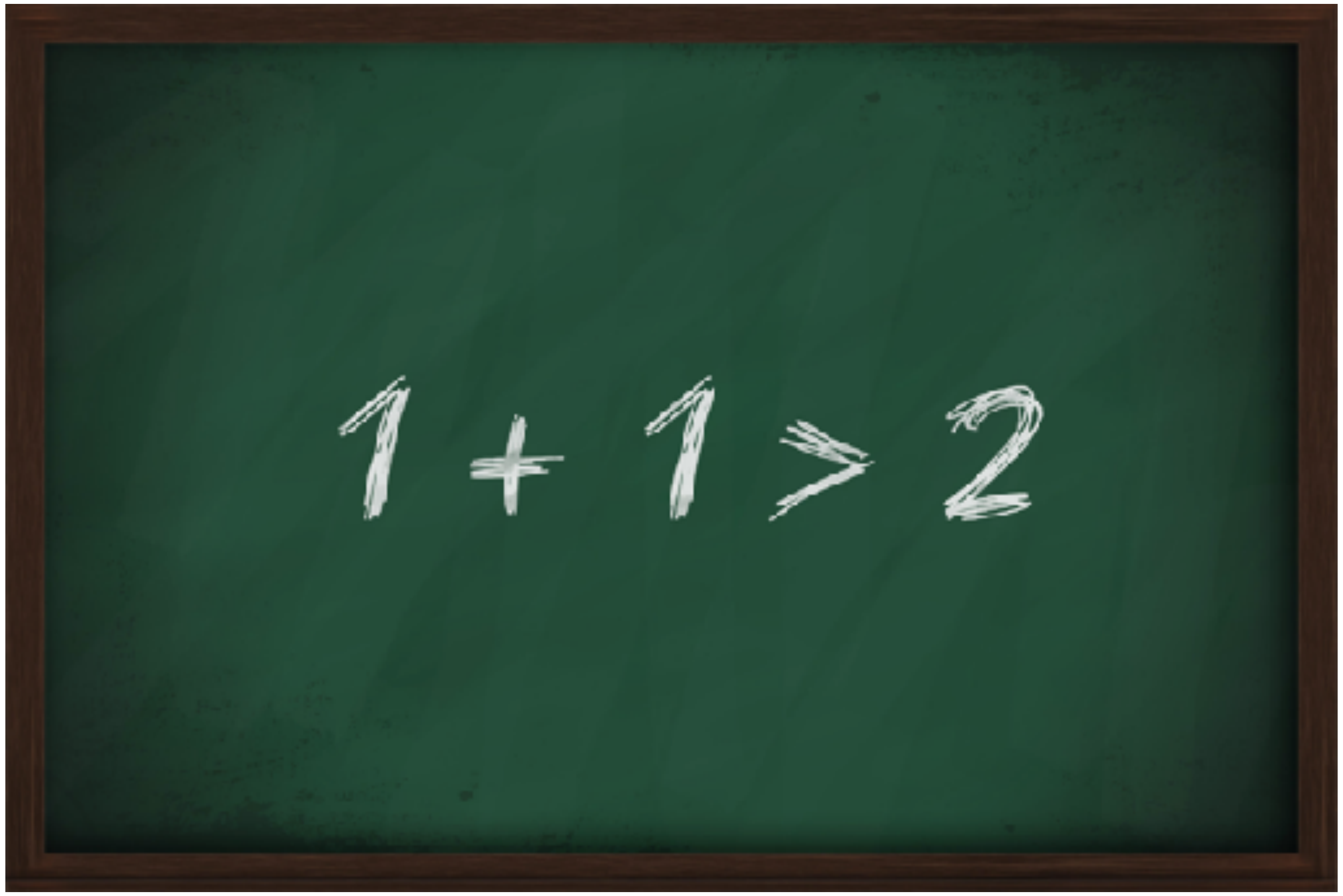
The Whole is Greater than The Sum of the Parts:



On the Value of Machine Learning Ensemble Methods

Leandro L. Minku
School of Computer Science, University of Birmingham, UK

Health / Medical Data

Financial Data

Fraud Data

Video Surveillance Data

Software Usage Data

Speech Data

Machine Learning

Study and develop computational models capable of improving their performance with experience and acquiring knowledge from data on their own.

- Supervised Learning:
  - Classification, e.g.:
    - Detect breast cancer based on a mammogram.
    - Predict whether a person will or will not default their payments based on characteristics such as age, salary, time holding a bank account, etc.
  - Regression (numerical estimations)
    - Estimate the future value of the stock market based on previous values.
    - Estimate how much effort will be required to develop a project, based on the characteristics of the tasks to be performed in the project.

Supervised Learning

[Pre-processed]
Training Data / Examples

x_1 (age)	x_2 (salary)	x_3 (bank)	…	y (default?)
18	1000	Halifax	…	No
30	900	NatWest	…	Yes
20	5000	Halifax	…	No
…	…	…	…	…

Supervised Learning Algorithm

New instance
for which we want
to predict the output

Predictive Model

Prediction

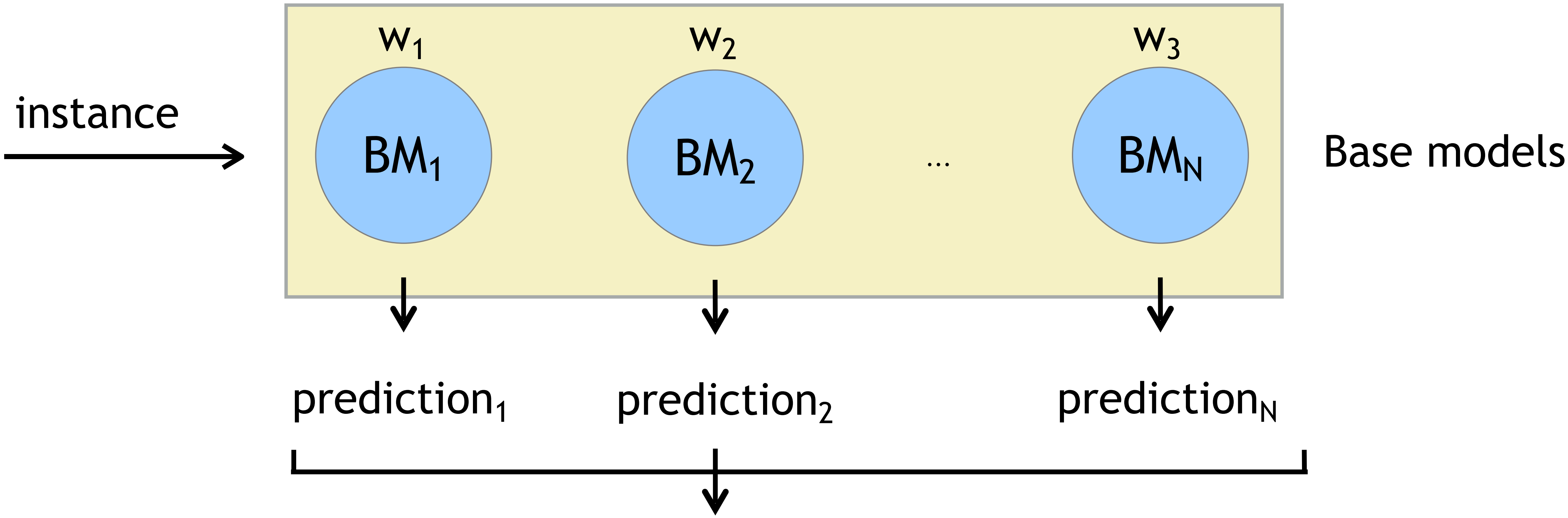
Should we use a single model?

What about using multiple models?

