Introduction to Statistical Machine Learning

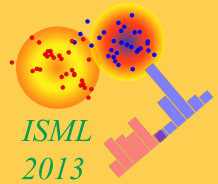
Christfried Webers

Statistical Machine Learning Group
NICTA
and
College of Engineering and Computer Science
The Australian National University

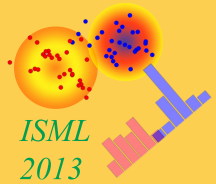
Canberra
February – June 2013

(Many figures from C. M. Bishop, "Pattern Recognition and Machine Learning")

Christfried Webers
NICTA
The Australian National University



Outlines



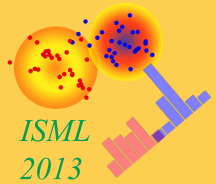
Part XXIII

Combining Models

Motivation for Combining Models

- Coming up with a very precise prediction rule can be very hard.
- It may be easier to come up with a number of not so precise prediction rules.
- Can combine them to make better predictions?
- Example: A team of people often performes better than an individual.
- Bayesian Averaging versus Model Combination.
- Tree based methods.

©2013 Christfried Webers
NICTA
The Australian National University



Model Combination

- Review density estimation via a mixture of Gaussian.
- Model specified via the joint distribution

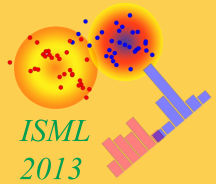
$$p(\mathbf{x}, \mathbf{z})$$

with $\mathbf{x}$ the observed variable, and $\mathbf{z}$ the unobserved variable.

- Probability over the observed variable is found by marginalisation

$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{z}).$$

©2013 Christfried Webers
NICTA
The Australian National University



Model Combination

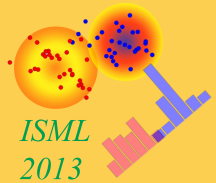
- For the Gaussian mixture we get for each observed data point $p(\mathbf{x})$

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k).$$

- If all the data are i.i.d, we can find the probability of all data $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$

$$p(\mathbf{X}) = \prod_{n=1}^{N} p(\mathbf{x}_n) = \prod_{n=1}^{N} \left[\sum_{\mathbf{z}_n} p(\mathbf{x}_n, \mathbf{z}_n) \right].$$

Christfried Webers
NICTA
The Australian National University



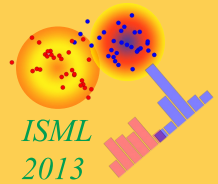
Bayesian Model Averaging

- Now suppose different models index by $h = 1, \ldots, H$ with prior probabilities $p(h)$. (One model might be a mixture of Gaussian, another model a mixture of Cauchy distributions.)
- Now the marginal distribution over the data set $\mathbf{X}$ can be expressed as Bayesian Model Averaging

$$p(\mathbf{X}) = \sum_{h=1}^{H} p(\mathbf{X} \mid h)\, p(h).$$

- Interpretation: Just one model is responsible for creating the data $\mathbf{X}$, but we do not know which one. Therefore $p(h)$ reflects the uncertainty of which model is responsible.
- If more and more data become available, the posterior probability $p(h \mid \mathbf{X})$ will become increasingly focussed on just one model.

©2013 Christfried Webers
NICTA
The Australian National University



Model Combination vs. Bayesian Model Averaging

- Bayesian Model Averaging : One model is responsible for generating all data.

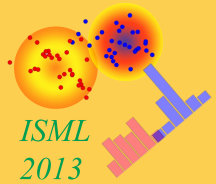
$$p(\mathbf{X}) = \sum_{h=1}^{H} p(\mathbf{X} \mid h)\, p(h).$$

- Combine Multiple Models : Different data points can be created from different values of the latent variable, and therefore by different components.

$$p(\mathbf{X}) = \prod_{n=1}^{N} p(\mathbf{x}_n) = \prod_{n=1}^{N} \left[\sum_{\mathbf{z}_n} p(\mathbf{x}_n, \mathbf{z}_n) \right].$$

- Same applies to predictive density $p(\mathbf{x} \mid \mathbf{X})$ or conditional distributions such as $p(\mathbf{t} \mid \mathbf{x}, \mathbf{X}, \mathbf{T})$ instead of $p(\mathbf{x})$.

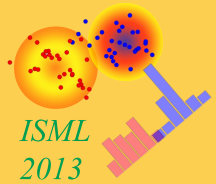
Christfried Webers
NICTA
The Australian National University



