Chapitre 08 – MTH8302

Réseau de Neurones Artificiels (RNA)

- Cerveau humain
- Références et vidéos
- Types de réseaux
- Quelques éléments
- Fonctions d'activation
- STATISTICA : mise en œuvre
- JMP Pro : mise en œuvre

Exemples ……………..

- Ex 1 : 4-bar linkage …..
- Ex 2 : Iris ……………….
- Ex 3 : Boston ………….
- Ex 4 : Série G ………….

…. Réponse Y

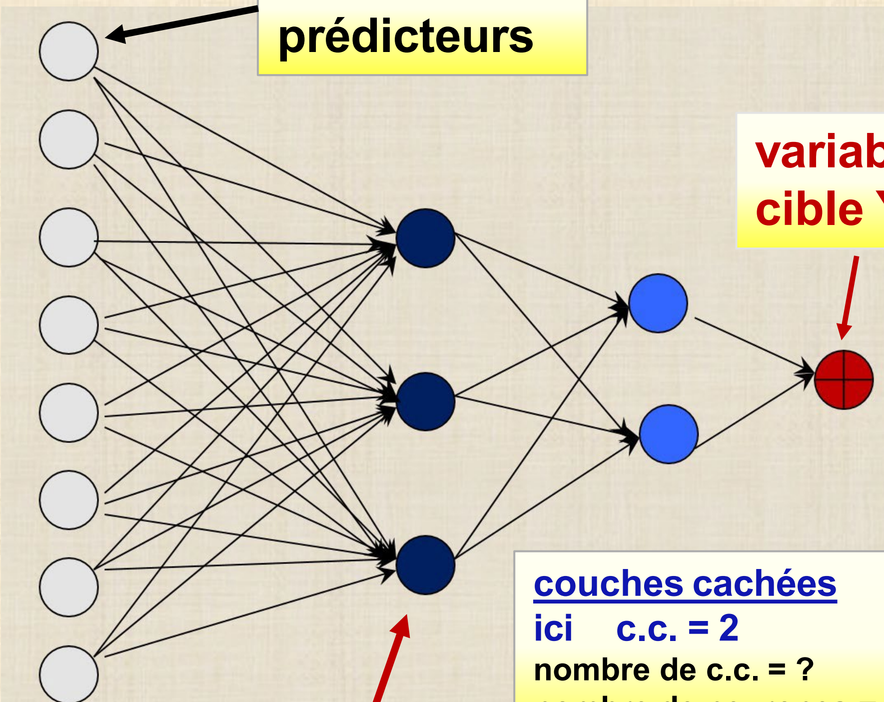
- régression
- classification
- régression
- temporelles

Copyright © Génistat Conseils Inc., 2021, Montréal, Canada

Genèse des réseaux neurones mathématiques : cerveau humain

variables X prédicteurs

variable cible Y

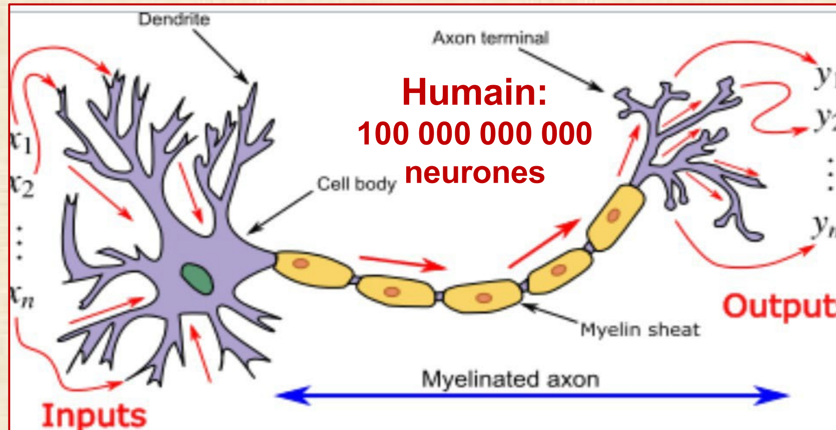


couches cachées
ici c.c. = 2
nombre de c.c. = ?
nombre de neurones = ?

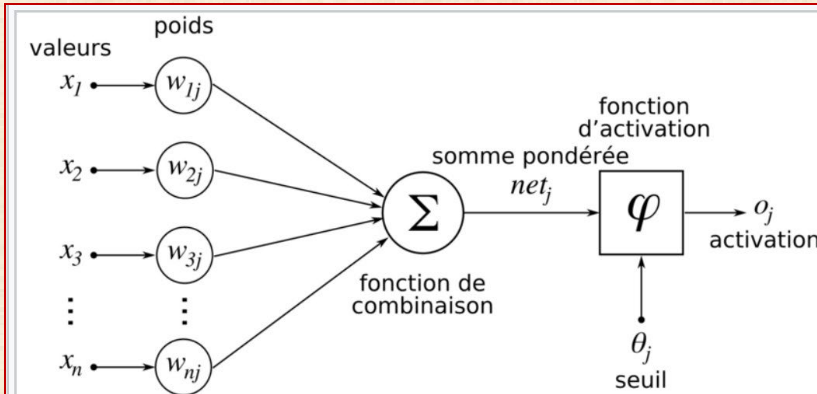
Introduction de variables artificielles

Réseau Neurones Artificiels (RNA)
Prédit une ou plusieurs réponses Y (continue, binaire, catégorique) en employant une function très flexible des variables d'entrées X.

RNA : très bon prédicteur non paramétrique
pas nécessaire de préciser la forme fonctionnelle de la réponse.

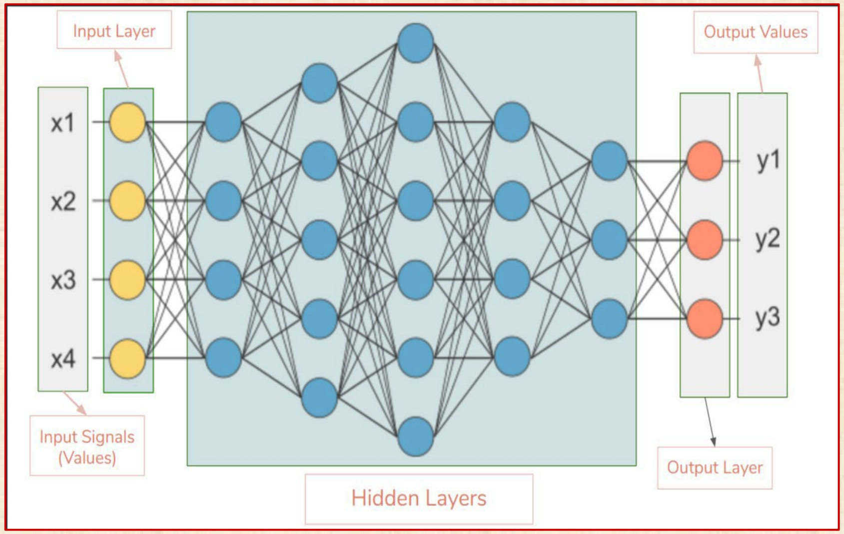


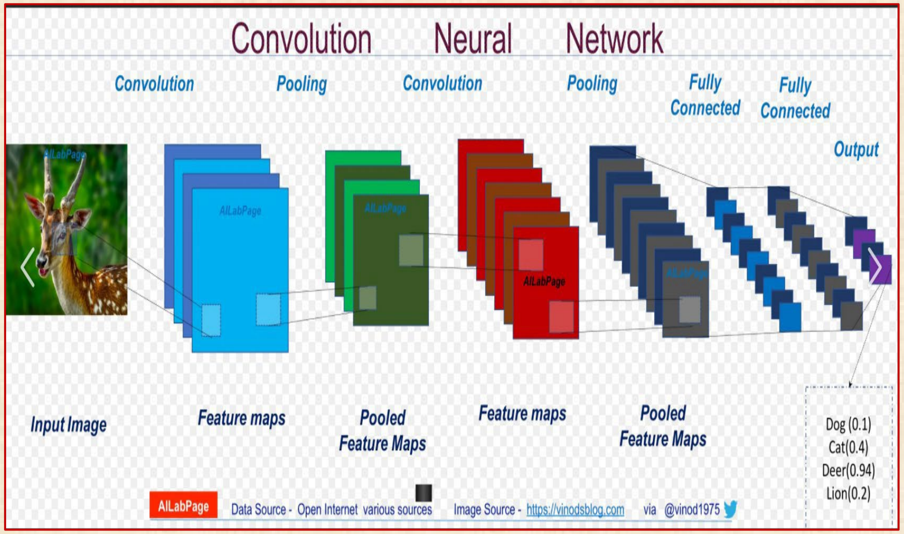
Neuron and myelinated axon, with signal flow from inputs at dendrites to outputs at axon terminals

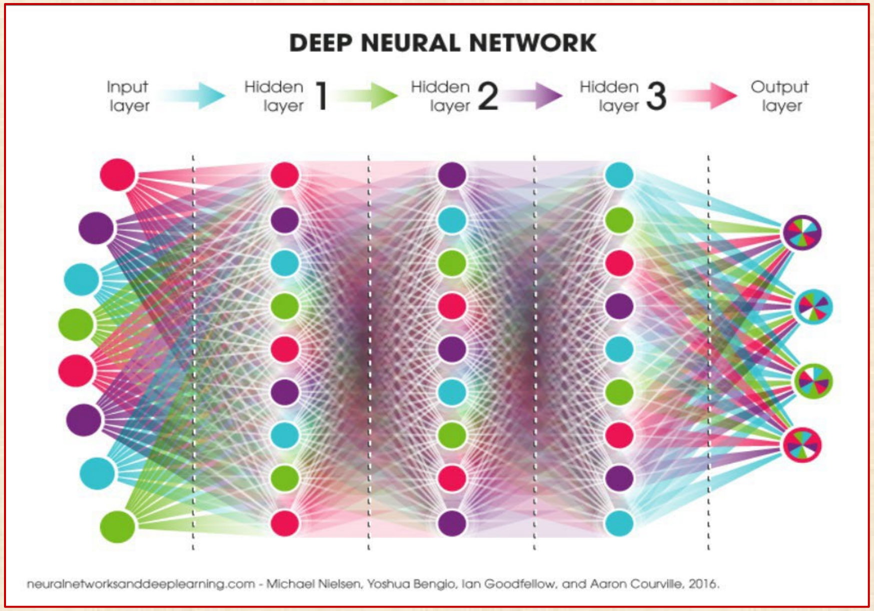


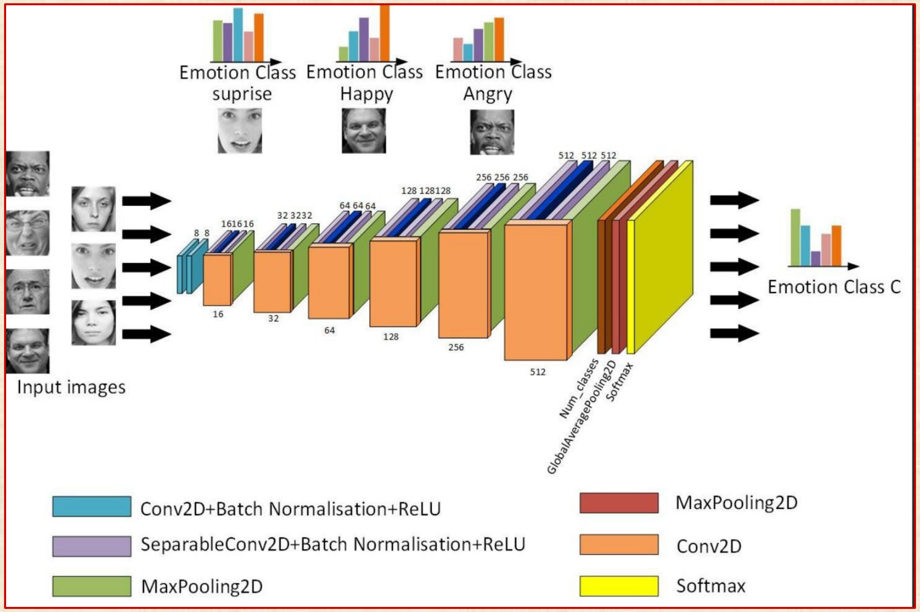
Structure d'un neurone artificiel. Le neurone calcule la somme de ses entrées puis cette valeur passe à travers la fonction d'activation pour produire sa sortie.

Images de réseaux de neuronnes









ajustement des poids w par algorithme Backpropagation

Backpropagation Algorithm

Before Weight Adjustment

Parameters

$For\ w_2 = 5 \wedge x = 2 \wedge w_1 = 3.5$

$Where\ MAE_1 = w_1x - y \wedge MAE_2 = w_2x - y$

$For\ w_2x = 10 \wedge w_1x = 7 \wedge y = 4$

$$f(x) = \frac{MAE_2^2}{2}$$

$$g(x) = \frac{MAE_1^2}{2}$$

$Backpropagation\ of\ Error = g'(x)$

Chain Rule

$$\frac{\partial g}{\partial x} = \frac{\partial g}{\partial f}\frac{\partial f}{\partial x} = g'(x) = g'(f(x)).f'(x)$$

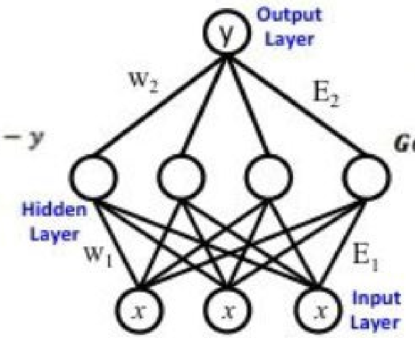
$$\frac{\partial f}{\partial x} = f'(x) = 2.\frac{E_2}{2} = E_2 = w_2x = 10$$

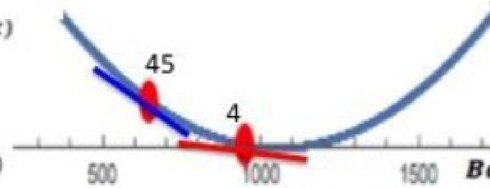
$$\frac{\partial g}{\partial x} = \frac{\partial g}{\partial f}\frac{\partial f}{\partial x} = g'(x) = g'(f(w_1x - y)).f'(x)$$

$$g'(x) = g'(f(3)).10$$

$$g'(x) = \frac{1}{2}.3^2.10 = \frac{90}{2} = 45$$

Derivative of error





Rubens Zimbres

After Weight Adjustment

Weight Adjustement

$Adjust\ w_2\ from\ 5\ to\ 4 \wedge w_1\ from\ 3.5\ to\ 2.5$

$Goal\ is\ to\ decrease\ derivative\ of\ error\ g'(x) \rightarrow 0$

$For\ w_2x = 8 \wedge w_1x = 5 \wedge y = 4$

$$f(x) = \frac{E_2^2}{2}$$

$$\frac{\partial f}{\partial x} = f'(x) = 2.\frac{E_2}{2} = E_2 = w_2x = 8$$

$$g(x) = \frac{E_1^2}{2}$$

$Backpropagation\ of\ Error = g'(x)$

$$\frac{\partial g}{\partial x} = \frac{\partial g}{\partial f}\frac{\partial f}{\partial x} = g'(x) = g'(f(x)).f'(x)$$

$$g'(x) = g'(f(w_1x - y)).f'(x)$$

$$g'(x) = g'(f(1)).8$$

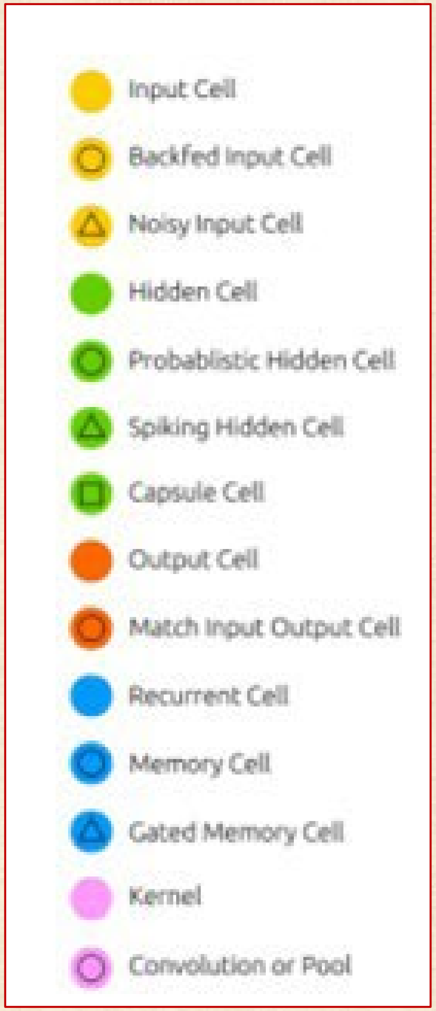
Derivative of error

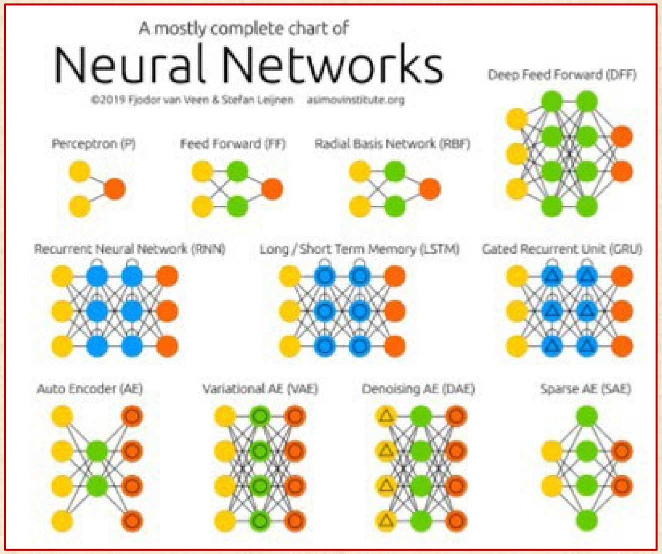
$$g'(x) = \frac{1}{2}.1^2.8 = \frac{8}{2} = 4$$

