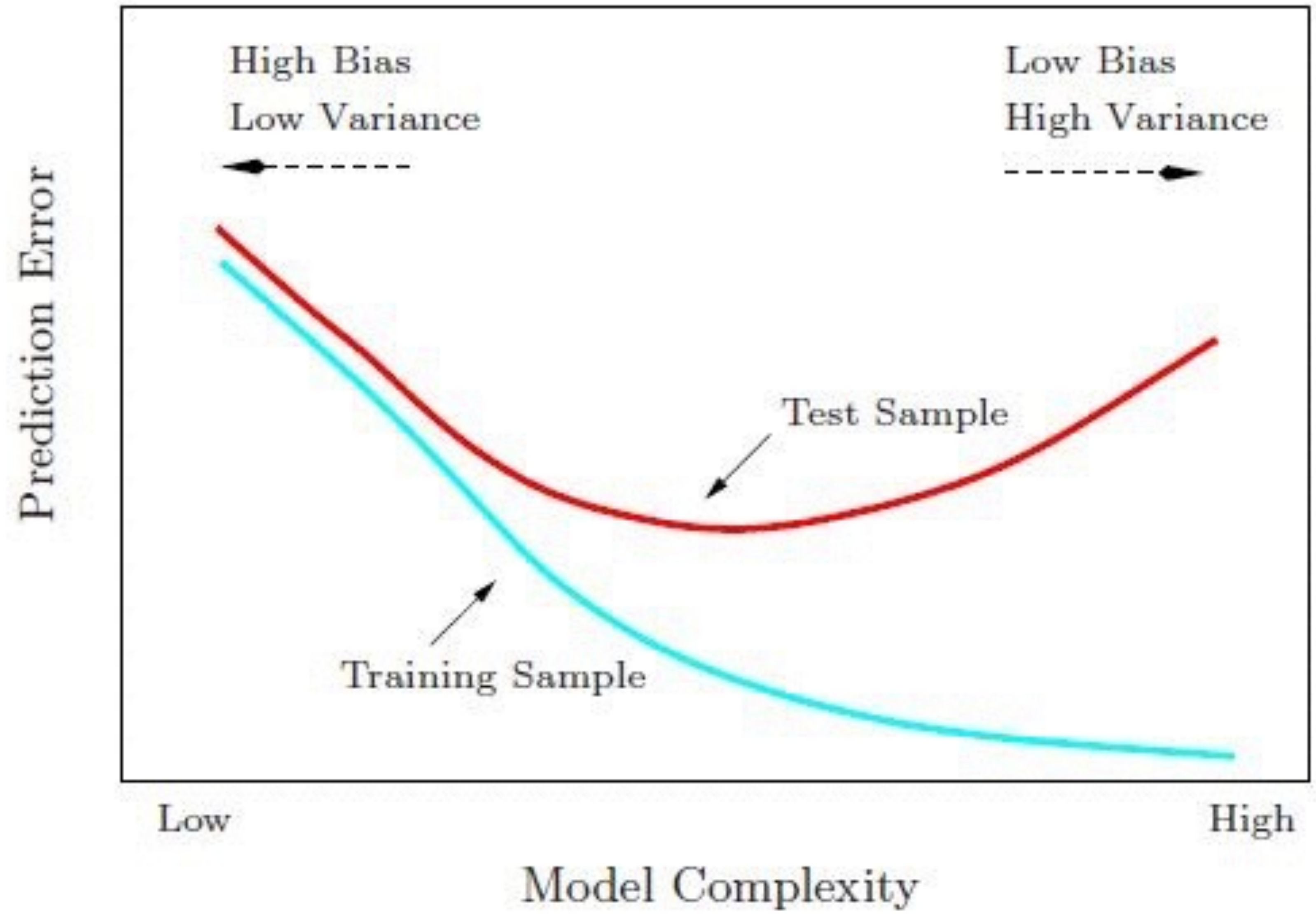


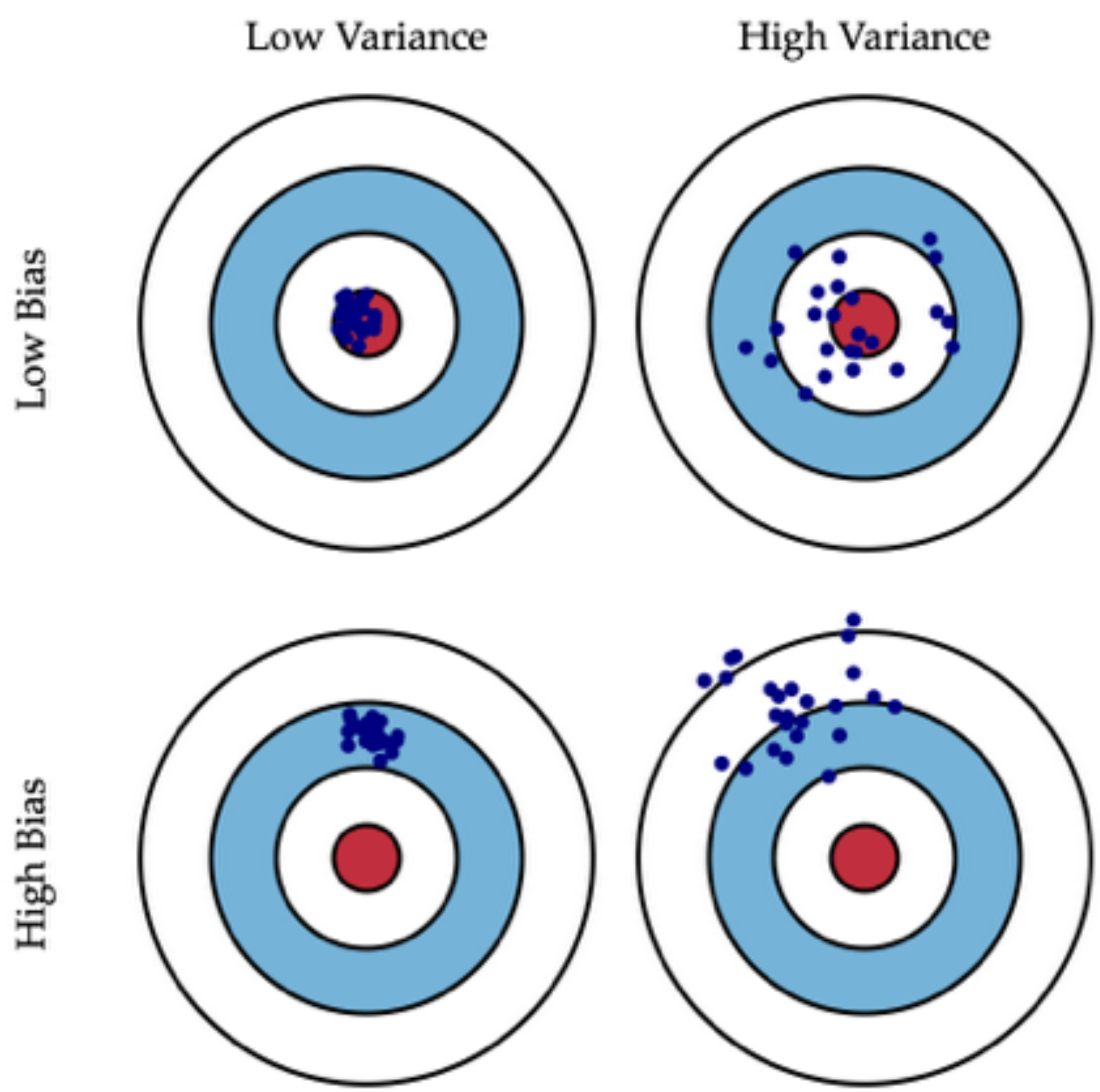
Model Performance and Validation

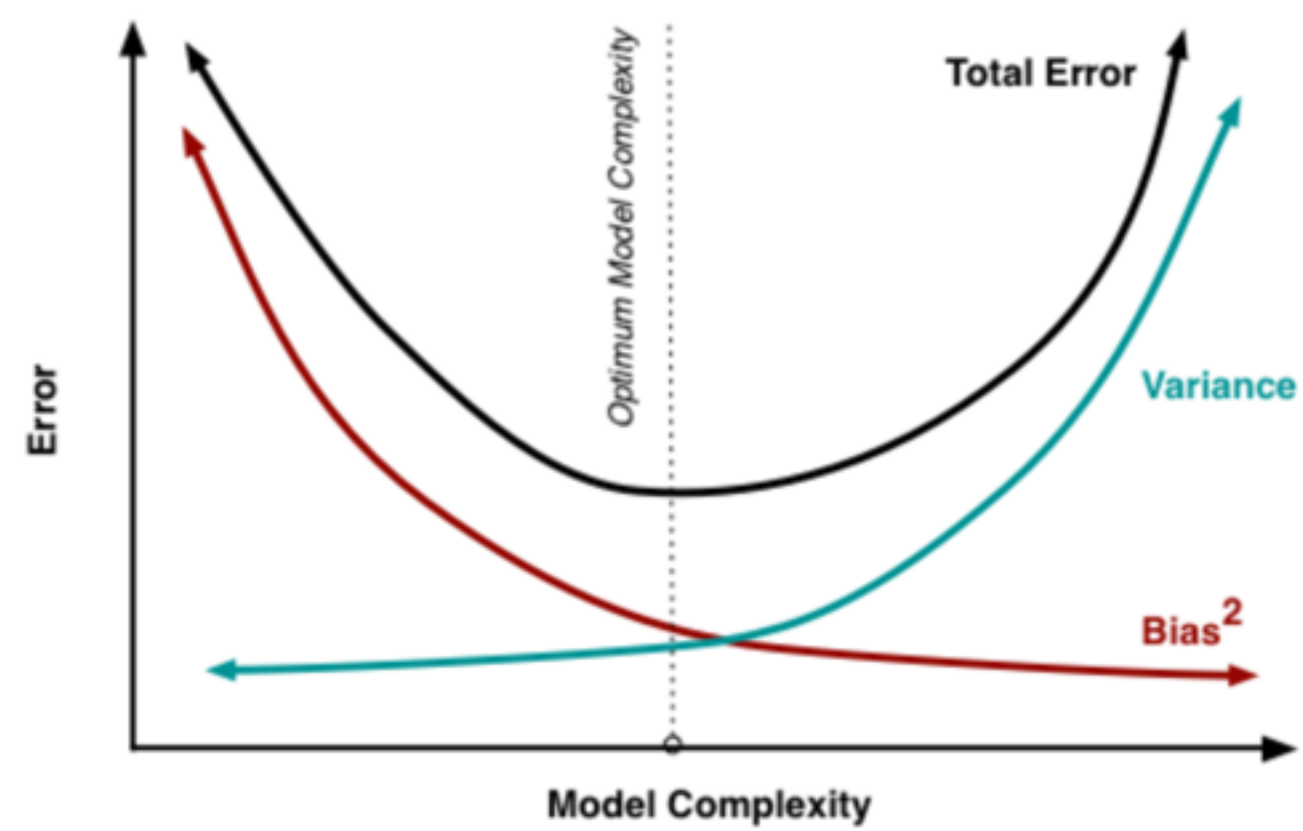
Two-day workshop

Duke University
Adam Chekroud

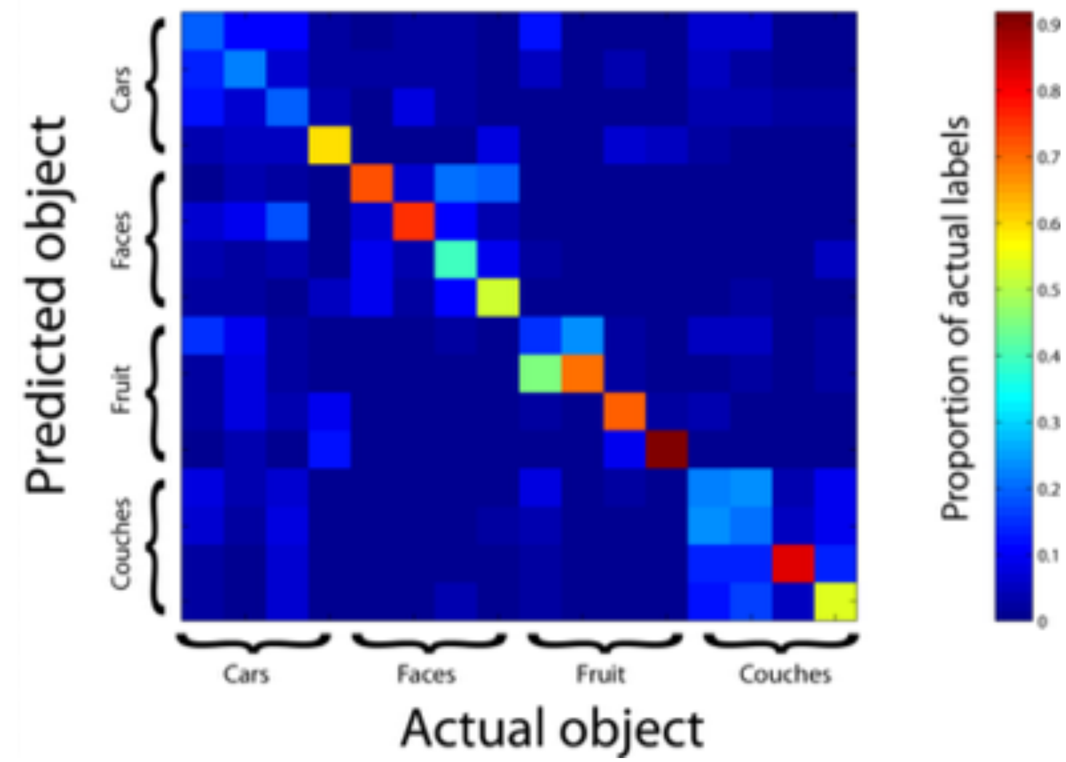
Objectives

		Condition (as determined by "Gold standard")		
		Condition Positive	Condition Negative	
Test Outcome	Test Outcome Positive	**True Positive**	**False Positive** (Type I error)	Positive predictive value = $\frac{\Sigma \text{ True Positive}}{\Sigma \text{ Test Outcome Positive}}$