Committees

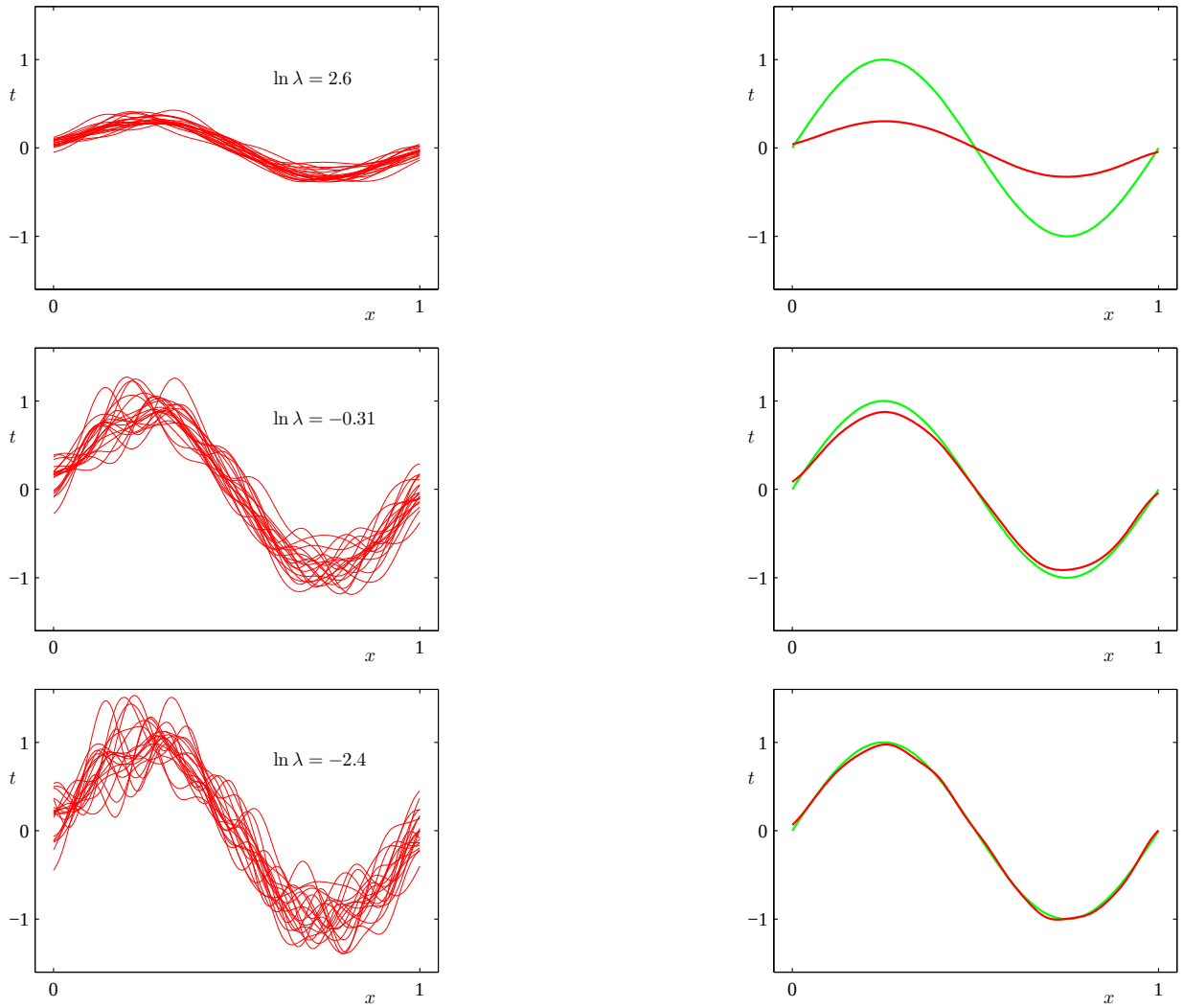
- Construct a committee by averaging the predictions of a set of individual models.
- Remember training multiple polynomials using the sinusoidal data, and then averaging?
- Using low-bias models (higher order polynomials) and averaging resulted in better prediction.

Christfried Webers
NICTA
The Australian National University



The Bias-Variance Decomposition

Dependence of bias and variance on the model complexity



Left: Result of fitting the model to 100 data sets, only 25 shown.
Right: Average of the 100 fits in red, the sinusoidal function

©2013
Christfried Webers
NICTA
The Australian National University

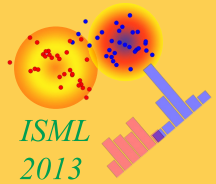
Where does the variability come from?

- Only one data set available.
- How to create new data sets?
- Use for instance bootstrap approach: Given N data points $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$, draw a set of N points from this data set with replacement. The new set $\mathbf{X}'$ may contain some of the data $\mathbf{x}_n$ multiple times, others may be absent.
- Created M 'artificial' data sets and train several predictive models $y_m(\mathbf{x})$ where $m = 1, \ldots, M$.
- The committee prediction y_{COM} is then given by

$$y_{\text{COM}} = \frac{1}{M} \sum_{m=1}^{M} y_m(\mathbf{x})$$

- Also called bootstrap aggregation or bagging.

©2013 *Christfried Webers*
NICTA
The Australian National University



How big is the error?

- Suppose true regression function is given by $h(\mathbf{x})$.
- Write the output of each model as sum of $h(\mathbf{x})$ and some model dependend error $\epsilon_m(\mathbf{x})$.

$$y_m(\mathbf{x}) = h(\mathbf{x}) + \epsilon_m(\mathbf{x})$$

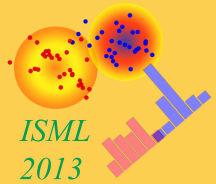
- Average sum-of-squares error by each model individually is

$$\mathbb{E}_{\mathbf{x}}\left[\{y_m(\mathbf{x}) - h(\mathbf{x})\}^2\right] = \mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})^2\right]$$

  where $\mathbb{E}_{\mathbf{x}}[\cdot]$ is the expectation with respect to the distribution of the data $\mathbf{x}$.
- Average by all models acting individually

$$E_{\mathsf{AV}} = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})^2\right]$$

Christfried Webers
NICTA
The Australian National University



How big is the error?