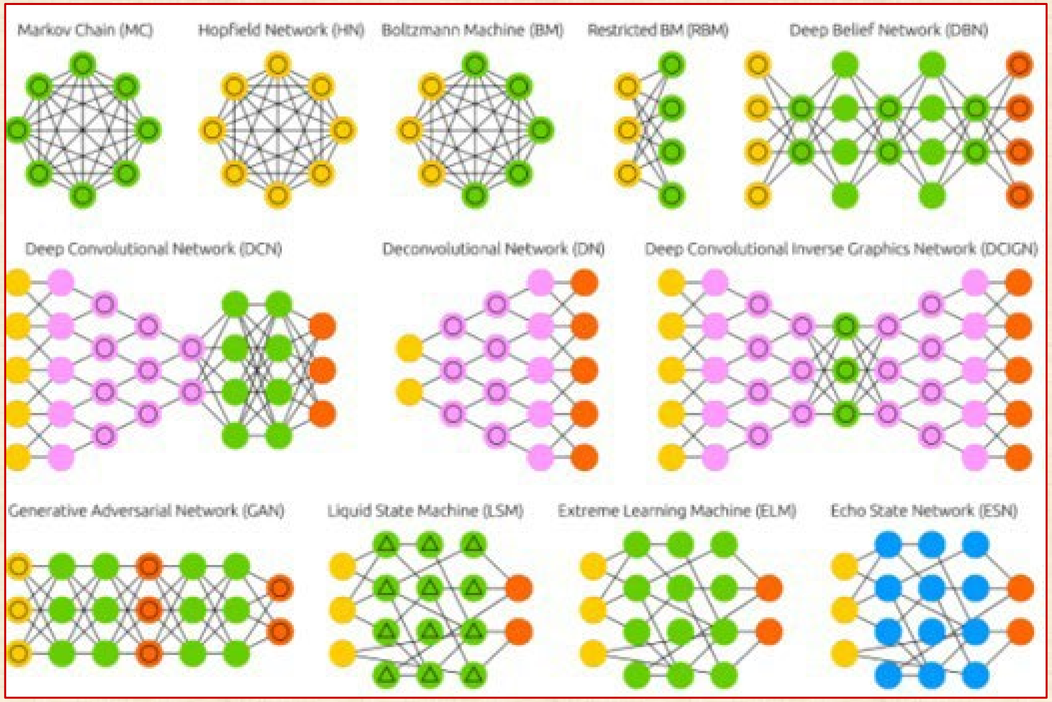
After Backprop

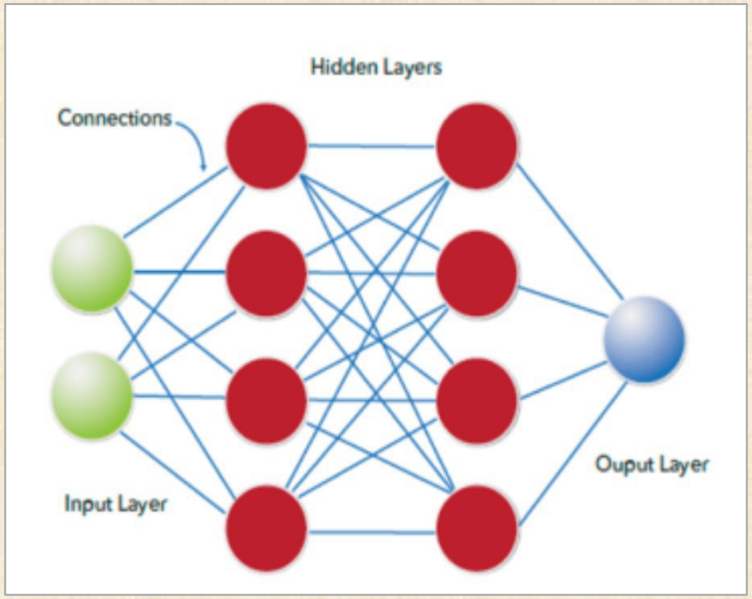
Images de réseaux de neuronnes

The Neural Network Zoo - The Asimov Institute









Réseaux Neurones Multicouches

STATISTICA
une seule couche interne

JMP Pro
1 ou 2 couches internes

Réseaux de neurones standard

- **Feedforward Neural Networks (FNN)**
- **Autoencoder Neural Networks (ANN)**

Réseaux employés en Intelligence Artificielle

- **Convolutional Neural Networks (CNN)**
- **Recurrent Neural Networks (RNN)**
- **Deep Neural Networks (DNN)**

Quelques références

Bishop, C. (1995). *Neural Networks for Pattern Recognition*. Oxford: University Press.

Carling, A. (1992). *Introducing Neural Networks*. Wilmslow, UK: Sigma Press.

Fausett, L. (1994). *Fundamentals of Neural Networks*. New York: Prentice Hall.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation*. New York: Macmillan Publishing.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59-69.

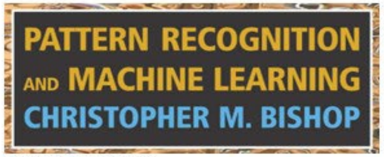
Patterson, D. (1996). *Artificial Neural Networks*. Singapore: Prentice Hall.

Ripley, B.D. (1996). *Pattern Recognition and Neural Networks*. Cambridge University Press.

Rumelhart, D.E., and J.L. McClelland (1986), *Parallel Distributed Processing*, Volume 1. The MIT Press. Foundations.

- Bishop-Pattern_Recognition_and_Machine_Learning.pdf
- JMP v15-Neural Network.pdf
- Shin-Réseaux Neurones.pdf
- Tsai-Real_World_Not_Normal.pdf
- Wikipedia-Artificial_Neural_Network.pdf
- Wikipedia-Johnson_s SU-distribution.pdf

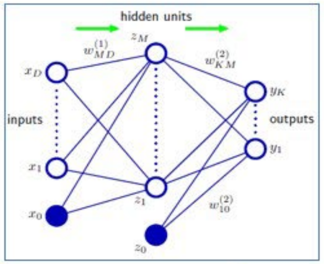
Quelques extraits de Bishop - Pattern Recognition



Springer 2006

Chapter 5

Neural Networks



$$y_k(\mathbf{x}, \mathbf{w}) = \sigma\left(\sum_{j=1}^{M} w_{kj}^{(2)} h\left(\sum_{i=1}^{D} w_{ji}^{(1)} x_i + w_{j0}^{(1)}\right) + w_{k0}^{(2)}\right)$$

$$y_k(\mathbf{x}, \mathbf{w}) = \sigma\left(\sum_{j=0}^{M} w_{kj}^{(2)} h\left(\sum_{i=0}^{D} w_{ji}^{(1)} x_i\right)\right).$$

$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N} \|\mathbf{y}(\mathbf{x}_n, \mathbf{w}) - \mathbf{t}_n\|^2.$$

$$p(t|\mathbf{x}, \mathbf{w}) = \mathcal{N}\left(t|y(\mathbf{x}, \mathbf{w}), \beta^{-1}\right)$$

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} p(t_n|\mathbf{x}_n, \mathbf{w}, \beta).$$

$$\frac{\beta}{2}\sum_{n=1}^{N}\{y(\mathbf{x}_n, \mathbf{w}) - t_n\}^2 - \frac{N}{2}\ln\beta + \frac{N}{2}\ln(2\pi)$$

$$E(\mathbf{w}) \simeq E(\widehat{\mathbf{w}}) + (\mathbf{w} - \widehat{\mathbf{w}})^{\mathrm{T}}\mathbf{b} + \frac{1}{2}(\mathbf{w} - \widehat{\mathbf{w}})^{\mathrm{T}}\mathbf{H}(\mathbf{w} - \widehat{\mathbf{w}})$$

$$\nabla E \simeq \mathbf{b} + \mathbf{H}(\mathbf{w} - \widehat{\mathbf{w}}).$$

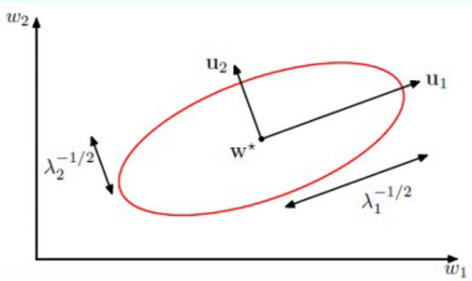
$$E(\mathbf{w}) = E(\mathbf{w}^\star) + \frac{1}{2}(\mathbf{w} - \mathbf{w}^\star)^{\mathrm{T}}\mathbf{H}(\mathbf{w} - \mathbf{w}^\star)$$

eigenvalue equation for the Hessian matrix

$$\mathbf{H}\mathbf{u}_i = \lambda_i \mathbf{u}_i$$

$$E(\mathbf{w}) = E(\mathbf{w}^\star) + \frac{1}{2}\sum_i \lambda_i \alpha_i^2.$$

In the neighbourhood of a minimum $\mathbf{w}^\star$, the error function can be approximated by a quadratic. Contours of constant error are then ellipses whose axes are aligned with the eigenvectors $\mathbf{u}_i$ of the Hessian matrix, with lengths that are inversely proportional to the square roots of the corresponding eigenvectors λ_i.



5.2.3 Use of gradient information

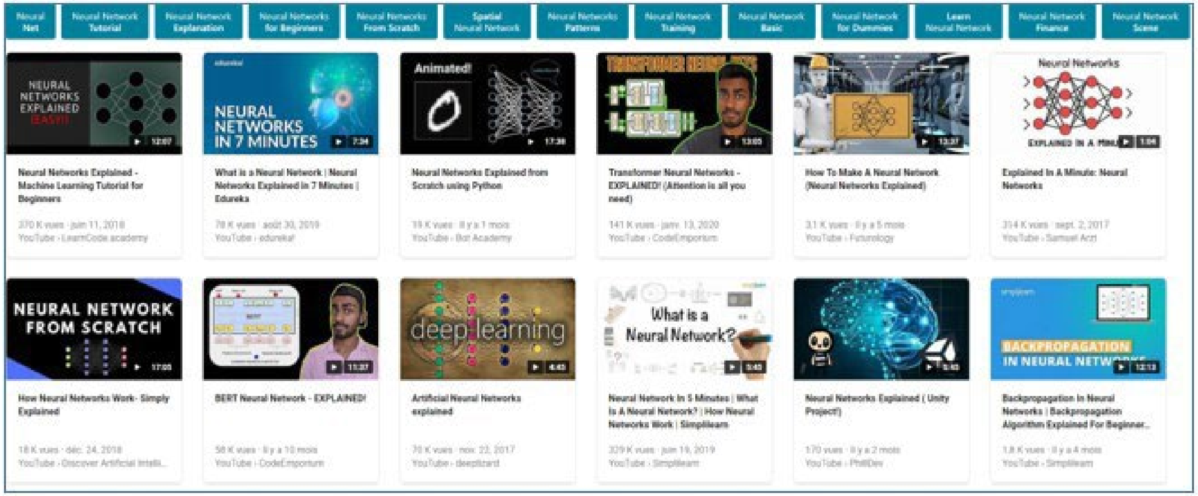
As we shall see in Section 5.3, it is possible to evaluate the gradient of an error function efficiently by means of the backpropagation procedure. The use of this gradient information can lead to significant improvements in the speed with which the minima of the error function can be located. We can see why this is so, as follows.

5.2.4 Gradient descent optimization

The simplest approach to using gradient information is to choose the weight update in (5.27) to comprise a small step in the direction of the negative gradient, so that

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta\nabla E(\mathbf{w}^{(\tau)}) \qquad (5.41)$$

Quelques vidéos sur les réseaux de neurones sur le WEB



[The Stilwell Brain - YouTube](https://www.youtube.com/watch?app=desktop&v=rA5qnZUXcqo) (26 minutes) :

https://www.youtube.com/watch?app=desktop&v=rA5qnZUXcqo

Très bon vidéo sur le fonctionnement du cerveau et les réseaux de neurones

Neural Networks Explained - Machine Learning Tutorial for Beginners - Bing video

Élémentaire

Backpropagation In Neural Networks | Backpropagation Algorithm Explained For Beginners | Simplilearn - Bing video

Explication des détails de l'optimisation des poids et l'algorithme de propagation arrière

The Stilwell Brain - YouTube

https://www.youtube.com/watch?app=desktop&v=rA5qnZUXcqo

Vidéos sur les réseaux de neuronnes

https://www.youtube.com/watch?v=aircAruvnKk

https://www.youtube.com/watch?v=IHZwWFHWa-w

https://www.patreon.com/3blue1brown

https://www.sas.com/en_us/insights/analytics/neural-networks.html

What is a Neural Network | Neural Networks Explained in 7 Minutes | Edureka - Bing video

Neural Networks Explained from Scratch using Python - Bing video

Transformer Neural Networks - EXPLAINED! (Attention is all you need) - Bing video

How To Make A Neural Network (Neural Networks Explained) - Bing video

How Neural Networks Work- Simply Explained - Bing video

Neural Network In 5 Minutes | What Is A Neural Network? | How Neural Networks Work | Simplilearn - Bing video

Weight Agnostic Neural Networks Explained! - Bing video

Vidéo de STATISTICA en français

avec le fichier Beverage Manufacturing.sta

35 variables X 2838 observations

disponible dans le répertoire examples dataset de Statistica

https://www.youtube.com/watch?v=0-PYXDYiN_c

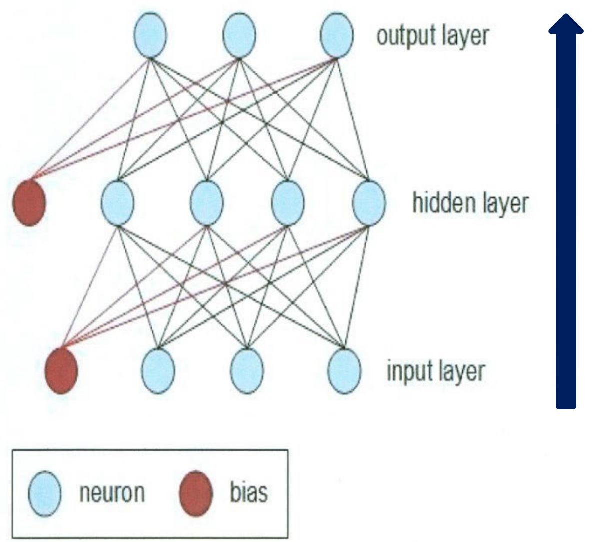
Types de réseaux de neurones

MLP Multi Layer Perceptron

SANN Overviews - Network Types

SANN Overviews - Network Types

The Multilayer Perceptron Neural Networks

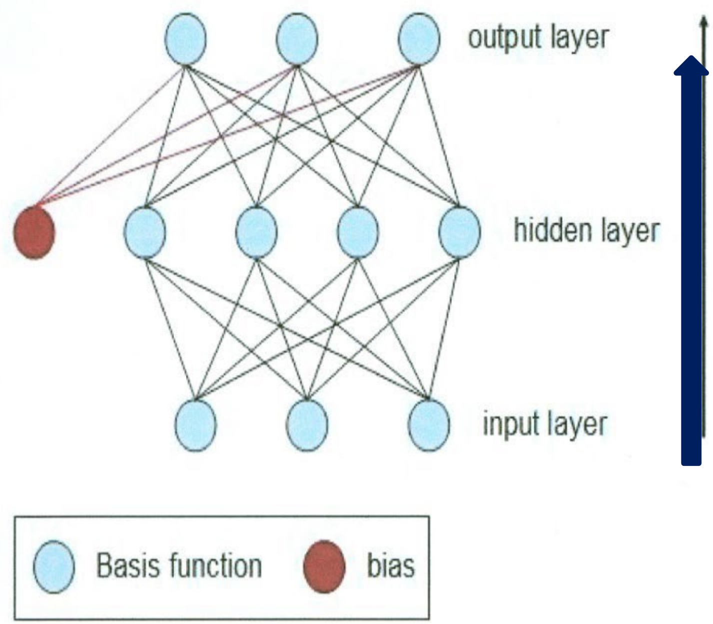


A schematic diagram of a fully connected MLP2 neural network with three inputs, four hidden units (neurons), and three outputs. Note that the hidden and output layers have a bias term. Bias is a neuron that emits a signal with strength 1.

type de réseau le plus employé

RBF Radial Basis Function

The Radial Basis Function Neural Networks



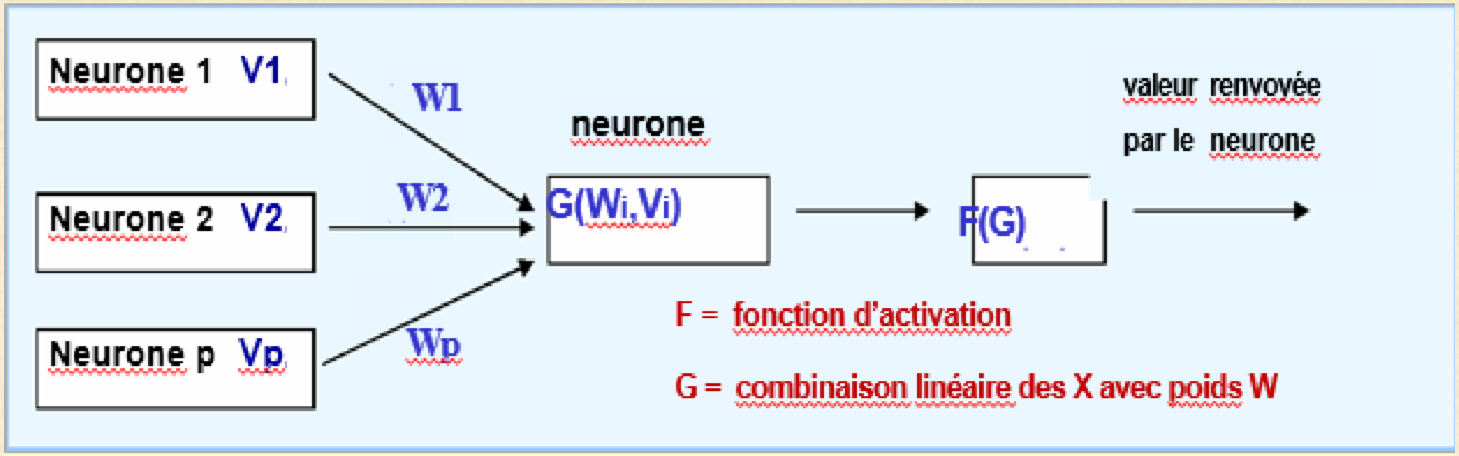
A schematic diagram of an RBF neural network with three inputs, four radial basis functions and 3 outputs. Note that, in contrast to MLP networks, it is only the output units that have a bias term.