	Test Outcome Negative	**False Negative** (Type II error)	**True Negative**	Negative predictive value = $\frac{\Sigma \text{ True Negative}}{\Sigma \text{ Test Outcome Negative}}$
		Sensitivity = $\frac{\Sigma \text{ True Positive}}{\Sigma \text{ Condition Positive}}$	**Specificity** = $\frac{\Sigma \text{ True Negative}}{\Sigma \text{ Condition Negative}}$	





Outline

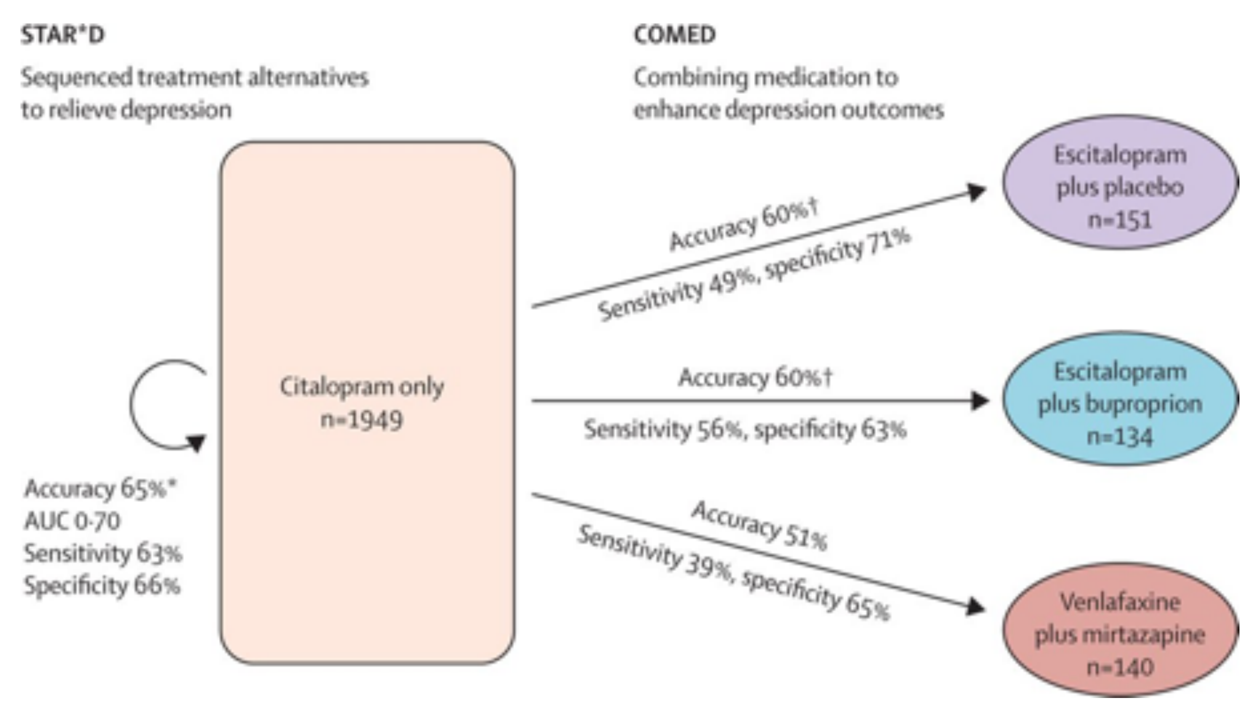


How good is my model?

