Calculating the VC-Dimension of Decision Trees

Özlem Aslan[1] Olcay Taner Yıldız[2] Ethem Alpaydın[1]

[1]Department of Computer Engineering
Boğaziçi University

[2]Department of Computer Engineering
Işık University

24th International Symposium on Computer and Information Sciences, 2009

Outline

1. Introduction
   - Model Complexity
   - VC Dimension
2. Proposed Method
   - Exhaustive Search Algorithm
   - Estimating VC-Dimension By Regression
   - Complexity Control Using VC-Dimension Estimates
3. Conclusion

Outline

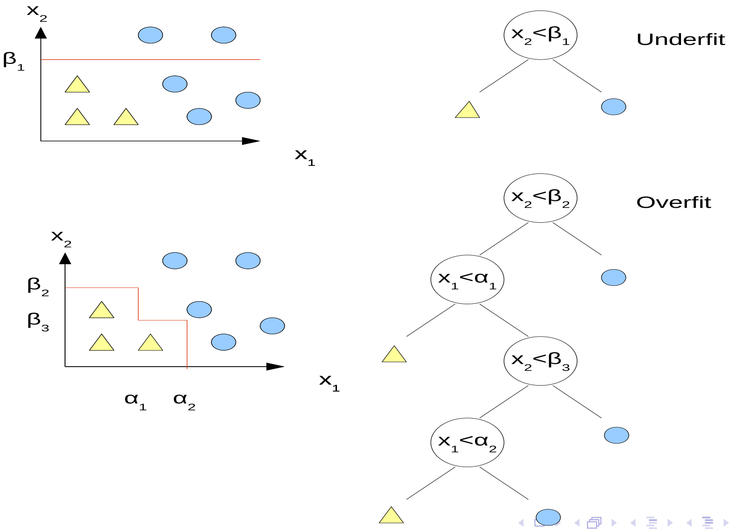
1. Introduction
   - Model Complexity
   - VC Dimension
2. Proposed Method
   - Exhaustive Search Algorithm
   - Estimating VC-Dimension By Regression
   - Complexity Control Using VC-Dimension Estimates
3. Conclusion

Outline

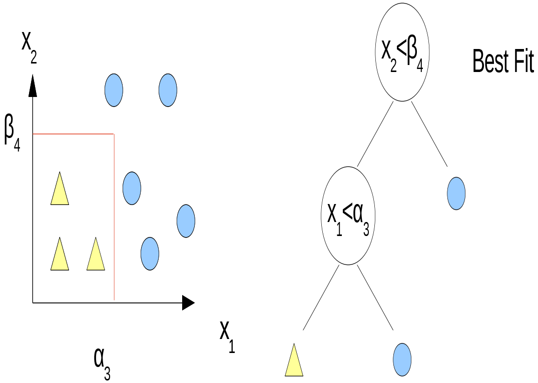
Outline

1. Introduction
   - Model Complexity
   - VC Dimension
2. Proposed Method
   - Exhaustive Search Algorithm
   - Estimating VC-Dimension By Regression
   - Complexity Control Using VC-Dimension Estimates
3. Conclusion

Underfit vs Overfit



Best Model



Structural Risk Minimization

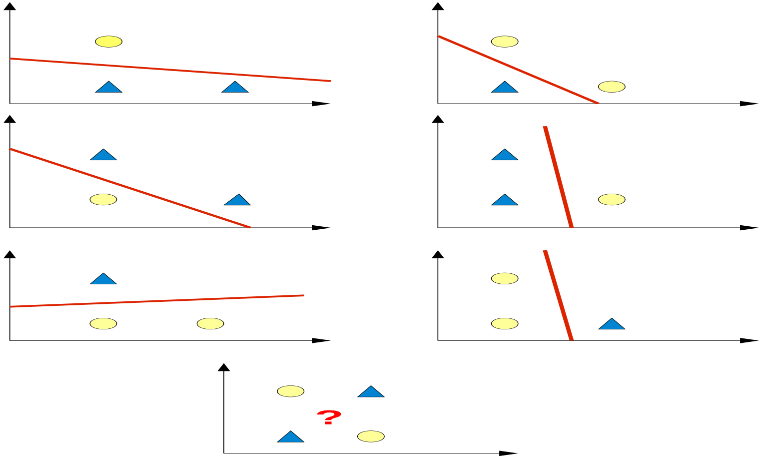
$$E_g = E_t + \frac{\epsilon}{2}\left(1 + \sqrt{1 + \frac{4E_t}{\epsilon}}\right) \quad (1)$$

$$\epsilon = a_1 \frac{V[\log(a_2 N/V) + 1] - \log(\nu)}{N} \quad (2)$$

(Vapnik95)

