Data Mining
(Mining Knowledge from Data)

Combining Models

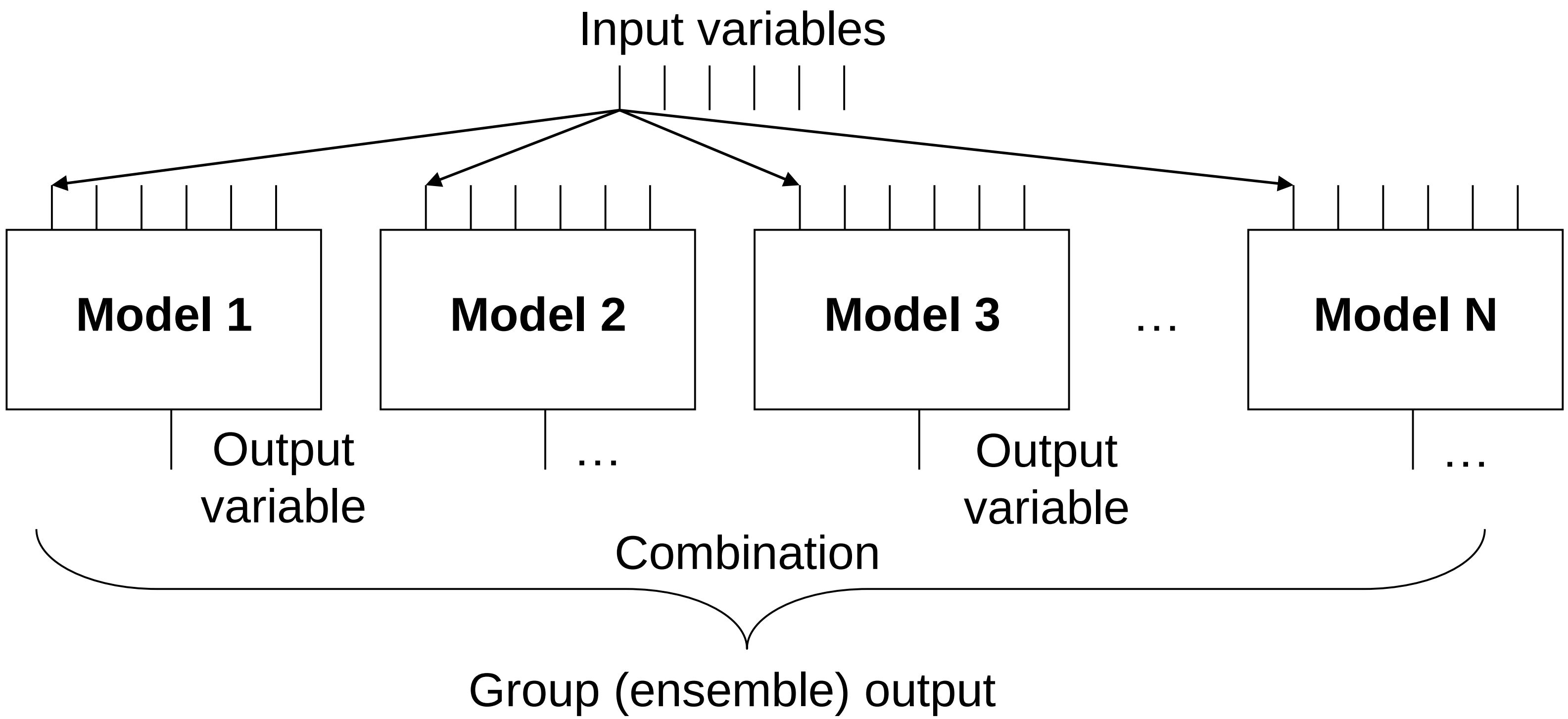
Marcel Jiřina, Pavel Kordík

ČESKÉ
VYSOKÉ
UČENÍ
TECHNICKÉ
V PRAZE

FIT

The principle of combining models

- A group of models (e.g. decision trees) will learn the same (similar) task.
- Outputs of the learned models are combined.



Diversity of ensemble models

- What happens when all models will be the same?

  => Degradation to one model.

- How do we ensure that all the models will be diverse?
  - Different sets of training data (initial conditions)
  - Different methods of construction of models

- How the diversity of models can be measured?
  - Deviations of outputs for each test data.
  - Structural differences

Does it work?

- We want to determine the conditions under which it pays to combine models.
- We are interested why ensembling works.

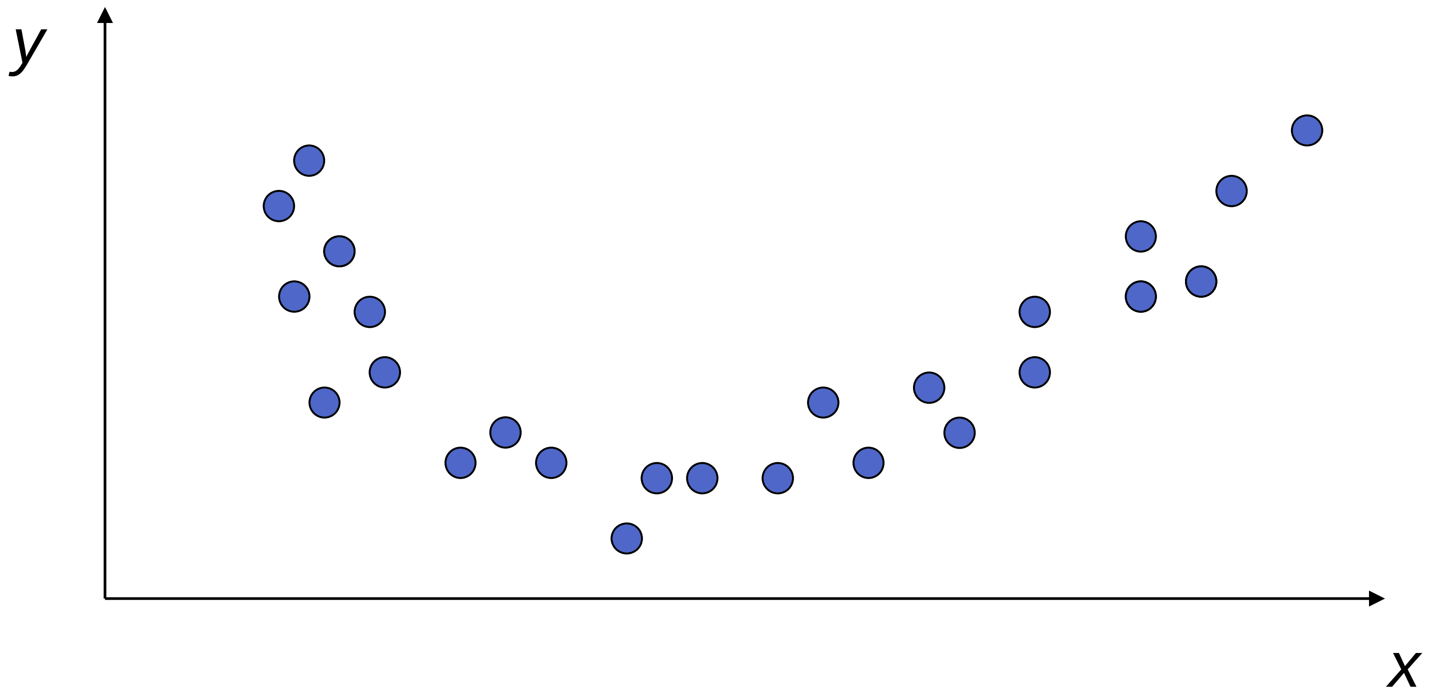
- We need to analyze what caused the error of the models