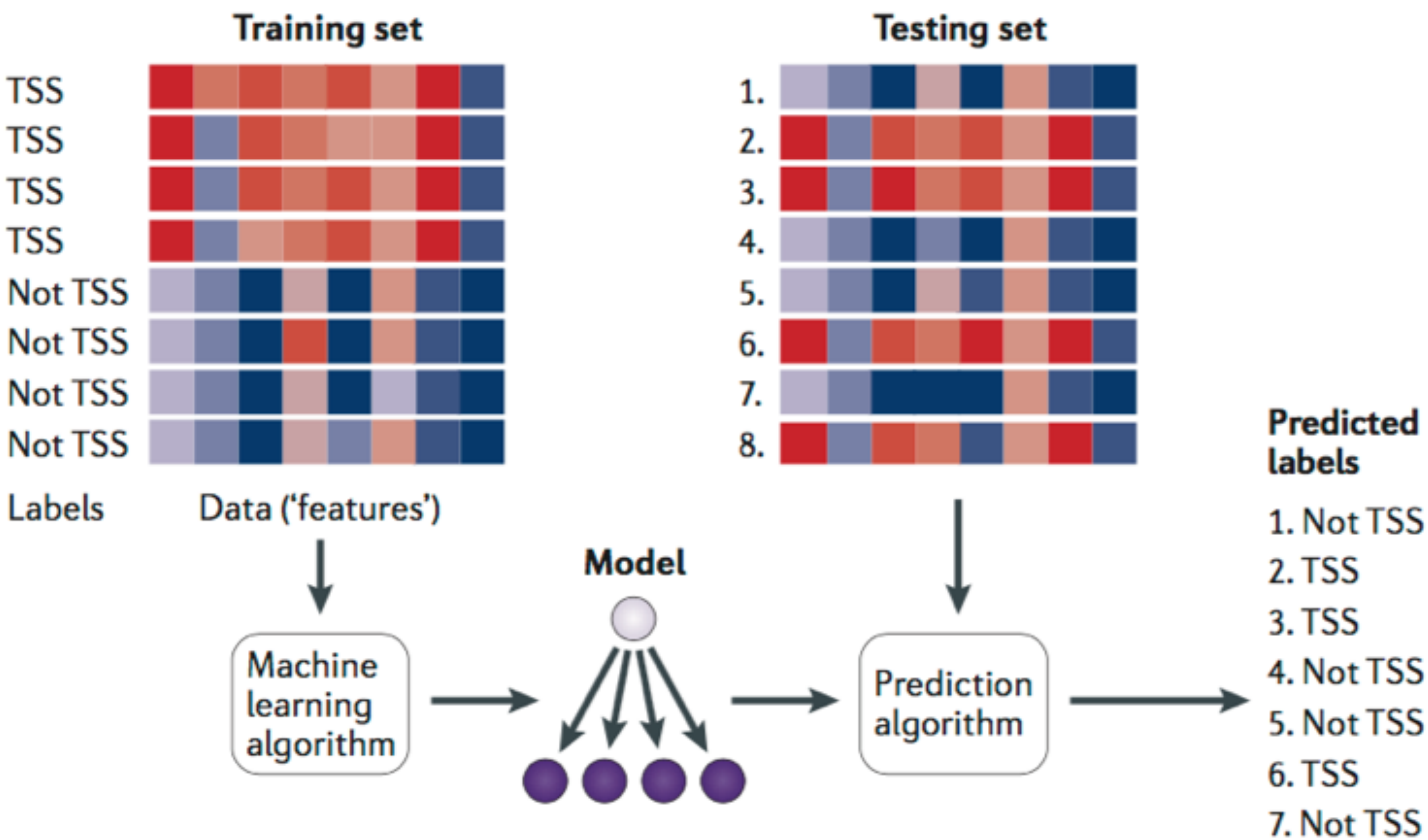
- How do we measure performance?
- Bias-variance tradeoff

Is my model real?

- Validation procedures
- Prospective validation!



How good is my model?



1 - Get the predictions from our model

2 - Compare them to the truth ——>

$$\text{accuracy} = \frac{\#\ \text{correct predictions}}{\#\ \text{total data points}}$$

How good is my model?

What if outcomes are not equally likely?

- If only 1% of people suicide, guessing “no suicide” for everyone will give 99% accuracy!

Confusion matrix

		Condition (as determined by "Gold standard")		
		Condition Positive	Condition Negative	
Test Outcome	Test Outcome Positive	**True Positive**	**False Positive** (Type I error)	Positive predictive value = $\frac{\Sigma \text{ True Positive}}{\Sigma \text{ Test Outcome Positive}}$
	Test Outcome Negative	**False Negative** (Type II error)	**True Negative**	Negative predictive value = $\frac{\Sigma \text{ True Negative}}{\Sigma \text{ Test Outcome Negative}}$
		Sensitivity = $\frac{\Sigma \text{ True Positive}}{\Sigma \text{ Condition Positive}}$	**Specificity** = $\frac{\Sigma \text{ True Negative}}{\Sigma \text{ Condition Negative}}$	

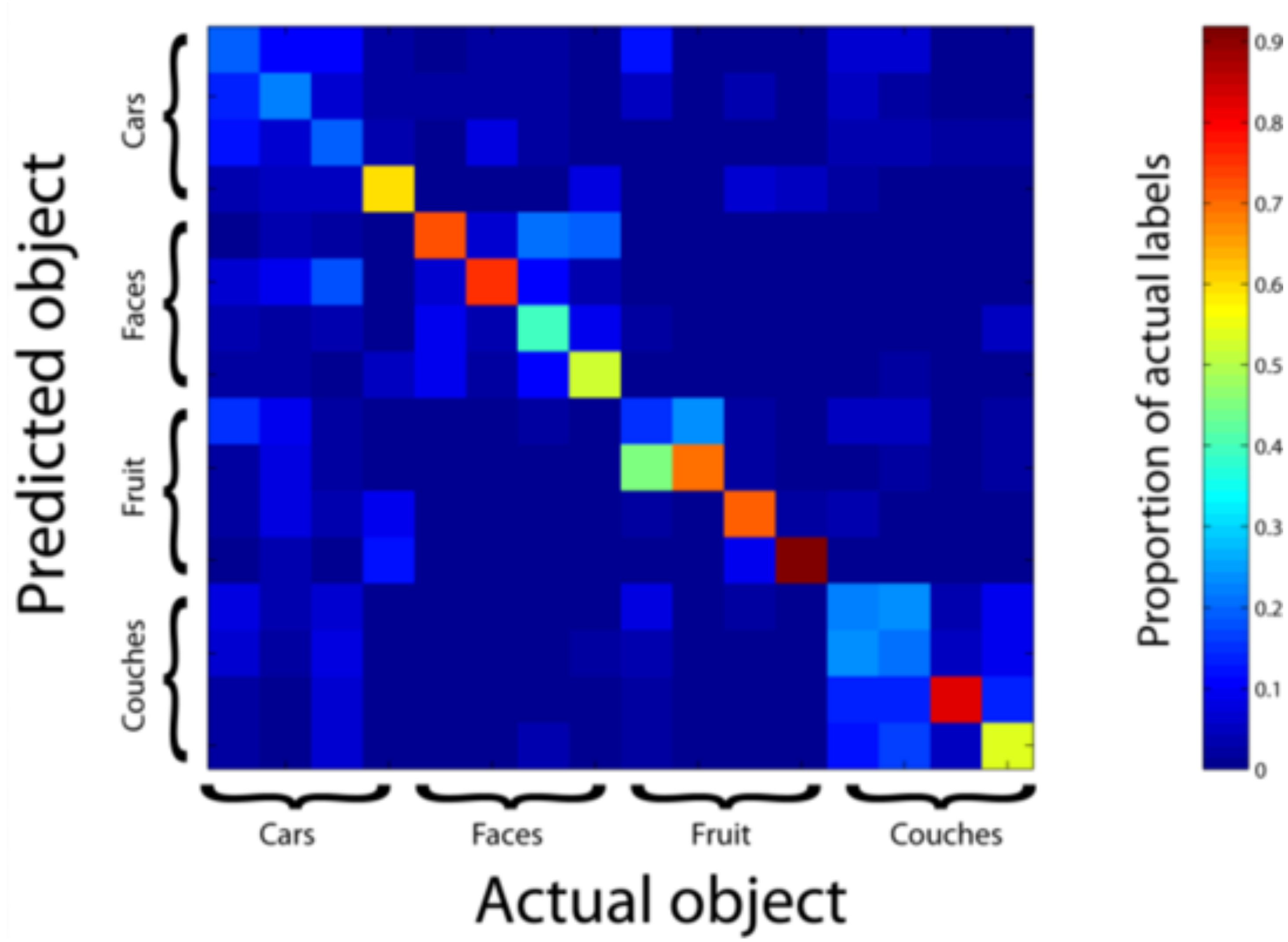
How good is my model?

