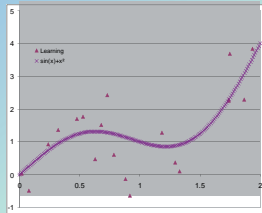


Knowledge Extraction from Data

Learning Data

- f is the unknown target function to discover
- A learning sample is randomly chosen
- This learning base can be noisy



Illustrative example:

- $f(x) = \sin(x) + x^2$
- 20 points randomly chosen are used for training
- A normal noise is added ($\sigma=20\%$ of output variation range)

Learning Strategy

- Apprentice \hat{f} : input variables, model structure (polynom degree, number of rules, number of neurons...)
- Error estimation
- Learning algorithm
- Regulation methods:
 - Compactness control (Δ dimensionality, Δ complexity of the apprentice)
 - Smoothness control (random noise injection on inputs)

Model Generator

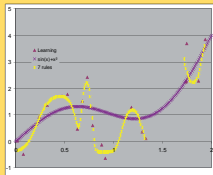
$$\hat{f}(x)$$

- Aim: induce generic knowledge from the learning database
- Model \hat{f} must be as close to f as possible
- Model prediction error on the whole population = Actual risk $R(\hat{f})$

Commonly used estimators

Empirical risk ($R_{emp}(\hat{f})$)

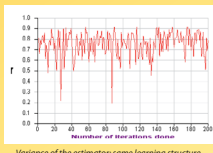
- Empirical risk = prediction error computed on the learning data
- An accurate model has a low $R_{emp}(\hat{f})$, but can have a poor generalization ability ($R(\hat{f})$ high): *overfitting phenomenon*
- Overfitting = learning by heart
- Multiple causes: model structure too complex, too few learning points, bad distribution of the learning sample, too noisy data...



→ $R_{emp}(\hat{f})$: bad estimator of $R(\hat{f})$

Validation Set (VS)

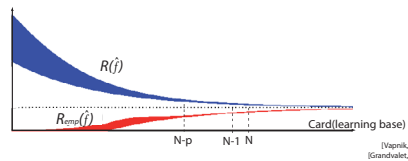
- Most widely used: "2/3 - 1/3 split"
- Drawbacks:
 - 1/3 data not used for learning
 - sensitivity to choice of the validation set: risk of overfitting
 - unbiased estimator of $R_{N,p}(\hat{f})$, but very high variance



→ VS: bad estimator of $R(\hat{f})$ except when a lot of data is available

$R(\hat{f})$: usually impossible to compute but can be estimated

Relation between empirical risk and actual risk



Bias of estimators

- An unbiased estimator of $R_{N,p}(\hat{f})$ is a pessimistic estimator of $R(\hat{f})$
- Bias is not dependant on the structure of \hat{f} : possibility of ranking models generated with the same learning data [Efron et al, 1998]

Variance of estimators

- Low N: high variance
- VS > LOO > MC or Exhaustive CV (p small) > MC or Exhaustive CV (p large)

$R_{emp}(\hat{f})$ or VS: bad estimators of modeling performance

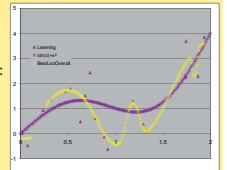
Difficulty to select the proper CV estimator: strong variance if p is low, strong bias if p is high

Cross validation (CV)

- Whole database used for the model generation
- Same learning strategy applied on a large number of learning/validation splits
- $R(\hat{f})$ estimated on the scatter of validation points of all test models
- Computationally intensive: model generation for each of the $\approx 50-300$ iterations
- Insensitivity to an arbitrary choice of learning/validation bases

Leave one out (LOO)

- Database: N samples
- N-1 learning points, 1 validation point: N iterations
- Quick or prohibitive: depending on N
- Unbiased estimator of $R_{N,p}(\hat{f})$, high variance if N is low

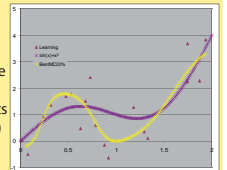


Exhaustive CV

- N-p learning points, p validation points
- Every combination is tested: C_N^p iterations
- Unbiased estimator of $R_{N,p}(\hat{f})$, low variance but usually computationally too expensive

Monte Carlo (MC)

- Statistical approximation of Exhaustive CV
- N-p learning points, p validation points
- M independant iterations ($\approx [50-300]$) depending on process convergence
- Unbiased estimator of $R_{N,p}(\hat{f})$, low variance [Shao, 1993]



xtractis® modeling heuristics

1 - Models generation

- Generate lots of models with all available data
- Use different proprietary learning strategies

2 - Actual risk estimation

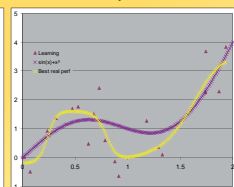
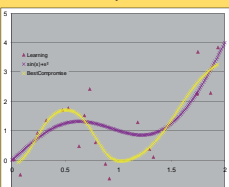
- Perform several CV methods on all generated models
- LOO: almost unbiased but high variance
- MC with $p = 10\% N$, $p = 20\% N$ and $p = 30\% N$: pessimistic but low variance

3 - Models ranking

- Select models with the best compromise between all CV estimators and compact enough

Results

- Best compromise on all CV estimators: $R(\hat{f}) = 0.874$
- Best model generated by xtractis®: $R(\hat{f}) = 0.897$



Name	Nb. of var. rules	Error				Remarque	
		Correlation	Mean	Hamming	Max.		
MostPreciseModel_4R	1	4	0.876	0.000	10.00%	25.30%	Generation120
			0.720	0.01	16.15%	47.24%	300 x MC 10%, all points
			0.669	0.035	16.72%	59.88%	300 x MC 20%, all points
			0.653	-0.01	17.44%	71.62%	300 x MC 30%, all points
			0.708	0.019	16.41%	36.22%	1 x LOO, all points
			0.897	-0.18	8.76%	22.13%	Actual risk
BestCompromise_Overall_3r	1	3	0.859	0.020	10.67%	28.53%	Generation106
			0.753	0.078	14.16%	45.30%	300 x MC 10%, all points
			0.679	0.015	16.48%	55.19%	300 x MC 20%, all points
			0.628	0.003	17.39%	72.28%	300 x MC 30%, all points
			0.720	0.065	14.95%	38.25%	1 x LOO, all points
			0.874	-0.14	8.76%	22.48%	Actual risk
BestAcc_Overall_7r	1	7	0.964	-0.01	5.05%	22.20%	Generation610
			0.608	-0.03	18.84%	62.34%	300 x MC 20%, all points
			0.583	-0.10	19.90%	39.93%	1 x LOO, all points
			0.744	-0.23	12.27%	34.02%	Actual risk

Powered by xtractis® an Intellitech [Intelligent Technologies] software. Copyright © Intellitech [Intelligent Technologies] 2002-2007. All rights reserved.

Benefits of xtractis®

- CV estimators avoid overfitting
- CV estimators give a correct estimation of the actual performance of the model
- Relying on several estimators reduces the effects of variance of one single estimator
- Contrary to VS methods, all available data is used for model generation: more robust models
- Models ranking not perturbed by the bias of estimators
- Thanks to its efficient learning strategies, xtractis® is able to:
 - handle very complex problems (thousands of variables, incomplete database, fuzzy/noisy data)
 - generate robust models (despite needs of intensive computational capacity)
 - select top-models close to the best possible generated model (highest robustness on the whole population)