CO3091 - Computational Intelligence and Software Engineering

Lecture 24


Image from: http://www.kidscowsandmore.org/wp-content/uploads/2010/10/Beef-MC-Something-Special-129W14-Web.jpg

Ensembles of Learning Machines

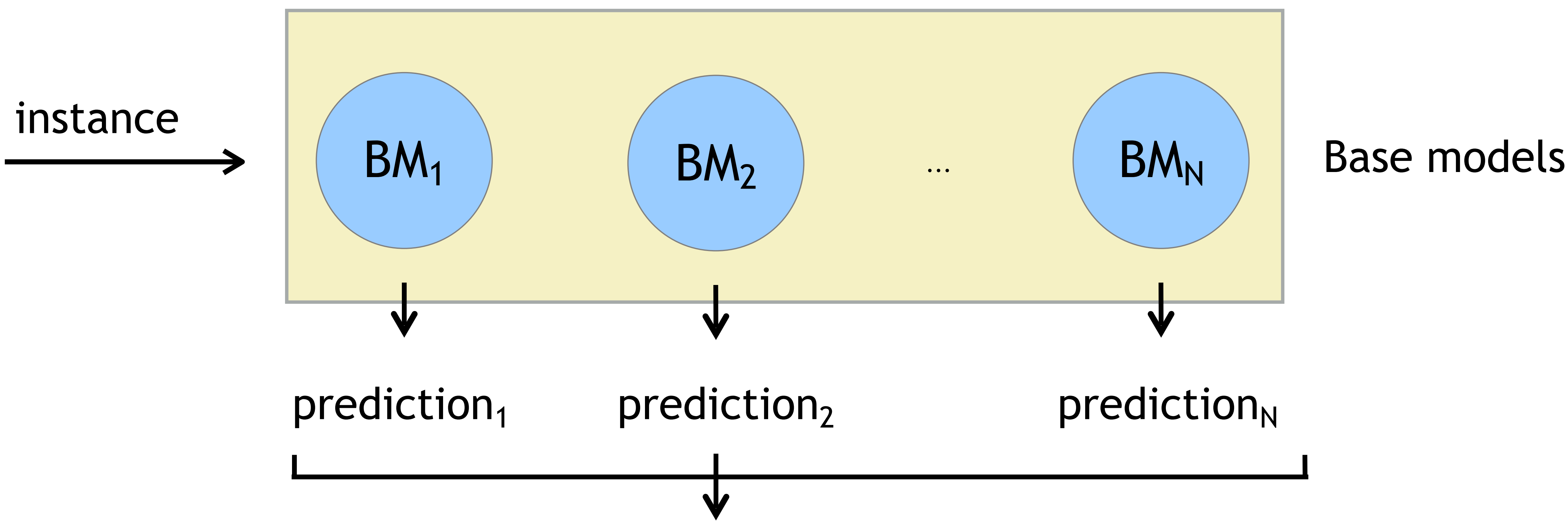
Leandro L. Minku

Overview

- So far in the machine learning part of the module…
  - k-Nearest Neighbours.
  - Naive Bayes.
  - Decision Trees.
- Today:
  - Combining learners into ensembles in order to improve predictive performance.
  - When do ensembles work?
  - Bagging ensembles.
  - Software effort estimation.

What are Ensembles?

Ensembles are sets of learning machines grouped together with the aim of reducing error / increasing accuracy.

