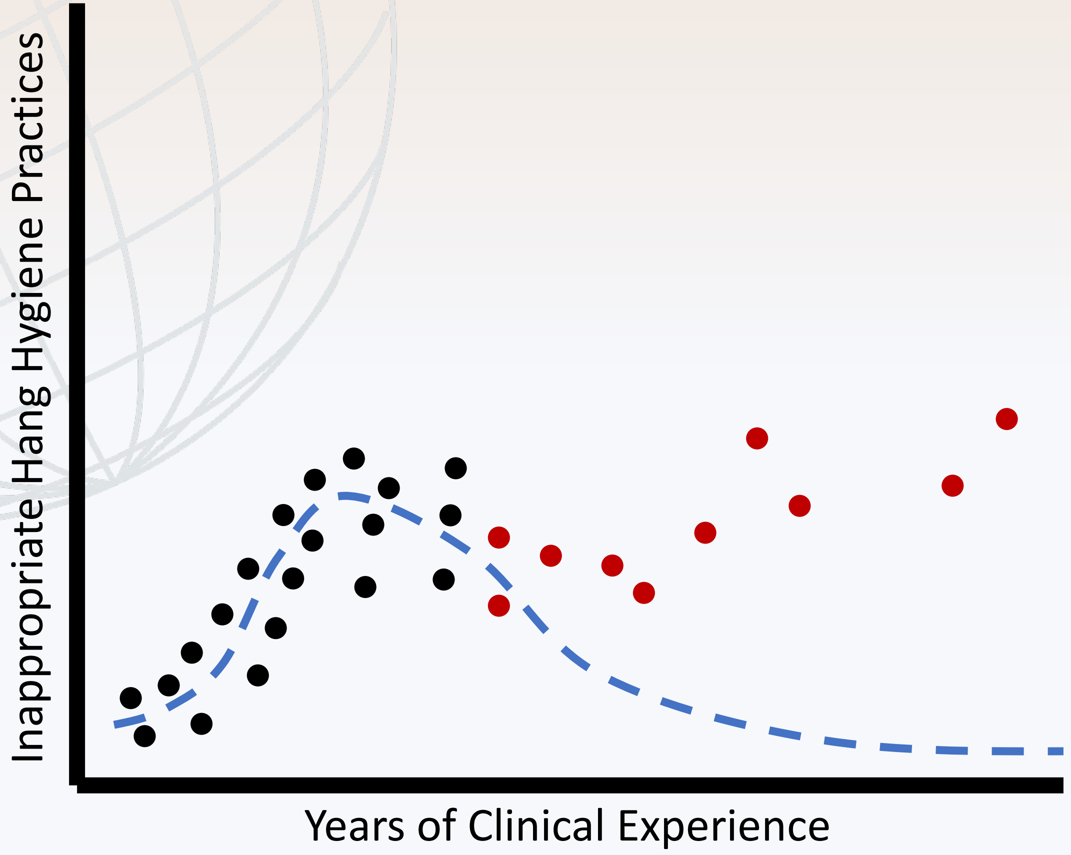
Except…



- “She’s the head of the department, I’m sure she knows better than I do.”*
- “She’s the one who reads the report, so what’s the point?”
- “He’s really busy, I’m sure he just forgot, I’ll let it go this time.”
- “If none of the senior physicians care, why should I bother?”