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

Ensembles of Models

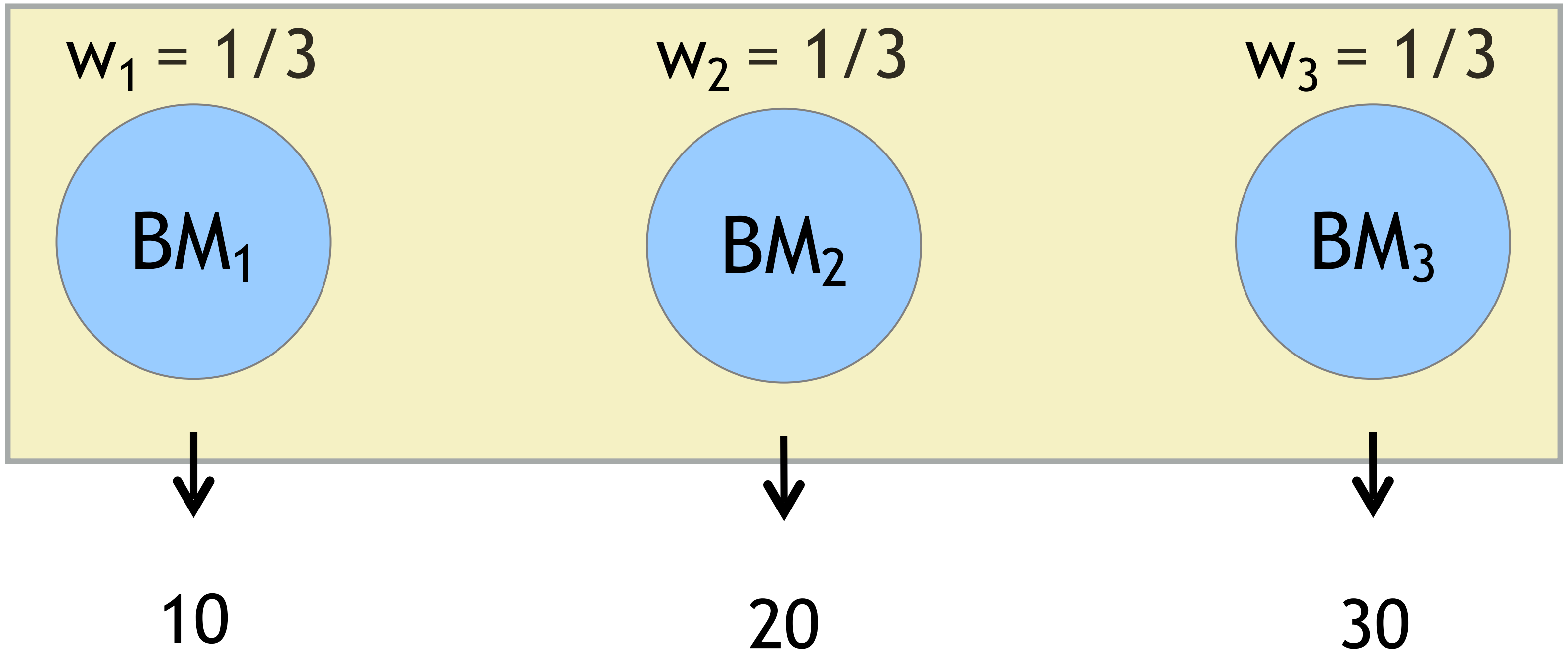
Ensembles are sets of learning machines grouped together with the aim of reducing error / increasing accuracy.



E.g.: ensemble prediction = $\Sigma\, w_i\, prediction_i$, $\Sigma\, w_i = 1$

E.g.: ensemble prediction = weighted majority vote among predictions

Example of Predictions

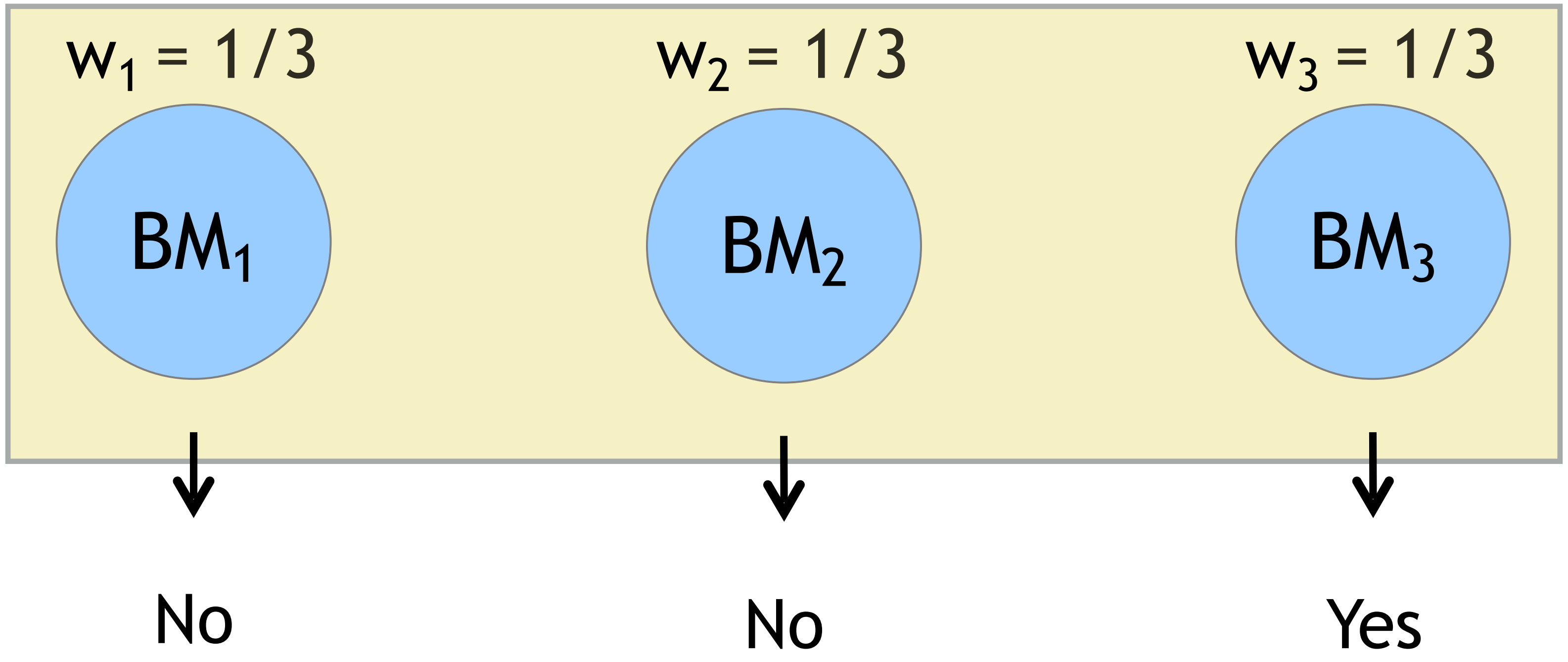


ensemble prediction = $\Sigma\ w_i$ prediction_i

ensemble prediction = simple average of predictions

ensemble prediction = ?

