Machine Learning 520
Advanced Machine Learning

Lesson 5: Stacking and Blending

Today's Agenda

- Combine classifiers
- Stacking
- Blending

Learning Objectives

By the end of this session, you should be able to:

- Describe the theory of how ensemble learning reduces errors.
- Use stacking to improve model performance.
- Use blending to improve model performance.

Weak Learner & Strong Learner

> A **weak learner**: it can make predictions (slightly) **better than random guessing**.
  - Weak learners have high bias and cannot solve hard learning problems.
  - e.g., naïve Bayes, logistic regression, decision stumps (decision trees of depth 1)

> A **strong learner**: it has **arbitrarily small error rate**.
  - Strong learners are our goal of machine learning.
  - e.g., random forest, deep neural networks

Can we turn weak learners into a strong one?

Weak Learner 1

Weak Learner 2

Weak Learner 3

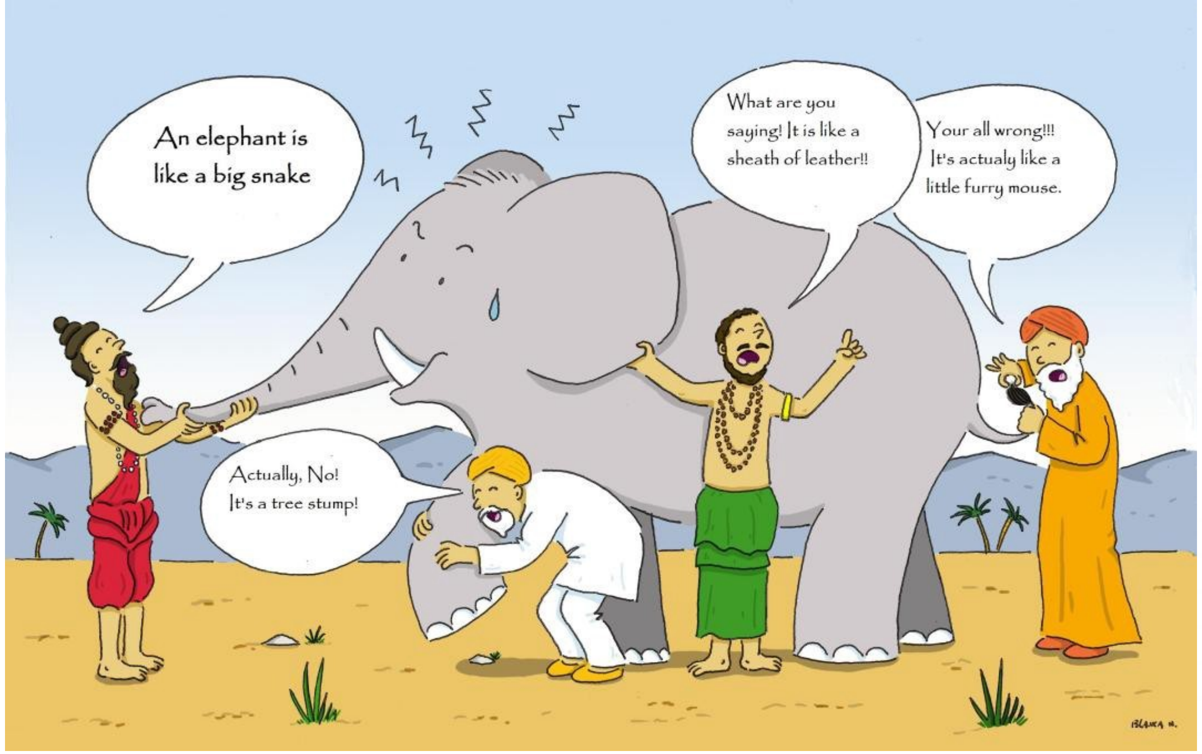
⋮

Weak Learner K

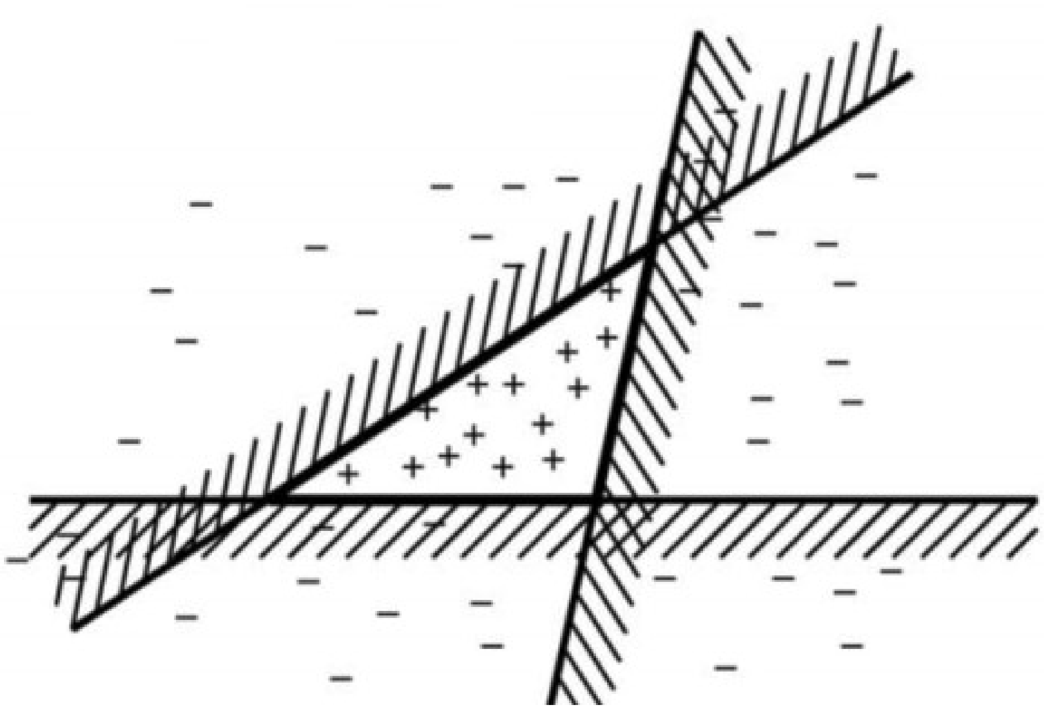
Strong Learner

YES, ENSEMBLE LEARNING

Ensemble Learning Intuition



Fable of blind men and elephant



Combine 3 linear classifiers

Ensemble Learning

- Instead of learning a single classifier, we learn a set of classifiers.

  How do we learn a set of classifiers?

- Combine the predictions of multiple classifiers to produce the final prediction.

