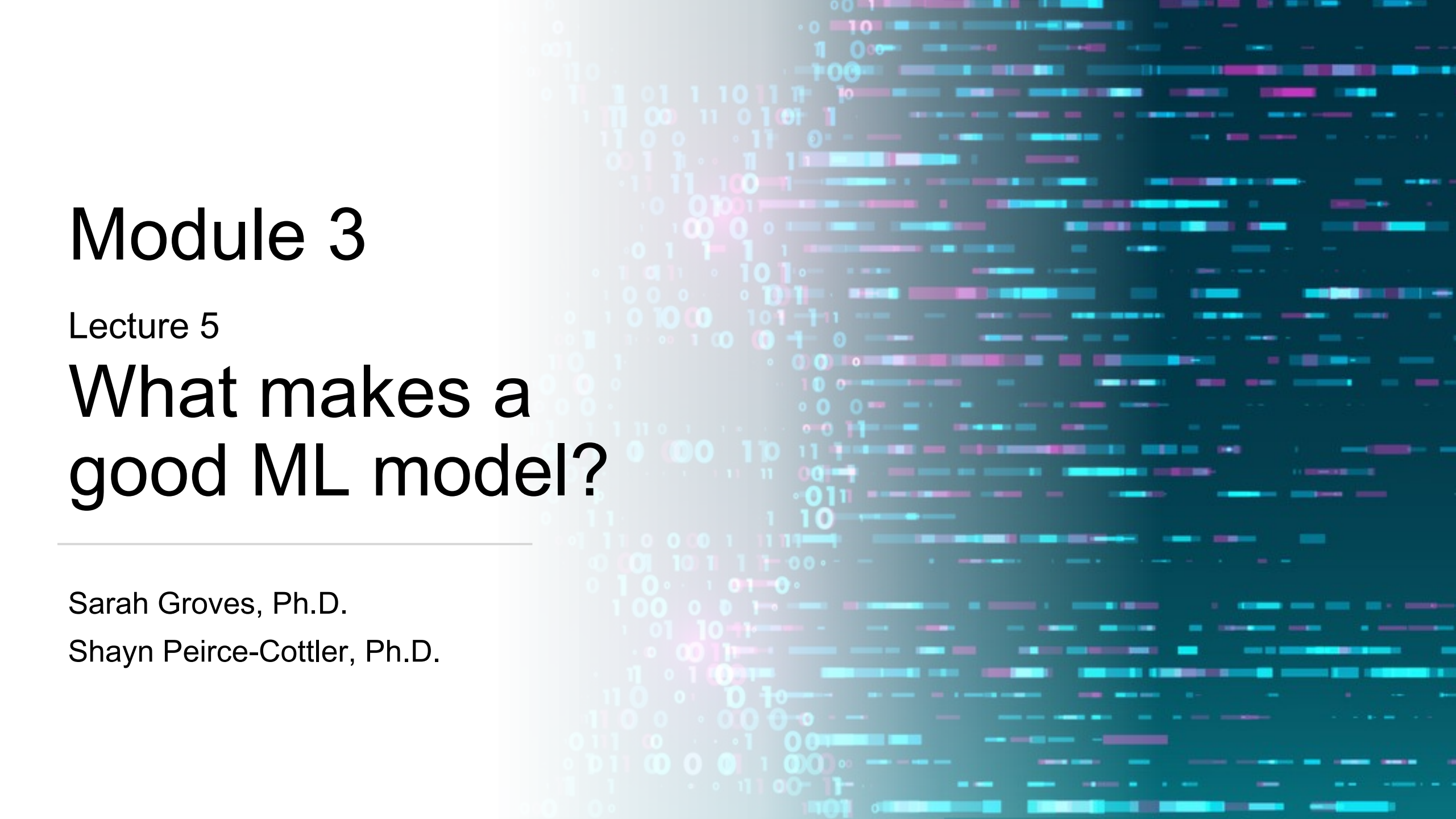
Module 3

Lecture 5

What makes a good ML model?

Sarah Groves, Ph.D.

Shayn Peirce-Cottler, Ph.D.



Announcements

- I looked over your notebooks, and they are looking good!
  - Many of you have already done train-test splitting and/or predictions with scoring
  - Today we will go into the “why” behind these methods and metrics
  - If you have not implemented a model yet, come see us for assistance!
- Coding workshop: November 14th, 5-6:30pm with pizza and snacks!
  - Let us know if you can attend via PIAZZA or EMAIL BY TOMORROW!