*Gender of examples determined by rolling a dice

Sampling and Selection Bias

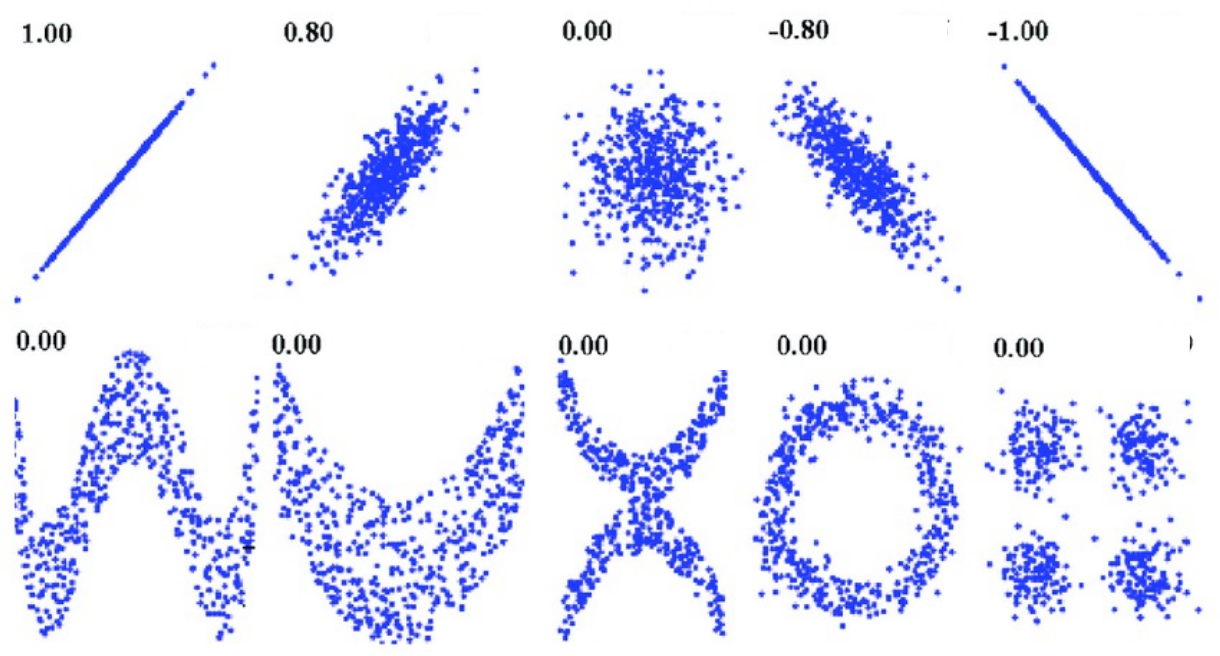


Tainted Samples

- Your sample contains something that is simple inextricably biased
- Often a subjective outcome
  - Performance evaluations
  - "Cultural Fit"
  - Awards and honors

Limited Features and Model Misspecification

- A feature may describe one subset of your population well, and another poorly



T. Vu et al., 2018

Sample Size Disparity

- Sampling and prediction errors will be larger in smaller populations
- If those smaller populations are from specific subgroups, you will have worse accuracy in those subgroups

Inappropriate Proxies

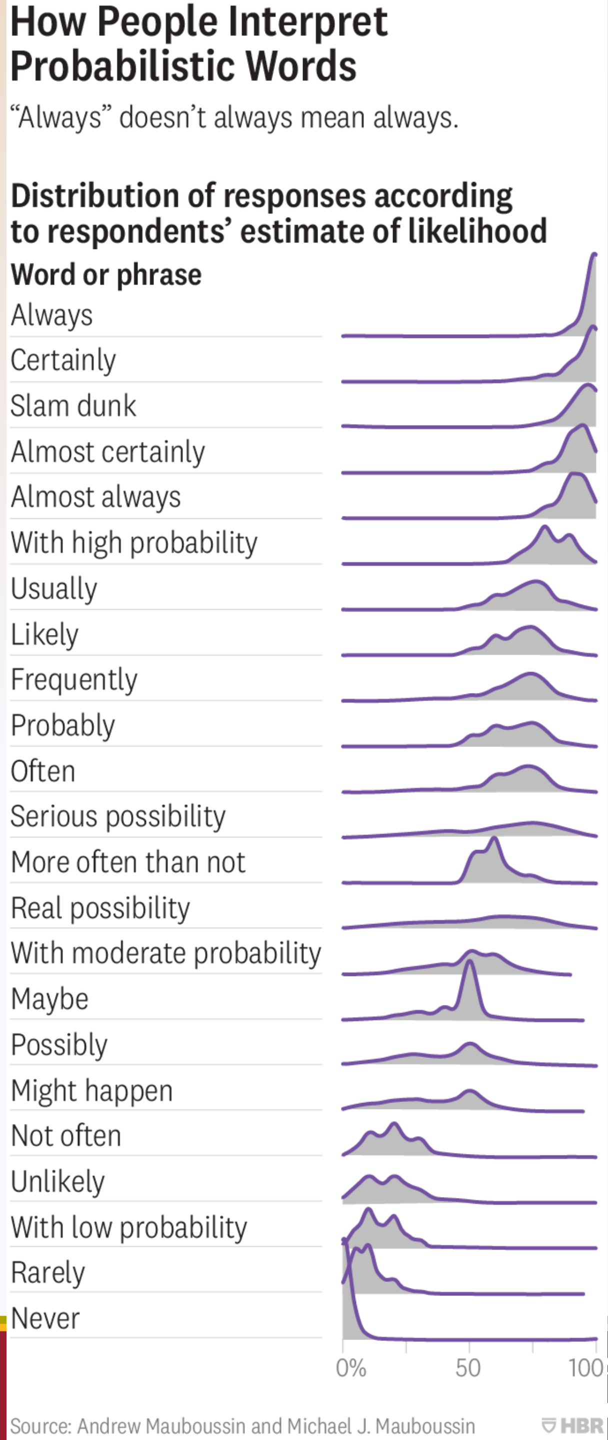
- You may remove variables you think are biased, but accidentally retain other variables that are highly correlated, essentially creating proxies
- Racial or Ethnic Background and Neighborhood for example