  How do we combine all the classifiers?

Why do ensembles work?

- Suppose there are 25 classifiers where each classifier has an error rate of 0.35.
  - Assume classifiers are **independent**: a mistake from one classifier does not depend on the predictions from other classifiers.
  - In practice they are NOT completely independent.

- *Majority Voting*: The ensemble makes a wrong prediction if the majority of the classifiers predict the wrong prediction.

- What is the probability that the ensemble makes a wrong prediction? (hint: 13 or more classifiers make wrong predictions).

Why do ensembles work?

- Suppose there are 25 base classifiers
  - Each classifier has error rate, $\varepsilon = 0.35$
  - Assume classifiers are independent
  - Probability that the ensemble classifier makes a wrong prediction:

$$\sum_{i=13}^{25} \binom{25}{i} \varepsilon^{i} (1-\varepsilon)^{25-i} = 0.06$$

How it works

- Majority vote
- Suppose we have 5 completely independent classifiers…
  - If accuracy is 70% for each
    - $(.7^5)+5(.7^4)(.3)+ 10\ (.7^3)(.3^2)$
    - **83.7% majority vote accuracy**
  - 101 such classifiers
    - **99.9% majority vote accuracy**

Note: Binomial Distribution: The probability of observing *x* heads in a sample of *n* independent coin tosses, where in each toss the probability of heads is *p*, is