- Average by all models acting individually

$$E_{\mathrm{AV}} = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})^2\right]$$

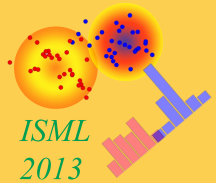
- But the expected error from the committee is

$$E_{\mathrm{COM}} = \mathbb{E}_{\mathbf{x}}\left[\left\{\frac{1}{M}\sum_{m=1}^{M}y_m(\mathbf{x}) - h(\mathbf{x})\right\}^2\right] = \mathbb{E}_{\mathbf{x}}\left[\left\{\frac{1}{M}\sum_{m=1}^{M}\epsilon_m(\mathbf{x})\right\}^2\right]$$

- Assume errors have zero mean $\mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})\right] = 0$ and are uncorrelated $\mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})\epsilon_l(\mathbf{x})\right] = 0$ for $m \neq l$.
- 'Dramatic' error reduction by using a committee.

$$E_{\mathrm{COM}} = \frac{1}{M}E_{\mathrm{AV}}$$

Christfried Webers
NICTA
The Australian National University



How big is the error?

- Average by all models acting individually

$$E_{\text{AV}} = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})^2\right]$$

- But the expected error from the committee is

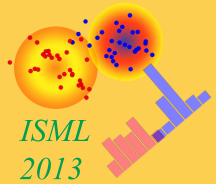
$$E_{\text{COM}} = \mathbb{E}_{\mathbf{x}}\left[\left\{\frac{1}{M}\sum_{m=1}^{M}y_m(\mathbf{x}) - h(\mathbf{x})\right\}^2\right] = \mathbb{E}_{\mathbf{x}}\left[\left\{\frac{1}{M}\sum_{m=1}^{M}\epsilon_m(\mathbf{x})\right\}^2\right]$$

- Assume errors have zero mean $\mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})\right] = 0$ and are uncorrelated $\mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})\epsilon_l(\mathbf{x})\right] = 0$ for $m \neq l$.
- 'Dramatic' error reduction by using a committee.

$$E_{\text{COM}} = \frac{1}{M}E_{\text{AV}}$$

- Unfortunately, not true. Why?

Christfried Webers
NICTA
The Australian National University



How big is the error?

- 'Dramatic' error reduction by using a committee.

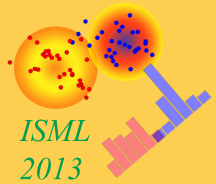
$$E_{\text{COM}} = \frac{1}{M} E_{\text{AV}}$$

- Unfortunately, not true. Why?
- The errors are typically highly correlated $\longrightarrow$ the reduction in overall error is generally small.
- However, can show that

$$E_{\text{COM}} \leq E_{\text{AV}}$$

- Expected committee error does not exceed the expected average error of the individual models.

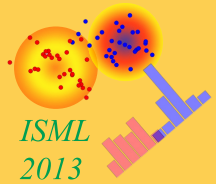
Christfried Webers
NICTA
The Australian National University



Learning from Learners

- In the committee method, the individual models did not know of each other. They do not learn from each other.
- How can models learn from each other?

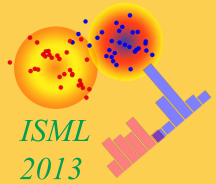
©2013 Christfried Webers
NICTA
The Australian National University



Learning from Learners

- In the committee method, the individual models did not know of each other. They do not learn from each other.
- How can models learn from each other?
- For example: Learn the models in sequence and use some information from learning one model in the learning process for the next model.
- How?

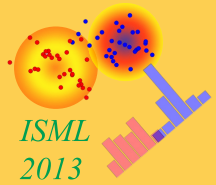
©2013
Christfried Webers
NICTA
The Australian National University



Learning from Learners

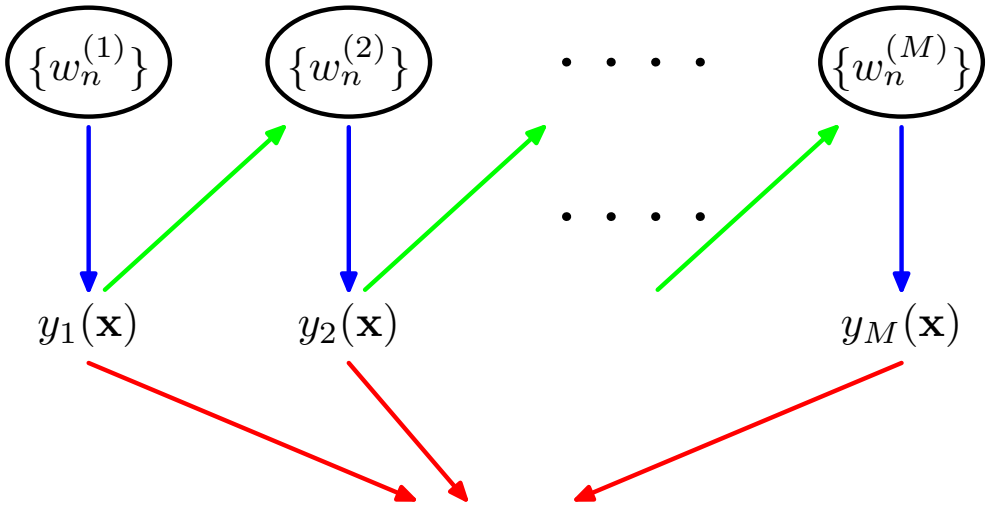
- In the committee method, the individual models did not know of each other. They do not learn from each other.
- How can models learn from each other?
- For example: Learn the models in sequence and use some information from learning one model in the learning process for the next model.
- How?
- For example: Tell the next model learning process which data points the current model did not learn well. These data points can then be focused on in the next learning process.

