Similarity Networks for Heterogeneous Data

Lluís A. Belanche Technical University of Catalonia, Barcelona (Spain)
Jerónimo Hernández University of the Basque Country, Donostia (Spain)
belanche@lsi.upc.edu jeronimo.hernandez@ehu.es

Contribution

A two-layer neural network is developed in which the neuron model computes a user-defined *similarity function* between inputs and weights.

The model is capable of dealing directly with variables of potentially different nature (continuous, ordinal, categorical); there is also provision for missing values.

The network is trained using a fast two-stage procedure involving only one parameter.

The network achieves superior performance on a set of challenging problems with respect to both RBF nets and RBF-kernel SVMs.

The *interpretability* is greatly enhanced:

1. **the output is a linear combination of the set of similarities of the input to a subset of prototypes**
2. **the similarities are user-defined; both input and weight vectors are expressed in the original variables**

Basic Concepts

A *similarity measure* s is a symmetric function expressing how "like" two observations are.

The idea is to use specific similarity measures for different types of variables, defined in the common codomain $I_s = [0,1]$.

We use s_{ijk} to mean $s_k(x_{ik}, x_{jk})$, the similarity of observations $\mathbf{x}_i, \mathbf{x}_j$ according to variable k. We use $\mathcal{X}$ to denote a missing value.

We rescale all the similarities as $\hat{s}_{ijk} = \frac{s_{ijk}}{s_{..k}}$. Then a normalization function $n : (0, +\infty) \to (0,1)$ is applied:

$$n(x) = \frac{x^a}{x^a + 1}$$

where a controls the shape of the function. The similarity between two elements x_{ik}, x_{jk} is now computed as:

$$s_{ijk} = \begin{cases} n\left(\frac{s(x_{ik}, x_{jk})}{s_{..k}}\right) & x_{ik}, x_{jk} \neq \mathcal{X} \\ \frac{1}{2} & \text{otherwise} \end{cases}$$

Similarity-based clustering

In a clustering task the examples are grouped attending to some similarity measure.

We have developed a new version of the Leader algorithm. Given $s_0 \in I_s$, the Leader2 is guaranteed to fulfill a number of properties:

1. For any example, the similarity to its leader is *at least* s_0.
2. The similarity between any two leaders is *lower* than s_0.
3. If two examples are repeated in the dataset, they will have the *same* leader.
4. The similarity of any example to its leader is *higher* than that with any other leader.

Similarity Neural Networks

A Similarity neural network (SNN) is a feed-forward architecture, with a hidden layer composed of S-neurons and a standard output layer.

$$\Phi_k(\mathbf{x}) = \sum_{i=1}^{h} w_{ki}\phi_i(\mathbf{x}) + w_{k0}, \;\; k = 1, \ldots, m$$

where $h > 0$ is the number of hidden S-neurons, m is the number of outputs and $W = (w_{ki})$ is the weight matrix.

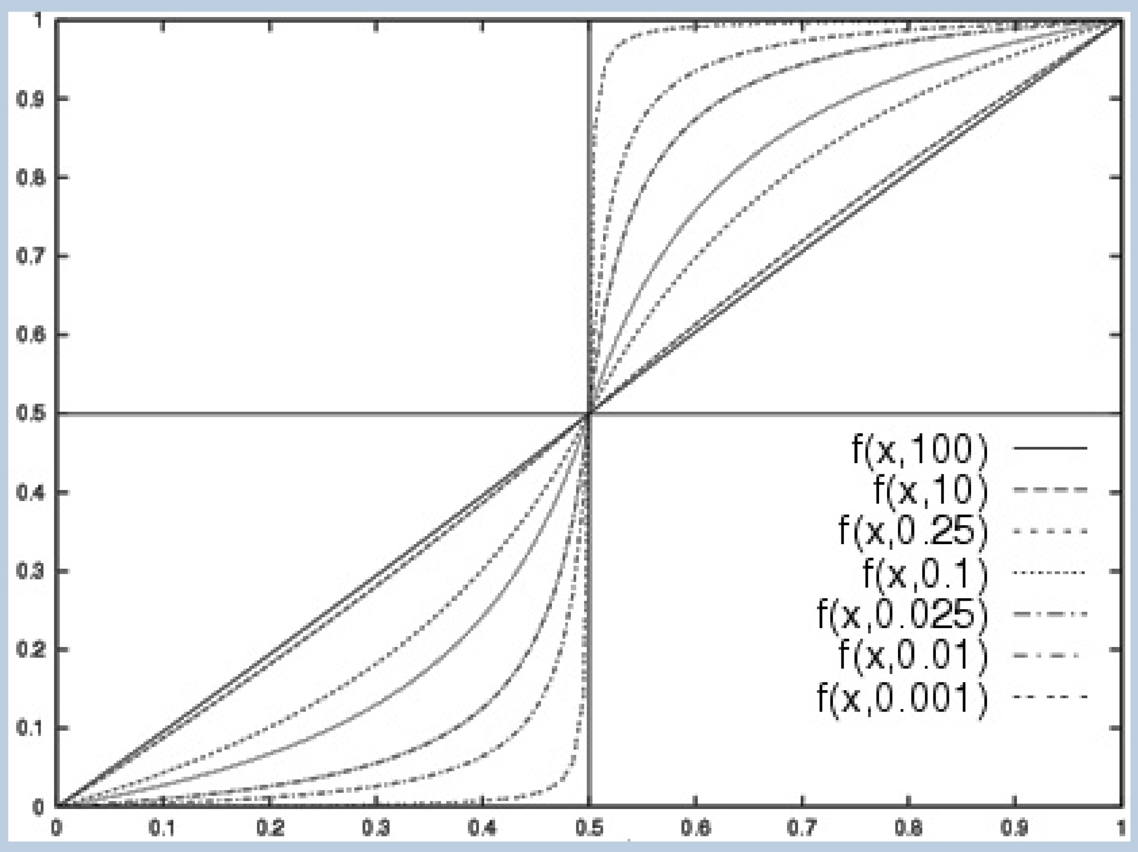
The SNN can be naturally seen as a generalization of the RBF.