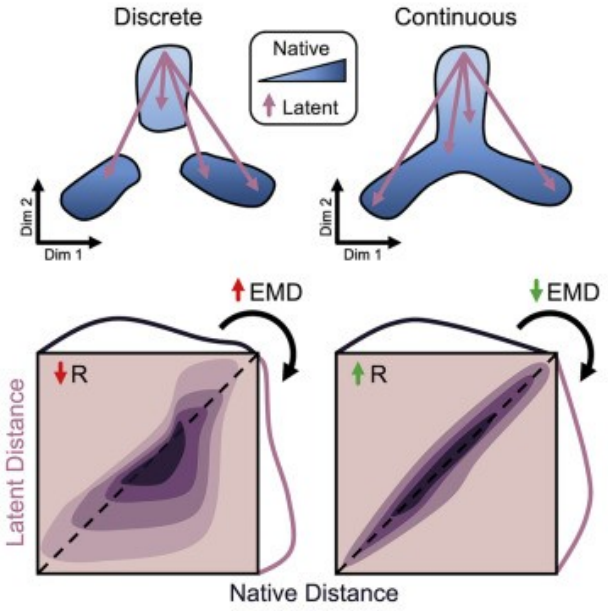
Machine Learning Model Validation

- *I know how to tell if my model is performing well or needs improvement.*
- *I understand why scoring the performance of my model is important for communicating the reliability of my results.*
- *I can split my data into two for training and testing the model.*

Is my model "good enough?"

Loss and scoring metrics across model types

- Regression
  - R^2: coefficient of determination
  - Errors: mean absolute error, mean squared error, etc.
  - Visualization of residual: PredictionErrorDisplay class
- PCA
  - Sanity check: How much variance is explained by PC1 and PC2? How many components do you need to explain a large percentage of the variance? What genes make up each PC loading?
- Manifold learning
  - Active area of research; often un-validated
- Classification
  - Accuracy score
  - Confusion matrix
  - Precision, recall, AUC
- Clustering
  - Evaluate the model without ground truth: silhouette coefficient
  - Compare score against a d

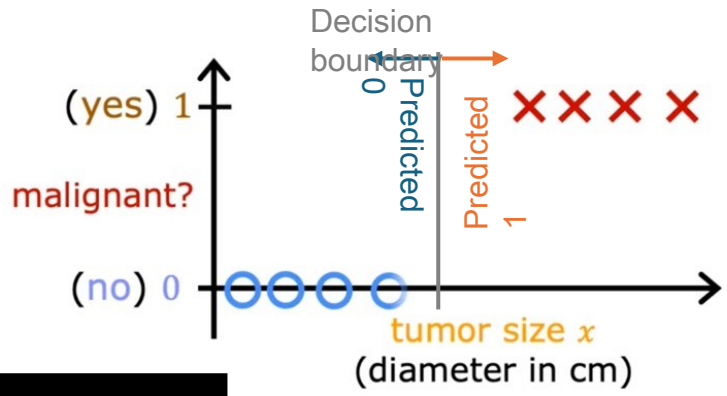


Confusion Matrix

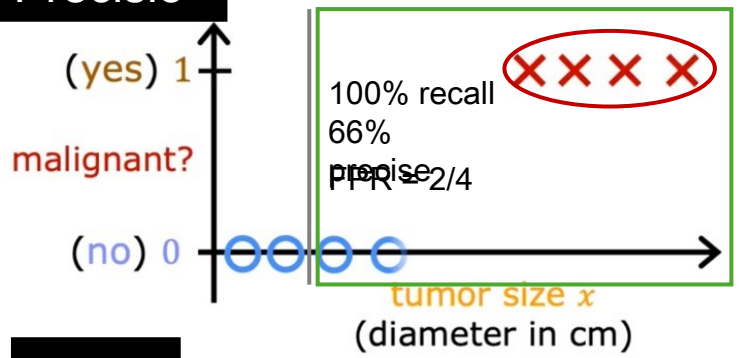
	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

Heiser et al.

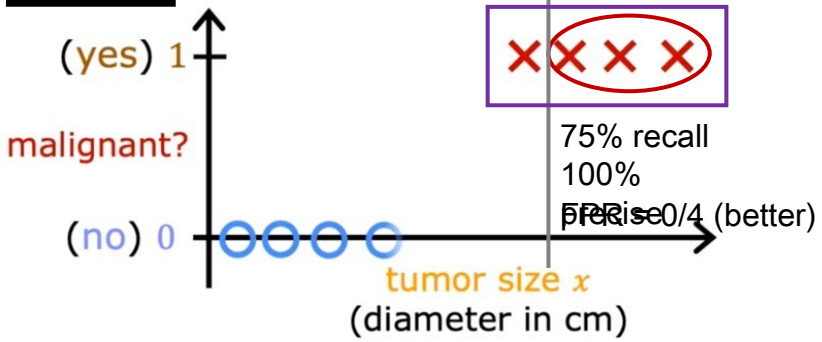
Precision and recall (and much more…)

