Ecole KDD - Lyon, 19.12.97

Louis WEHENKEL
Université de Liège - Belgique
http://www.montefiore.ulg.ac.be/~lwh/

MENU.

Méthodes d'apprentissage complémentaires et approches hybrides : arbres et réseaux de neurones, arbres et méthode du plus proche voisin.

Méthodes de validation : Que valider ? (une méthode ou un résultat particulier), quelles informations veut-on obtenir ? (types d'erreurs, domaine de validité). Techniques statistiques (resubstitution, ensemble de test, méthode de rotation). Confrontation avec les connaissances du domaine.

Exemples d'applications dans les réseaux éléctriques

Partie I
Méthodes d'apprentissage complémentaires et approches hybrides

- Motivations
- Types d'approches hybrides
- Principales classes de méthodes d'apprentissage
- Exemples d'approches hybrides

Motivations

- Diversité de problèmes d'extraction de connaissances
  - Description (du contenu d'une BD)
    Analyses de similarités
    Analyses de corrélations
    Exploration de données
  - Induction (de modèles généraux)
    Classification
    Régression
    Prédiction de séries temporelles
  - Transduction (ponctuelle)
    Prédiction d'informations au cas par cas

- Diversité de critères d'optimalité
  - Pouvoir prédictif
    Précision/fiabilité en généralisation
  - Compréhensibilité par l'expert humain
    Assimilation et Validation
  - Complexité computationnelle
    Apprentissage sur de grandes bases de données
    Utilisation en temps-réel
  - Robustesse
    Variance par rapport aux échantillons d'apprentissage
    Valeurs manquantes
  - Coût
    Collecte de base de données
    Obtention de mesures

- Diversité de méthodes d'apprentissage
  - Apprentissage symbolique
    Formalismes de représentation ± puissants
    Déterministe vs probabiliste
    Net vs flou
  - Modèles statistiques
    Reconnaissance de formes
    Estimation de densités de probabilités
    Régression
  - Approches neuronales
    Supervisé
    Non-supervisé

Et surtout, pas de méthodes universelles...

Types d'approches hybrides

- Boîte à outils
  Utilisation des différentes méthodes,
  successivement sur une même BD

- Couplage lâche
  Exploitation des résultats d'une méthode
  lors de l'application d'une autre
  Exemple : sélection des attributs significatifs

- Approches intégrées
  Combinaison de différents algorithmes en un seul
  Exemple : combinaisons linéaires dans les arbres de décision

But : combiner les avantages en évitant les inconvénients

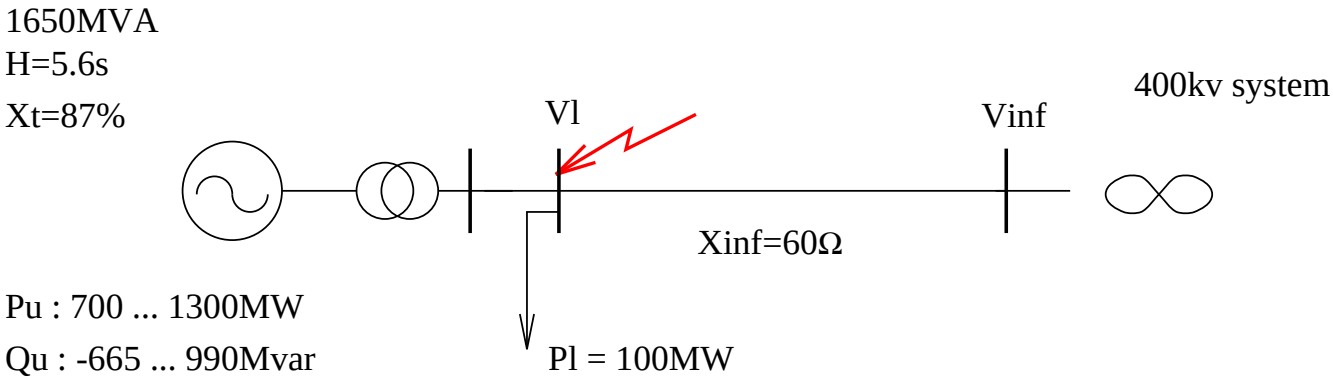
Illustration de l'approche boîte à outils

- Problème d'apprentissage illustratif
- Arbres de décision
- Réseaux de neurones
- Méthode du plus proche voisin

Exemple académique de problème d'apprentissage

Stabilité transitoire des réseaux électriques

NB. "Toy problem"...



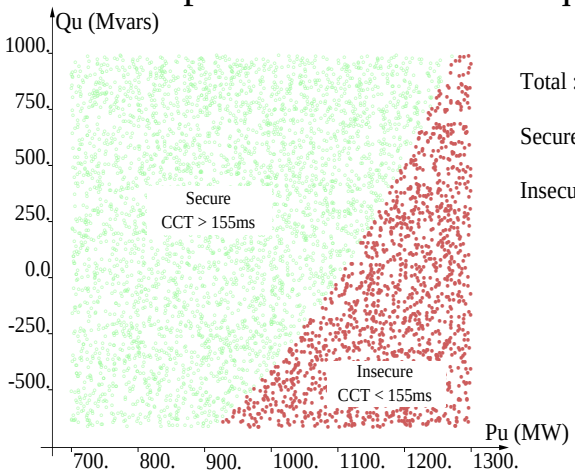
Entrées : Pu et Qu

Sorties : CCT ou classes de sécurité

Exemple académique (fin)

BD : 5000 points de fonctionnement (distributions uniformes de Pu et Qu)

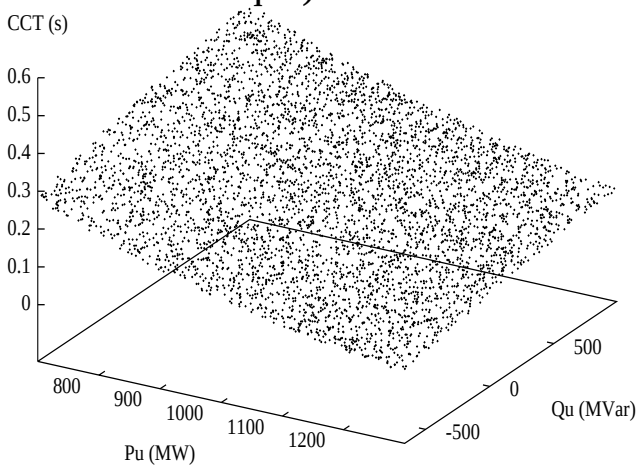
CCTs : par simulation numérique (recherche dichotomique)



Total : 5000 states

Secure : 3510 states

Insecure : 1490 states

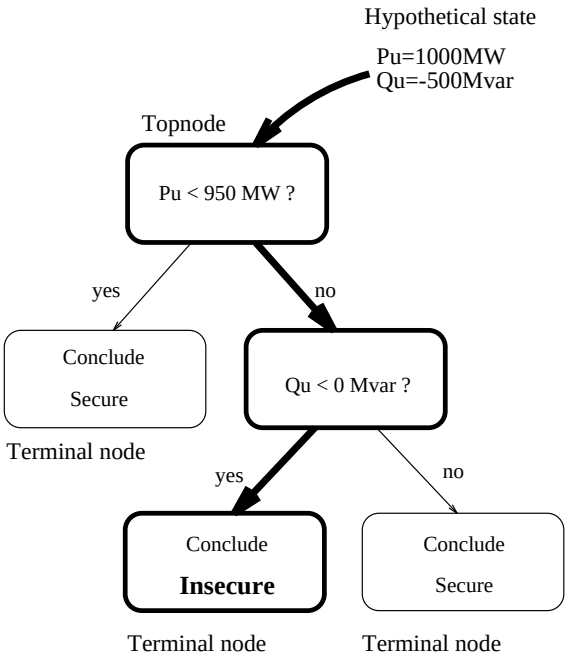


EA : ensemble d'apprentissage (LS) : 3000 états

ET : ensemble de test (TS) : 2000 états différents

Arbres de décision

Qu'est-ce qu'un arbre de décision ?



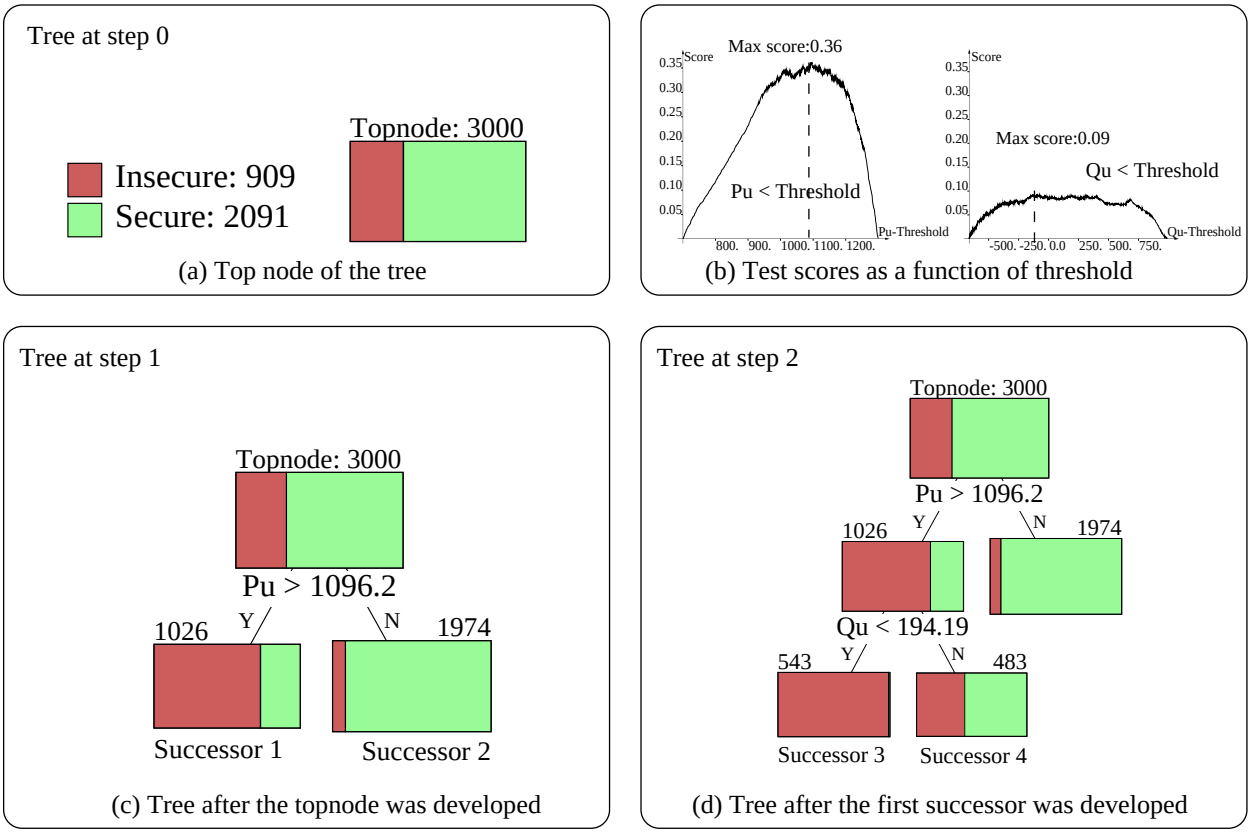
Equivalent If-Then rules :