type de réseau pas très recommandé pour des Inputs X catégoriques

Source : Statistica Electronic Manual

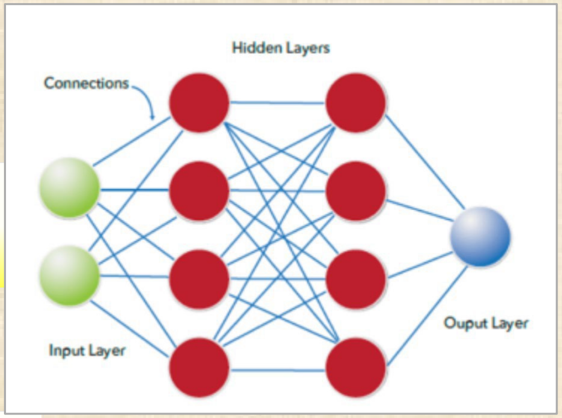
Élements des réseaux de neurones

Perceptron Multi Couches



- Les solutions ne sont pas calculées directement :
  un algorithme d'optimisation calcule itérativement les solutions
- Le modèle est construit sur une base d'apprentissage et testé à chaque étape sur une base de validation
- L'apprentissage du réseau s'effectue en mode supervisé :
  les poids W_i et les valeurs V_i synaptiques des neurones de chaque couche (cachée) sont évalués pour minimiser une fonction d'erreurs
- Les entrées (variables explicatives) et sorties (variables à expliquer) du réseau sont connues

Éléments des réseaux de neurones



Interprétation des modèles et des traitements

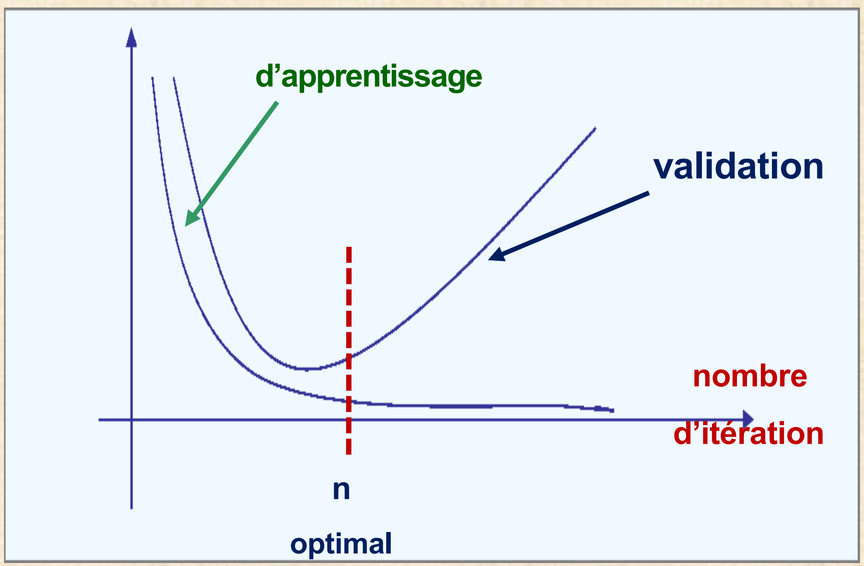
- La valeur Y_j d'un individu j est une fonction composée des valeurs prises par les X_i (p variables)

- Des fonctions d'activations liste - page suivante

- Des poids W (équivalents au beta) associés aux neurones

Évaluation complexe + présence de colinéarités (variables, poids, nombre de couches, nombre de neurones) => éviter le « sur apprentissage »

Erreur



Erreur : Sum Of Square

$= E_{SOS} = \sum (y_i - t_i)^2$

t_i continue

Erreur : Cross Entropy

$= E_{CE} = -\sum t_i \ln(y_i / t_i)$

t_i catégorique

y_i : prédictions t_i : target

Algorithmes

- BFGS : Broyden-Fletcher-Goldfarb-Shanno
- SCG : Scaled Conjugate Gradient

Fonctions d'activation

Source : Statistica Electronic Manual

Function	Definition	Description	Range
Identity	a	The activation of the neuron is passed on directly as the output	$(-\infty,+\infty)$
Logistic sigmoid	$\frac{1}{1+e^{-a}}$	An S-shaped curve	$(0,1)$
Hyperbolic tangent	$\frac{e^{a}-e^{-a}}{e^{a}+e^{-a}}$	A sigmoid curve similar to the logistic function. Often performs better than the logistic function because of its symmetry. Ideal for multilayer perceptrons, particularly the hidden layers	$(-1,+1)$
Exponential	e^{-a}	The negative exponential function	$(0,+\infty)$
Sine	$\sin(a)$	Possibly useful if recognizing radially distributed data. Not used by default	$[0,1]$
Softmax	$\frac{\exp(a_i)}{\sum \exp(a_i)}$	Mainly used for (but not restricted to) classification tasks. Useful for constructing neural networks with normalized multiple outputs which makes it particularly suitable for creating neural network classifiers with probabilistic outputs.	$[0,1]$
Gaussian	$\frac{1}{\sqrt{2\pi}\sigma}\exp\left[\frac{(x-\mu)^2}{2\sigma^2}\right]$	This type of isotropic Gaussian activation function is solely used by the hidden units of an RBF neural network which are also known as radial basis functions. The location (also known as prototype vectors) and spread parameters are equivalent to the input-hidden layer weights of an MLP	

ajout : fonction RuL très populaire : voir document MARS

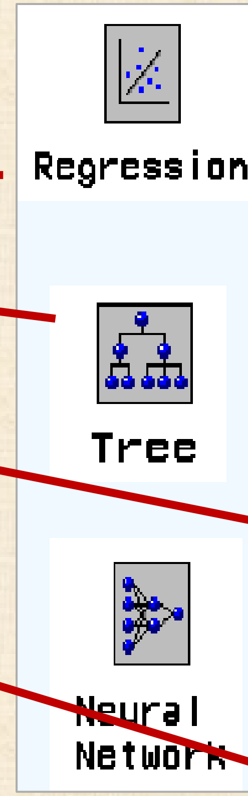
MÉTHODES & OUTILS

STATISTICA

- Data Miner Recipes
- General Classification/Regression Tree Models
- General CHAID Models
- Interactive Trees (C&RT, CHAID)
- Boosted Tree Classifiers and Regression
- Random Forests for Regression and Classification
- Generalized Additive Models
- MARSplines (Multivariate Adaptive Regression Splines)
- Generalized EM & k-Means Cluster Analysis
- Automated Neural Networks
- Machine Learning (Bayesian, Support Vectors, K-Nearest)
- Independent Components Analysis
- Text & Document Mining
- Web Crawling, Document Retrieval
- Association Rules
- Sequence, Association, and Link Analysis
- Rapid Deployment of Predictive Models (PMML)
- Goodness of Fit, Classification, Prediction
- Feature Selection and Variable Screening
- Optimal Binning for Predictive Data Mining
- Data Mining - Workspaces
- Process Optimization

méthodes vues dans ces notes

StatSoft
STATISTICA



4 types

- Linéaire
- Logistique
- Multinomiale
- Généralisée

CRT Classification Regression Tree

arbres de décision

MARS Multivariate Adaptive Regression Splines

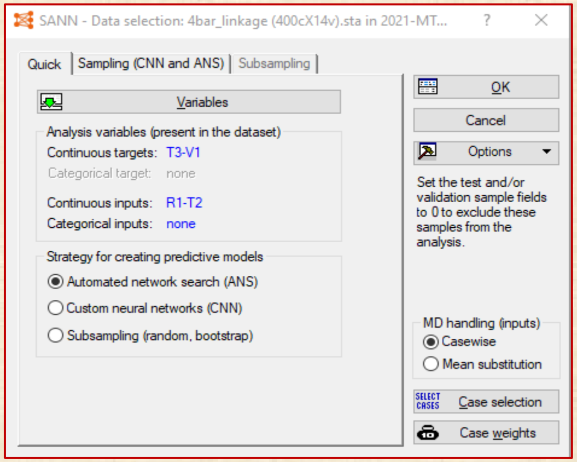
ANN Artificial Neural Networrk

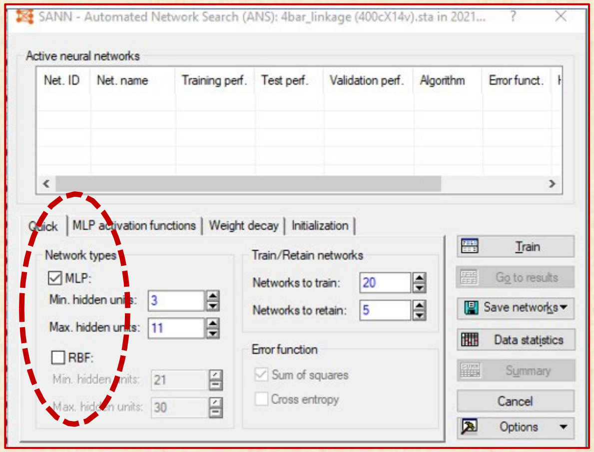
réseau de neurones

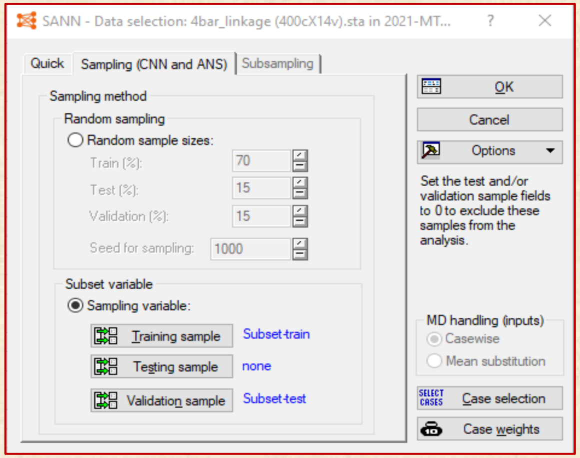
Autres

ACP, PLS, Bootstrap, Forêts aléatoires, SVM …

STATISTICA : commandes procédure ANN



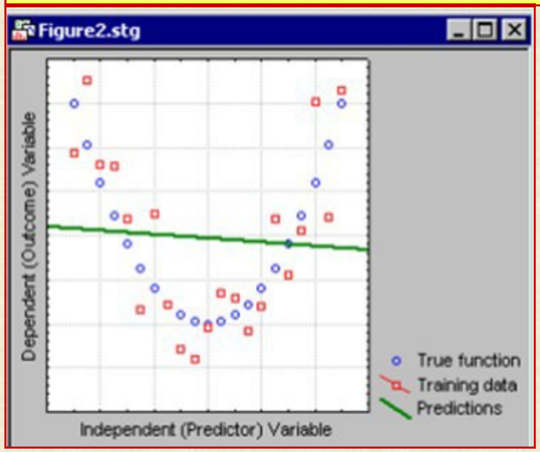


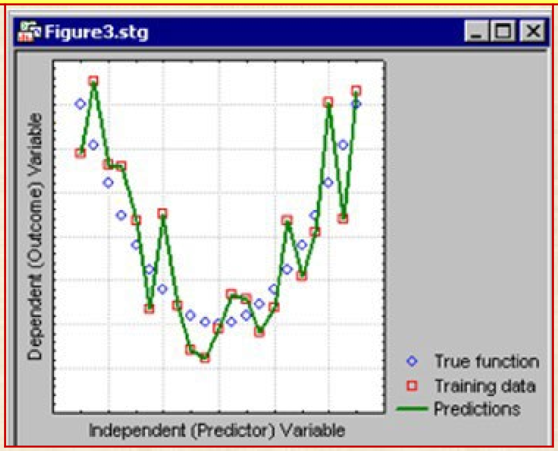


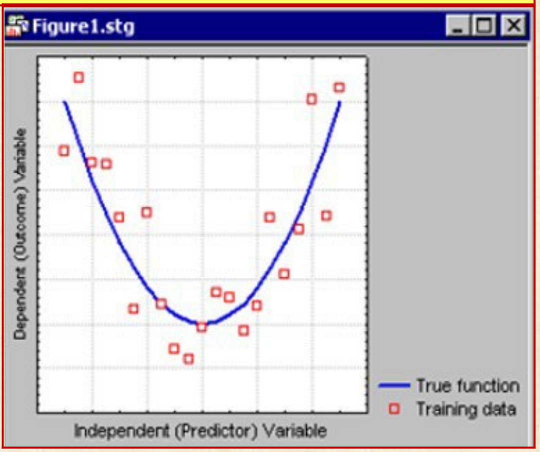
Generalization **major concern** when training neural networks when training data have been **overfit (= fitting the random noise)** then difficult to make accurate predictions using new data

To avoid overfitting cross-validation / v-fold cross-validation

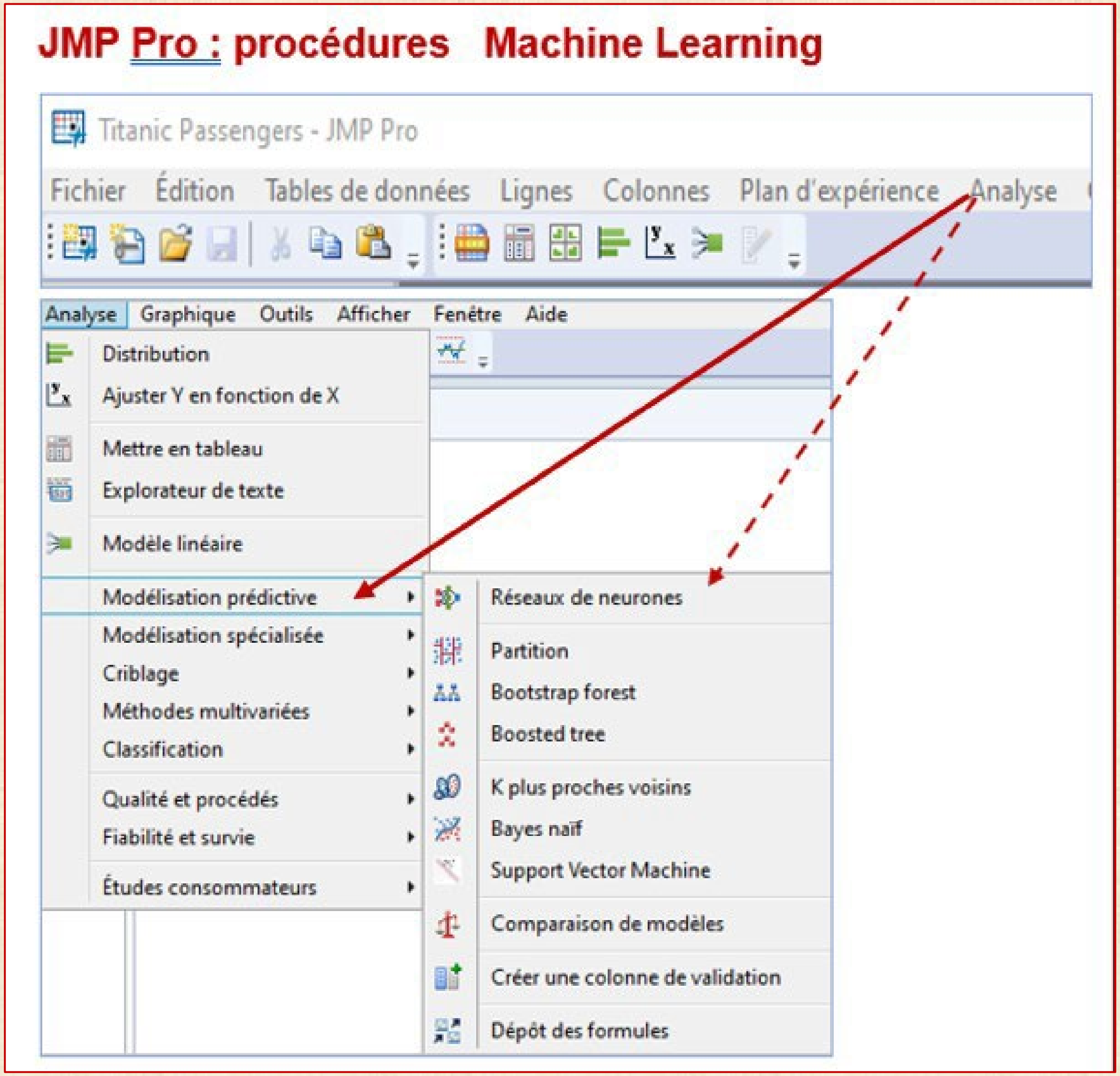
UNDERFFING	OVERFITTING	GOOD FITTING







JMP Pro : procédure réseaux neuronnes

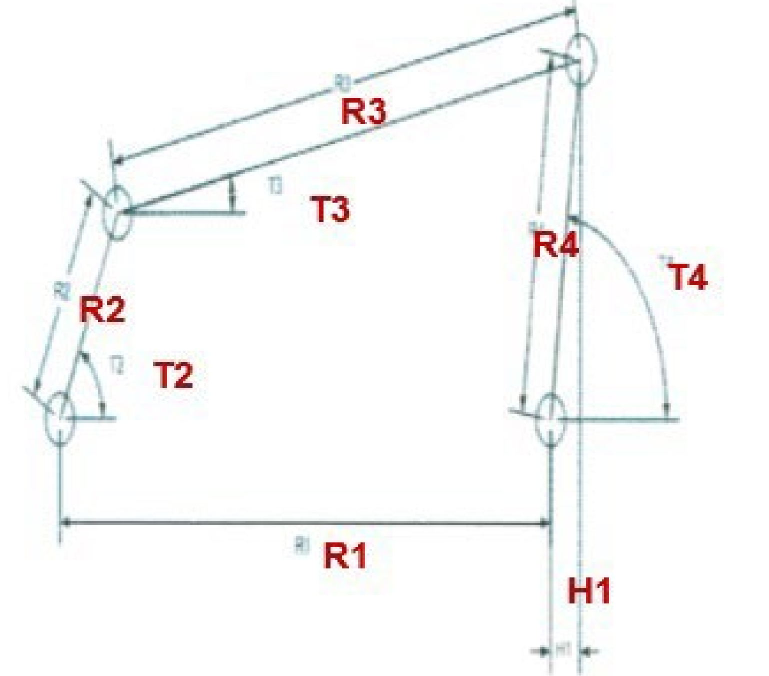


Exemple 1 : « 4-bar linkage »