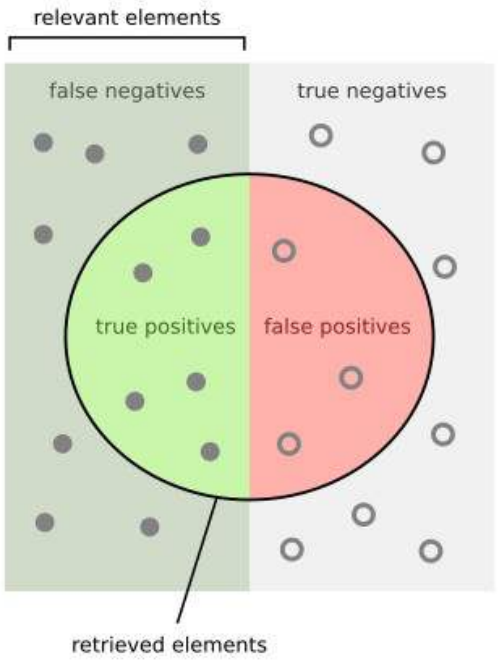


Precisio



Recal





There is a trade-off between precision and recall

In this example with a clear best boundary in between these examples, you can get 100% recall and precision!

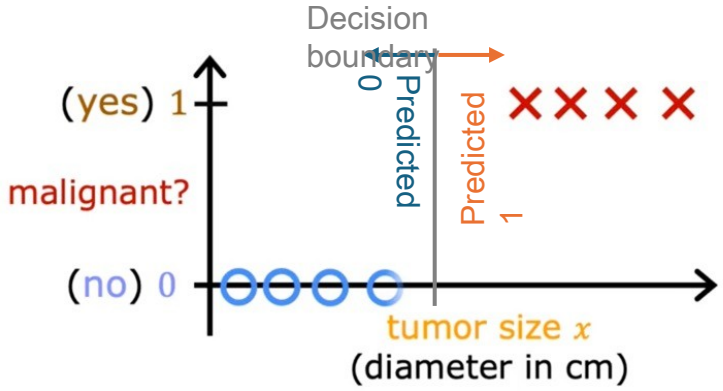
	Predicted condition		Informedness, bookmaker informedness (BM) $= \mathrm{TPR} + \mathrm{TNR} - 1$	Prevalence threshold (PT) $= \frac{\sqrt{\mathrm{TPR} \times \mathrm{FPR}} - \mathrm{FPR}}{\mathrm{TPR} - \mathrm{FPR}}$
Total population $= \mathrm{P} + \mathrm{N}$	**Predicted positive**	**Predicted negative**		
Real Positive (P) [a]	**True positive** (TP), hit[b]	**False negative** (FN), miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{\mathrm{TP}}{\mathrm{P}} = 1 - \mathrm{FNR}$	False negative rate (FNR), miss rate, type II error [c] $= \frac{\mathrm{FN}}{\mathrm{P}} = 1 - \mathrm{TPR}$
Real Negative (N)[d]	**False positive** (FP), false alarm, overestimation	**True negative** (TN), correct rejection[e]	False positive rate (FPR), probability of false alarm, fall-out, type I error [f] $= \frac{\mathrm{FP}}{\mathrm{N}} = 1 - \mathrm{TNR}$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{\mathrm{TN}}{\mathrm{N}} = 1 - \mathrm{FPR}$
Prevalence $= \frac{\mathrm{P}}{\mathrm{P} + \mathrm{N}}$	Positive predictive value (PPV), precision $= \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} = 1 - \mathrm{FDR}$	False omission rate (FOR) $= \frac{\mathrm{FN}}{\mathrm{TN} + \mathrm{FN}} = 1 - \mathrm{NPV}$	Positive likelihood ratio (LR+) $= \frac{\mathrm{TPR}}{\mathrm{FPR}}$	Negative likelihood ratio (LR−) $= \frac{\mathrm{FNR}}{\mathrm{TNR}}$
Accuracy (ACC) $= \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{P} + \mathrm{N}}$	False discovery rate (FDR) $= \frac{\mathrm{FP}}{\mathrm{TP} + \mathrm{FP}} = 1 - \mathrm{PPV}$	Negative predictive value (NPV) $= \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FN}} = 1 - \mathrm{FOR}$	Markedness (MK), deltaP (Δp) $= \mathrm{PPV} + \mathrm{NPV} - 1$	Diagnostic odds ratio (DOR) $= \frac{\mathrm{LR+}}{\mathrm{LR-}}$
Balanced accuracy (BA) $= \frac{\mathrm{TPR} + \mathrm{TNR}}{2}$	F_1 score $= \frac{2\,\mathrm{PPV} \times \mathrm{TPR}}{\mathrm{PPV} + \mathrm{TPR}} = \frac{2\,\mathrm{TP}}{2\,\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$	Fowlkes–Mallows index (FM) $= \sqrt{\mathrm{PPV} \times \mathrm{TPR}}$	*phi* or Matthews correlation coefficient (MCC) $= \sqrt{\mathrm{TPR} \times \mathrm{TNR} \times \mathrm{PPV} \times \mathrm{NPV}} - \sqrt{\mathrm{FNR} \times \mathrm{FPR} \times \mathrm{FOR} \times \mathrm{FDR}}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP}}$