Example of Predictions

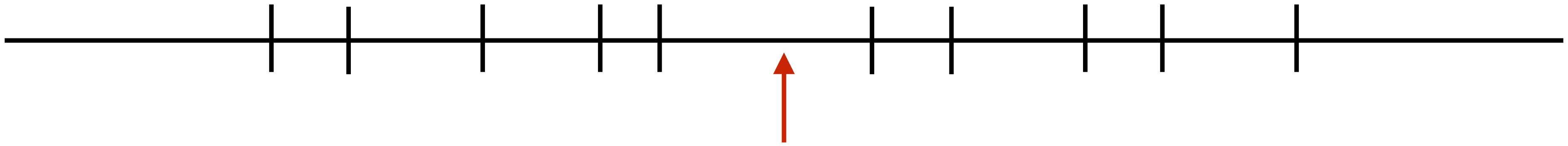


ensemble prediction = weighted majority vote among predictions

ensemble prediction = majority vote among predictions

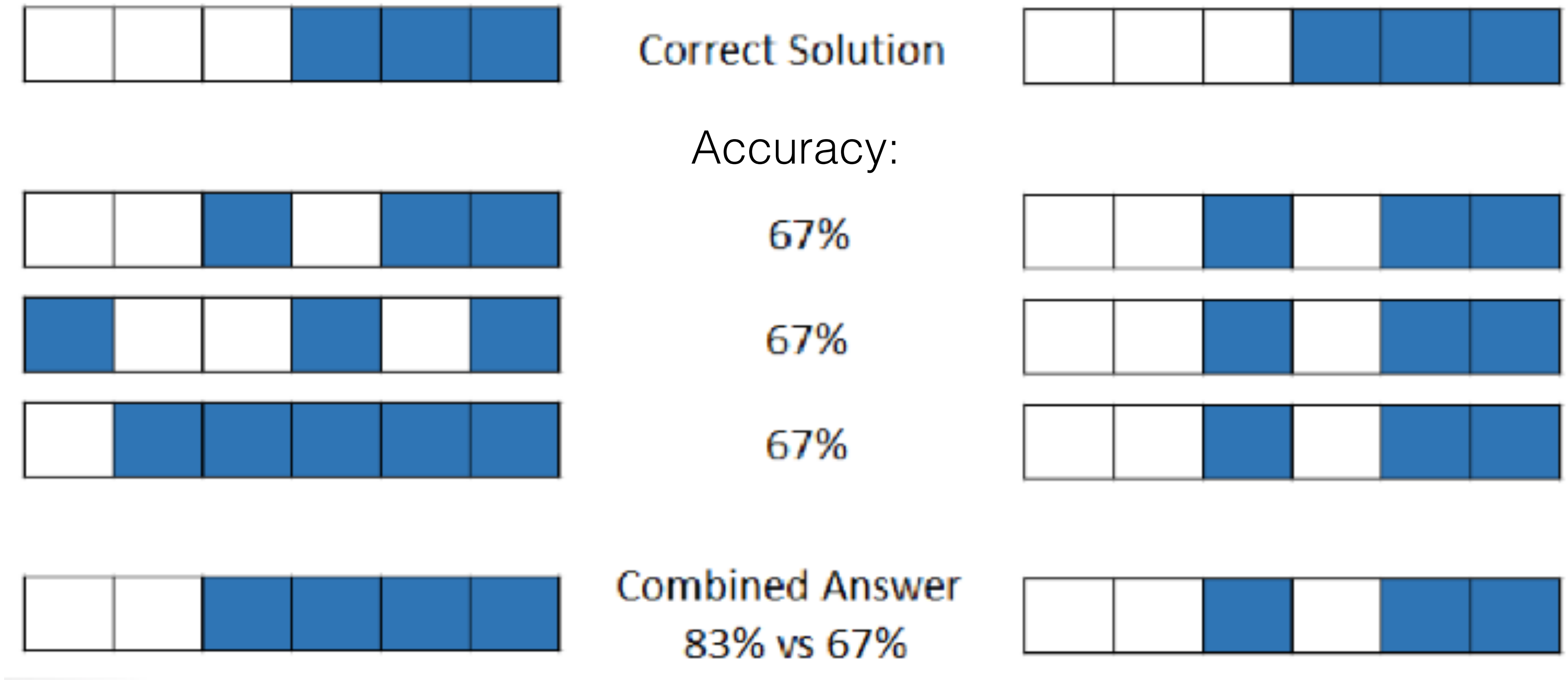
ensemble prediction = ?

Intuition for Regression Problems



Intuition for Classification Problems

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



Correct Solution

Accuracy:

67%

67%

67%

Combined Answer

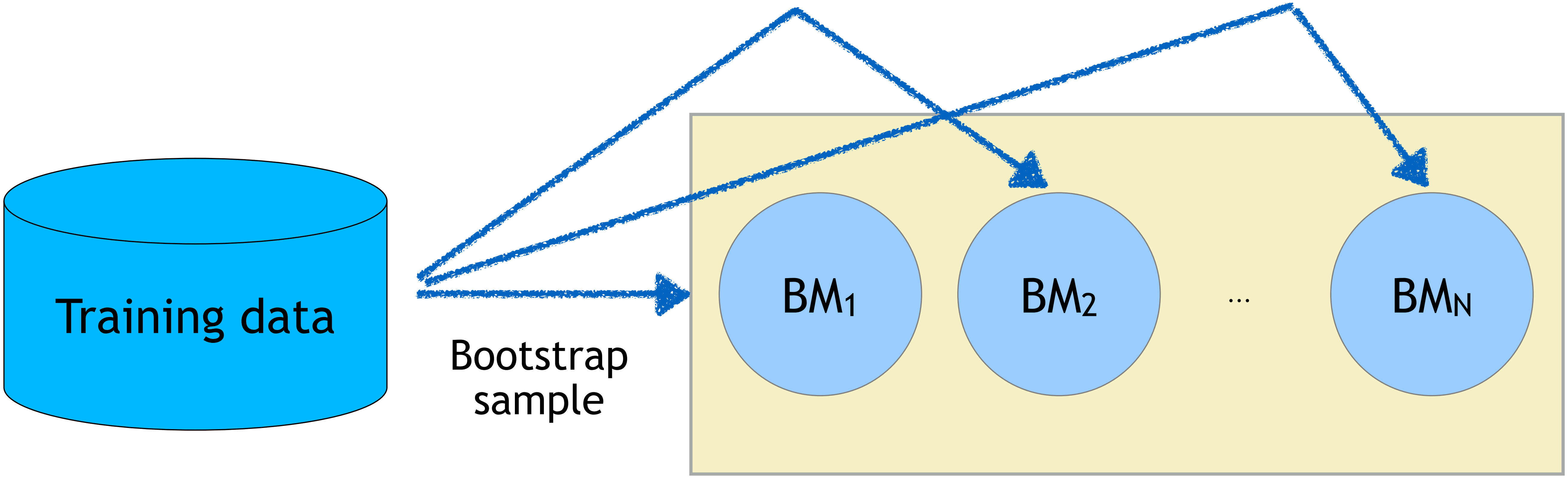
83% vs 67%

When Do Ensembles Work Well?

- Individual 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.

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

Bagging (Bootstrap Aggregating)



Investigation of Ensembles for Software Effort Estimation

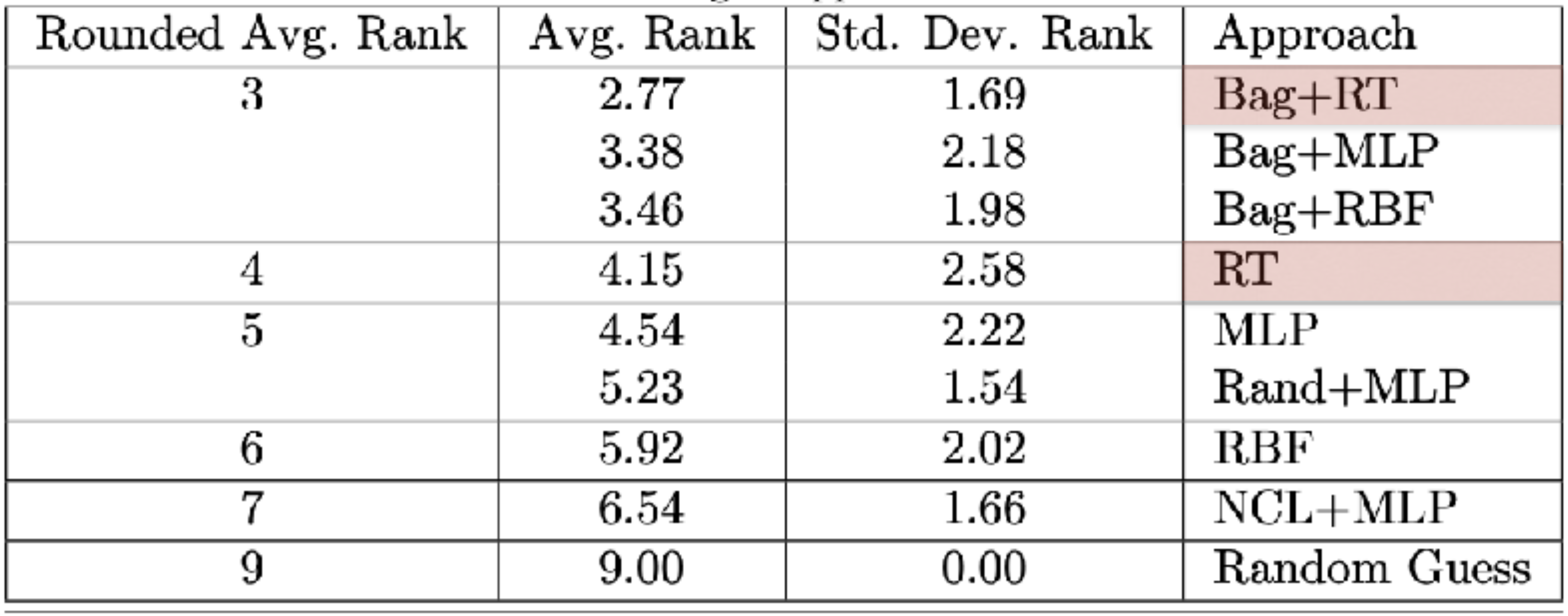
- Single learning machines:
  - REPTree Regression Trees (RTs);
  - Radial Basis Function networks (RBFs);
  - MultiLayer Perceptrons (MLPs).

- Ensembles of learning machines:
  - Bagging with MLPs (Bag+MLPs), with RBFs (Bag+RBFs) and with RTs (Bag+RTs);
  - Random with MLPs (Rand+MLPs);
  - Negative Correlation Learning with MLPs (NCL+MLPs).