©2013 Christfried Webers
NICTA
The Australian National University



Boosting - AdaBoost (for classification)

- Assume 'base'classifiers (also called weak learners).
- AdaBoost (adaptive boosting): Points that are misclassified by one of the base classifiers are given greater weight when used to train the next classifier in the sequence.
- Combine the predictions of all the classifiers through a majority voting scheme.



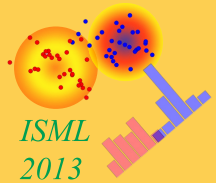
$$Y_M(\mathbf{x}) = \text{sign}\left(\sum_m^M \alpha_m y_m(\mathbf{x})\right)$$

Boosting - AdaBoost (for classification)

1. Initialise the data weighting coefficients $\{w_n\}$ to $w_n^{(1)} = 1/N$.
2. For each model $m = 1, \ldots, M$:
   1. Fit classifier $y_m(\mathbf{x})$.
   2. Compute the model weights α_m.
   3. Evaluate weighting coefficients $w_n^{(n+1)}$.
3. Make predictions using the final model

$$Y_M(\mathbf{x}) = \text{sign}\left(\sum_{m=1}^{M} \alpha_m y_m(\mathbf{x})\right).$$

Christfried Webers
NICTA
The Australian National University



Boosting - AdaBoost (for classification)

For each model $m = 1, \ldots, M$:

1. Fit classifier $y_m(\mathbf{x})$ by minimising the weighted error function

$$J_m = \sum_{n=1}^{N} w_n^{(n)} \mathbb{I}(y_m(\mathbf{x}) \neq t_n)$$

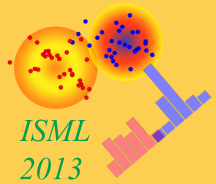
where $\mathbb{I}(y_m(\mathbf{x}) \neq t_n)$ is the indicator function which equals 1 if $y_m(\mathbf{x}) \neq t_n$ and is 0 otherwise.

2. Compute the model weights α_m by calculating

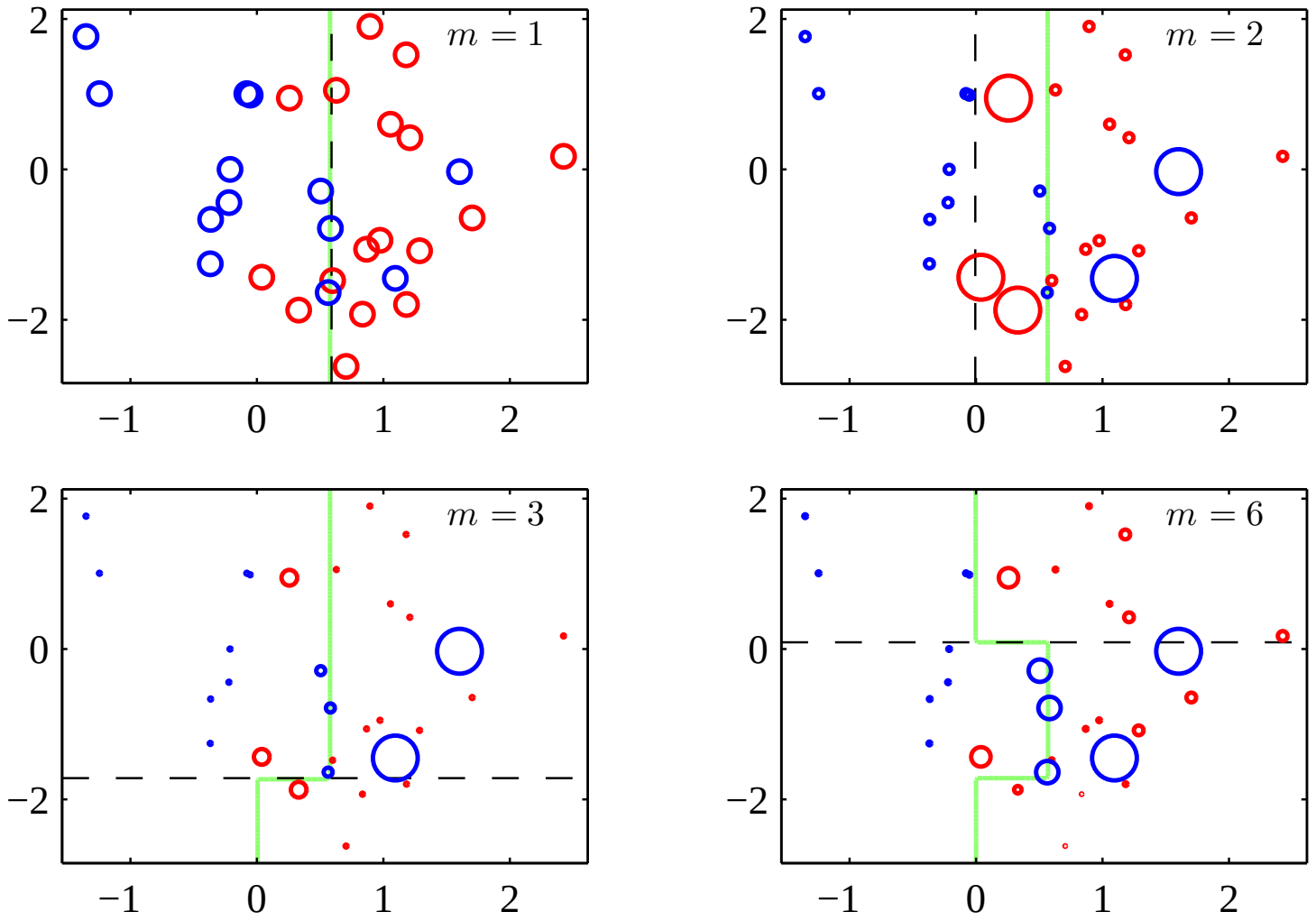
$$\epsilon_m = \frac{\sum_{n=1}^{N} w_n^{(n)} \mathbb{I}(y_m(\mathbf{x}) \neq t_n)}{\sum_{n=1}^{N} w_n^{(n)}} \quad \text{and} \quad \alpha_m = \ln \left\{ \frac{1 - \epsilon_m}{\epsilon_m} \right\}.$$