The Algorithm

- Actual bias in the algorithm itself or its implementation
  - Turnitin is more likely to flag non-native speakers of English, as native speakers can better obfuscate long, plagiarized passages with subtle changes
- Biased outcomes based on over-representation
  - African-American photos are overrepresented are in facial recognition databases, which gives more opportunities for false positive identifications
- Implementation choices
  - Are the inputs or results sorted? If so, how are they sorted?
  - Flaws in random number generation

Production

- Now that the algorithm has run, someone has to use it
- We're back to human beings and our wealth of biases
- How do we use language?
- "Lets call the top 25% of the population High Risk and mark them with the color red"
  - Many people may interpret that as >75%, most will interpret it as >50%
  - What if it's 17?%
  - What if that person is deciding whether you get released on bond or probation, and your score is your likelihood of re-offense?



The Danger of Reinforcing Loops

- The data given to an algorithm is biased in some way
- The algorithm, not knowing the data is biased, reinforces this bias
- Implementing the algorithm means the new data that you have coming in confirms the existing bias
- Lum and Isaac, 2016 (in the readings) go through a detailed example of this for predictive policing

Questions?

Fairness

- Fairness is harder
- One of the interesting effects of the growth of machine learning is forcing the question of formally specifying what is “fair”
- Notation reminder: $p(O \mid \mathbf{Z}, A=0) = p(O \mid \mathbf{Z}, A=1)$
- We’re now going to drop **Z**

Types of Fair

- Unawareness: The algorithm is not made aware of certain variables we have decided have historical, social, etc. roots.
  - p(O)
- Demographic Parity: O is independent of A
  - p(O) = p(O | A)
- Equalized Odds: O is independent of A conditional on Y
  - p(O | A=0, Y=1) = p(O | A=1, Y=1)
  - Where Y is some other factor. For example, “Qualified Applicants”
- There are *many* papers with a lot of math discussing these ideas