Rule 1 : If (Pu < 950MW) then Conclude Secure

Rule 2 : If (Pu > 950MW) and (Qu < 0Mvar) then Conclude Insecure

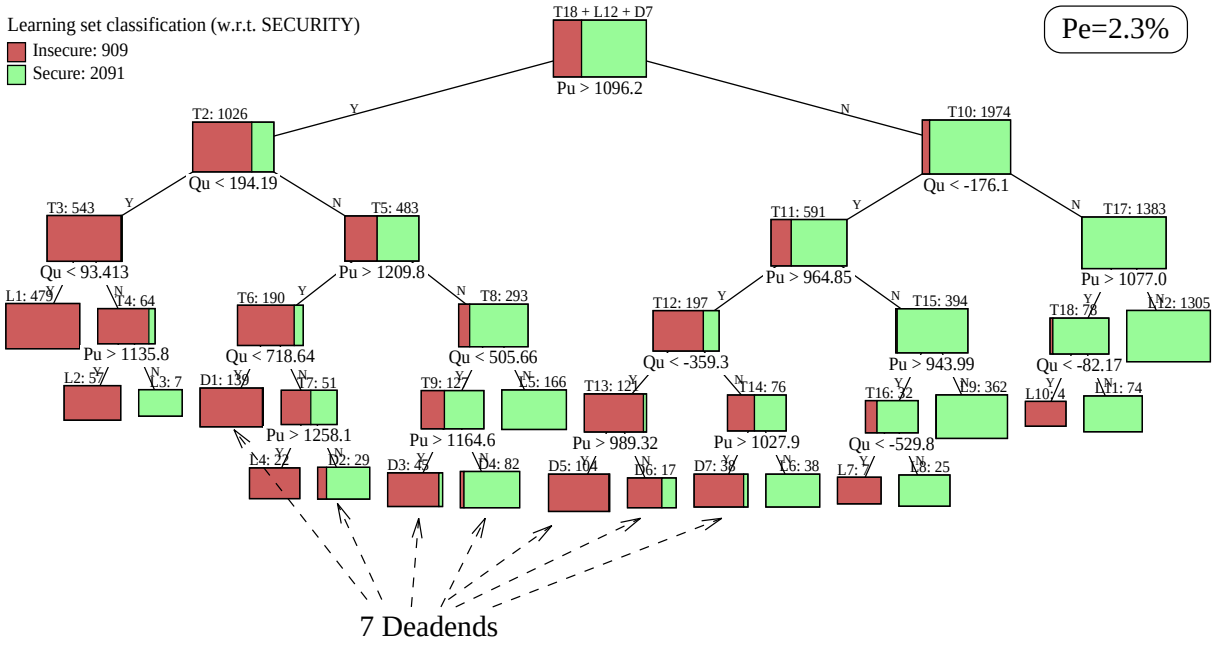
Rule 3 : If (Pu > 950MW) and (Qu > 0Mvar) then Conclude Secure

Construction à partir d'un échantillon d'apprentissage ?



Détails techniques : partitionnement optimal, contrôle de complexité

Résultat obtenu sur l'exemple académique



NB. Validation (échantillon de test) : Combien et quel type d'erreurs

En pratique :

Definition des classes de securité : nombre, seuils

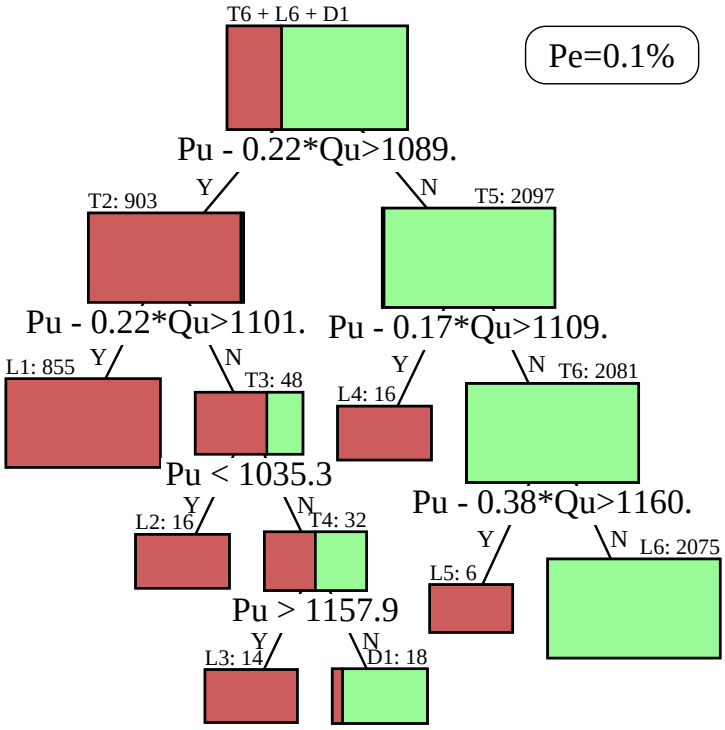
NB : biaiser les seuils pour éviter les erreurs dangereuses...

On propose beaucoup d'attributs candidats

L'algorithme sélectionnera les plus "appropriés" automatiquement

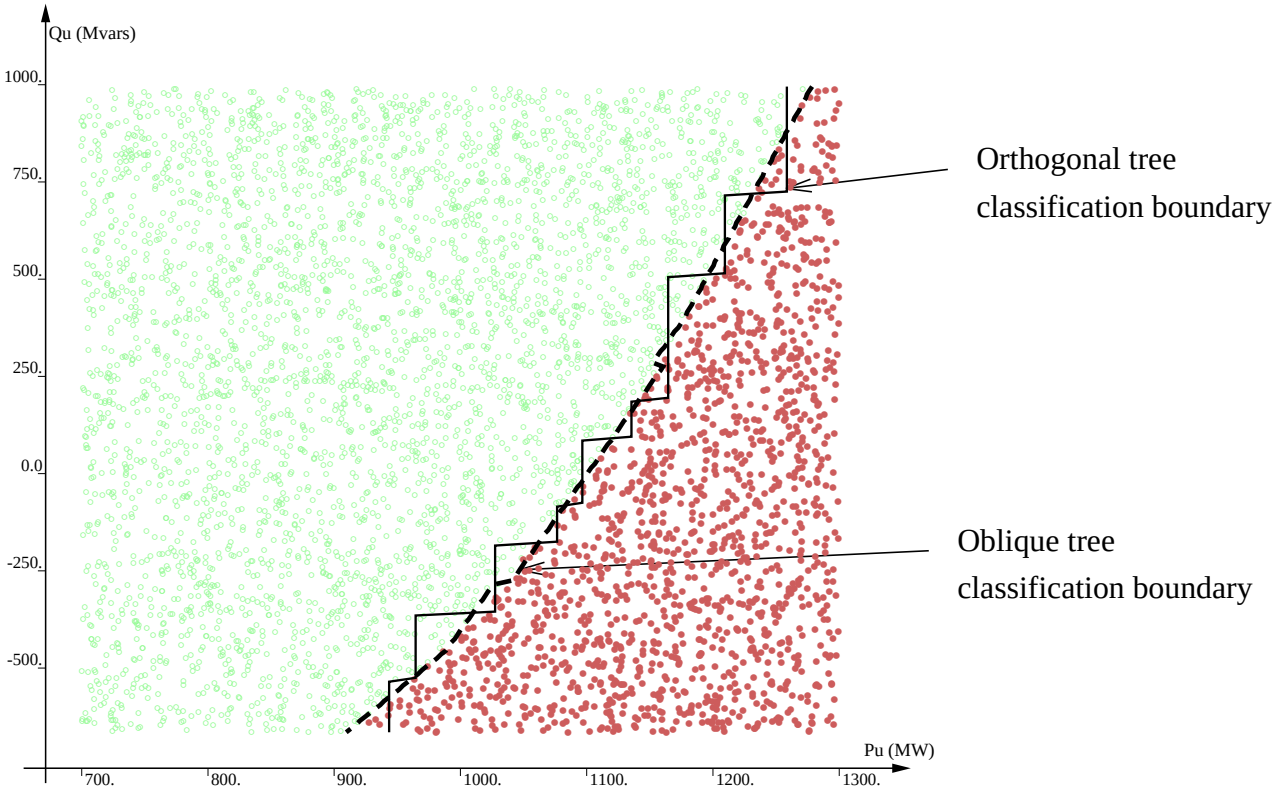
Construction d'un arbre ⇒ sous-produits intéressants

Raffinements : arbres obliques, de régression, logique floue...



... plus simple, plus fiable : meilleur

Frontières de classification orthogonales vs obliques



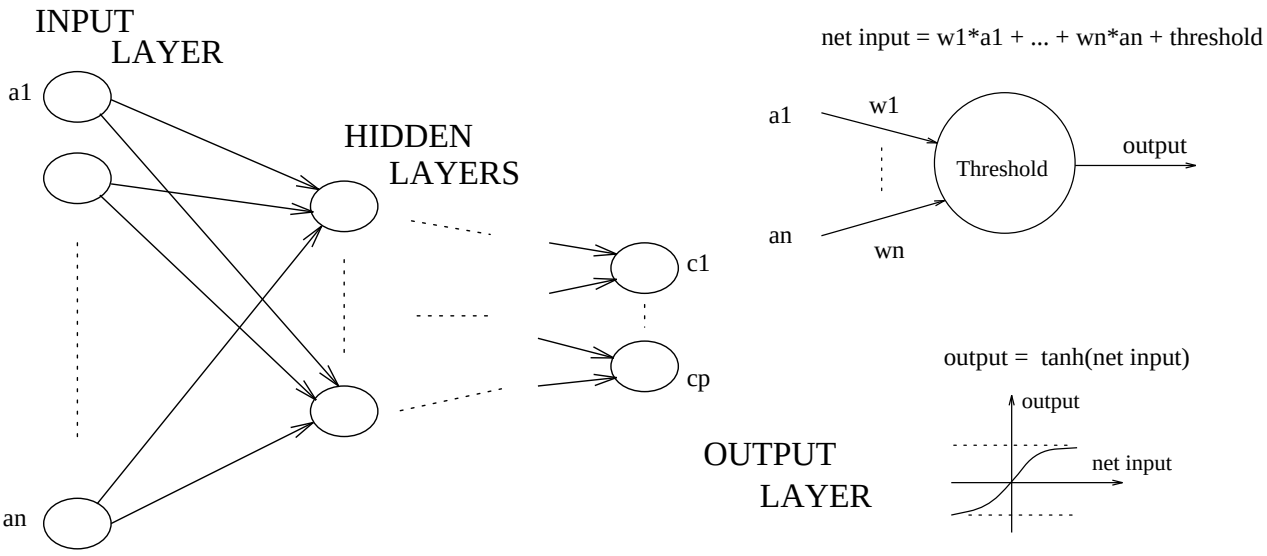
Caractéristiques saillantes des arbres de décision

- Interprétabilité
  - ⇒ compréhension physique
- Identification attributs significatifs
  - ⇒ réduction de la dimensionalité
- Efficacité computationnelle
  - ⇒ essais multiples
- Discrets et plutôt frustes ... (améliorations possibles)

⇒ Coeur de la boîte à outils

Réseaux de neurones (MLPs)

Qu'est-ce qu'un perceptron à couches multiples ?

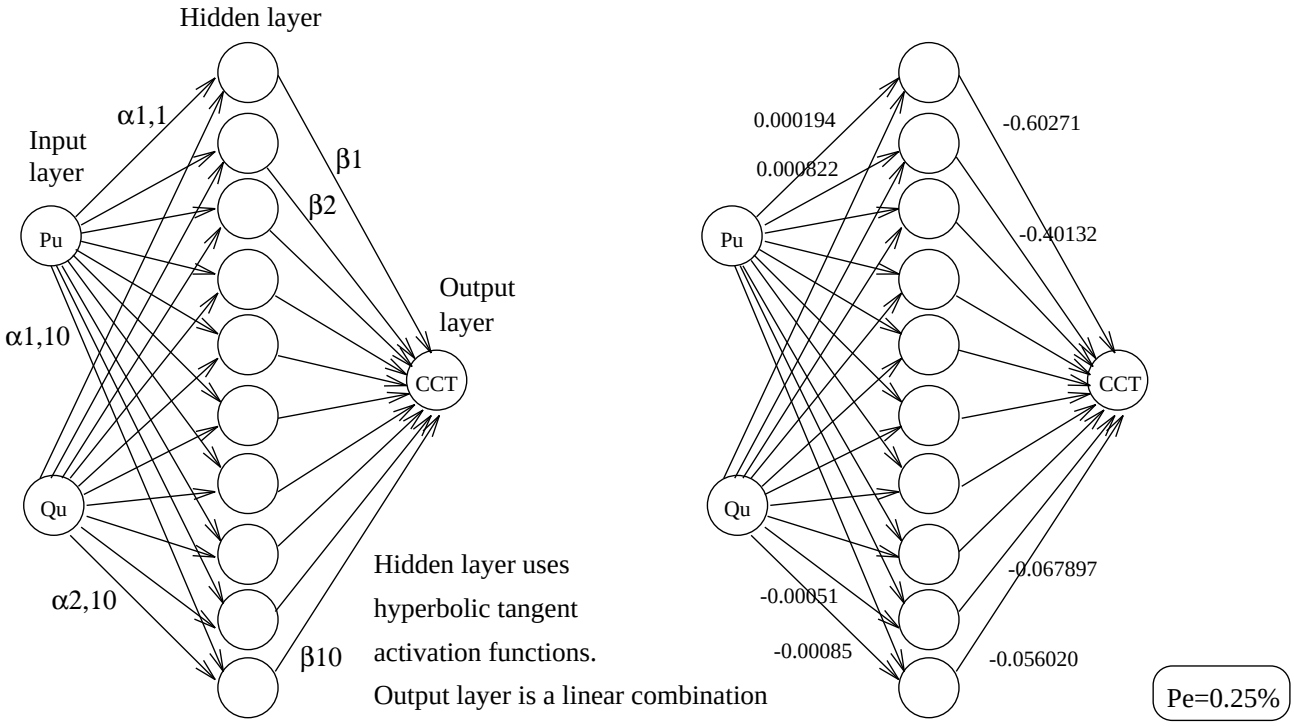


Apprentissage ?