MINKU, L. L.; YAO, X. . "Ensembles and Locality: Insight on Improving Software Effort Estimation.", *Information and Software Technology*, Special Issue on Best Papers from PROMISE 2011, Elsevier, v. 55, n. 8, p. 1512-1528, August 2013

Results Based on 13 Software Effort Estimation Data Sets

Table 9: Friedman ranking of approaches in terms of MAE.



Rounded Avg. Rank	Avg. Rank	Std. Dev. Rank	Approach
3	2.77	1.69	Bag+RT
	3.38	2.18	Bag+MLP
	3.46	1.98	Bag+RBF
4	4.15	2.58	RT
5	4.54	2.22	MLP
	5.23	1.54	Rand+MLP
6	5.92	2.02	RBF
7	6.54	1.66	NCL+MLP
9	9.00	0.00	Random Guess

P-value = 0.0134

P-value = 0.4143

P-value = 0.4143

- Ensembles are not always better than single models that are well suited to a certain application.
- Ensembles frequently provide a competitive advantage in terms of predictive performance.

Data Stream Learning

[Pre-processed]
Training Data / Examples

x_1 (age)	x_2 (salary)	x_3 (gender)	…	y (good/bad payer)
18	1000	female	…	Good
30	900	male	…	Bad
20	5000	female	…	Good
…	…	…	…	…

→

Supervised Learning Algorithm

→

New instance
for which we want
to predict the output

↓

Predictive Model

↓

Prediction

Challenges

- Data streams are potentially infinite.
- Data frequently arrive with a high incoming rate.
- Data arrive in a certain order, and come from non-stationary environments.
- There may be class evolution and class imbalance.

Challenges

- Data streams are potentially infinite.
- Data frequently arrive with a high incoming rate.
- Data arrive in a certain order, and come from non-stationary environments.
- There may be class evolution and class imbalance.

Offline Supervised Learning

[Pre-processed]
Training Data / Examples

x_1 (age)	x_2 (salary)	x_3 (gender)	…	y (good/bad payer)
18	1000	female	…	Good
30	900	male	…	Bad
20	5000	female	…	Good
…	…	…	…	…

Supervised Learning Algorithm

New instance
for which we want
to predict the output

Predictive Model

Prediction

Continuous Supervised Learning

[Pre-processed]
Training Data / Examples

x_1 (age)	x_2 (salary)	x_3 (gender)	…	y (good/bad payer)
18	1000	female	…	Good
30	900	male	…	Bad
20	5000	female	…	Good
…	…	…	…	…

Supervised Learning Algorithm

New instance
for which we want
to predict the output

Predictive Model

Prediction

moa

Machine Learning for Data Streams

scikit-multiflow

Challenges

- Data streams are potentially infinite.
- Data frequently arrive with a high incoming rate.
- Data arrive in a certain order, and come from non-stationary environments.
- There may be class evolution and class imbalance.

Non-Stationary Environments

- Concept drift: changes in the joint probability distribution of the problem:
    - $p_t(\mathbf{x},y) = \boxed{p_t(\mathbf{x}|y)} P_t(y)$

E.g., change affecting the relationship between inputs and outputs.

Example of Potential Change in P(**x**|y)

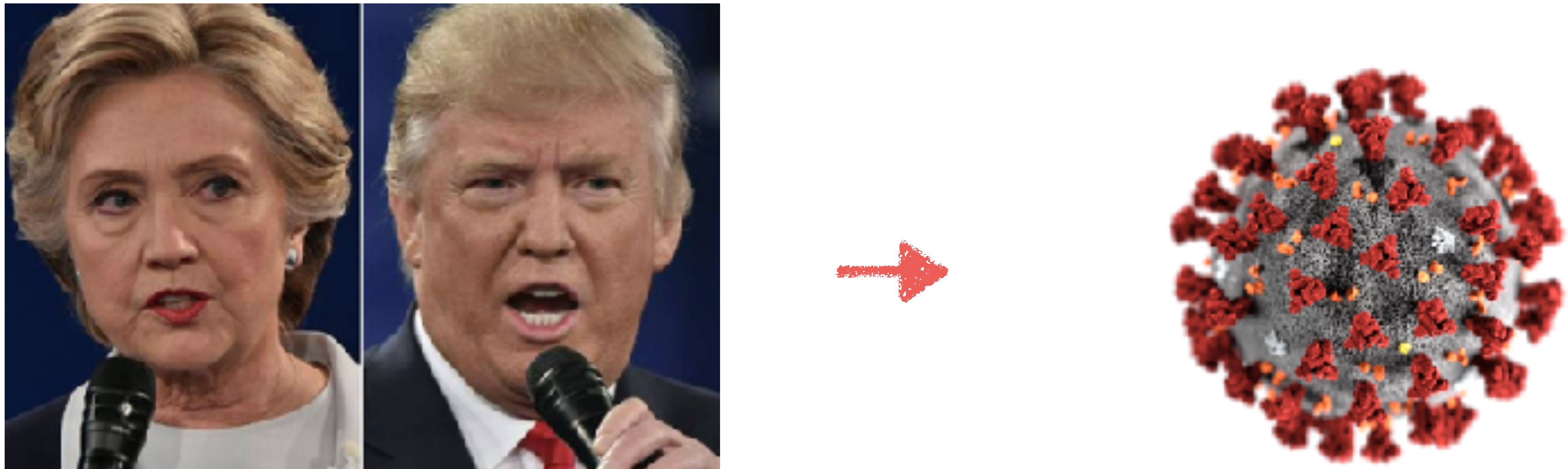
E.g.: credit card approval being affected by economic crises.

Non-Stationary Environments

- Concept drift: changes in the joint probability distribution of the problem:
  - $p_t(\mathbf{x},y) = p_t(\mathbf{x}|y)\,\boxed{P_t(y)} \longrightarrow$ Change affecting the proportion of the classes.

Example of Potential Change in P(y)

E.g.: twitter topic becoming more or less popular.



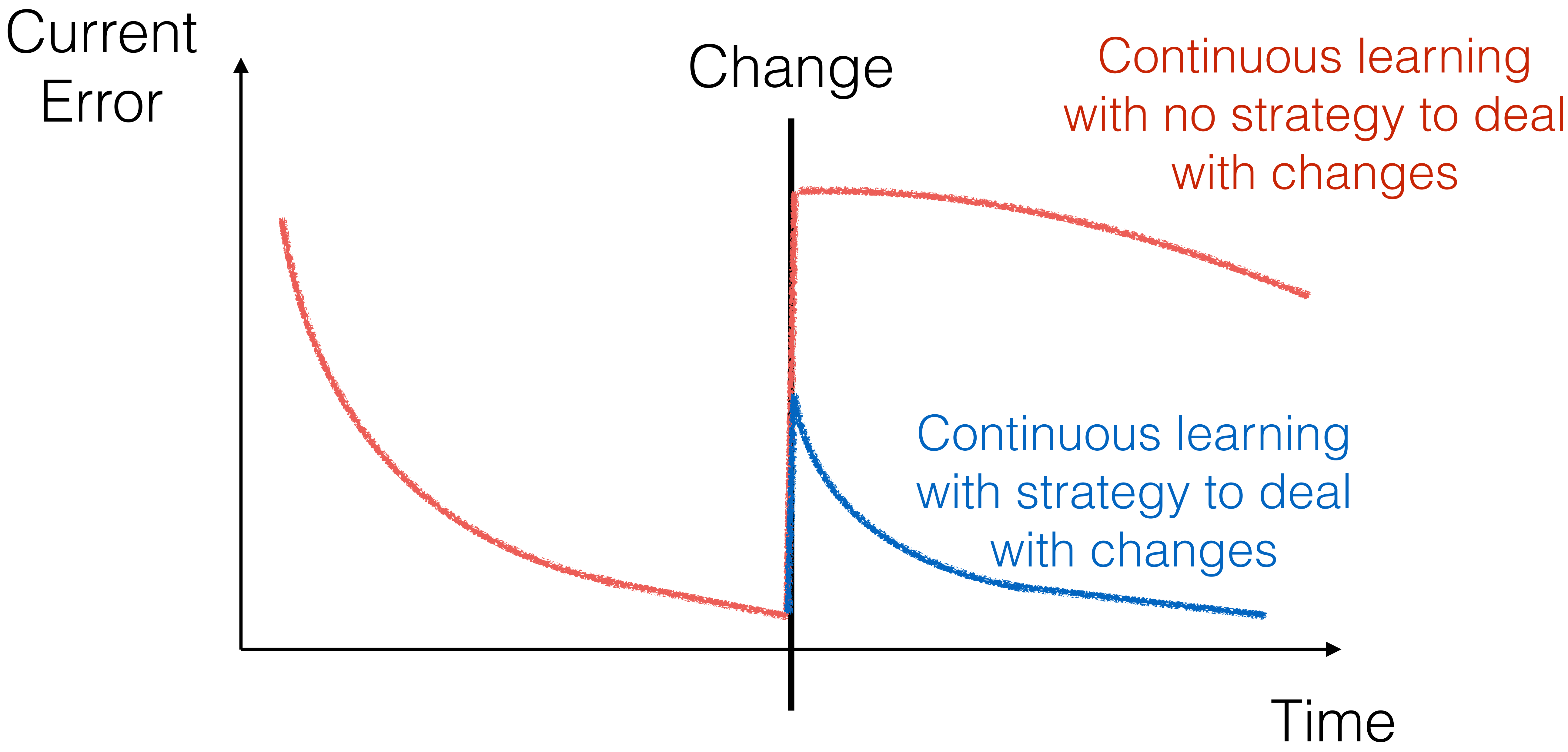
Why Are Changes A Problem?