3. Evaluate weighting coefficients $w_n^{(n+1)}$ as

$$w_n^{(n+1)} = w_n^{(n)} \exp \{\alpha_m \mathbb{I}(y_m(\mathbf{x}) \neq t_n)\}.$$

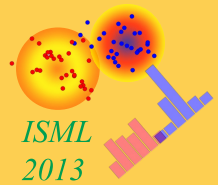


Boosting - AdaBoost (for classification)

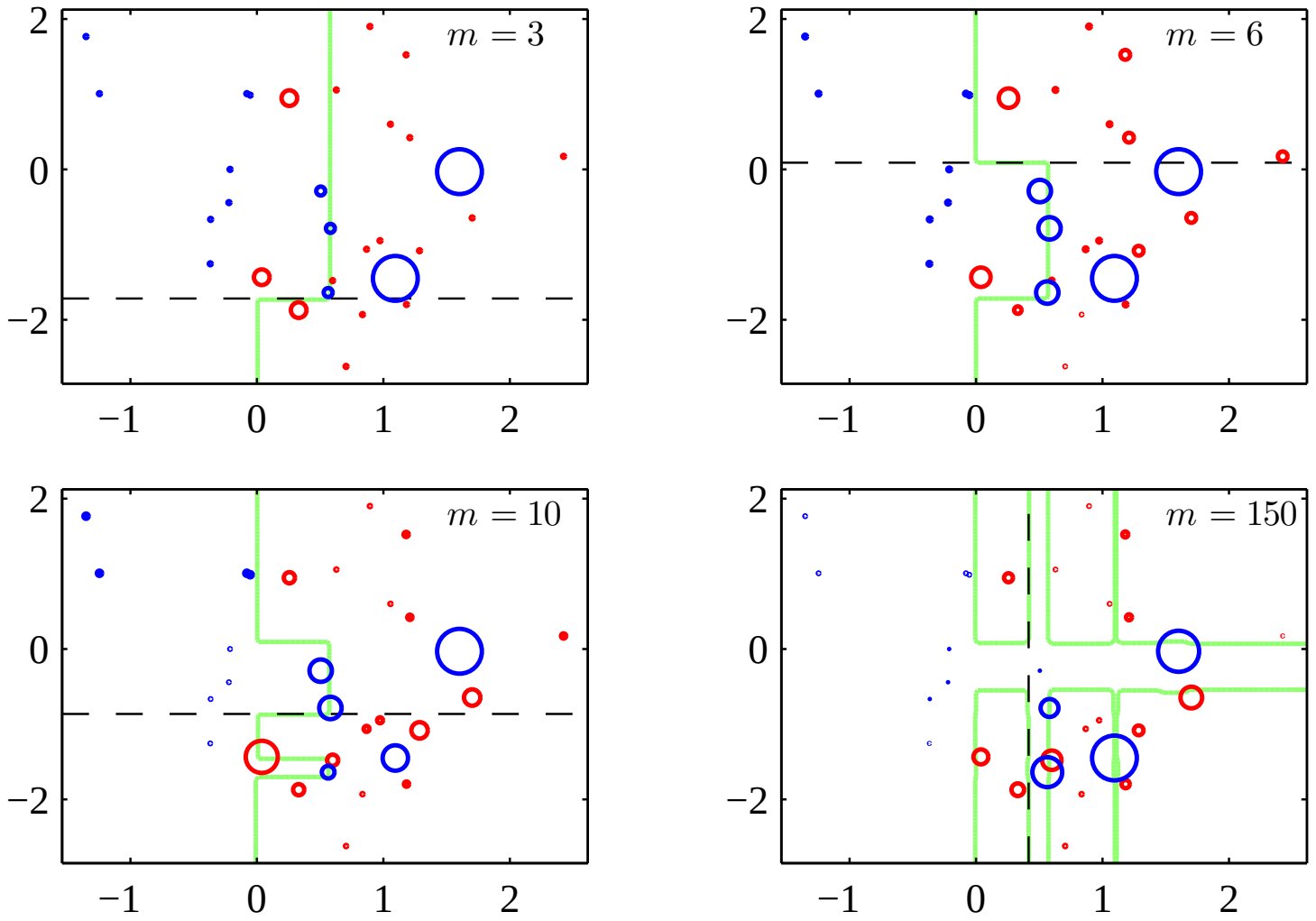


Decision boundaries: last learner (black), ensemble (green).
Data point weights of most recent learner (circle sizes).

