University of Pittsburgh | School of Public Health

More statistical learning tools
- Ensemble learning
- Model averaging and SuperLearner

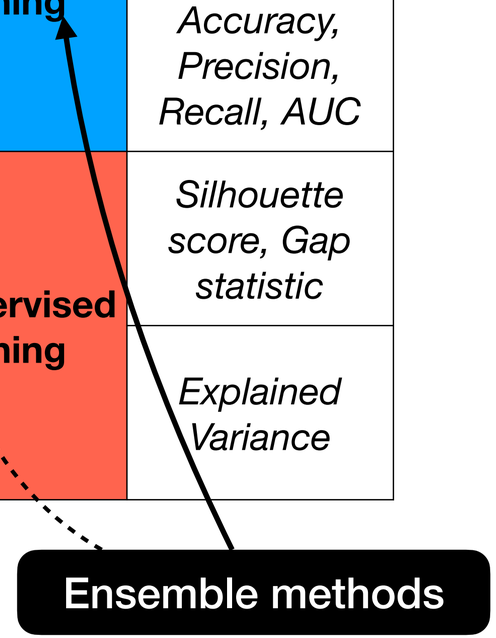
BIOST 2155

Class 11
November 20th, 2026

CLASS 1

Mapping what we have learnt so far…

	Features (input)	Targets/ labels (output)	Task type	ML type	Metric
Prediction of brain volume from gestational age	✅	✅ (Numeric)	Regression	**Supervised learning**	*RMSE, R-squared*
Classification of heart failure using ultrasound or magnetic resonance imaging	✅	✅ (Categorical)	Classification		*Accuracy, Precision, Recall, AUC*
Patient health profiles using wearable device data	✅	🚫	Clustering	**Unsupervised learning**	*Silhouette score, Gap statistic*
Compressing information from 30 features to study similarities/differences in benign and malignant breast tumors	✅	🚫	Dimensionality reduction		*Explained Variance*



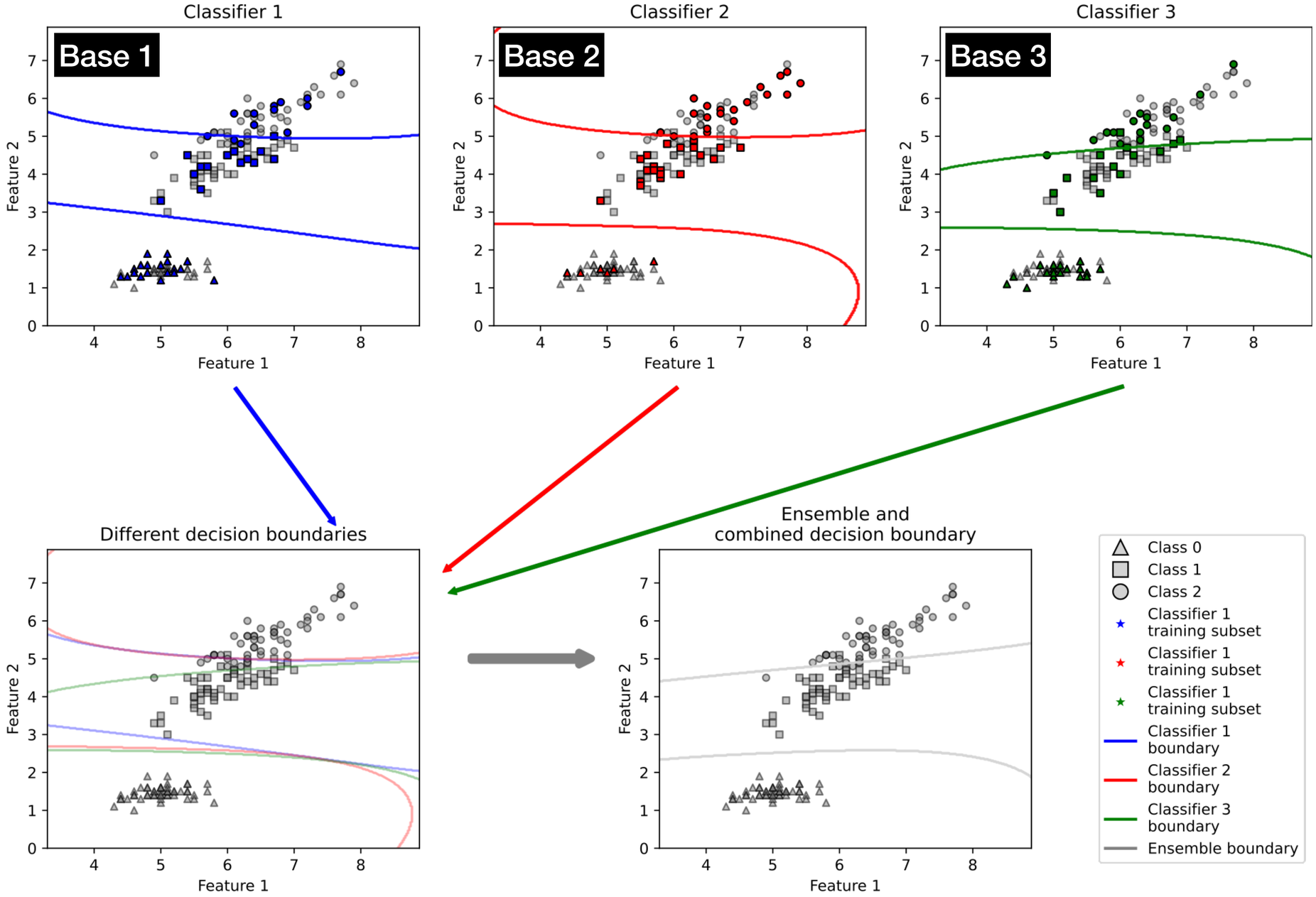
Ensemble learning: what?