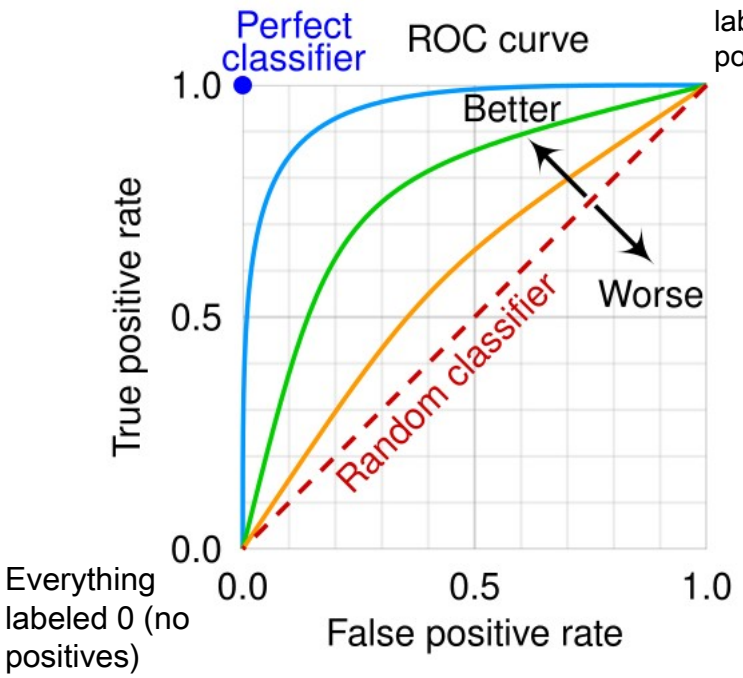
Sources: [16][17][18][19][20][21][22][23] view · talk · edit

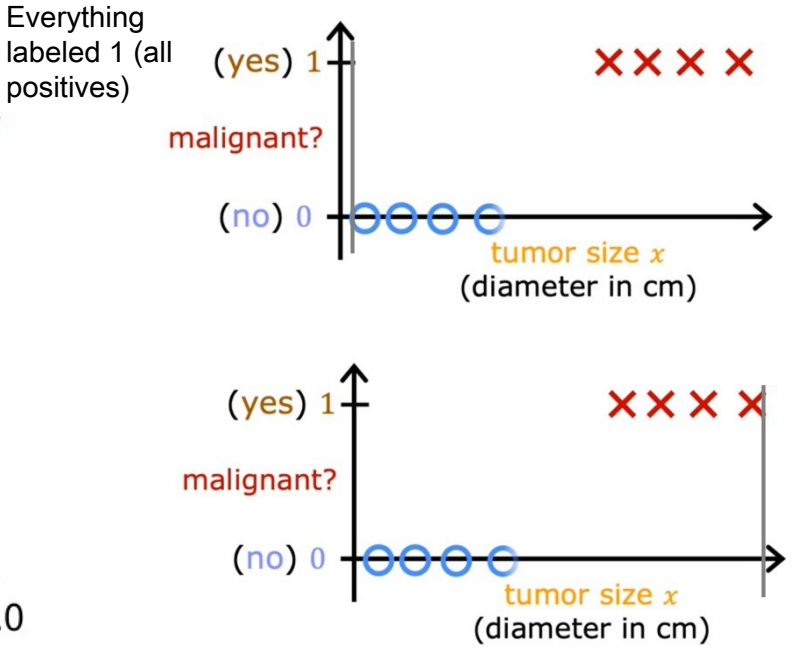
There is a trade-off between TPR/recall and FPR!

