Tanagra

Un logiciel de Data Mining gratuit pour l'enseignement et la recherche

Ricco RAKOTOMALALA

Université Lyon 2

Laboratoire ERIC

http://eric.univ-lyon2.fr/~ricco

Ricco – Qui est-il ? Que fait-il ?

Enseignant chercheur – CNU 27 – Informatique

Université Lumière Lyon 2

Culture Économétrie (Statistique)

Thèse Apprentissage automatique – Data Mining

- Arbres de décision, Sélection de variables, Échantillonnage, ...
- Applications *(classement de protéines, classement de planctons, reconnaissance de la langue, etc.)*

Développement et diffusion de logiciels libres (TANAGRA, *SIPINA*)

Rédaction et diffusion de didacticiels

Rédaction et diffusion de fascicules de cours

Plan

1. Data Mining
2. Pourquoi le logiciel libre dans le data mining
3. Tanagra – Spécification, développement, promotion
4. Et les autres logiciels libres ?

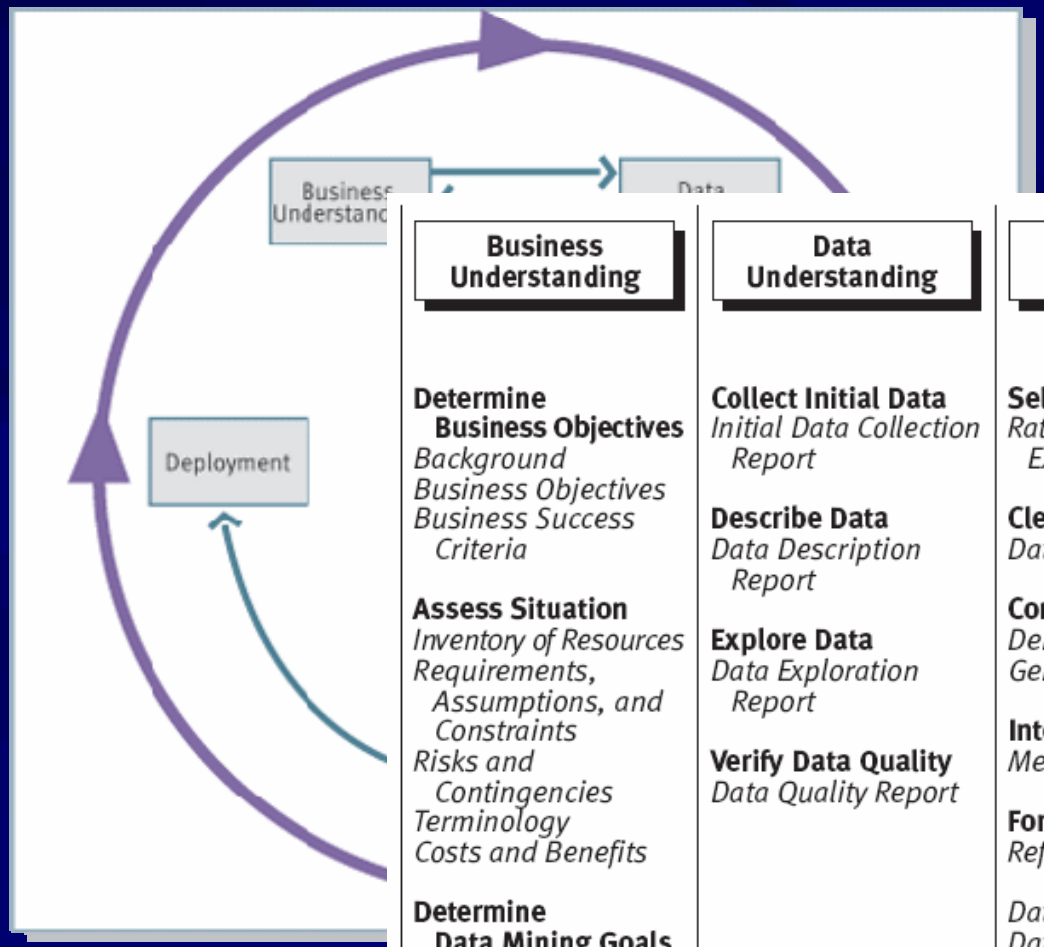
   Knime, Orange, R, RapidMiner, Weka,...

5. Démonstration
6. Conclusion

1. Data Mining

ECD : Extraction de connaissances à partir de données
(Knowledge Discovery in Databases)

Data Mining



CRISP DM, Step-by-step Data Mining Guide, SPSS Publication

Business Understanding	Data Understanding	Data Preparation	Modeling	Evaluation	Deployment
Determine Business Objectives *Background* *Business Objectives* *Business Success Criteria*	**Collect Initial Data** *Initial Data Collection Report*	**Select Data** *Rationale for Inclusion/ Exclusion*	**Select Modeling Techniques** *Modeling Technique* *Modeling Assumptions*	**Evaluate Results** *Assessment of Data Mining Results w.r.t. Business Success Criteria* *Approved Models*	**Plan Deployment** *Deployment Plan*
Assess Situation *Inventory of Resources* *Requirements, Assumptions, and Constraints* *Risks and Contingencies* *Terminology* *Costs and Benefits*	**Describe Data** *Data Description Report*	**Clean Data** *Data Cleaning Report*	**Generate Test Design** *Test Design*	**Review Process** *Review of Process*	**Plan Monitoring and Maintenance** *Monitoring and Maintenance Plan*
Determine Data Mining Goals *Data Mining Goals* *Data Mining Success Criteria*	**Explore Data** *Data Exploration Report*	**Construct Data** *Derived Attributes* *Generated Records*	**Build Model** *Parameter Settings* *Models* *Model Descriptions*	**Determine Next Steps** *List of Possible Actions* *Decision*	**Produce Final Report** *Final Report* *Final Presentation*
Produce Project Plan *Project Plan* *Initial Assessment of Tools and Techniques*	**Verify Data Quality** *Data Quality Report*	**Integrate Data** *Merged Data*	**Assess Model** *Model Assessment* *Revised Parameter Settings*		**Review Project** *Experience Documentation*
		Format Data *Reformatted Data* *Dataset* *Dataset Description*			

Data Mining – Est-ce vraiment novateur ?

Définition (Fayyad, 1996) : Processus non trivial d'identification des structures inconnues, valides et potentiellement exploitables dans les bases de données.

Data Mining : Une nouvelle façon de faire de la statistique ? (G. Saporta)

L'analyse des données est un outil pour dégager de la gangue des données le pur diamant de la véridique nature.» (J.P.Benzécri1973)

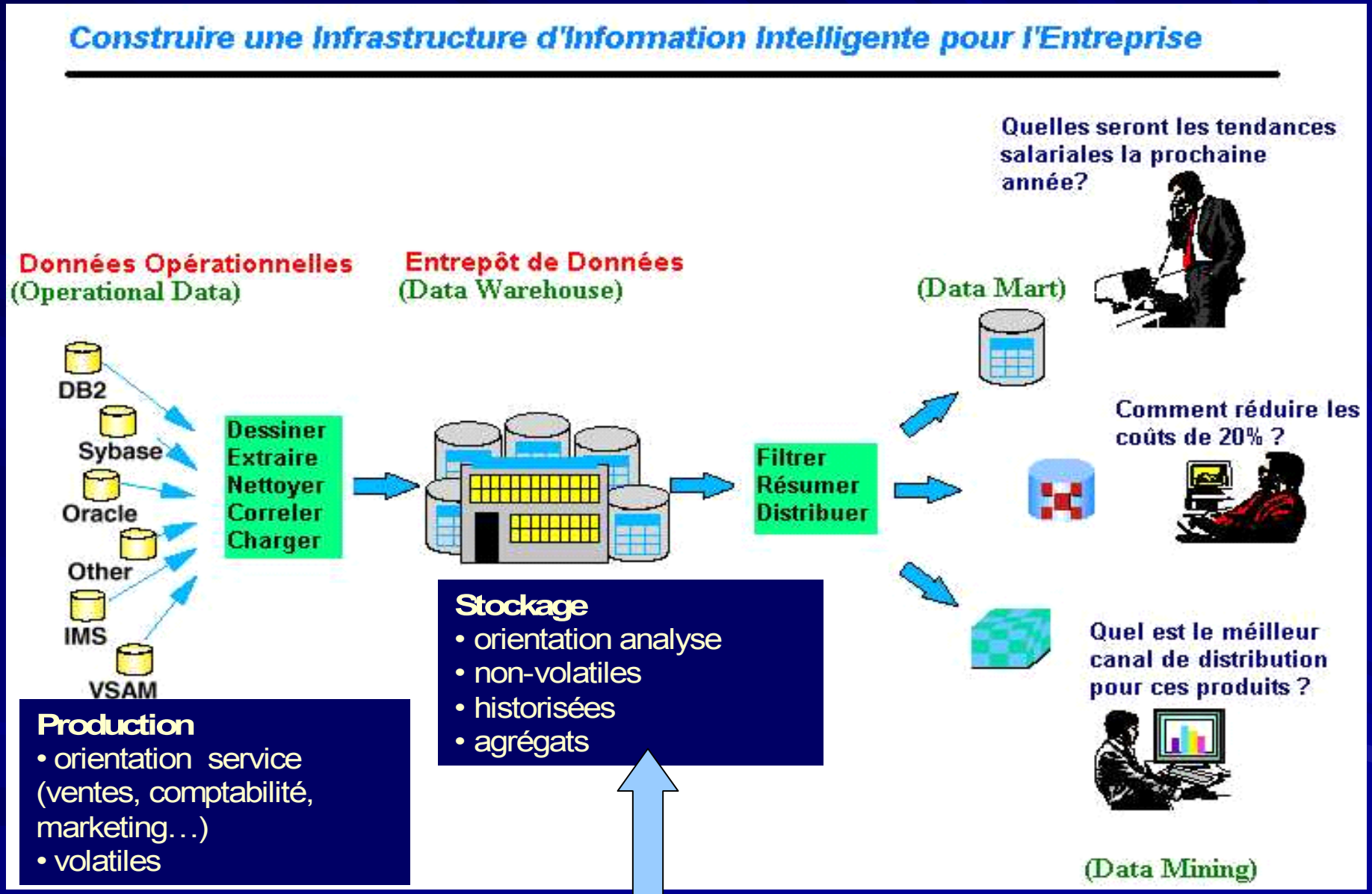
The basic steps for developing an effective process model ?

http://www.itl.nist.gov/div898/handbook/pmd/section4/pmd41.htm

1. Model selection
2. Model fitting
3. Model validation

Travailler sur des entrepôts de données

Faire partie intégrante du flux d'informations dans l'entreprise



Problème de volumétrie

Mixer des techniques d'horizons différents

Apprentissage automatique, Reconnaissance de formes, Statistique, Analyse de données, ...

Statistiques

Théorie de l'estimation, tests
Économétrie

Maximum de vraisemblance et moindres carrés
Régression logistique, …

Analyse de données (Statistique exploratoire)

Description factorielle
Discrimination
Clustering

Méthodes géométriques, probabilités
ACP, ACM, Analyse discriminante, CAH, …

	var 1	var 2	…	var J
individu 1				
individu 2		valeurs		
…				
individu n				

Le point de ralliement est le tableau « attribut valeur ».

Même si, de plus en plus, on essaie d'aller plus loin en traitant des tableaux de structure plus élaborées, plus adaptées au problème à traiter (ex. données transactionnelles, données séquentielles, données complexes en XML, etc.)

Informatique (Intelligence artificielle)

Apprentissage symbolique
Reconnaissance de formes

Une étape de l'intelligence artificielle
Réseaux de neurones, algorithmes génétiques…

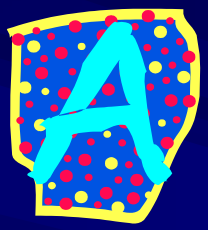
Informatique (Base de données)

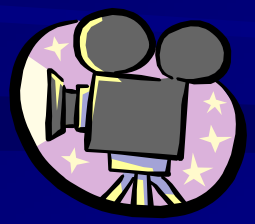
Exploration des bases de données

Volumétrie
Règles d'association, motifs fréquents, …

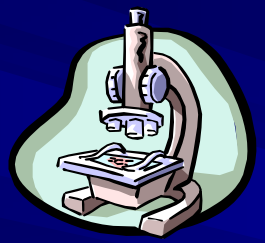
Traitement des données non structurées

Textes, images, etc... autre que le simple « attribut-valeur »





Rôle fondamental de la préparation des données



	var 1	var 2	...	var J
individu 1				
individu 2		valeurs		
...				
individu n				

Prédiction
Structuration
Description
Association

<u>Les applications</u>

Filtrage automatique des e-mails (spams, terrorisme,...)
Reconnaissance de la langue à une centrale téléphonique
Analyse des mammographies
Etc.

2. Data Mining et logiciel libre

Attention, les informaticiens (et internet) arrivent...

Quel espace pour les logiciels libres ?
Aspects du data mining prolifiques en développement

Développer des méthodes au cœur des entrepôts de données

Les B.D. sont surtout intéressés par le développement des plate-formes B.I.

