Data Mining with Sparse and Simplified Interaction Selection

Gerald Fahner

International Computer Science Institute
1947 Center Street - Suite 600
Berkeley, CA 94704, U.S.A.
fahner@icsi.berkeley.edu

Abstract

We introduce a novel, greatly simplified classifier for binarized data. The model contains a sparse, "digital" hidden layer of **Parity** interactions, followed by a sigmoidal output node. We propose priors for the cases: a) input space obeys a metrics, b) inputs encode discrete attributes. Stochastic search for the hidden layer allows capacity and smoothness of the approximation to be controlled by two complexity parameters. Aggregation of classifiers improves predictions. Interpretable results are obtained in some cases. We point out the impact of our model on real-time systems, suitability for sampling and aggregation techniques, and possible contributions to nonstandard learning devices.

Introduction

For huge databases with many variables that interact in complex ways, careful human selection of a feature space can become unmanageable (Elder & Pregibon 1996). (Vapnik 1995) emphasizes a complementary approach to data modelling, namely to approximate the unknown dependency by a "smart" linear combination of "weak features". Any reasonable feature space may be chosen; the predictive power arises entirely from capacity control. Data mining tools operating accordingly must identify the relevant features or interactions between variables. Here we focus on the issues

- **Sparseness**: how many interaction terms should a reasonable model include?
- **Preference**: can *a priori* preferences be assigned within a group of models of same size?
- **Simplicity**: are there reasonable feature sets that are particularly simple to compute?

The first issue is dealt with by applying Vapnik's Structural Risk Minimization. For the model family discussed in this paper, a nested set of models of increasing size is created and the optimum compromise between low training error and tight worst case bound for the test error is determined.

We tackle the second issue by assigning preferences to individual input features, speeding up the search process and improving performance over the worst case bounds. Interactions that seem natural are given high probability. Less obvious interactions are also explored to allow discovery of unexpected dependencies. Without domain knowledge, general priors are used that punish rapidly oscillating or high order interactions.

The third issue gains importance for mining huge amounts of data, for recent computationally intensive methods that sample in model space, for real-time data analysis, and for possible use within future optical or biomolecular hardware. Here we present a model with greatly simplified interaction features, as compared to "classic" neural networks.

In the following, we discuss heuristic methods for the identification of models for binary data. In order to discover knowledge we may estimate joint probabilities or perform soft classification of binary vectors $\underline{x} \in \{-1, 1\}^N$.

Sparse Multinomial Logistic Model

We model a stochastic dependency between $\underline{x}$ and a two-valued outcome variable $y \in \{0, 1\}$. The regression $p(y = 1 \mid \underline{x})$ is estimated from a training database $\mathcal{T}$ of labeled examples $(\underline{x}^i; y^i)_{i=1}^{\#\mathcal{T}}$. For approximation of the regression, we use the logistic model

$$\hat{y} = \frac{1}{1 + \exp\{-f(\underline{x}, \underline{\theta})\}} \tag{1}$$

with $f : \{-1, 1\}^N \rightarrow I\!R$:

$$\begin{aligned} f(\underline{x}, \underline{\theta}) &= \theta_0 + \theta_1 x_1 + \dots + \theta_N x_N \\ &+ \theta_{1,2} x_1 x_2 + \dots + \theta_{N-1,N} x_{N-1} x_N \\ &+ \theta_{1,2,3} x_1 x_2 x_3 + \dots \\ &+ \dots \\ &+ \theta_{1,2,\dots,N} x_1 x_2 \dots x_N \end{aligned} \tag{2}$$

subject to the constraint that

$$\underline{\theta} = \begin{pmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_{1,2} \\ \vdots \\ \theta_{1,2,\dots,N} \end{pmatrix}$$

is a *sparse* vector of reals with an *a priori* fixed number of non-vanishing components. Fixed size models are fitted to the data by maximizing the log likelihood:

$$log\ lh = \sum_{i=1}^{\#\mathcal{T}} y^i\ log\, \hat{y}(\underline{x}^i, \underline{\theta}) + (1 - y^i)\ log\, (1 - \hat{y}(\underline{x}^i, \underline{\theta}))$$

Maximization is over all models of given size. This hard combinatorial problem can in general only be solved approximately.

The unconstrained expression (2) is known as "Walsh expansion" (Walsh 1923). The additive and higher order interactions form an orthogonal base (of dimension 2^N) for real valued functions over binary strings. Model (1) can thus approximate any regression function. In contrast, the sparse version has finite capacity. By enlarging the number of interactions sufficiently, any dichotomy over the input space can be approximated. Determination of a reasonable model size is crucial for obtaining low generalization error.

