EPIDEMIOLOGIA
PIEMONTE

SAS System: caratteristiche e utilizzo della Proc HPNEURAL

Marco Dalmasso

Certified Advanced Programmer for SAS 9

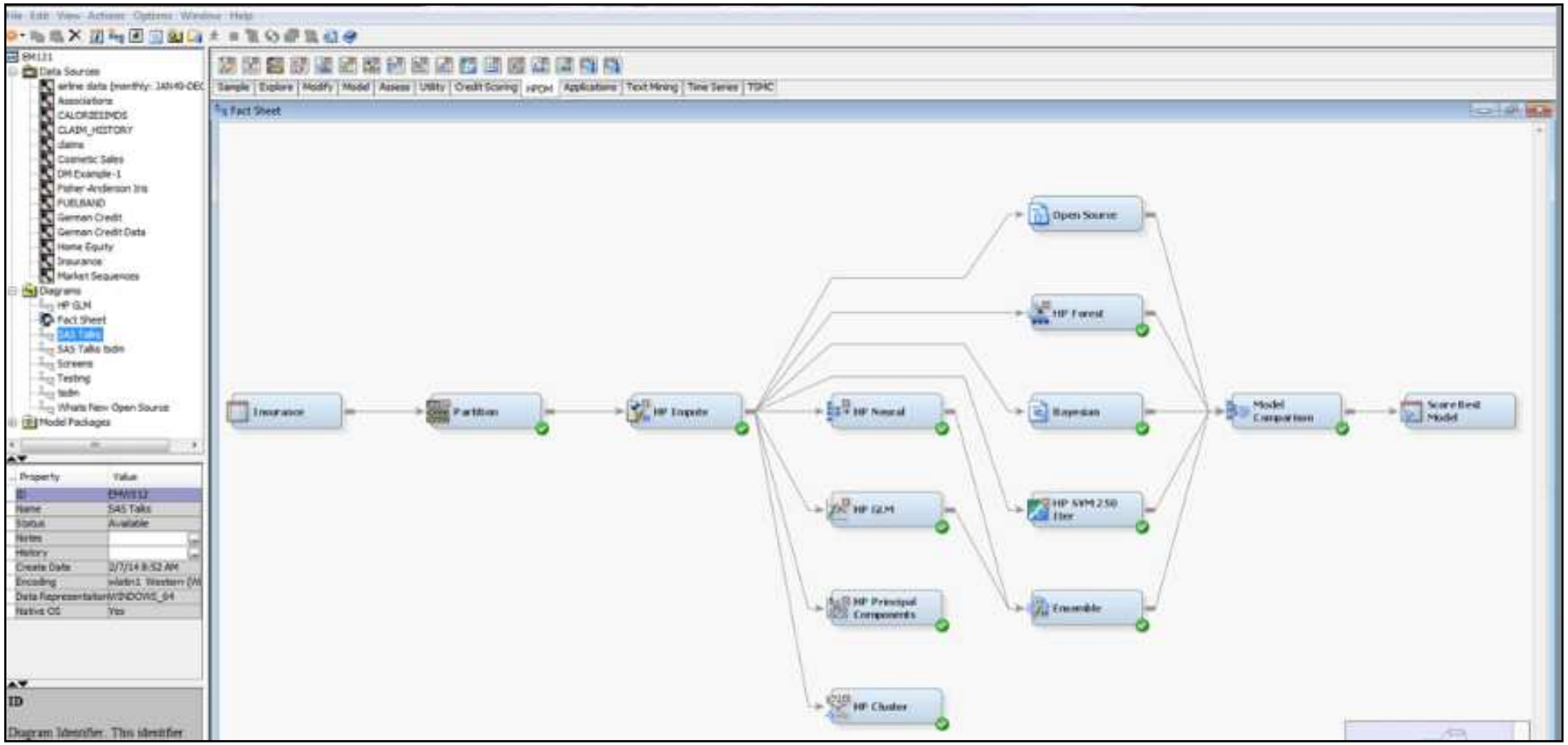
Gruppo di progetto Amministrazione DWH

Torino, martedì 8 ottobre 2019

Ambienti e funzionalità / EM

EPIDEMIOLOGIA PIEMONTE

nterprise Miner - EM costituisce l'interfaccia grafica che mette a disposizione i principali strumenti per il processo di data mining
isponibili, raggruppati in un flusso dei processi, permettono la definizione e il confronto di modelli descrittivi e predittivi, oltre a
nalità di trattamento e trasformazione dati (accesso e integrazione, campionamento, esplorazione grafica, generazione di codic
odi di modellazione e apprendimento comprendono alberi decisionali, regressione, reti neurali, random forest, reti bayesiane,
ring, SVM; sono presenti nodi per la registrazione, il confronto e la distribuzione dei modelli costruiti.



Ambienti e funzionalità / le Proc HPxxxx

EPIDEMIOLOGIA PIEMONTE

o disponibili 8 procedure di data mining HP - high performance sviluppate per utilizzare architetture distribuite su più macchine
nunque utilizzabili anche su macchine singole.