Decomposition bias/variance

- The error of a model consists of 3 components :
- **Noise**
  - Quantifies the deviation of output y from the optimal model
  - Predicted value has a non-zero variance, and so it can never be predicted exactly
  - This error can not be reduced
- **Bias**
  - An error of an average model with respect to the optimal one.
  - Big for weak-learners, which are not complex enough to capture the pattern in data
- **Variance**
  - How much differs prediction $\hat{y}(\underline{x})$ for different learning sets *LS*.
  - While learning more models on subsets of the training data, the output of these models can differ significantly
  - Big for over-learned models

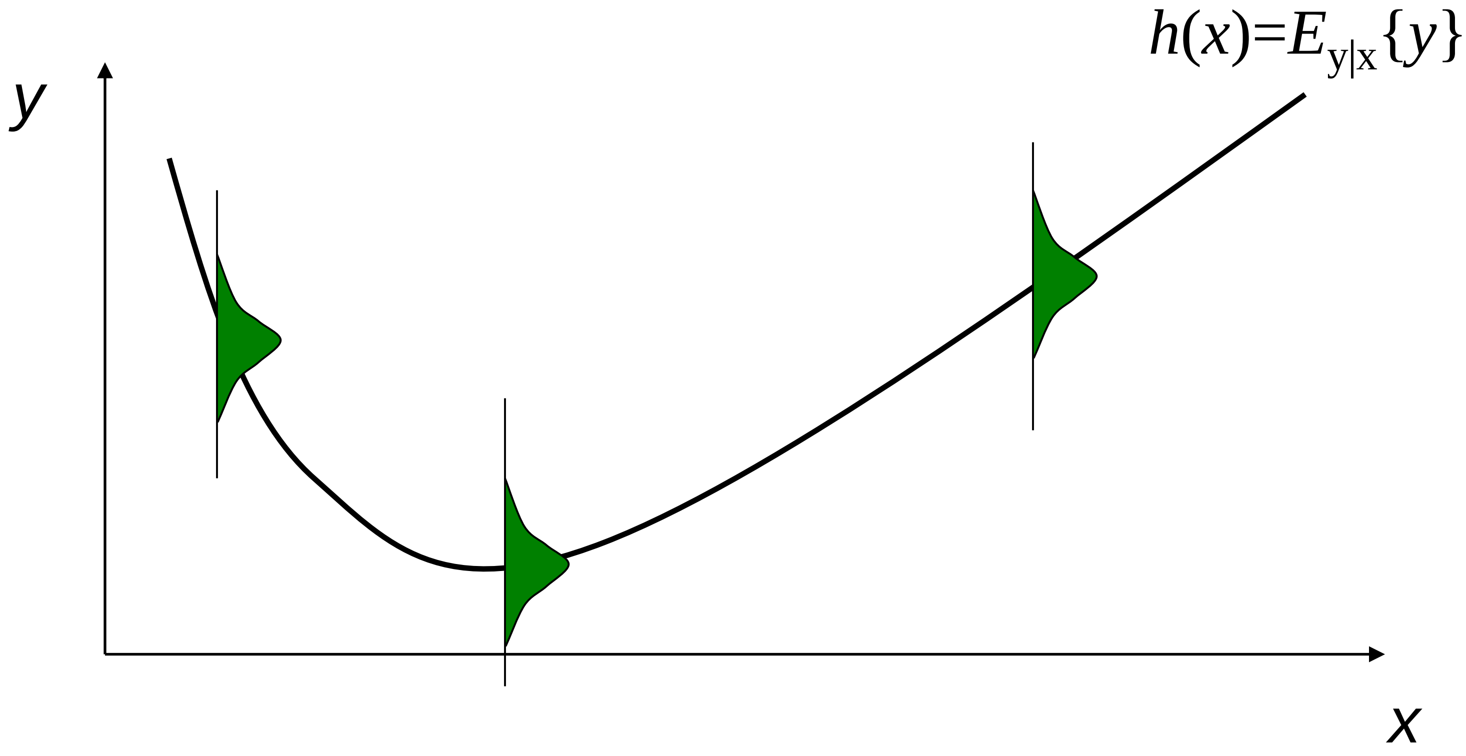
Example (1)

- Find an algorithm that produces the best possible models for the following data:



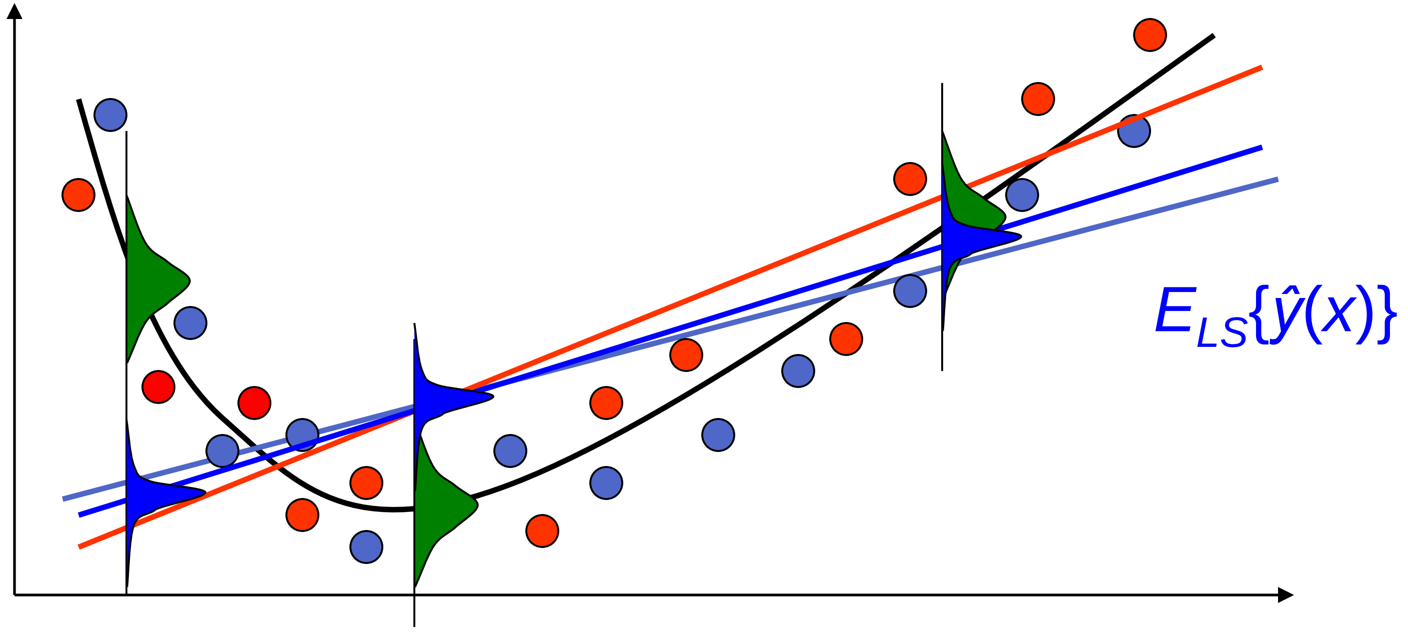
Example (2)

- Optimal model:
  - Input *x,* random variable uniformly distributed in the interval [0,1]
  - $y=h(x)+\varepsilon$, where $\varepsilon \sim N(0,1)$ is Bayes model y and noise



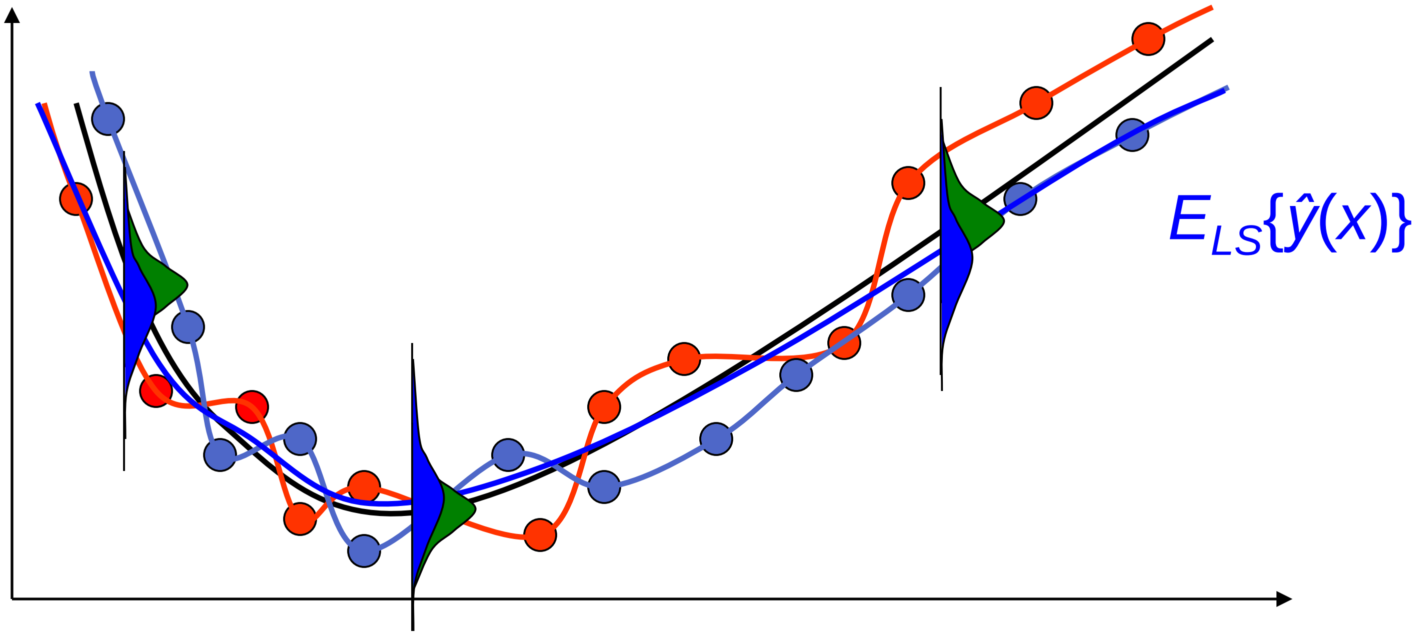
Example – algorithm of linear regression

- The models have a low variance, but the large bias $\Rightarrow$ under-learned (not sufficiently learned)