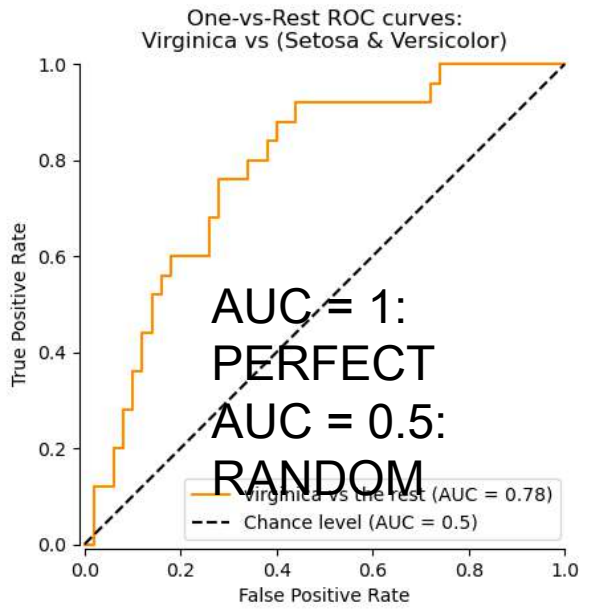
Area under the curve of the ROC plot

- Two parts to classification: (1) make the model predictions and then (2) *decide on a cutoff/decision boundary*
- After fitting your model and making predictions, you can sweep the decision boundary over different values to see the tradeoff between true positive rate (recall) and false positive rate
- This is visualized as a receiver-operating characteristic (ROC) curve









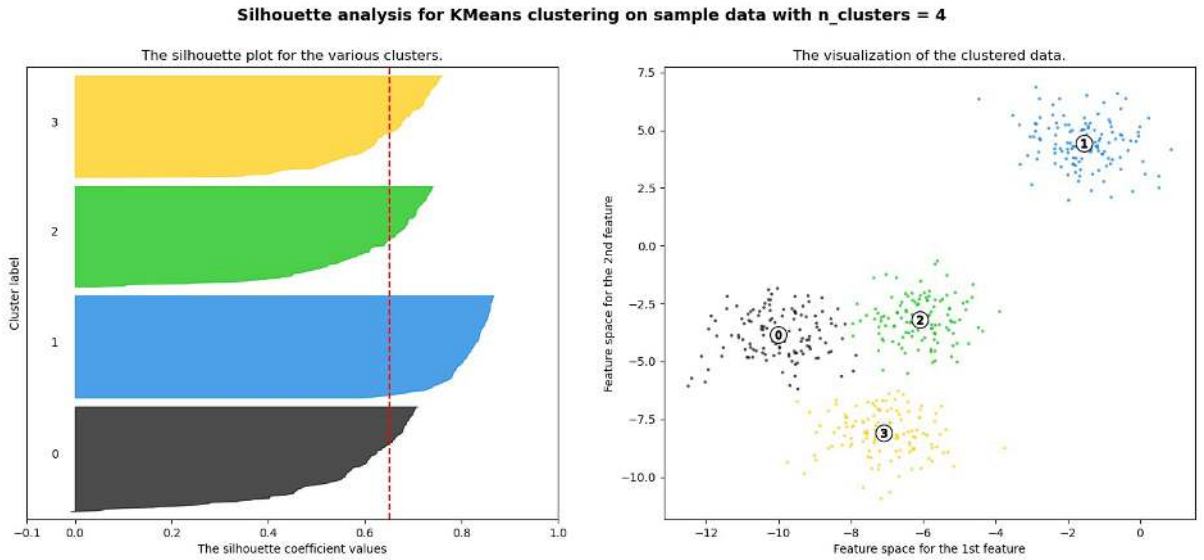
Scoring unsupervised clustering (when you don't know the "right" answer)

- Unlabeled data means we can't compare predictions to known labels
- **Silhouette coefficient** compares the distance between clusters with the distances within clusters

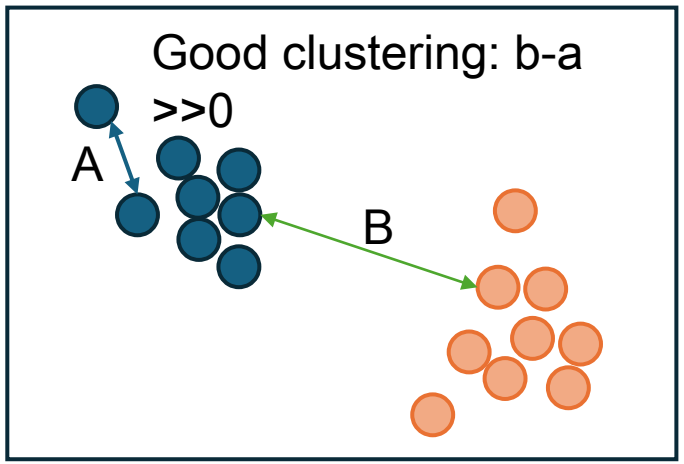
  - **a**: The mean distance between a sample and all other points in the same class.
  - **b**: The mean distance between a sample and all other points in the *next nearest cluster*.