E pluribus unum

- Ensemble methods use **multiple learning algorithms** to obtain one final prediction.
- Predictive **performance of ensemble methods is better** than performance obtained from any of the **constituent learning algorithms alone**.
- A **collection of individual learning algorithms** can not only outperform other algorithms in practice but also has been shown to be theoretically optimal (will skip this bit)
- Although ensemble methods fall primarily within supervised learning tasks, there are approaches for unsupervised learning tasks as well (will skip this bit).

Ensemble learning aims to reduce variance and increases accuracy

An ensemble of classifiers usually has smaller classification error than base models.



Sourced from: https://en.wikipedia.org/wiki/Ensemble_learning

Ensemble learning: why?

Additional benefits

- **Ensemble methods outperform individual (base) learning algorithms.**
- By combining a set of individual learning algorithms using a metalearning algorithm, **ensemble methods can approximate complex functional relationships that one or more of the base algorithms may have missed**.
- When the true functional relationship is not in the set of base learning algorithms, ensemble methods approximate the true function well.
- There are **three major classes** of strategy for ensemble learning.

Ensemble learning: how?

Common strategies

- **Bagging** — used mostly for trees, can be used for other ML models too
  - Robust against outliers
  - Can be done in parallel
- **Boosting** — used mostly for trees, can be used for other ML models too
  - Not robust against outliers
  - Can only be done sequentially
- **Stacking**
  - Uses a metalearning algorithm to combine the set of learners
  - Not robust against outliers
  - Can be done in parallel

Bagging combines models using a simple, democratic process (like averaging or voting).

Stacking combines models by training a new model (the meta-model) to learn the best way to combine their predictions


Stacking
Weighted ensembling
Superlearning

Ensemble learning: stacking

Approach 1: Model averaging

- Model selection: we have already discussed model selection
  - Best subset selection
  - Lasso/Ridge/Elastic net
- More models, more problems: when there are many models under consideration, the probability of selecting the correct model may be very small, even when the true unknown model (that generated the data we are analyzing) is among the models under consideration.

- Often there will be a bunch of models which perform similarly:
  - Why force ourselves to arbitrarily pick one model?
  - Could we somehow use more than one model?

Model averaging: how

Weighted combination of models

- Suppose we have an ‘ensemble’ of models: $s_1, s_2, \ldots, s_M$
- Each model s_k in the ensemble yields prediction $\hat{Y}_k$ for a feature input $\boldsymbol{x}$:

$$\hat{Y}_k = s_k(\boldsymbol{x}),\ 1 \leq k \leq M$$

- We obtain a weighted combination of model predictions:

$$\hat{Y}_E = \sum_{k=1}^{M} w_k \hat{Y}_k$$

- For regression, we look at **weighted averages**. For classification, we look at **weighted voting**.
- **We don’t average model parameters, only model outputs.**

Model averaging: why?

A toy example…

- Suppose we are trying to estimate parameter μ using M different estimates $\hat{\mu}_1, \hat{\mu}_2, \ldots, \hat{\mu}_M$

- Suppose we consider the (unweighted) average as our 'ensemble' estimate:

$$\hat{\mu}_E = M^{-1} \sum_{k=1}^{M} \hat{\mu}_k$$

$$RSS(\hat{\mu}_E) = \left(\mu - \hat{\mu}_E\right)^2 = M^{-1} \sum_{k=1}^{M} \left(\mu - \hat{\mu}_k\right)^2 - V = M^{-1} \sum_{k=1}^{M} RSS(\hat{\mu}_k) - V,$$

$$\text{where } V = M^{-1} \sum_{k=1}^{M} \left(\hat{\mu}_k - \hat{\mu}_E\right)^2$$

Models with small RSS will help!

Diverse models (will increase V) will help!

Model averaging only helps if the model ensemble makes different predictions.

Including really bad models to increase diversity is unlikely to help - raises average risk more than it increases diversity

Overall risk of ensemble = average of individual risk - ensemble diversity

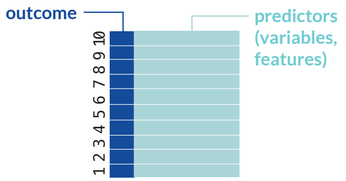
Ensemble learning: stacking

Approach 2: SuperLearner

Superlearning, or stacking, **weights** the results of **many individual statistical learning algorithms** to create an optimal overall prediction algorithm. Superlearner predictions are expected to perform at least as well as any of the individual learners in large sample sizes.