Proximité très (trop) forte avec les applications industrielles (ORACLE, SQL-Server…)

Développement lourds, peu valorisants pour l' « apprentissage automatique » (publications)

Récupération d'outils existants. Ex. intégration de WEKA dans PENTAHO...

Traitement des données non structurées

Trop spécifique – Impossible de développer un outil générique

Proximité des applications industrielles

Développer des outils génériques de traitement de données

Intégrer des méthodes avec des finalités (origines) différentes

Pouvoir les faire coopérer entre elles

Tester et diffuser une nouvelle méthode publiée

Développement de plate-forme peu onéreuse, c'est le développement des algorithmes de traitements qui est difficile *(ex. RAPIDMINER et KNIME reposent en partie sur le moteur WEKA)*

Quel public pour le logiciel libre de data mining ?

Qui sont les utilisateurs, quels sont leurs besoins ?

Un logiciel pour l'enseignement et le profil « utilisateur - praticien » du Data Mining

Les cours, explication des méthodes, outil pédagogique

Illustrer les techniques en cours, les mettre en oeuvre en TD

Sans connaissances spécifiques (langage de prog., etc.) - Former sur le fond et non la forme

Avec un niveau de qualité conforme aux « standards » du domaine

Les études « réelles » - les « dossiers » - les chercheurs des autres domaines (biologie, médecine, etc.)

Une plate-forme pour la recherche

Plate-forme d'expérimentation pour tests à grande échelle

Implémenter ses méthodes (et les tester)

Les comparer (toutes choses égales par ailleurs i.e. dans le même environnement)

Les diffuser (pour d'autres, à des fins d'expérimentation, de comparaison)

Une publication n'est crédible que si reproductible (données, outils)

Un outil pédagogique pour l'apprentissage de la programmation

Spécifications et conception de ce type de logiciel - Apprendre par l'exemple

Connaître les outils et les bibliothèques types

Sujets de stages pour les étudiants

Logiciel libre de data mining

Pourquoi (accès) libre ? Pourquoi *open source* ?

Protéger les chercheurs, protéger les utilisateurs

A qui appartient un logiciel développé par un enseignant-chercheur ?

Est-ce le même statut que pour les ouvrages ?

Pourvoir développer sans contraintes (chercheur)

Pouvoir utiliser sans mauvaises surprises (utilisateur)

Diffusion du logiciel = valider les publications

Logiciels accessibles à tous → Comparaison et vérification des résultats

Reproduire « exactement » les expérimentations

Comparer le code = comparer les implémentations

Comparer les interprétations d'un même problème (ex. Relieff WEKA)

Lecture du code par d'autres chercheurs (ex. Naive Bayes classifier)

Optimiser le code avec différentes versions

Outil ouvert = Outil vivant

Introduire ses propres algorithmes

Discuter sur la base de prototypes et d'évolutions

Monter et partager des bibliothèques types (ex. générateurs aléatoires, fonctions de répartitions, les fameux packages...)

Logiciel de data mining
Quelles fonctionnalités implémenter

Accès et préparation des données
Accéder à un fichier / une BD
Rassembler des sources différentes

Méthodes de Fouille de données
Lancer les calculs avec différents algorithmes
Bibliothèque de méthodes

Enchaîner les traitements
Faire coopérer les méthodes sans programmer

Évaluer les connaissances
Validation croisée, etc.

Exploiter les sorties
Rapports, visualisation interactive, etc.

Appliquer/exploiter les modèles
Modèles en XML (PMML), code C, DLL compilées
Prédiction directe sur de nouveaux fichiers

Logiciels commerciaux
Prototypes de recherche

Logiciel de data mining
Quel mode opératoire ?

Logiciels pilotés par menu (STATISTICA, OPEN STAT, SIPINA, ...)

(+) Organisation de type « tableur »

(+) Rapidité de prise en main

(-) Enchaînement « à la main » des traitements

(-) Pas de trace des opérations effectuées

(-) Et donc reproductibilité difficile des traitements

Ligne de commande (SAS, S-PLUS, R, ...)

(+) Souplesse et puissance de la programmation

(+) Sauvegarde des traitements, reproductilibité

(-) Apprentissage d'un langage

Filière (diagramme de traitements) – Estampillé « Data Mining »

(+) Programmation « visuelle » - Pas d'apprentissage

(+) Enchaînement des traitements

(+) Sauvegarde des traitements, reproductilibité

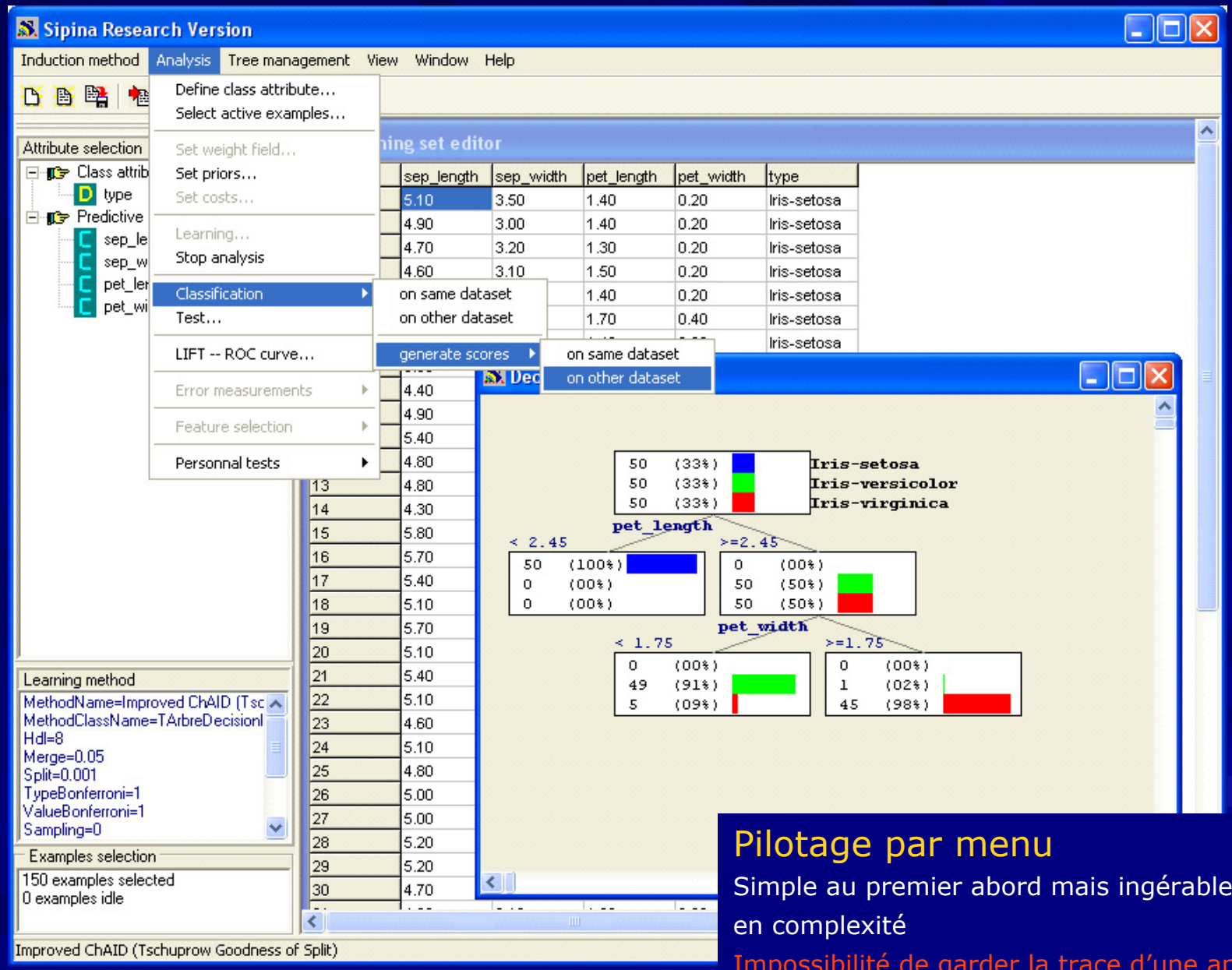
(-) Pas la puissance d'un « vrai » langage de programmation

SPAD, SAS Enterprise Miner, SPSS Clementine, S-PLUS Insightfull Miner, STATISTICA Data Miner, ...

KNIME, ORANGE, RAPIDMINER, TANAGRA, WEKA

Exemple de pilotage par menu

Sipina



Pilotage par menu

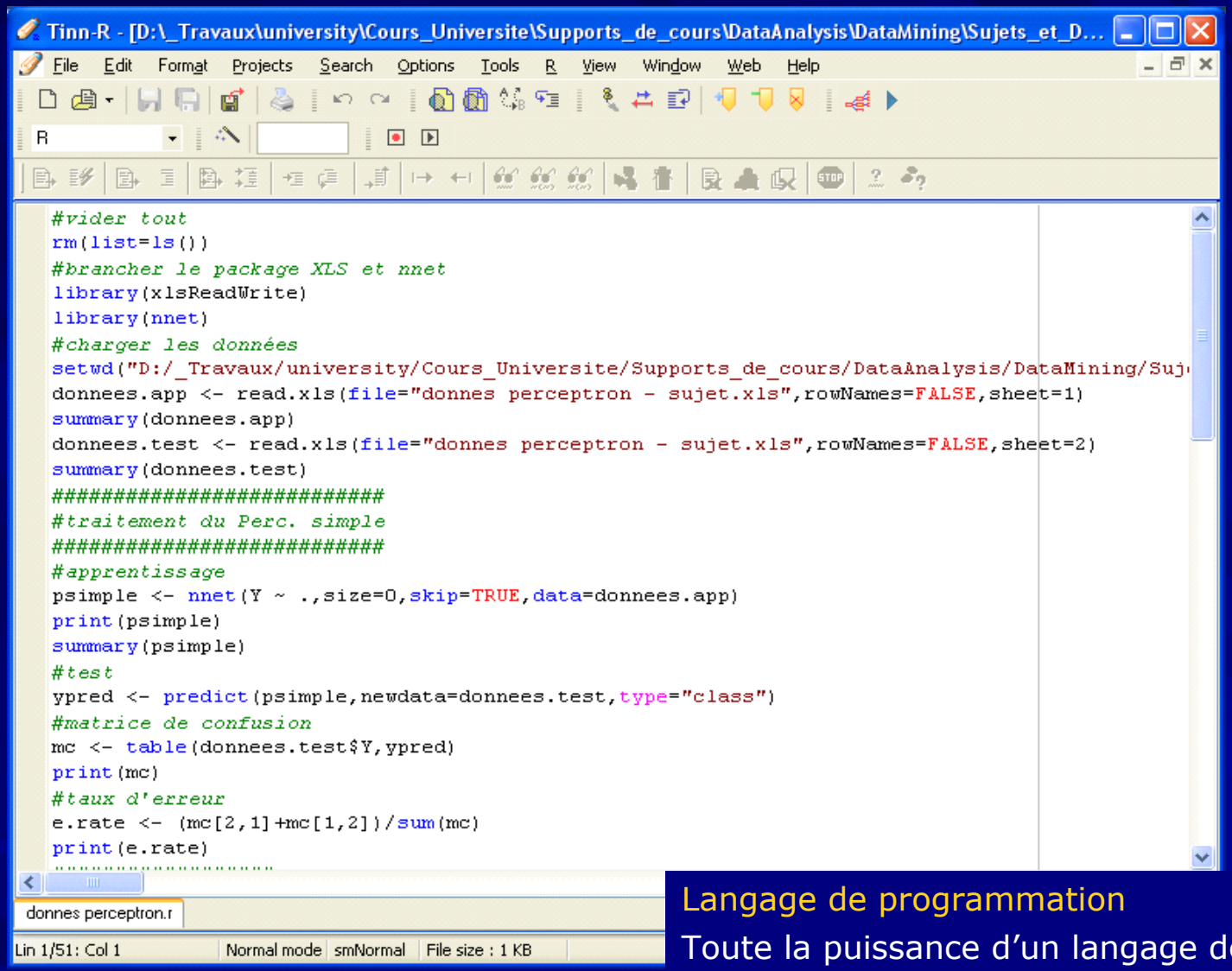
Simple au premier abord mais ingérable dès que le logiciel gagne en complexité

Impossibilité de garder la trace d'une analyse complète et donc de la reproduire

Exige une documentation complète et constamment à jour (Open Stat & Stat 4U sont dans la même situation)