Ways To Get at Fair

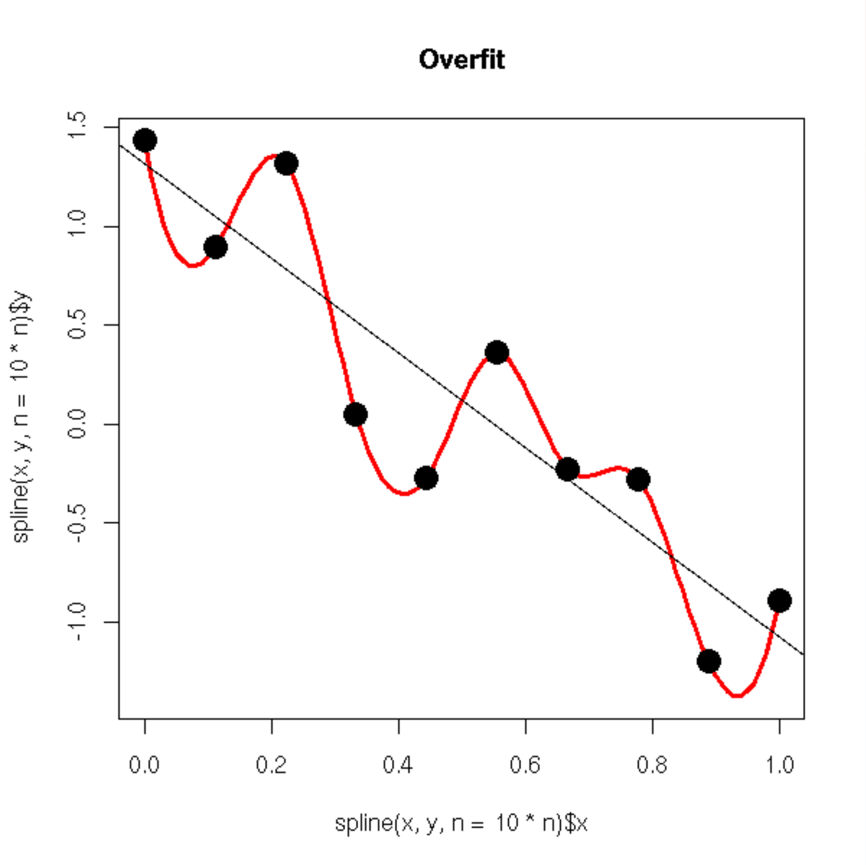
- First, trying to address bias as much as possible
- Remove variables that are historical or systematic sources of unfairness, and which shouldn't factor into decisions
  - Easy in principle, harder in practice, domain specific
- Differing thresholds for different groups, if the algorithm is being used to make a binary decision
  - Formalized versions of "soft" criteria, etc.
- All of these have hazards, some are mutually exclusive, all of them are hard
- There is potentially an accuracy vs. fairness tradeoff

Shifting Gears…

- Questions?

Over and Underfitting

- Familiar concepts from regression
- Underfit: The terms in your model are inadequate to predict your outcome
- Overfit: The prediction of the model corresponds too closely to your dataset, and will fail to predict outside it
  - Fails at “out of sample” prediction
  - The extreme case is a model with a term for every point
- Both are to be avoided
  - Underfitting is often indicative of data problems, and may be “too late” once you’re at the model fitting stage
  - Overfitting is easier to avoid



Test and Training Data

- Training Data
  - What your model is “allowed to see” – this is what you are fitting to
  - You *don’t* evaluate the accuracy of your model against this data – that way lies overfitting
- Test Data
  - Some portion of your data that is “set aside” and not used for fitting
  - You evaluate the accuracy of your model against this data to see how well it predicts data it has not encountered
- On occasion, there is also a “Validation Set”, which is used after training but before testing to tune hyperparameters

How To Split Your Data

- This is often a 70/30 split between training and test data
- But…
  - You want to make sure you have enough in your training data set that you can fit small N combinations of factors
  - This is where that sample size bias problem crops up
  - You also want enough in your test data to know if you're poorly predicting specific groups
- This is a decision you should make looking at your data, frequency tables if you can, etc.
- It will also depend on the size of your dataset

How Not To Split Your Data

- Anything that makes your training and test datasets systematically different from one another
- "I'll take the first six months of data as my training, and my last two as test" – A very sad hypothetical person who started collecting data in August 2019
  - Structures of the system that evolve over time are called "non-stationary" and are challenging
  - This gets very hard for time series modeling
- "Clinic A and B will be the training data, Clinic C will be the test"
- There *may* be exceptions to this, but generally there's safety in random sampling

Cross-Validation

- Repeated resampling of your data into test and training data sets
- Lots of variety
- The most common in machine learning is *k*-fold, where k is the number of resampling runs
- One can argue that a simple test-train split is 1-fold cross-validation
- Most commonly 10-fold