The second and higher order interaction terms can be considered as hidden nodes in a sparse network. Each node basically evaluates the Parity predicate (in the $(0, 1)$ representation) over some selected submask of input bits, which can be done in parallel. Heuristic supervised learning algorithms were proposed for problems of unknown order (Fahner and Eckmiller 1994).

Model Identification by Pruning and Replacement

The algorithm presented in the box below determines model size, selects a set of interaction terms, and simultaneously computes respective coefficients $\theta_{i,k,..}$.

```
1) chose model size within interval $[\frac{\#T}{100}, \#T]$
2) chose prior distribution $p(complexity; \mu)$
   for individual interactions
3) initialize model with tentative inter-
   actions drawn according to $p(complexity; \mu)$
4) maximize $log\ lh\ (\underline{\theta}\,|\,T)$ and obtain weight
   vector $\underline{\theta}^*$
5) prune "brittle" interactions
6) install novel tentative interactions
   replacing the pruned ones
7) back to 4) until stopping criterium is
   met; output final sparse model
```

The maximization **4)** is over a fixed model structure, and the likelihood function possesses a single maximum. Fig.1 depicts the inner "Pruning and Replacement" loop (**4**) to **7)**). The stopping criterion of the algorithm varies with applications. For any preselected size and prior distribution, the algorithm outputs a sparse multinomial. Search for the best model (minimum validation error) is over the two-dimensional parameter plane spanned by model size and the single parameter μ, which determines the prior distribution for feature complexity (see explanation to Fig.2). We sample an ensemble of models from a reasonable region of the plane. We distribute the training of individual models over a network of workstations, which requires no communication.

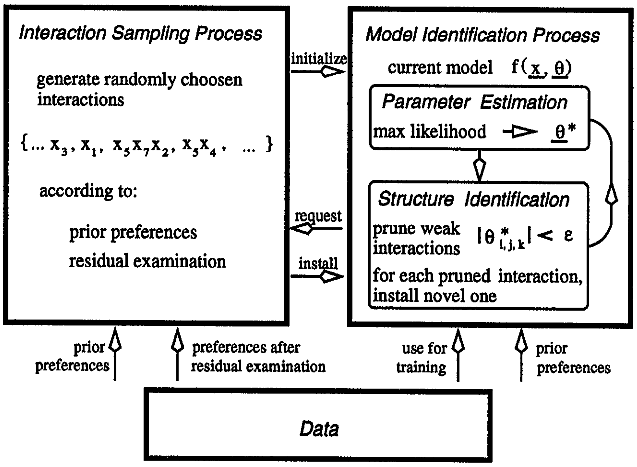
Interaction Sampling Process
generate randomly choosen interactions
{... x_3, x_1, $x_5x_7x_2$, x_5x_4, ... }
according to:
prior preferences
residual examination
initialize
request
install
Model Identification Process
current model f(x, θ)
Parameter Estimation
max likelihood → θ*
Structure Identification
prune weak interactions |θ*_{i,j,k}| < ε
for each pruned interaction, install novel one
prior preferences
preferences after residual examination
use for training
prior preferences
Data

Figure 1: Stochastically driven interaction selection: Right module modifies the current model by iterative application of parameter estimation and structure modification. In the simplest case, an interaction is pruned if its weight is below some threshold ϵ. More advanced pruning mechanisms include prior preferences or statistical significance tests to reveal "brittle" interactions. For any interaction pruned, a request for a novel term is sent to the left module.

Left module generates several candidate terms according to prior preferences, and ranks them according to their correlation with residual misclassification error. Greedy selection installs the term with the highest correlation.

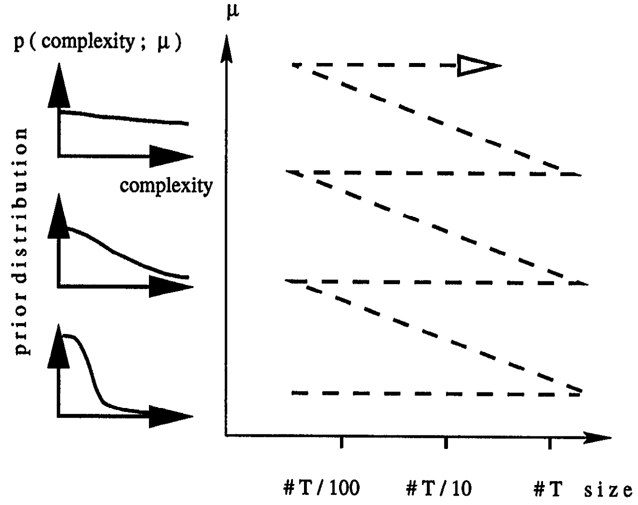

Figure 2: Capacity control plane: with increasing size, reduction of training error is possible at the expense of overfitting. μ parametrizes the form of a generic complexity prior distribution (as indicated qualitatively) for individual features. With increasing μ, complex interactions are more likely to be included in the model, thereby increasing the effective model space.