[YouTube video posted by : https://youtu.be/NXUUMLJbPCE]

Why Are Strategies to Adapt to Changes Necessary?

- A model is trained on new examples over time.
- When a change happens, many examples describing the situation before the change have been used.
- It would take many examples of the new situation to compensate for the old examples that were learnt.
  - Adaptation is slow.

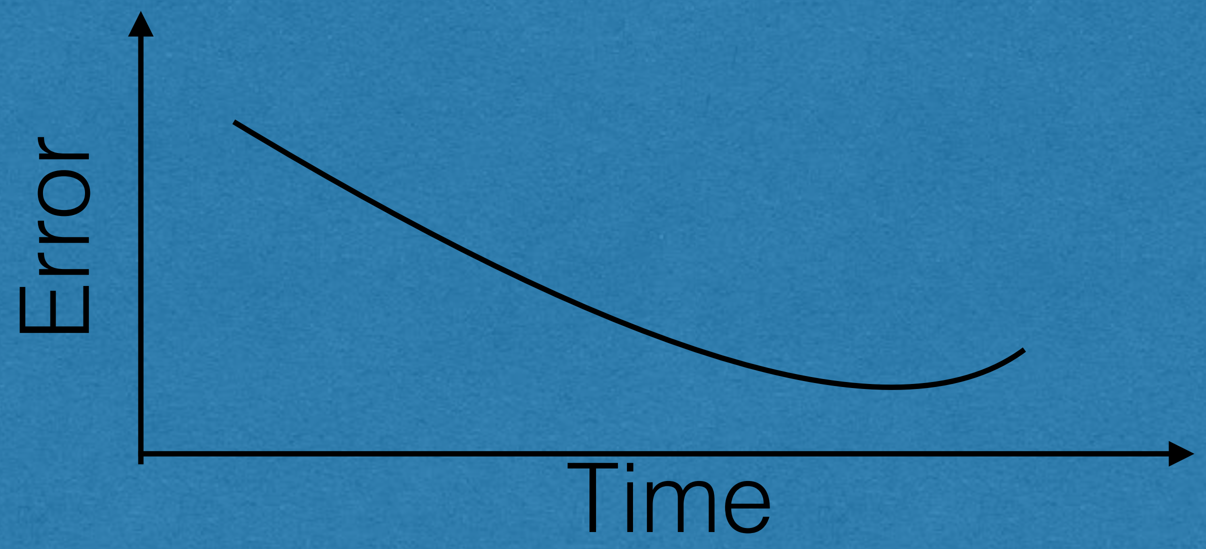
Why Are Strategies to Adapt to Changes Necessary?



Dealing with Concept Drifts — Principles

- Increase in error:
  - Error of a model is expected to reduce or stay constant over time if there is no change.
  - If the error of a model starts to increase, it is likely that there is a change causing this model to become inadequate.

Concept Drift Detection



Condition for change detection:

$$error(t) + stdev(t) \geq error_{min} + \alpha \cdot stdev_{min}$$

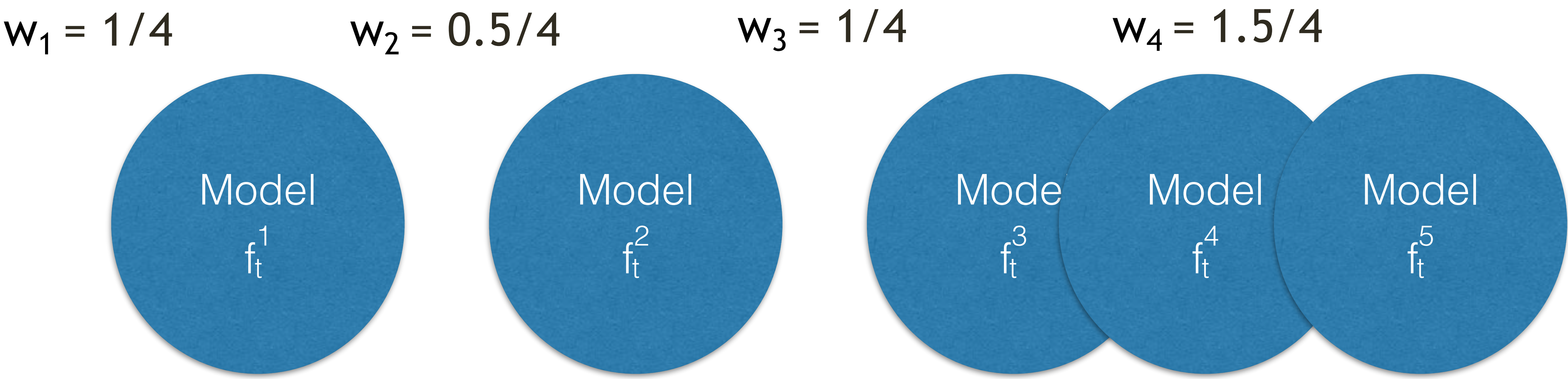
$$error(t) = \begin{cases} 0, & \text{if } t = 0 \\ error(t-1) + \frac{error_{ex}(t) - error(t-1)}{t+1}, & \text{if } t > 0 \end{cases}$$

Dealing with Concept Drifts — Principles

- Increase in error:
    - Error of a model is expected to reduce or stay constant over time if there is no change.
    - If the error of a model starts to increase, it is likely that there is a change causing this model to become inadequate.
- Building new model:
    - If current model(s) is(are) deemed inadequate for the current situation (increased error), create a new model to start learning the situation from scratch.