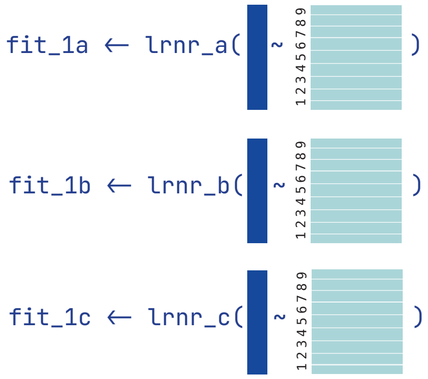
STEP 1

Split data into 10 blocks in preparation for 10-fold cross validation.



STEP 2

Train multiple base learners on 9 of the 10 blocks of data.

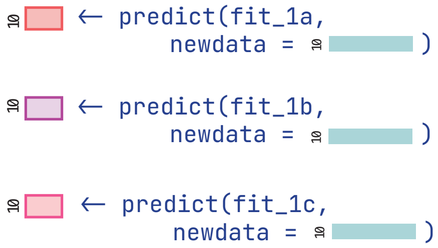


Base learners can include any number of parametric or non-parametric supervised statistical learning algorithms.

An example of three base learners for a binary outcome could be random forest, gradient boosting, and logistic regression.

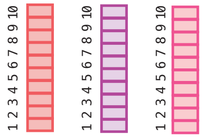
STEP 3

Obtain predictions from each base learner for the held-out block of data.



STEP 4

Repeat until each of the 10 blocks have served as the hold-out data and you have three sets of cross-validated predictions spanning the full data set.



STEP 5

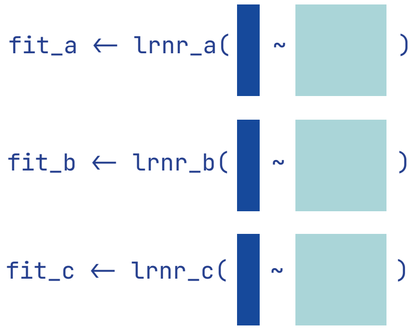
Using a new learner, a metalearner, predict the outcome using the three sets of cross-validated predictions.



The metalearner can be as simple as a generalized linear model. As with any statistical learning algorithm, the choice reflects a loss function we want to minimize.

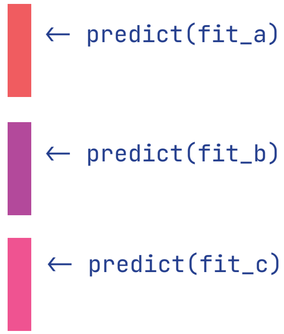
STEP 6

Fit the base learners on the entire data set.



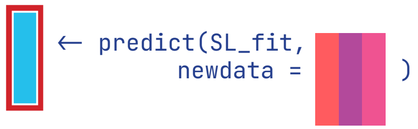
STEP 7

Obtain predictions from the full data set for each learner.



STEP 8

Use the coefficients from Step 5 to weight the full data predictions from Step 7. These are the final superlearner predictions.



The final superlearner predictions are a weighted combination, or ensemble, of the base learners' predictions.

STEP 9

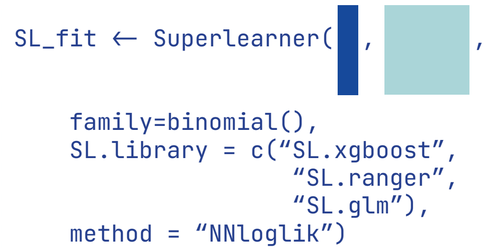
To predict on new data, use the base learner fits to obtain base learner predictions (similar to Step 7), then input the base learner predictions into the metalearner fit (similar to Step 8) to obtain the final prediction.

EVALUATION

To test the prediction cabability of the superlearner algorithm and prevent overfitting, the entire algorithm (Steps 1-8) could be cross-validated.

APPLICATION

There are several R packages to implement superlearning. This example uses the `SuperLearner` package to create a superlearner model for a binary outcome with three learners: gradient boosting (`xgboost`), random forest (`ranger`), and logistic regression (`glm`) with a loss function/ metalearning step of negative log-likelihood (`method="NNloglik"`):



REFERENCES

Targeted Learning, Chapter 3: Superlearning by Eric Polley, Sherri Rose, and Mark van der Laan.

For a step-by-step tutorial with R code, explanations, and more references: www.khstats.com/blog/sl/superlearning.

By Katherine Hoffman - khstats.com

Post-class debrief

Today we discussed…

- Selecting one model is often random (user-dependent) and may be noisy.
- We can consider many different models and pick the best performing one…but it may be overfitting.

- Averaging over an ensemble (collection) of models is less random, reduces overfitting risks
- Ensemble risk = Average risk of all models in ensemble - diversity of predictions from ensemble.
- Need to balance between ‘good’ models with low risk but don’t contribute much to diversity and ‘worse’ models with higher risk but contribute to ensemble diversity.

- Model averaging averages (weighted) predictions from models in ensemble, while SuperLearner trains a super-model on predictions from models in ensemble.


That's all Folks!