		True condition			
	Total population	Condition positive	Condition negative	Prevalence $= \frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	
Predicted condition	Predicted condition positive	**True positive**	**False positive** (Type I error)	Positive predictive value (PPV), Precision $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Test outcome positive}}$	False discovery rate (FDR) $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Test outcome positive}}$
	Predicted condition negative	**False negative** (Type II error)	**True negative**	False omission rate (FOR) $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Test outcome negative}}$	Negative predictive value (NPV) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Test outcome negative}}$
	Accuracy (ACC) $= \frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	True positive rate (TPR), Sensitivity, Recall $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR), Fall-out $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) $= \frac{\text{LR+}}{\text{LR−}}$
		False negative rate (FNR), Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	True negative rate (TNR), Specificity (SPC) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$	Negative likelihood ratio (LR−) $= \frac{\text{FNR}}{\text{TNR}}$	

DOR = (TP/FP)/(FN/TN)

— is yours greater than 1 ?

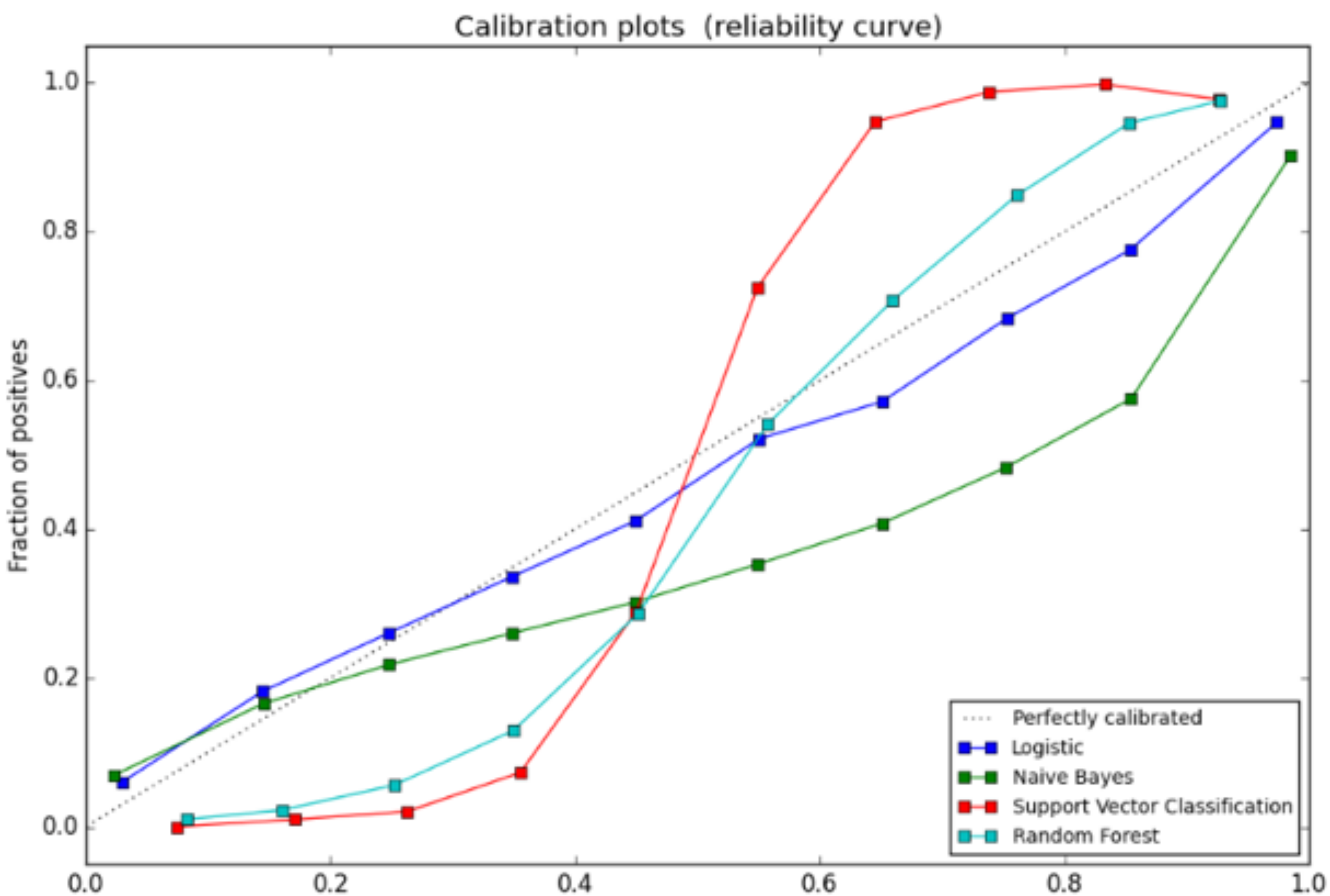
How good is my model?

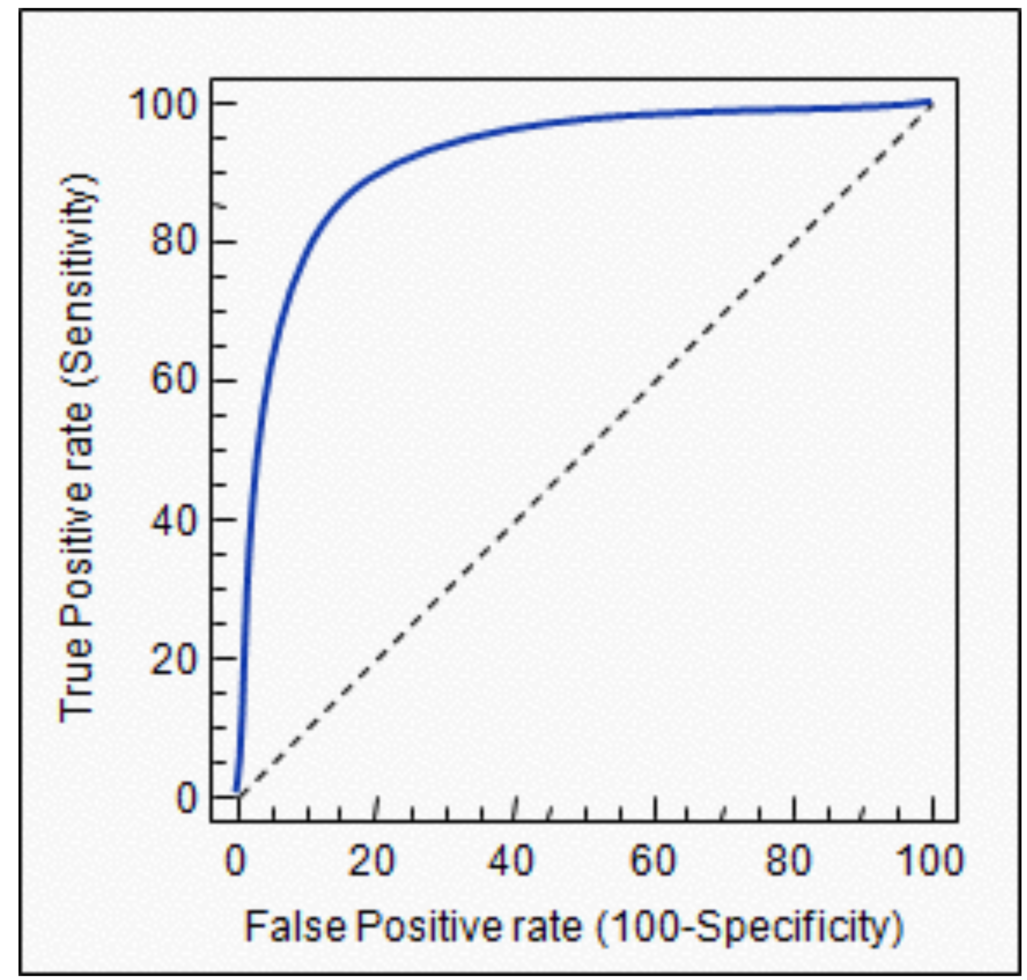


<u>Multi-class extensions</u>

- Confusion matrix is most helpful here
- Can calculate within-category performance measures.