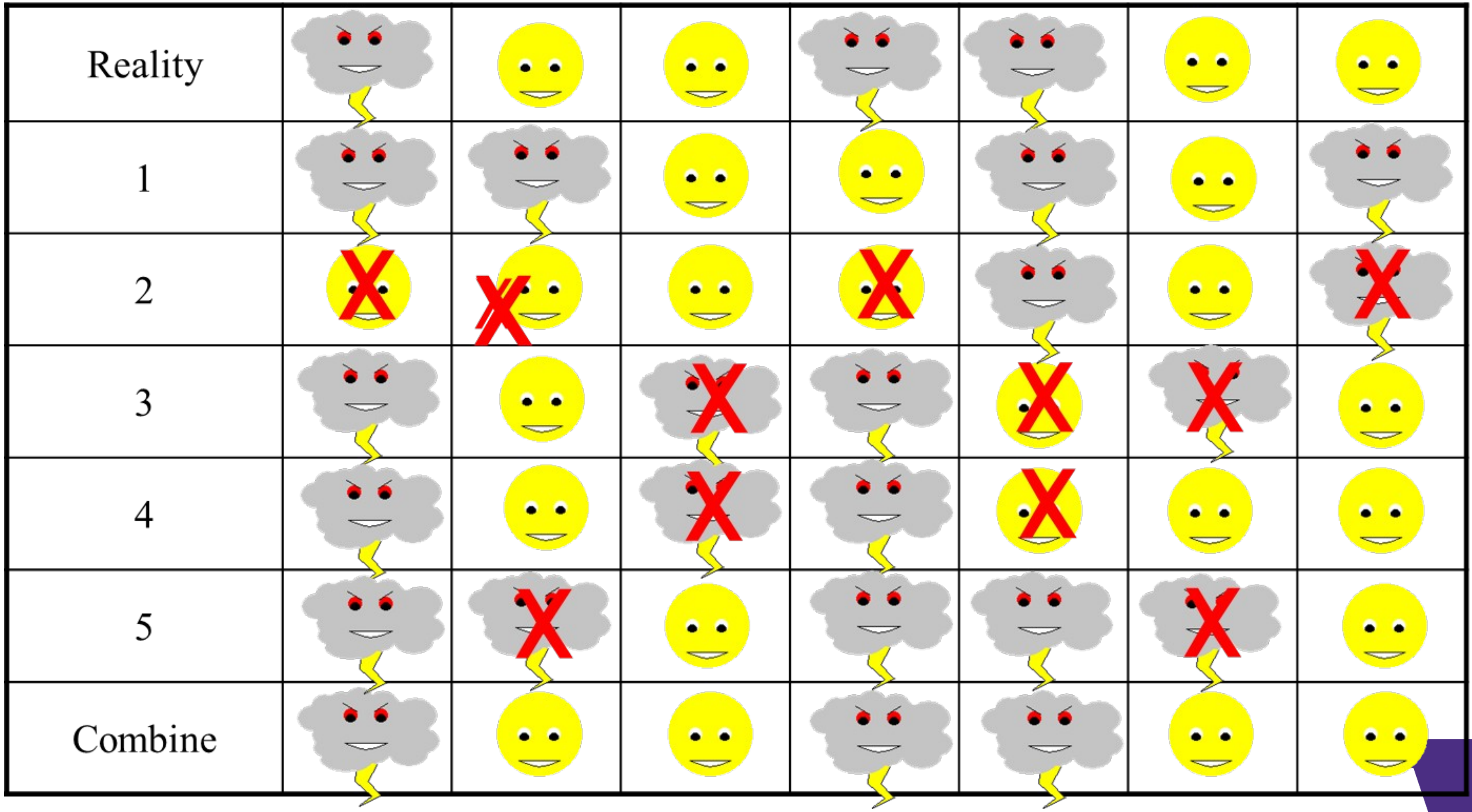
$$P(X = x|p, n) = \frac{n!}{r!(n-x)!} p^x (1-p)^{n-x}$$

Value of Ensembles

- "No Free Lunch" Theorem
  - No single algorithm wins all the time!

- When combing multiple independent and diverse decisions each of which is at least more accurate than random guessing, random errors cancel each other out, correct decisions are reinforced.

Example: Weather Forecast



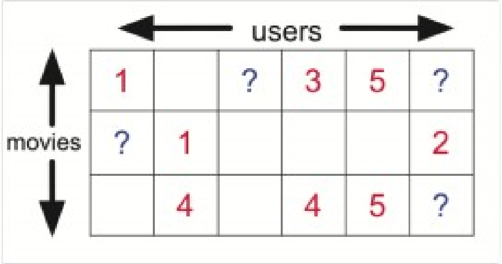
Ensemble Learning in Netflix Prize

Machine learning competition with a $1 million prize

Leaderboard

Display top 20 leaders.