Proc	Funzionalità
PBNET	Reti bayesiane
PCLUS	Clustering
PDECIDE	Matrici di decisione
PFOREST	Random forest
P4SCORE	Valutazione dell'importanza delle variabili derivanti da modelli HPFOREST
PNEURAL	Reti neurali
PREDUCE	Selezione di variabili
PSVM	Support vector machine

```
proc hpneural data=iris;
   input SepalLength SepalWidth PetalLength PetalWidth;
   target Species / level=nom;
   hidden 2;
   train outmodel=model_iris maxiter=1000;
   score out=scores_iris;
   code file = 'c:\personale\d\codice.sas';
run;
```

Ambienti e funzionalità / la Proc HPNEURAL

EPIDEMIOLOGIA PIEMONTE

gth	SepalWidth	PetalLength	PetalWidth	Species
50	33	14	2	Setosa
64	28	56	22	Virginica
65	28	46	15	Versicolor
67	31	56	24	Virginica
63	28	51	15	Virginica
46	34	14	3	Setosa
69	31	51	23	Virginica
62	22	45	15	Versicolor
59	32	48	18	Versicolor
46	36	10	2	Setosa
61	30	46	14	Versicolor

```
proc hpneural data=iris;
    input SepalLength SepalWidth PetalLength PetalWidth;
    target Species / level=nom;
    hidden 2;
    train outmodel=model_iris maxiter=1000;
    score out=scores_iris;
    code file = 'c:\personale\d\codice.sas';
run;
```

Informazioni sul modello	
Origine dati	W
Architettura	
Numero di variabili di input	
Numero di lager nascosti	
Numero di neuroni nascosti	
Numero di variabili target	
Numero di pesi	
Tecnica di ottimizzazione	BFGS memoria
Numero osservazioni lette	
Numero osservazioni usate	
Numero usato per il train	
Numero usato per la valid	

gth	SepalWidth	PetalLength	PetalWidth	Species	PVIRGINICA	PVERSICOLOR	PSETOSA	Predicted
50	33	14	2	Setosa	0.00007	0.01288	0.98705	SETOSA
64	28	56	22	Virginica	0.96427	0.03573	0	VIRGINICA
65	28	46	15	Versicolor	0.13005	0.86886	0.00109	VERSICOLOR
67	31	56	24	Virginica	0.96335	0.03665	0	VIRGINICA
63	28	51	15	Virginica	0.32113	0.67839	0.00047	VERSICOLOR
46	34	14	3	Setosa	0.00006	0.01132	0.98862	SETOSA
69	31	51	23	Virginica	0.9492	0.0508	0.00001	VIRGINICA
62	22	45	15	Versicolor	0.8855	0.11448	0.00002	VIRGINICA
59	32	48	18	Versicolor	0.07784	0.92005	0.00212	VERSICOLOR
46	36	10	2	Setosa	0.00005	0.00941	0.99054	SETOSA
61	30	46	14	Versicolor	0.00671	0.9864	0.00689	VERSICOLOR

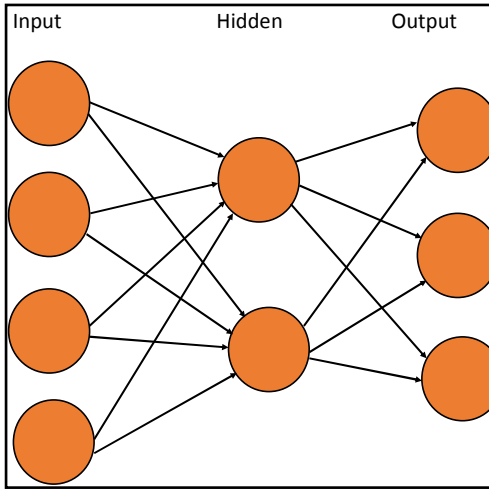
Tabella degli errori di classificazione per Species			
Classe:	VIRGINICA	VERSICOLOR	SI
VIRGINICA	12	2	
VERSICOLOR	0	11	
SETOSA	0	0	

Ambienti e funzionalità / la Proc HPNEURAL

EPIDEMIOLOGIA PIEMONTE

```
oneural data=iris;
t SepalLength SepalWidth PetalLength PetalWidth;
et Species / level=nom;
en 2;
 outmodel=model_iris maxiter=1000;
e out=scores_iris;
 file = 'c:\personale\d\codice.sas';
```

SepalLength	SepalWidth	PetalLength	PetalWidth	Species
50	33	14	2	Setosa
64	28	56	22	Virginica
65	28	46	15	Versicolor
67	31	56	24	Virginica
63	28	51	15	Virginica
46	34	14	3	Setosa
69	31	51	23	Virginica
62	22	45	15	Versicolor
59	32	48	18	Versicolor
46	36	10	2	Setosa
61	30	46	14	Versicolor



VarName	LevelName	Level	Layer	StdType	Min	Max	Avg	Stdev	ActFunc	ErrFunc	MostFrequentLevel	WeightsTo_H1	WeightsTo_H2	WeightsTo_T1_0	WeightsTo_T1_1	WeightsTo_
1.0.39	MLP	5	3		.	.	.	.			.	.	.	.	.	
		.	.		.	.	.	.			.	1.420977	-0.76511	0.184592	0.981901	
epalLength		.	0	Range	43	79	58.43	8.28			.	-0.35909	-0.48253	.	.	
epalWidth		.	0	Range	20	44	30.57	4.36			.	2.018893	1.79605	.	.	
etalLength		.	0	Range	10	69	37.58	17.7			.	-1.46605	-1.34202	.	.	
etalWidth		.	0	Range	1	25	11.99	7.62			.	-2.30786	-0.73196	.	.	
		.	1		.	.	.	.	Tanh		.	.	.	-4.66187	1.080332	
		.	1		.	.	.	.	Tanh		.	.	.	-0.6483	-2.03888	
pecies	VIRGINICA	0	2		.	.	.	.	Softmax	CrossEnt	0	.	.	.	.	
pecies	VERSICOLOR	1	2		.	.	.	.	Softmax	CrossEnt	0	.	.	.	.	
pecies	SETOSA	2	2		.	.	.	.	Softmax	CrossEnt	0	.	.	.	.	