Theory of Machine and Mechanisms, Shigley and Ulcker, p. 57-59

But création d'un réseau de neurones décrivant les **relations non linéaires** entre les angles **T** (targets) et les barres **R** (variables d'entrée) qui contrôlent le système mécanique

source : Statistica Electronic Manual



Input : R1, R2, R3, R4, T2 **intermédiaire :** γ

$$\gamma = \cos^{-1}\left(\frac{R_3^2+R_4^2-R_1^2-R_2^2+2R_1R_2\cos(T_2)}{2R_3R_4}\right)$$

Target (output): T3, T4, H1, V1

$$T_3 = 2\tan^{-1}\frac{-R_2\sin(T_2)\pm R_4\sin(\gamma)}{R_3+R_1-R_2\cos(T_2)-R_4\cos(\gamma)}$$

$$T_4 = 2\tan^{-1}\frac{R_2\sin(T_2)\mp R_3\sin(\gamma)}{R_4-R_1+R_2\cos(T_2)-R_3\cos(\gamma)}$$

idNum	R1	R2	R3	R4	T2	gamma	T3	T4	H1	V1	Subset
1	8,515	2,211	7,549	6,202	-1,222	1,234	1,078	2,312	-4,189	4,573	Train
2	8,580	2,694	7,530	5,716	2,433	1,885	0,366	2,251	-3,593	4,446	Train
6	9,418	2,960	7,529	5,806	-4,268	1,934	0,270	2,204	-3,435	4,681	Train
.	.	.	.	.	.	.	.	.	.	.	.
398	9,127	2,034	7,957	5,950	6,138	1,039	0,846	1,884	-1,836	5,660	Validation
399	9,205	2,117	8,329	5,920	1,355	1,330	0,462	1,791	-1,294	5,777	Validation
400	8,794	2,287	7,841	6,045	-3,803	1,743	0,459	2,202	-3,569	4,879	Validation

$$H_1 = R_4\cos(T_4)$$

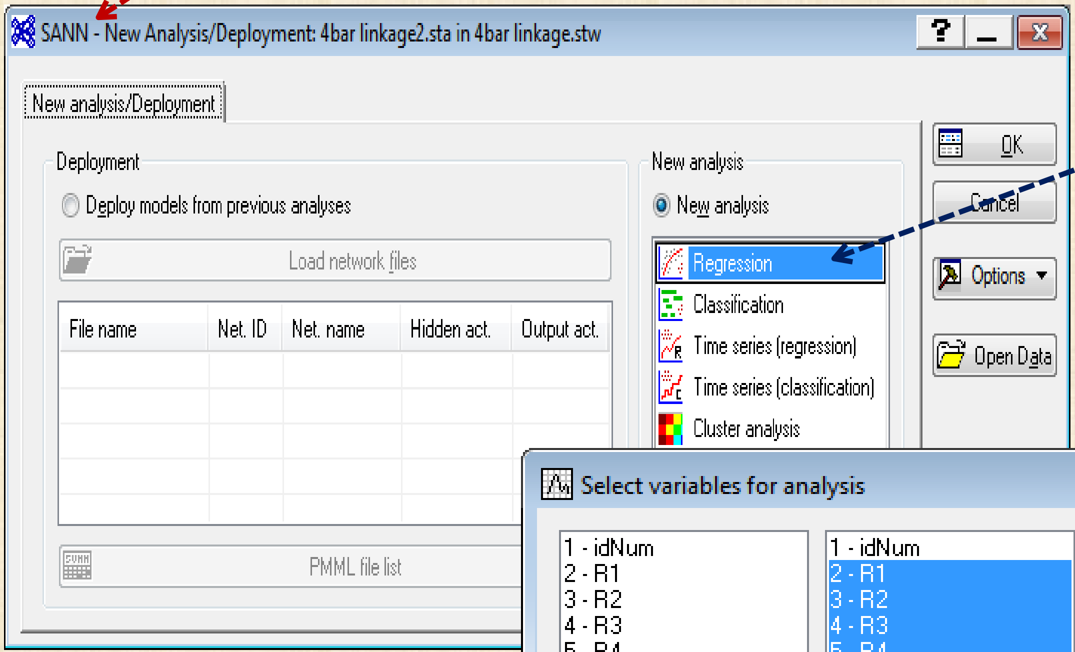
$$V_1 = R_4\sin(T_4)$$

on connait les réponses !

Copyright © Génistat Conseils Inc., 2021, Montréal, Canada

Exemple 1 : dynamique « 4-bar linkage »

Barre d'outils DATA MINING

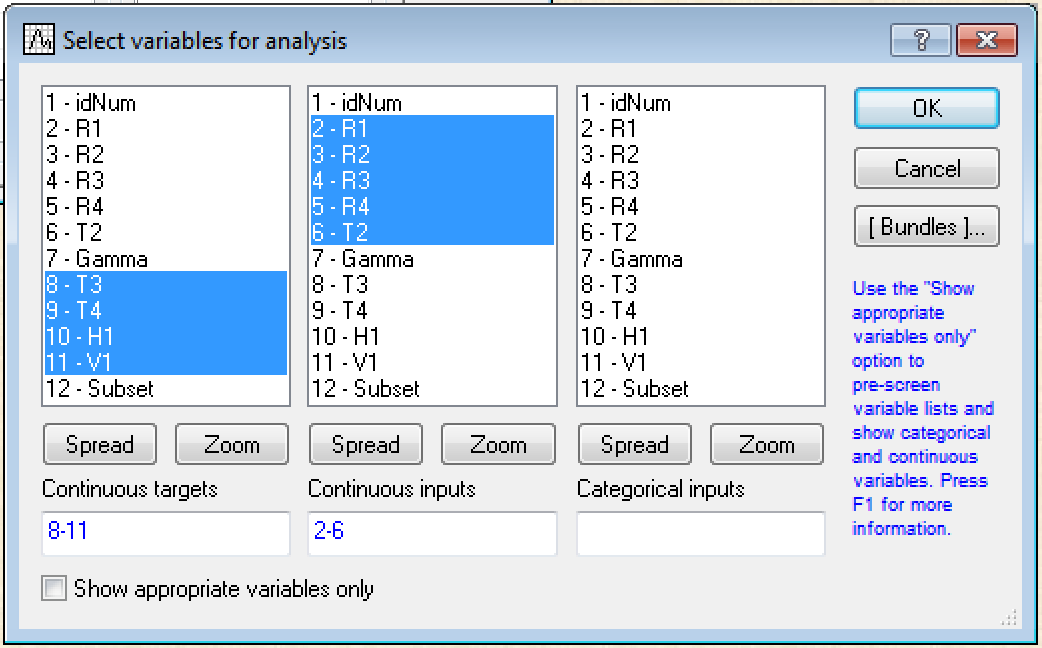


Régression:

entrées continues

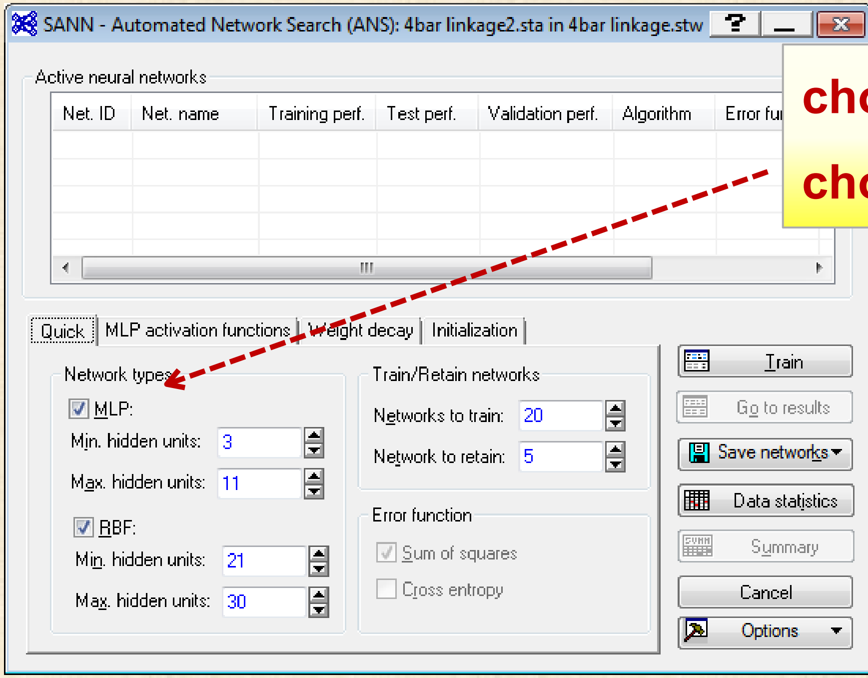
&

sorties continues



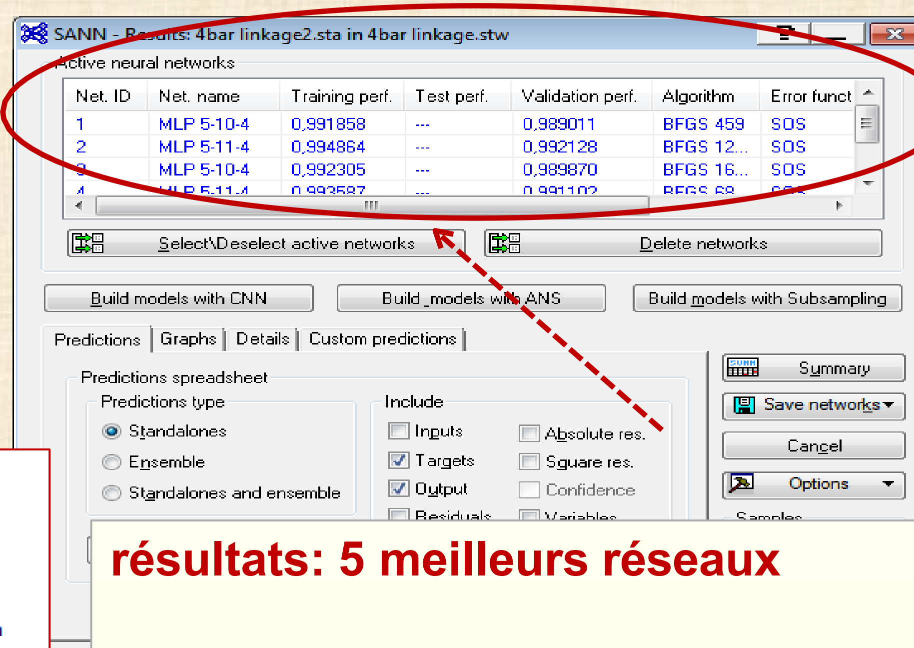
choix des variables

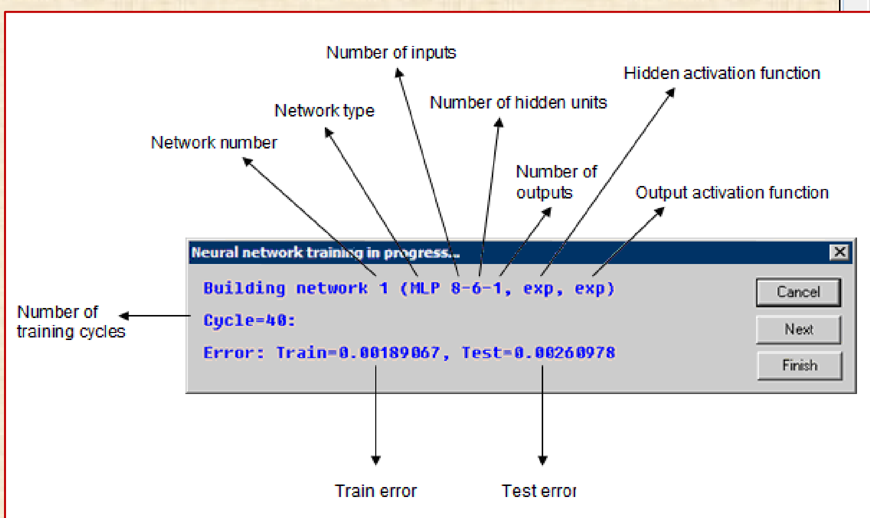
Exemple 1 : dynamique « 4-bar linkage »



choix de de l'architecture : MLP ou RBF

choix implicite (par défaut)





résultats: 5 meilleurs réseaux

Name	Train Perf.	Valid. Perf.	Algo.	Error Funct.
MLP 5-10-4	0,99	0,989	BFGS	SOS
...	...	...	...	...

Exemple 1 : dynamique « 4-bar linkage »

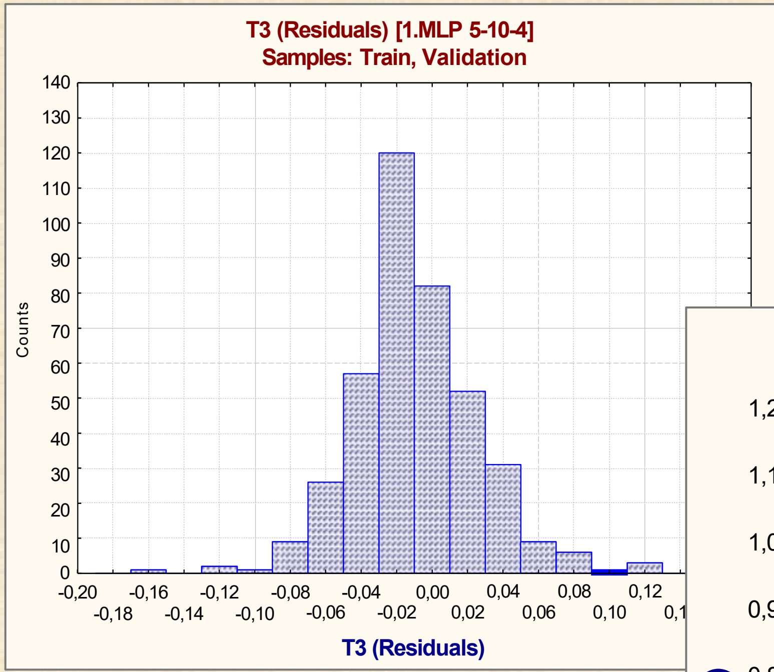
Summary of active networks (4bar linkage2.sta) - 5 meilleurs retenus

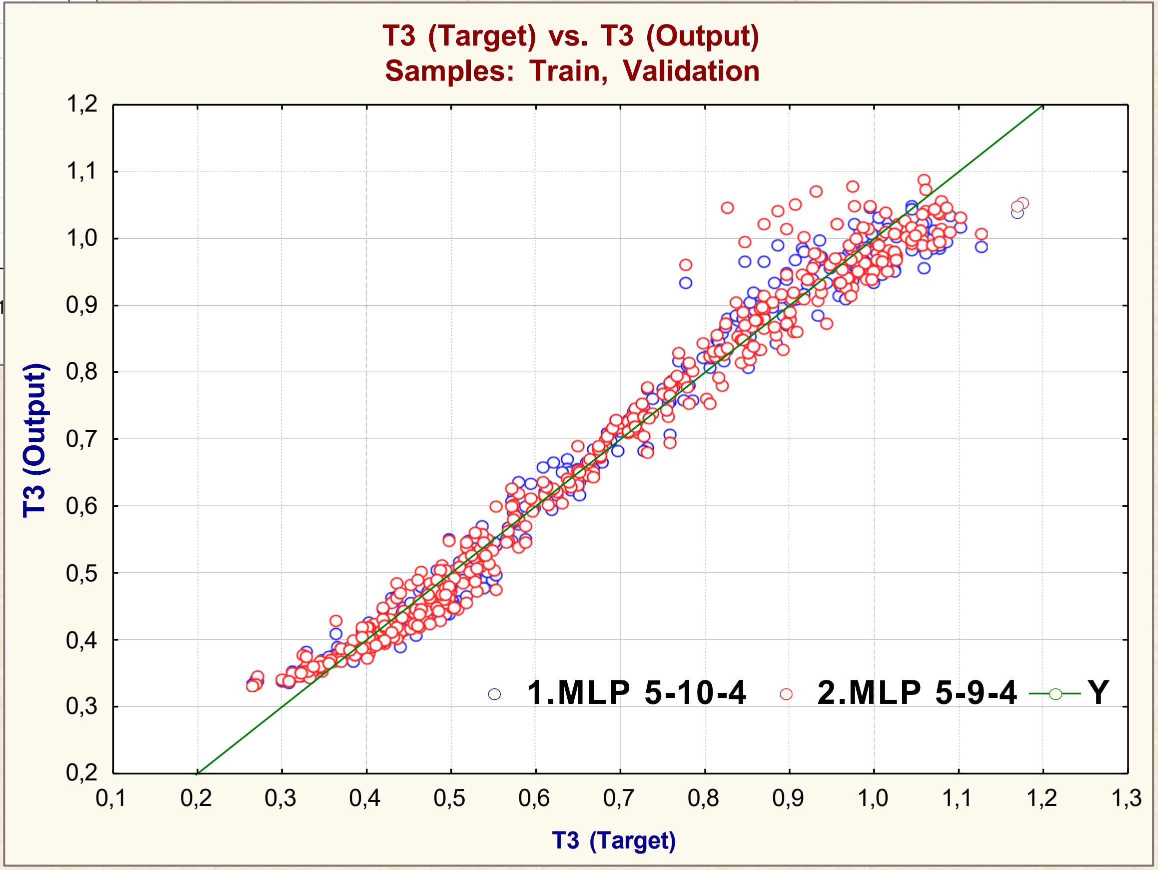
Index	Net. name	Training perf.	Test perf.	Validation perf.	Training error	Test error	Validation error	Training algorithm	Error function	Hidden activation	Output activation
1	MLP 5-10-4	0,993		0,988	0,024223		0,044244	BFGS 10000	SOS	Tanh	Logistic
2	MLP 5-9-4	0,992		0,988	0,024860		0,044657	BFGS 4327	SOS	Logistic	Logistic
3	MLP 5-9-4	0,993		0,991	0,029366		0,036294	BFGS 10000	SOS	Logistic	Identity
4	MLP 5-10-4	0,991		0,986	0,025774		0,042961	BFGS 5398	SOS	Logistic	Logistic
5	MLP 5-10-4	0,994		0,992	0,026699		0,041834	BFGS 10000	SOS	Tanh	Identity