Ensembles for Adapting to Changes

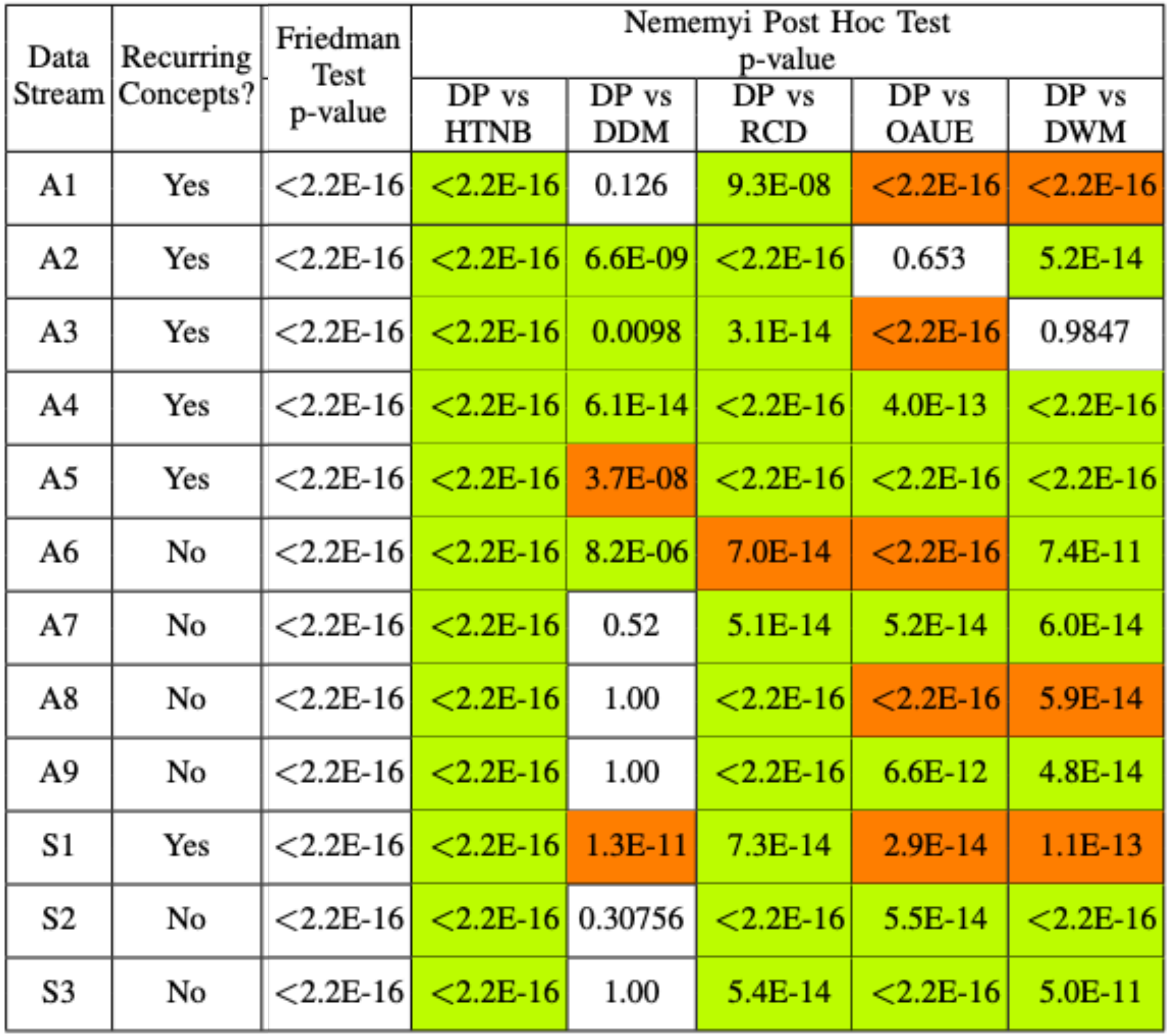
We can keep models representing different states, so that we can benefit from them when a previous state reoccurs.



$$Q_{i,k} = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}}$$

CHIU, C.W.; MINKU, L.L. . "Diversity-Based Pool of Models for Dealing with Recurring Concepts", *International Joint Conference on Neural Networks*, p. 2759-2766, July 2018.

Summary of Results



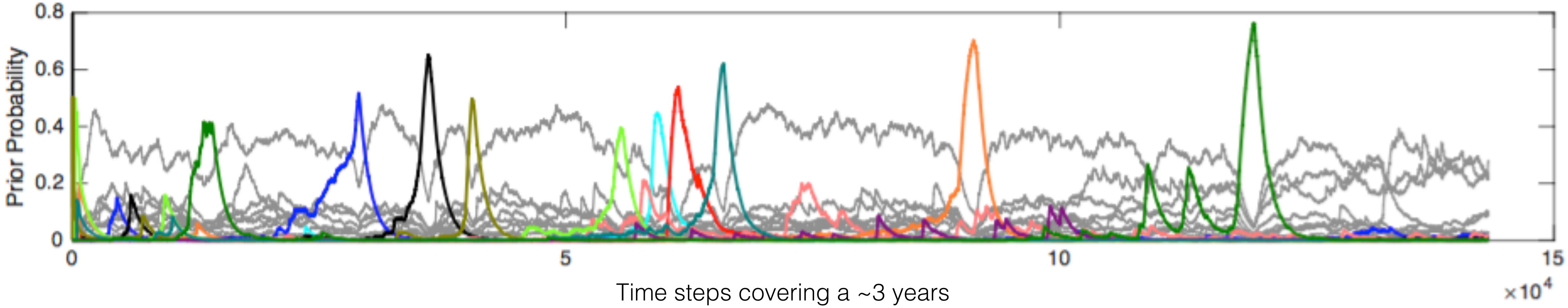
Data Stream	Recurring Concepts?	Friedman Test p-value	Nememyi Post Hoc Test p-value				
			DP vs HTNB	DP vs DDM	DP vs RCD	DP vs OAUE	DP vs DWM
A1	Yes	<2.2E-16	<2.2E-16	0.126	9.3E-08	<2.2E-16	<2.2E-16
A2	Yes	<2.2E-16	<2.2E-16	6.6E-09	<2.2E-16	0.653	5.2E-14
A3	Yes	<2.2E-16	<2.2E-16	0.0098	3.1E-14	<2.2E-16	0.9847
A4	Yes	<2.2E-16	<2.2E-16	6.1E-14	<2.2E-16	4.0E-13	<2.2E-16
A5	Yes	<2.2E-16	<2.2E-16	3.7E-08	<2.2E-16	<2.2E-16	<2.2E-16
A6	No	<2.2E-16	<2.2E-16	8.2E-06	7.0E-14	<2.2E-16	7.4E-11
A7	No	<2.2E-16	<2.2E-16	0.52	5.1E-14	5.2E-14	6.0E-14
A8	No	<2.2E-16	<2.2E-16	1.00	<2.2E-16	<2.2E-16	5.9E-14
A9	No	<2.2E-16	<2.2E-16	1.00	<2.2E-16	6.6E-12	4.8E-14
S1	Yes	<2.2E-16	<2.2E-16	1.3E-11	7.3E-14	2.9E-14	1.1E-13
S2	No	<2.2E-16	<2.2E-16	0.30756	<2.2E-16	5.5E-14	<2.2E-16
S3	No	<2.2E-16	<2.2E-16	1.00	5.4E-14	<2.2E-16	5.0E-11

CHIU, C.W.; MINKU, L.L. . "Diversity-Based Pool of Models for Dealing with Recurring Concepts", *International Joint Conference on Neural Networks*, p. 2759-2766, July 2018.

Challenges

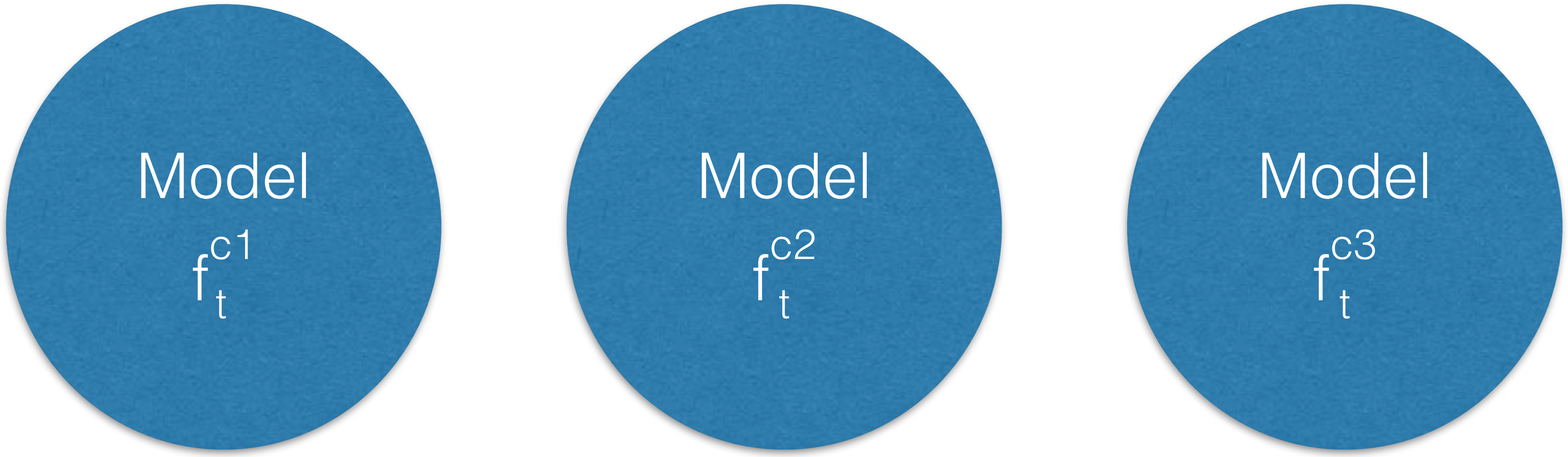
- Data streams are potentially infinite.
- Data frequently arrive with a high incoming rate.
- Data arrive in a certain order, and come from non-stationary environments.
- There may be class evolution and class imbalance.