Exemple de ligne de commande + langage de programmation

R



Langage de programmation

Toute la puissance d'un langage de programmation

L'accès au langage est une barrière à l'entrée qui rebute certains

Exemple de diagramme de traitements

Tanagra

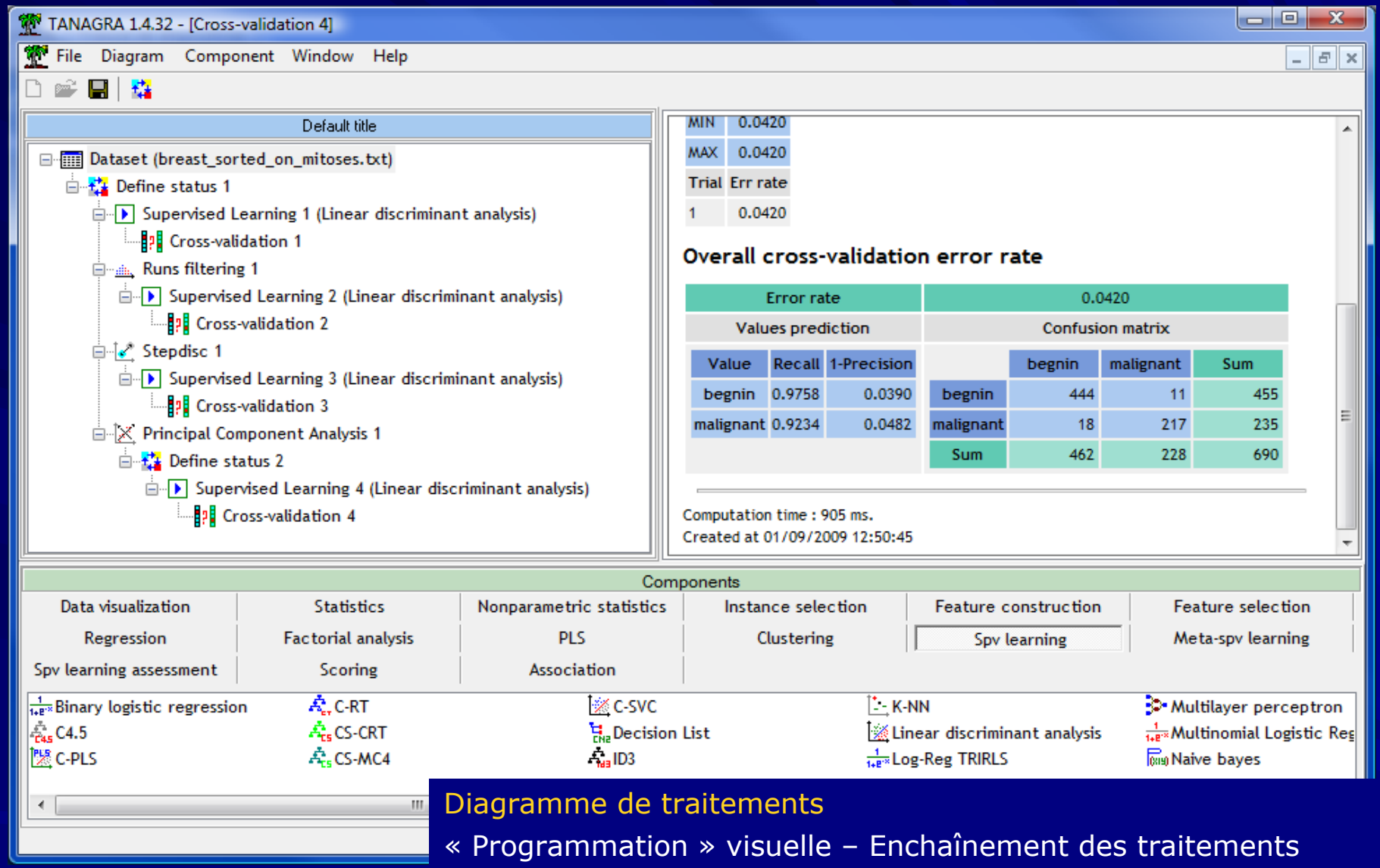


Diagramme de traitements

« Programmation » visuelle – Enchaînement des traitements

Mais pas toutes les fonctionnalités d'un langage de programmation

Mise à jour facilitée par adjonction de composants

Garder une trace de l'analyse et pouvoir la sauvegarder

Possibilité de fragmenter la documentation par « composants »

C'est le standard actuel

Exemple de diagramme de traitements

Knime

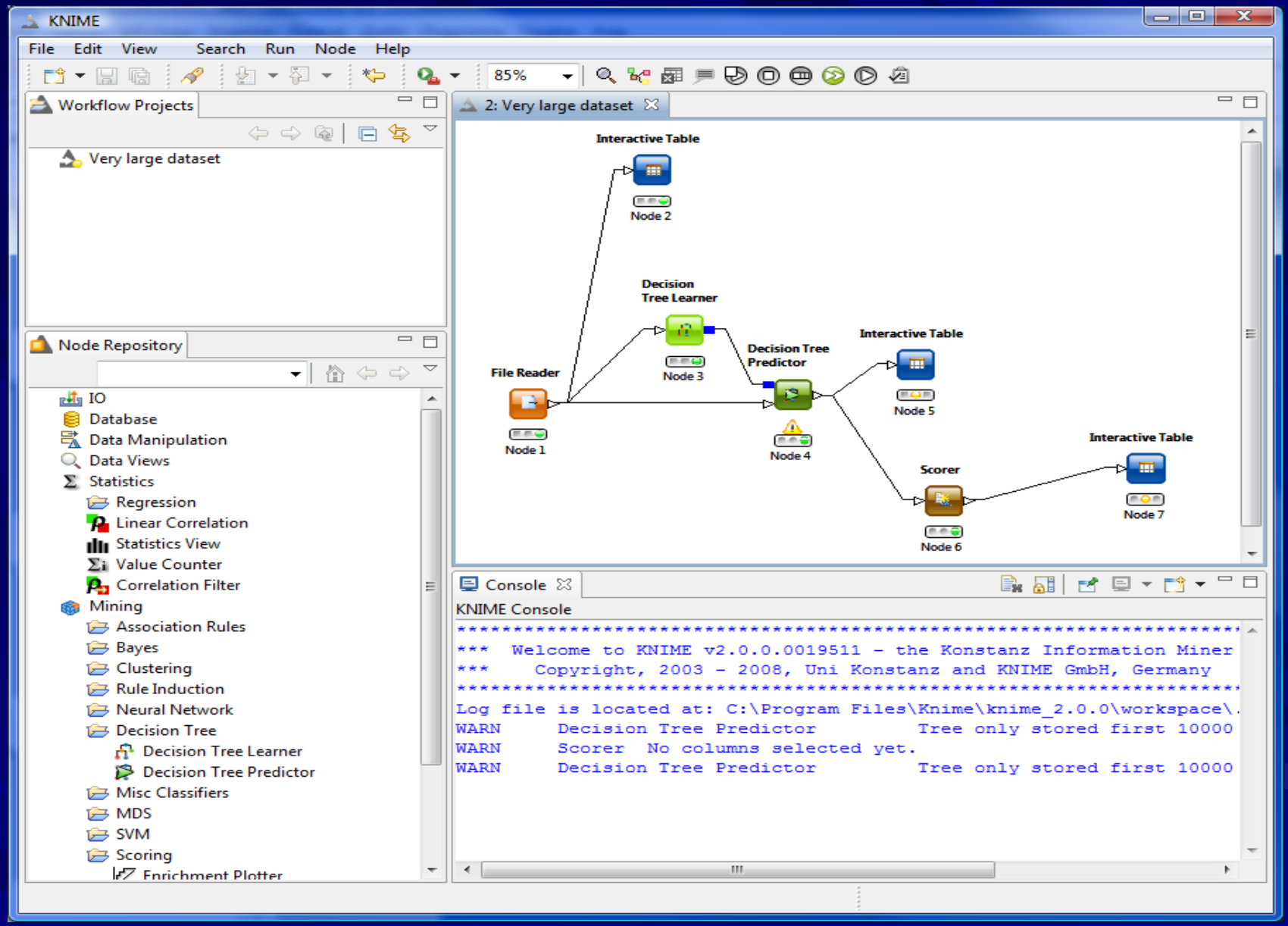


Diagramme de traitements

Une autre manière de présenter les diagrammes de traitements

KNIME est un des très rares à savoir représenter une boucle (notion de « méta composant »)

3. Tanagra

Tanagra

Définir un cahier des charges aussi précis que possible

Miser sur la simplicité d'utilisation

Installation simplifiée – Pas de serveurs lourds à installer

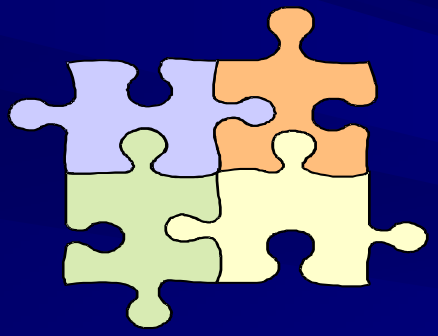
Gestion simplifiée des données - Format texte et accès au format tableur

Fonctionnement par diagramme de traitements

Couvrir les statistiques, l'analyse de données et le data mining. De manière unifiée.

Résultats lisibles, en adéquation avec les « standards »

Interfaçage avec les tableurs (Excel, Open Office Calc)



Mettre définitivement de côté les aspects « professionnels »

Interfaçage fort avec les SGBD

Déploiement et mise en production des résultats

Reporting dynamique et performant

Exploration graphique évoluée et interactive des données

Simplicité également pour le programmeur

Simplifier à l'extrême le code permettant d'ajouter une nouvelle méthode d'analyse

Minimiser le code dédié à la gestion des données et de l'interface

Pouvoir intégrer facilement n'importe quelle technique traitant des tableaux « individus x variables »

Simplicité pour les utilisateurs
Installation simplifiée et automatisée

Tout doit être automatisé

L'utilisateur ne doit jamais avoir à intervenir à l'installation

Attention aux bibliothèques externes (SGBD, TCL/TK, PYTHON, etc.)

Choisir la configuration au pire cas

Exclure les bibliothèques externes compilées / payantes

Bibliothèque externe compilée = dépendance accrue

Bibliothèque payante = pieds et poings liés (y compris sur les architectures)

Miser sur des versions stables et sources libres

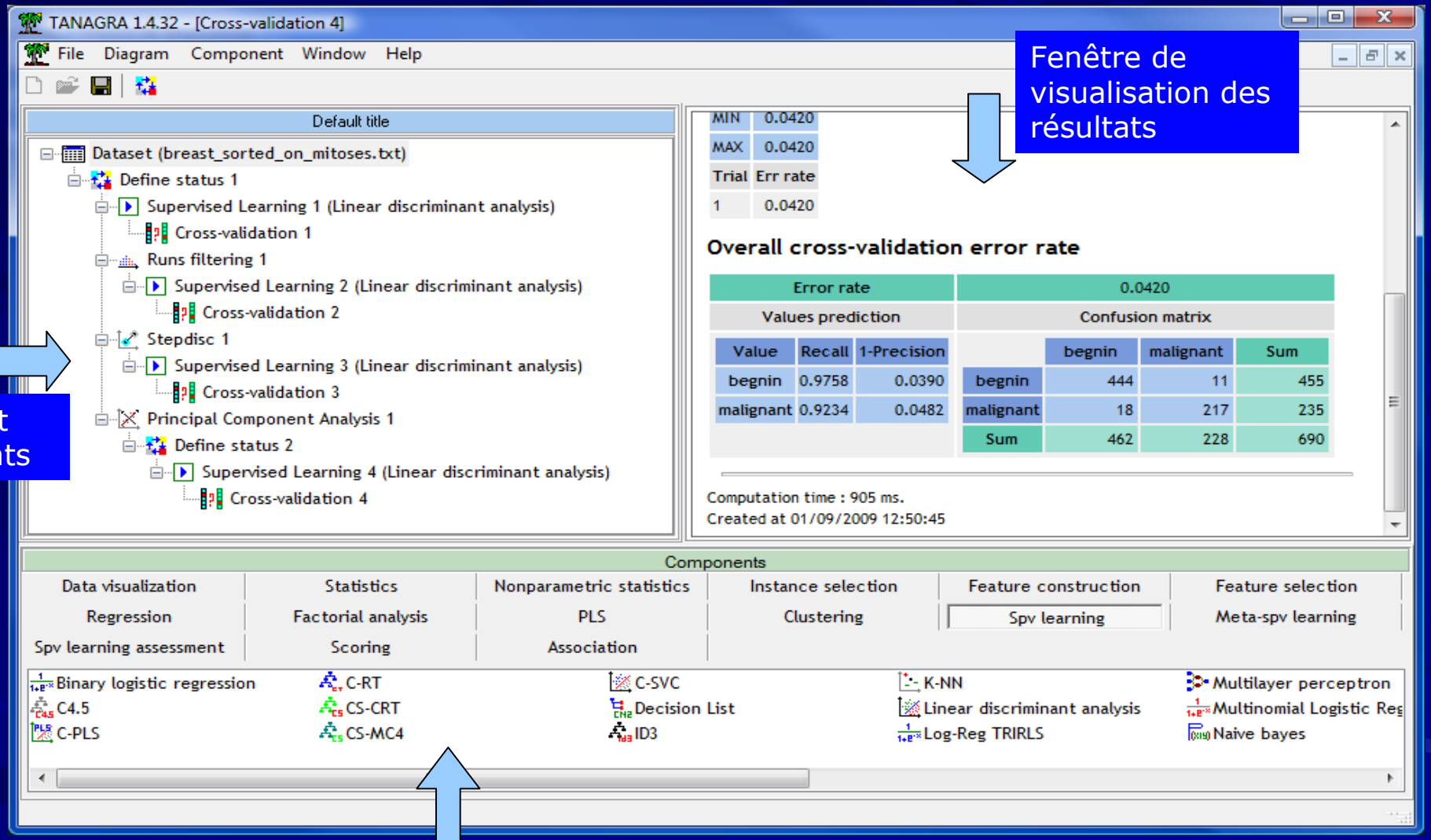
Attention à la gestion des mises à jour

Mettre des exemples de traitements (ex. RapidMiner)