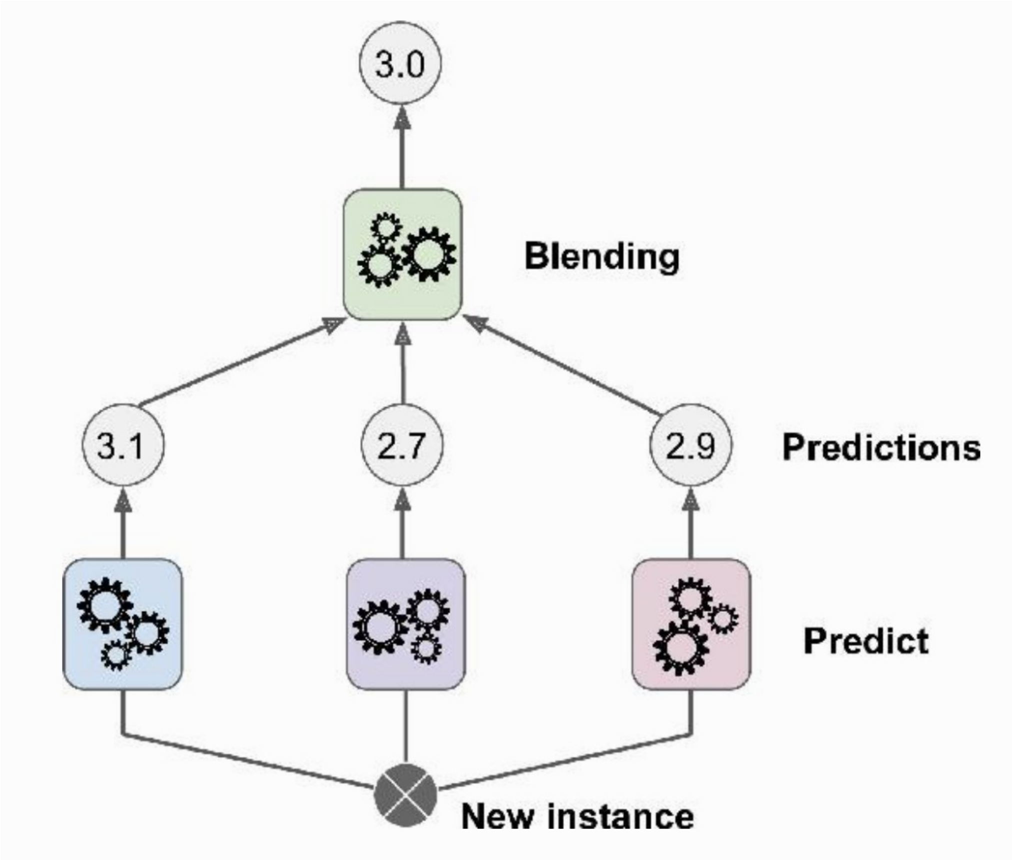
Rank	Team Name	Best Score	% Improvement	Last Submit Time
1	The Ensemble	0.8553	10.10	2009-07-26 18:38:22
2	BellKor's Pragmatic Chaos	0.8554	10.09	2009-07-26 18:18:28
Grand Prize - RMSE <= 0.8563				
3	Grand Prize Team	0.8571	9.91	2009-07-24 13:07:49
4	Opera Solutions and Vandelay United	0.8573	9.89	2009-07-25 20:05:52
5	Vandelay Industries !	0.8579	9.83	2009-07-26 02:49:53
6	PragmaticTheory	0.8582	9.80	2009-07-12 15:09:53
7	BellKor in BigChaos	0.8590	9.71	2009-07-26 12:57:25
8	Dace_	0.8603	9.58	2009-07-24 17:18:43
9	Opera Solutions	0.8611	9.49	2009-07-26 18:02:08
10	BellKor	0.8612	9.48	2009-07-26 17:19:11
11	BigChaos	0.8613	9.47	2009-06-23 23:06:52
12	Feeds2	0.8613	9.47	2009-07-24 20:06:46
Progress Prize 2008 - RMSE = 0.8616 - Winning Team: BellKor in BigChaos				
13	xiangliang	0.8633	9.26	2009-07-21 02:04:40
14	Gravity	0.8634	9.25	2009-07-26 15:58:34
15	Ces	0.8642	9.17	2009-07-25 17:42:38
16	Invisible Ideas	0.8644	9.14	2009-07-20 03:26:12
17	Just a guy in a garage	0.8650	9.08	2009-07-22 14:10:42
18	Craig Carmichael	0.8656	9.02	2009-07-25 16:00:54
19	J Dennis Su	0.8658	9.00	2009-03-11 09:41:54
20	acmehill	0.8659	8.99	2009-04-16 06:29:35
Progress Prize 2007 - RMSE = 0.8712 - Winning Team: KorBell				
Cinematch score on quiz subset - RMSE = 0.9514				



The Ensemble was an ensemble solution of teams which had been competing individually for the prize.