Predictions for T3 (4bar linkage2.sta) Samples: Train, Validation

	Sample	T3 - Target	T3 – Output 1. MLP 5-10-4	T3 – Output 2. MLP 5-9-4	T3 – Output Ensemble
1	Train	1,078214	1,042669	1,055345	1,049007
2	Train	0,365675	0,389180	0,378669	0,383925
3	Train	0,270116	0,338447	0,345377	0,341912
4	Train	0,495735	0,550709	0,548063	0,549386
.	.	.	.	.	.
398	Validation	0,845572	0,965174	0,995354	0,980264
399	Validation	0,461604	0,444103	0,438404	0,441253
400	Validation	0,459103	0,429583	0,422225	0,425904

Exemple 1 : dynamique « 4-bar linkage »





Exemple 1 : dynamique « 4-bar linkage »

Network weights : réseau 1 et réseau 2				
	Connections 1. MLP 5-10-4	Weight values 1. MLP 5-10-4	Connections 2. MLP 5-9-4	Weight values 2. MLP 5-9-4
1	R1 --> hidden neuron 1	0,0019	R1 --> hidden neuron 1	-0,036
2	R1 --> hidden neuron 2	0,0444	R1 --> hidden neuron 2	-0,500
3	R1 --> hidden neuron 3	0,0155	R1 --> hidden neuron 3	0,119
4	R1 --> hidden neuron 4	-0,0848	R1 --> hidden neuron 4	-0,104
5	R1 --> hidden neuron 5	1,2969	R1 --> hidden neuron 5	8,558
.	.	.	.	.
90	hidden neuron 8 --> T4	-13,9466	hidden neuron 9 --> V1	-42,514
91	hidden neuron 8 --> H1	7,2849	hidden bias --> T3	-309,194
92	hidden neuron 8 --> V1	37,8808	hidden bias --> T4	-89,269
93	hidden neuron 9 --> T3	22,7443	hidden bias --> H1	132,587
94	hidden neuron 9 --> T4	-14,7457	hidden bias --> V1	44,763
.	.	.		
97	hidden neuron 10 --> T3	4,1392		
98	hidden neuron 10 --> T4	21,8987		
99	hidden neuron 10 --> H1	28,3003		
100	hidden neuron 10 --> V1	-12,9366		
101	hidden bias --> T3	0,4641		
102	hidden bias --> T4	-0,1723		
103	hidden bias --> H1	-3,1272		
104	hidden bias --> V1	-14,1290		

Exemple 1 : dynamique « 4-bar linkage »

Sensitivity analysis Samples: Train, Validation					
Modèle RN	T2	R3	R4	R1	R2
1.MLP 5-10-4	144,86	2,75	2,60	1,97	1,86
2. MLP 5-9-4	137,95	2,69	2,50	1,96	1,93
Average	141,41	2,72	2,55	1,96	1,90

Pointwise sensitivity analysis for T3
Network: 1.MLP 5-10-4

Output	grid point	R1 Sensitivity	R2 Sensitivity	R3 Sensitivity	R4 Sensitivity	T2 Sensitivity
T3	min	-0,068	-0,111	-0,058	0,183	-0,378
T3	2	-0,068	-0,104	-0,058	0,184	-0,139
T3	3	-0,068	-0,095	-0,058	0,185	0,076
T3	4	-0,067	-0,085	-0,058	0,185	0,386
T3	5	-0,067	-0,072	-0,057	0,184	0,030
T3	6	-0,066	-0,059	-0,057	0,183	-0,318
T3	7	-0,066	-0,043	-0,057	0,181	-0,159
T3	8	-0,065	-0,027	-0,057	0,179	0,069
T3	9	-0,064	-0,009	-0,057	0,176	0,374
T3	max	-0,064	0,009	-0,057	0,172	0,067

Analyse de sensibilité GLOBALE

data set is submitted to the network repeatedly with each variable in turn replaced with its mean value calculated from the training sample, and the resulting network error is recorded.

If an important variable changed in this fashion, the error would increase a great deal;
if an unimportant variable is removed, the error will not increase very much.

importance relative des variables d'input sur un modèle particulier de neurones particulier calcul fait sur des localisations arbitraires dans l'espace des variables d'entrée

- **T2 est la plus importante**
- **T2 est 70 fois plus importante que les autres**

Analyse de sensibilité LOCALE

semblable à sensibilité globale sauf

calcul des dérivées partielles (= sensibilité) fait en 10 points locaux de l'espace également espacés entre le min et le max

on fait varier légèrement les variables autour de ces 10 points

- **R4 et T2 sont plus importantes**
- **autres variables: importance égale mais relativement faible**

Exemple 1 : dynamique « 4-bar linkage »

Code en C / C++ pour déploiement

```
//Analysis Type - Regression
#include <stdio.h>
#include <conio.h>
#include <math.h>
#include <stdlib.h>
double 4bar_linkage2_sta_in_RéseauxNeurones_1_MLP_5_10_4_input_hidden_weights[10][5]=
{
 {2.02837435462445e-003, -9.19300736782718e-002, -8.33394542388945e-003, 5.96815120368463e-003, -9.72173081422972e-001 },
 {-4.10541254965628e+001, -1.57048702434030e+001, 7.24298697308776e+000, 7.54136136116508e-001, 7.23575930276387e+000
 },
 {-4.27953674542435e-003, 5.22558713365225e-002, -2.02630063174860e-002, 3.55526806539767e-001, 4.38086865729618e-001 },
 {8.36988375566669e-003, 6.51471505482633e-002, -1.87374240864233e-005, -2.93668874571122e-002, 4.28066974047903e+000 },
 {-4.51758787016232e-003, -1.65295098364779e-002, -9.55504020044409e-003, 3.37743795686801e-002, -3.92503675526061e+000 },
 {8.09715832028303e-003, 1.40380167160264e-001, -5.11574576386343e-002, 4.63819910138810e-002, -4.14377599772541e+000 },
 {2.64573937244475e-002, 1.38171451159571e-001, -5.05816891278269e-002, 3.03640534484363e-002, -4.99478506036909e+000 },
 {3.56819229312110e+001, -1.44427879618125e+001, 5.05330645369423e+000, 6.79476064151221e+000, -1.20589227870316e+001
 },
 {1.63237325776004e-002, 9.93417867044676e-002, -4.79318114375762e-003, -2.87432514207203e-002, 3.81634117448330e+000 },
 {5.06016857284542e-003, -6.65190232893867e-003, 1.98066912357954e-002, -3.29635020222122e-002, 3.65513674103006e+000 }
};
```

etc

Exemple 1 : dynamique « 4-bar linkage »

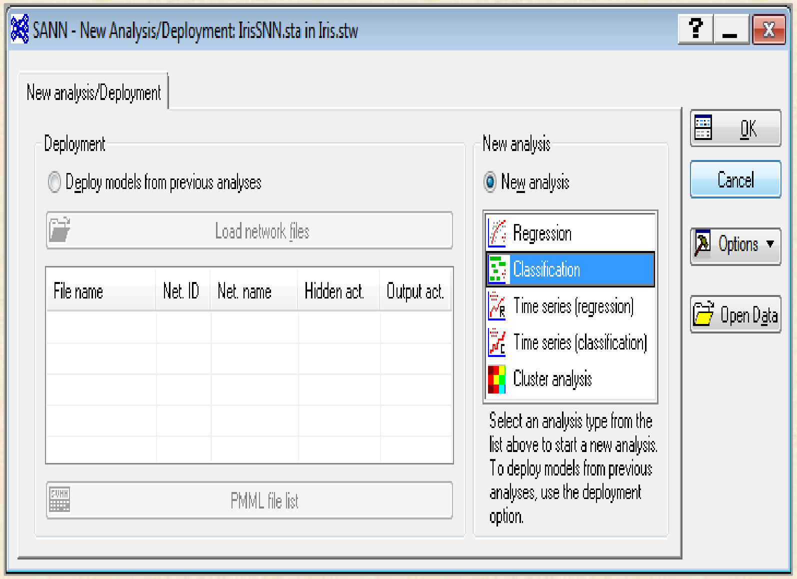
```xml
<?xml version="1.0" encoding="UTF-8" ?>
- <PMML version="3.0">
- <Header copyright="Copyright (c) StatSoft, Inc. All Rights Reserved.">
  <Application name="STATISTICA Automated Neural Networks (SANN)"
version="2.0" />
  </Header>
- <DataDictionary numberOfFields="9">
  <DataField name="T3" optype="continuous" />
  <DataField name="T4" optype="continuous" />
  <DataField name="H1" optype="continuous" />
  <DataField name="V1" optype="continuous" />
  <DataField name="R1" optype="continuous" />
  <DataField name="R2" optype="continuous" />
  <DataField name="R3" optype="continuous" />
  <DataField name="R4" optype="continuous" />
  <DataField name="T2" optype="continuous" />
  </DataDictionary>
- <NeuralNetwork modelName="4bar linkage2.sta in 4 l
functionName="regression">
- <MiningSchema>
  <MiningField name="T3" usageType="predicted" />
  <MiningField name="T4" usageType="predicted" />
  <MiningField name="H1" usageType="predicted" />
  <MiningField name="V1" usageType="predicted" />
  <MiningField name="R1" lowValue="8.502734" highValue="9.499846" />
  <MiningField name="R2" lowValue="2.008976" highValue="2.996489" />
  <MiningField name="R3" lowValue="7.503913" highValue="8.496991" />
  <MiningField name="R4" lowValue="5.502913" highValue="6.498295" />
  <MiningField name="T2" lowValue="-6.180893" highValue="6.274329" />
  </MiningSchema>
- <NeuralInputs numberOfInputs="5">
- <NeuralInput id="0">
-<DerivedField>

- Etc….
```

Predictive Model Markup Language (PMML) is an XML-based language that allows for the efficient exchange of (trained) predictive models and shared models between different applications (e.g., *STATISTICA* and *WebSTATISTICA*). A PMML document usually contains information describing fully trained or parameterized analytic models so that they can be readily deployed (applied to new cases) by another application. PMML documents can be saved from practically all methods available in *STATISTICA* for prediction and predictive classification, and they are used extensively in the context of *STATISTICA Data Miner*

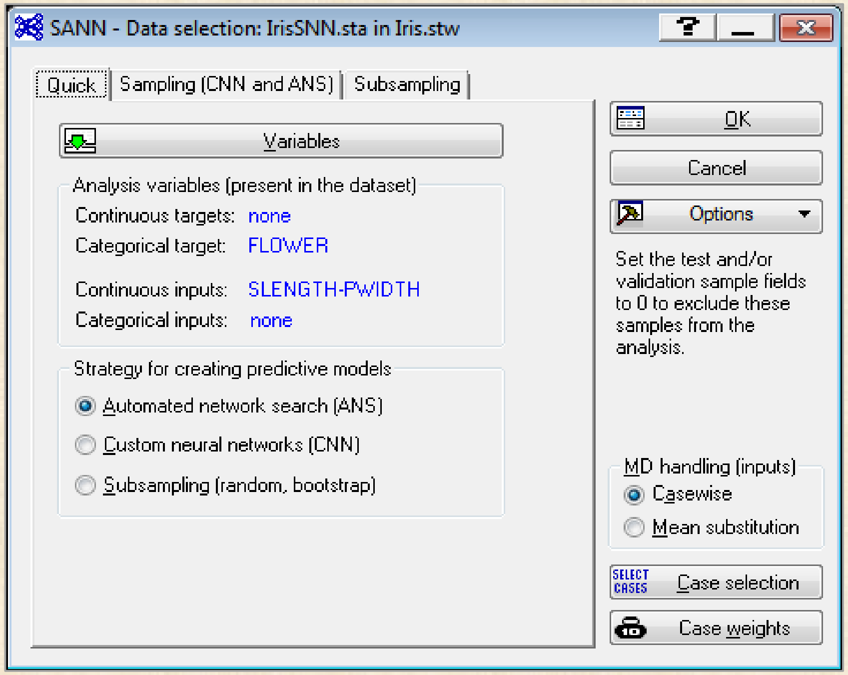
Exemple 2 : classification – données = iris

idNum	NNSET	SLENGTH	SWIDTH	PLENGTH	PWIDTH	FLOWER
1	Select	5,1	3,5	1,4	0,2	Setosa
2	Select	4,9	3,0	1,4	0,2	Setosa
3	Train	4,7	3,2	1,3	0,2	Setosa
.	.	.	.	.	.	.
51	Train	7,0	3,2	4,7	1,4	Versicol
52	Select	6,4	3,2	4,5	1,5	Versicol
53	Train	6,9	3,1	4,9	1,5	Versicol
.	.	.	.	.	.	.
101	Select	6,3	3,3	6,0	2,5	Virginic
102	Select	5,8	2,7	5,1	1,9	Virginic
.	Select	7,1	3,0	5,9	2,1	Virginic
.	.	.	.	.	.	.
150	Train	5,9	3,0	5,1	1,8	Virginic

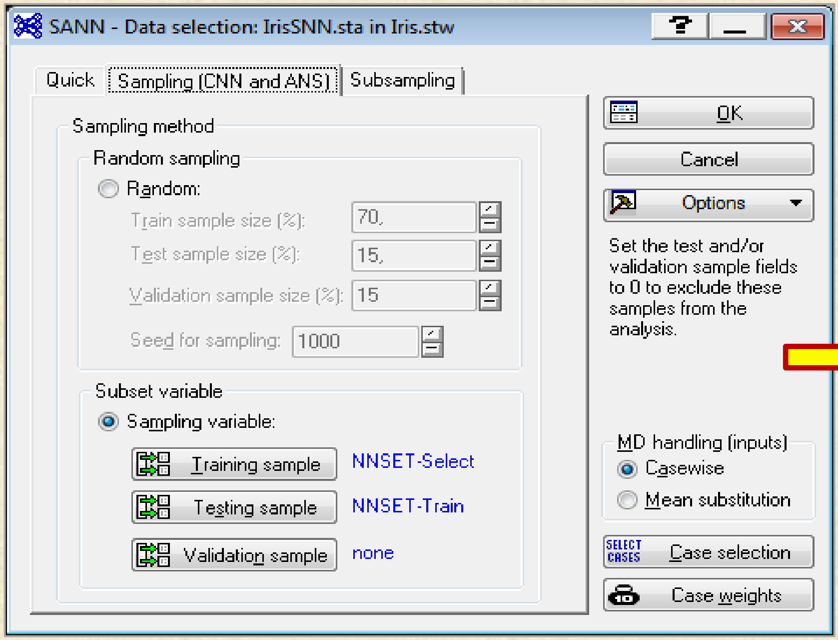


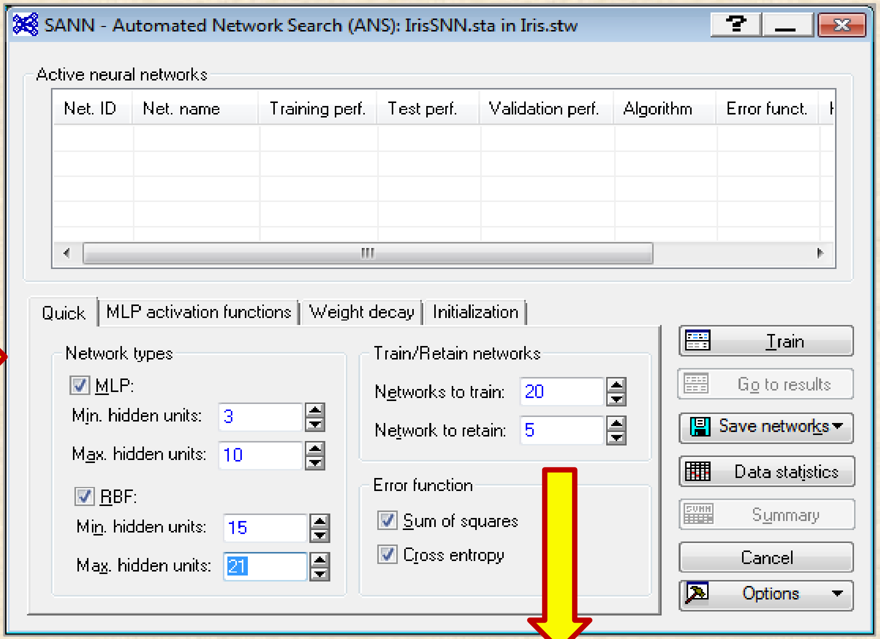
Summary Frequency Table
Marked Cells have counts > 10

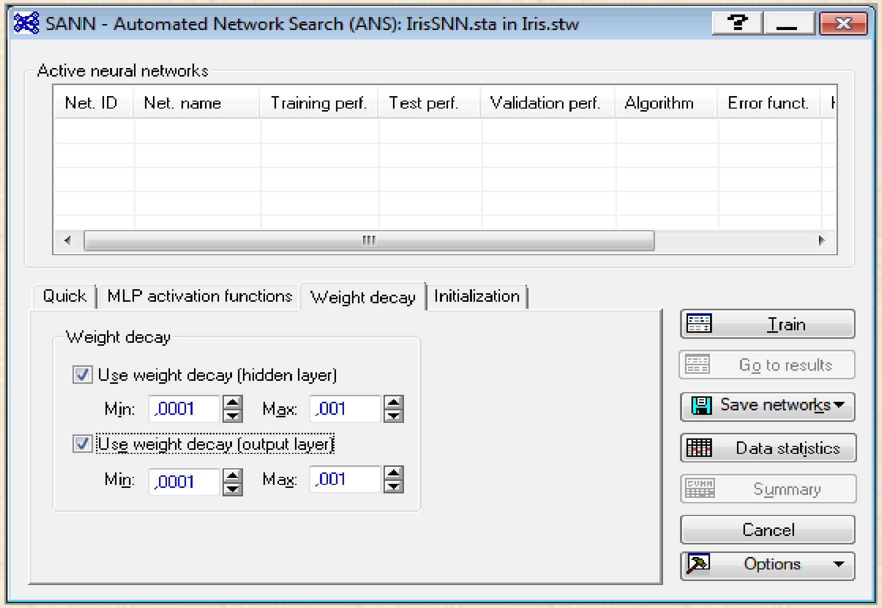
	NNSET	FLOWER Setosa	FLOWER Versicol	FLOWER Virginic	Row Totals
	Select	23	24	23	70
	Train	27	26	27	80
	All Grps	50	50	50	150



Exemple 2 : classification – données = iris







Exemple 2 : classification – données = iris

Summary of active networks								
	Net. name	Training perf.	Test perf.	Validati on perf.	Training algorithm	Error function	Hidden activation	Output activation
	21. MLP 4-3-3	95,71	97,50		BFGS 24	SOS	Logistic	Tanh
	22. RBF 4-21-3	94,29	98,75		RBFT	SOS	Gaussian	Identity
	23. MLP 4-3-3	95,71	97,50		BFGS 21	SOS	Exponential	Identity
	24. RBF 4-21-3	100,00	98,75		RBFT	Entropy	Gaussian	Softmax
	25. MLP 4-3-3	65,71	67,50		BFGS 7	Entropy	Sine	Softmax

Predictions for FLOWER Samples: Train						
	FLOWER-Target	FLOWER – Output 21. MLP 4-3-3	FLOWER – Output 22. RBF 4-21-3	FLOWER – Output 23. MLP 4-3-3	FLOWER – Output 24. RBF 4-21-3	FLOWER – Output 25. MLP 4-3-3
1	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
2	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
4	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
8	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
9	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
10	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
16	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
17	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
19	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
20	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
22	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
24	Setosa	Setosa	Setosa	Setosa	Setosa	Versicol
26	Setosa	Setosa	Setosa	Setosa	Setosa	Setosa
.	.	.	.	.	.	.

