AGENDA

The central idea of **ensemble learning** is building prediction models by combining the predictions of a number of simpler models.

- **Ensemble Methods**
- **Bagging**
- **Stacking**

CENTRAL IDEA

- Three individual taggers, each committing errors

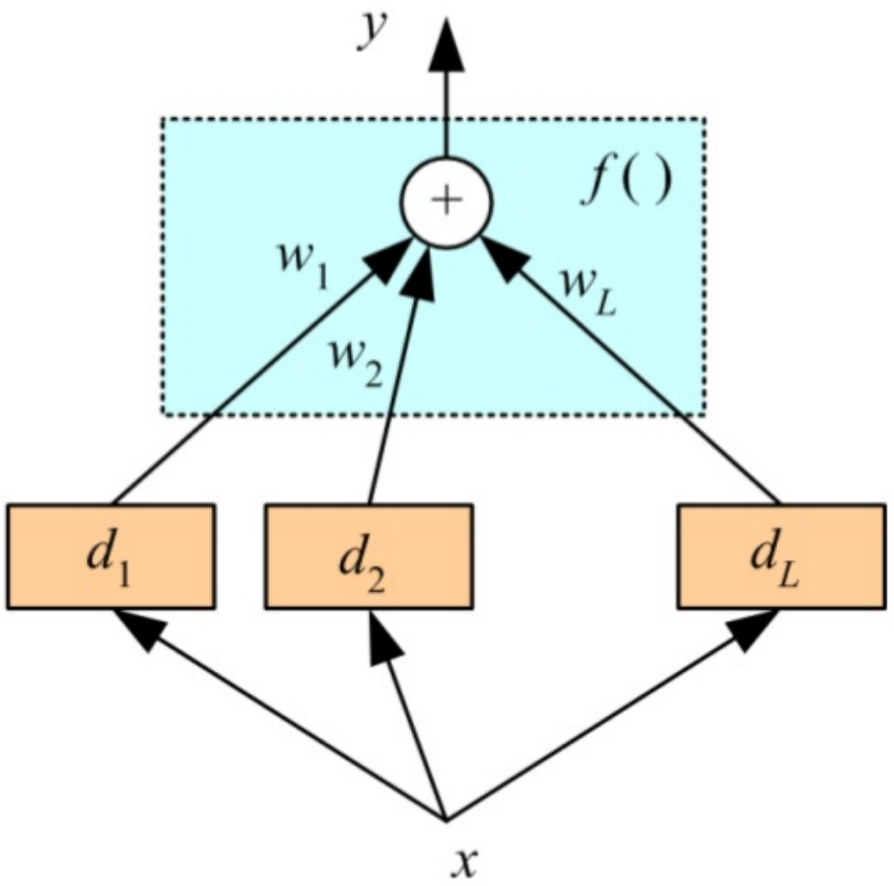
	John	gave	Mary	the	book	ACC
Tagger 1	V	V	N	DT	N	0.8
Tagger 2	N	N	V	DT	N	0.6
Tagger 3	N	V	N	PN	N	0.8
Majority	N	V	N	DT	N	1.0

- Average accuracy ≈ 0.73. Majority accuracy $= 1.0$
- Majority vote better than individual models

Diverse base classifiers — uncorrelated errors on new data points.

CENTRAL IDEA

An ensemble of classifiers is a set of classifiers whose individual outputs are combined in some way to classify new examples.



OTHER FUSION FUNCTIONS

Rule	Fusion function $f(\cdot)$
Sum	$y_i = \frac{1}{L}\sum_{j=1}^{L} d_{ji}$
Weighted sum	$y_i = \sum_j w_j d_{ji}, w_j \geq 0, \sum_j w_j = 1$
Median	$y_i = \text{median}_j d_{ji}$
Minimum	$y_i = \min_j d_{ji}$
Maximum	$y_i = \max_j d_{ji}$
Product	$y_i = \prod_j d_{ji}$

	C_1	C_2	C_3
d_1	0.2	0.5	0.3
d_2	0.0	0.6	0.4
d_3	0.4	0.4	0.2
Sum	0.2	**0.5**	0.3
Median	0.2	**0.5**	0.4
Minimum	0.0	**0.4**	0.2
Maximum	0.4	**0.6**	0.4
Product	0.0	**0.12**	0.032