- Choix du codage
  - Entrées : sélection des attributs et normalisation
  - Sorties : choix d'experts... ⇒ marges de sécurité ou classes
- Choix de la structure
  - Couches, neurones, connexions
  - Règles de bonne pratique + essai/erreur (manuel ou systémat.)
- Ajustement des paramètres
  - Pour une structure et un codage fixés
  - Optimisation non-linéaire (divers critères et algorithmes)

NB. Validation croisée pour éviter le sur-apprentissage

Illustration sur l'exemple académique



(a) Initial values of the weights are random

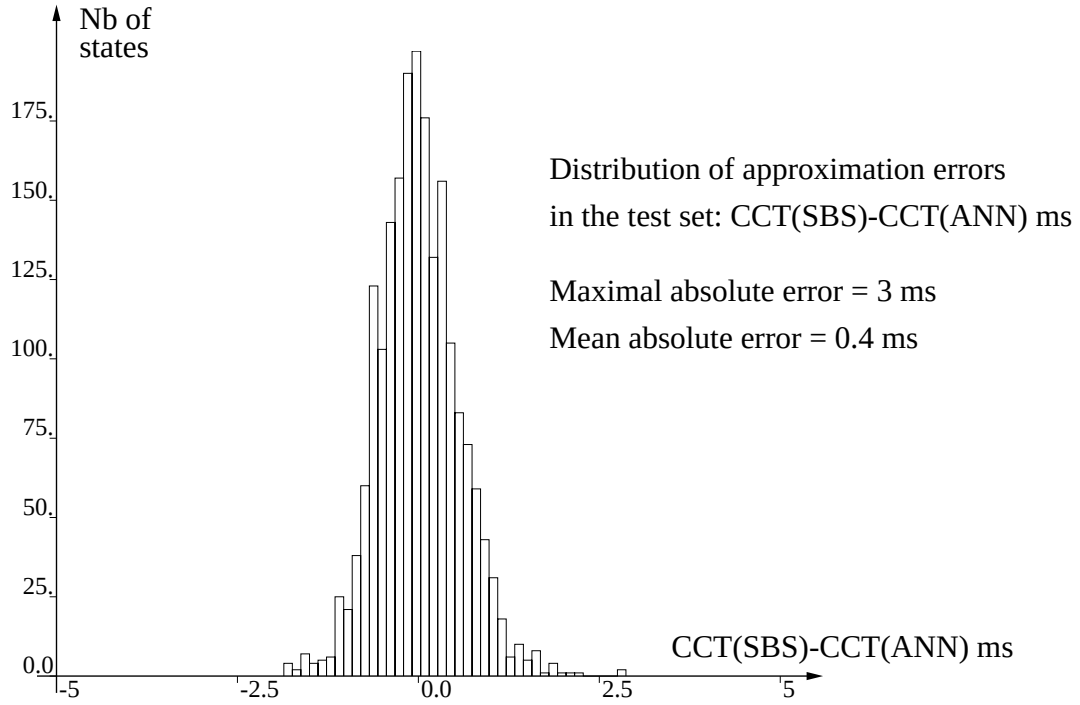
(b) After training the weights are tuned to fit the problem

L'optimisation des paramètres (BFGS) donne après 46 iterations

$$
\begin{aligned}
CCT_{MLP} = \quad & -0.602710 \tanh(0.000194 Pu - 0.00034 Qu - 0.93219) \\
& -0.401320 \tanh(0.000822 Pu - 0.00020 Qu - 0.76681) \\
& +0.318249 \tanh(0.000239 Pu - 0.00050 Qu - 0.29351) \\
& -0.287230 \tanh(0.002004 Pu - 0.00034 Qu - 1.20080) \\
& +0.184522 \tanh(0.000131 Pu - 0.00057 Qu - 0.03152) \\
& +0.177701 \tanh(0.001799 Pu - 0.00011 Qu - 2.08190) \\
& -0.150720 \tanh(0.001530 Pu - 0.00056 Qu - 1.68040) \\
& +0.142678 \tanh(0.002152 Pu - 0.00046 Qu - 1.72280) \\
& -0.067897 \tanh(0.001910 Pu - 0.00051 Qu - 1.71343) \\
& -0.056020 \tanh(0.000202 Pu - 0.00085 Qu - 0.39876)
\end{aligned}
$$

NB: une structure plus simple aurait pu convenir aussi bien...

Test du MLP en généralisation :



ou $P_e = 0.25\%$ ....très précis en effet!

Caractéristiques saillantes des MLPs

- Sorties numériques continues (marges de sécurité)
- Très flexibles et précis en pratique
- Boîte noire (problèmes à grand nombre de variables)
- Risques de sur-apprentissage, de généralisation abusive
- Lourdeur des algorithmes d'optimisation des paramètres

⇒ approche hybride

Plus proche voisin

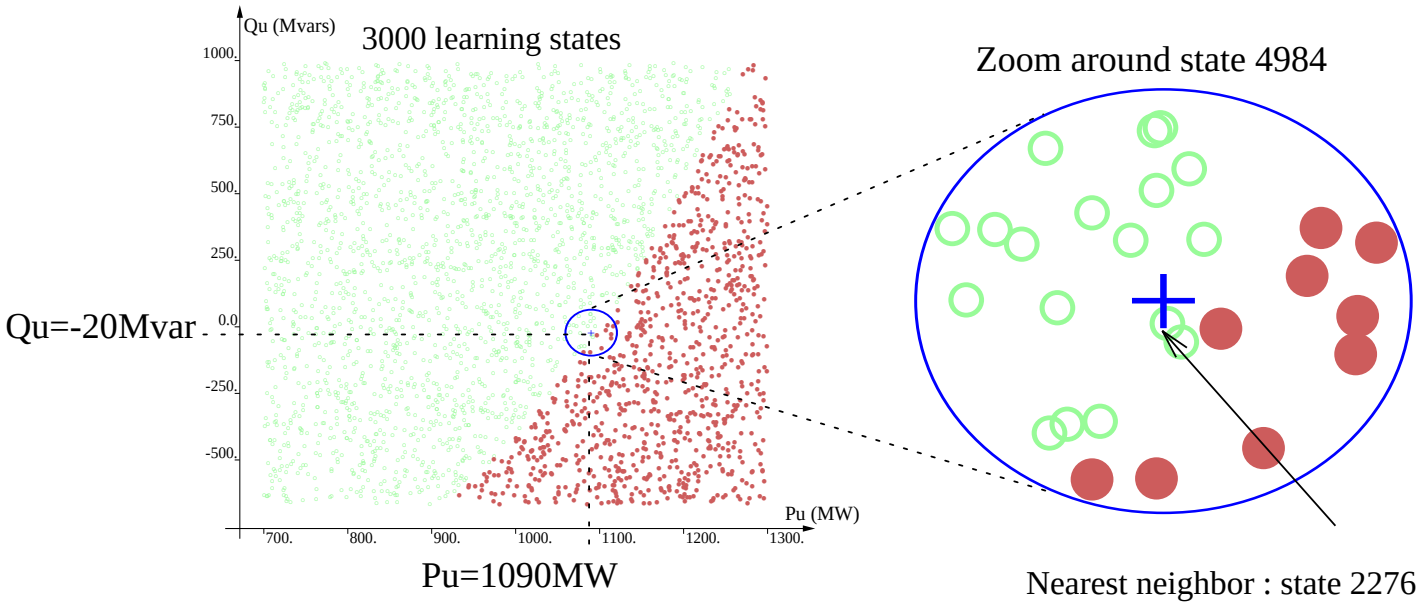
De quoi s'agit-il ?

Definition d'une mesure de distance (ou de (dis-)similarité) dans l'espace des attributs.

Prédiction de la sortie pour un état inconnu :

- Recherche des K états les plus similaires de la BD
- Extrapolation de leurs sorties à l'état inconnu (différentes règles d'agrégation possibles)

Illustration sur l'exemple académique



CCT(état 2276) = 0.157s (erreur = -0.001s) $P_e = 0.9\%$

Caractéristiques saillantes du KNN

- Traite des sorties numériques ou symboliques
- Analyse au cas par cas (détection de cas aberrants)
- Pas de vision globale, synthétique
- Peu robuste : attributs non-relevants ou redondants
- Lent en utilisation; très lent en apprentissage (AGs)

$\Rightarrow$ Approche hybride

Raffinements

Adaptation de K au problème

Sélection des attributs et optimisation de distance

Distances locales

Techniques de recherche efficaces

...

Exemples d'approches hybrides

- Arbres et réseaux de neurones
  Arbres de décision : compréhensibilité et identification des attributs significatifs.
  Réseaux de neurones : souplesse de modélisation (sorties continues)
- Arbres et méthode du plus proche voisin
  Plus proche voisin : diagnostic plus fin, possibilité de cerner le domaine de validité (rejet de distance)
- Arbres flous : approche intégrée qui combine les possibilités des arbres de décision et des réseaux de neurones

Arbres et réseaux de neurones

- Construction d'un arbre de décision
  Grand nombre d'attributs candidats
  Assez rapide même pour une grande base de données
  Précision souvent sous-optimale
- Construction d'un réseau de neurones
  Uniquement les attributs identifiés comme potentiellement utiles
  En général relativement lent lors de l'apprentissage
  Résultat difficile (voire impossible) à interpréter
- Extraction de régles du réseau de neurones
  Construction d'un arbre de décision
  pour traduire le réseau de neurones

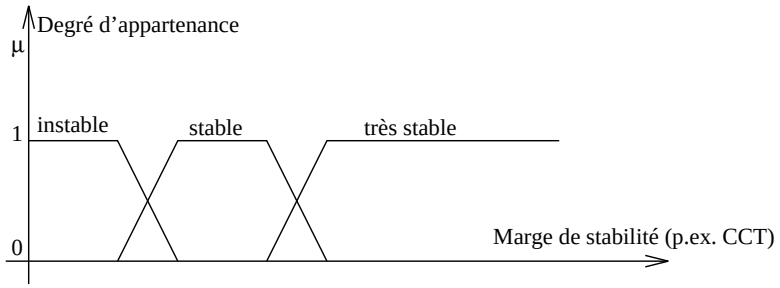
Arbres et méthode du plus proche voisin

- Construction d'un arbre de décision : identification des attributs et mesure de leur pouvoir de discrimination (% quantité d'information apportée dans l'arbre)
- Utilisation des attributs identifés et leur pouvoir de discrimination pour définir la distance.
- Application du KNN et utilisation du "leave-one-out" pour déterminer la bonne valeur de K
- Eventuellement
  - Ajustement des pondérations dans la distance (AG)
  - Définition de distances locales : utilisation de la décomposition de l'espace des attributs définie par l'arbre.

Arbres flous (ou continus)

Dans beaucoup de problèmes les classes ne sont pas vraiment définies de manière nette.