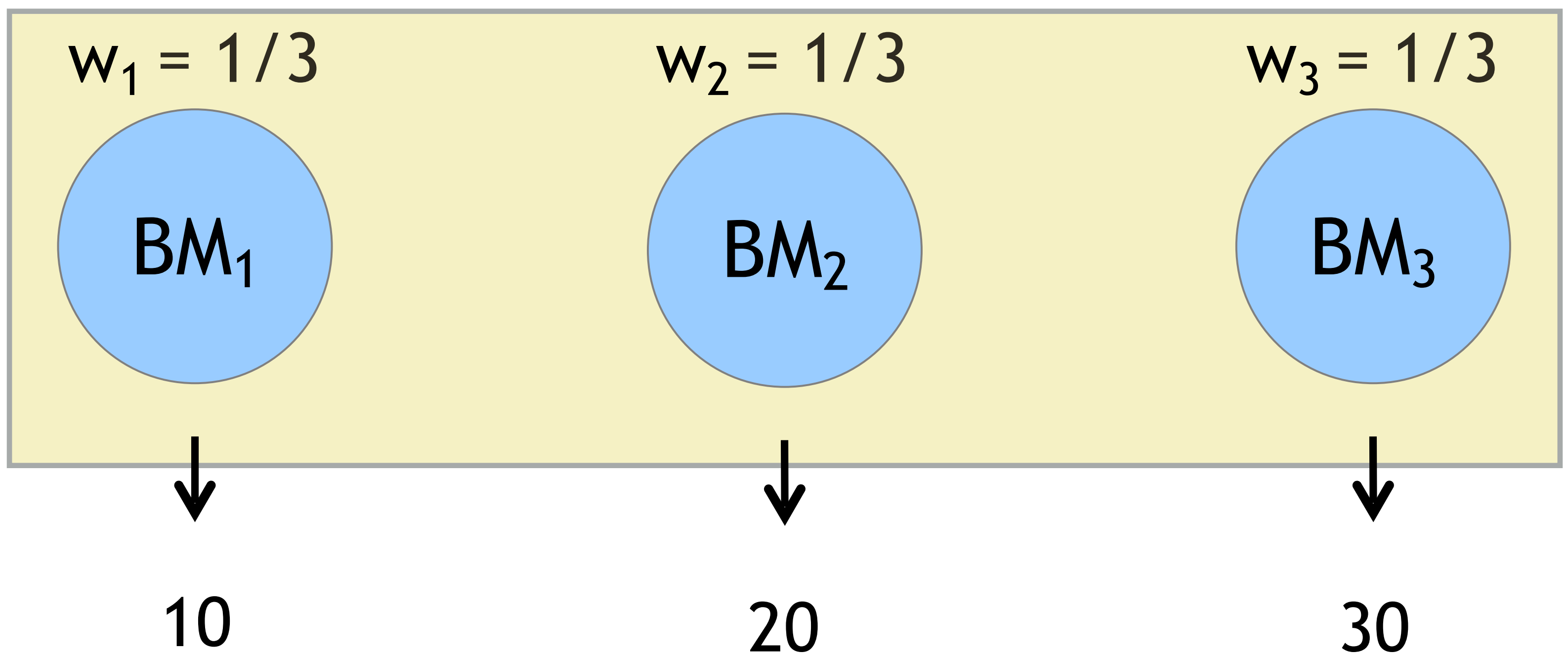


E.g.: ensemble prediction = $\Sigma\, w_i\, prediction_i$

E.g.: ensemble prediction = weighted majority vote among $prediction_i$

$\Sigma\, w_i = 1$

Example of Predictions

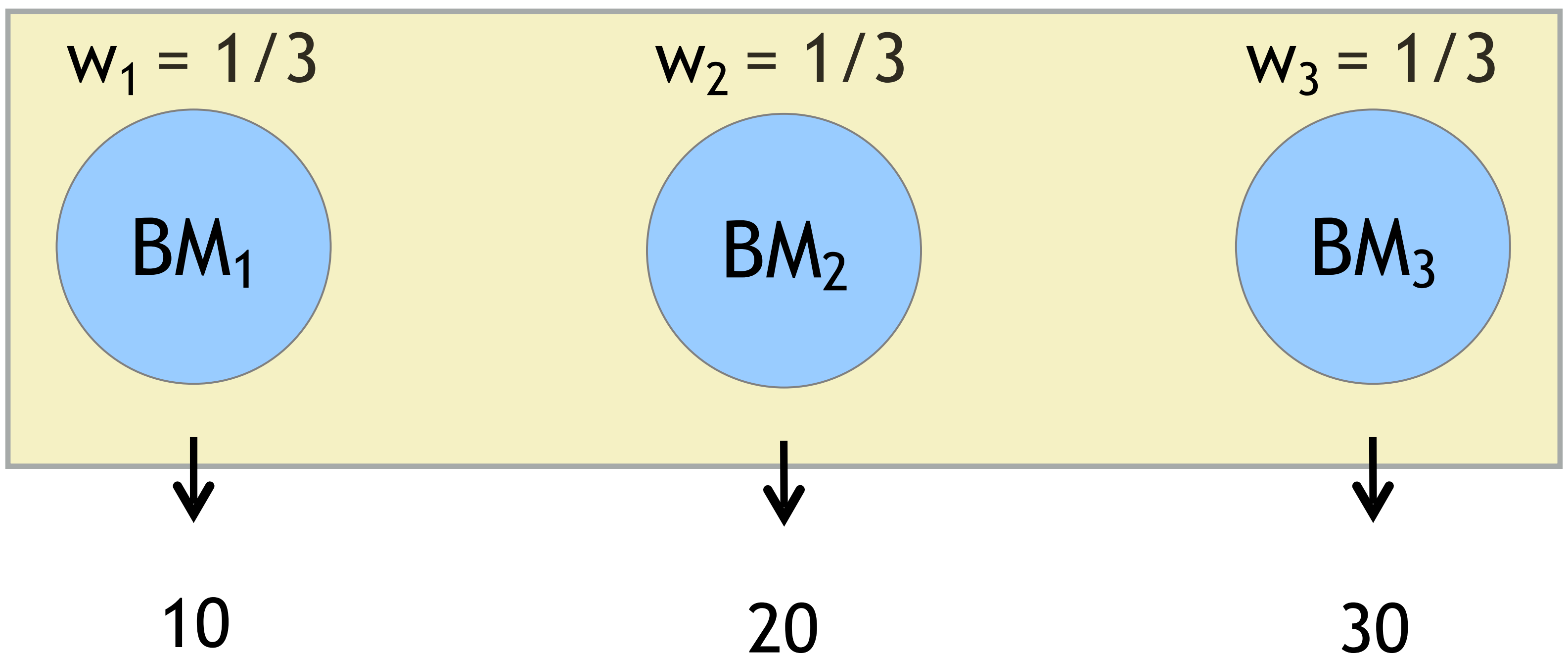


ensemble prediction = $\Sigma\, w_i$ prediction$_i$

ensemble prediction = simple average of prediction$_i$

ensemble prediction = ?