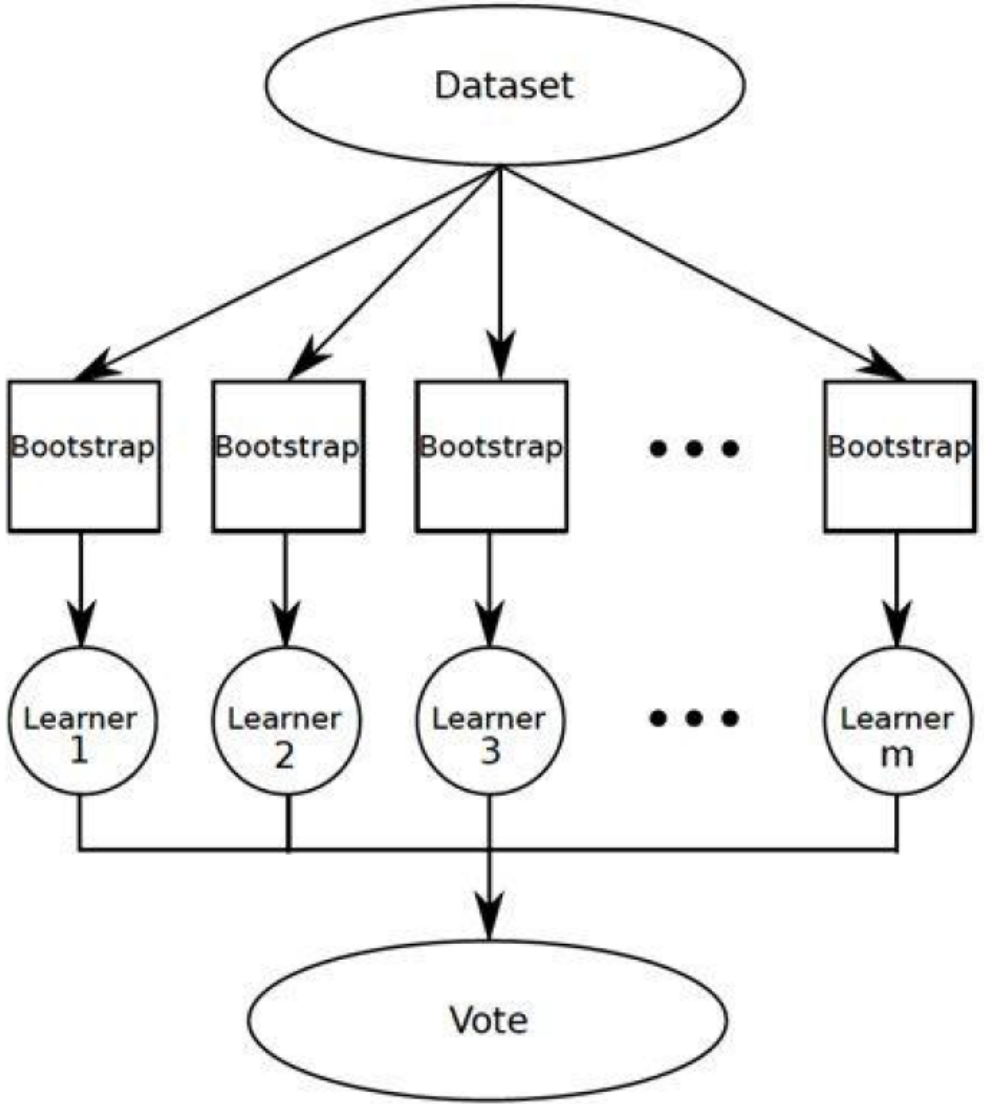
BOOTSTRAP AGGREGATION

- **Bagging** and **Random Forests** are two ensemble methods for classification that we discussed previously (also tbd).

- Broadly, ensemble methods consists of two distinct steps —

  i. Training a population of base learners from the data.
  ii. Combining them to form the composite prediction model.

BAGGING — VOTE-BASED



BAGGING

- Formally, train a model with a training data set $Z = \{(x_1, y_1), ..., (x_n, y_n)\}$ that yields prediction $\hat{f}(x)$ at input x.
- Obtain B bootstrapped samples Z^b, $b = 1, ..., B$ and train the model on each bootstrapped sample giving the prediction $\hat{f}^b(x)$.
- Let $\hat{f}(x, k)$ denote the proportion of the B classifiers that predict class k for input x.
- The bagging prediction $\hat{f}_{bag}(x)$ is then given by

$$\hat{f}_{bag}(x) = \arg\max_k \hat{f}(x, k).$$

MODEL AVERAGING

- Suppose we have a candidate set of M models $\mathcal{M}_m$, $m = 1, ..., M$ for our training set Z.
- The M models could be of the same type trained on different samples, or different models (like in your group exercise).
- Let $f(x)$ be the true output for a given input x and let $f\hat{(}x)$ be the aggregated prediction.
- Adopting a Bayesian perspective, the posterior distribution of $\hat{f}(x)$ is given by

$$P(\hat{f}(x)|Z) = \sum_{m=1}^{M} P(\hat{f}(x)|\mathcal{M}_m)P(\mathcal{M}_m|Z)$$