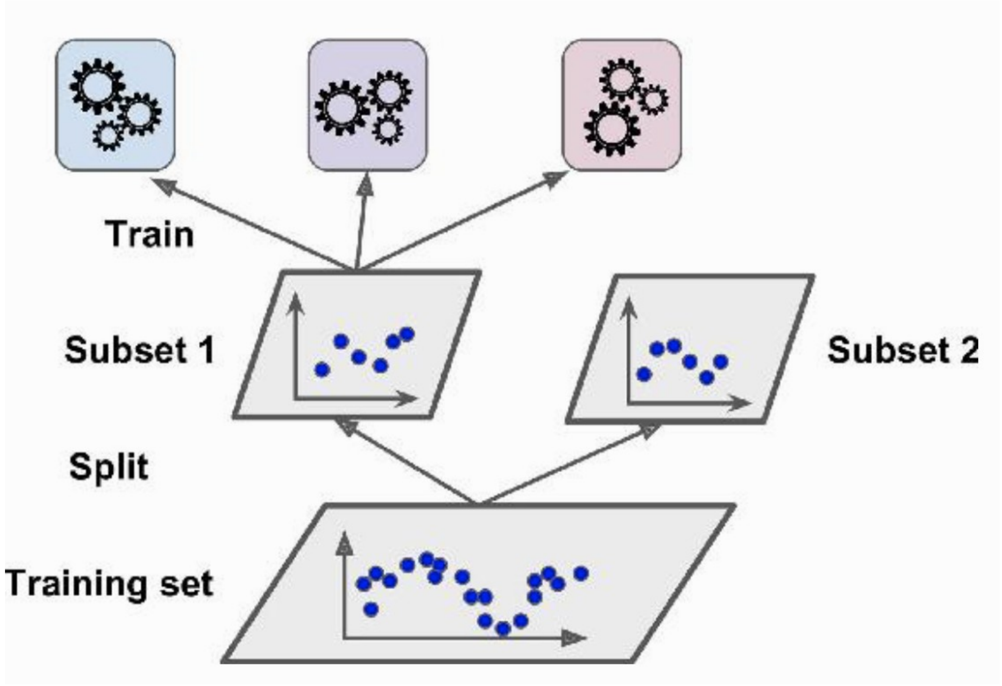
The Idea of Stacking

> Instead of using trivial functions (such as majority voting / averaging) to aggregate the predictions of all predictors in an ensemble, we **train a model (aka blender)** to perform this aggregation.



Training stacked model – Step 1

- The training set is split in two subsets.
- The first subset is used to train the predictors in the first layer.
- The second subset is used to train the blender in the second layer.