Complexity Measures for Interactions

For two types of input space semantics: a) binary representation of metrical input data, b) binary encodings of discrete attribute values, we propose respective complexity measures for individual interactions:

- **a) zero crossings**: maximum number of sign flips along straight line through input space
- **b) order**: number of multiplicative factors included in the interaction

Fig.3 illustrates case a) for a two-dimensional rectangular input space. Both continuous $\vec{a}$ and $\vec{b}$ axes are uniformly discretized into 4 intervals. For each dimension, the intervals are encoded by increasing binary numbers (− stands for 0, + for 1), preserving the order relation of the aligned intervals. Each box in the rectangular region is encoded as a 4-tuple $x_1x_2x_3x_4$ formed by the concatenation of the discretized and binarized coordinate values of the boxes. The given example generalizes to higher dimensions, to arbitrary binary resolutions individually chosen for each dimension and to nonuniform parcellings.

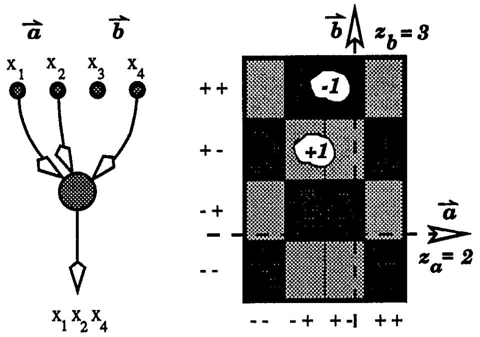

Figure 3: Behavior of the interaction term $x_1x_2x_4$ in two dimensions. The interaction term oscillates between −1's and 1's, undergoing zero crossings at some box borders. Along the two dashed arrows, the number of zero crossings z_a and z_b is counted separately for each of the coordinate axes. The maximum achievable number of zero crossings for an arbitrary direction linesweep is given by $z_a + z_b = 5$.

Simulation Results

We illustrate the working of the model for the **2-spirals** problem. In the original formulation (Lang and Witbrok 1988), the classifier has to separate two continuous point sets in $\mathbb{R}^2$ that belong to one or the other of intertwined spirals. The problem is formulated for binarized inputs as follows: each point in the plane is represented by some bitstring B_aB_b which is the concatenation of the truncated binary expansions for the points a- and b-coordinates. We chose 7 bit resolution for each coordinate, which is much more than required to distinguish between any two training examples. We use the "zero crossings" prior. A particular choice of coordinate axes breaks shift invariance and isotropy of the original problem, due to the invariance group properties of the Walsh functions. In order to restore the effect of broken symmetries, we apply the binarization for a transformation set (TS) of several randomly shifted and rotated (around coordinate center $(0,0)$, not around center of the spirals, since we assume no *a priori* knowledge) versions of the original input vectors. For each coordinate system, a separate model is trained on 335 examples per class. Size and prior complexity are constant over TS. The approximation over the discretized $[0,1)^2$ is computed by averaging over the estimates of all members of TS. Training is stopped as soon as all training examples are correctly classified or no further error reduction is achieved within a reasonable number of iterations. Fig.4 shows a result with adequately designed models.

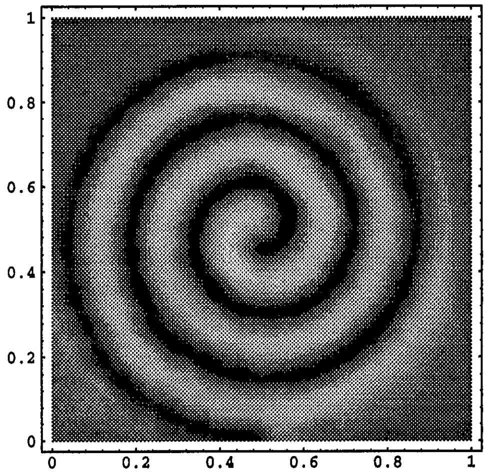

Figure 4: Approximation for models with size = 50, and with a prior that encourages a moderate number of zero crossings. TS contains 679 models, contributing to the apparent smoothness of the approximation.

The best models mainly use the 4 most significant bits of the binary encoded input vectors, and hardly contain interactions with more than 15 zero crossings. In contrast, models with a more flat zero crossings prior exhibit rich connectivity also to the least significant bits, thus overfitting the data. Using a well designed *single* model (instead of TS) yields approximations with an orthogonal axes bias, and a tendency of the approximation to undergo changes at fractions of low powers of two. But the concept of the two spirals is clearly learned by capacity controlled single models, proving the high flexibility of this model class.