MODEL AVERAGING

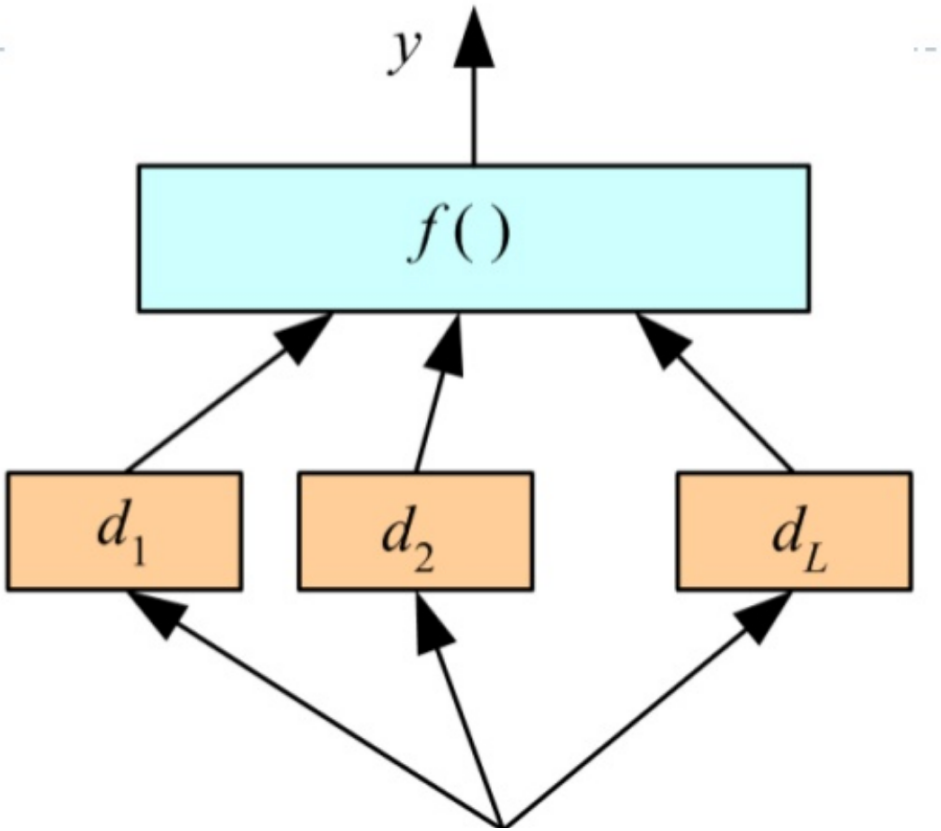
- The Bayesian approach is a weighted average of the individual predictions, with weights proportional to the posterior probability of each model.
- One could take a simple unweighted average of the predictions from each model, essentially giving equal probability to each model.
- Or more sophisticatedly, in some cases, one could weight the different models using the corresponding BIC scores.

STACKING

- **Stacking** is another approach to construct an ensemble classifier.
- Although an attractive idea, it is less widely used than bagging and boosting in practice.
- Unlike bagging and boosting, stacking may be (and normally is) used to **combine models of different types**.