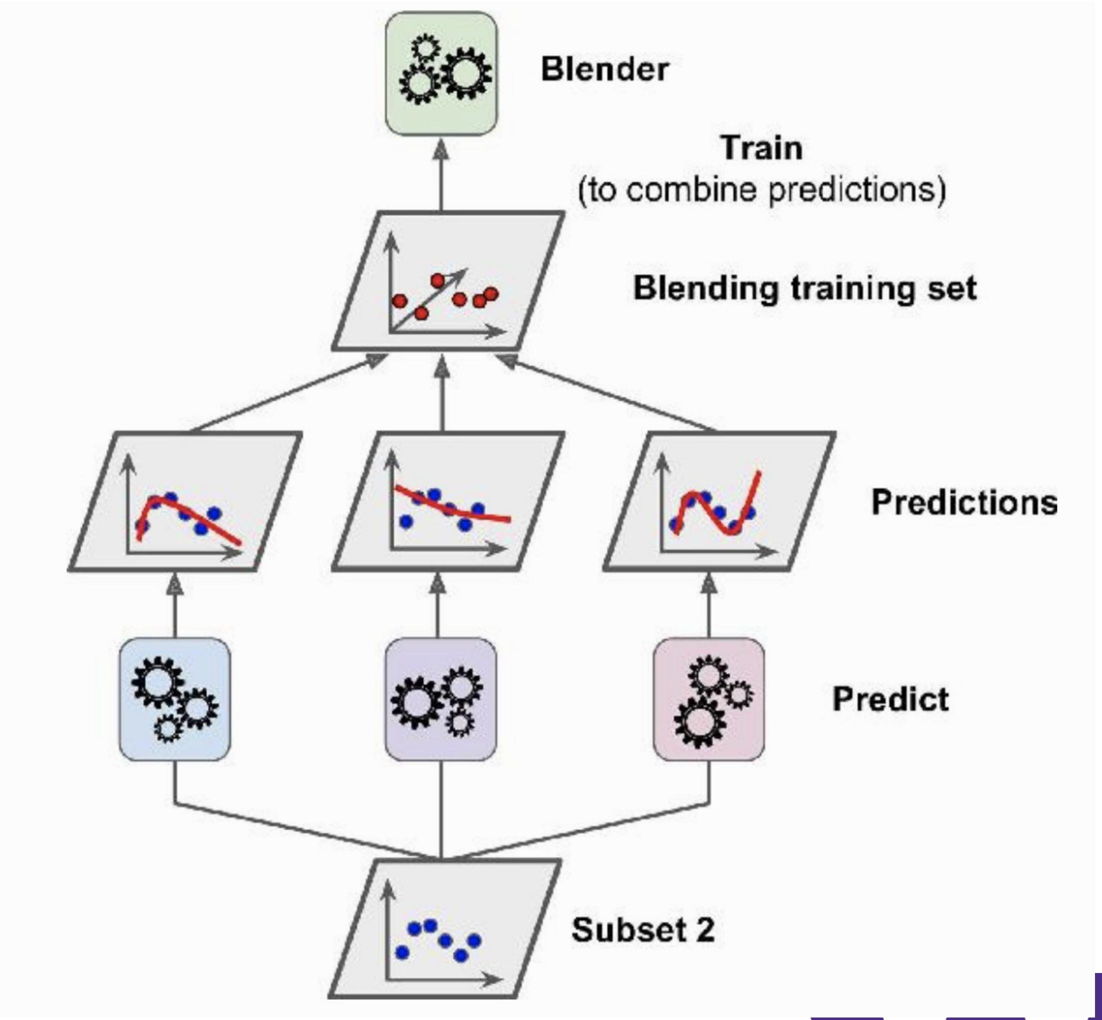


Training stacked model – Step 2

- The first layer predictors are used to make predictions on the second subset.

- Create a new training set using these predicted values as input features.

- The blender is trained on this new training set, so it learns to predict the target value given the first layer's predictions.