©2013 Christfried Webers NICTA The Australian National University

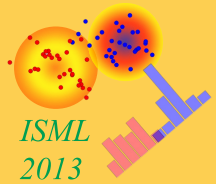


Boosting - AdaBoost (for classification)



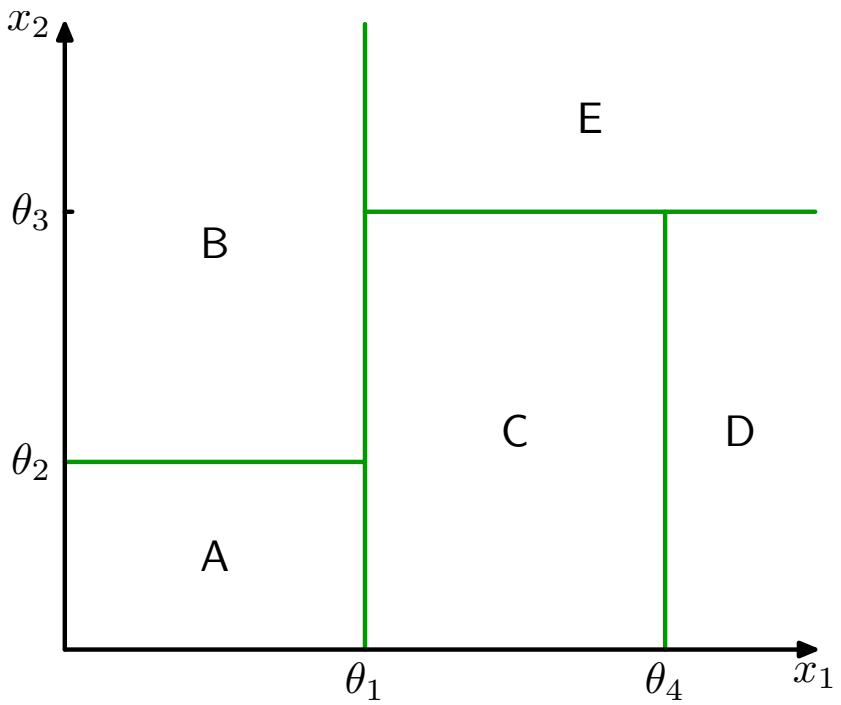
Decision boundaries: last learner (black), ensemble (green).
Data point weights of most recent learner (circle sizes).