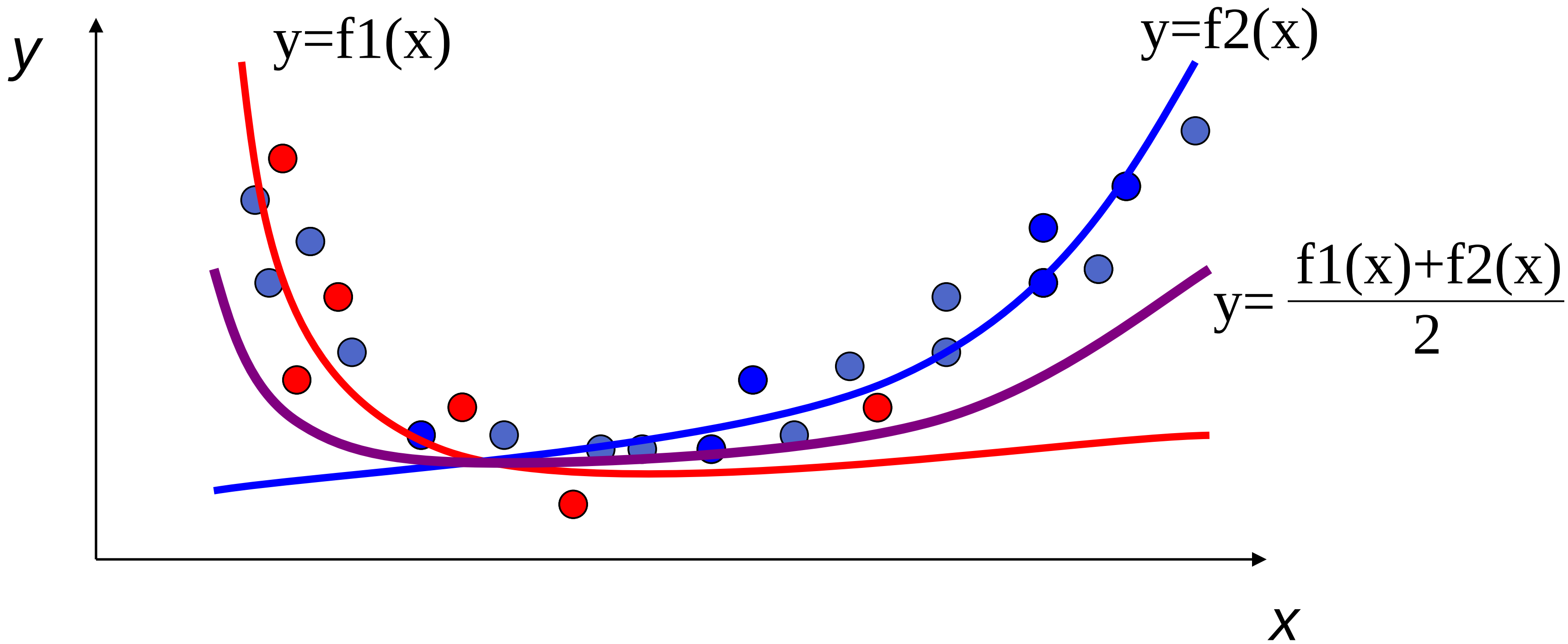
Example – algorithm RBFN with a number of neurons equal to the size of the dataset