Exemple 2 : classification – données = iris

FLOWER : Confusion matrix samples: Train	FLOWER Setosa	FLOWER Versicol	FLOWER Virginic
21.MLP 4-3-3-Setosa	23	0	0
21.MLP 4-3-3-Versicol	0	23	2
21.MLP 4-3-3-Virginic	0	1	21
22.RBF 4-21-3-Setosa	23	0	0
22.RBF 4-21-3-Versicol	0	22	2
22.RBF 4-21-3-Virginic	0	2	21
23.MLP 4-3-3-Setosa	23	0	0
23.MLP 4-3-3-Versicol	0	24	3
23.MLP 4-3-3-Virginic	0	0	20
24.RBF 4-21-3-Setosa	23	0	0
24.RBF 4-21-3-Versicol	0	24	0
24.RBF 4-21-3-Virginic	0	0	23
25.MLP 4-3-3-Setosa	21	0	0
25.MLP 4-3-3-Versicol	2	2	0
25.MLP 4-3-3-Virginic	0	22	23

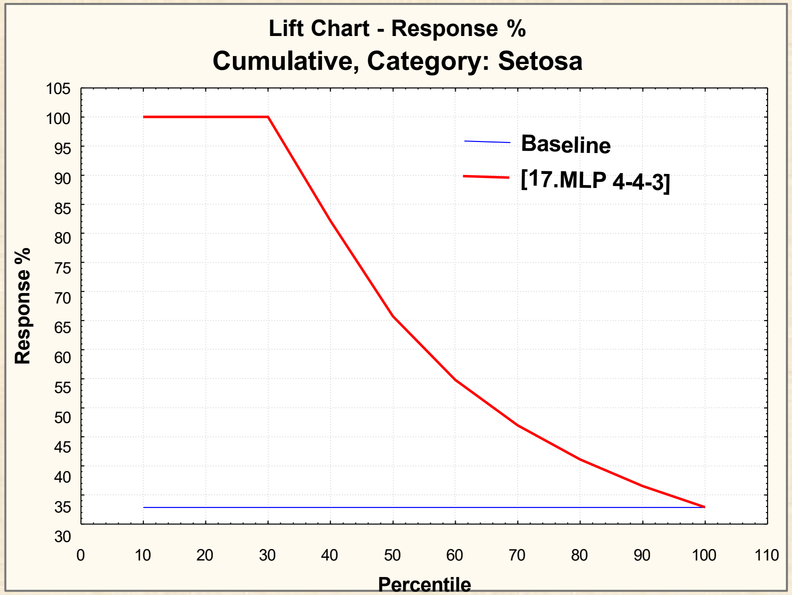
FLOWER (Classification summary) Samples: Train		FLOWER Setosa	FLOWER Versicol	FLOWER Virginic	FLOWER All
21.MLP 4-3-3	Total	23,00	24,00	23,00	70,00
	Correct	23,00	23,00	21,00	67,00
	Incorrect	0,00	1,00	2,00	3,00
	Correct (%)	100,00	95,83	91,30	95,71
	Incorrect (%)	0,00	4,17	8,70	4,29
22.RBF 4-21-3	Total	23,00	24,00	23,00	70,00
	Correct	23,00	22,00	21,00	66,00
	Incorrect	0,00	2,00	2,00	4,00
	Correct (%)	100,00	91,67	91,30	94,29
	Incorrect (%)	0,00	8,33	8,70	5,71
23.MLP 4-3-3	Total	23,00	24,00	23,00	70,00
	Correct	23,00	24,00	20,00	67,00
	Incorrect	0,00	0,00	3,00	3,00
	Correct (%)	100,00	100,00	86,96	95,71
	Incorrect (%)	0,00	0,00	13,04	4,29
24.RBF 4-21-3	Total	23,00	24,00	23,00	70,00
	Correct	23,00	24,00	23,00	70,00
	Incorrect	0,00	0,00	0,00	0,00
	Correct (%)	100,00	100,00	100,00	100,00
	Incorrect (%)	0,00	0,00	0,00	0,00
25.MLP 4-3-3	Total	23,00	24,00	23,00	70,00
	Correct	21,00	2,00	23,00	46,00
	Incorrect	2,00	22,00	0,00	24,00
	Correct (%)	91,30	8,33	100,00	65,71
	Incorrect (%)	8,70	91,67	0,00	34,29

Exemple 2 : classification – données = iris

Sensitivity analysis	Samples: Train			
	SLENGTH	PWIDTH	PLENGTH	SWIDTH
21.MLP 4-3-3	1,37	5,49	8,12	1,22
22.RBF 4-21-3	1,80	4,36	3,13	1,01
23.MLP 4-3-3	1,03	4,71	6,84	1,04
24.RBF 4-21-3	136,71	51,20	39,39	7,63
25.MLP 4-3-3	1,23	2,47	2,81	1,15
Average	28,43	13,64	12,06	2,41

Copyright © Génistat Conseils Inc., 2021, Montréal, Canada

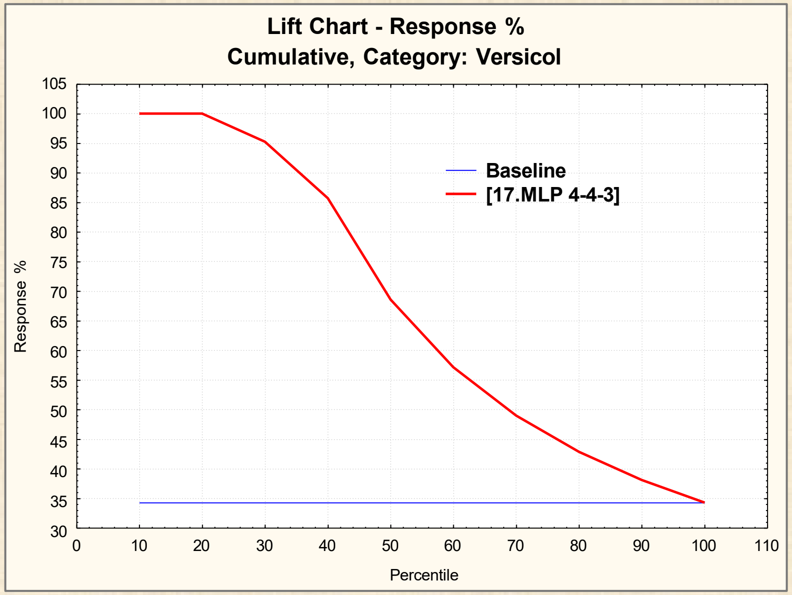
Exemple 2 : classification – données = iris

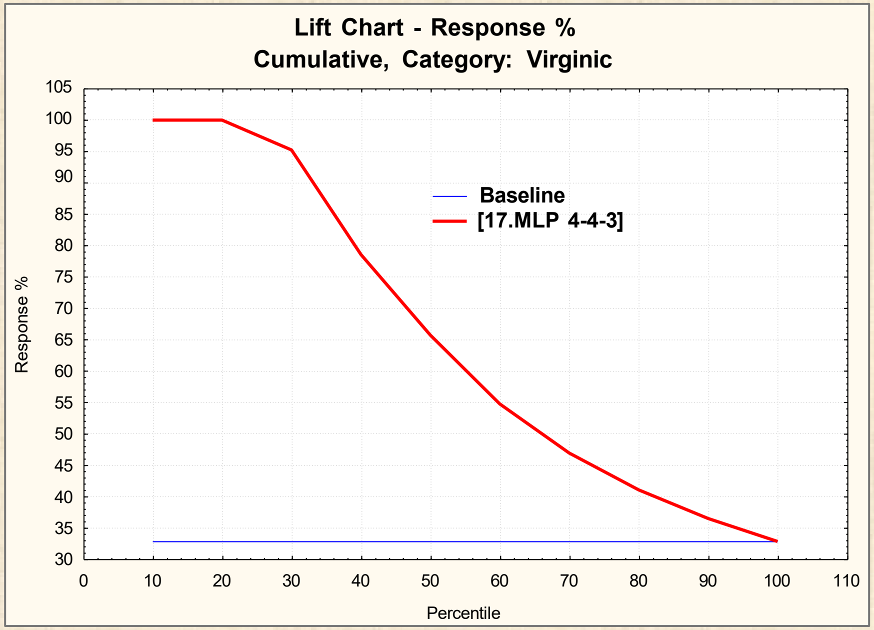


The *lift chart* provides a visual summary of the usefulness of the information provided by one or more statistical models for predicting a binomial (categorical) outcome variable

(dependent variable);
for multinomial (multiple-category) outcome variables,
lift charts can be computed for each category.

the chart summarizes the utility that one may expect by using the respective predictive models, as compared to using baseline information only.

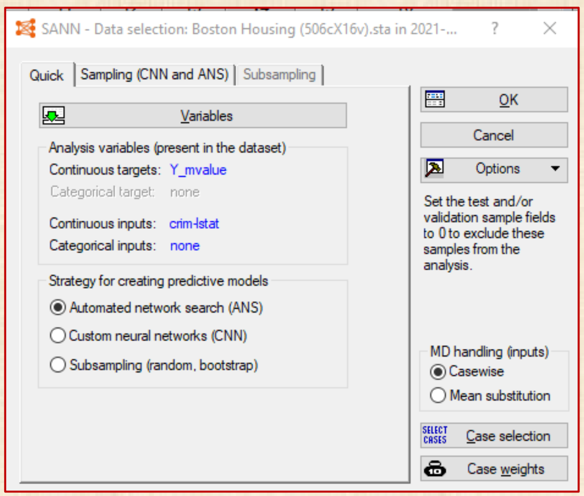


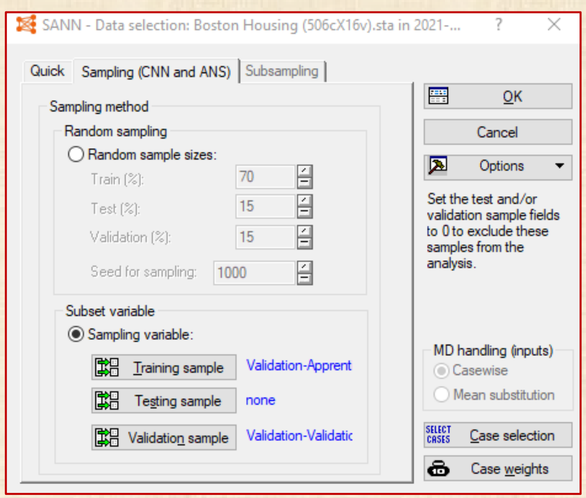


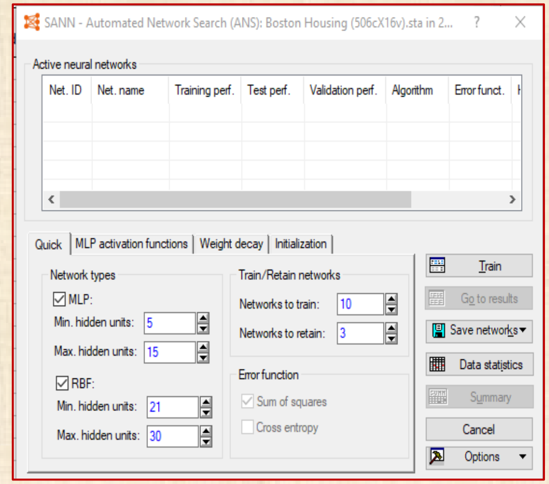
Exemple 3 : classification – données = Boston Houses - regression - analyse STATISTICA

Boston Housing source : importé de JMP Boston Housing
validation : séparation des données en 2 groupes 1=Apprentissage (404 obs.) 2=Validation (202 obs)
crim : per capita crime rate by town
zn : proportion of a town's residential land zoned for lots greater than 25,000 square feet
indus : proportion of nonretail business acres per town
chas : Charles River dummy variable with value 1 if tract bounds on the Charles River
nox : nitrogen oxide concentration
rooms : number of rooms
age : age (year) of house
distance : logarithm of the weighted distances to five employment centers in the Boston region
radial : logarithm of index of accessibility to radial highways
tax : full-value property-tax rate (per $10,000)
pt : pupil-teacher ratio by town
b : (P - 0.63)^2 where P is a measure of ethnic diversity
lstat : logarithm of the proportion of the population that is lower status
Y_mvalue : logarithm of the median value of owner-occupied homes.

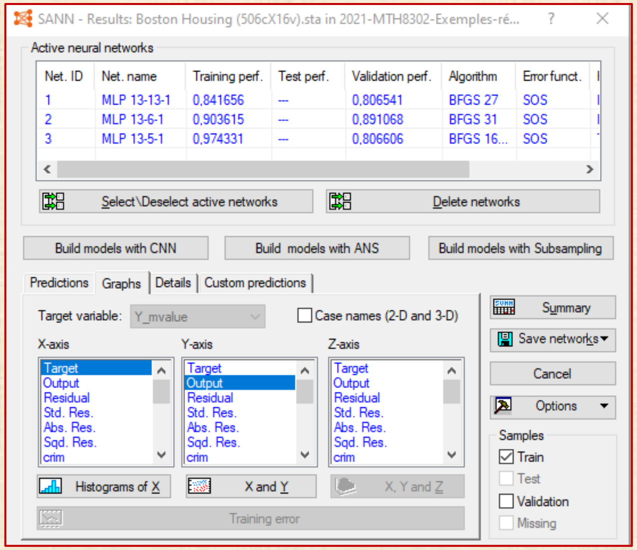
	1 ID	2 Validation	3 crim	4 zn	5 indus	6 chas	7 nox	8 rooms	9 age	10 distance	11 radial	12 tax	13 pt	14 b	15 lstat	16 Y_mvalue
1	1	Apprentissage	0,0063	18,0	2,31	0	0,538	6,575	65,2	4,0900	1	296	15,3	396,90	4,98	24,0
2	2	Apprentissage	0,0273	0,0	7,07	0	0,469	6,421	78,9	4,9671	2	242	17,8	396,90	9,14	21,6
3	3	Apprentissage	0,0273	0,0	7,07	0	0,469	7,185	61,1	4,9671	2	242	17,8	392,83	4,03	34,7
4	4	Apprentissage	0,0324	0,0	2,18	0	0,458	6,998	45,8	6,0622	3	222	18,7	394,63	2,94	33,4
5	5	Validation	0,0691	0,0	2,18	0	0,458	7,147	54,2	6,0622	3	222	18,7	396,90	5,33	36,2
6	6	Apprentissage	0,0299	0,0	2,18	0	0,458	6,430	58,7	6,0622	3	222	18,7	394,12	5,21	28,7
7	7	Apprentissage	0,0883	12,5	7,87	0	0,524	6,012	66,6	5,5605	5	311	15,2	395,60	12,43	22,9
503	503	Apprentissage	0,0453	0,0	11,93	0	0,573	6,120	76,7	2,2875	1	273	21,0	396,90	9,08	20,6
504	504	Apprentissage	0,0608	0,0	11,93	0	0,573	6,976	91,0	2,1675	1	273	21,0	396,90	5,64	23,9
505	505	Apprentissage	0,1096	0,0	11,93	0	0,573	6,794	89,3	2,3889	1	273	21,0	393,45	6,48	22,0
506	506	Apprentissage	0,0474	0,0	11,93	0	0,573	6,030	80,8	2,5050	1	273	21,0	396,90	7,88	11,9