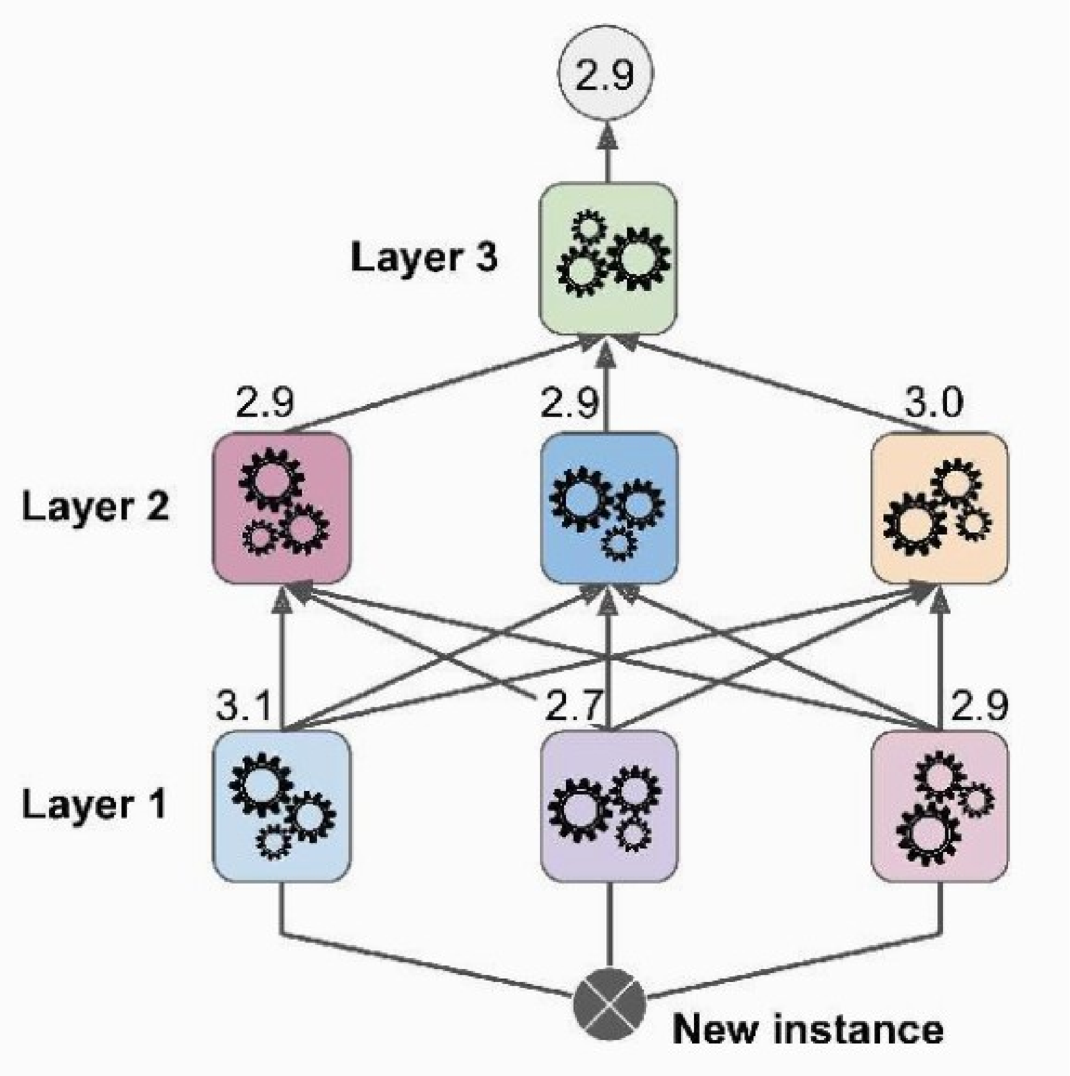


Multilayer Stacking Ensemble

> We can easily extend the 2-layer stacking model to multi-layer stacking.

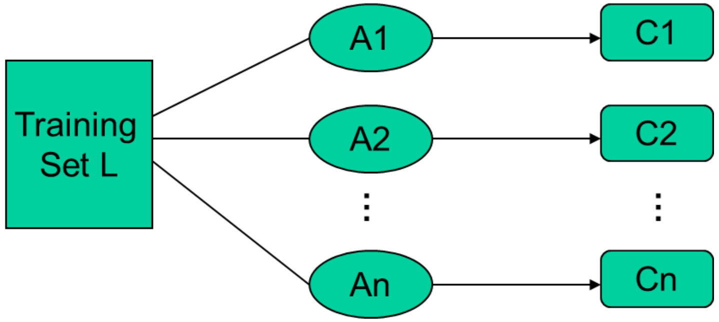
> Split the original training data into K subsets for a K-layer stacking model.

> The i^{th} subset of the data is used to learn the blenders in the i^{th} layer to avoid data leakage.