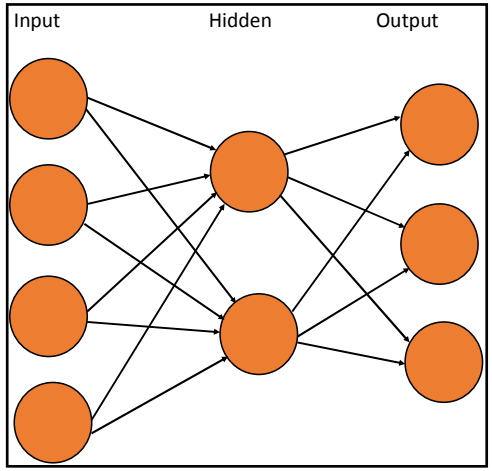
Ambienti e funzionalità / la Proc HPNEURAL

EPIDEMIOLOGIA PIEMONTE

```
oneural data=new_iris;
e model=model_iris
  out=scores_new_iris;
```

```
_H1 = 1.420977 - 0.35909 * _I1 + 2.018893 * _I2 - 1.46605 * _I3 - 2.30786 * _I4;
_H2 = -0.76511 - 0.48253 * _I1 + 1.79605 * _I2 - 1.34202 * _I3 - 0.73196 * _I4;

_T1_0 =  0.184592 - 4.66187 * _H1 - 0.6483 * _H2;
_T1_1 =  0.981901 + 1.080332 * _H1 - 2.03888 * _H2;
_T1_2 = -1.320156988 + 3.6566887582 * _H1 + 2.4424117074 * _H2;
```



ngth	SepalWidth	PetalLength	PetalWidth
50	33	14	2
64	28	56	22
65	28	46	15
67	31	56	24
63	28	51	15
46	34	14	3
69	31	51	23
62	22	45	15
59	32	48	18
46	36	10	2
61	30	46	14
60	27	51	16
65	30	52	20
56	25	39	11
65	30	55	18
58	27	51	19

Into: Species	Predicted: Species=VIRGINICA	Predicted: Species=VERSICOLOR	Predicted: Species=SETOSA
SETOSA	0.00007	0.01288	0.98705
VIRGINICA	0.96427	0.03573	0
VERSICOLOR	0.13005	0.86886	0.00109
VIRGINICA	0.96335	0.03665	0
VERSICOLOR	0.32113	0.67839	0.00047
SETOSA	0.00006	0.01132	0.98862
VIRGINICA	0.9492	0.0508	0.00001
VIRGINICA	0.8855	0.11448	0.00002
VERSICOLOR	0.07784	0.92005	0.00212
SETOSA	0.00005	0.00941	0.99054
VERSICOLOR	0.00671	0.9864	0.00689
VIRGINICA	0.67144	0.32845	0.00011
VIRGINICA	0.89367	0.10631	0.00002
VERSICOLOR	0.00324	0.98817	0.00859
VIRGINICA	0.80452	0.19543	0.00005
VIRGINICA	0.91725	0.08274	0.00001

Ambienti e funzionalità / la Proc HPNEURAL

EPIDEMIOLOGIA PIEMONTE

OC HPNEURAL <DATA=*SAS-data-set*> <DISTR=ALL | SPLIT> <NOPRINT> **;**
RFORMANCE *performance-options*;
CHITECTURE *architecture-options*;
ariables;
UT *variables* </ LEVEL=INT | LEVEL=NOM <MISSING=MAP>>;
IGHT *variable* | _INVERSE_PRIORS_;
DEN *number* </ ACT= *activation-function*>;
RGET *variables* </ <LEVEL=INT | LEVEL=NOM >
CT=*activation-function*> <ERROR=*error-function*>>;
RTITION ROLEVAR=*variable(* TRAIN=*value* | VALIDATE=*value)*;
RTITION FRACTION*(* TRAIN=*number* | VALIDATE=*number)*;
AIN <NUMTRIES=*number*> <MAXITER=*number*>
ALID=NONE> <OUTMODEL=*SAS-data-set*> **;**
ORE OUT=*SAS-data-set* <MODEL=*SAS-data-set*> **;**
DE FILE=*'external-file'* | *fileref*;

ning: INPUT, TARGET, and TRAIN required. HIDDEN statement is required unless you use the logistic
hitecture (in which case, the HIDDEN statement is not allowed).
ring: only SCORE, ID, PERFORMANCE, and CODE statements are allowed.

Ambienti e funzionalità / la Proc HPNEURAL

EPIDEMIOLOGIA PIEMONTE

Statement

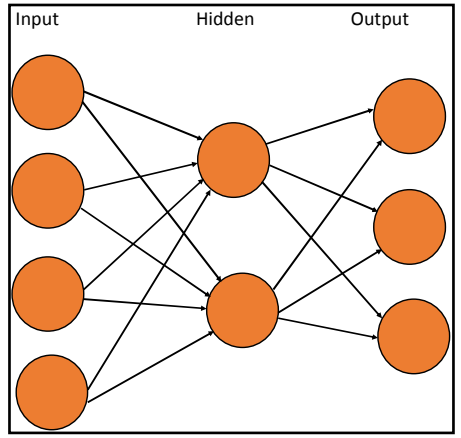
variables </ LEVEL=INT | LEVEL=NOM <MISSING=MAP>>;
UT statement identifies the *variables* in the input data set that are inputs to the neural network.
=INT
s that the *variables* are interval variables, which must be numeric. This is the default.
=NOM
s that the *variables* are nominal variables, also known as classification variables, which can be numeric or character.
aining, you must include one or more INPUT statements. You need more than one INPUT statement when you have both interval and nominal
riables.
val input variables are automatically standardized to the range [–1, 1].