©2013
Christfried Webers
NICTA
The Australian National University

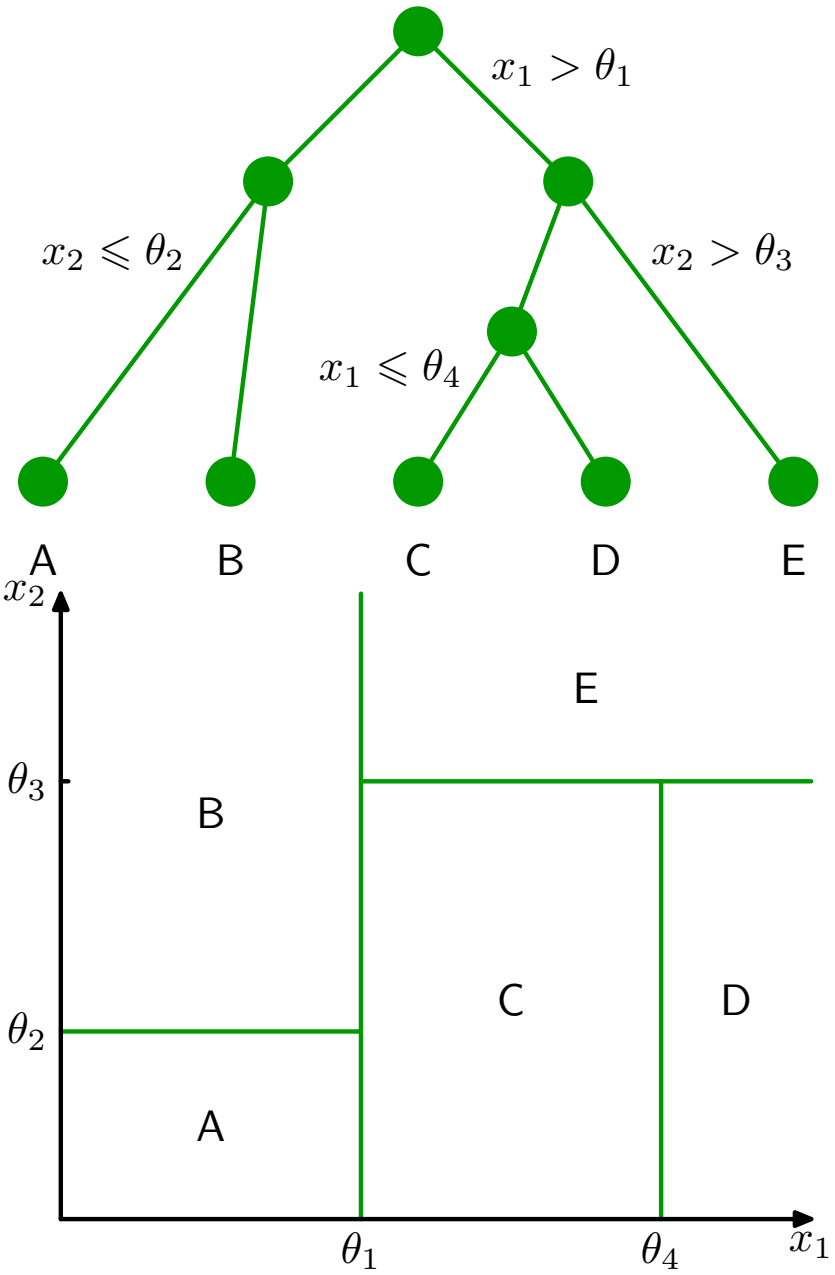


Tree-based Models

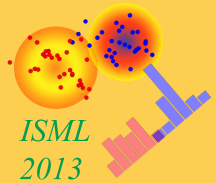
- Idea: Partition the input space into cuboid regions (edges aligned with the axes).
- Assign a model to each of the regions.
- Can be viewed as model combination where only one model is responsible for making predictions at each point in input space.
- Which model is responsible for the prediction given a new data point? Traverse the binary tree.



Tree-based Models

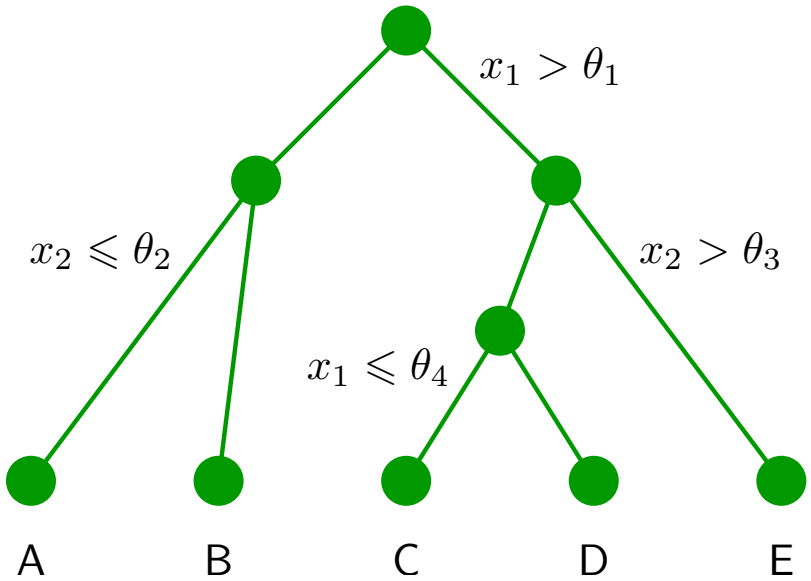


©2013
Christfried Webers
NICTA
The Australian National University



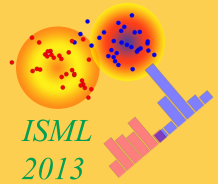
Tree-based Models

- Advantages
  - Decision trees are readily interpretable by humans. (e.g. medical analysis). (White box model.)
  - Allow easy incorporation of expert knowledge.
  - Can be combined with other decision techniques.
- Disadvantages :
  - Why should decision boundaries be parallel to axes?
- How to learn the tree?



Note: Decision trees are NOT probabilistic graphical models!

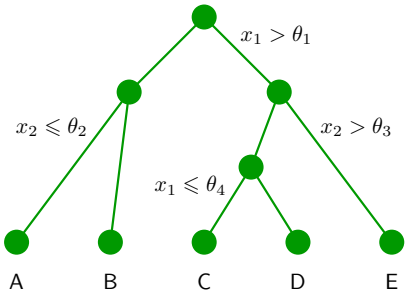
©2013
Christfried Webers
NICTA
The Australian National University



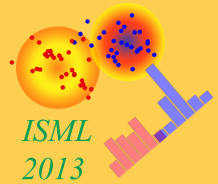
Classification and Regression Trees (CART)

- How to learn the structure of the tree?
- Given training data $\{\mathbf{x}_1, \ldots, \mathbf{x}_N$ with each $\mathbf{x}_n \in \mathbb{R}^D$, and training targets $\{t_1, \ldots, t_N\}$.
- Goal: Predict the target t for a new data point $\mathbf{x}$.
- If the partitioning of the input space is given, and we minimise the sum-of-squares error function, then the optimal prediction value in each region is the average of the training targets t_n in this regions.
- Training: For each node in the tree, we have to choose
  - an input variable to split on, and
  - a corresponding threshhold

  such that the sum-of-squares error is minimised.
- Combinatorial explosion! Infeasible!