Class Evolution: Emergence, Disappearance, Reoccurrence



- Certain classes may be underrepresented during certain periods of time (class imbalance).
- The ratio between different classes changes over time (dynamic class imbalance).

Class-Based Ensemble for Class Evolution (CBCE)



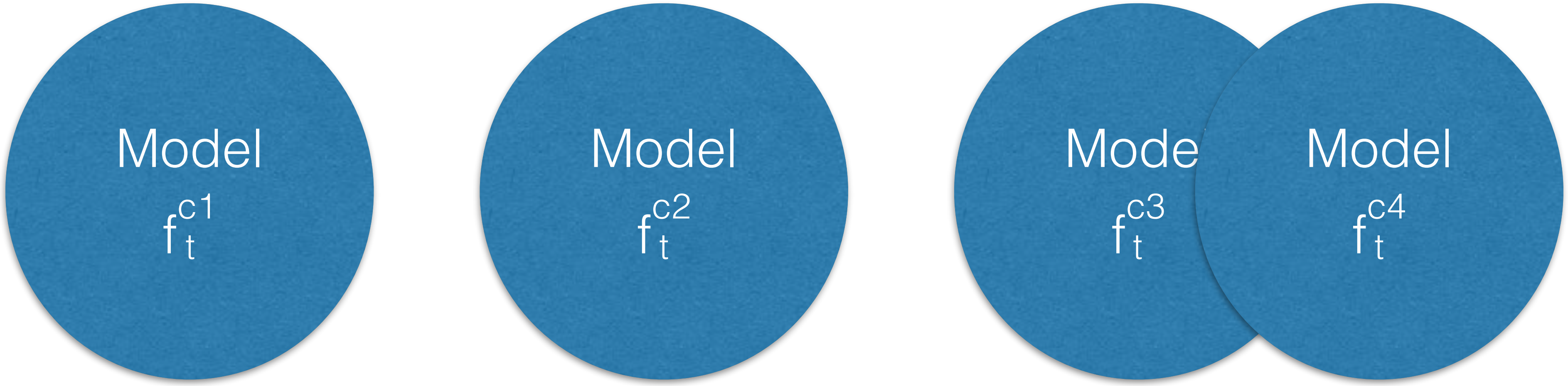
- Each base model is a binary classifier which implements the one-versus-all strategy.
  - Class represented by the model is the positive +1 class.
  - All other classes compose the negative -1 class.

- The class c_i predicted by the ensemble is the class with maximum likelihood $p(\mathbf{x}| c_i)$.

SUN, Y.; TANG, K.; MINKU, L.L.; WANG, S.; YAO, X. . "Online Ensemble Learning of Data Streams with Gradually Evolved Classes", *IEEE Transactions on Knowledge and Data Engineering*, v. 28, n. 6, p. 1532-1545, June 2016

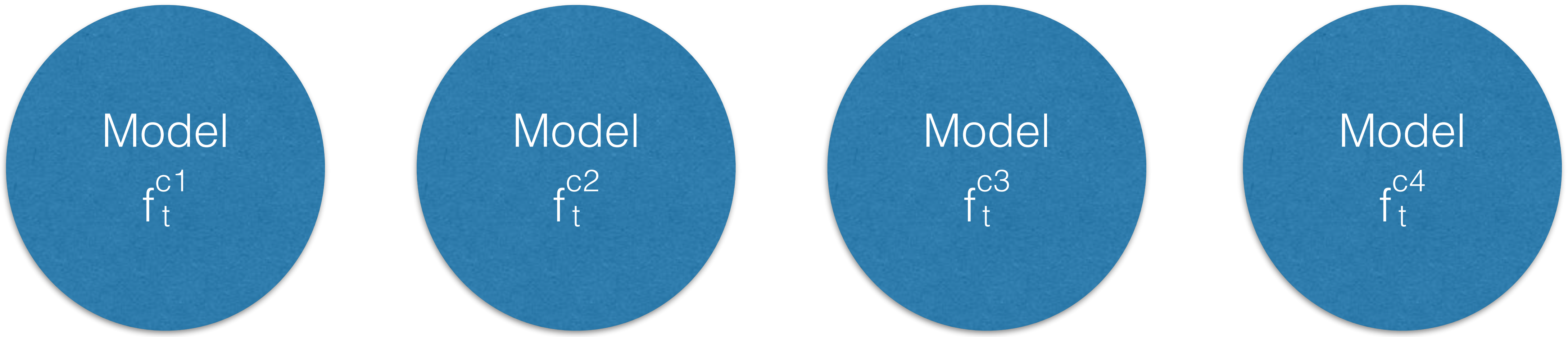
Dealing with Class Evolution

- The use of one base model for each class is a natural way of dealing with class emergence, disappearance and reoccurrence.



Dealing with Class Evolution

- The use of one base model for each class is a natural way of dealing with class emergence, disappearance and reoccurrence.



Model f_t^{c1}

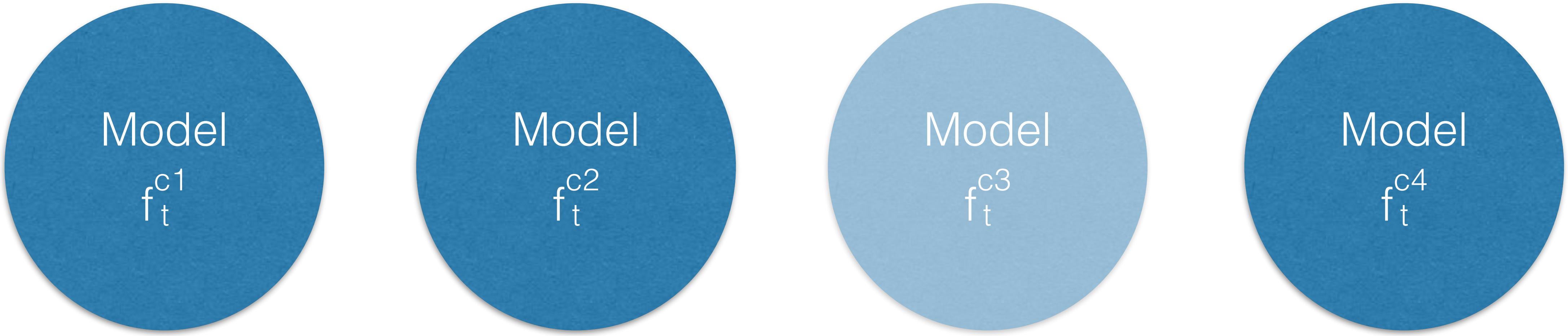
Model f_t^{c2}

Model f_t^{c3}

Model f_t^{c4}

Dealing with Class Evolution

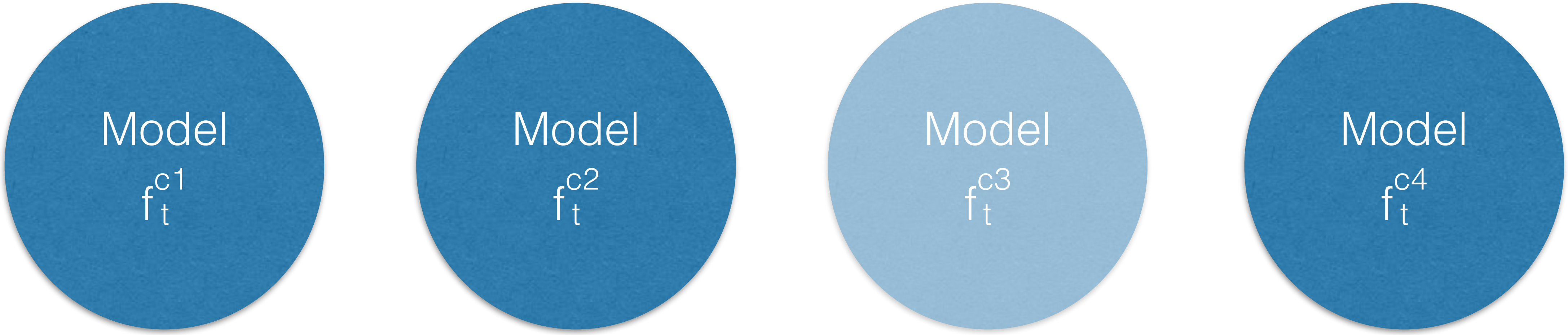
- The use of one base model for each class is a natural way of dealing with class emergence, disappearance and reoccurrence.