ET Statement

T *variables* </ <LEVEL=INT | LEVEL= NOM > <ACT=*activation-function*> <ERROR=*error-function*>>;
GET statement identifies the *variables* in the input data set that the network is to be trained to predict.
INT | NOM
the *variables* type. **INT i**nterval variables, which must be numeric. **NOM** nominal variables which can be numeric or character.
OS | EXP | IDENTITY | TANH
the activation function for interval target variables only. For nominal variables is always the softmax function.
ecifies the cosine function **EXP** specifies the exponential function **IDENTITY** specifies the identity function **TANH** specifies the hyperbolic
angent function.
=GAMMA | NORMAL | POISSON
variables have one target neuron per class level, except for nominal variables that have only two levels (binary variables), which have a
uron.
ining, you must include one or more TARGET statements.

N Statement

l *number* </ ACT=*activation-function*>;
the number of hidden layers in the network and the number of neurons in each hidden layer. The number of HIDDEN statements specifies the
of layers in the network.
HIDDEN statement specifies the number of hidden neurons in the first hidden layer. The second HIDDEN statement specifies the number of
eurons in the second hidden layer, and so on.
um of 100 HIDDEN statements are allowed.
OS | IDENTITY | TANH
ining, you must include one or more HIDDEN statements, unless you specify ARCHITECTURE LOGISTIC (in which case the HIDDEN
t is not allowed).




```
proc hpneural data=iris;
   input SepalLength SepalWidth PetalLengtl
   target Species / level=nom;
   hidden 2;
   train outmodel=model_iris maxiter=1000;
   score out=scores_iris;
   code file = 'c:\personale\d\codice.sas';
run;

proc hpneural data=new_iris;
   score model=model_iris out=scores_new_
run;
```

Ambienti e funzionalità / la Proc HPNEURAL

EPIDEMIOLOGIA PIEMONTE

Statement

of training is to determine a set of network weights that best predicts the targets in the training data while still doing a good job of predicting f unseen data. Training starts with a pseudorandomly generated set of initial weights. PROC HPNEURAL then computes the objective or the training subset, and the optimization algorithm adjusts the weights. This process is repeated until the objective function that is d using the training subset stops improving. The weights that result in the smallest value of the objective function are saved and used for g fit statistics and for scoring.

ES specifies the *number* of times the network is to be trained using a different starting points.

R specifies the maximum number of iterations (weight adjustments) for the optimizer to make before terminating.

DEL specifies the data set to which to save the model parameters for the trained network. You can use the model data set later to score a input data set.

E Statement

OUT=*SAS-data-set* <MODEL=*SAS-data-set*> ;

ORE statement causes the HPNEURAL procedure to write the network's target and predicted output for each observation in the input data set tput data set that is specified by the OUT option.

S-data-set: specifies the data set to contain the predicted values of the target variables. For nominal variables, each observation also the computed probabilities of each class level. This keyword is required.

=*SAS-data-set*: specifies the data set that contains the model parameters for a previously trained network.

u are training, the SCORE statement is optional but the MODEL= keyword is not allowed. When you are doing stand-alone scoring, the statement is required and the MODEL= keyword must be used.

Statement

DE statement uses the current neural network model to generate SAS DATA step statements and save them in an external text file that sed later.

loes not contain the surrounding PROC and RUN statements.

TECTURE Statement

C multilayer perceptron that has no hidden units (which is equivalent to a logistic regression).

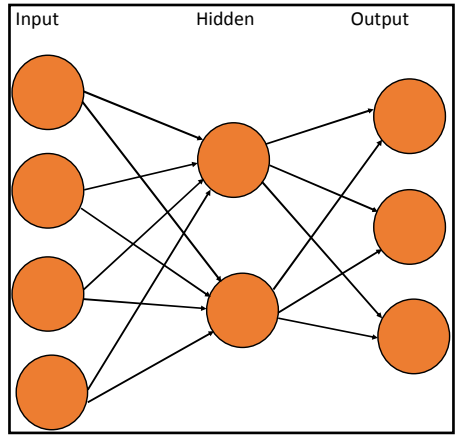
tilayer perceptron that has one or more hidden layers.

RECT specifies a multilayer perceptron that has one or more hidden layers and has direct connections between each input and each uron.

TION Statement

how to divide the input data set into a training subset and a validation subset.

not include the PARTITION statement, every fourth observation (starting with the first observation) is used as a validation observation.




```
proc hpneural data=iris;
   input SepalLength SepalWidth PetalLength P
   target Species / level=nom;
   hidden 2;
   train outmodel=model_iris maxiter=1000;
   score out=scores_iris;
   code file = 'c:\personale\d\codice.sas';
run;

proc hpneural data=new_iris;
   score model=model_iris out=scores_new_iris
run;
```

Ambienti e funzionalità / la Proc HPNEURAL

EPIDEMIOLOGIA PIEMONTE

```
pneural data = comuni;
comune ycomune den_com sigla_prov;
ut xcomune ycomune / level = int;
get sigla_prov / level = nom;
den x / act = xxxxxxx;
n outmodel = model maxiter = 100;
re out = scores;
```