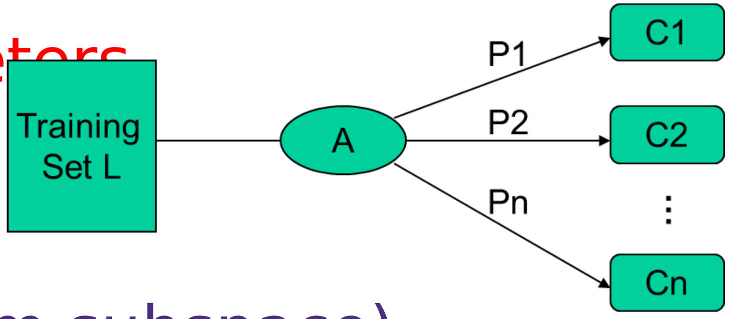


Types of Ensemble Learning

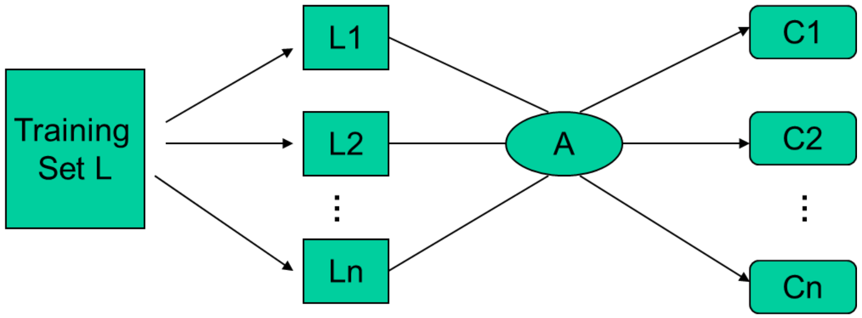
- Different learning algorithms



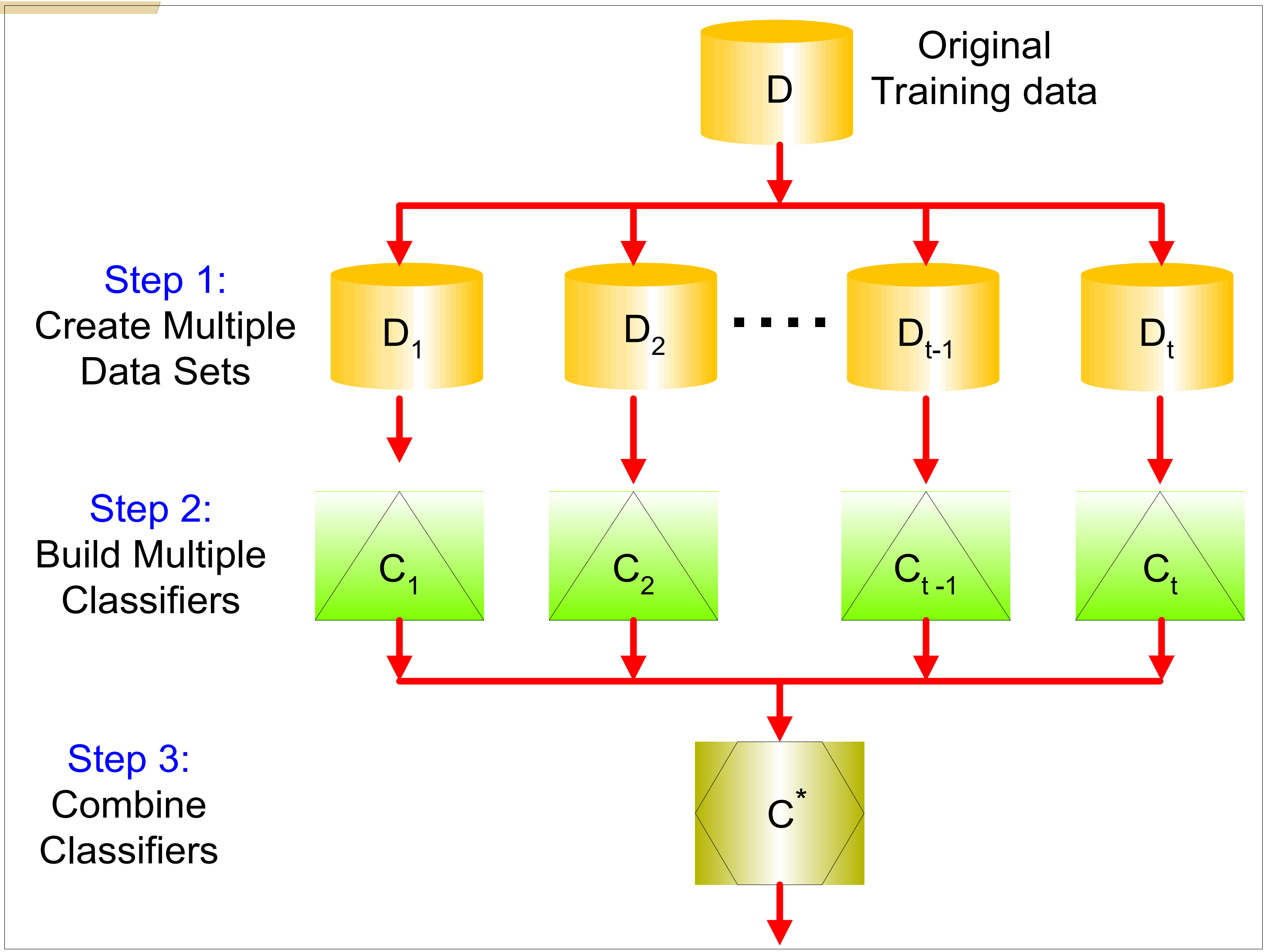
- Algorithms with different choice for parameters



- Data set with different features (e.g. random subspace)
- Data set = different subsets (e.g. bagging, boosting)



General Idea



Fixed Combination Rules

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

How to Combine Classifiers

- Voting:
  - Classifiers are combined in a static way
  - Each base-level classifier gives a (weighted) vote for its prediction
  - Plurality vote: each base-classifier predict a class

Example

- Stacking: a stack of classifiers
  - Classifiers are combined in a dynamically
  - A machine learning method is used to learn how to combine the prediction of the base-level classifiers.
  - Top level classifier is used to obtain the final prediction from the predictions of the base-level classifiers

Example

Notebook Time

What is the Main Challenge for Developing Ensemble Models?

- The main challenge is not to obtain highly accurate base models, but rather to obtain base models which make different kinds of errors.
- For example, if ensembles are used for classification, high accuracies can be accomplished if different base models misclassify different training examples, even if the base classifier accuracy is low. Independence between two base classifiers can be assessed in this case by measuring the degree of overlap in training examples they misclassify ($|A \cap B| / |A \cup B|$)—more overlap means less independence between two models.