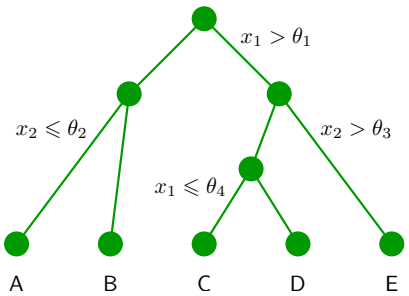


©2013
Christfried Webers
NICTA
The Australian National University

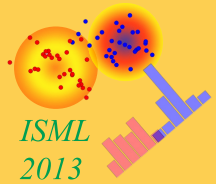


Classification and Regression Trees (CART)

- Use greedy algorithm.
- Start with a single root node.
- For each new training data point, consider only regions which could be potentially split.
- Choose the split and threshold which minimises the sum-of-squares error.
- Local optimisation problem.
- When to stop?

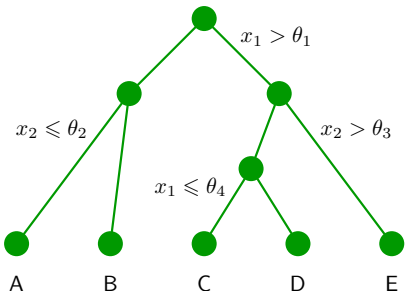


©2013 Christfried Webers
NICTA
The Australian National University

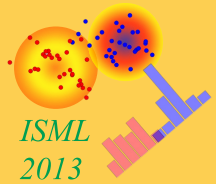


Classification and Regression Trees (CART)

- Stop when there is no more reduction in residual error.
- However : empirically found that although none of the current splits reduces the error anymore, later splits can produce much smaller errors.
- Therefore: Grow a large tree (stopping criterion based only on the number of points associated with each leaf node). Then prune the tree.



©2013
Christfried Webers
NICTA
The Australian National University



Classification and Regression Trees (CART)

- Start pruning with tree T_0.
- Prune into tree $T \subset T_0$ by collapsing internal nodes.
- Suppose the leaf nodes of T are indexed by $\tau = 1, \ldots, |T|$.
- Optimal prediction for region $\mathcal{R}$ is given by

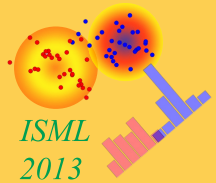
$$y_\tau = \frac{1}{N_\tau} \sum_{\mathbf{x} \in \mathcal{R}} t_n.$$

- Corresponding contribution to the residual sum-of-squares

$$Q_\tau(T) = \sum_{\mathbf{x}_n \in \mathcal{R}} \{t_n - y_\tau\}^2.$$

- Pruning criterion (regulariser λ from cross-validation)

$$C(T) = \sum_{\tau=1}^{|T|} Q_\tau(T) + \lambda |T|.$$



Classification and Regression Trees (CART)

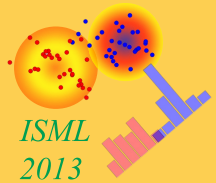
- Similar growing and pruning for classification problems.
- Replace the sum-of-squares error with an appropriate measure of performance.
- Define $p_{\tau k}$ to be the proportion of data points in region $\mathcal{R}_\tau$ assigned to class k , where $k = 1, \ldots, K$.
- For growing the tree use the differentiable
  - Cross-Entropy

$$Q_\tau(T) = \sum_{k=1}^{K} p_{\tau k} \ln p_{\tau k}$$

    or
  - Gini Index

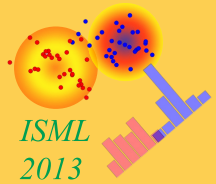
$$Q_\tau(T) = \sum_{k=1}^{K} p_{\tau k}(1 - p_{\tau k}).$$

- Both vanish for $p_{\tau k} = 0$ and $p_{\tau k} = 1$, and have a maximum at $p_{\tau k} = 0.5$.
- For pruning the tree, use the misclassification rate.



Classification and Regression Trees (CART)

- Human interpretability seen as major strength.
- However, structure is can be very sensitive to the details of the data set.(Small changes lead to very different splits. Order of the data presentation matters.)
- Splits are aligned with the axes. What happens if the decision boundary between two classes runs at 45 degrees to the axes? (Large number of splits necessary.)
- Each region in input space is associated with exactly one leaf. (Hard splits)
- Especially problematic for regression, where one often tries to model smooth functions. The CART only provides piecewise constant predictions with discontinuities at the split boundaries.



Conditional Mixture Models

Overcoming the restrictions of Classification and Regression Trees (and loosing interpretability on the way :-(

(A) Hierarchical Mixture of Experts : A fully probabilistc tree-based model.

- Allow soft, probabilistic splits that can be functions of all the input variables, not just one at a time.
- Give leaf nodes a probabilistic interpretation.

(B) Hierarchical Mixture of Experts :

- Start with mixture of Gaussians (or other unconditional distributions).
- Replace the unconditional distribution by a conditional distribution conditioned on the input data.
- Create a Mixture of Experts Model by allowing the mixing coefficients to depend on the input data $\pi_k \rightarrow \pi_k(\mathbf{x})$.
- Allow each component in the mixture model to be itself a mixture model. (Recurse definition $\rightarrow$ tree.)

©2013 Christfried Webers
NICTA
The Australian National University