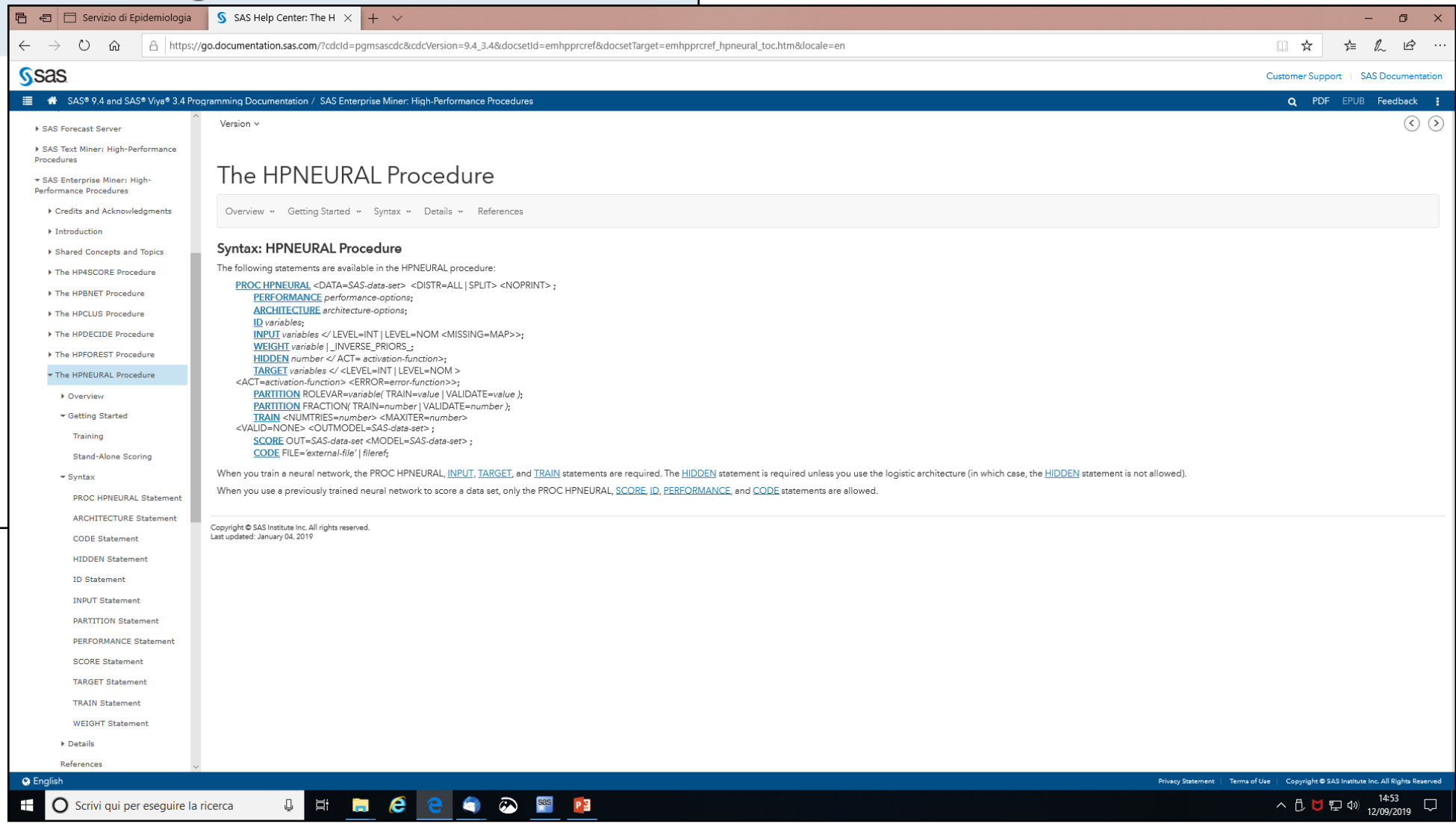
ero di bili di out	Numero di variabili target	Numero osservazioni usate	Numero usato per il training	Numero usato per la validazione	Numero di layer nascosti	Numero di neuroni nascosti	Funzione di attivazione	Numero di pesi	% Classificazione corretta
2	1	1181	885	296	1	1	Identity	19	53.76799
					1	2	Identity	30	89.16173
					1	3	Identity	41	88.99238
					1	2	Identity	30	89.16173
					1	2	cos	30	82.30313
					1	2	tanh	30	88.73836
					2	1*2	Identity	21	53.25995
					2	2*2	Identity	36	88.39966