The second task is the **Gene** benchmark (Nordewier, Towell and Shavlik 1991) for predicting splice-junctions. Problem description, data, and encoding conventions can be found and are adopted from the the *PROBEN1* collection in the Neural

Bench Archive at CMU (**ftp.cs.cmu.edu**, directory /afs/cs/project/connect/bench/contrib/prechelt). The problem provides 120 binary inputs, and 3175 data labeled according to three classes. Three models are trained separately to discriminate each class against the other classes. 900 patterns are used for training, 100 for crossvalidation, the rest for testing. Training is stopped when no further improvement can be achieved on the training set within a reasonable number of iterations. We use the "order" prior. We find models of size ≈ 60 with interactions up to second order to be superior. For prediction, we chose the label of the model with the maximum active output as class label. Test error is $7 - 8\%$, comparable to results from literature with MLPs, and superior to experiments with $ID3$ and Nearest Neighbor (Murphy and Aha 1992, UCI Repository of machine learning databases, **ftp.ics.uci.edu**, University of California, Irvine, CA).

A drastic reduction of prediction error to slightly above 6% is achieved by aggregating 50 models for each class, starting with different seeds for the random number generator, and using majority voting for the class decision. Best voting results are achieved with models of size $= 250$, and interactions up to order 4 are found significant. Models trained on the same class exhibit a strong overlap of the more significant interaction terms, and high variability among the weaker interactions. We conclude that the improvement in prediction accuracy arises from less biased size constraints in conjunction with reduced variance of the aggregated classifier.

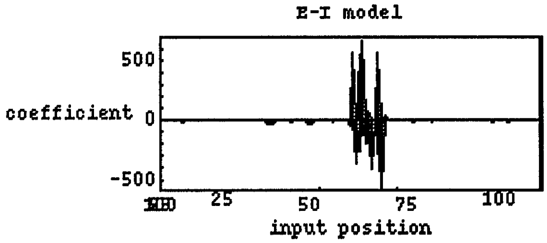

Figure 5: Structure histogram for an Exon-Intron boundary predictor aggregated from 50 models: Inputs are ordered along the horizontal axis. Vertical axis measures absolute weight coefficients of interactions. A peak indicates that the corresponding bit contributes (in an additive way or participating in some higher order interaction) to the classifier decision; the height of the peak measures the strength of this contribution.

Fig.5 reveals some gross information on the structure of interactions for the gene-splicing problem. A striking observation is that genes in the local neighborhood of the junction have the most impact on the type of junction, and that no significant long range interactions are present.

Discussion

The paper contributes original research to: integrated data and knowledge representation for numeric and categorical data, model simplicity and scalability issues, and distributed search for the best model. The computationally feasible automatic model identification determines relevant interactions between binary variables. It constitutes a much faster, less biased, and wider applicable modelling process than human feature selection. This makes our data mining tool a good candidate for real-time and high dimensional data analysis. A serious challenge for automatied search of wide model classes is the problem of overfitting. We overcome this difficulty by incorporating powerful novel regularization techniques for binary data formats. Our first simulation results hint that model aggregation further stabilizes predictions.

Simplifying the computation of feature sets becomes an important issue regarding explosive growth of data mines, shrinking time spans for data analysis and decision making, and state-of-the-art sampling and aggregation techniques. Our findings show that floating point multiplications can be avoided for some important data mining applications, speeding up the inference process significantly. A technological quantum jump may help to overcome severe scaling problems. The bit-interactions which make up the "atomic" knowledge entities of our model seem well suited for parallel distributed processing by novel computing technologies under development. It remains to be seen if models similar in spirit could simplify the implementation of statistical learning algorithms on future quantum computers for large scale, high-speed data mining.

References

Elder IV, J. F., and Pregibon, D. 1996. A statistical perspective on knowledge discovery in databases. In: *Advances in Knowledge Discovery and Data Mining*. U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy eds. Menlo Park, CA: The MIT Press.

Fahner G., and Eckmiller, R. 1994. Structural adaptation of parsimonious higher-order neural classifiers. *Neural Networks* 7(2):279-289.

Lang, K. J., and Witbrok M. 1988. Learning to tell two spirals apart. In: Proceedings of the 1988 Connectionists Model Summer School, 52-59. New York: Morgan Kaufmann Publishers.

Noordewier M. O., Towell G. G., and Shavlik J. W. 1991. Training knowledge-based neural networks to recognize genes in DNA sequences. In Advances in Neural Information Processing Systems 3:530-536. San Mateo, CA: Morgan Kaufmann.

Vapnik, V. N. 1995. *The Nature of Statistical Learning Theory.* New York: Springer.

Walsh, J. L. 1923. A closed set of orthogonal functions. *American Journal of Mathematics* 45:5-24.