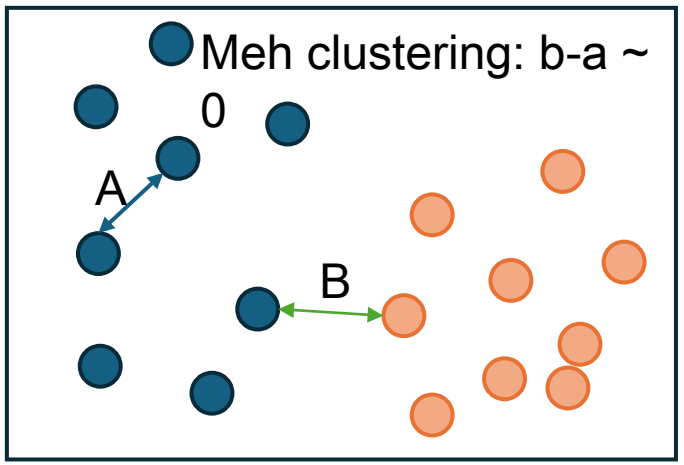
The Silhouette Coefficient *s* for a single sample is then given as:

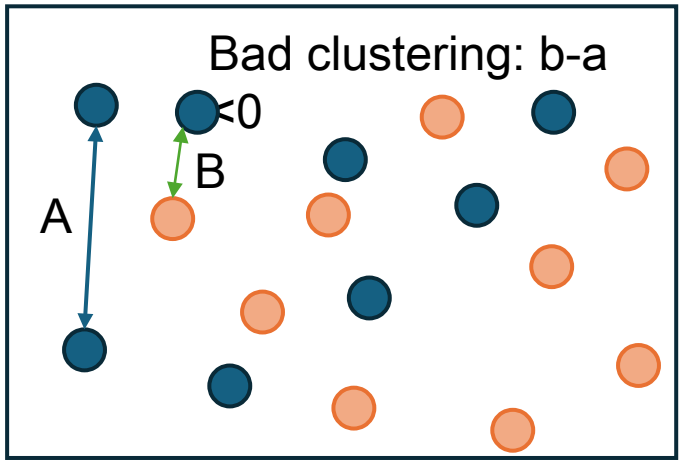
$$s = \frac{b - a}{max(a, b)}$$



For k-means: choose k that gives higher average silhouette score

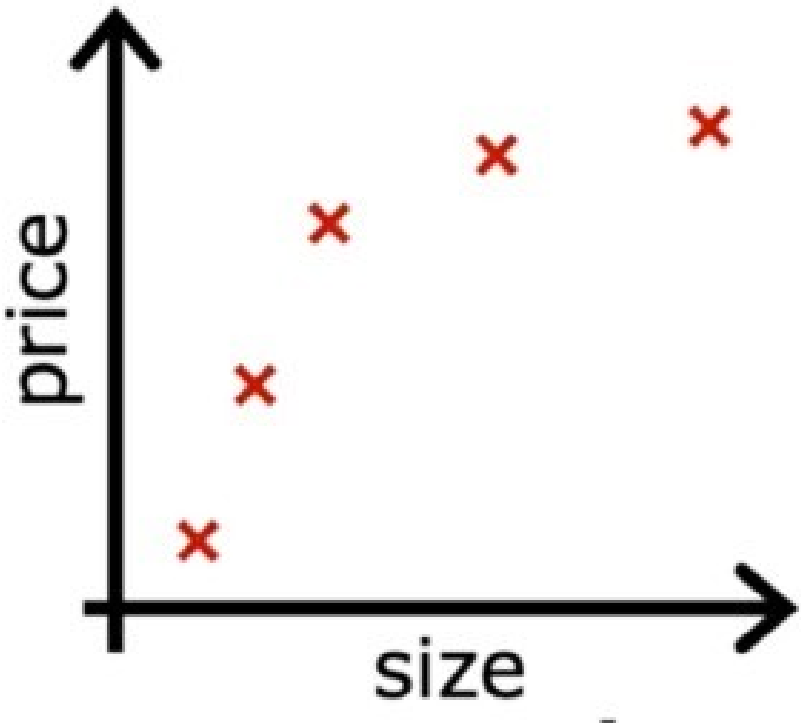






In sample versus out of sample error

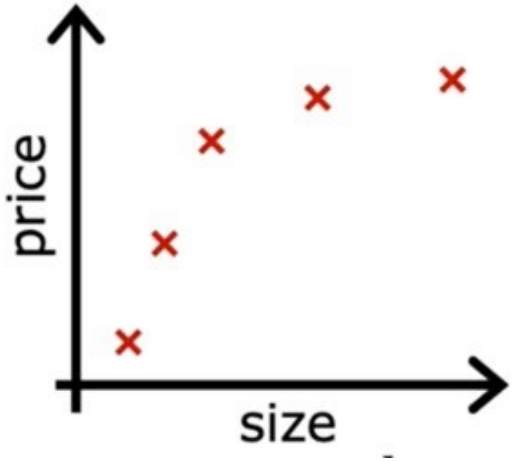
- Each of these scoring metrics can be used to approximate:
  - “in sample error”– the error in the predictions made for the same data the model was built on