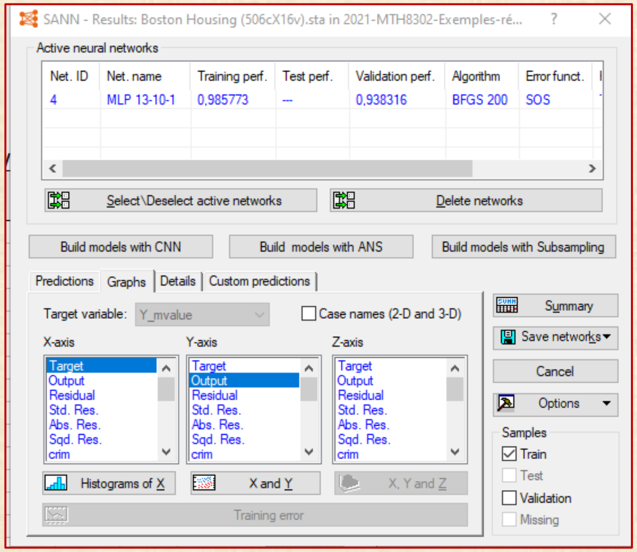


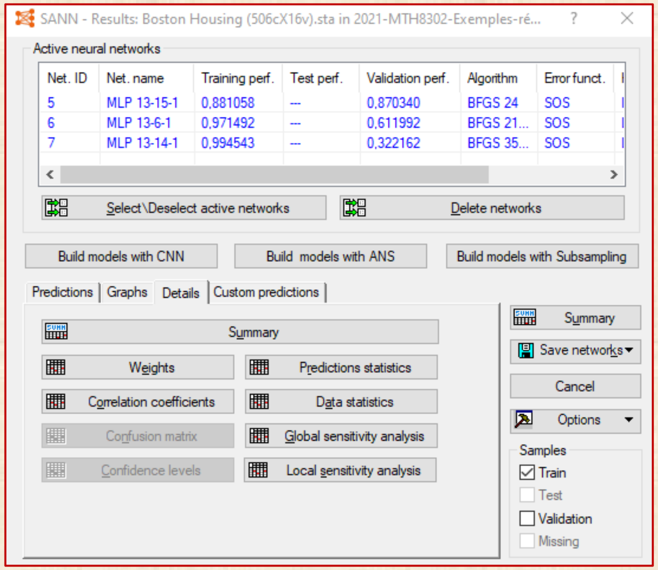




Exemple 3 : classification – données = Boston Houses - regression - analyse STATISTICA





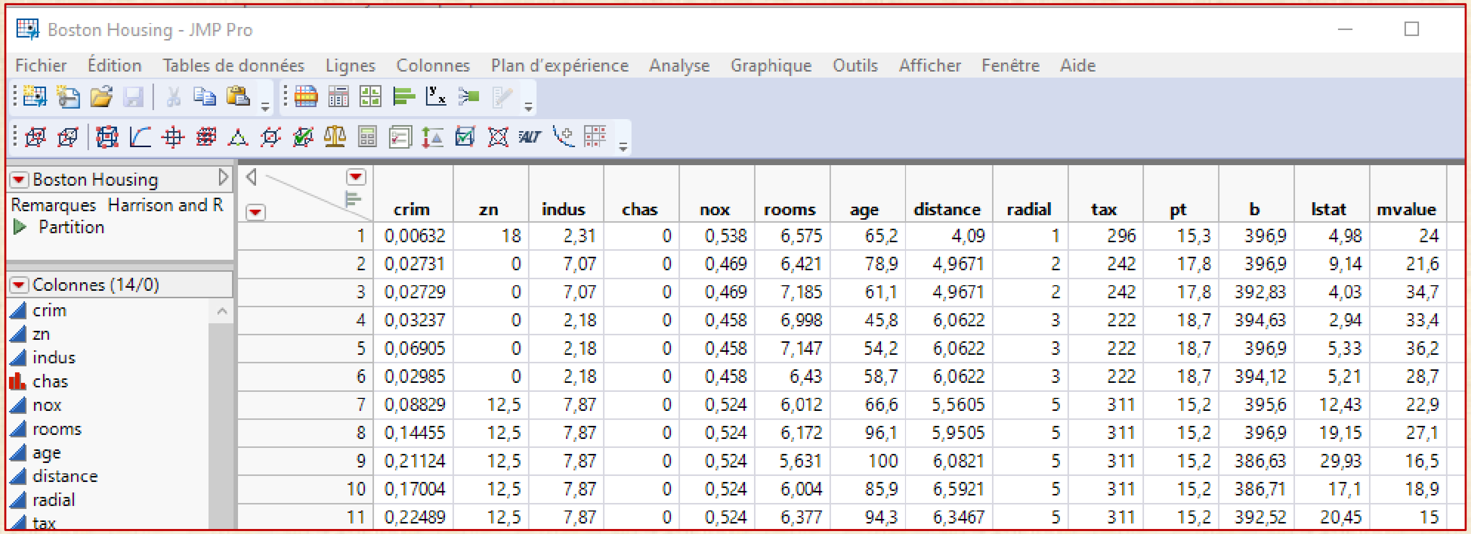


Index	Net. name	Training perf.	Test perf.	Validation perf.	Training error	Test error	Validation error
1	MLP 13-13-1	0,841656		0,806541	12,90751		13,0956
2	MLP 13-6-1	0,903615		0,891068	8,09589		7,1100
3	MLP 13-5-1	0,974331		0,806606	2,23654		12,9404
4	**MLP 13-10-1**	**0,985773**		**0,938316**	**1,24563**		**4,7343**
5	MLP 13-15-1	0,881058		0,870340	9,90193		8,4209
6	MLP 13-6-1	0,971492		0,611992	2,47993		40,6277
7	MLP 13-14-1	0,994543		0,322162	0,47984		255,0125

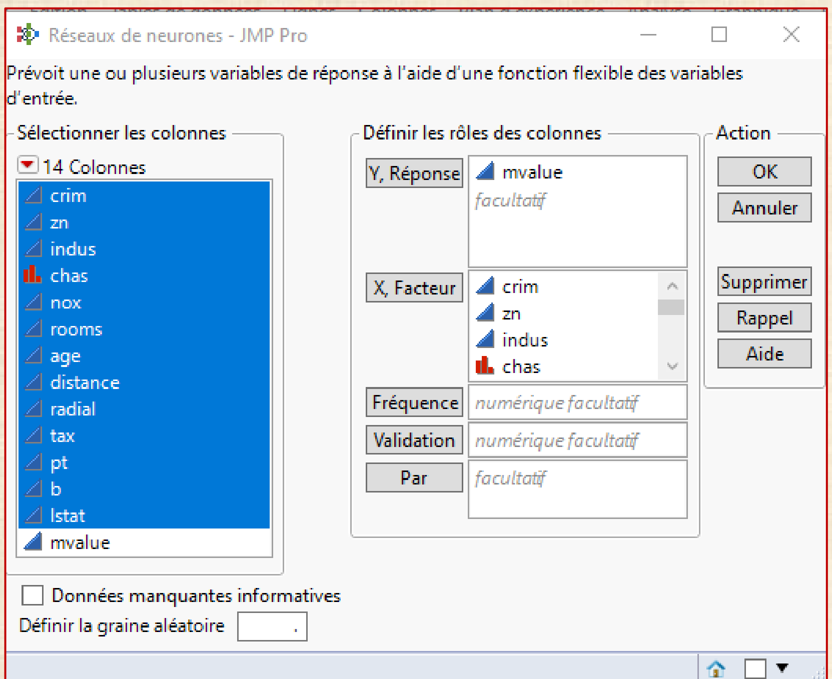
Index	Net. name	Training algorithm	Error function	Hidden activation	Output activation
1	MLP 13-13-1	BFGS 27	SOS	Identity	Tanh
2	MLP 13-6-1	BFGS 31	SOS	Identity	Exponential
3	MLP 13-5-1	BFGS 1680	SOS	Tanh	Tanh
4	**MLP 13-10-1**	**BFGS 200**	**SOS**	**Tanh**	**Identity**
5	MLP 13-15-1	BFGS 24	SOS	Identity	Logistic
6	MLP 13-6-1	BFGS 2142	SOS	Logistic	Tanh
7	MLP 13-14-1	BFGS 3584	SOS	Logistic	Identity

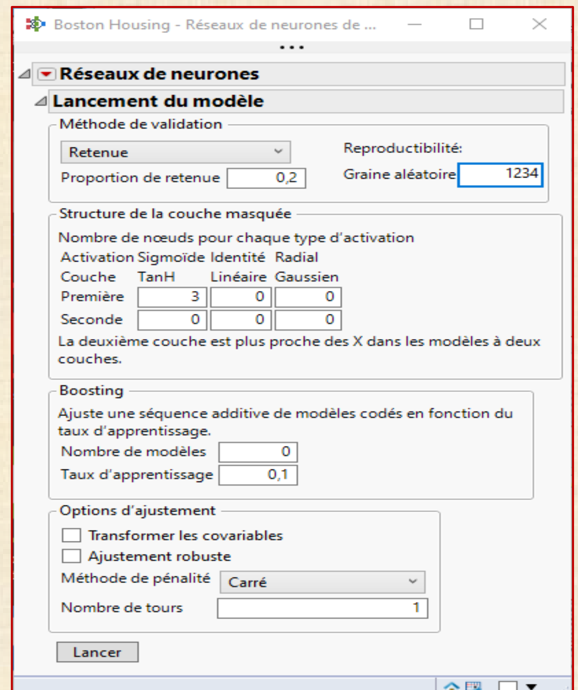
Copyright © Génistat Conseils Inc., 2021, Montréal, Canada

Exemple 3 : classification – données = Boston Houses - régression - analyse JMP Pro



	crim	zn	indus	chas	nox	rooms	age	distance	radial	tax	pt	b	lstat	mvalue
1	0,00632	18	2,31	0	0,538	6,575	65,2	4,09	1	296	15,3	396,9	4,98	24
2	0,02731	0	7,07	0	0,469	6,421	78,9	4,9671	2	242	17,8	396,9	9,14	21,6
3	0,02729	0	7,07	0	0,469	7,185	61,1	4,9671	2	242	17,8	392,83	4,03	34,7
4	0,03237	0	2,18	0	0,458	6,998	45,8	6,0622	3	222	18,7	394,63	2,94	33,4
5	0,06905	0	2,18	0	0,458	7,147	54,2	6,0622	3	222	18,7	396,9	5,33	36,2
6	0,02985	0	2,18	0	0,458	6,43	58,7	6,0622	3	222	18,7	394,12	5,21	28,7
7	0,08829	12,5	7,87	0	0,524	6,012	66,6	5,5605	5	311	15,2	395,6	12,43	22,9
8	0,14455	12,5	7,87	0	0,524	6,172	96,1	5,9505	5	311	15,2	396,9	19,15	27,1
9	0,21124	12,5	7,87	0	0,524	5,631	100	6,0821	5	311	15,2	386,63	29,93	16,5
10	0,17004	12,5	7,87	0	0,524	6,004	85,9	6,5921	5	311	15,2	386,71	17,1	18,9
11	0,22489	12,5	7,87	0	0,524	6,377	94,3	6,3467	5	311	15,2	392,52	20,45	15





Copyright © Génistat Conseils Inc., 2021, Montréal, Canada

Exemple 3 : classification – données = Boston Houses - régression - analyse JMP Pro

Boston Housing - Réseaux de neurones de mvalue - JMP Pro

Réseaux de neurones

Validation : retenue aléatoire

Lancement du modèle

Modèle NTanH(3)

Apprentissage

mvalue

Mesures	Valeur
R carré	0,8990414
Racine de l'erreur quadratique moyenne (RMSE)	2,9836503
Écart absolu moyen	2,1798916
-Log-vraisemblance	1014,8828
Somme des carrés des écarts (SSE)	3596,4764
Somme fréquences	404

Validation

mvalue

Mesures	Valeur
R carré	0,8769066
Racine de l'erreur quadratique moyenne (RMSE)	2,8842914
Écart absolu moyen	2,1396684
-Log-vraisemblance	252,77821
Somme des carrés des écarts (SSE)	848,55193
Somme fréquences	102

Réseaux de neurones

Modèle NTanH(3)

- ✓ Diagramme
- Afficher les estimations
- Profileur
- Profileur d'isoréponses
- Profileur de surface
- Tracer les valeurs observées en fonction des valeurs prévues
- Tracer les résidus en fonction des valeurs prévues
- Enregistrer les formules
- Enregistrer les formules de profil
- Enregistrer les formules rapides
- Publier la formule de prévision
- Réaliser une étape SAS DATA
- Enregistrer la validation
- Supprimer l'ajustement

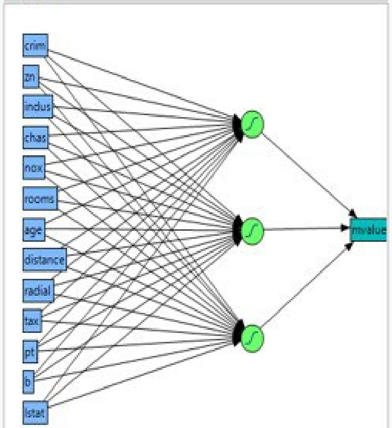
Boston Housing - Réseaux de neurones de mvalue - JMP Pro

Réseaux de neurones

Modèle NTanH(3)

Apprentissage — mvalue | Validation — mvalue

Diagramme

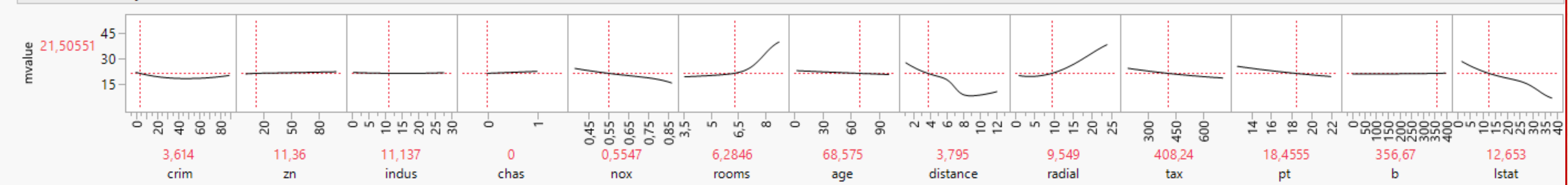


Estimations

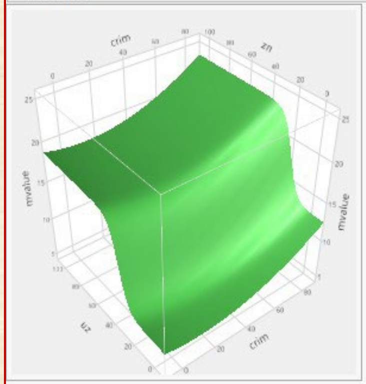
Coefficient	Estimation
H1_1:crim	-0,02834
H1_1:zn	-0,26044
H1_1:indus	0,229695
H1_1:chas:0	0,364441
H1_1:nox	17,58124
H1_1:rooms	1,217198
H1_1:age	-0,0036
H1_1:distance	2,079427
H1_1:radial	0,332004
H1_1:tax	0,002757
H1_1:pt	0,709338
H1_1:b	-0,00774
H1_1:lstat	0,337021
H1_1:Constante	-50,5351
H1_2:crim	0,02769
H1_2:zn	0,000532
H1_2:indus	0,098895
H1_2:chas:0	0,339402
H1_2:nox	0,72
H1_2:rooms	2,193806
H1_2:age	-0,00426
H1_2:distance	0,325575
H1_2:radial	-0,22505
H1_2:tax	-0,00111
H1_2:pt	-0,0981
H1_2:b	-0,00321
H1_2:lstat	-0,03876
H1_2:Constante	-14,2278
H1_3:crim	-0,03488
H1_3:zn	0,001954
H1_3:indus	-0,01524
H1_3:chas:0	-0,13591
H1_3:nox	-3,04636
H1_3:rooms	0,131159
H1_3:age	-0,00412
H1_3:distance	-0,37228
H1_3:radial	0,158363
H1_3:tax	-0,00225
H1_3:pt	-0,10771
H1_3:b	0,000781
H1_3:lstat	-0,09278
H1_3:Constante	3,471601
mvalue_1:H1_1	-5,3079
mvalue_2:H1_2	9,921245
mvalue_3:H1_3	19,22624
mvalue_4:Constante	39,32536

Exemple 3 : classification – données = Boston Houses - régression - analyse JMP Pro

Profileur de prévision



mvalue



Curseur de la grille de réponse

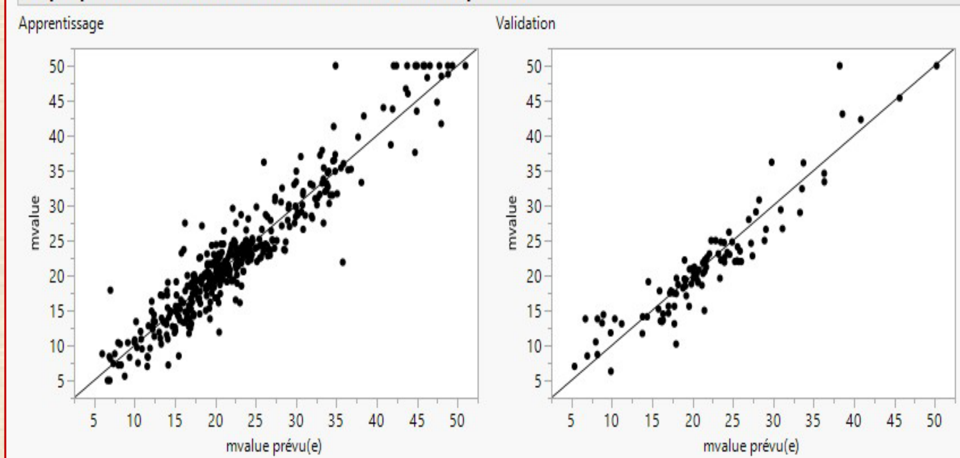
15

Variables indépendantes

X	Y		Valeur	Grille
●	○	crim	44,49126	☐
○	●	zn	50	☐
○	○	indus	14,1	
○	○	chas	0	
○	○	nox	0,628	
○	○	rooms	6,1705	
○	○	age	51,45	
○	○	distance	6,62805	
○	○	radial	12,5	
○	○	tax	449	
○	○	pt	17,3	
○	○	b	198,61	
○	○	lstat	19,85	