How robust is my model?





- Are the output probabilities well calibrated?
- What happens if we change the cutpoint for classification?

http://scikit-learn.org/stable/auto_examples/calibration/plot_compare_calibration.html

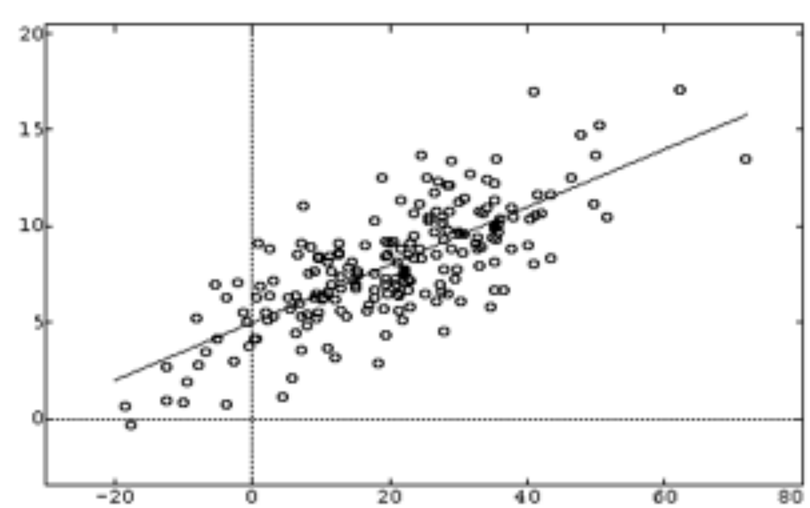
How good is my model?

<u>What about numeric outcomes?</u>

- RMSE

$$\mathrm{RMSE} = \sqrt{\frac{\sum_i (y_i - \hat{y}_i)^2}{n}}.$$

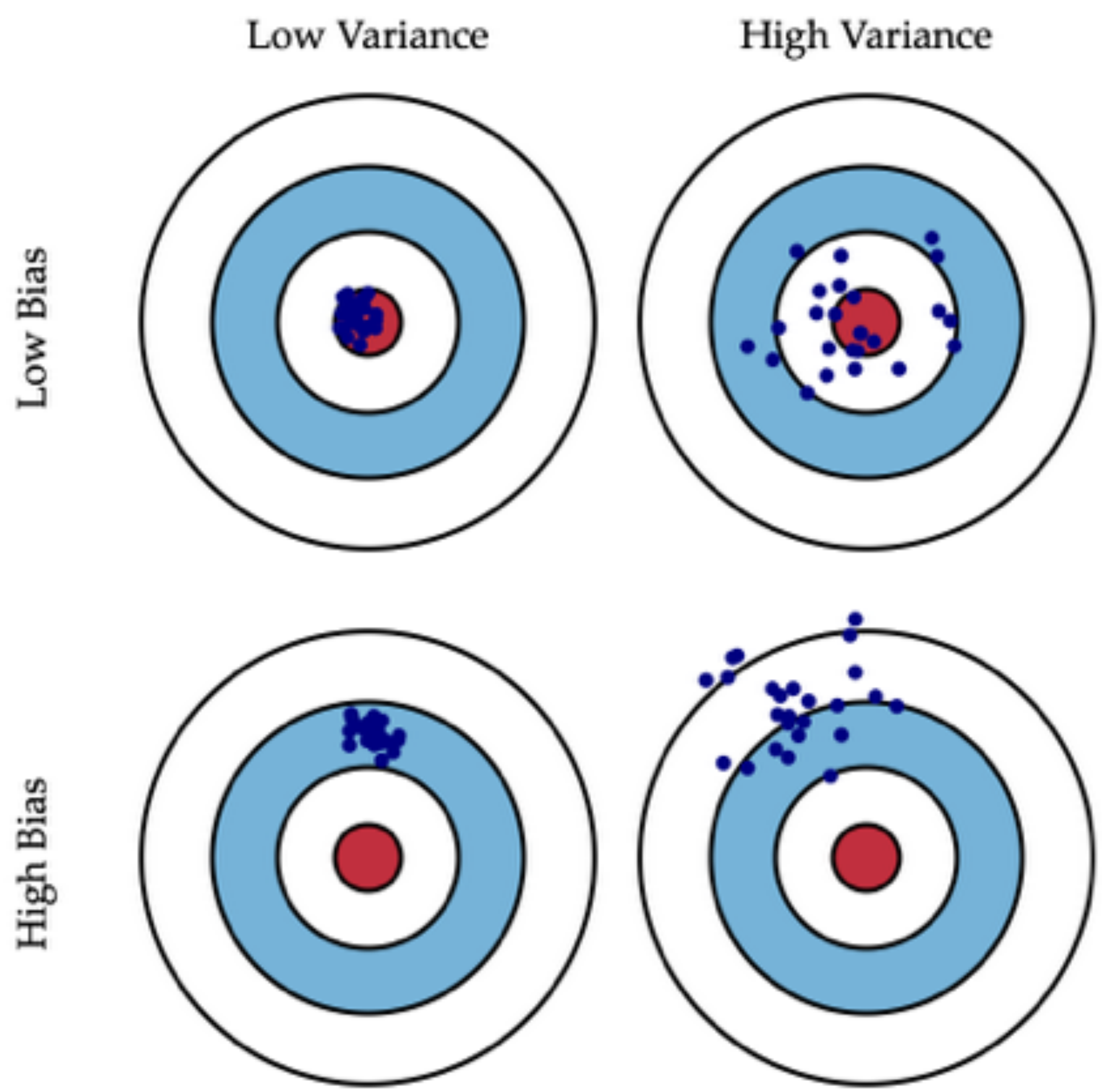
- R^2, aka coefficient of determination



  - R2 of 1 — model is a perfect fit for data
  - also indicates the proportion of variance in the data that is explained by the model

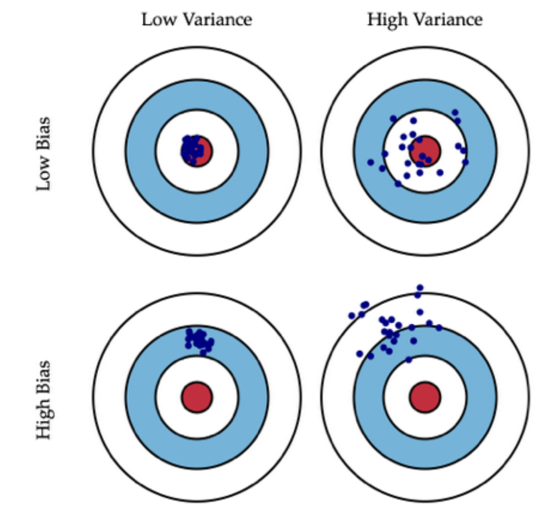
Bias-Variance Tradeoff

Model's error is not just a function of accuracy



- Also error due to variance!