  - “Out-of-sample error”– the error in predictions on new data that was not used to build/fit the model
- In sample: used with the .fit() or .fit_predict() method
- Out of sample: used with the .predict() method



In sample error: very low
Out of sample error: much higher

Why do we care about in sample AND out of sample error?

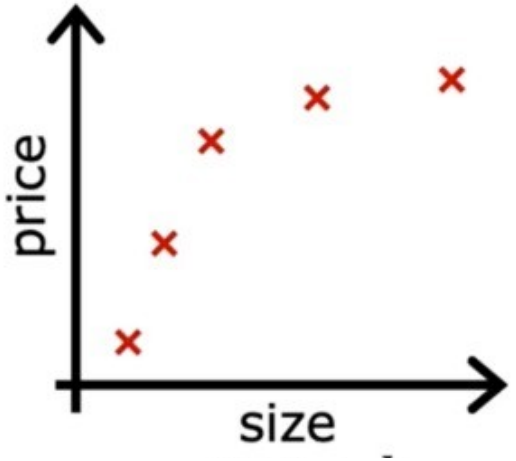
Picking the right model: in sample vs out of sample error



Low complexity model

Underfit model

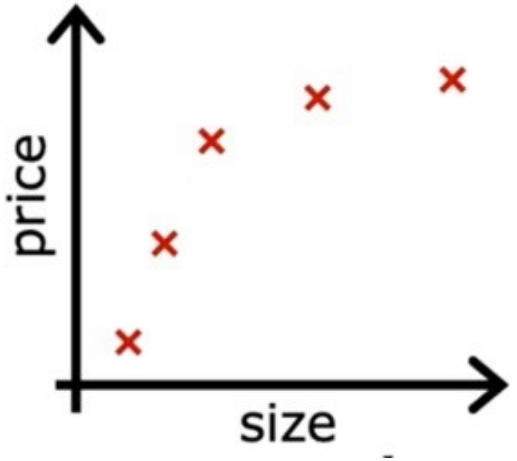
- Does not fit the training set well
- Not generalizable