Example of Predictions

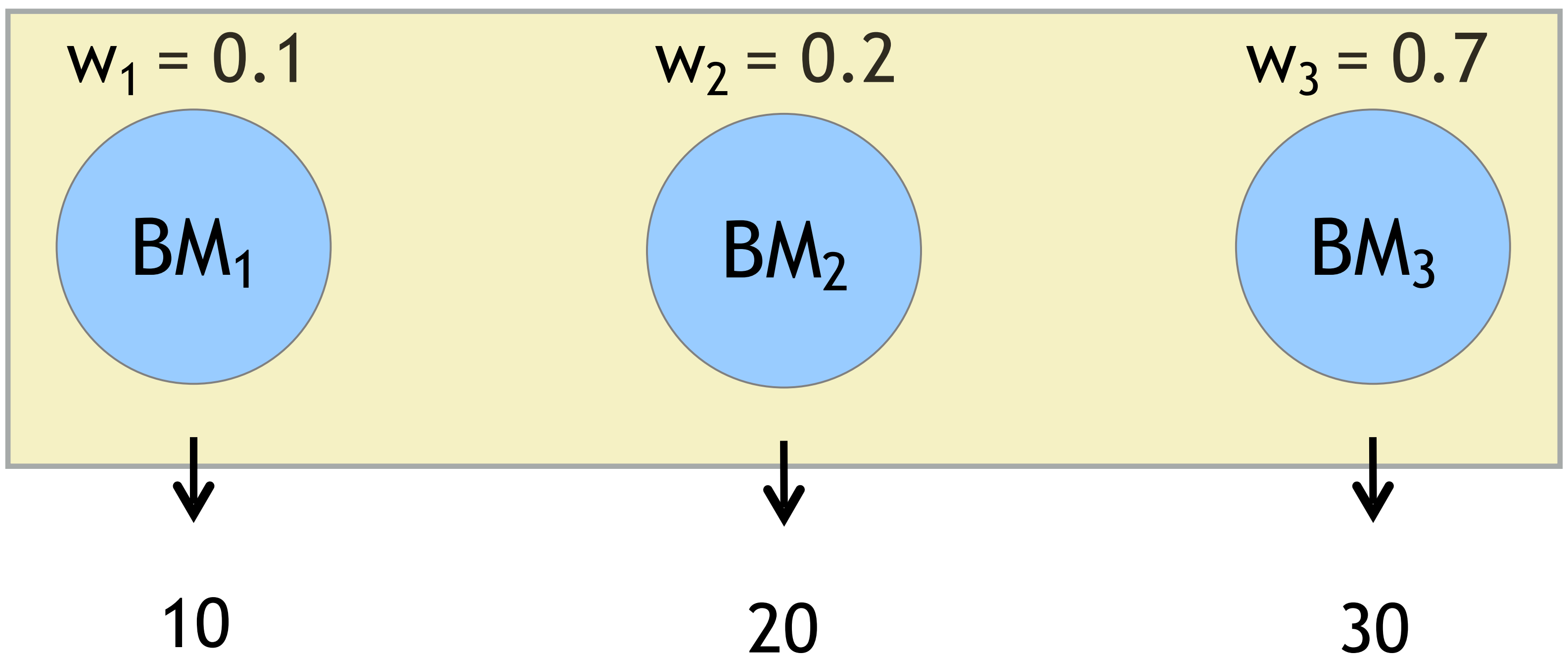


ensemble prediction = Σ w_i $prediction_i$

ensemble prediction = simple average of $prediction_i$

ensemble prediction = 20

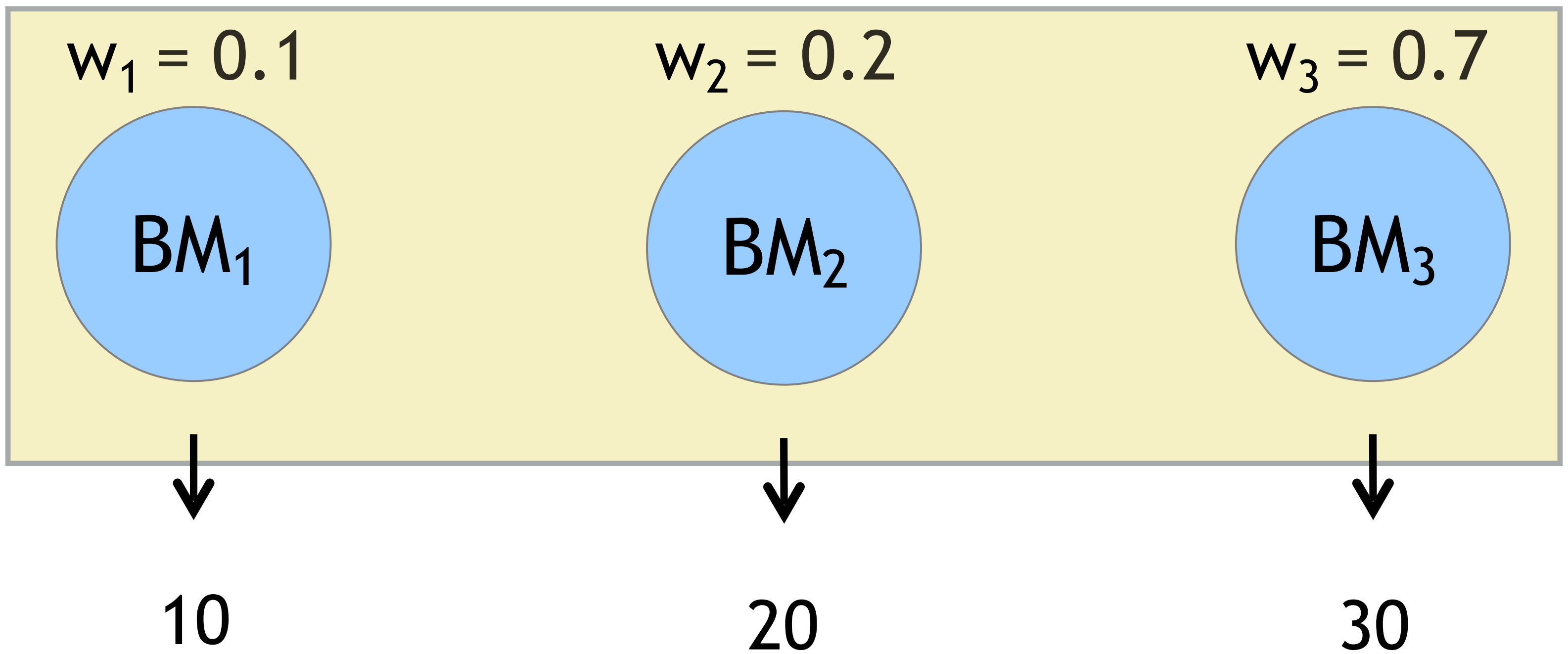
Example of Predictions



ensemble prediction = $\Sigma\, w_i$ prediction$_i$

ensemble prediction = ?