Ambienti e funzionalità / la Proc HPNEURAL

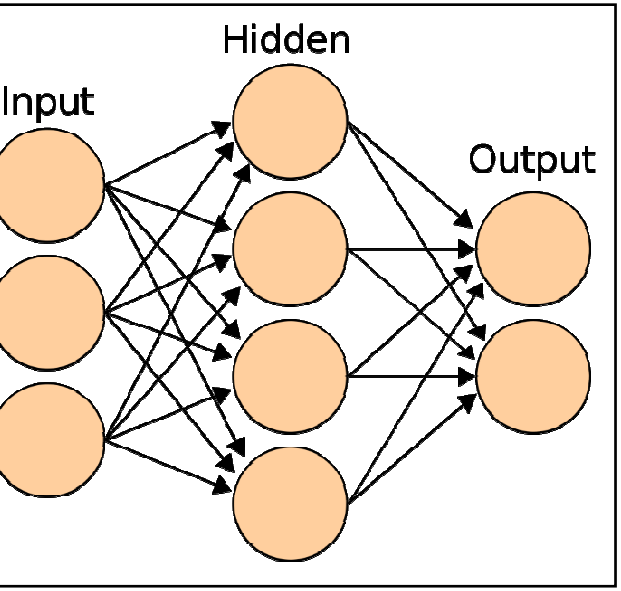
EPIDEMIOLOGIA PIEMONTE

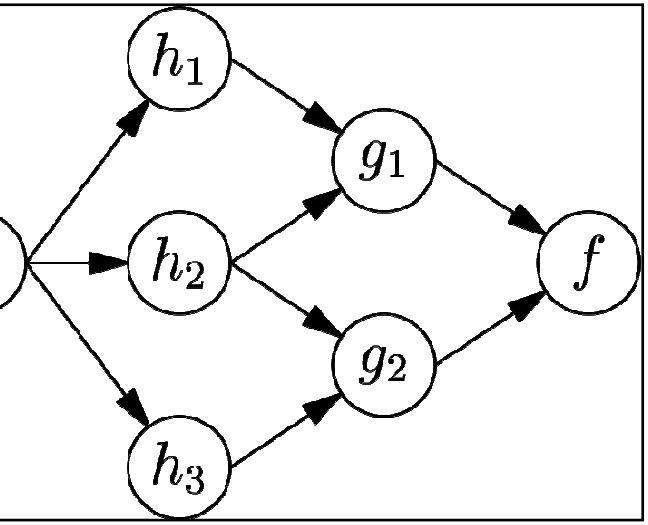
S Enterprise Miner 15.1: High-Performance Procedures

ts
luction
d Concepts and Topics
P4SCORE Procedure
PBNET Procedure
PCLUS Procedure
PDECIDE Procedure
PFOREST Procedure
PNEURAL Procedure
PREDUCE Procedure
PSVM Procedure



Rete neurale





... a neural network is a system composed of many simple processing elements operating in parallel whose function is determined by network structure, connection strengths, and the processing performed at computing elements or nodes.
DARPA Neural Network Study (1988)

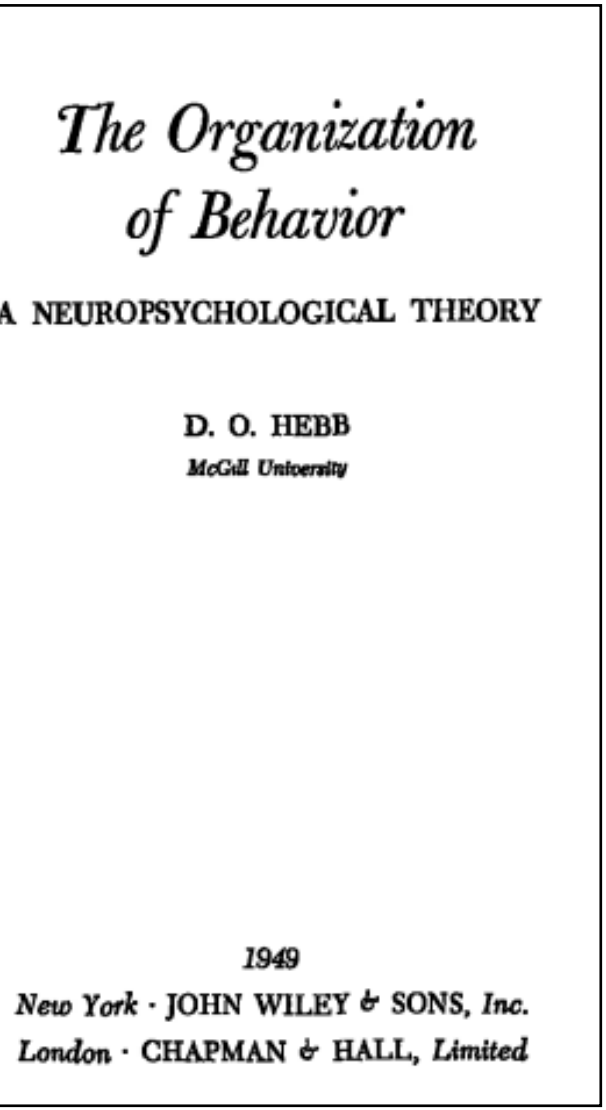
A neural network is a massively parallel distributed processor that has a natural propensity storing experiential knowledge and making it available for use. It resembles the brain in two respects:
•Knowledge is acquired by the network through a learning process.
•Interneuron connection strengths known as synaptic weights are used to store the knowle
Haykin (1994)