Bias-Variance Tradeoff



A tiny bit of math

- We are estimating a *model* of a function/mapping f(x)
- Assuming error is normally distributed about zero, then our expected error is:

$$Err(x) = E\left[(Y - \hat{f}(x))^2\right]$$

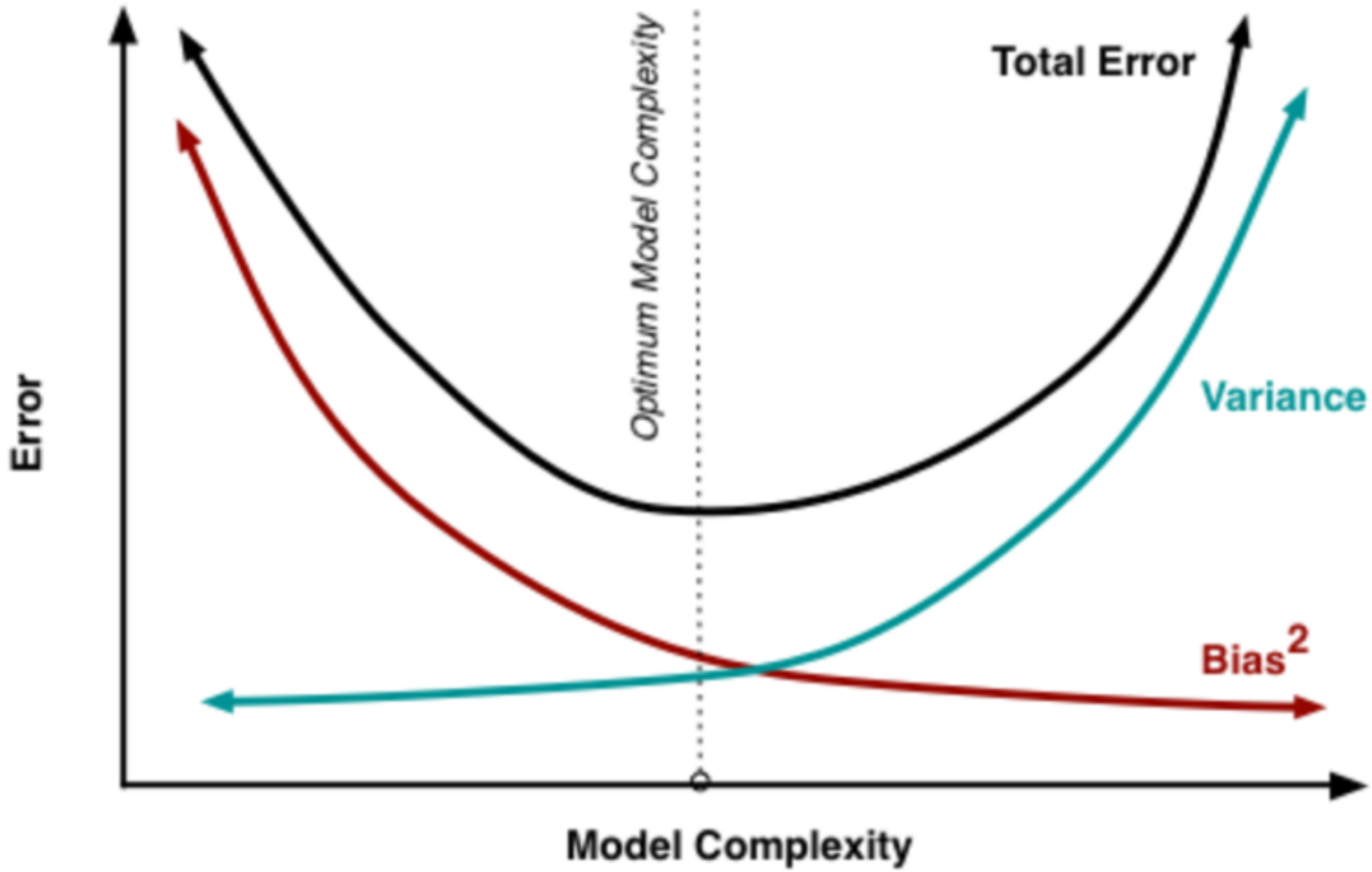
Which becomes a function of **both** bias (accuracy) and variance :

$$Err(x) = \left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\hat{f}(x) - E[\hat{f}(x)]\right]^2 + \sigma_e^2$$

$$Err(x) = \mathrm{Bias}^2 + \mathrm{Variance} + \mathrm{Irreducible\ Error}$$

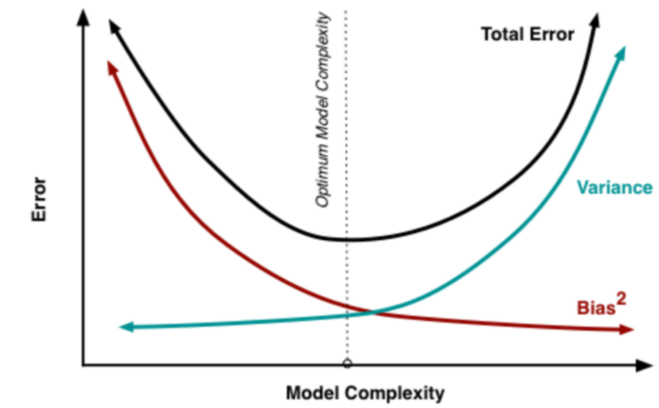
—-> Fundamental balance between accuracy and variability of our model

Bias-Variance Tradeoff



As any given model becomes more complex, its accuracy improves but generalizabilty is **necessarily** reduced

Model Validation

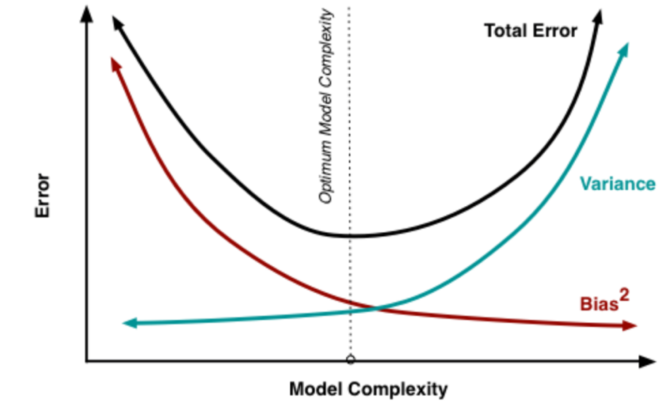


Model validation is crucial for measuring the second component of model error — its variance.

1. Take different training samples
2. Build models
3. See how variable their performance is!

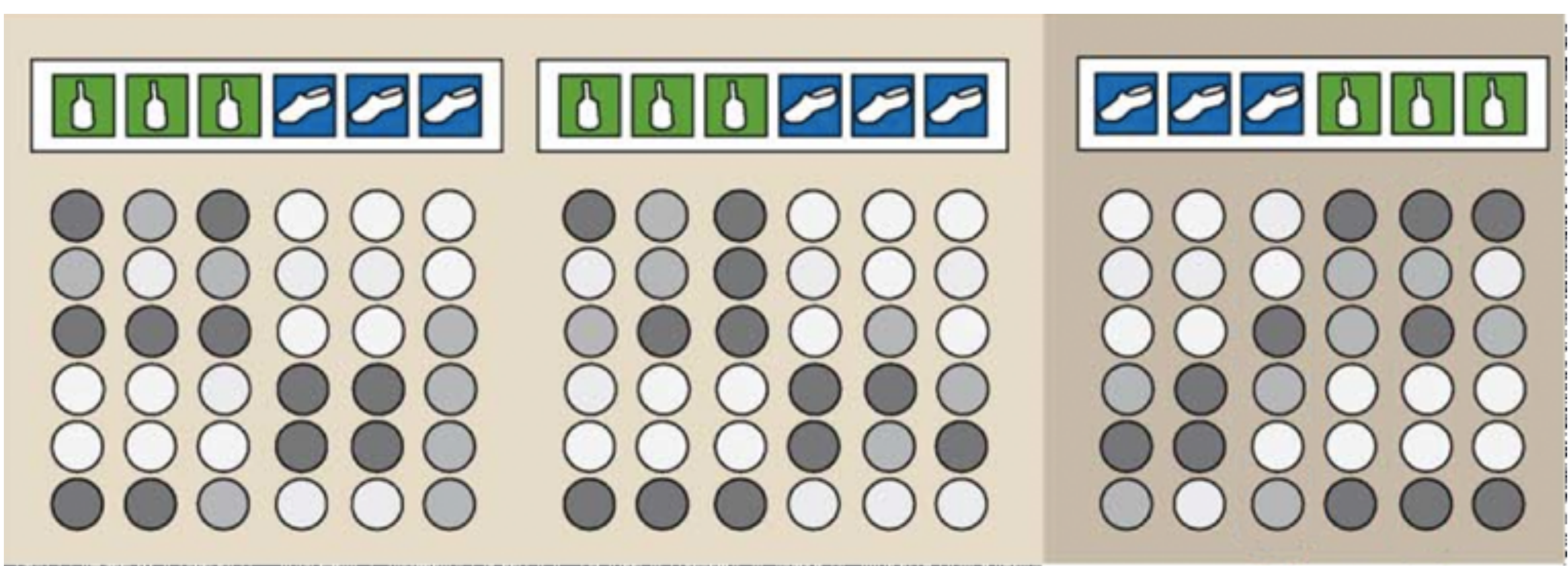
- But data is **expensive**! And **finite**!