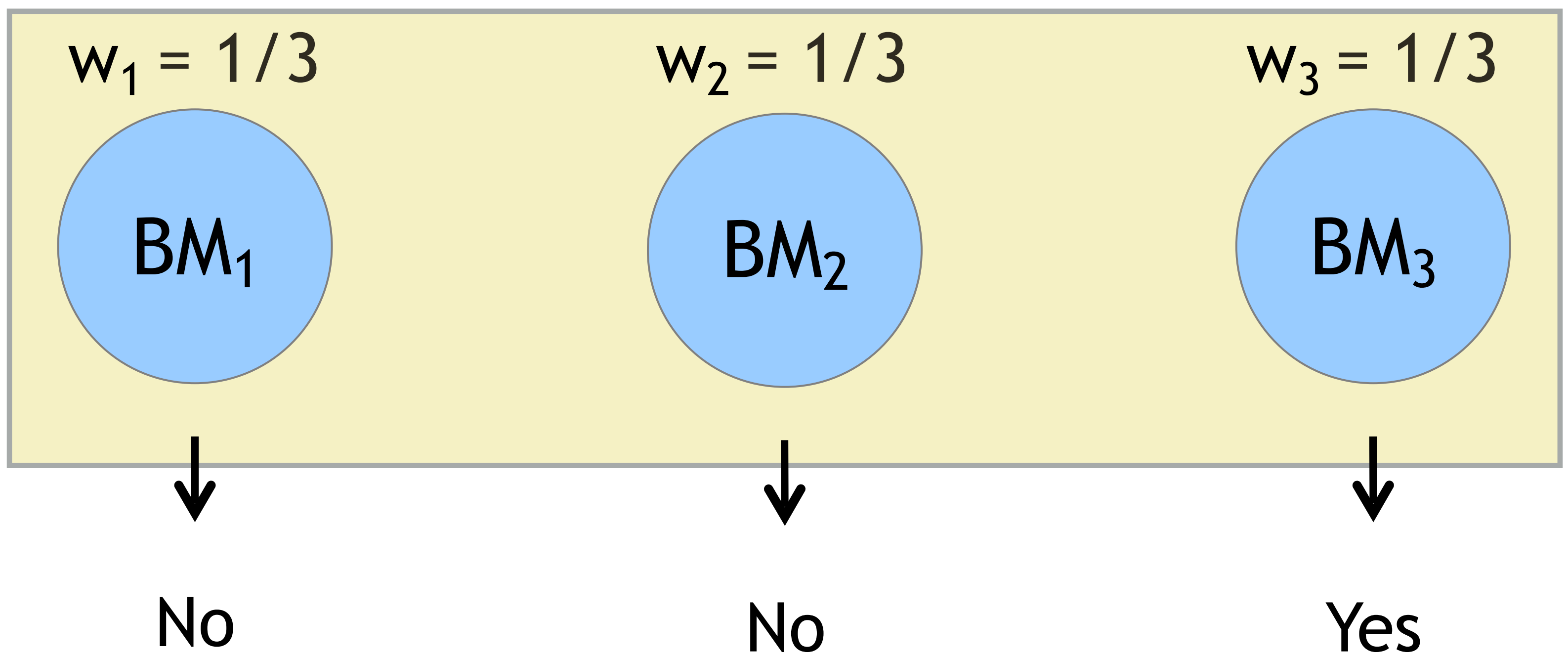
Example of Predictions



ensemble prediction = $\Sigma\, w_i$ prediction$_i$

ensemble prediction = 0.1 * 10 + 0.2 * 20 + 0.7 * 30 = 1 + 4 + 21 = 26

Example of Predictions

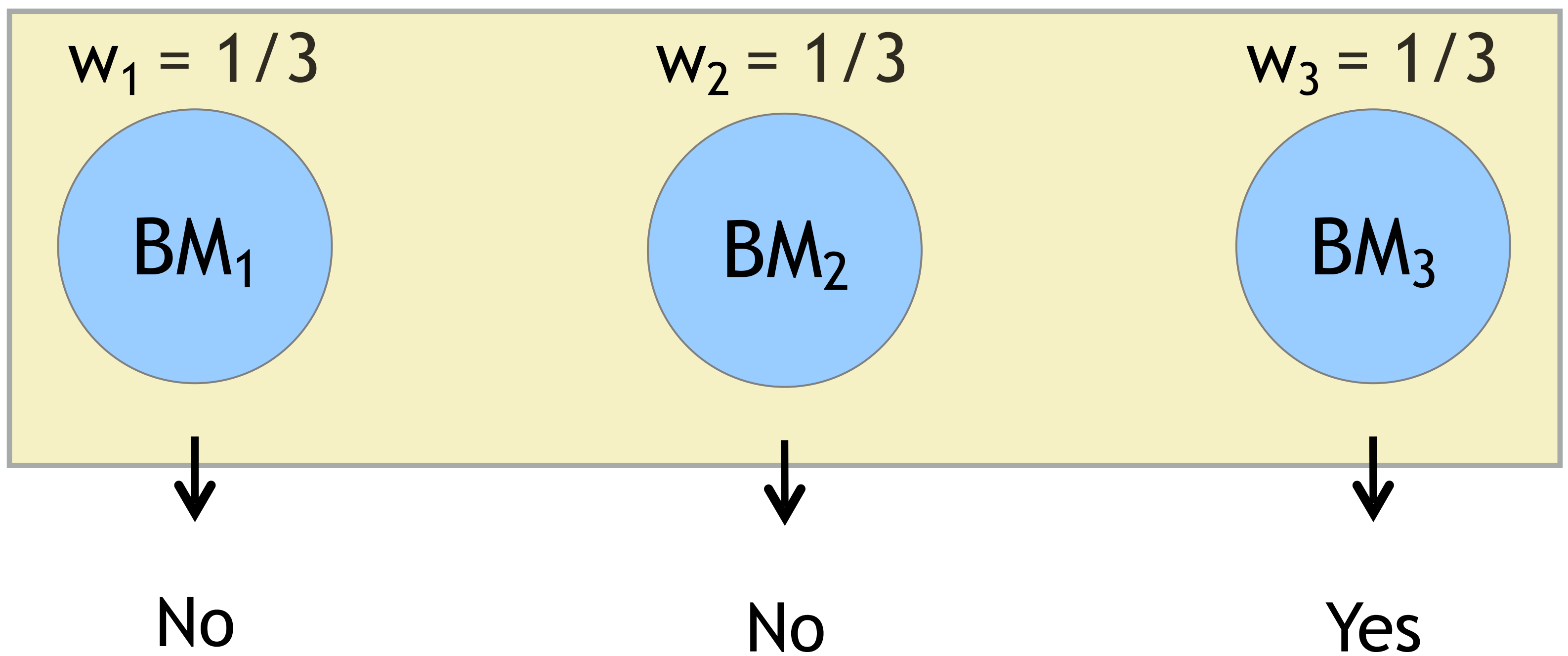


ensemble prediction = weighted majority vote among $prediction_i$

ensemble prediction = majority vote among $prediction_i$

ensemble prediction = ?