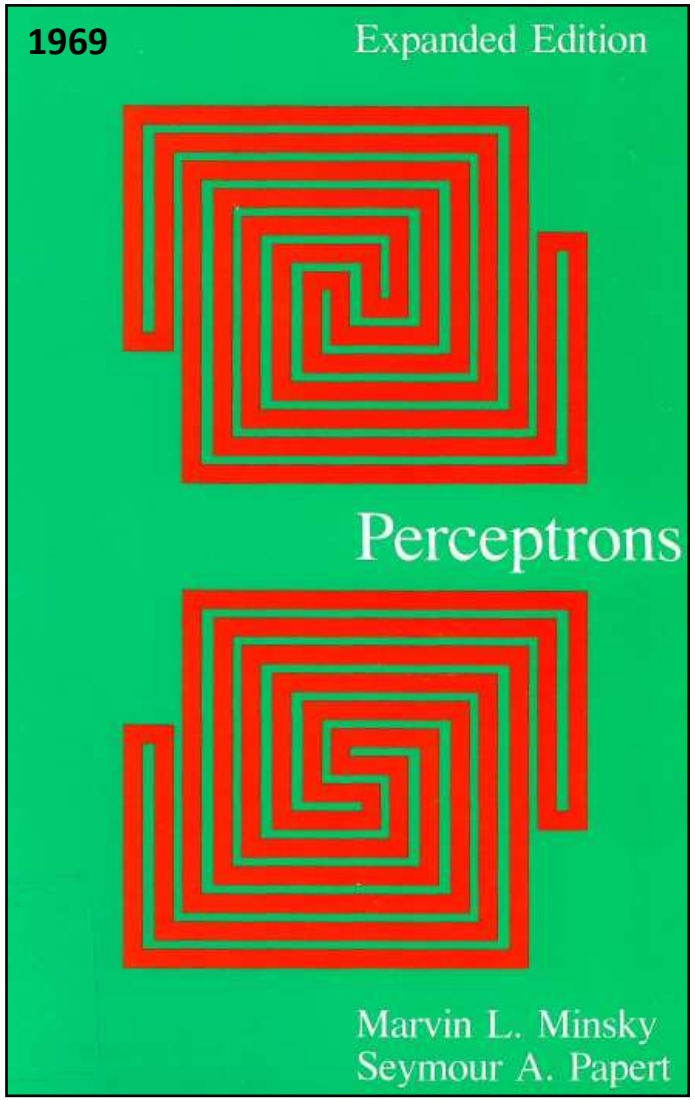
A neural network is a circuit composed of a very large number of simple processing element are neurally based. Each element operates only on local information. Furthermore each ele operates asynchronously; thus there is no overall system clock.
Nigrin (1993)

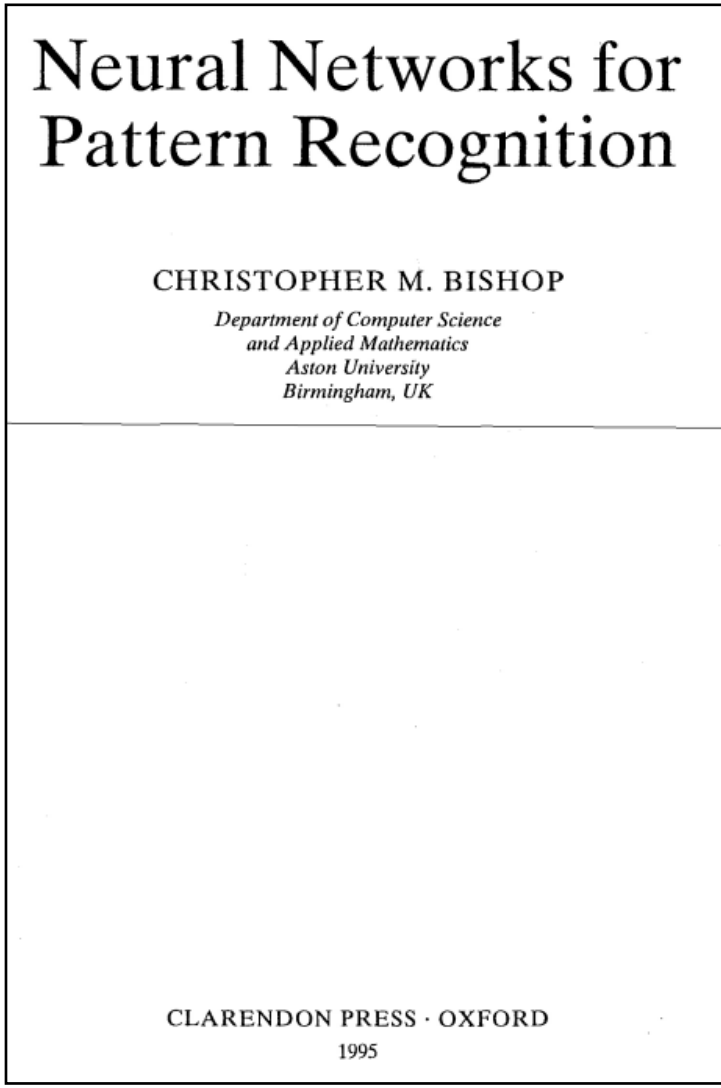
Artificial neural systems, or neural networks, are physical cellular systems which can acquire store, and utilize experiential knowledge.
Zurada (1992)