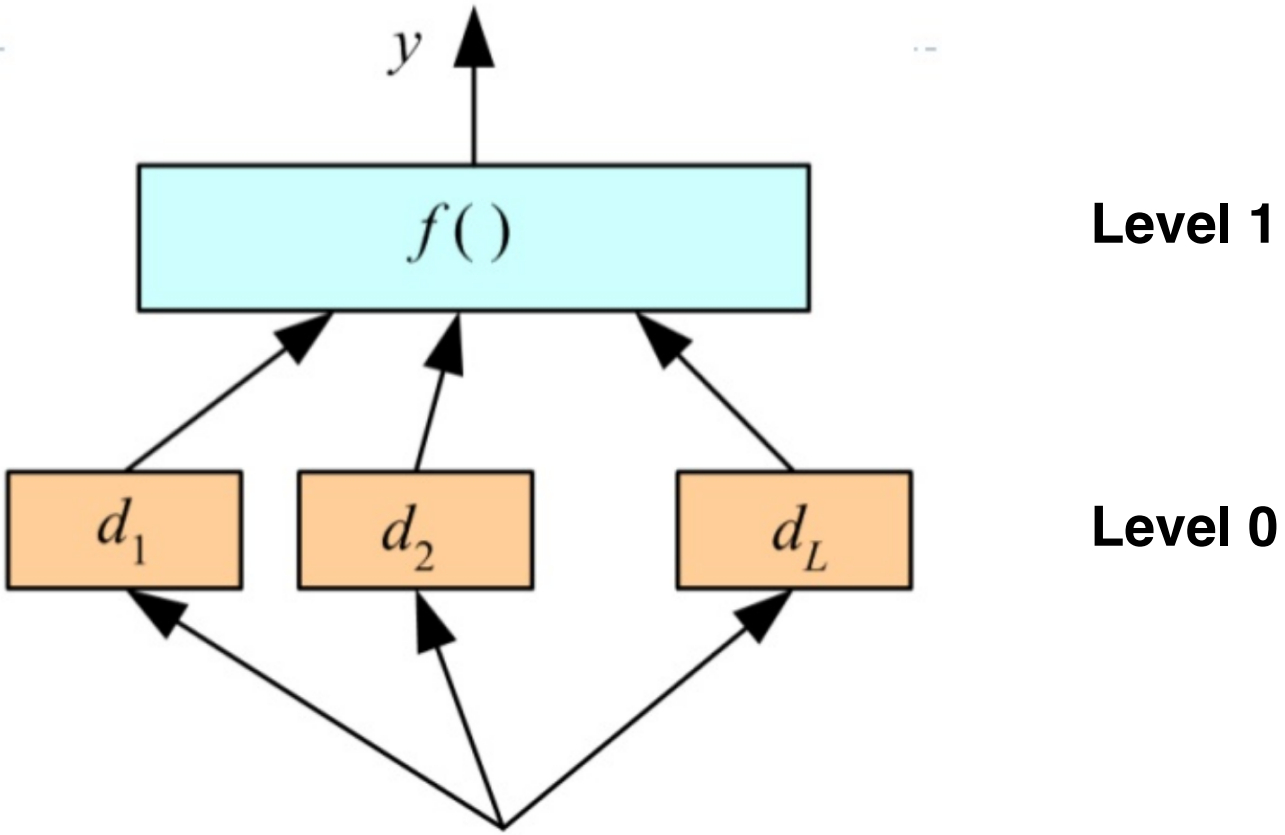
GENERAL IDEA BEHIND STACKING

- In stacked generalization (stacking), the combining of classifiers is **not restricted** to be a linear combination (such as voting), but instead is **another learner**.



GENERAL IDEA BEHIND STACKING

- At level-0, we use the original data and the base set of learners, and then, we use their outputs as level-1 data and we build a level-1 classifier.