Example of Predictions

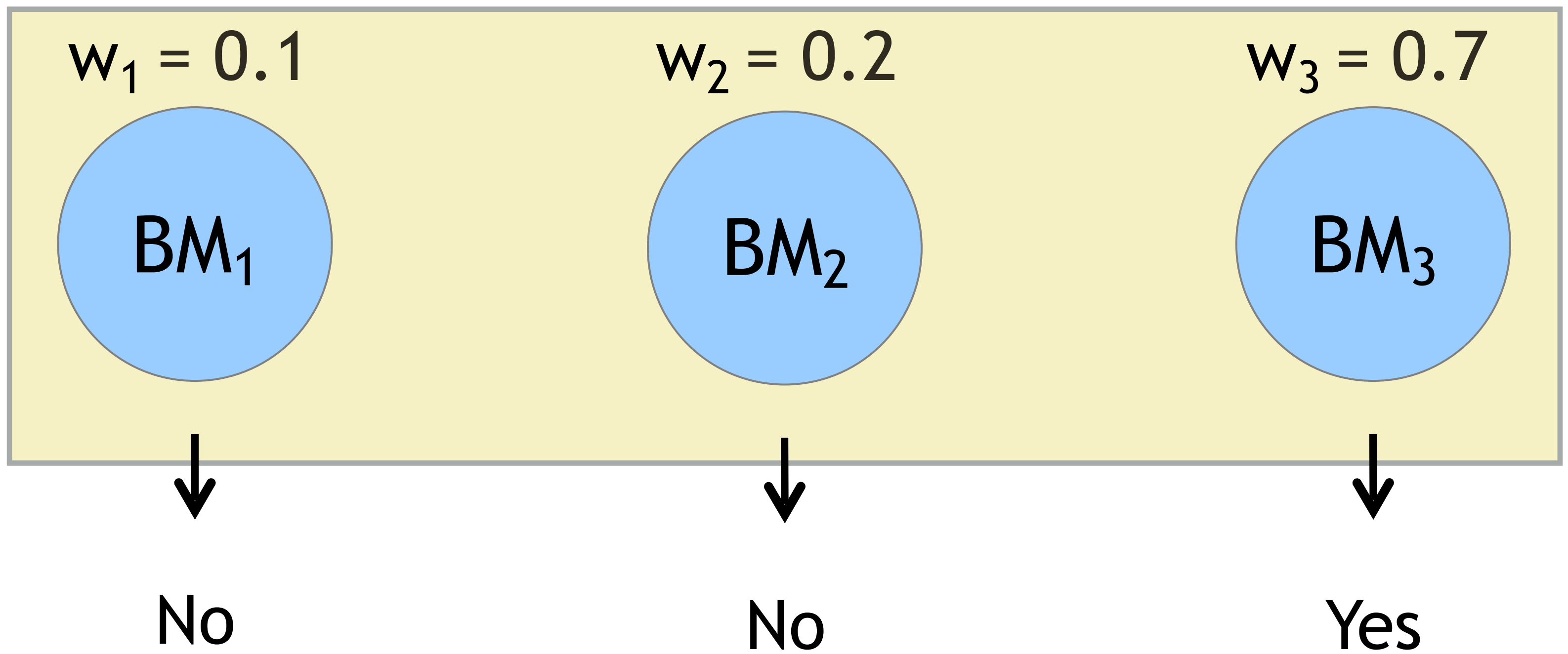


ensemble prediction = weighted majority vote among prediction_i

ensemble prediction = majority vote among prediction_i

ensemble prediction = No

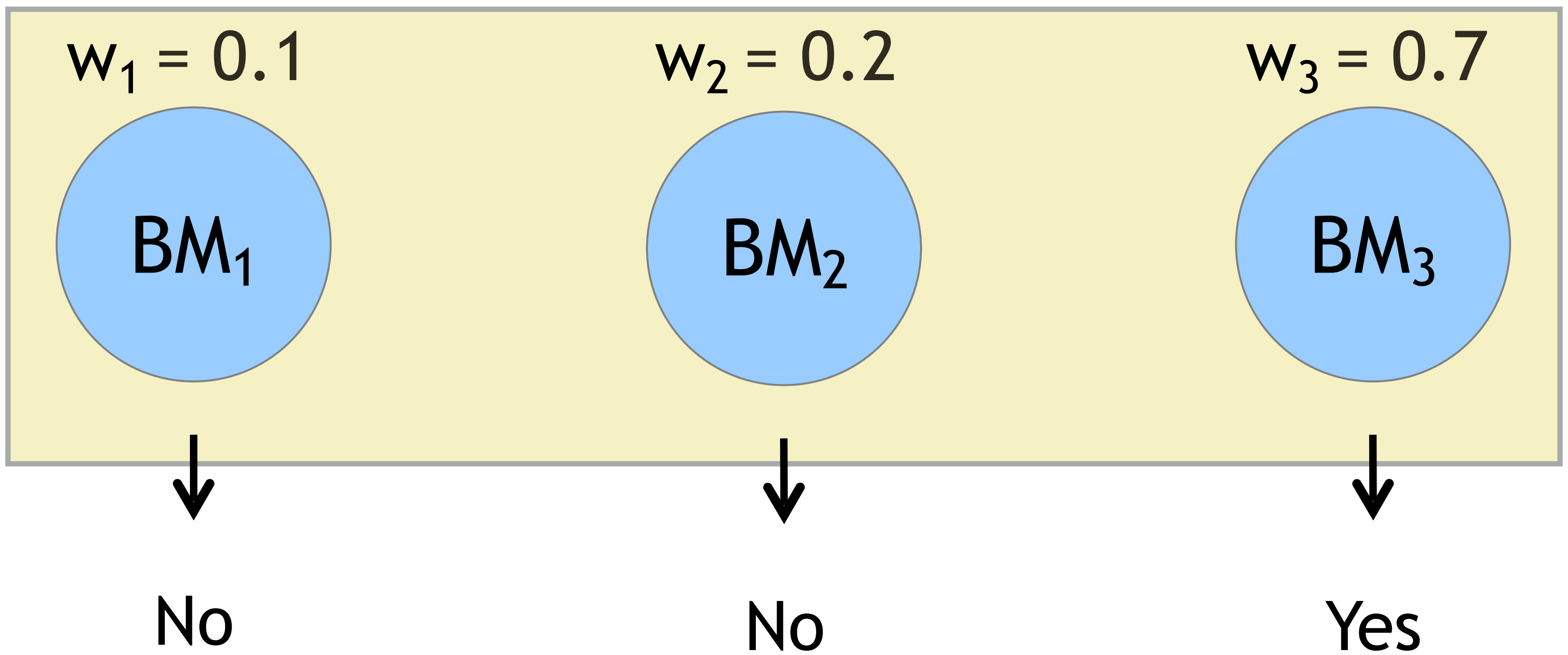
Example of Predictions