L'utilisateur lance toujours « pour voir » sans lire la documentation

Simplicité pour les utilisateurs

Définir les traitements

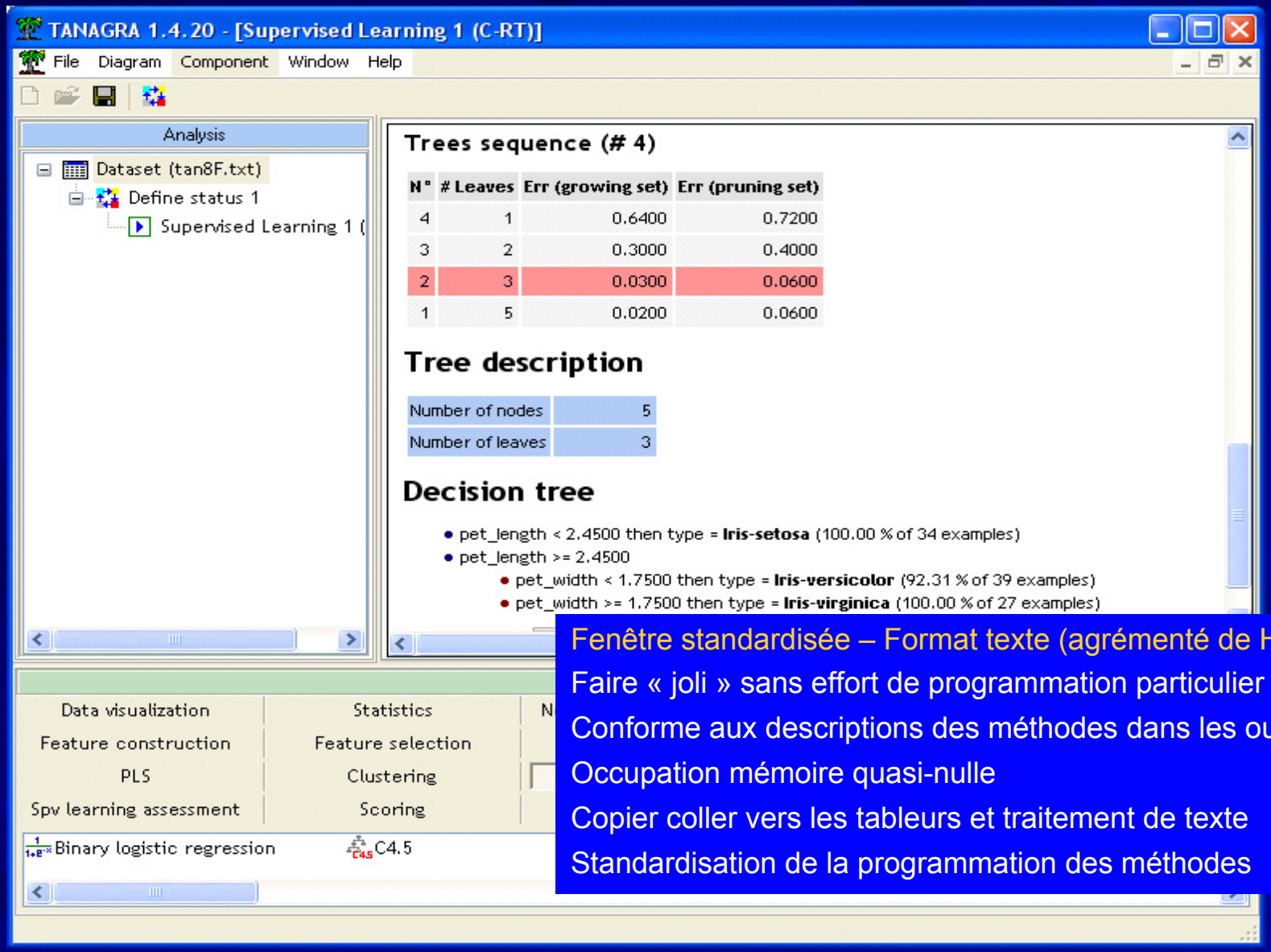


Fenêtre de visualisation des résultats

Enchaînement des traitements

Composants de calcul
Méthodes de data mining

Simplicité pour les utilisateurs
Standardisation des affichages



Fenêtre standardisée – Format texte (agrémenté de HTML)

Faire « joli » sans effort de programmation particulier

Conforme aux descriptions des méthodes dans les ouvrages

Occupation mémoire quasi-nulle

Copier coller vers les tableurs et traitement de texte

Standardisation de la programmation des méthodes

Simplicité pour les programmeurs
Vive la programmation objet

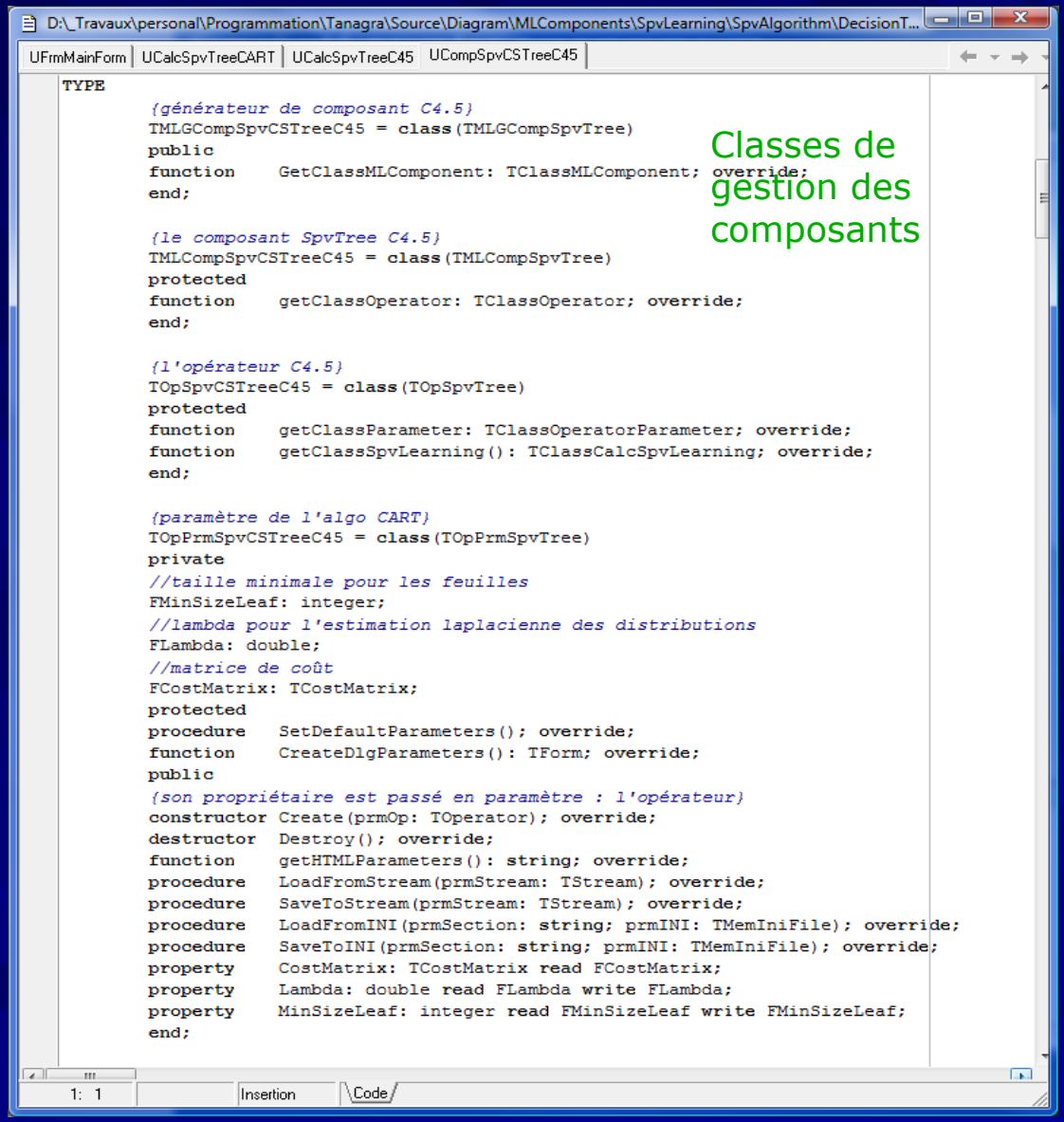
D:_Travaux\personal\Programmation\Tanagra\Source\Diagram\MLComponents\SpvLearning\SpvAlgorithm\DecisionTree\UCalcSpvTre...

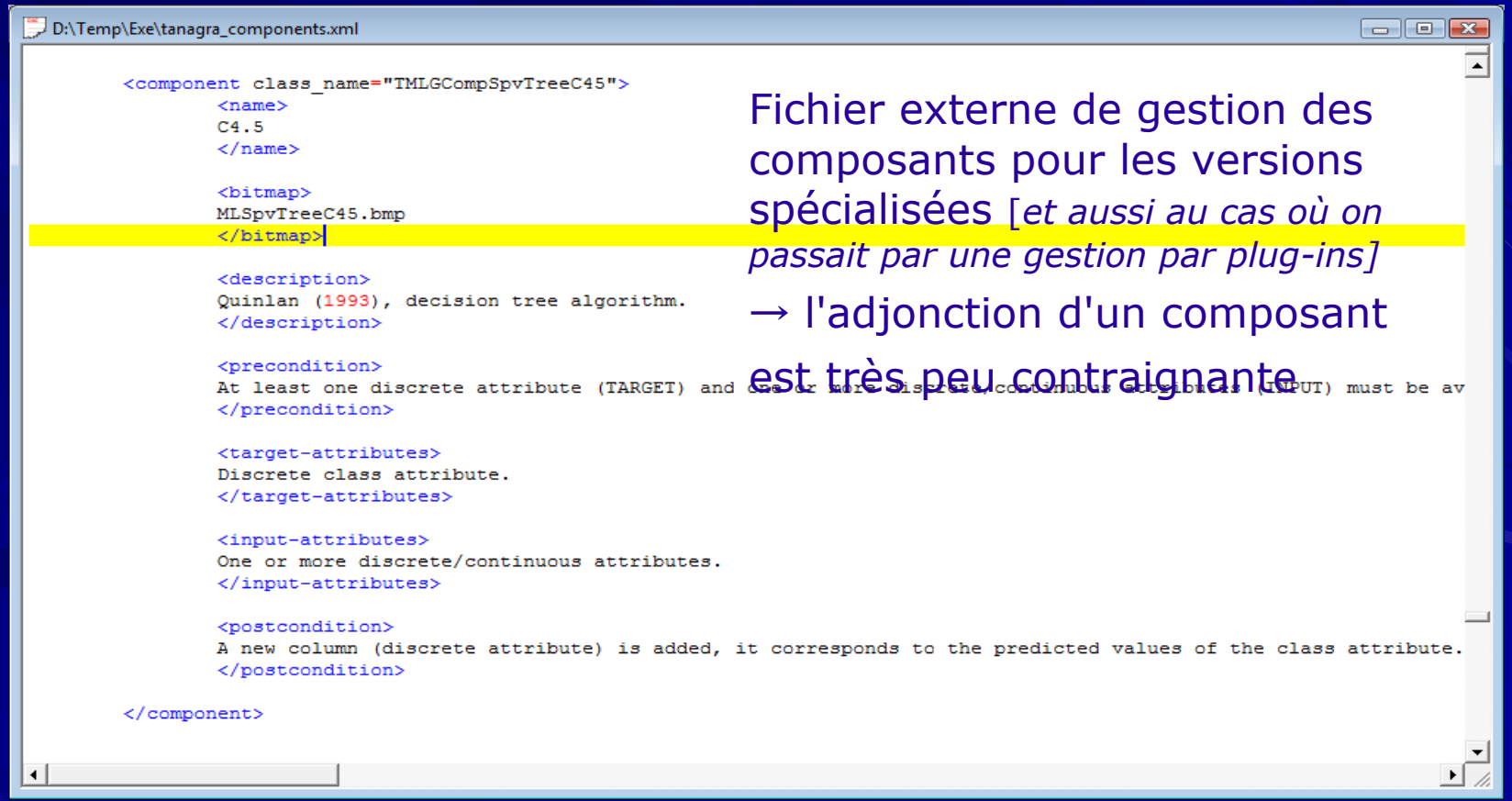
UFrmMainForm | UCalcSpvTreeCART | UCalcSpvTreeC45

```
    UCalcTreeStructureDefinition,
    UCalcSpvTreeDefinition;

TYPE
    //feuille
    TSplitLeafSpvC45 = class(TSplitLeafSpv)
                       end;

    //split
    TSplitAttributSpvC45 = class(TSplitAttributSpv)
                           protected
                           function    getClassSplitLeaf(): TClassSplitLeaf; override;
                           function    ComputeGoodness(): double; override;
                           function    ComputeAcceptSplit(): boolean; override;
                           end;

    //liste de splits
    TLstSplitAttSpvC45 = class(TLstSplitAttSpv)
                         protected
                         function  getClassSplitAttribut(): TClassSplitAttribut; override;
                         end;

    //noeud de l'arbre
    TMLTreeNodeSpvC45 = class(TMLTreeNodeSpv)
                        private
                        //calcul de l'écart à l'erreur pour avoir la borne -- taille sommet, contre-e
                        //extrait du livre de Quinlan, "Programs for Machine Learning..."
                        function  addErrs(N,CE,CF: double): double;
                        protected
                        procedure AssignConclusion(); override;
                        function  isNoSplitNeeded(): boolean; override;
                        function  getClassLstSplitAttributes(): TClassLstSplitAttributes; override;
                        public
                        //erreur pessimiste (c'est le nombre d'erreur ici !!!) -- calculée lors de l'
                        FPessimisticErr: double;
                        //savoir si un noeud est prunable, i.e. a des enfants, et ce sont tous des fe
                        function  isPrunable(): boolean;
                        //erreur pessimiste
                        property  pessimistic: double read FPessimisticErr;
                        end;

    //structure d'arbre
    TMLTreeStructureSpvC45 = class(TMLTreeStructureSpv)
                             protected
                             function   getClassMLTreeNode(): TClassMLTreeNode; override;
                             public
                             //ce qui est spécifique à C4.5
                             procedure   PostPruning(); override;
                             end;
```