Why This Is Good – and Why It Might Not Be

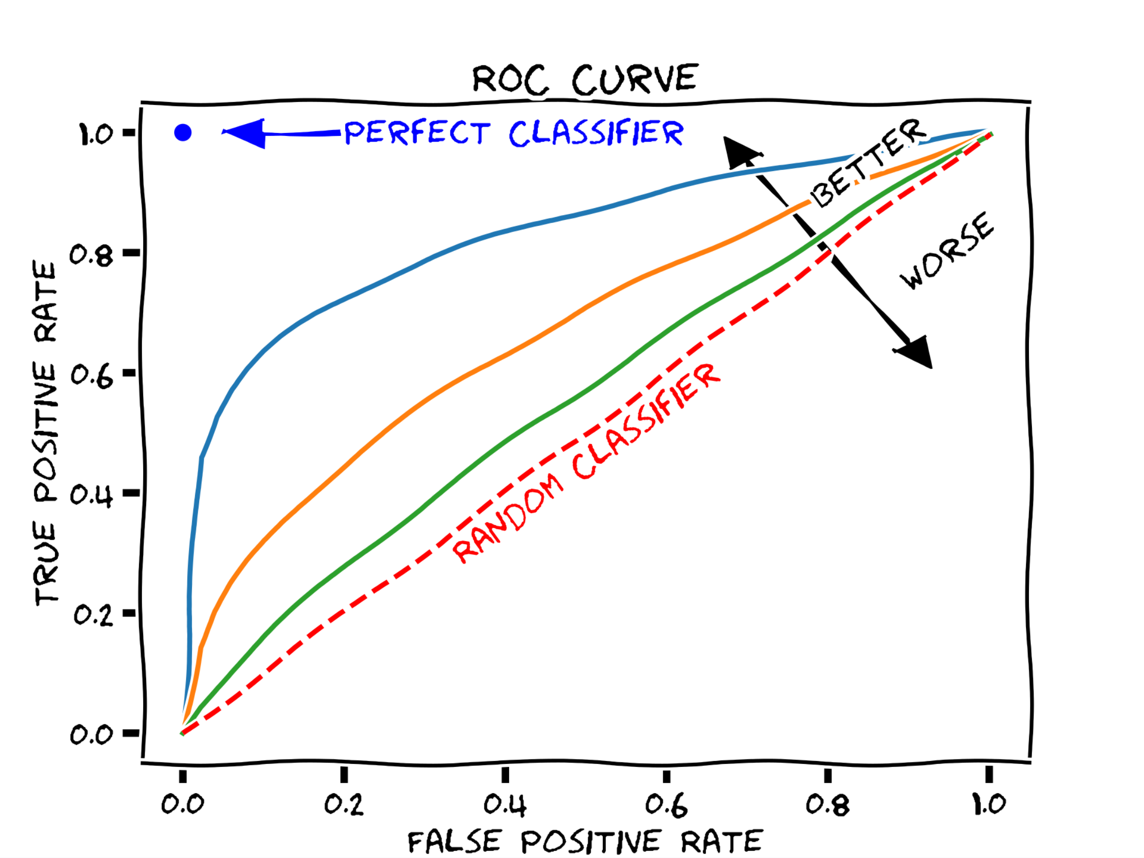
- Uses all your data – more efficient with small data sets
- Each resampling gives you a measure of your model's performance, so you can start to gauge the variability of performance
  - 98, 92,89, 94, 95 – Things look okay
  - 98, 92, 47, 94, 95 – Cause for concern
- Helps solve the need for a validation set for hyperparameter tuning
- The problem: While easy to implement, CV can be slow – each resampling means the model must be re-run.
  - Not a big deal if your model is fast
  - Very much a big deal if your model is slow

Measuring Performance

- Many of the same metrics used in measuring regression performance are used for other models
- For binary outcomes
  - ROC curves, C-statistics, etc.
- For continuous outcomes
  - Root mean squared error (RMSE), absolute error, etc.
- Many others of varying complexity

ROC Curves and C-statistics

- Receiver operating characteristic curve, shows the performance of a classifier over any discrimination threshold
- The C-statistic, or concordance statistic, is the area under the curve with 0.50 being equivalent to random, and 1.00 being perfect
  - Mathematically you can go below one, but practically, you now have a new classifier by doing the opposite of whatever your current one says with higher accuracy
- Very high C-statistic values have started to be viewed with a touch of skepticism