Medium complexity model

Well-fit model

- Fits the training set well
- Generalizable to new data



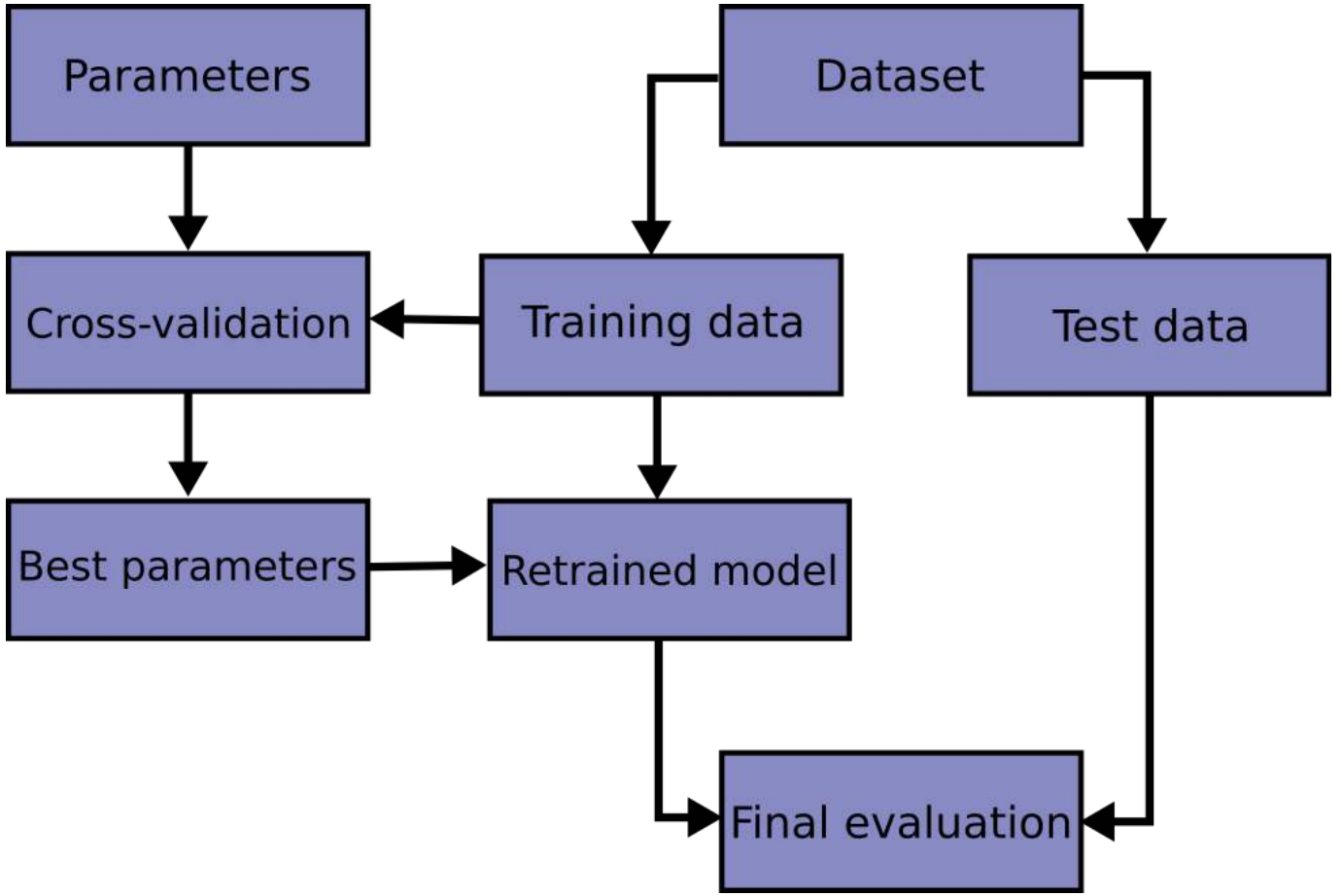
High complexity model

Overfit model

- Fits the training set extremely well
- Does not generalize well

The train-validation-test split

- Sklearn.model_selection.train_test_split
- **Training set:** Build the model on a subset of the data
- **Validation set**: Tune the model to work better by changing "hyperparameters" of the model
- **Test set**: Predict using the final version of the model on unseen data to get a true out-of-sample error estimate
- You should use train-test split *before* fitting your model to any data (or use the provided test set)
  - Make sure you apply the same preprocessing steps to the training and testing data (rescaling, subsetting to genes and cancer type, etc)
- Stratified train-test split



https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

Jupyter notebooks: model scoring and validation

- For check-in #3: you are expected to have
  - A **train-test split** of the data or **use the new dataset** posted as your validation dataset if you already fit a model to the full data! (We are not requiring the full train-validate-test split unless you want to do so)
  - You should **predict** on the new dataset and then **score** using sklearn.metrics
  - **If supervised ML:**
    - you can get a measure of how good your model does on the training set and validation set and use these results to decide if your model is overfitting (TRY AN ROC PLOT!)
    - **If regression, report multiple scoring metrics**
  - **If unsupervised clustering:**
    - Metrics should be applied across different numbers of clusters and the best model with n clusters is chosen based on the optimum of the metric (TRY A SILHOUETTE SCORE)
  - You should have some kind of **visualization** of your scoring metric(s)

Homework

- Let us know by tomorrow if you can attend the coding workshop next week!
- Third notebook check-in due on Saturday 11:59pm **November 8**!
  - You should have predictions on a test dataset and scoring for both training and test data (in the model validation section)
  - You should have a written section about what the outcomes of the scoring metrics mean (Is your model good enough? How do you know? Is it overfitting?)
- Final notebooks due on Thursday 9:30am November 13
  - These will include the final sections of the notebook– you should explore the literature to see if you can **validate your model results externally** (i.e. does the literature draw the same conclusions as you?)
  - What do your model predictions mean for **answering your project question**? What would be the **next steps** for making your model better? (next class)
  - **Ethics** and **conclusions** section of the notebook