Variable	Definition
E_t	training error
V	VC dimension of the model
ν	confidence level
a_1 and a_2	empirically fitted constants
N	sample size

VC Dimension



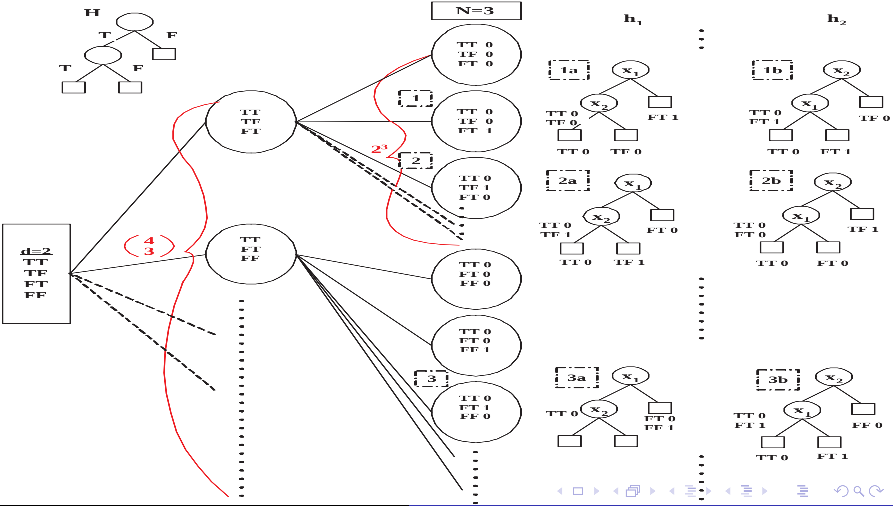
Outline

1. Introduction
   - Model Complexity
   - VC Dimension
2. Proposed Method
   - Exhaustive Search Algorithm
   - Estimating VC-Dimension By Regression
   - Complexity Control Using VC-Dimension Estimates
3. Conclusion

Procedure

- An exhaustive search algorithm to calculate the exact VC-dimensions.
- Fit a regressor so that we can estimate the VC-dimension of any tree.
- VC-dimension estimates in pruning to validate that they are indeed good estimates.

Illustration



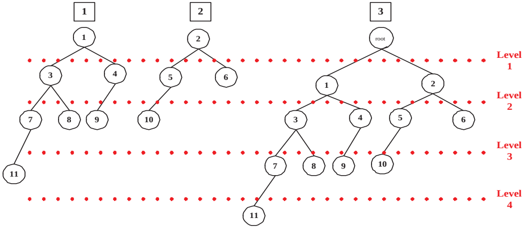
Computational Complexity

$$\sum_{N=1}^{V} \binom{2^d}{N} 2^N |H|$$

- The full tree with depth 4 and for 4 input features requires 2 days to complete on a quad-core computer
- Depth 5 and for 5 input features will require approximately 10^{13} days.
- We can run the exhaustive search algorithm only on few H and on cases with small d and $|H|$.

Experimental Setup

- We thoroughly searched decision trees with depth up to four.
- We use the fact that two isomorphic trees have the same VC dimension.



Regression Model

154 training instances

$$V = 0.7152 + 0.6775 V_l + 0.6775 V_r - 0.6600 \log d + 1.2135 \log M$$

R^2 is 0.9487.

Experimental Setup

- CVprune
- SRMprune
- NOprune

Experimental Setup

Functions:

$$
\begin{aligned}
F_1 &= x_0x_1 + x_0x_2 + x_1x_2 \\
F_2 &= x_0x_1 + x_0x_2 + x_0x_3 + x_1x_2 + x_1x_3 + x_2x_3 \\
F_3 &= x_0x_1' + x_0'x_1
\end{aligned}
$$

- The number of input features $d = 8$ and $d = 12$
- Five different noise levels $\rho = 0.0, 0.01, 0.05, 0.1$, and 0.2.
- Four different sample size percentage $S = 10, 25, 50, 100$.