ensemble prediction = weighted majority vote among prediction_i

ensemble prediction = ?

Example of Predictions



ensemble prediction = weighted majority vote among prediction_i

Votes for No: 0.1 + 0.2 = 0.3

Votes for Yes: 0.7

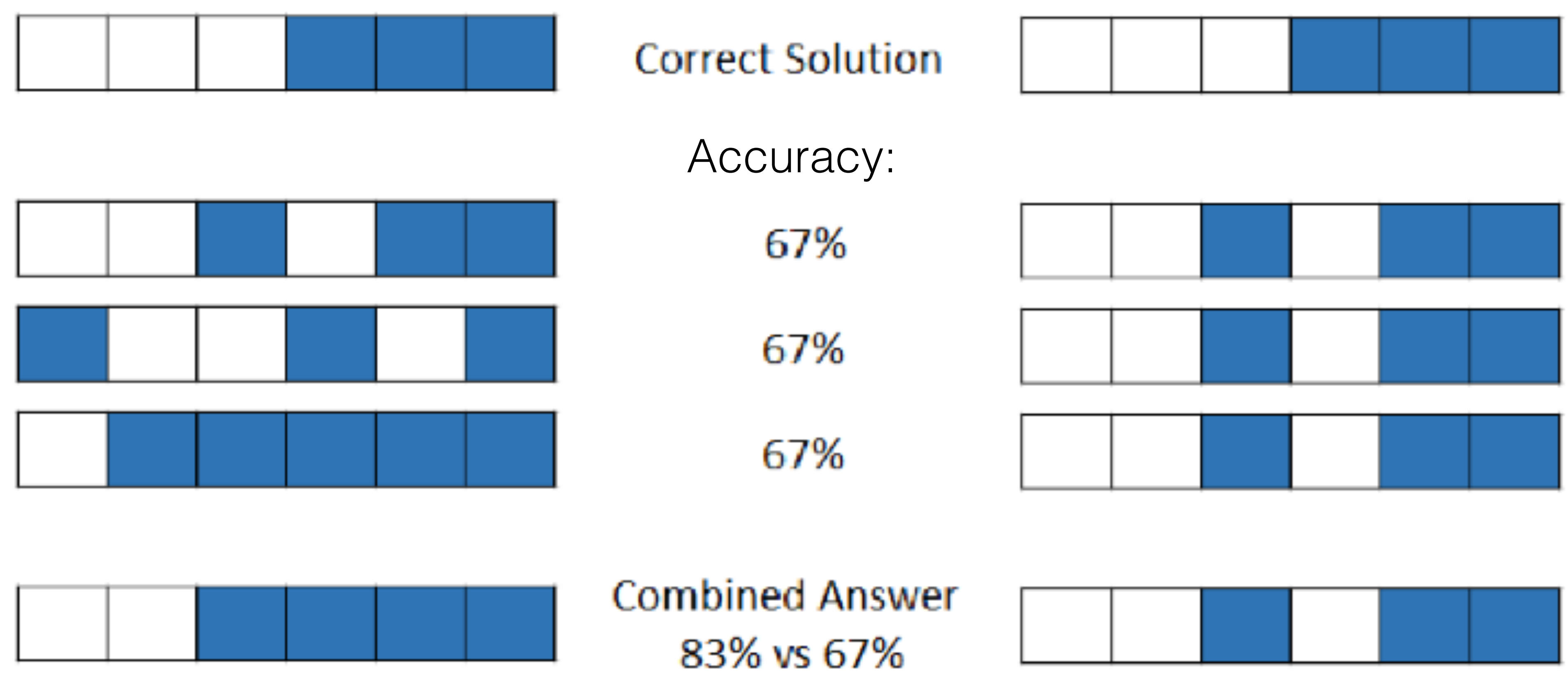
ensemble prediction = Yes

When Do Ensembles Work Well?