☐ Verrouiller l'échelle Z

Graphique des valeurs observées en fonction des valeurs prévues



Dépôt des formules

- Ajouter une formule de la colonne
- Afficher les scripts
- Copier les scripts
- Copier les formules en tant que fonctions
- Copier les formules en tant que transformations
- Exécuter les scripts
- Générer le code C
- Générer le Code Python
- Générer le Code JavaScript
- Générer le code SAS
- Générer le code SQL
- Comparaison de modèles
- Profileur
- Rétablir ▸
- Enregistrer le script ▸

	crim	zn	indus	chas	nox	rooms	age	distance	radial	tax	pt	b	lstat	mvalue	H1_1	H1_2	H1_3	mvalue Prévu	Validation
2	0,02731	0	7,07	0	0,469	6,421	78,9	4,9671	2	242	17,8	396,9	9,14	21,6	-0,999583677	-0,657432497	-0,803655457	22,657230367	Apprentissage
3	0,02729	0	7,07	0	0,469	7,185	61,1	4,9671	2	242	17,8	392,83	4,03	34,7	-0,999792546	0,1908842517	-0,656468668	33,904545846	Apprentissage
4	0,03237	0	2,18	0	0,458	6,998	45,8	6,0622	3	222	18,7	394,63	2,94	33,4	-0,999222189	-0,172336034	-0,671081378	30,016973605	Apprentissage
5	0,06905	0	2,18	0	0,458	7,147	54,2	6,0622	3	222	18,7	396,9	5,33	36,2	-0,99801401	-0,07783296	-0,731004501	29,796052388	Validation
6	0,02985	0	2,18	0	0,458	6,43	58,7	6,0622	3	222	18,7	394,12	5,21	28,7	-0,9991975	-0,700259251	-0,753937892	23,186167793	Apprentissage
7	0,08829	12,5	7,87	0	0,524	6,012	66,6	5,5605	5	311	15,2	395,6	12,43	22,9	-0,999824738	-0,876490211	-0,818668437	20,196541532	Apprentissage
8	0,14455	12,5	7,87	0	0,524	6,172	96,1	5,9505	5	311	15,2	396,9	19,15	27,1	-0,995907144	-0,865742943	-0,91982826	18,337450402	Apprentissage
9	0,21124	12,5	7,87	0	0,524	5,631	100	6,0821	5	311	15,2	386,63	29,93	16,5	-0,893474374	-0,969724178	-0,973810158	15,724255092	Apprentissage
10	0,17004	12,5	7,87	0	0,524	6,004	85,9	6,5921	5	311	15,2	386,71	17,1	18,9	-0,992890734	-0,866155981	-0,92266339	18,262832845	Apprentissage
11	0,22489	12,5	7,87	0	0,524	6,377	94,3	6,3467	5	311	15,2	392,52	20,45	15	-0,980868858	-0,777761832	-0,936391999	18,812052337	Apprentissage

Exemple 3 : classification – données = Boston Houses - régression - analyse JMP Pro

valeurs prédites - valeurs transformées des X

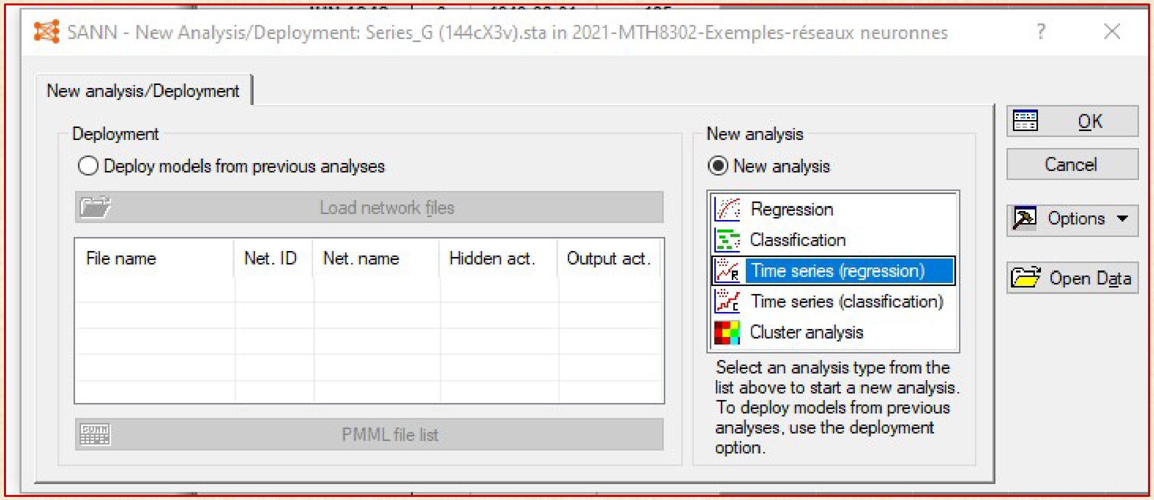
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	crim	zn	indus	chas	nox	rooms	age	distance	radial	tax	pt	b	lstat	mvalue	mvalue Prévu	crim trans	indus trans	nox trans	rooms trans	age trans	distance trans	radial trans	tax trans	pt trans	b trans	lstat trans
2	0,00632	18	2,31	0	0,538	6,575	65,2	4,09	1	296	15,3	396,9	4,98	24,00	25,85	-11,900	-2,588	-0,959	0,810	0,629	-1,078	-2,267	-1,909	-0,070	-0,447	-2,336
3	0,02731	0	7,07	0	0,469	6,421	78,9	4,9671	2	242	17,8	396,9	9,14	21,60	22,48	-8,408	-1,175	-1,695	0,612	1,314	-0,720	-1,857	-2,546	0,711	-0,447	-1,497
4	0,02729	0	7,07	0	0,469	7,185	61,1	4,9671	2	242	17,8	392,83	4,03	34,70	32,41	-8,409	-1,175	-1,695	1,397	0,453	-0,720	-1,857	-2,546	0,711	-3,141	-2,640
5	0,03237	0	2,18	0	0,458	6,998	45,8	6,0622	3	222	18,7	394,63	2,94	33,40	35,28	-8,198	-2,660	-1,851	1,245	-0,162	-0,327	-1,182	-2,907	1,038	-2,592	-3,132
6	0,06905	0	2,18	0	0,458	7,147	54,2	6,0622	3	222	18,7	396,9	5,33	36,20	28,65	-7,333	-2,660	-1,851	1,368	0,172	-0,327	-1,182	-2,907	1,038	-0,447	-2,240
7	0,02985	0	2,18	0	0,458	6,43	58,7	6,0622	3	222	18,7	394,12	5,21	28,70	26,64	-8,297	-2,660	-1,851	0,624	0,354	-0,327	-1,182	-2,907	1,038	-2,780	-2,272
8	0,08829	12,5	7,87	0	0,524	6,012	66,6	5,5605	5	311	15,2	395,6	12,43	22,90	21,16	-7,068	-1,026	-1,086	-0,018	0,690	-0,502	1,359	-1,777	-0,100	-2,095	-1,058
9	0,14455	12,5	7,87	0	0,524	6,172	96,1	5,9505	5	311	15,2	396,9	19,15	27,10	19,77	-6,548	-1,026	-1,086	0,241	3,149	-0,366	1,359	-1,777	-0,100	-0,447	-0,362
10	0,21124	12,5	7,87	0	0,524	5,631	100	6,0821	5	311	15,2	386,63	29,93	16,50	16,71	-6,155	-1,026	-1,086	-0,603	6,027	-0,320	1,359	-1,777	-0,100	-4,041	0,644
11	0,17004	12,5	7,87	0	0,524	6,004	85,9	6,5921	5	311	15,2	386,71	17,1	18,90	19,97	-6,379	-1,026	-1,086	-0,031	1,796	-0,147	1,359	-1,777	-0,100	-4,033	-0,559
12	0,22489	12,5	7,87	0	0,524	6,377	94,3	6,3467	5	311	15,2	392,52	20,45	15,00	18,84	-6,091	-1,026	-1,086	0,550	2,770	-0,230	1,359	-1,777	-0,100	-3,212	-0,241
13	0,11747	12,5	7,87	0	0,524	6,009	82,9	6,2267	5	311	15,2	396,9	13,27	18,90	20,82	-6,765	-1,026	-1,086	-0,023	1,571	-0,271	1,359	-1,777	-0,100	-0,447	-0,961
14	0,09378	12,5	7,87	0	0,524	5,889	39	5,4509	5	311	15,2	390,5	15,71	21,70	20,37	-7,004	-1,026	-1,086	-0,217	-0,438	-0,542	1,359	-1,777	-0,100	-3,578	-0,698
15	0,62976	0	8,14	0	0,538	5,949	61,8	4,7075	4	307	21	396,9	8,26	20,40	20,10	-5,040	-0,978	-0,959	-0,120	0,483	-0,821	0,169	-1,811	2,333	-0,447	-1,637
16	0,63796	0	8,14	0	0,538	6,096	84,5	4,4619	4	307	21	380,02	10,26	18,20	19,52	-5,027	-0,978	-0,959	0,119	1,686	-0,920	0,169	-1,811	2,333	-4,531	-1,335
17	0,62739	0	8,14	0	0,538	5,834	56,5	4,4986	4	307	21	395,62	8,47	19,90	20,07	-5,044	-0,978	-0,959	-0,304	0,265	-0,905	0,169	-1,811	2,333	-2,082	-1,603
18	1,05393	0	8,14	0	0,538	5,935	29,3	4,4986	4	307	21	386,85	6,58	23,10	21,21	-4,517	-0,978	-0,959	-0,143	-0,867	-0,905	0,169	-1,811	2,333	-4,019	-1,950
19	0,7842	0	8,14	0	0,538	5,99	81,7	4,2579	4	307	21	386,75	14,67	17,50	17,09	-4,818	-0,978	-0,959	-0,054	1,489	-1,005	0,169	-1,811	2,333	-4,029	-0,807
20	0,80271	0	8,14	0	0,538	5,456	36,6	3,7965	4	307	21	288,99	11,69	20,20	18,85	-4,794	-0,978	-0,959	-0,828	-0,540	-1,212	0,169	-1,811	2,333	-6,378	-1,148
21	0,7258	0	8,14	0	0,538	5,727	69,5	3,7965	4	307	21	390,95	11,28	18,20	19,04	-4,896	-0,978	-0,959	-0,466	0,823	-1,212	0,169	-1,811	2,333	-3,507	-1,200
22	1,25179	0	8,14	0	0,538	5,57	98,1	3,7979	4	307	21	376,57	21,02	13,60	14,20	-4,342	-0,978	-0,959	-0,685	3,829	-1,211	0,169	-1,811	2,333	-4,715	-0,188
23	0,85204	0	8,14	0	0,538	5,965	89,2	4,0123	4	307	21	392,53	13,83	19,60	17,51	-4,733	-0,978	-0,959	-0,094	2,095	-1,113	0,169	-1,811	2,333	-3,209	-0,898
24	1,23247	0	8,14	0	0,538	6,142	91,7	3,9769	4	307	21	396,9	18,72	15,20	14,74	-4,358	-0,978	-0,959	0,193	2,379	-1,129	0,169	-1,811	2,333	-0,447	-0,402
25	0,98843	0	8,14	0	0,538	5,813	100	4,0952	4	307	21	394,54	19,88	14,50	14,48	-4,583	-0,978	-0,959	-0,337	6,027	-1,076	0,169	-1,811	2,333	-2,628	-0,294
26	0,75026	0	8,14	0	0,538	5,924	94,1	4,3996	4	307	21	394,33	16,3	15,60	16,51	-4,863	-0,978	-0,959	-0,161	2,734	-0,945	0,169	-1,811	2,333	-2,707	-0,638

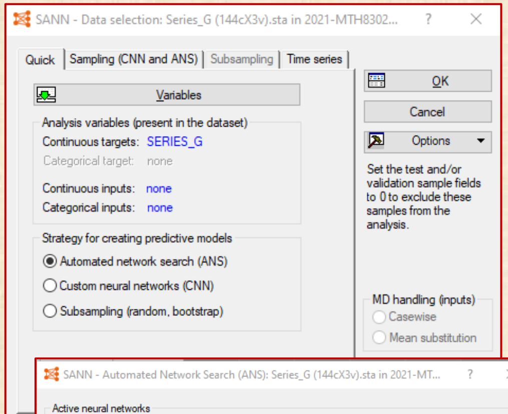
501	0,17783	0	9,69	0	0,585	5,569	73,5	2,3999	6	391	19,2	395,77	15,1	17,50	20,14	-6,333	-0,720	-0,574	-0,686	1,018	-2,066	1,956	-1,218	1,243	-1,976	-0,761
502	0,22438	0	9,69	0	0,585	6,027	79,7	2,4982	6	391	19,2	396,9	14,33	16,80	20,36	-6,093	-0,720	-0,574	0,007	1,362	-1,986	1,956	-1,218	1,243	-0,447	-0,843
503	0,06263	0	11,93	0	0,573	6,593	69,1	2,4786	1	273	21	391,99	9,67	22,40	21,88	-7,440	-0,383	-0,668	0,832	0,804	-2,002	-2,267	-2,142	2,333	-3,321	-1,419
504	0,04527	0	11,93	0	0,573	6,12	76,7	2,2875	1	273	21	396,9	9,08	20,60	21,02	-7,804	-0,383	-0,668	0,158	1,188	-2,164	-2,267	-2,142	2,333	-0,447	-1,506
505	0,06076	0	11,93	0	0,573	6,976	91	2,1675	1	273	21	396,9	5,64	23,90	29,57	-7,473	-0,383	-0,668	1,226	2,293	-2,278	-2,267	-2,142	2,333	-0,447	-2,162
506	0,10959	0	11,93	0	0,573	6,794	89,3	2,3889	1	273	21	393,45	6,48	22,00	27,24	-6,838	-0,383	-0,668	1,054	2,105	-2,075	-2,267	-2,142	2,333	-2,984	-1,971
507	0,04741	0	11,93	0	0,573	6,03	80,8	2,505	1	273	21	396,9	7,88	11,90	21,28	-7,751	-0,383	-0,668	0,012	1,431	-1,981	-2,267	-2,142	2,333	-0,447	-1,702

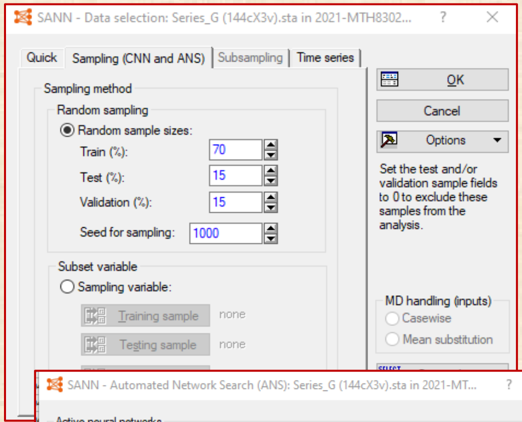
Exemple 4 : classification – données = série G – données chronologiques

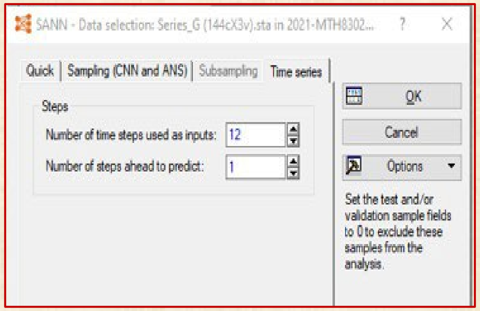
Variation mensuelle de la clientèle totale (en 1000) 1949-1960 industrie aéronautique Box & Jenkins, 1976; series G.

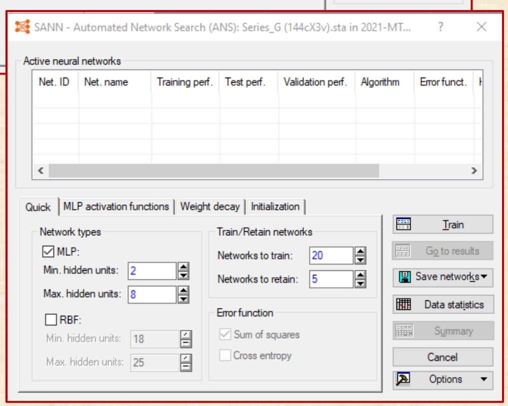
	1 ID	2 DATE	3 SERIES_G
JAN 1949	1	1949-01-01	112
FEB 1949	2	1949-02-01	118
MAR 1949	3	1949-03-01	132
APR 1949	4	1949-04-01	129
MAY 1949	5	1949-05-01	121
JUN 1949	6	1949-06-01	135
…	…	…	…
APR 1960	136	1960-04-01	461
MAY 1960	137	1960-05-01	472
JUN 1960	138	1960-06-01	535
JUL 1960	139	1960-07-01	622
AUG 1960	140	1960-08-01	606
SEP 1960	141	1960-09-01	508
OCT 1960	142	1960-10-01	461
NOV 1960	143	1960-11-01	390
DEC 1960	144	1960-12-01	432

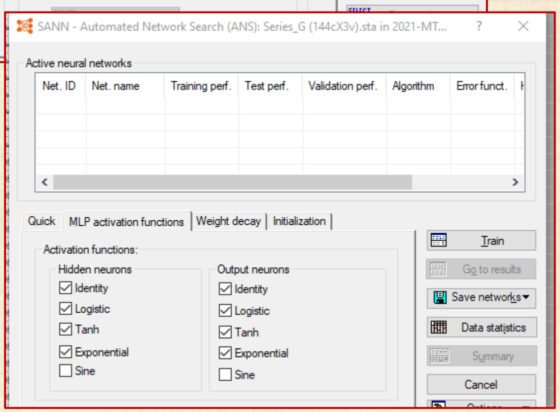


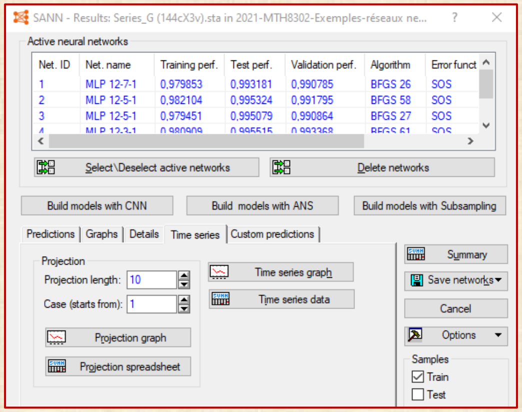












Exemple 4 : classification – données = série G – données chronologiques