MECHANICS OF STACKING

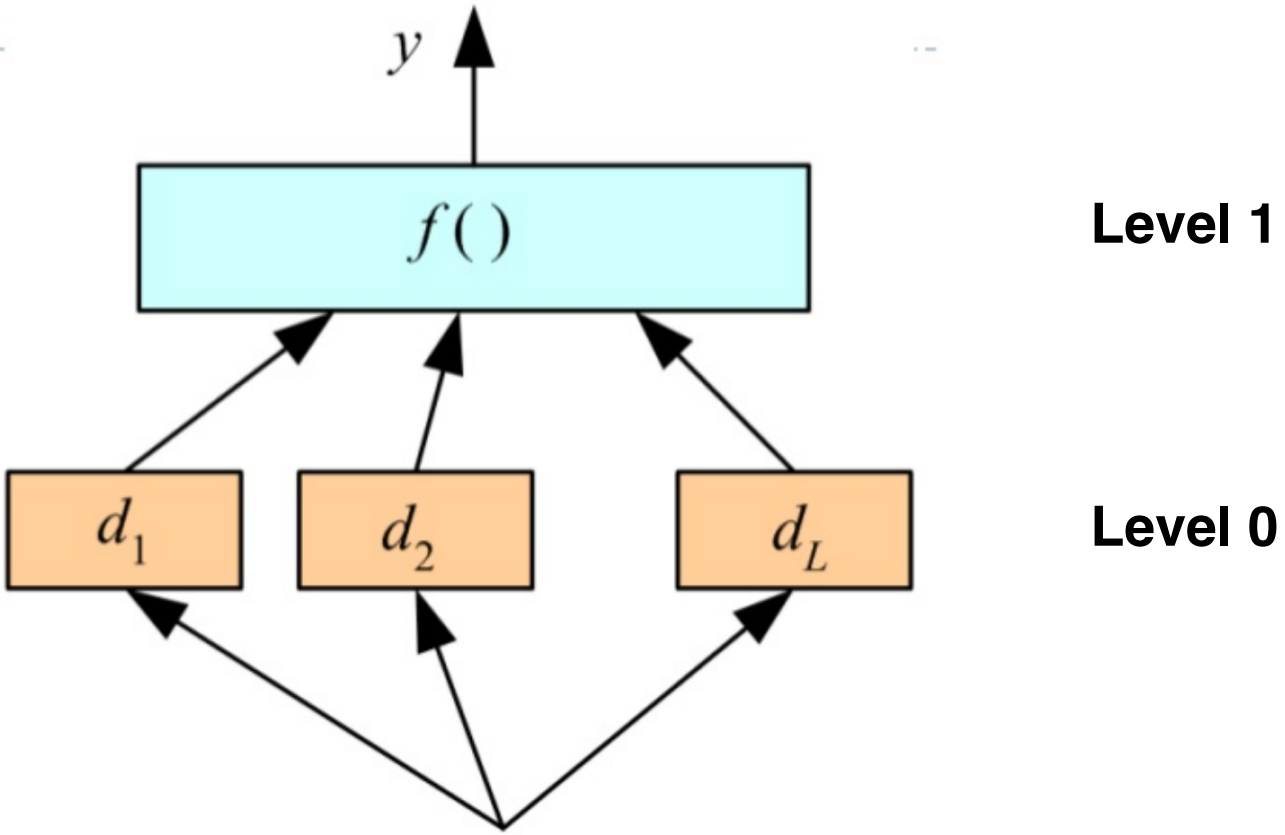
- Consider a dataset $\mathcal{L} = \{(x_1, y_1), ..., (x_N, y_N)$ where y_n is the class value and x_n is the set of predictors for the nth observation.
- Randomly split the data into J equal parts $\mathcal{L}_1, ..., \mathcal{L}_J$ and define $\mathcal{L}_j$ and $\bar{\mathcal{L}}_j = \mathcal{L} - \mathcal{L}_j$ as the test and training set for the j^{th} fold of a J-fold cross-validation.
- Given K classifiers (level-0 classifiers), invoke the k^{th} algorithm on the data in the training set $\bar{\mathcal{L}}_j$ to build a classification model $\mathcal{M}_k^{(-j)}$ for $k = 1, ..., K$.

MECHANICS OF STACKING

- For each instance $x \in \mathcal{L}_j$, the test set for the jth cross-validation fold, let $v_k^{(-j)}(x)$ be the prediction of model on instance x.
- Let z_{kn} denote the prediction of the k^{th} model on instance x_n, that is, $z_{kn} = v_k^{(-j)}(x)$.
- At the end of the entire exercise, we construct the dataset assembled from the outputs of all the models $\mathcal{L}_{CV} = \{(y_n, z_{1n}, ..., z_{Kn}), n = 1, ..., N\}$. This is level-1 data.
- Use some classification model on this level-data to build a level-1 classifier $\tilde{\mathcal{M}}$. This is the final step. Model $\tilde{\mathcal{M}}$ in conjunction with the base classifiers will classify any new input.

GENERAL IDEA BEHIND STACKING

- At level-0, we use the original data and the base set of learners, and then, we use their outputs as level-1 data and we build a level-1 classifier.



GROUP PROJECT

- Subsequent to the group project, we will attempt to build both a ***voting-based ensemble*** as well as a ***stacked ensemble*** and judge their performance on the given classification task.

REFERENCES

- *Some of the figures and concepts in this presentation are adapted from "Elements of Statistical Learning" (Springer, 2008), Hastie et al.*
- *"Ensemble Methods", Marina Santini, Uppsala University.*
- *"Stacked generalization: when does it work?" (1997) Ting, K. M. and Witten, I. H. Proc. International Joint Conference on Artificial Intelligence (IJCAI).*
- *"Is combining classifiers with stacking better than selecting the best one?" (2004) Dzeroski S, Zenko B. Machine Learning 54(3):255273*