Par exemple : en stabilité transitoire

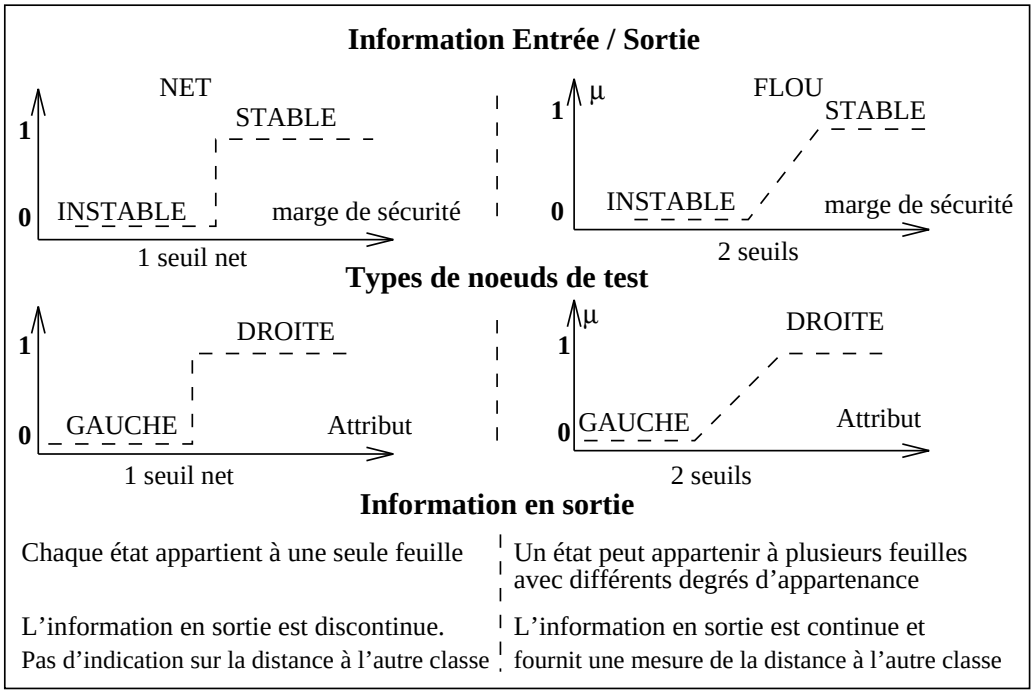


Les arbres flous exploitent des informations *floues*

NB. Sous-ensemble flou X de U est décrit par une *fonction d'appartenance*

$$\mu_X(o) \quad : U \rightarrow [0...1]$$

Différences principales entre arbres flous et nets



Discrétisation floue : recherche de zones de transition (2 seuils).

Caractéristiques saillantes des arbres flous

- Traitement de données floues (entrées et sorties)
- Réduction significative de la variance
- Possibilité d'optimisation globale (cf MLPs)

⇒ Information plus riche, plus précise, plus robuste

- Apprentissage significativement plus lent

Synthèse

	Méthode	Fonctionnalités	Computation. Apprent.	Utilis.
Pures	AD Nets	Bonne interprétabilité (globale). Bonne précision pour problèmes simples, localisés. Faible précision pour problèmes complexes, diffus.	Très bon	Très bon
	MLP	Bonne précision. Interprétabilité faible. Marges et sensibilités.	Très lent	Bon
	kNN	Bonne interprétab. (locale). Simplicité conceptuelle.	Très lent	Très lent
Hybrides	AD Flous	Bonne interprétab. (globale). Symboliques et continus. Plus précis et plus robustes que AD nets. Marges et sensibilités.	Lent	Bon
	AD-MLP	AD + MLP	Lent	Bon
	AD+KNN	AD + KNN	Lent	Lent

Partie II
Méthodes de validation

- Que valider ?
- Techniques statistiques de validation
- Confrontation avec les connaissances du domaine

Que valider ?

Vision théorique de l'erreur en généralisation :

Définitions probabilistes

- Problèmes de classification :
  $P(D_i|C_j)$ (matrice de confusion) et P_e=1-$\sum_i P(D_i|C_i)P(C_i)$
  NB. D_i décision du modèle; C_j classe réelle

- Problèmes de régression :
  $E\{(r-y)^2\}$ ou $E\{|r-y|\}$
  NB. r sortie estimée par le modèle; y sortie réelle.

Remarque : du point de vue statistique, les modèles sont des fonctions aléatoires, car déduits d'un échantillon aléatoire de taille finie.

Trois sources d'erreurs en généralisation

- Incertitude résiduelle : minimum théorique (cf modèle de Bayes)
- Biais : erreur du modèle moyen (moyenne sur différents EA)
- Variance : écart moyen vis-à-vis du modèle moyen

Tendances générales

- Incertitude résiduelle ↗ lorsque le niveau de bruit ↗
  Ne dépend que des caractéristiques du problème
  (relations entrée/sortie et distributions de probabilité)
- Biais ↘ lorsque $\frac{\text{Complexité du modèle}}{\text{Complexité du problème}}$ ↗.
- Variance ↗ lorsque $\frac{\text{Complexité du modèle}}{\text{Taille EA}}$ ↗.

⇒ Compromis biais/variance

Validation d'un algorithme d'apprentissage

- On cherche à évaluer les perfomances en moyenne sur une certaine gamme de problèmes

⇒ Problèmes de test diversifiés

- On cherche à évaluer aussi la robustesse

⇒ Evaluation de la variance par rapport à l'échantillon d'apprentissage

⇒ Evaluation de l'influence de divers paramètres (niveau de bruit, dimensions de la BD, type de problèmes...)

- Notion importante : variance des paramètres des modèles
  Peut rendre l'interprétation plus difficile

Validation d'un résultat particulier

Ce qui intéresse un utilisateur en pratique

- Une estimée de P_e ou $E\{(r-y)^2\}$, certes
- Mais il faudrait qu'elle soit représentative des conditions d'utilisation (cf. problèmes non-stationnaires, BD obtenues de façon synthétique...)
- De plus, il est en général souhaitable d'effectuer une analyse plus fine des erreurs
  - Fausses alarmes vs non-détections
    (coûts associés peuvent être très différents)
  - Erreurs graves vs erreurs marginales
    (proches des frontières, de faible amplitude...)
- Identification des régions où les performances s'écartent de la moyenne

- Cerner le domaine de validité d'un modèle extrait d'une BD
  - Revient à cerner le domaine de validité de la BD
  - Outils auxiliaires : analyses de corrélations, histogrammes, distribution des distances aux K plus proches voisins
  - Nécessite des méthodes fournissant des résultats interprétables
- Confrontation avec les connaissances du domaine
  - En général, sous le contrôle de l'expert
  - Nécessite donc une présentation des résultats sous une forme compréhensible par les experts humains
  - Systèmes TMS (Truth-Maintenance-Systems) utilisés dans les systèmes experts, si les connaissances disponibles sont formalisées sous forme informatisée (mais c'est rarement le cas)

Techniques statistiques

- Resubstitution de l'ensemble d'apprentissage
  ! En général hautement biaisé de façon optimiste

  Exemple en régression :
  Cas trivial : estimation d'un modèle sans paramètres μ_{EA}
  Résultat classique : $\sigma^2_{EA} = \frac{N\text{-}1}{N} \sigma^2_{\text{réel}}$

  Autre exemple (classification ou régression) :
  Il est facile (sauf exception rare) de construire un arbre dont l'erreur de resubstitution est nulle, même si l'erreur résiduelle est très importante.

⇒ fournit une borne inférieure de l'erreur en généralisation, rarement proche de l'erreur réelle
(d'autant moins qu'on exploite bien l'information de la BD).

- Ensemble de test (de taille M)

  Un échantillon indépendant (observations faites indépendamment du modèle et indépendamment les unes des autres)

  On calcule la moyenne des erreurs sur chaque observation

⇒ estimée des espérances mathématiques $P(D_i|C_j)$ ou P_e en classification, $E\{(r-y)^2\}$ ou $E\{|r-y|\}$ en régression.

Théorème central-limite

Les estimées convergent en probabilité vers les vraies valeurs, avec un intervalle de confiance qui décroît en $\frac{1}{\sqrt{M}}$.

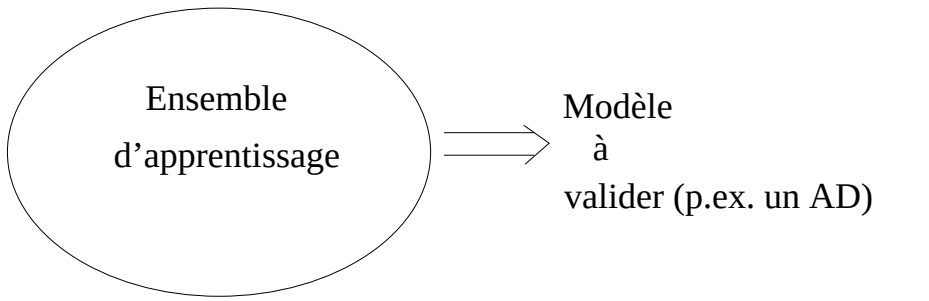
! Théorème applicable si les deux premiers moments existent...

Pour des estimées $P(D_i|C_j)$ ou P_e ça marche (Moivre-Laplace)

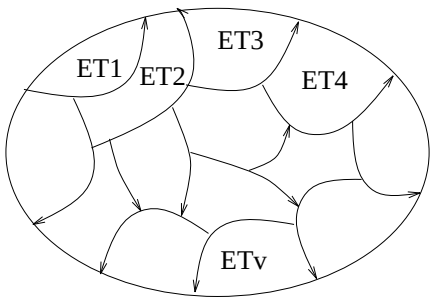
De plus on a $\hat{\sigma}_{\hat{P}} \approx \sqrt{\frac{\hat{P}(1-\hat{P})}{M}}$

⇒ En pratique, cela donne des estimées précises si $M \geq 1000$.

- Méthode de rotation (si pas d'assez de données...)



Partitionnement de l'EA en V parties de meme taille (N/V) ETi



Construction de V modèles auxiliaires (selon la méthode utilisée pour construire le modèle à valider)

Pour chacun EA i = EA \ ETi
testé sur ETi → Erreur i

Moyenne des Erreuri

Inconvénient : temps de calculs ($V \geq 10$) (Exception : KNN)
Cas particulier : $V{=}N$ (leave-one-out)

Partie II
Applications dans les réseaux électriques

- Sources de données
- Types d'applications
- Un exemple réel

Sources de données

- Capteurs à base de microprocesseurs
  Systèmes de protections numériques
  Compteurs
  Enregistreurs d'événements
  NB. Distribués dans le réseau électrique et en très grand nombre
- Bases de données d'archivage
  Centres de conduite
  Services clientèle, maintenance des équipements...
  NB. Hiérarchie de BD
- Données produites à partir de simulations
  Environnements de conception et d'exploitation