Internal Validation



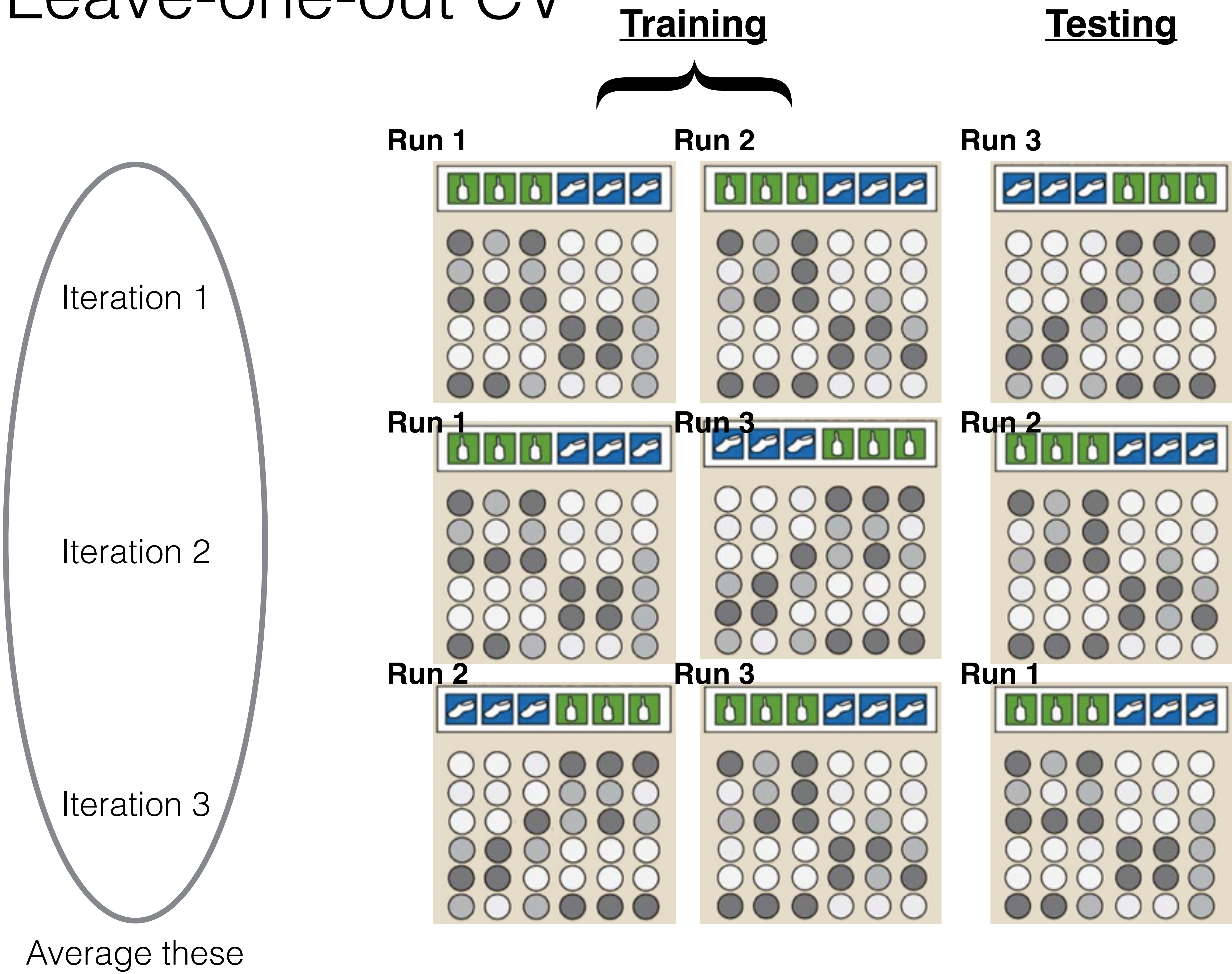
- Two motivations for internal validation:
    1. Estimate variance of modeling approach
    2. No point testing model on things it has seen before!