- Low bias, large variance of models ⇒ overlearning

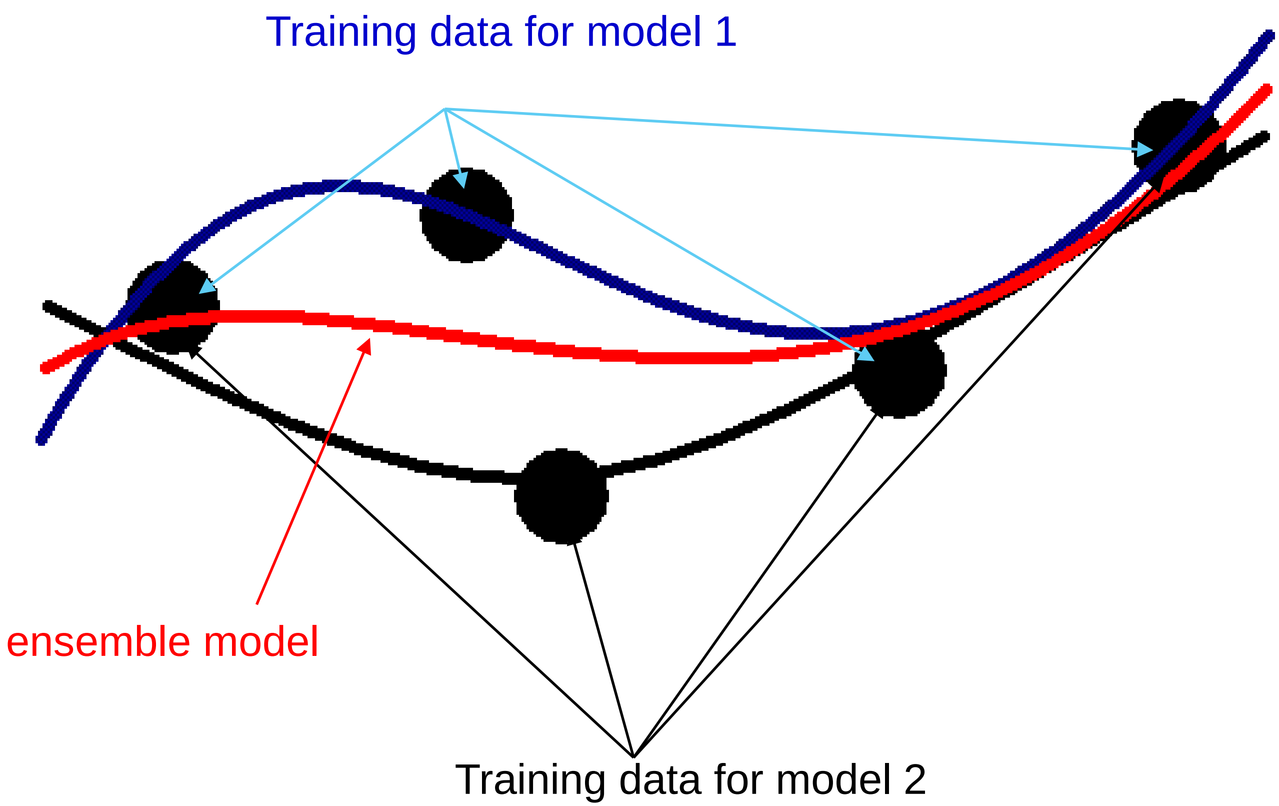


Back to combining models

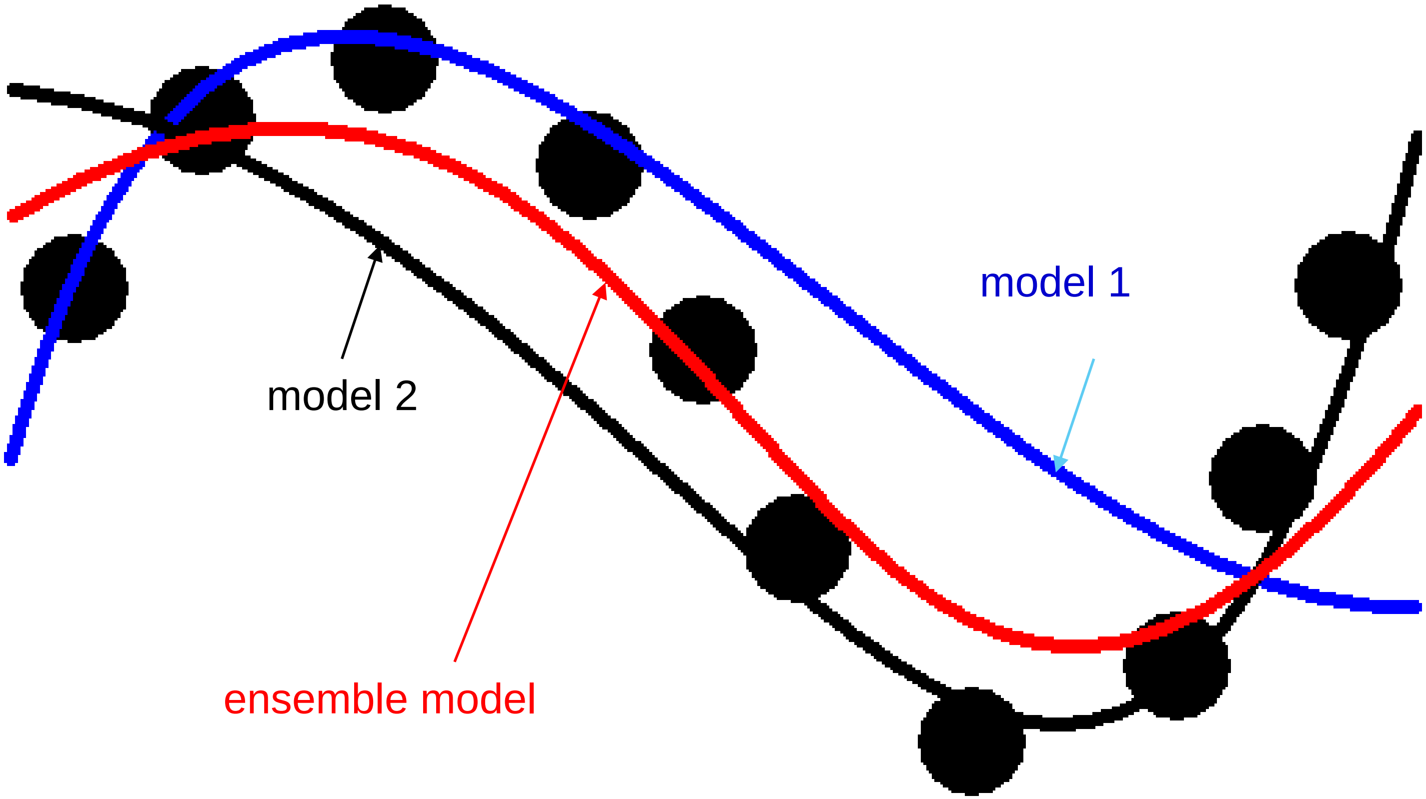
- What happens if I learn two simple models on different subsets of the learning set?