$\Rightarrow$ Volumes de données gigantesques, qui demandent à être exploités

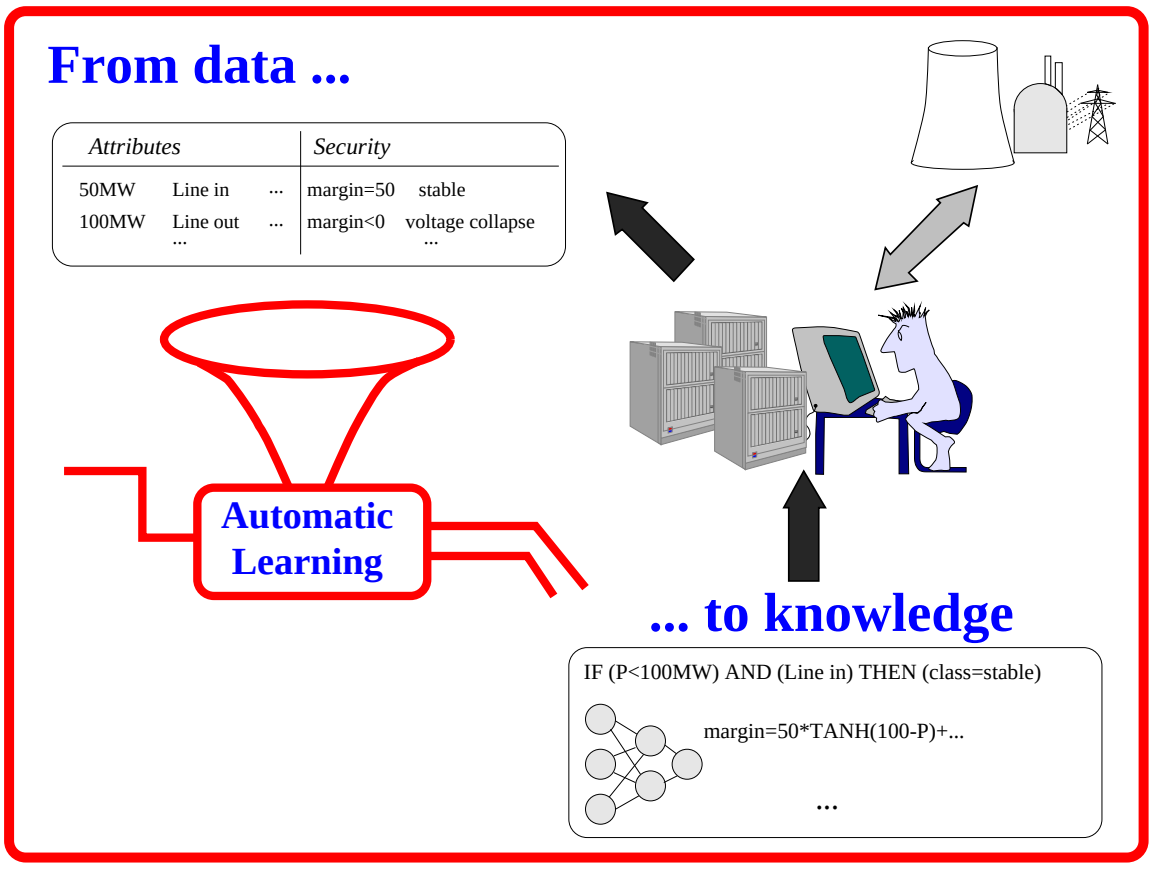
Types d'applications

- Conception
  Basée sur la simulation numérique (cf comportement non-linéaire)
  Exemples : réglage de protections; localisation de nouvelles centrales; valorisation économique de nouveaux équipements...

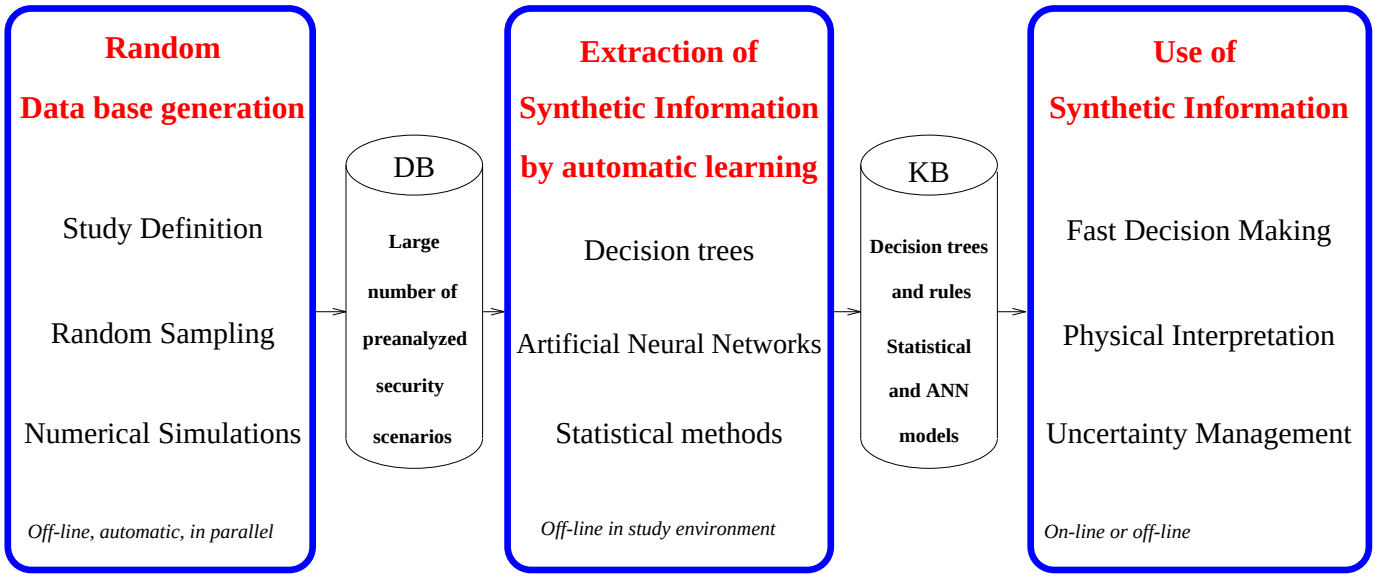
- Modélisation
  Lignes; Centreles; Agrégats de consommateurs; réseaux étrangers...

- Prédiction
  Consommation; hydraulicité; coûts des combustibles...

- Surveillance et détection de comportements anormaux
  Barrages; Réacteurs nucléaires; Arbres de machines tourantes; Transformateurs...



Approche pour l'évaluation de la sécurité



DB : data base KB : knowledge base

NB. Sécurité vis-à-vis des risques de perte de stabilité suite un incident

Data base generation

Preliminary remarks

- Security information DB : several 1000 security scenarios
- Quality of the DB : determines quality of extracted knowledge
- Sound methodology is needed : specification and validation
- Scenarios should be uncorrelated : to apply AL tools

NB. DB specification is time consuming.

But, if done properly once, may serve several times.

⇒ Let the computer do the job for which it is best suited...

What is a security scenario ?

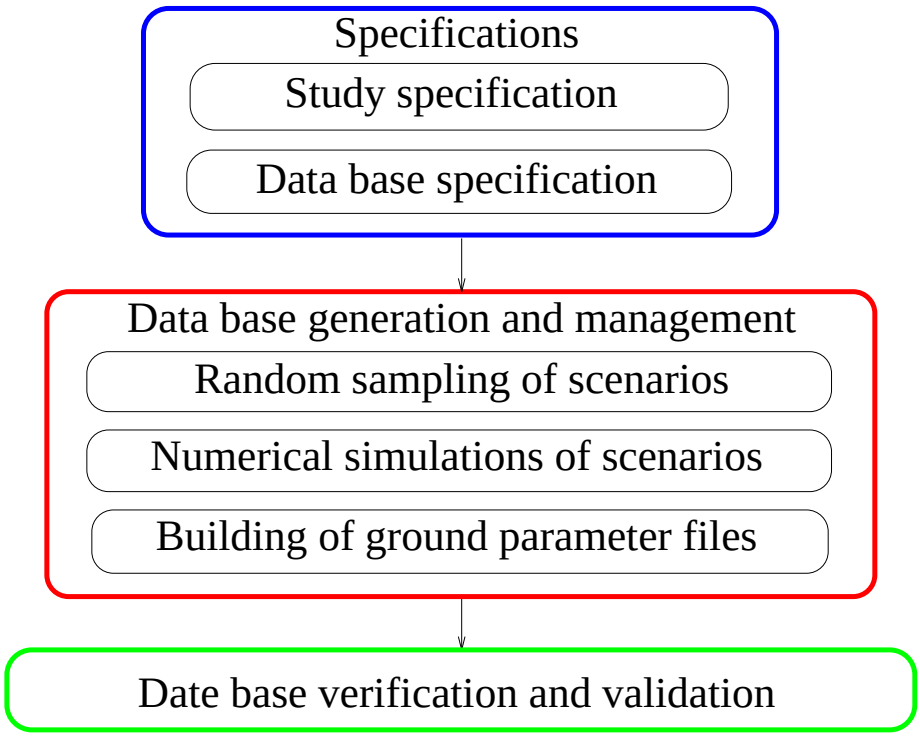
Three components

- Initial operating point (OP) :
  Equilibrium, defined by available equipments and their initial state
- External disturbances (ED) :
  Events which initiate dynamics (faults, load trends...)
- Dynamic modeling hypothesis (MH) :
  Assumptions on how the system is supposed to behave

NB.

1. All three may vary from one scenario to another (examples later)
2. How they vary depends on the objective of the security study

Overall data base generation process



Specifications

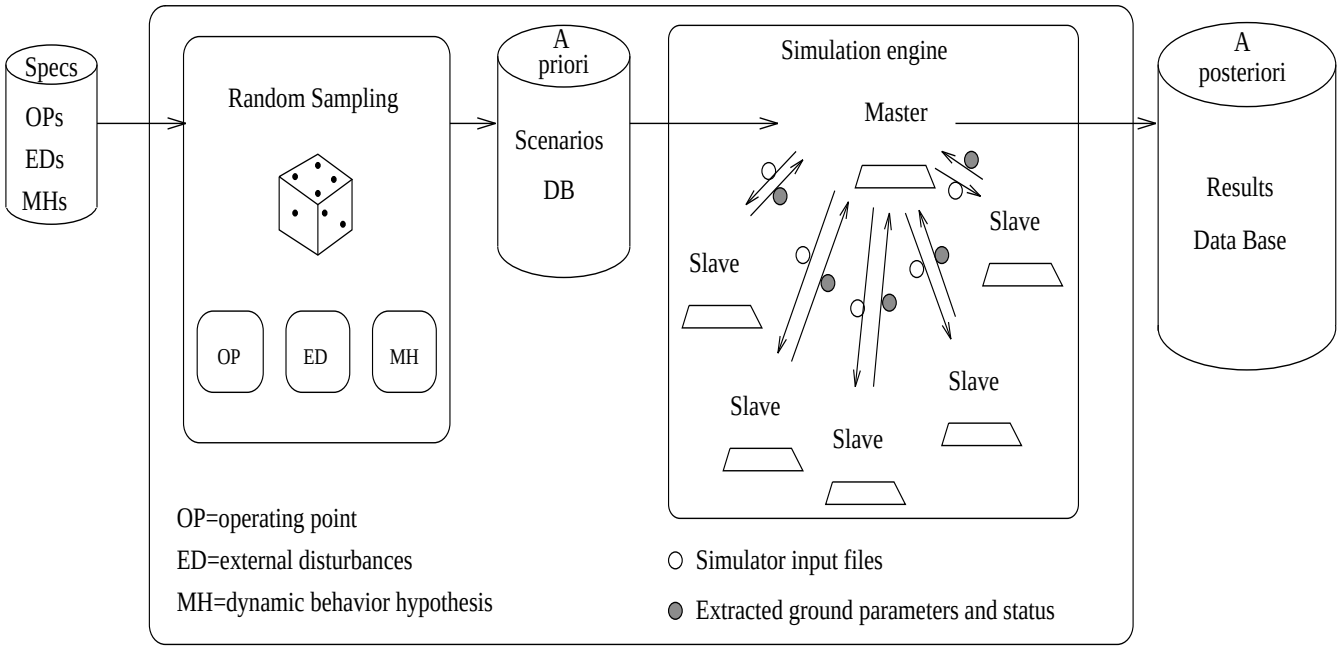
- Study scope

  Define objectives of the study and range of target conditions. Types of phenomena. Range of operating points and faults. Modeling assumptions.