Continuous Predictions

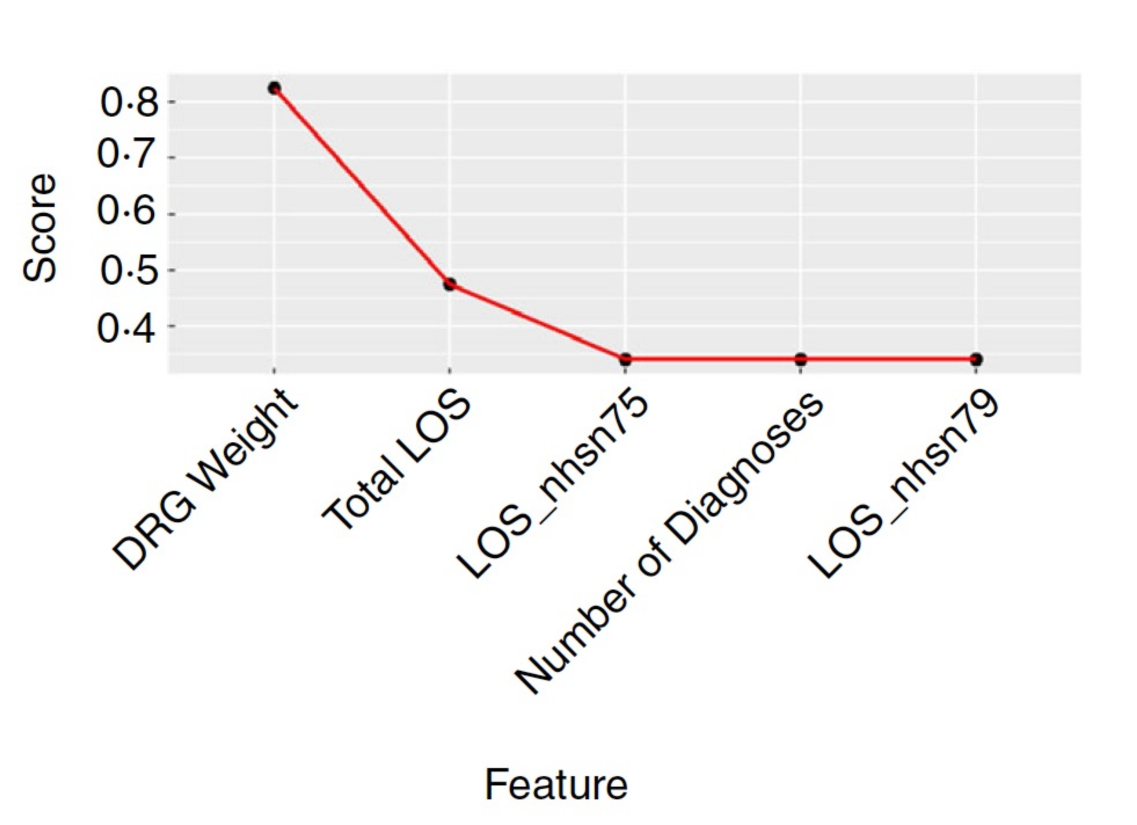
- Mean Absolute Error: The average of the difference between the predicted values and the actual values.
  - $\frac{1}{N}\sum_{j=1}^{N}|y_j - \widehat{y_j}|$
- Mean Squared Error: The average of the square of the difference between the predicted values and the actual values
  - $\frac{1}{N}\sum_{j=1}^{N}(y_j - \widehat{y_j})^2$
  - This expands the impact of larger errors, allowing focus on those areas
  - This makes sense if being off by 10 is more than twice as bad as being off by 5
- Root Mean Squared Error: The root of MSE
  - Returns MSE to a somewhat interpretable unit