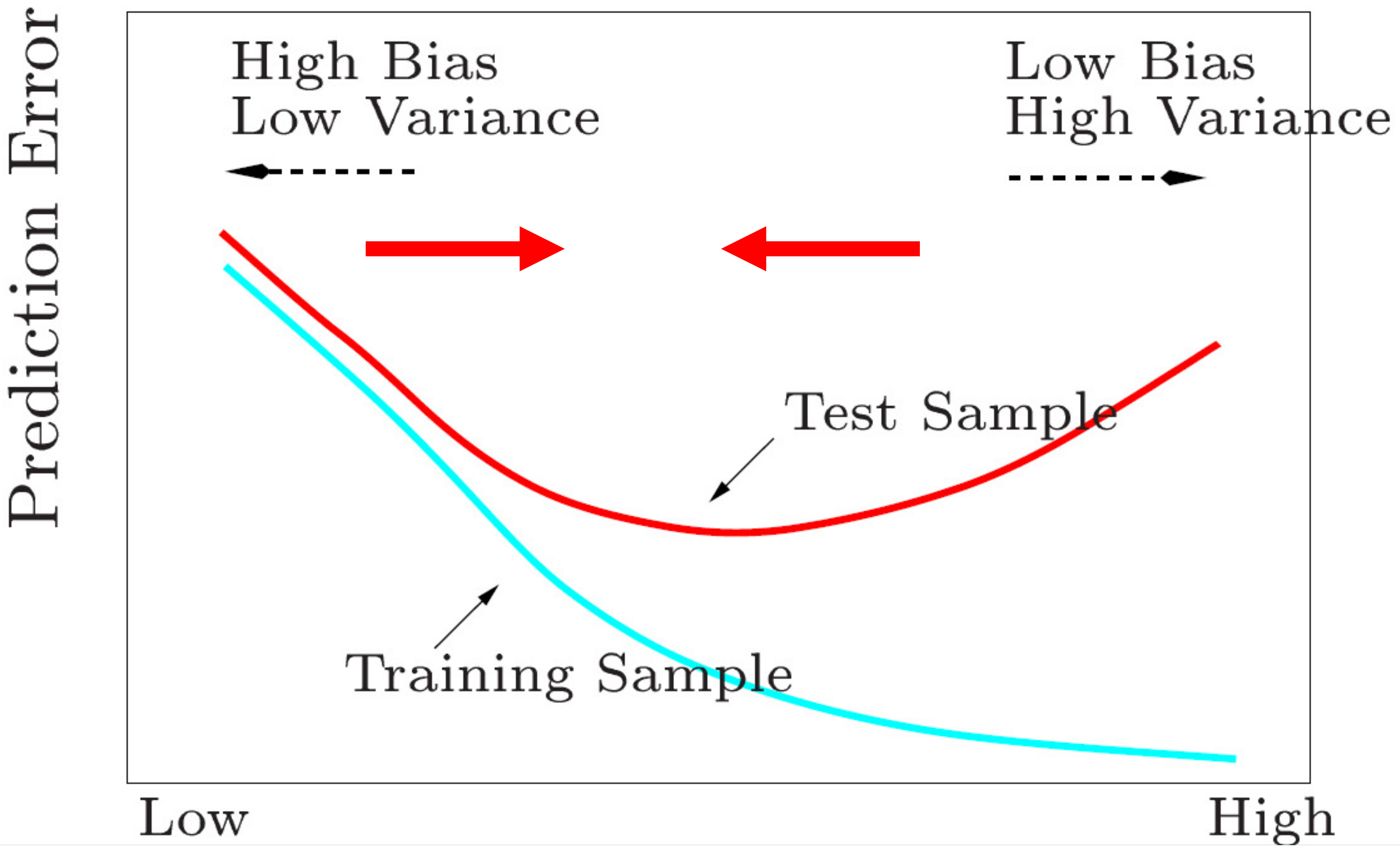
Ensembling reduces **variance**



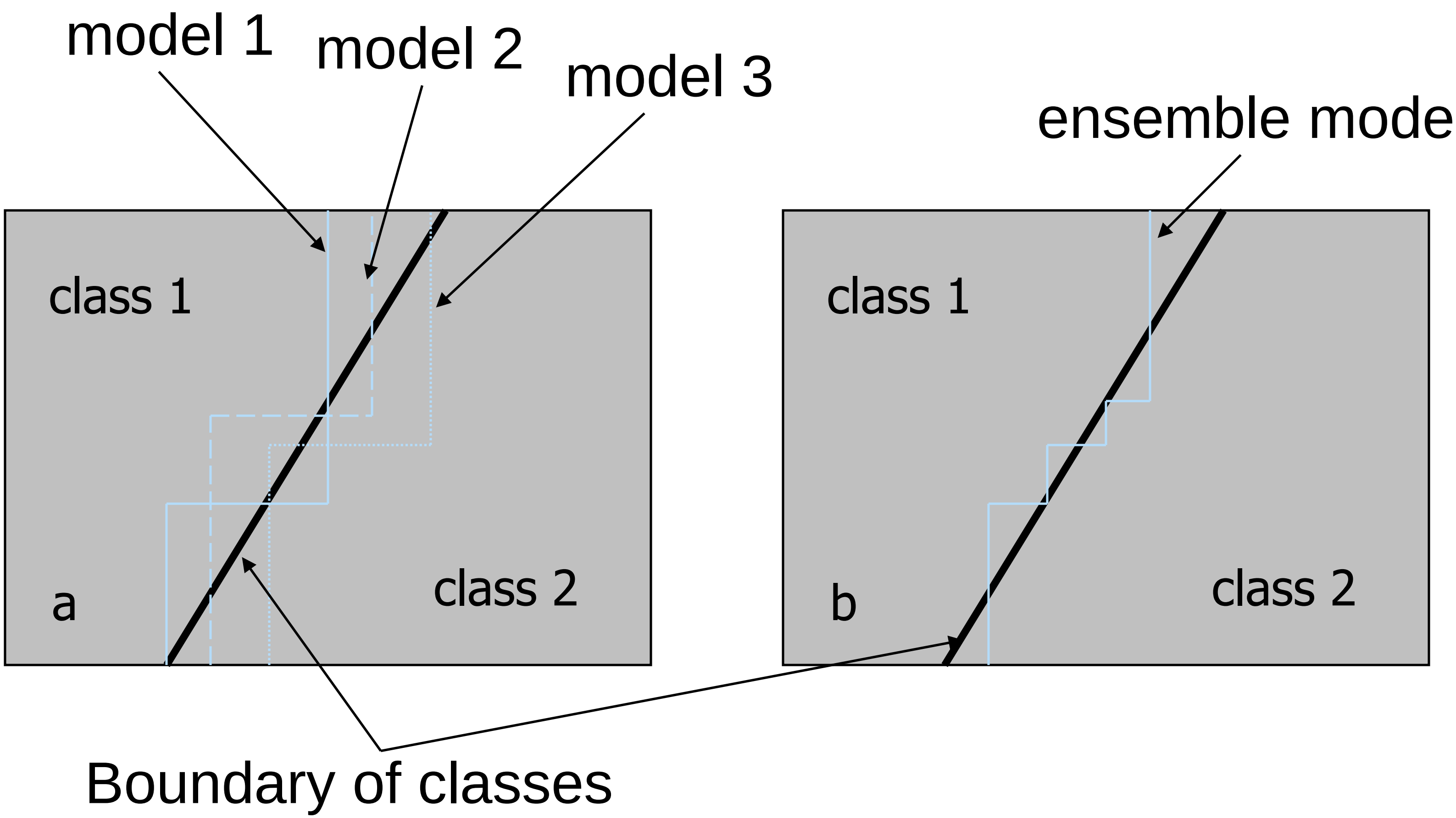
Ensembling reduces **bias**



Thus:


Prediction Error
High Bias
Low Variance
Low Bias
High Variance
Test Sample
Training Sample
Low
High

Similarly for classification?

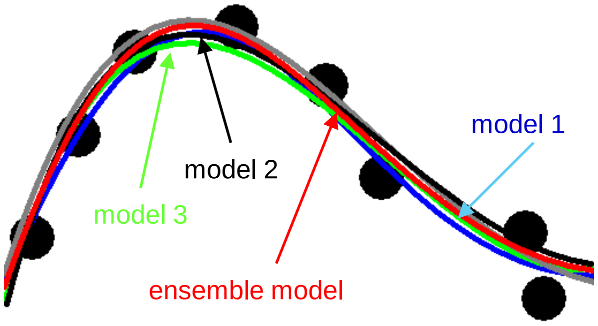


Pictures from TCD AI Course, 2005

Is **bias** or **variance** reduced?

What models can be combined?

- What happens when optimally learned models are combined?



- Simple models (i.e. Weak learners) can be favourably combined.

Bias is reduced

- Models must be diverse! They must show different errors on individual training patterns.

Variance is reduced

Popular ensemble methods

- Bagging (Bootstrap Aggregating)
  - Models are learn independently and their outputs are combined easily
- Boosting
  - Models are learned sequentially, the training data are dependent on the mistakes (errors) of previous models
- Stacking
  - Models are learning independently, the outputs are combined by learning of a special model