Complexity Control Results

$d = 12$, $\rho = 0.0$, and $S = 100$

Function	Error Rate			Number of Nodes		
	NOprune	CVprune	SRMprune	NOprune	CVprune	SRMprune
F_1	0.0± 0.0	0.0± 0.0	0.0± 0.0	5.0± 0.0	5.0± 0.0	5.0± 0.0
F_2	0.0± 0.0	0.0± 0.0	0.0± 0.0	9.0± 0.0	9.0± 0.0	9.0± 0.0
F_3	3.9± 2.8	8.5± 7.0	3.9± 2.8	177.6±115.8	83.3±59.5	174.9±115.6

Complexity Control Results

$\rho = 0.2$, $S = 100$, and $F = F_2$

d	Error Rate			Number of Nodes		
	NO prune	CV prune	SRM prune	NO prune	CV prune	SRM prune
8	38.1± 4.1	37.8± 5.3	35.3± 2.7	57.5± 6.3	3.8± 3.3	12.8± 7.9
12	35.5± 1.2	28.2± 3.0	21.0± 0.6	869.2± 15.1	4.2± 1.5	9.0± 0.0

Complexity Control Results

$\rho = 0.2$, $S = 100$, and $F = F_2$

d	Error Rate			Number of Nodes		
	NO prune	CV prune	SRM prune	NO prune	CV prune	SRM prune
8	38.1± 4.1	37.8± 5.3	35.3± 2.7	57.5± 6.3	3.8± 3.3	12.8± 7.9
12	35.5± 1.2	28.2± 3.0	21.0± 0.6	869.2± 15.1	4.2± 1.5	9.0± 0.0

Complexity Control Results

$d = 12$, $S = 50$, and $F = F_1$

ρ	Error Rate			Number of Nodes		
	NO prune	CV prune	SRM prune	NO prune	CV prune	SRM prune
0.0	0.0± 0.0	0.0± 0.0	0.0± 0.0	5.0± 0.0	5.0± 0.0	5.0± 0.0
0.01	3.6± 0.5	1.5± 0.3	1.5± 0.3	62.5± 11.0	5.0± 0.0	5.0± 0.0
0.05	12.2± 0.8	5.0± 0.5	5.0± 0.5	167.0± 10.6	5.0± 0.0	5.0± 0.0
0.1	21.7± 0.9	12.8± 4.7	10.6± 0.2	283.2± 13.0	5.2± 2.2	5.0± 0.0
0.2	35.7± 1.4	29.3± 5.4	20.6± 0.9	419.5± 13.7	2.6± 1.6	5.0± 0.0

Complexity Control Results

$d = 8$, $\rho = 0.05$, and $F = F_3$

S	Error Rate			Number of Nodes		
	NO prune	CV prune	SRM prune	NO prune	CV prune	SRM prune
100	19.0± 5.9	25.3± 14.9	15.8± 8.6	36.3± 10.6	8.4± 5.1	23.8± 18.9
50	23.7± 14.7	28.9± 17.2	23.4± 14.6	19.4± 9.1	4.4± 3.3	18.1± 9.7
25	27.0± 11.7	37.4± 15.7	27.0± 11.7	9.4± 4.1	1.3± 1.7	9.4± 4.1
10	41.7± 17.1	45.0± 17.2	41.7± 17.1	5.3± 0.9	0.9± 1.4	5.3± 0.9

Outline

1. Introduction
   - Model Complexity
   - VC Dimension
2. Proposed Method
   - Exhaustive Search Algorithm
   - Estimating VC-Dimension By Regression
   - Complexity Control Using VC-Dimension Estimates
3. Conclusion

Conclusion

- VC-dimension calculation by exhaustive search
- Estimation of VC-dimension via regression
- VC-dimension used in SRM based model selection
- Find trees that are as accurate as in CV pruning

Future Work

- The approach can easily by extended to univariate decision trees with discrete and/or continuous features.
- Extension to K-class

Extension

Discrete features with 3 values:

$$\begin{aligned} V &= -3.0014 + 0.5838V_1 + 0.5838V_2 + 0.5838V_3 \\ &+ 2.5312 \log d + 1.9064 \log M \end{aligned}$$

R^2 is 0.91.
4 values:

$$\begin{aligned} V &= -1.6294 + 0.5560V_1 + 0.5560V_2 + 0.5560V_3 + 0.5560V_4 \\ &+ 3.9830 \log d - 0.4073 \log M \end{aligned}$$

R^2 is 0.861.

Extension

Discrete features with 5 values:

$$\begin{aligned} V &= 14.4549 + 0.3924V_1 + 0.3924V_2 + 0.3924V_3 \\ &+ 0.3924V_4 + 0.3924V_5 - 4.7687\log d - 1.3857\log M \end{aligned}$$

R^2 is 0.782.