Easiest solution: split data into training and testing

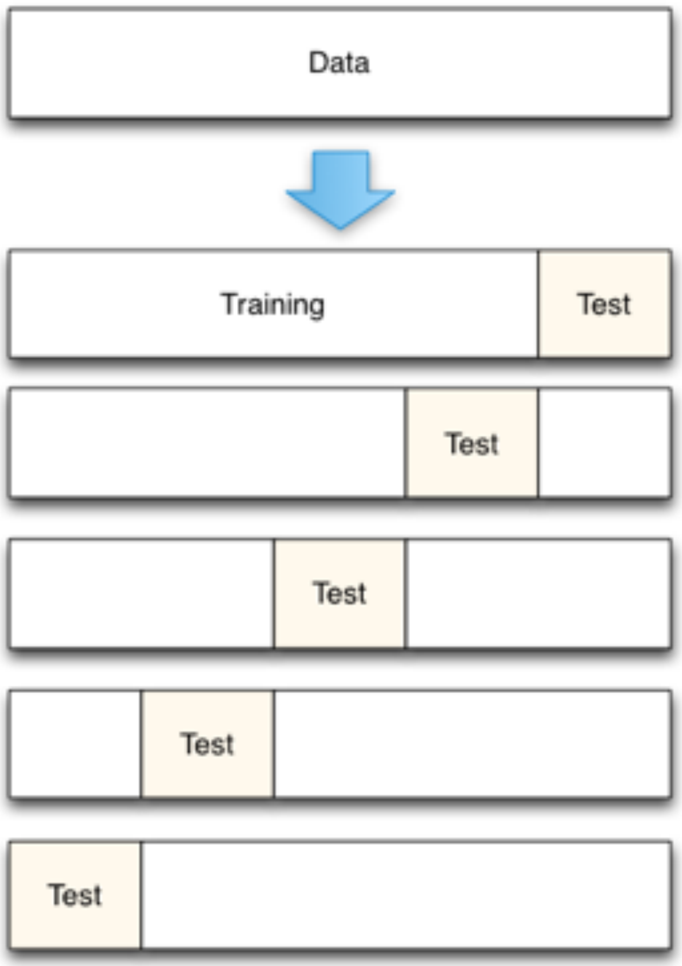


Run 1 Run 2 Run 3

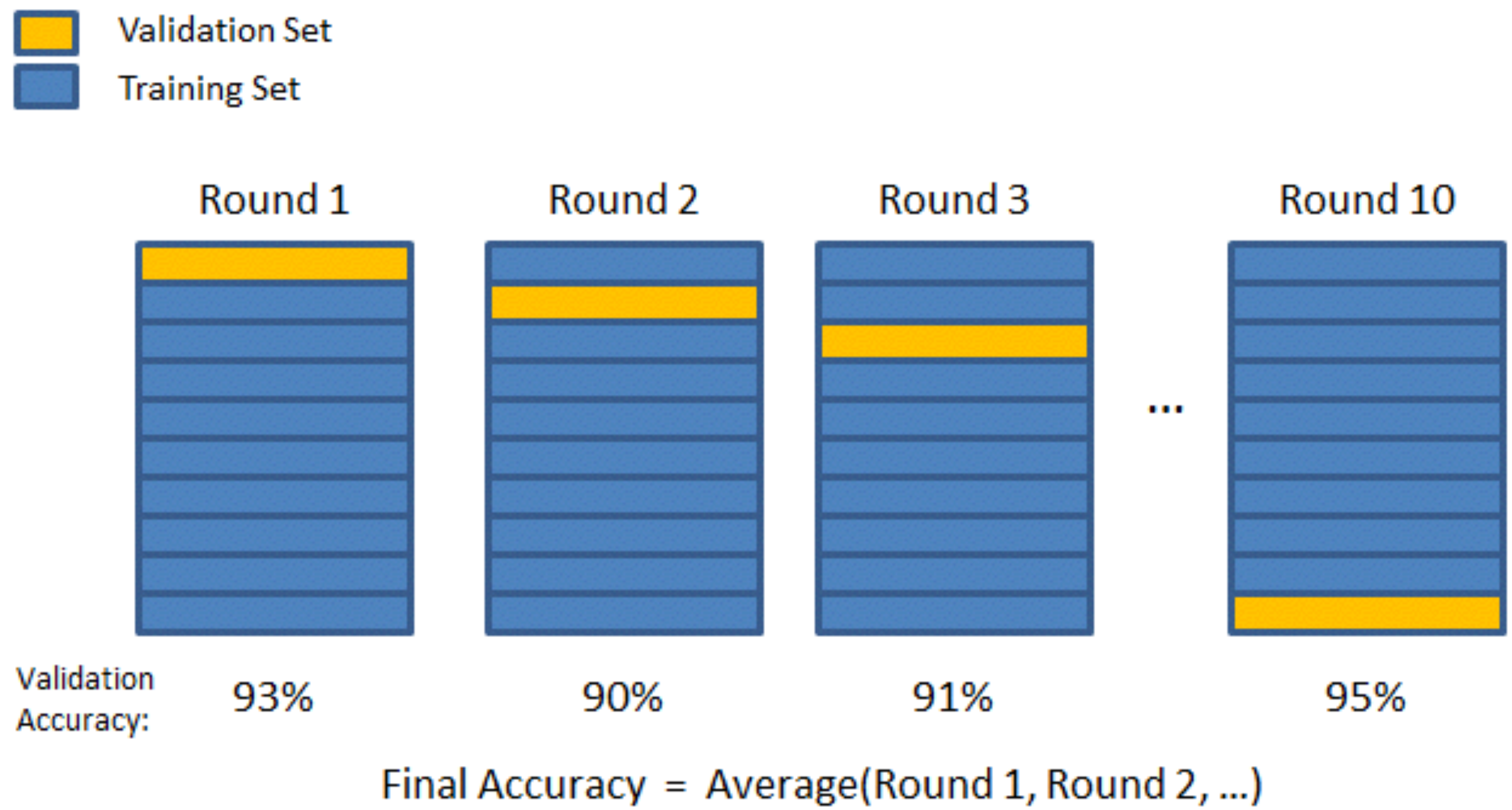
Leave-one-out CV



Better CV



5-fold CV



10-fold CV

Is my model *real* ?

- Internal validation allows us to train models and get the best balance between bias and variance
  - But how well will our model perform *in general*?
- Examine performance on totally new data
  - If good, gives us confidence that the model is picking up on “real” signal
    - Better candidate for the truth!

Best practice: prospective validation

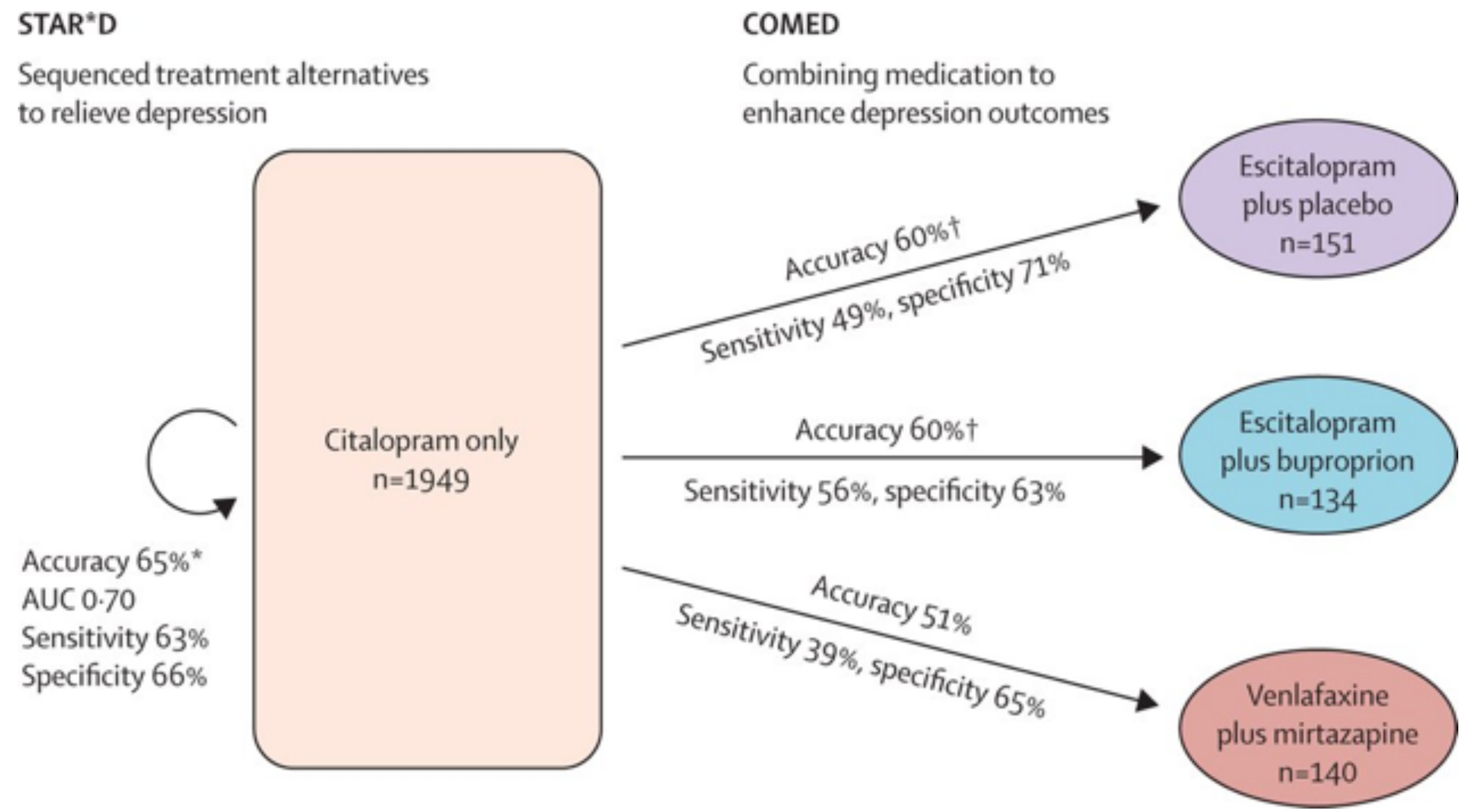
Is my model *real* ?

Articles

Cross-trial prediction of treatment outcome in depression: a machine learning approach

Adam Mourad Chekroud, Ryan Joseph Zotti, Zarrar Shehzad, Ralitza Gueorguieva, Marcia K Johnson, Madhukar H Trivedi, Tyrone D Cannon, John Harrison Krystal, Philip Robert Corlett

- Trained in one trial, tested prospectively in another trial
- Model performance was weak (60%)
- Some (significant) models did not generalize!



Is my model *real* ?

Molecular Psychiatry (2016), 1–6

www.nature.com/mp

ORIGINAL ARTICLE

Testing a machine-learning algorithm to predict the persistence and severity of major depressive disorder from baseline self-reports

RC Kessler[1], HM van Loo[2], KJ Wardenaar[2], RM Bossarte[3], LA Brenner[4], T Cai[5], DD Ebert[1,6], I Hwang[1], J Li[5], P de Jonge[2], AA Nierenberg[7], MV Petukhova[1], AJ Rosellini[1], NA Sampson[1], RA Schoevers[2], MA Wilcox[8] and AM Zaslavsky[1]

- Predict persistence/severity of MDD
- Model performance was weak (AUC=0.6-0.7)
- But reiterates possibility of using historic/archival data to make predictions that might guide patient care

Table 2. AUC of Survey 1 risk scores based on ML models and logistic regression models predicting Survey 2 outcomes ($N = 1056$)

	AUC of risk scores based on	
	ML models	*Logistic models*
High persistence	0.71	0.68
High chronicity	0.63	0.62
Hospitalization	0.73	0.65
Disability	0.74	0.69
Suicide attempt	0.76	0.70

Abbreviations: AUC, area under the receiver operating characteristic curve; ML, machine learning.

Summary

		Condition (as determined by "Gold standard")		
		Condition Positive	Condition Negative	
Test Outcome	Test Outcome Positive	**True Positive**	**False Positive** (Type I error)	Positive predictive value = Σ True Positive / Σ Test Outcome Positive
	Test Outcome Negative	**False Negative** (Type II error)	**True Negative**	Negative predictive value = Σ True Negative / Σ Test Outcome Negative
		Sensitivity = Σ True Positive / Σ Condition Positive	**Specificity** = Σ True Negative / Σ Condition Negative	

- Choose most appropriate performance metric
- Always consider bias-variance tradeoff
- **Always keep training and testing data separate at all times**
  - Internal (k-fold) CV is a minimum
- Do you think your model is real? Why?