Bagging

- **B**ootstrap **AGG**regat**ING**

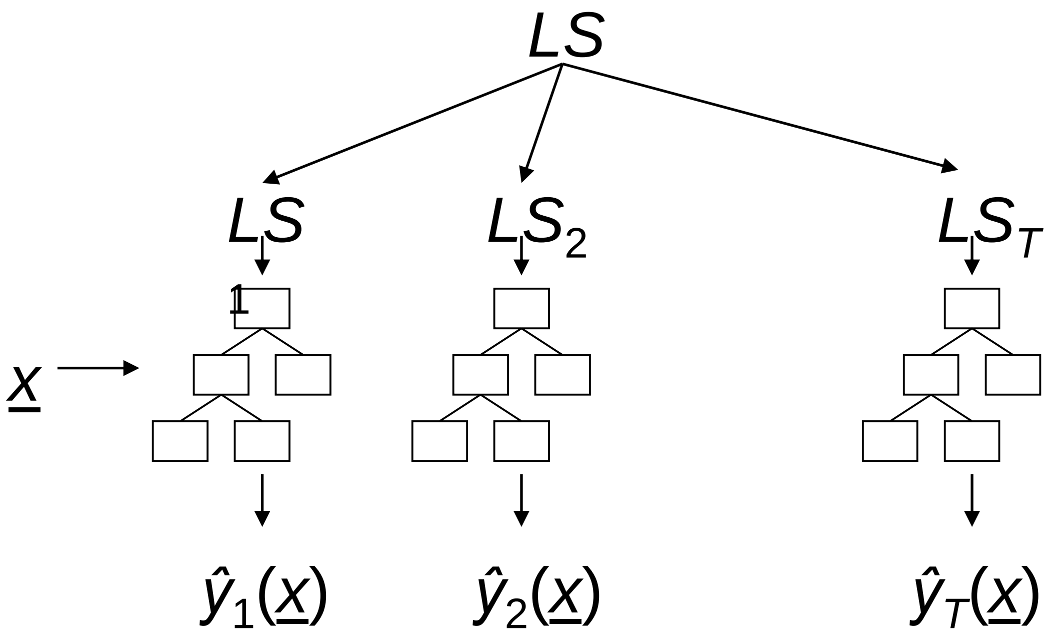
- Idea:
  - More models are learned independently
  - As a learning set for each model we use a subset of randomly (with replacement) selected instances from the original data
  - We aggregate the output of multiple models in the final outcome

Example of bootstrap (Opitz, 1999)

- We use selection with replacement
  - in the selected subset an instance from the original dataset can occure more than once or not at all

Training instances	1	2	3	4	5	6	7	8
Sample 1	2	7	8	3	7	6	3	1
Sample 2	7	8	5	6	4	2	7	1
…	…	…	…	…	…	…	…	…
Sample M	4	5	1	4	6	4	3	8

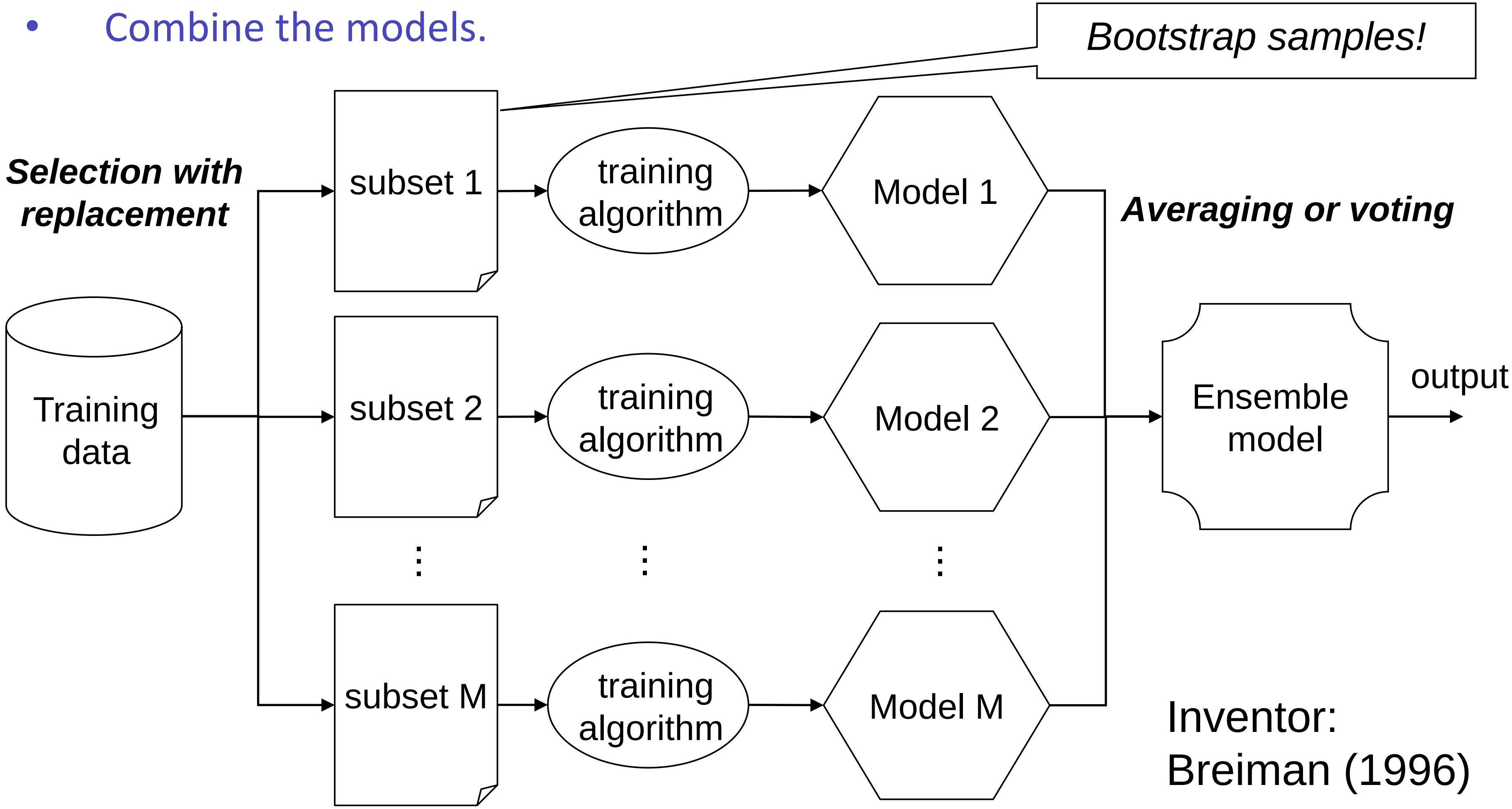
Combining of results (outputs)



- For regression:
    - Average value of $\hat{y}_1, \hat{y}_2, \dots, \hat{y}_T$
- For classification:
    - Majority class of $\hat{y}_1, \dots, \hat{y}_T$

Bagging (Bootstrap aggregating)

- By the selection with replacement create M training subsets with ***n*** samples (instead of a single original set with ***n*** samples).
- Create models for each training subset.
- Combine the models.