Classes de calcul

1: 1 | Insertion | Code


D:_Travaux\personal\Programmation\Tanagra\Source\Diagram\MLComponents\SpvLearning\SpvAlgorithm\DecisionT...
UFrmMainForm
UCalcSpvTreeCART
UCalcSpvTreeC45
UCompSpvCSTreeC45
TYPE
{générateur de composant C4.5}
TMLGCompSpvCSTreeC45 = class(TMLGCompSpvTree)
public
function GetClassMLComponent: TClassMLComponent; override;
end;
{le composant SpvTree C4.5}
TMLCompSpvCSTreeC45 = class(TMLCompSpvTree)
protected
function getClassOperator: TClassOperator; override;
end;
{l'opérateur C4.5}
TOpSpvCSTreeC45 = class(TOpSpvTree)
protected
function getClassParameter: TClassOperatorParameter; override;
function getClassSpvLearning(): TClassCalcSpvLearning; override;
end;
{paramètre de l'algo CART}
TOpPrmSpvCSTreeC45 = class(TOpPrmSpvTree)
private
//taille minimale pour les feuilles
FMinSizeLeaf: integer;
//lambda pour l'estimation laplacienne des distributions
FLambda: double;
//matrice de coût
FCostMatrix: TCostMatrix;
protected
procedure SetDefaultParameters(); override;
function CreateDlgParameters(): TForm; override;
public
{son propriétaire est passé en paramètre : l'opérateur}
constructor Create(prmOp: TOperator); override;
destructor Destroy(); override;
function getHTMLParameters(): string; override;
procedure LoadFromStream(prmStream: TStream); override;
procedure SaveToStream(prmStream: TStream); override;
procedure LoadFromINI(prmSection: string; prmINI: TMemIniFile); override;
procedure SaveToINI(prmSection: string; prmINI: TMemIniFile); override;
property CostMatrix: TCostMatrix read FCostMatrix;
property Lambda: double read FLambda write FLambda;
property MinSizeLeaf: integer read FMinSizeLeaf write FMinSizeLeaf;
end;
Classes de gestion des composants
1: 1
Insertion
Code



Fichier externe de gestion des composants pour les versions spécialisées [*et aussi au cas où on passait par une gestion par plug-ins*]

→ l'adjonction d'un composant est très peu contraignante

Simplicité pour les programmeurs

Encore plus loin dans la modularité : les plugins

La solution idéale ?

L'application mère est une matrice qui gère et transmet les données

Les techniques sont des procédures programmées sous forme de bibliothèques externes

Mais des contraintes fortes

Organisation ultra rigoureuse des protocoles

→ Passage des informations et des données

→ Affichage des résultats

→ Documentation (fichier d'aide)

Bref…

Souvent rédhibitoire, alors que l'objectif était d'offrir un outil modulaire

Intéressant si plugins = procédures de calculs qui renvoient des objets standardisés

Et qu'une vraie équipe organise la vie autour du logiciel

→ Le logiciel R est le seul à avoir su le faire convenablement

Implémentation

Quels outils pour la programmation ?

Spécifications

Outil libre (*ça coûte moins cher*)

Largement diffusé (*pour avoir des programmeurs*)

Avec une large bibliothèque de classes (*calculs, conteneurs, etc.*)

Qui permet de faire des interfaces agréables, simplement, rapidement

Pourquoi DELPHI pour Tanagra ?

DELPHI 6.0 PERSO est gratuite

Cours de DELPHI en L3 et M1 dans le département « Informatique – Statistique »

Accès aux anciennes bibliothèques de calculs, validées depuis longtemps déjà

Connaissance étendue des bibliothèques libres (Turbo Power, etc.)

Permet de faire des interfaces agréables, simplement, rapidement

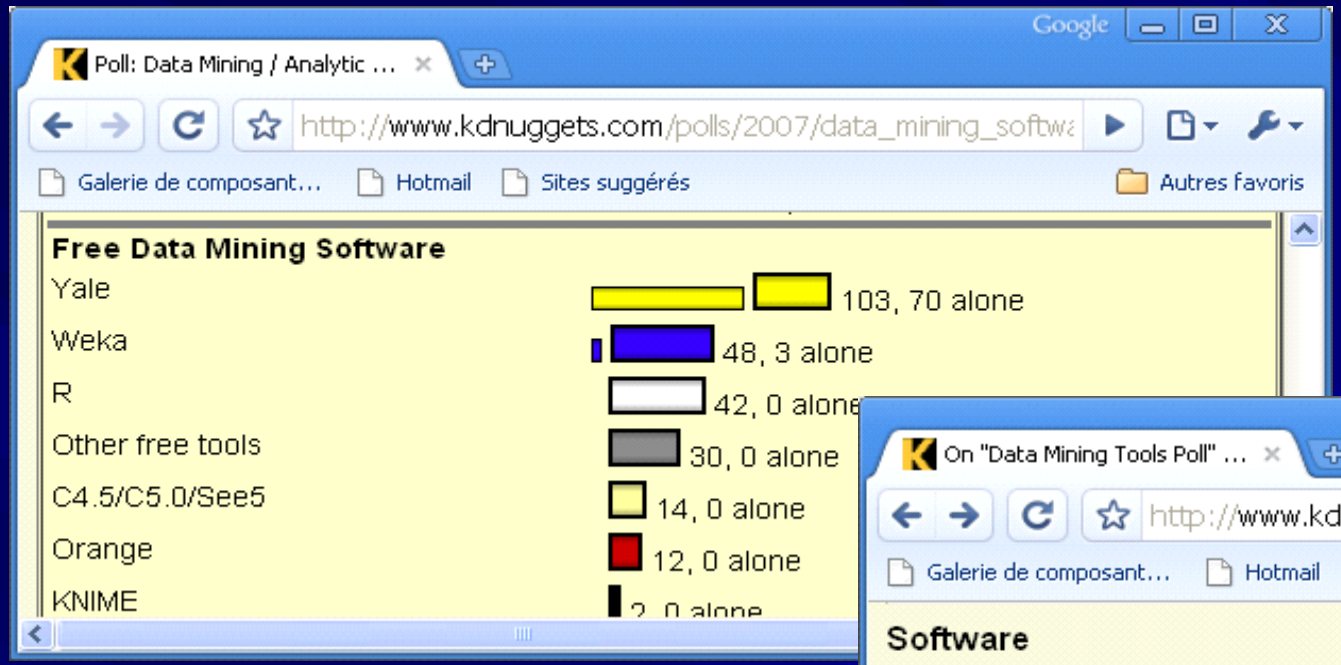
Affinités personnelles…

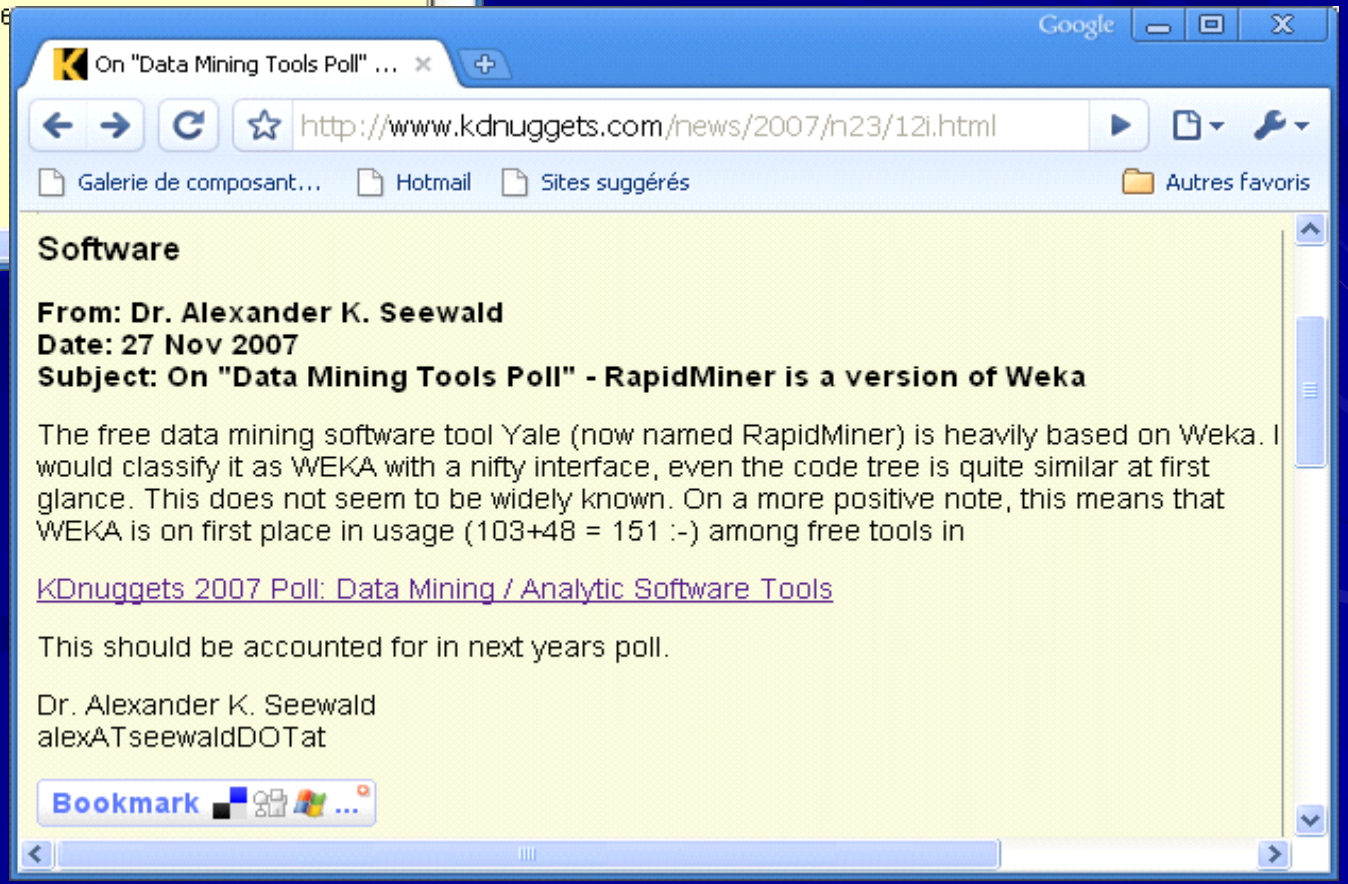
J'aurais du le faire en JAVA ? L'écueil WEKA →

Implémentation

Pourquoi ne pas avoir intégré des bibliothèques de calcul existantes ?

Sondage : quel logiciel utilisez vous en 2007 ?





Promotion

Comment faire connaître le logiciel… sachant qu'on n'a rien à vendre ?

La promotion dans les conférences

Ateliers, démonstrations, contacts chercheurs, mailing- list, etc.

Écrire un article de référence

Voilà toujours une publication de plus

Marquer le coup en annonçant le logiciel

C'est la référence que citeront les utilisateurs

Monter un site web attrayant (attractif)

La visibilité internet est primordiale

Le téléchargement du logiciel n'est pas le seul enjeu

Système de « news - actualités » pour soutenir l'attention des utilisateurs

Documenter le logiciel

Documenter les méthodes : description théorique

Documenter leur mise en oeuvre : les tutoriels

Facilitée par le découpage en « composants » du logiciel

Mon principal cheval de bataille aujourd'hui

→ Travail de fond. A son rythme. Sans contraintes de temps. Sans pression.