- Base models should be both accurate and diverse.
  - Accuracy:
    - If base models make too many mistakes, the whole ensemble prediction will also make too many mistakes.
  - Diversity: models are diverse if they make different mistakes.
    - If all base models make the same mistakes, the whole ensemble prediction will make the same mistakes as the base models.

Intuition for Classification Problems

Intuition: correct predictions given by some models compensate for the incorrect predictions given by the other models.



Intuition for Regression Problems

[YouTube Video posted by timb6: https://youtu.be/iOucwX7Z1HU]

Intuition for Regression Problems

Intuition: overestimations are compensated by underestimations if the base models are diverse enough.

Diversity is necessary both for classification and for regression problems!

Ensemble Approaches

Different ensemble learning algorithms can be seen as different approaches to generate accurate and diverse base models.

Bagging (Bootstrap Aggregating)

- Bootstrap sampling:
  - Sample |D| examples from a training set of size |D| uniformly with replacement.
  - Uniformly = all examples have the same chance to be selected.
  - With replacement = allow an example to be selected more than once to be part of the sample.

Project	Size	Team Expertise	Programming Language	Effort
P1	100	High	Java	80
P2	200	Normal	C++	220
P3	150	High	Python	100
P4	500	Normal	Java	600
P5	550	Normal	C++	700

Training Set With Size |D| = 5

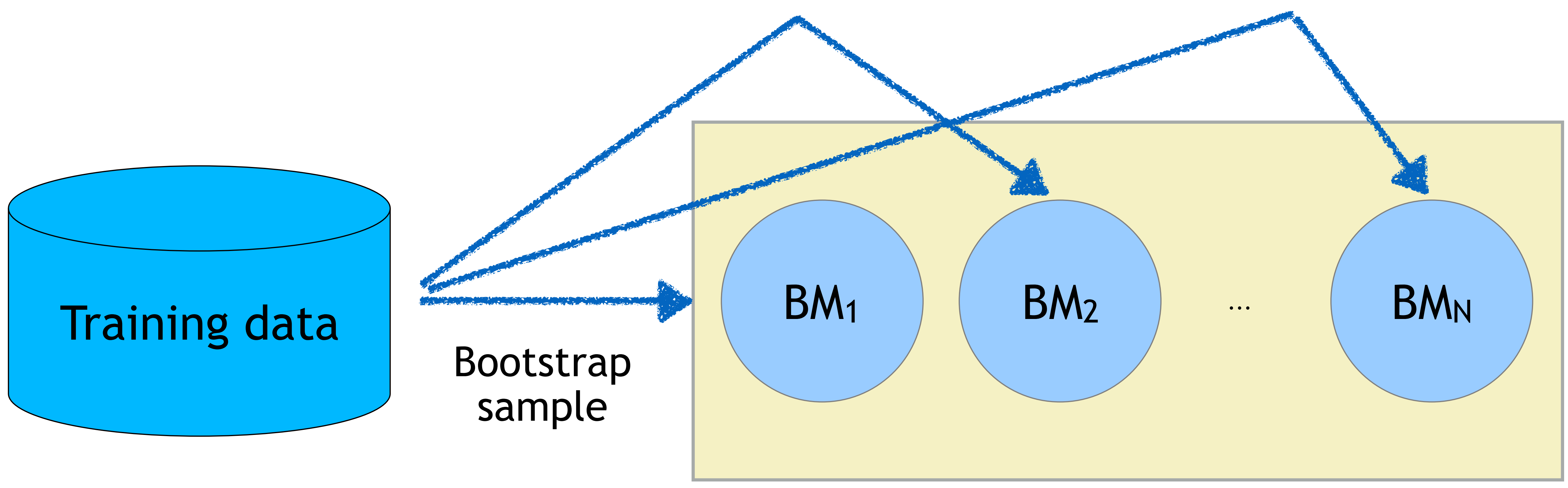
Project	Size	Team Expertise	Programming Language	Effort
P2	200	Normal	C++	220
P5	550	Normal	C++	700
P1	100	High	Java	80
P5	550	Normal	C++	700
P4	500	Normal	Java	600

Sample from Training Set With Size |D| = 5

Bagging (Bootstrap Aggregating)

- Bootstrap sampling:
  - If you use bootstrap sampling to a sample data set D’ with size |D| based on an original data set D of size |D|, it is expected that D’ will contain 63.2% of the unique examples from D.
  - If you repeat bootstrap sampling several times, you are likely to get several different sample data sets.
  - Different samples of the training set will normally lead to different base models.

Bagging (Bootstrap Aggregating)



Pseudocode

Bagging (ensemble size N, training set D, base learning algorithm)

ensemble = {}

For i=1 to N,

D’ <— sample |D| examples from D uniformly with replacement (bootstrap sample)

Build base model BM_i using base learning algorithm with D’

ensemble = ensemble U {BM_i}

Return ensemble

Bagging Ensemble Predictions

- The ensemble predictions are:
  - Regression problems: simple average of the predictions given by the base models.
  - Classification problems: majority vote of the predictions given by the base models.

When and Why Bagging Works?

- Success of bagging depends on base learning algorithm being good but unstable.
  - Good: creates accurate base models.
  - Unstable: small change in the training sample can result in a large change in the predictions given by the resulting base model.
    - This means that enough diversity would be generated.
- If a stable base learning algorithm is used, bagging can even worsen the predictive accuracy.