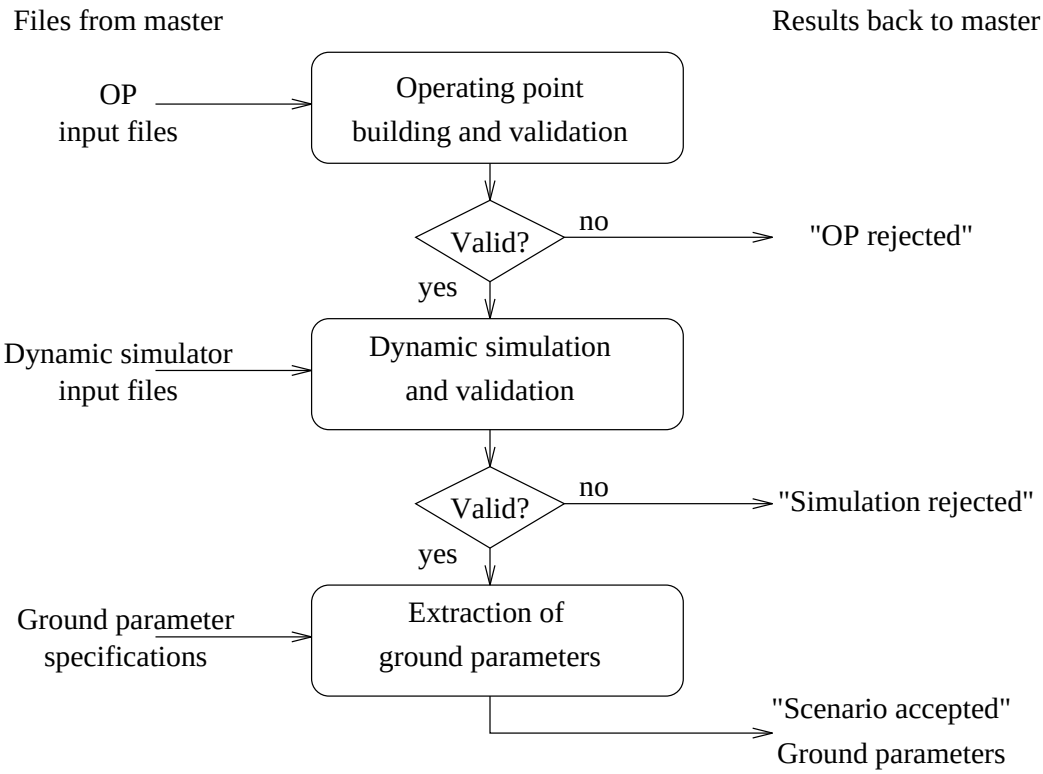
- Data base per se
  - Random sampling specifications (variables and probability distributions)
  - Extracted grounds parameters (attributes, security information)
  - Acceptability criteria and filtering
  - Number of scenarios to simulate

⇒ Many discussions with experts

Data base generation tool



Scenario simulation



Data base validation

Using low level data mining tools (e.g. visualizations)

Two steps

- A priori data base validation
  ⇒ Assess effect of filtering on random sampling distributions

- A posteriori data base validation
  ⇒ Become familiar with the data base
  ⇒ Assess whether the data base is rich enough

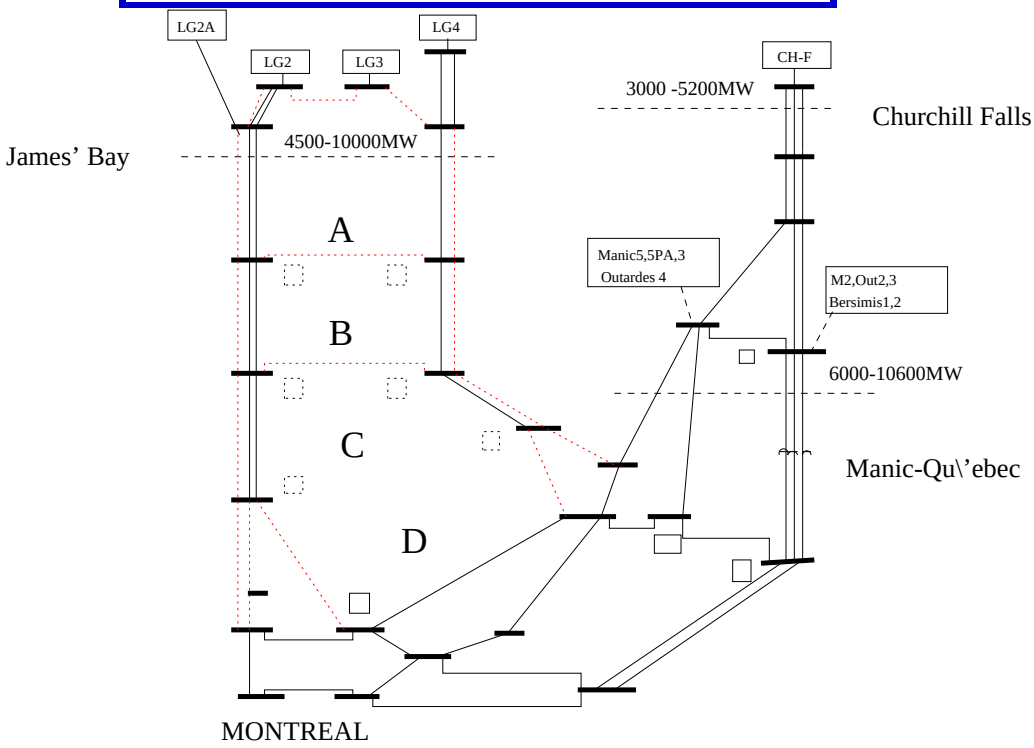
⇒ Decide to modify specifications, models, simulation tools

⇒ Accept the data base and proceed

Two large scale examples

- Hydro-Québec : transient stability limits
- Electricité de France : extreme scenarios

First example : Hydro-Québec



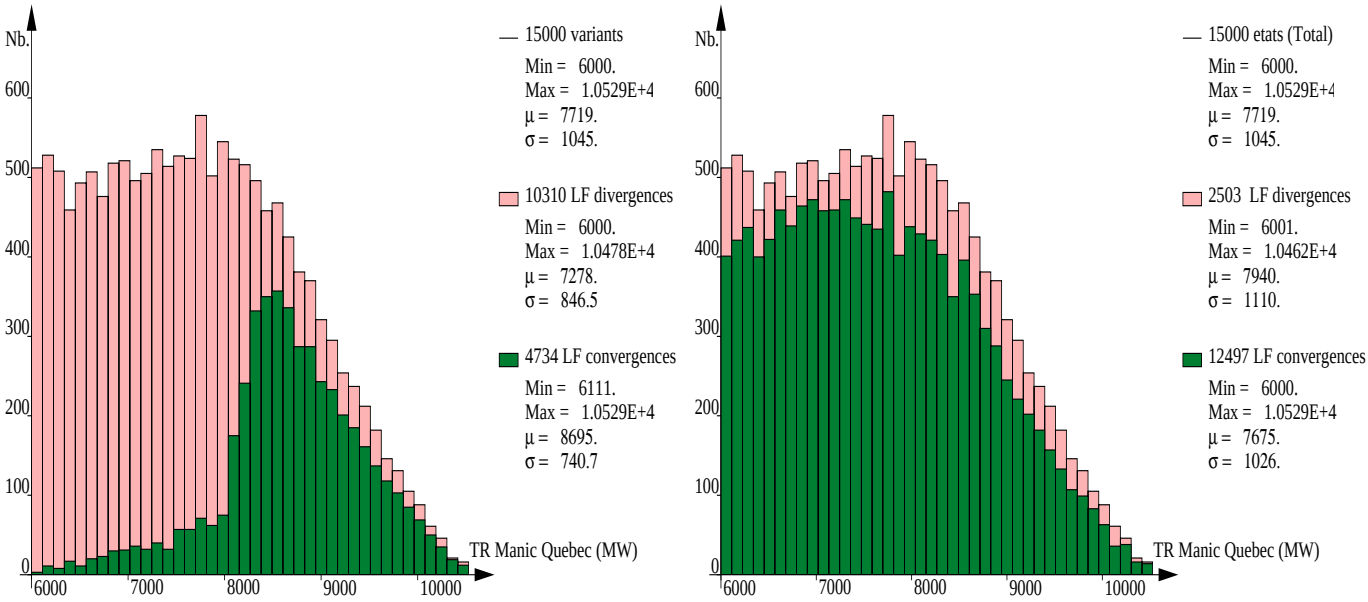
Transient stability limits of James Bay corridor.

Hydro-Québec DB specifications

- Global stability classification
  (Stable with respect to all single phase line faults in the corridor)
- Objective : express stability limits on power flows, as a function of topology and available var support
- About 12,500 different operating points
- 300 different topologies, large variations in power flows in various corridors, variable number of SVCs in operation
- 100 attributes :
  (Flows, generation, topology, number of SVCs in operation)
- States are preclassified by LIMSEL

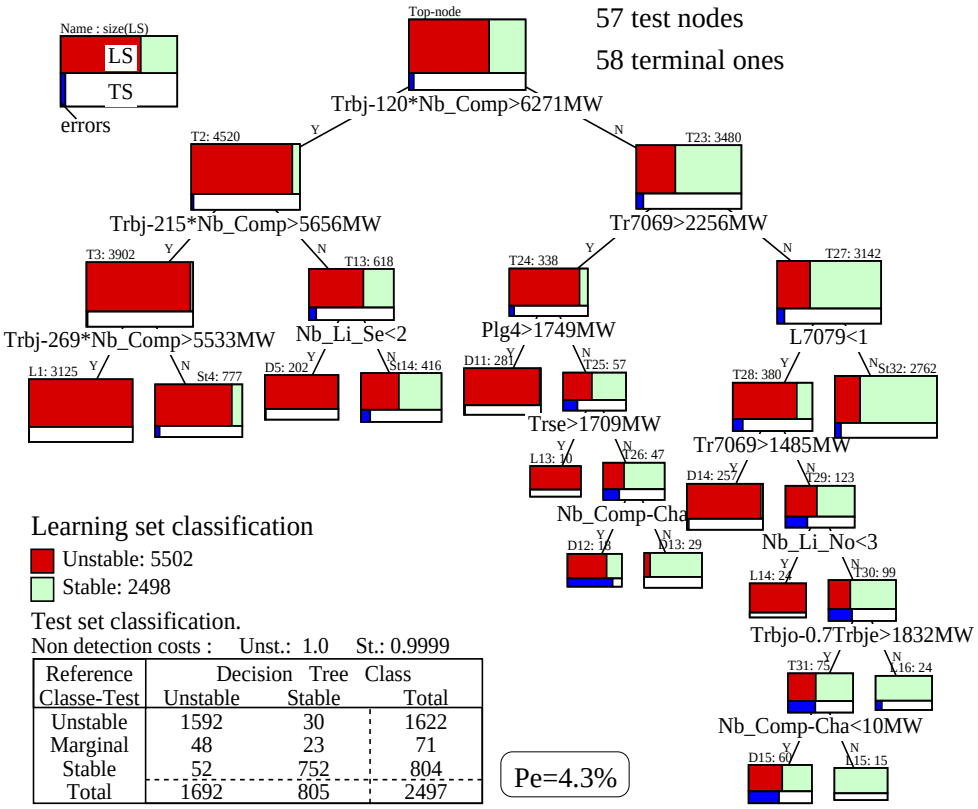
Illustration of filtering

Load-flow non convergence problem...



Needed several iterations to be fixed

Example global decision tree



57 test nodes

58 terminal ones

Learning set classification

Unstable: 5502

Stable: 2498

Test set classification.

Non detection costs : Unst.: 1.0 St.: 0.9999

Reference Classe-Test	Decision Tree Unstable	Stable	Class Total
Unstable	1592	30	1622
Marginal	48	23	71
Stable	52	752	804
Total	1692	805	2497

Pe=4.3%

Other AL methods

MLP: 2.4%

KNN: 5.7%

CPU Times (USparc 200MHz):

	Learn	Test/st
DT	22min	0.1ms
KNN	1.5hour	0.5s
MLP	16hours	1.2ms

Other byproducts

DT building identifies most relevant attributes

And evaluates their overall contribution in purification

Ex. building the above tree on the whole data base yields

TRBJ+B*NB_CO:	51.8	TR7069 :	9.7	L7079 :	6.2
TRSE :	5.8	NB_COMP :	4.6	TR7062 :	4.3
NB_LI_SE :	3.4	NB_LI_NE :	1.9	PLG4 :	1.9
L7090 :	1.8	NB_COMP-CHA :	1.6	TRSO :	1.2
CLASSE-BASE :	1.0	TR7094 :	1.0	PLG+B*TRBJ :	0.8
TRNEM :	0.6	NB_LI_NO :	0.5	TR7025 :	0.4
TRNO :	0.3	TRABI :	0.3	TRBJO+B*TRBJ:	0.3
TR7044 :	0.2	PLG3 :	0.2	TR7016 :	0.2

Second example : Electricité de France

Objective.