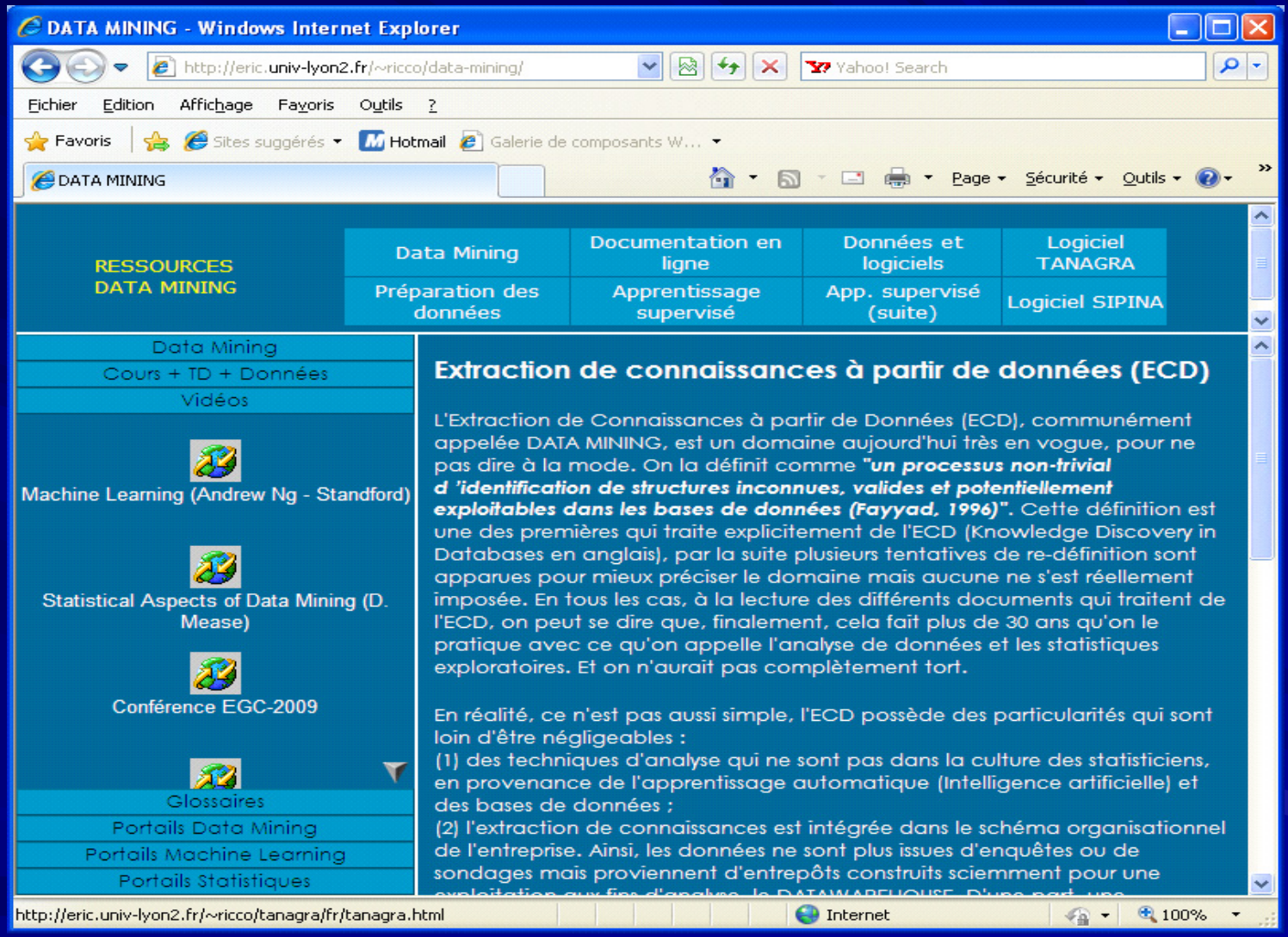
→ Étendu aux autres logiciels. Je documente les autres logiciels libres !!!

Promotion
Le site web Tanagra


TANAGRA - A free DATA ...
http://eric.univ-lyon2.fr/~ricco/tanagra/en/tanagra.html
e-mail
Prévisions météo de...
YouTube - Page Yo...
Bmw Story
Autres favoris
TANAGRA
Presentation
Screenshots
Functionalities
Tutorials
Download
Sipina
Presentation
TANAGRA project
Releases NEW!
History
References
Related softwares
Contact
TANAGRA project
TANAGRA is a free DATA MINING software for academic and research purposes. It proposes several data mining methods from exploratory data analysis, statistical learning, machine learning and databases area.
This project is the successor of SIPINA which implements various supervised learning algorithms, especially an interactive and visual construction of decision trees. TANAGRA is more powerful, it contains some supervised learning but also other paradigms such as clustering, factorial analysis, parametric and nonparametric statistics, association rule, feature selection and construction algorithms...
TANAGRA is an "open source project" as every researcher can access to the source code, and add his own algorithms, as far as he agrees and conforms to the software distribution license.
The main purpose of Tanagra project is to give researchers and students an easy-to-use data mining software, conforming to the present norms of the software development in this domain (especially in the design of its GUI and the way to use it), and allowing to analyse either real or synthetic data.
The second purpose of TANAGRA is to propose to researchers an architecture allowing them to easily

Promotion

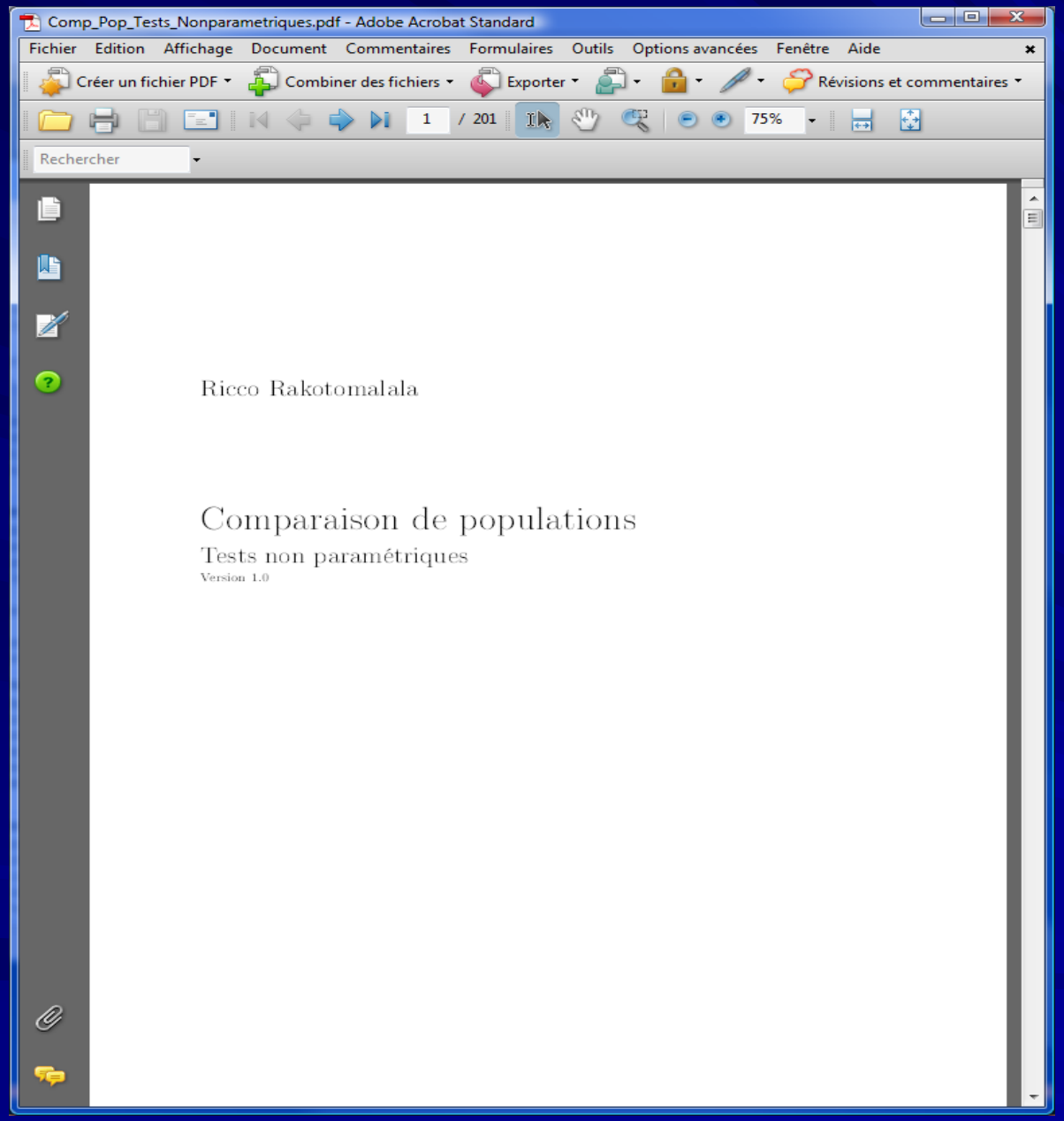
Documentation des méthodes – Pointeurs vers les ressources



Promotion

Documentation des méthodes – Écrire et diffuser des supports libres

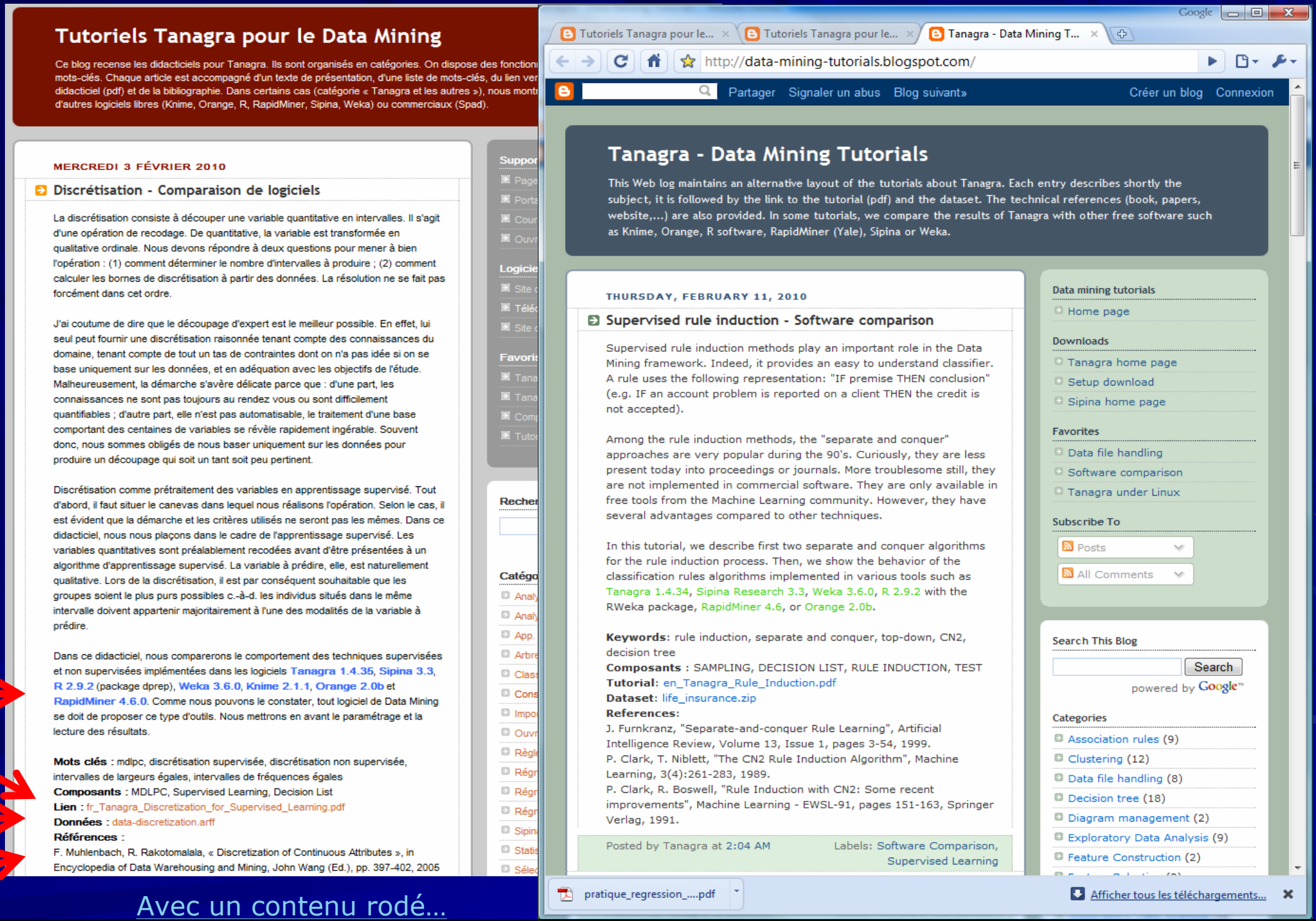
Présenter à la fois les aspects pratiques et théoriques


Comp_Pop_Tests_Nonparametriques.pdf - Adobe Acrobat Standard
Fichier Edition Affichage Document Commentaires Formulaires Outils Options avancées Fenêtre Aide
Créer un fichier PDF
Combiner des fichiers
Exporter
Révisions et commentaires
1 / 201
75%
Rechercher
Ricco Rakotomalala
Comparaison de populations
Tests non paramétriques
Version 1.0


http://eric.univ-lyon2.fr/...
http://eric.univ-lyon2.fr/~ricco/cours/cours/pratique_regression_logistique.pdf
e-mail
Prévisions météo de...
YouTube - Page Yo...
Bmw Story
Autres favoris
100%
Rechercher
1 / 272
Présenter à la fois les aspects pratiques et théoriques
Ricco Rakotomalala
Pratique de la Régression Logistique
Régression Logistique Binaire et Polytomique
Version 2.0

Promotion

Documenter la mise en œuvre des méthodes – Les tutoriels



Tutoriels Tanagra pour le Data Mining

Ce blog recense les didacticiels pour Tanagra. Ils sont organisés en catégories. On dispose des fonction
mots-clés. Chaque article est accompagné d'un texte de présentation, d'une liste de mots-clés, du lien ver
didacticiel (pdf) et de la bibliographie. Dans certains cas (catégorie « Tanagra et les autres »), nous montr
d'autres logiciels libres (Knime, Orange, R, RapidMiner, Sipina, Weka) ou commerciaux (Spad).