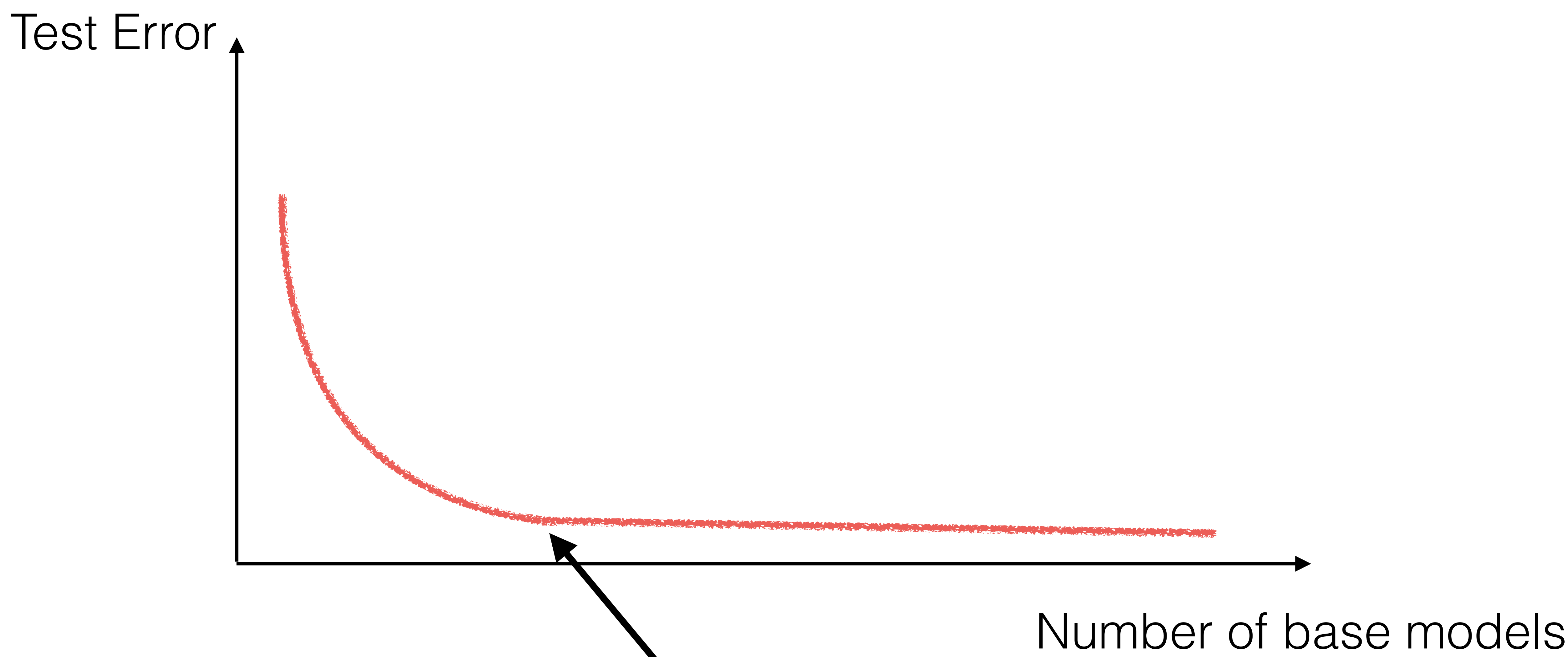
Stable / Unstable Based Learning Algorithms

- Example of unstable base learning algorithm: decision trees.
- Example of stable base learning algorithm: k-NN.

What Ensemble Size Should We Use?

- Early work suggested that ensembles with as few as 10 base models were adequate to sufficiently reduce the test error.
- However, later on it was found that test error can be further reduced even after 10 base models have been added.
- Ensemble size is a parameter of ensemble approaches.
    - The best ensemble size will depend on the data.
    - Too few base models will not be enough to reduce the test error.
    - Too many will waste resources.

Typical Test Error Curve



In Bagging, this point may be ~10–15 base models
for classification problems

Software Effort Estimation

- Estimation of the effort required to develop a software project.
    - Effort is measured in person-hours, person-months, etc.
- Based on features such as programming language, team expertise, estimated size, development type, required reliability, etc.
- Main factor influencing project cost.
- Overestimation vs underestimation.

Machine Learning for Software Effort Estimation

- Software effort estimation is difficult to perform by humans.
  - Affected by irrelevant features.
  - Lack of improvement in the predictions over time.
- Machine learning can help.

x_1 = programming language	x_2 = team expertise	x_3 = estimated size	…	y = required effort
Java	low	1000	…	10 p-month
C++	medium	2000	…	20 p-month
Java	high	2000	…	8 p-month
…	…	…	…	…

Machine Learning for Software Effort Estimation

- k-NN:
  - Can be intuitive for practitioners, helping them to find completed projects that are most similar to the new project to be developed.
  - Offer competitive accuracy in comparison to other approaches, but performs poorly for some companies.
- Decision trees:
  - Can be visualised.
  - Frequently among the best machine learning approaches for software effort estimation.
- Bagging ensembles of decision trees.
  - Can improve accuracy further with respect to decision trees.
  - Difficult to visualise.
- Naive Bayes:
  - Usually doesn't work well for regression problems.

Further Reading

Menzies et al.

Sharing Data and Models in Software Engineering

Elsevier, 2014

Chapter 20 (Ensembles of Learning Machines) until section 20.3

http://readinglists.le.ac.uk/lists/D888DC7C-0042-C4A3-5673-2DF8E4DFE225.html

David Opitz

Ensemble Size

https://www.cs.cmu.edu/afs/cs/project/jair/pub/volume11/opitz99a-html/node10.html