Build a data base representative of all possible failure modes of the South-Eastern part of the EDF system.

⇒ Model all kinds of slow and fast phenomena

⇒ Take into account all possible causes of failure

⇒ Multiple faults, protection mis-operation, uncertainties

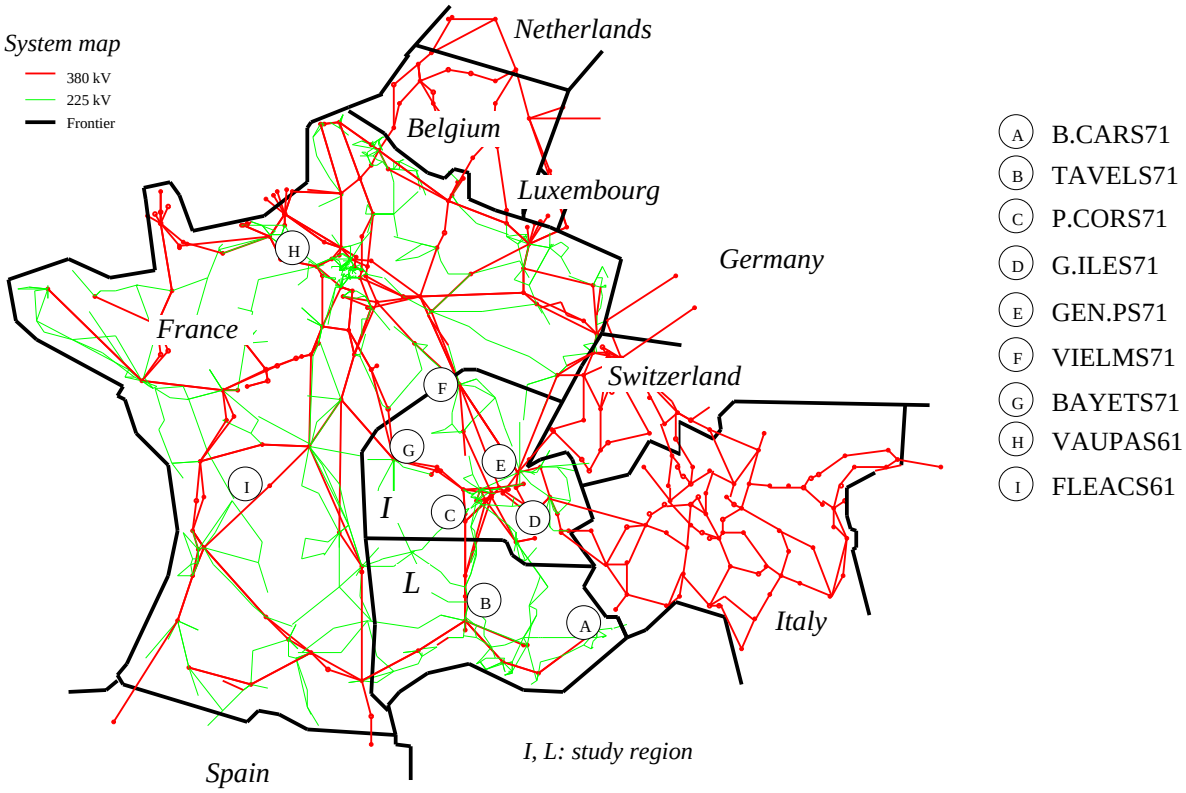
⇒ Data base will contain dynamic variables along system trajectory

⇒ Lots of temporal variables, sequences of events ...

⇒ Requires parallel computations to reduce DB generation time

⇒ Requires auxiliary software to extract relevant ground parameters from raw simulation results

EDF system



DB specifications

Three operating points selected manually (N-1 secure)

Concentrate on diversity of faults and modeling hypotheses

Generate about 1500 simulation scenarios

Extract about 800 temporal attributes

- Voltages, load levels
- Rotor angles, excitation currents, mechanical powers
- Power flows,
- Discrete events (action of various protections)

Resulting Data Base

Data base of about 1100 extreme scenarios on the EDF system

Each scenario is simulated during 40-50 minutes (very detailed model)

Attributes are temporal variables (curves)

800 attributes $\Rightarrow$ 1GB data base (sic)

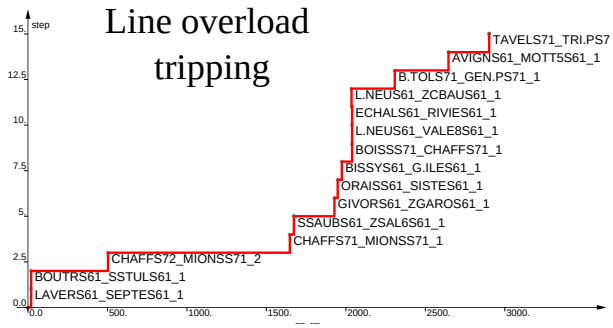
Various collapse phenomena :

- Line overloads (cascades)
- Voltage collapse (local, regional)
- Loss of synchronism (plant and area modes)

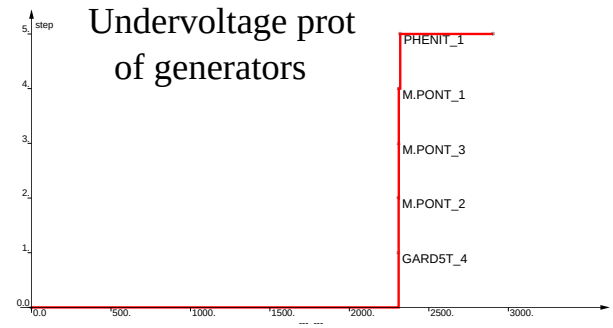
Some statistics of the resulting DB

CPU simulation time (s)	0	99000	38000	22000
time steps before interpolation	89	46800	3800	3071
time steps after interpolation	4	1728	145	137.2
size (Mb) before interpolation	4	2140	174	140
size (Kb) after interpolation and compression	32.4	3720	840	460
Salient scenario characteristics	Min	Max	Mean	σ
Nb of lines lost				
380 kV	0	48	5	6.3
225 kV	0	149	9.6	18.53
Thermal units				
Nb of units lost	0	15	1.15	2.1
mechanical power lost (MW)	0	13000	617	1213
mechanical power variation (EDF system, MW)	-20800	546	-1265	2445
Hydro units				
Nb of units lost	0	32	2.7	5.3
mechanical power lost	0	2952	271	584
mechanical power variation (EDF system, MW)	-3039	60.5	-332	604
Load variation (MW)				
I region	-9046	654	-864	1714
L region	-8944	288	-1194	2368
EDF system	-22000	426.8	-2417	4323

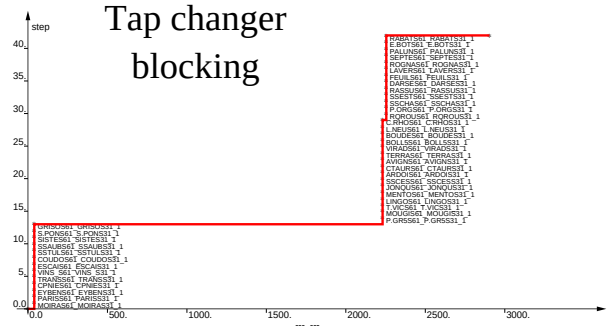
Line overload tripping



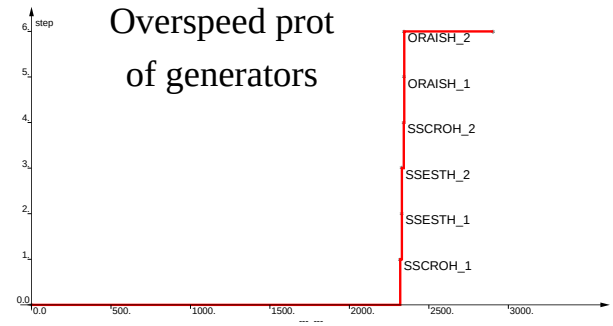
Undervoltage prot of generators

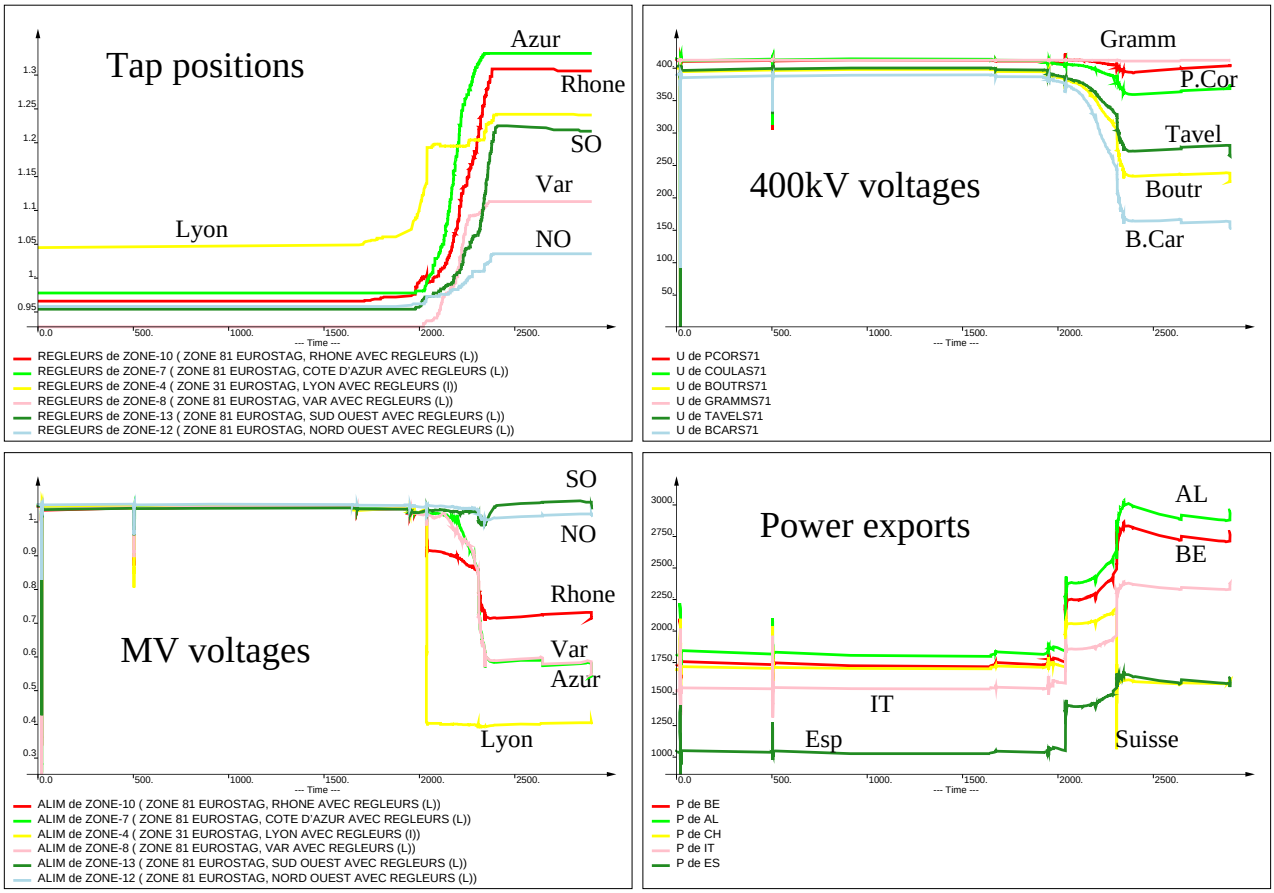


Tap changer blocking



Overspeed prot of generators





First application : grouping of scenarios

Attributes : characterize the scenarios in terms of consequences

- Number of lines and generators tripped
- Variation of active power generation and interface flows
- Amount of load lost in various regions
- Voltages at end of scenario in various places

K-means algorithm used to find groups of similar scenarios. Trial and error with various numbers of clusters, finally 7 classes :

- 702 stable scenarios (150MW load lost in the mean),
- 77 local losses of synchronism (2000MW),
- 90 local losses of load without collapse (2000MW),
- 90 local losses of load with local voltage collapses (2000MW),
- 113 regional voltage collapses (7400MW),
- 33 wide area voltage collapses (17000MW),
- 17 regional losses of synchronism (9500MW)

Second example : coherency of voltages

Consider EHV voltages (85 buses) and mean MV voltages of 13 load areas.