- How to determine whether to eliminate a base model?
  - Monitor the proportion of examples coming from its corresponding class.
  - If this number falls below a given threshold, temporarily eliminate that base model.

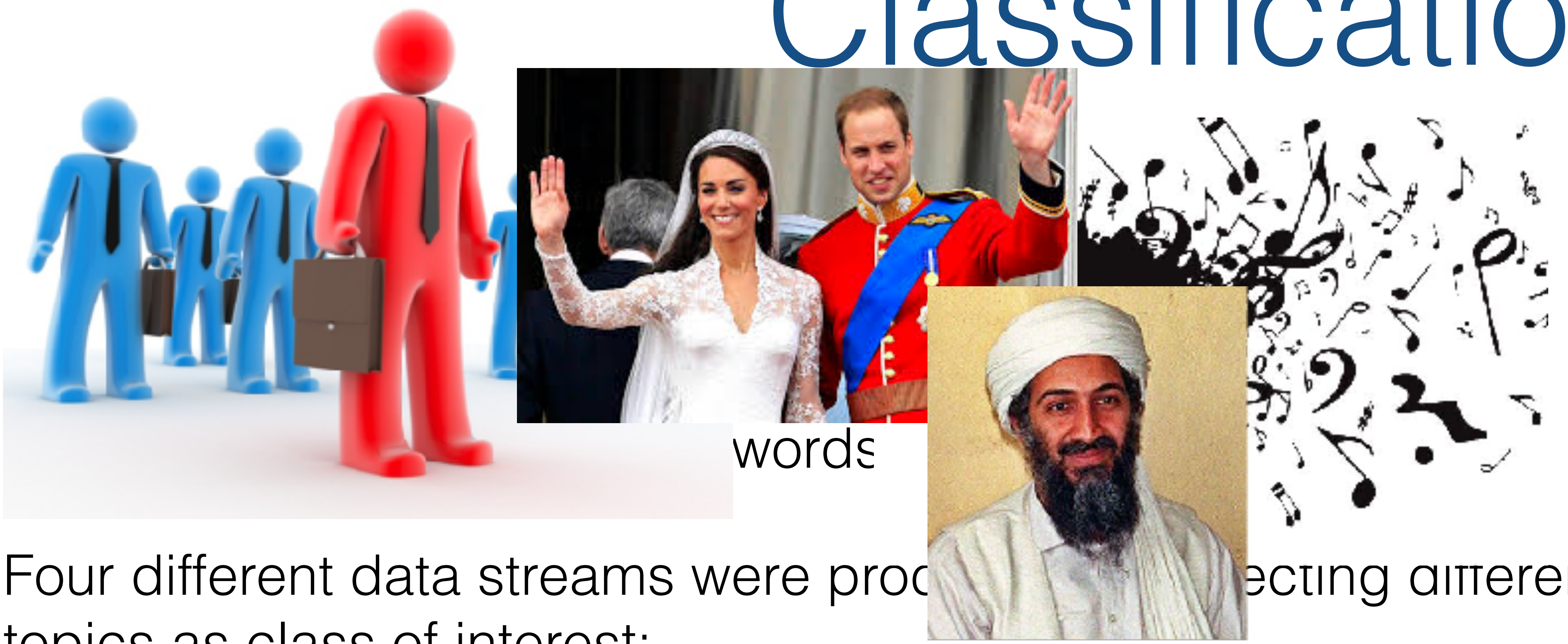
Dealing with Dynamic Class Imbalance

- The positive class is likely to be a minority.



$(x_1,+)$ $(x_2,-)$
$(x_3,+)$ $(x_4,-)$
$(x_5,-)$
$(x_6,-)$
$(x_7,-)$

T Classification

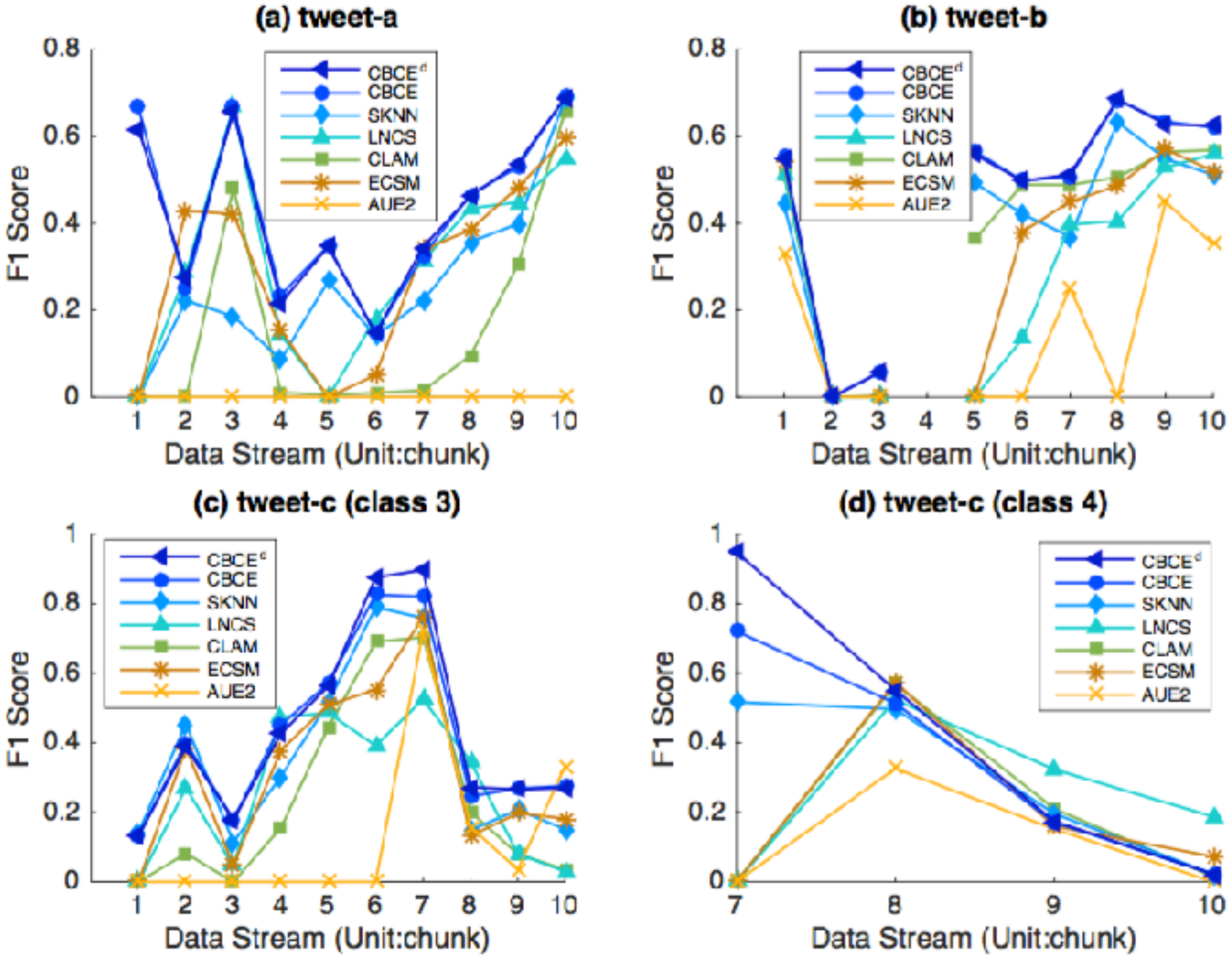


- words

- Four different data streams were proc ecting different topics as class of interest:
  - Tweet-*a*: 39,600 tweets, 2 stable and 1 evolved class.
  - Tweet-*b*: 15,004 tweets, 2 stable and 1 evolved class.
  - Tweet-*c*: 68,750 tweets, 2 stable and 2 evolved classes.
  - Tweet-*d*: 143,381 tweets, 9 stable and 11 evolving classes.

- CBCE is evaluated with and without Gama et al. (2004)'s drift detection method, denoted as CBCE^d and CBCE.

Real World Data Streams - F1-Score of the Evolved Class


(a) tweet-a

(b) tweet-b

(c) tweet-c (class 3)

(d) tweet-c (class 4)

$$F_1 = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}.$$

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

Conclusions

- The whole is greater than the sum of the parts!
- Grouping models into ensembles can help to increase predictive performance:
  - In standard offline learning.
  - In continuous learning in non-stationary environments.
- When analysing your data, be careful to check whether:
  - you have very large amounts of data;
  - you have a data stream non-stationary learning problem;
  - you have a class imbalanced problem.

Thank you!