A *neuron model* can be devised that is both similarity-based and handles data heterogeneity and missing values:

$$\phi_i(\mathbf{x}) = f(s(\mathbf{x}, \mu_i), p_i)$$

where $s(\mathbf{x}, \mu_i) = \frac{1}{n}\sum_{k=1}^{n} s_k(x_k, \mu_{ik})$

This S-neuron adds a non-linear *activation* function to the linearly aggregated similarities:



Family of sigmoidal functions $f(\cdot, p)$ for different values of p

Neural Network Training

Training can be solved efficiently in a two-stage procedure:

1. The first layer weights are a subset of the examples in the sample dataset: the cluster *leaders* returned by the Leader2 clustering algorithm.
2. The second layer weights are found by solving a regularized least-squares problem.

Experimental comparison

The SNN is compared to:

1. a standard RBF neural network (RBFNN)
2. a SVM using the RBF kernel

The values of the different parameters (s_0 for the SNN, number of clusters and RBF width for the RBFNN, and cost and RBF width for the SVM) are optimized using a grid search.

Data set	C/R	Size	Variables	Missing?
Horse colic	C	368	21 (6N,7C,8O)	28%
Credit approval	C	690	15 (9N,6C)	5%
Voting records	C	435	16 (16N)	5.3%
Servo data	R	167	4 (2C,2N)	none

C/R stands for Classification/Regression. Legend for variable types: (N)ominal, (C)ontinuous, (O)rdinal

The resampling method consists in five repetitions of two-fold cross-validation (5×2 CV).

A paired F-test can be computed to assess potential statistical significance in the differences in performance. This test is difficult to satisfy.

Highlights

The SNN obtains:

- Better MSEs in all the problems
- Enhanced interpretability
- Full control of the modeling by means of user-defined similarities
- Possibility of adding prior knowledge
- Only one hyper-parameter (which is an intuitive one)
- Extensible to other data types

Name	Type	Value
age	BINARY	`Adult`
rectal temperature	REAL	38.48
peripheral pulse	ORDINAL	`normal`
mucous membranes	NOMINAL	`normal pink`
peristalsis	ORDINAL	`normal`
abdominal distension	ORDINAL	`none`
nasogastric reflux pH	REAL	1.12
rectal examination	NOMINAL	`decreased`

Acknowledgements

This research has been funded by the Spanish Government project TIN2009-13895-C02-01.

Results

Method	Horse colic error(%)	Horse colic MSE	Credit approval error(%)	Credit approval MSE	Voting records error(%)	Voting records MSE	Servo MSE
SNN	16.74	0.128	14.09	0.110	4.60	0.039	0.933
RBF	20.06	0.153	14.81	0.116	4.64	0.064	0.997
SVM	19.94	-	16.06	-	6.53	-	2.230

Results in terms of average 5×2 CV and mean square errors (MSE)

Method	Horse colic $F_{\%}$	Horse colic F_{MSE}	Credit approval $F_{\%}$	Credit approval F_{MSE}	Voting records $F_{\%}$	Voting records F_{MSE}	Servo F_{MSE}
RBF	3.053	**12.786**	1.337	2.884	0.805	1.036	0.786
SVM	2.584	-	3.276	-	2.298	-	**14.386**

Results of the F statistic against the SNN, both for average 5×2 CV ($F_{\%}$) and mean square errors (F_{MSE}). Significant results (> 4.74) are boldfaced