Take voltage values at the end of the simulation.

Compute correlation coefficients of all pairs of variables (4753 correlations, estimated from the 1100 learning states).

Hiearchical grouping of variables : first the most correlated a.s.o.

Build dendrogram (graphical representation of the grouping tree) and analyse visually

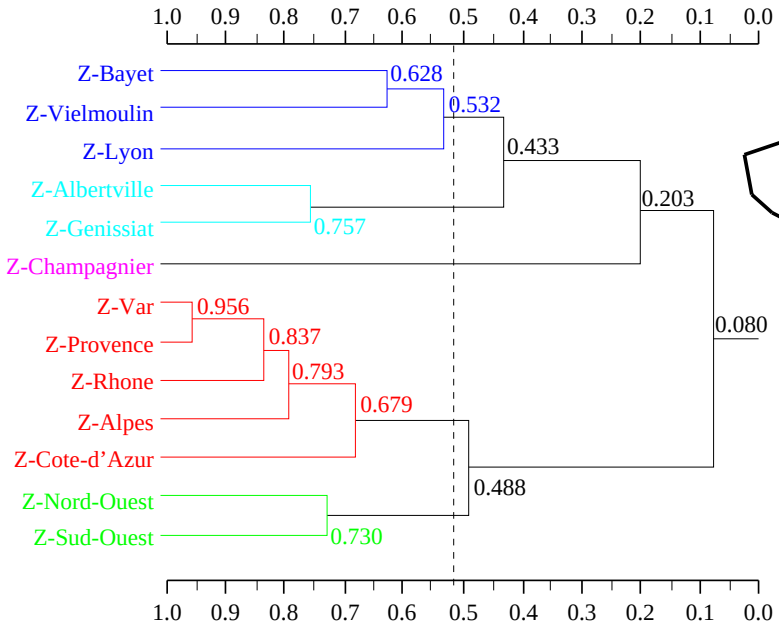
Find out zones of coherent behaviour :

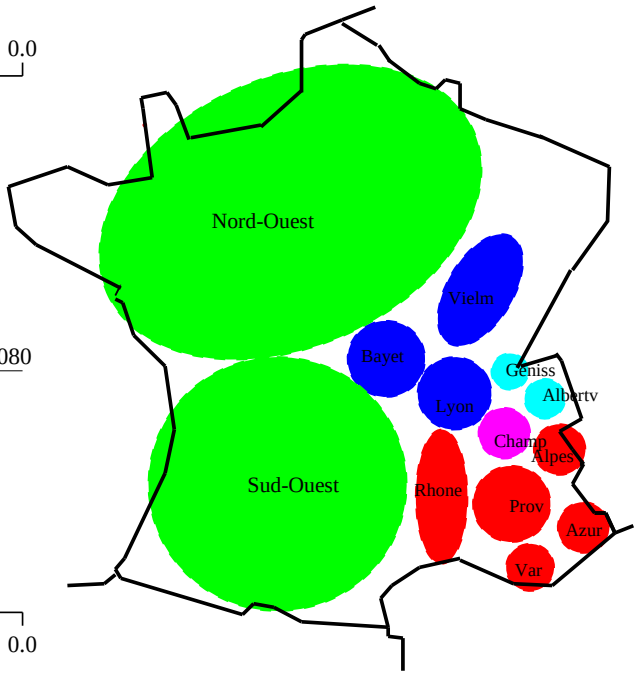
MV $\Rightarrow$ voltage collapse zones $\Rightarrow$ where to act to mitigate

EHV $\Rightarrow$ impact on EHV system $\Rightarrow$ where to put triggering signals

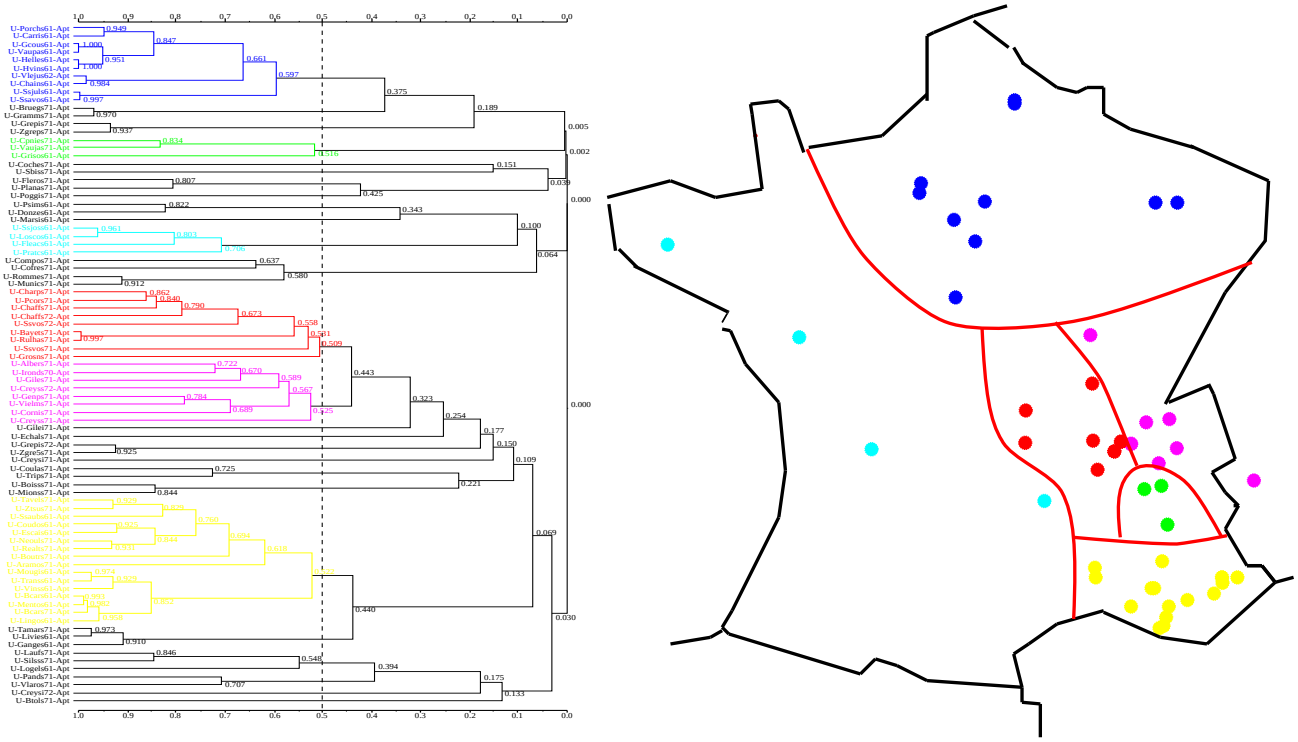
MV voltages at end of simulation

Dendrogram (correlations on 1100 scenarios)



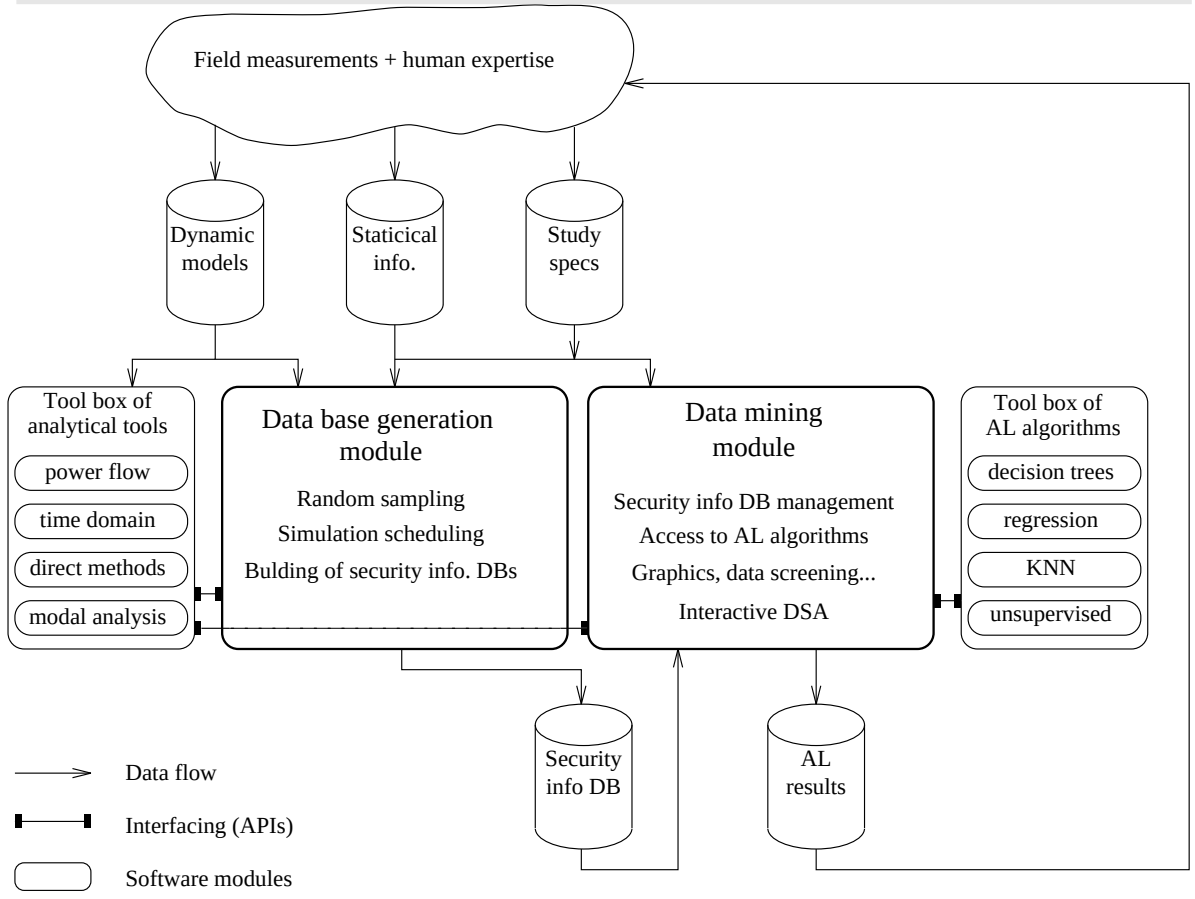


EHV voltages at some buses



Conclusions

- Applications très diverses dans les réseaux électriques
  - Méthodologie appliquée à divers problèmes réels
  - Besoin de validation par l'expert humain
  - Règles d'exploitation pour le passage de cet hiver ont été mises au point avec cette approche
- Approche
  - Peut s'appliquer à de nombreux problèmes similaires
- Développement d'outils industriels en projet



Quelques références bibliographiques pour en savoir plus.

L. Wehenkel.
Automatic learning techniques in power systems.
Kluwer Academic, Boston (Massachusetts), 1998.

L. Wehenkel.
Machine learning approaches to power-system security assessment.
IEEE Expert, Intelligent Systems & their Applications, 12(3), November 1997.

L. Wehenkel, C. Lebrevelec, M. Trotignon, and J. Batut.
A probabilistic approach to the design of power systems protection schemes against blackouts. In *Proc. IFAC-CIGRE Symp. on Control of Power Plants and Power Systems*, pages 506–511, Beijing, 1997.

X. Boyen and L. Wehenkel.
Fuzzy decision tree induction for power system security assessment.
In *Proc. of SIPOWER'95, 2nd IFAC Symp. on Control of Power Plants and Power Systems*, pages 151–156, Mexico, December 1995.

X. Boyen and L. Wehenkel.
Automatic induction of continuous decision trees.
In *Proc. of IPMU'96, Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pages 419–424, Granada, July 1996.

I. Houben, L. Wehenkel, and M. Pavella.
Coupling of K-NN with decision trees for power system transient stability assessment.
In *IEEE Conference on Control Applications*, pages 825–832, Albany (NJ), 1995.