Other Considerations

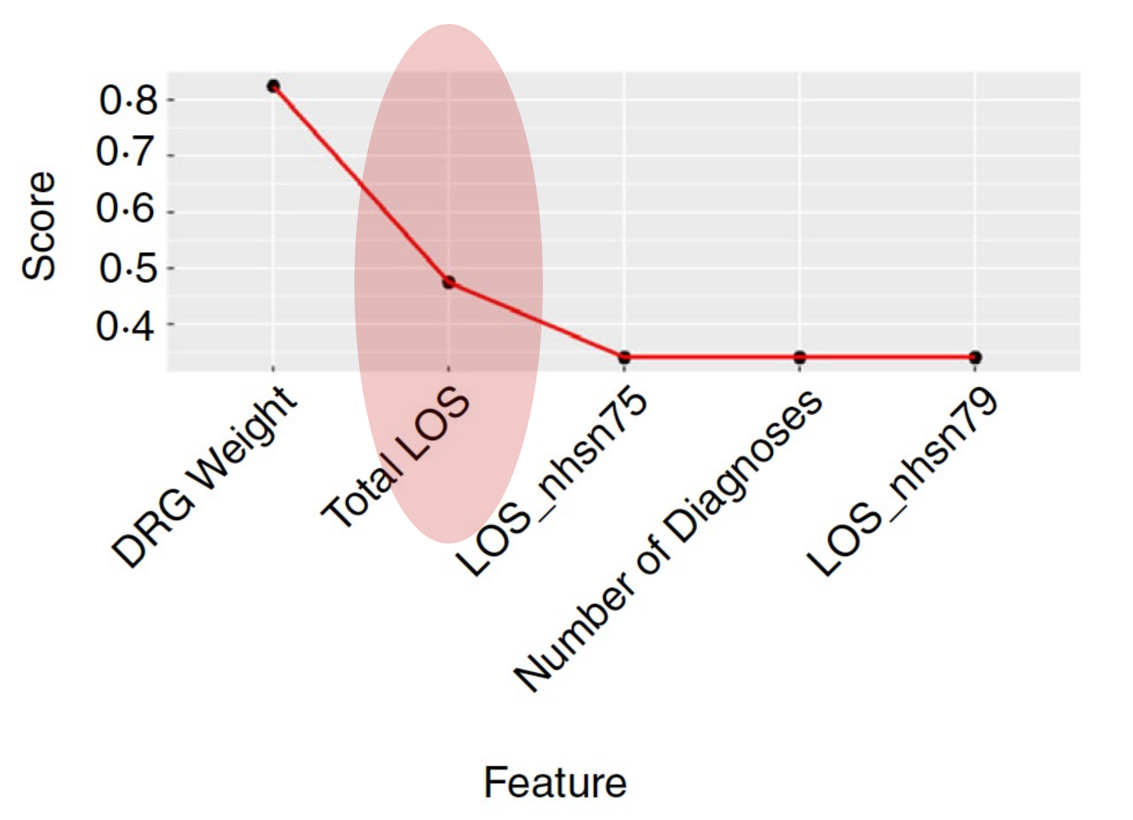
- Is the model performing well for its intended role?
  - Is it fast enough to help decisions be made?
  - Are the factors that are being included acceptable to the audience, perceived to be fair, etc.?
- Consider including "null models" – models with very simple predictions, to make sure that the effort going into the model is producing actual value

Example: Predicting Antibiotic Usage

- We performed a study asking what predicted antimicrobial administration
- The CDC uses a model to do this that, while intended for prediction, is a somewhat more traditional “semi-causal” epi model
- We wanted to see what would come out if we just went for raw predictive performance
- In the readings as Chowhury, 2020
- Results were then discussed by a panel of clinician experts



Adult SAAR group	Null LM	Null NB-GLM	SVR	CB
All-antibacterials	8·16	8·60	5·86	5·17
Beta-lactam	1·50	1·90	1·48	1·48
CDI	2·73	2·90	2·47	2·42
Community-onset	2·28	2·48	2·24	2·09
Hospital-onset	3·22	3·37	2·62	2·45
Resistant Gram-positive	2·39	2·61	2·32	1·98



Adult SAAR group	Null LM	Null NB-GLM	SVR	CB
All-antibacterials	8·16	8·60	5·86	5·17
Beta-lactam	1·50	1·90	1·48	1·48
CDI	2·73	2·90	2·47	2·42
Community-onset	2·28	2·48	2·24	2·09
Hospital-onset	3·22	3·37	2·62	2·45
Resistant Gram-positive	2·39	2·61	2·32	1·98

Feedback

- More accurate SVR models were too slow to be useful
- Race was discarded entirely as a predictor (it also wasn't very important) by general agreement that it wasn't collected with any degree of uniformity or accuracy
- Total Length of Stay was considered problematic because it's highly "gameable", and these predicted scores would be used in evaluation
  - This is somewhat more problematic, as LOS is a fairly important predictor
- For some rare antibiotic classes, "Everyone is the Average" worked nearly as well as a prediction model as some fairly intensive ML work
  - This revealed there's a distinct zero-inflation problem in the data, and current work is modeling Ever/Never administration of antibiotics + a new predictor for what value you take if $p(\text{Ever}) > 0.5$