MERCREDI 3 FÉVRIER 2010

Discrétisation - Comparaison de logiciels

La discrétisation consiste à découper une variable quantitative en intervalles. Il s'agit d'une opération de recodage. De quantitative, la variable est transformée en qualitative ordinale. Nous devons répondre à deux questions pour mener à bien l'opération : (1) comment déterminer le nombre d'intervalles à produire ; (2) comment calculer les bornes de discrétisation à partir des données. La résolution ne se fait pas forcément dans cet ordre.

J'ai coutume de dire que le découpage d'expert est le meilleur possible. En effet, lui seul peut fournir une discrétisation raisonnée tenant compte des connaissances du domaine, tenant compte de tout un tas de contraintes dont on n'a pas idée si on se base uniquement sur les données, et en adéquation avec les objectifs de l'étude. Malheureusement, la démarche s'avère délicate parce que : d'une part, les connaissances ne sont pas toujours au rendez vous ou sont difficilement quantifiables ; d'autre part, elle n'est pas automatisable, le traitement d'une base comportant des centaines de variables se révèle rapidement ingérable. Souvent donc, nous sommes obligés de nous baser uniquement sur les données pour produire un découpage qui soit un tant soit peu pertinent.

Discrétisation comme prétraitement des variables en apprentissage supervisé. Tout d'abord, il faut situer le canevas dans lequel nous réalisons l'opération. Selon le cas, il est évident que la démarche et les critères utilisés ne seront pas les mêmes. Dans ce didacticiel, nous nous plaçons dans le cadre de l'apprentissage supervisé. Les variables quantitatives sont préalablement recodées avant d'être présentées à un algorithme d'apprentissage supervisé. La variable à prédire, elle, est naturellement qualitative. Lors de la discrétisation, il est par conséquent souhaitable que les groupes soient le plus purs possibles c.-à-d. les individus situés dans le même intervalle doivent appartenir majoritairement à l'une des modalités de la variable à prédire.

Dans ce didacticiel, nous comparerons le comportement des techniques supervisées et non supervisées implémentées dans les logiciels **Tanagra 1.4.35**, **Sipina 3.3**, **R 2.9.2** (package dprep), **Weka 3.6.0**, **Knime 2.1.1**, **Orange 2.0b** et **RapidMiner 4.6.0**. Comme nous pouvons le constater, tout logiciel de Data Mining se doit de proposer ce type d'outils. Nous mettrons en avant le paramétrage et la lecture des résultats.

Mots clés : mdlpc, discrétisation supervisée, discrétisation non supervisée, intervalles de largeurs égales, intervalles de fréquences égales
Composants : MDLPC, Supervised Learning, Decision List
Lien : fr_Tanagra_Discretization_for_Supervised_Learning.pdf
Données : data-discretization.arff
Références :
F. Muhlenbach, R. Rakotomalala, « Discretization of Continuous Attributes », in Encyclopedia of Data Warehousing and Mining, John Wang (Ed.), pp. 397-402, 2005

Tanagra - Data Mining Tutorials

This Web log maintains an alternative layout of the tutorials about Tanagra. Each entry describes shortly the subject, it is followed by the link to the tutorial (pdf) and the dataset. The technical references (book, papers, website,...) are also provided. In some tutorials, we compare the results of Tanagra with other free software such as Knime, Orange, R software, RapidMiner (Yale), Sipina or Weka.

THURSDAY, FEBRUARY 11, 2010

Supervised rule induction - Software comparison

Supervised rule induction methods play an important role in the Data Mining framework. Indeed, it provides an easy to understand classifier. A rule uses the following representation: "IF premise THEN conclusion" (e.g. IF an account problem is reported on a client THEN the credit is not accepted).

Among the rule induction methods, the "separate and conquer" approaches are very popular during the 90's. Curiously, they are less present today into proceedings or journals. More troublesome still, they are not implemented in commercial software. They are only available in free tools from the Machine Learning community. However, they have several advantages compared to other techniques.

In this tutorial, we describe first two separate and conquer algorithms for the rule induction process. Then, we show the behavior of the classification rules algorithms implemented in various tools such as Tanagra 1.4.34, Sipina Research 3.3, Weka 3.6.0, R 2.9.2 with the RWeka package, RapidMiner 4.6, or Orange 2.0b.

Keywords: rule induction, separate and conquer, top-down, CN2, decision tree
Composants : SAMPLING, DECISION LIST, RULE INDUCTION, TEST
Tutorial: en_Tanagra_Rule_Induction.pdf
Dataset: life_insurance.zip
References:
J. Furnkranz, "Separate-and-conquer Rule Learning", Artificial Intelligence Review, Volume 13, Issue 1, pages 3-54, 1999.
P. Clark, T. Niblett, "The CN2 Rule Induction Algorithm", Machine Learning, 3(4):261-283, 1989.
P. Clark, R. Boswell, "Rule Induction with CN2: Some recent improvements", Machine Learning - EWSL-91, pages 151-163, Springer Verlag, 1991.

Posted by Tanagra at 2:04 AM Labels: Software Comparison, Supervised Learning

Data mining tutorials
Home page
Downloads
Tanagra home page
Setup download
Sipina home page
Favorites
Data file handling
Software comparison
Tanagra under Linux
Subscribe To
Posts
All Comments
Search This Blog
Search
powered by Google
Categories
Association rules (9)
Clustering (12)
Data file handling (8)
Decision tree (18)
Diagram management (2)
Exploratory Data Analysis (9)
Feature Construction (2)

pratique_regression_....pdf
Afficher tous les téléchargements...

Avec un contenu rodé...

Tanagra
Bilan (1)

Écriture du cahier des charges

Janvier 2003, plusieurs prototypes de janvier à juin 2003

Début du développement

Juillet 2003

Création du site web et mise en ligne

Janvier 2004 (~25 visiteurs par jour sur 2004)

Techniques implémentées (version 1.4.36 – Mars 2010)

170 méthodes stat., analyse de données, data mining

Documentation libre en ligne (Mars 2010)