Inventor:
Breiman (1996)

Bagging

- The diversity of individual models is caused by the random selection of a training subset

- Usually bagging substantially reduces the variance and preserves the bias of models.

Random forests

- To the bagging we still add a randomly selected subset of input attributes
- Thus:
  - Create a decision tree on a bootstrap subset
  - Find the best split among **k** attributes of a random subset (not among all attributes as obvious)
  - (= bagging, when **k** equals the number of attributes)
- Do we estimate the influence of **k**?
  - Lower **k** reduces variance and increases bias

Boosting

- Idea:
- Learning the models runs sequentially - to learn the i-th model, we need to know the performance of the previous model
- For each model we selected the training set at random
  - The probability of selecting a new instance to the training set is not the same for all models
  - <u>We select more likely instances on which the previous model has lower performance (higher error)</u>
  - For the first model is the probability the same for all instances
- We aggregate the output of multiple models in the final outcome

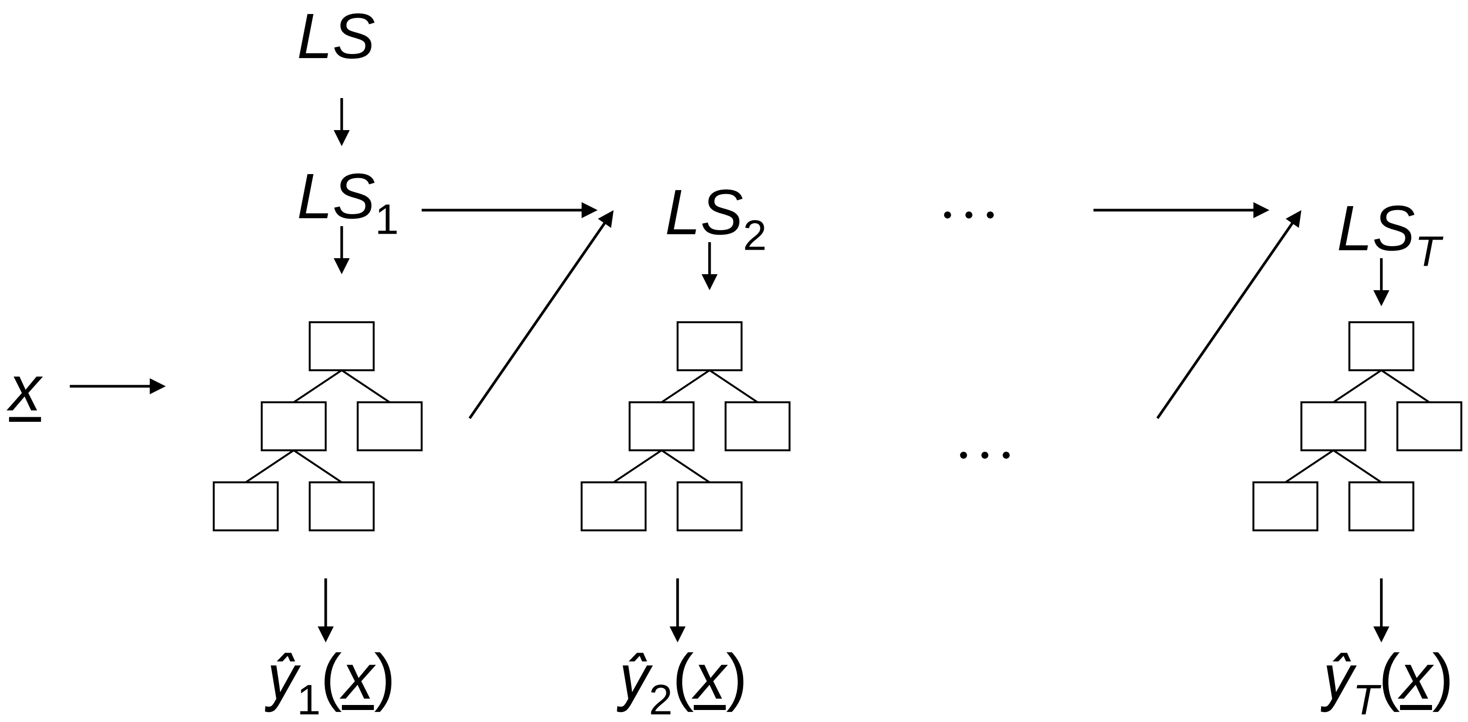
Selection of training set

- At the beginning all models have the same weight
- After learning a model we will increase weights of poorly classified patterns and reduce weights of properly classified patterns

- **Example**: The pattern 4 is incorrectly classified
- Its weight is gradually increased and hence the probability of selecting the patter to the learning set

Original Data	1	2	3	4	5	6	7	8	9	10
Boosting (Round 1)	7	3	2	8	7	9	4	10	6	3
Boosting (Round 2)	5	4	9	4	2	5	1	7	4	2
Boosting (Round 3)	4	4	8	10	4	5	4	6	3	4

Combining of outcomes (results)



- For regression:
  - Average value of $\hat{y}_1, \hat{y}_2, \dots, \hat{y}_T$
- For classification:
  - Weighted majority of $\hat{y}_1, \hat{y}_2, \dots, \hat{y}_T$

Stacking

- A meta model for combination of outputs of more models is used (compared with a simple averaging or voting)
  - The outputs of ensemble models are used as training data for the meta model
- Ensemble models are usually learned by different algorithms
  - It causes a variety of models

Slide from Ensembles of Classifiers by *Evgueni Smirnov*

Arguments against combining models?

- Occam's razor - keep it simple
  - It is better to have a single optimal model than a combination of many models
  - ... but how to find the optimal model?
  - Domingos, P. Occam's two Razors: the sharp and the blunt. KDD 1998.
- Combining models often camouflages imperfections of methods producing under-learned or over-learned models
- Combining models we get a model with poorer results on test data than have the combined models

Arguments for combining models

- Mostly I improve my results on test data
  - Algorithms are implicitly set, it is necessary to experiment with their configuration, to produce models optimized on concrete data
- I get awareness about the certainty of the model
  - When individual models vary a lot for a given input vector, we are probably outside the training data set
- Netflix prize

Questions

- What new information is obtained by using multiple models versus using only one model?

- Can ensemble increate prediction? For what models?

- What are disadvantages of the ensemble prediction?

Further improvement

- Hierarchical combining of models
- Meta-learning templates
- Breeding of ensemble topology on concrete data

  ... more in MI-MVI