Rete neurale

EPIDEMIOLOGIA PIEMONTE

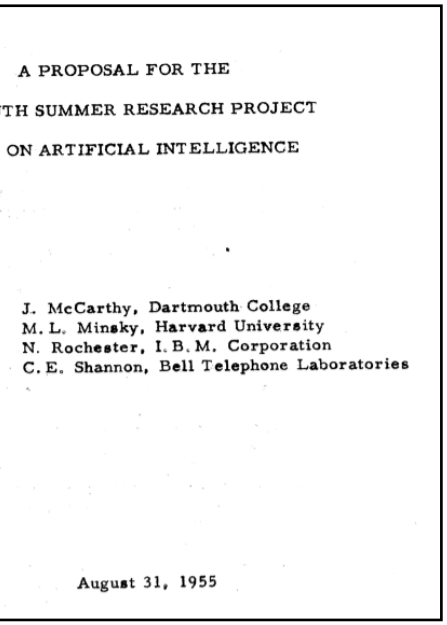






Rete neurale

EPIDEMIOLOGIA PIEMONTE



A PROPOSAL FOR THE

TH SUMMER RESEARCH PROJECT

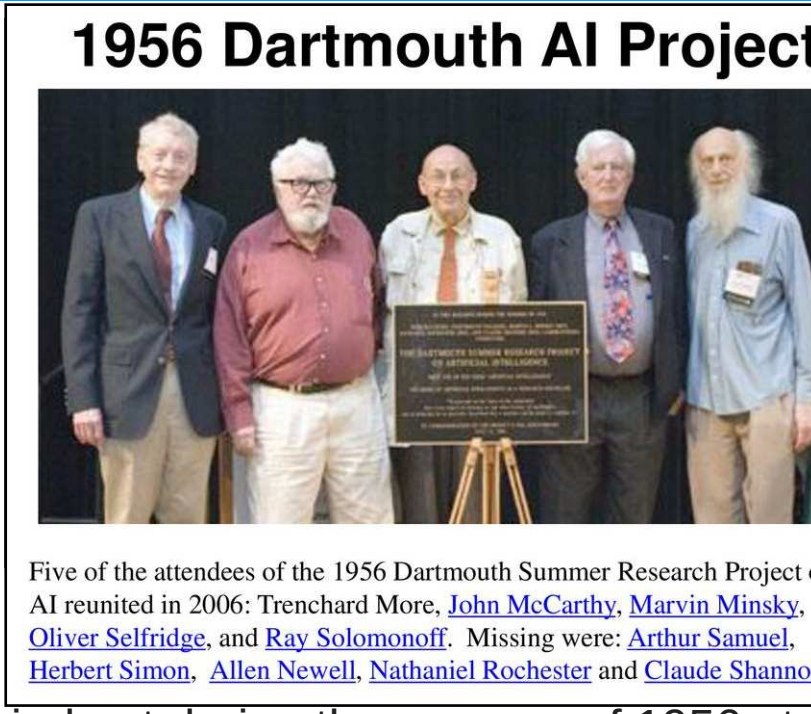
ON ARTIFICIAL INTELLIGENCE

J. McCarthy, Dartmouth College
M. L. Minsky, Harvard University
N. Rochester, I. B. M. Corporation
C. E. Shannon, Bell Telephone Laboratories

August 31, 1955

Photo from the 1956 Dartmouth Conference with Marvin Minsky, Ray Solomonoff, Claude Shannon, John McCarthy, Trenchard More, Oliver Selfridge and Nathaniel Rawchester

From about June 18 to August 17, 1956.



1956 Dartmouth AI Project

Five of the attendees of the 1956 Dartmouth Summer Research Project
AI reunited in 2006: Trenchard More, John McCarthy, Marvin Minsky,
Oliver Selfridge, and Ray Solomonoff. Missing were: Arthur Samuel,
Herbert Simon, Allen Newell, Nathaniel Rochester and Claude Shanno

propose that a 2-month, 10-man study of artificial intelligence be carried out during the summer of 1956 at
mouth College. The study is to proceed on the basis of the conjecture that every aspect of learning or any
r feature of intelligence can in principle be so precisely described that a machine can be made to simulate
ttempt will be made to find how to make machines use language, form abstractions and concepts, solve ki
oblems now reserved for humans, and improve themselves.

studio procederà sulla base della congettura per cui, in linea di principio, ogni aspetto dell'apprendimento
siasi altra caratteristica dell'intelligenza possano essere descritte così precisamente da poter costruire una
china che le simuli.»

Rete neurale

EPIDEMIOLOGIA PIEMONTE

eti neurali sono particolarmente utili quando sono presenti grandi quantità di dati per il training e quando regole ed algoritmi esatti non
o disponibili. Ovviamente anche le reti neurali non possono estrarre informazione che non sia contenuta nel training set.
alcuni problemi anche le reti neurali non sono in grado di imparare (tranne che memorizzando l'intero training set); ad esempio predire
neri casuali, determinare se un numero è primo, decriptare informazioni criptate da un buon algoritmo.

ne altri strumenti di data mining, l'apprendimento tramite reti neurali può imparare (e riprodurre nel model dataset) comportamenti
izionalisti o addirittura preconcetti derivanti dai comportamenti passati: ad esempio, il software di reclutamento del personale di Amaz
serie di "distorsioni" storiche privilegiava le assunzioni maschili rispetto a quelle femminili (oppure la previsione di candidati autisti GTT
ire dal dataset storico delle assunzioni difficilmente individuerebbe lavoratori stranieri). È necessario almeno rinfrescare il training set ir
enere presenti anche le ultime tendenze (e magari confrontare i risultati con modelli costruiti a partire da training set precedenti).

fferenza di altri metodi, l'estrazione di conoscenza (in questo caso apprendimento) è dipendente (almeno quando il numero di osservazi
zzate è ridotto) dall'ordine in cui il dataset di input viene presentato (e quindi dalla convergenza iniziale e successiva nell'assegnazione d
).

Training set non ordinato				Training set ordinato per specie			
Classe:	VIRGINICA	VERSICOLOR	SETOSA	Classe:	VIRGINICA	VERSICOLOR	SETOSA
VIRGINICA	12	2	0	VIRGINICA	13	0	0
VERSICOLOR	0	11	0	VERSICOLOR	0	12	0
SETOSA	0	0	13	SETOSA	0	0	13

ettura dei valori dei pesi (modello di apprendimento) non è semplice e diventa quasi impossibile al crescere del numero dei livelli e dei n
è quindi agevole verificare (o anche solo ipotizzare) la plausibilità dei risultati.

zzando il dataset ComuniPiemonte.sas7bdat, costruire e commentare un modello basato su rete neurale con target la popolazione e inp
udine, grado di urbanizzazione e superficie.