8 ouvrages libres en PDF

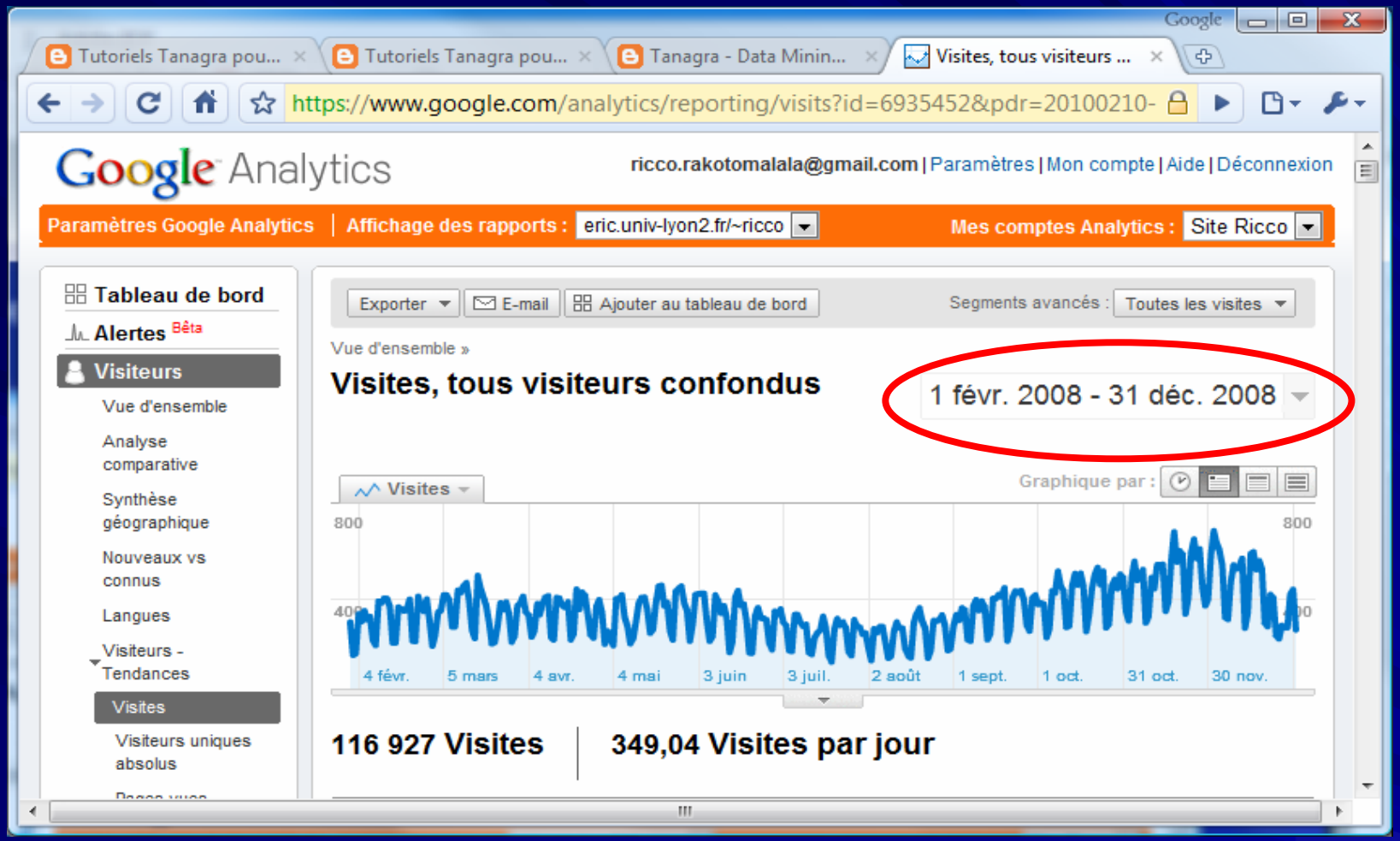
30 « slides » en PDF

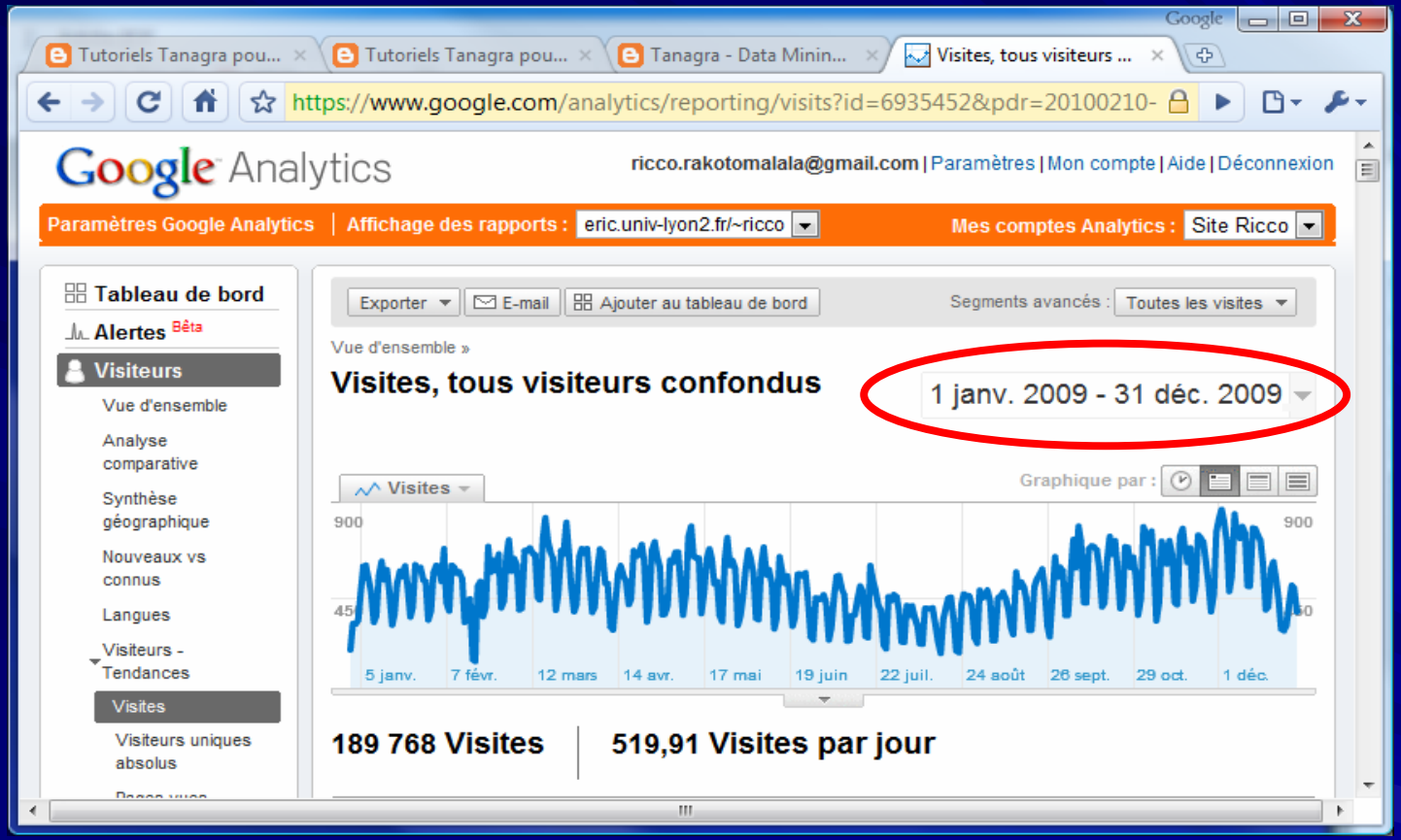
140 didacticiels en français

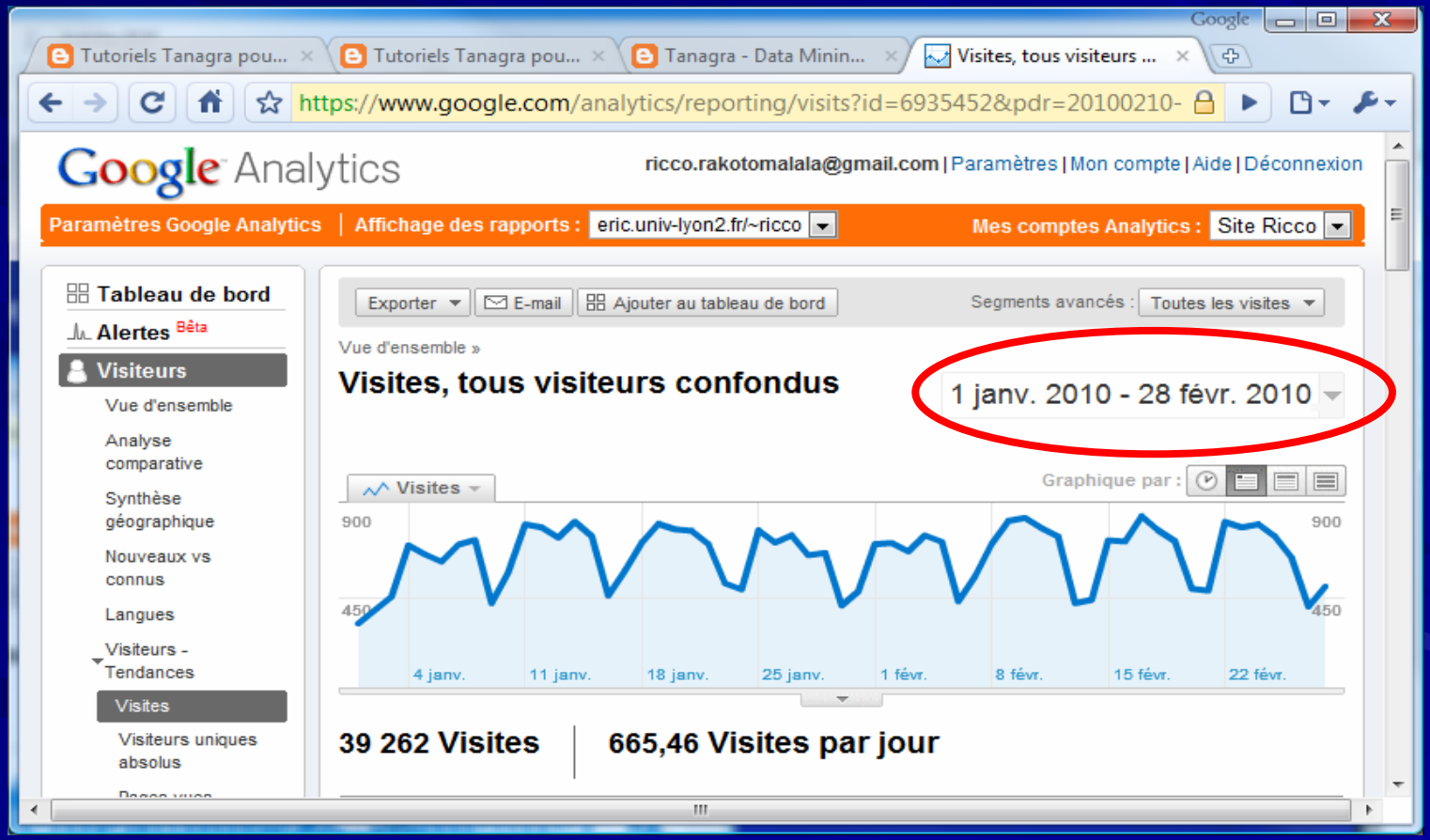
100 didacticiels en anglais

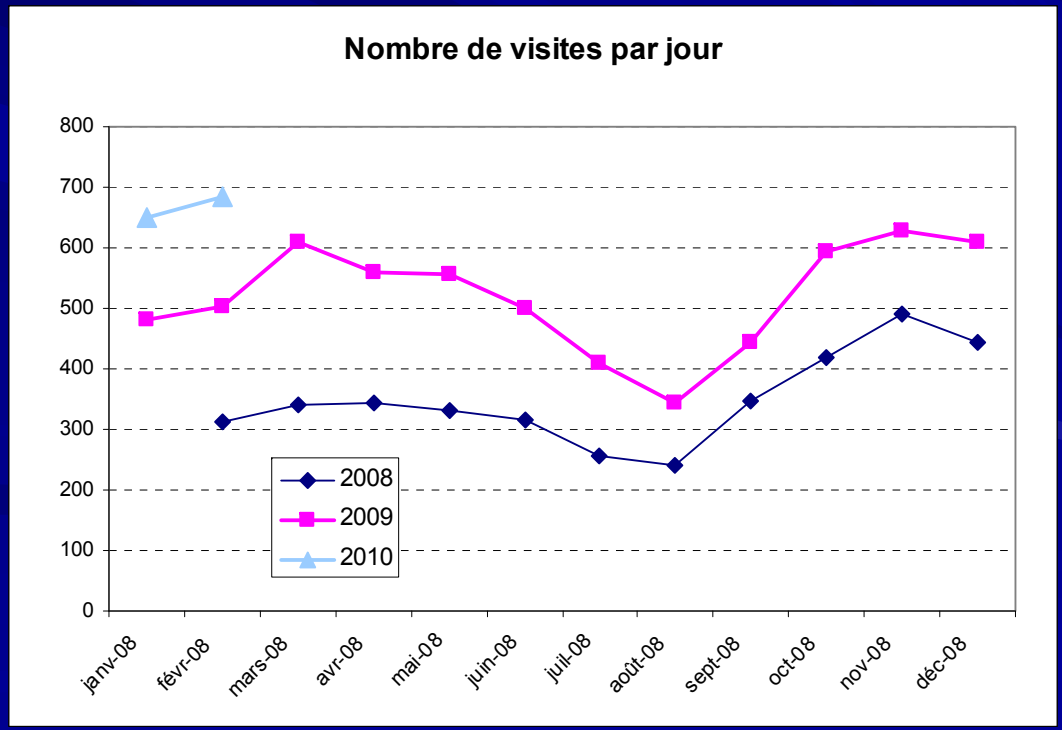
Tanagra

Bilan (2) – Quelle fréquentation du site ?



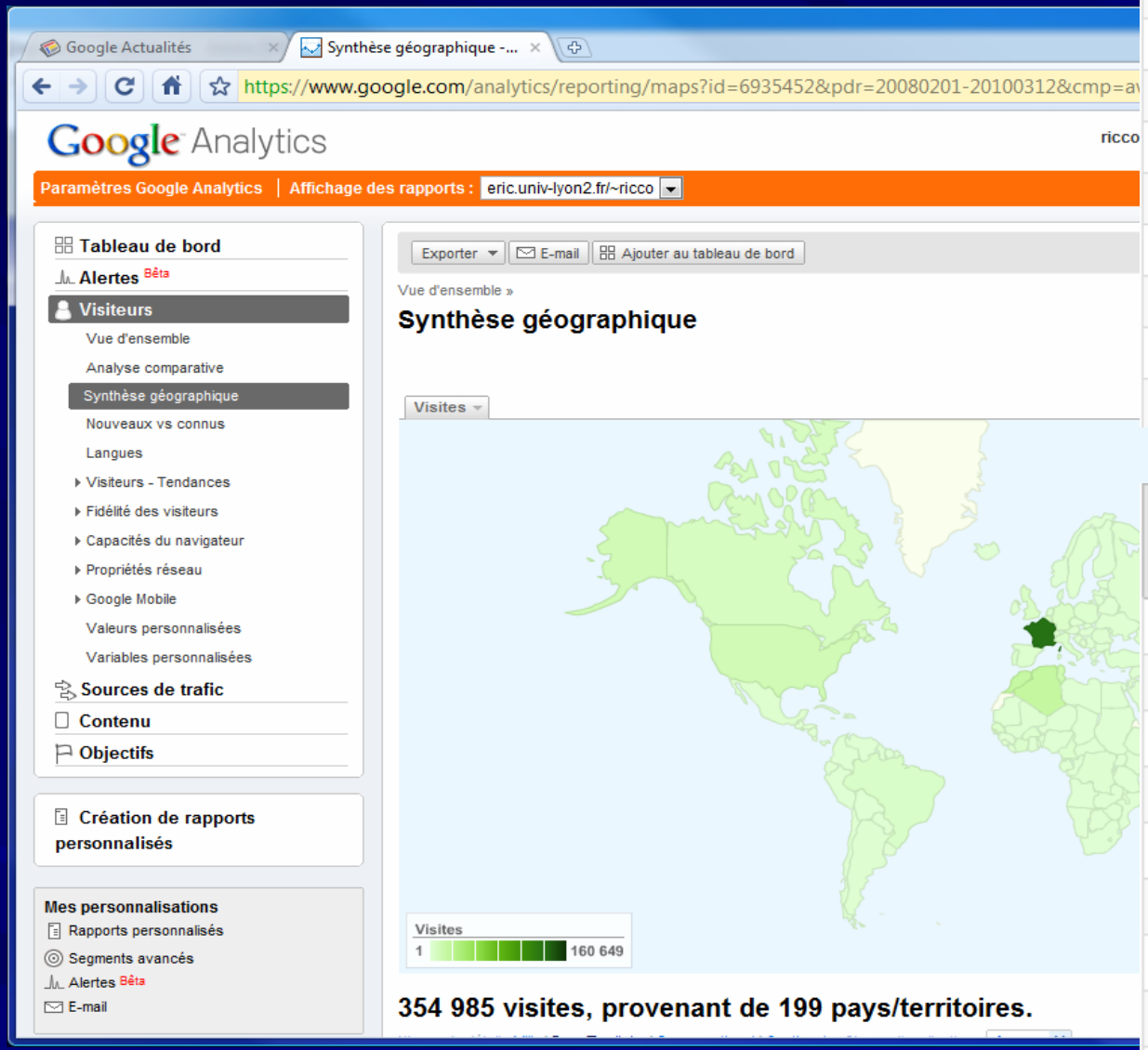

Tutoriels Tanagra pou...
Tutoriels Tanagra pou...
Tanagra - Data Minin...
Visites, tous visiteurs ...
https://www.google.com/analytics/reporting/visits?id=6935452&pdr=20100210-
Google Analytics
ricco.rakotomalala@gmail.com | Paramètres | Mon compte | Aide | Déconnexion
Paramètres Google Analytics | Affichage des rapports : eric.univ-lyon2.fr/~ricco | Mes comptes Analytics : Site Ricco
Tableau de bord
Alertes Bêta
Visiteurs
Vue d'ensemble
Analyse comparative
Synthèse géographique
Nouveaux vs connus
Langues
Visiteurs - Tendances
Visites
Visiteurs uniques absolus
Exporter
E-mail
Ajouter au tableau de bord
Segments avancés : Toutes les visites
Vue d'ensemble »
Visites, tous visiteurs confondus
1 janv. 2009 - 31 déc. 2009
Visites
Graphique par :
900
900
5 janv.
7 févr.
12 mars
14 avr.
17 mai
19 juin
22 juil.
24 août
26 sept.
29 oct.
1 déc.
189 768 Visites
519,91 Visites par jour


Tutoriels Tanagra pou...
Tutoriels Tanagra pou...
Tanagra - Data Minin...
Visites, tous visiteurs ...
https://www.google.com/analytics/reporting/visits?id=6935452&pdr=20100210-
Google Analytics
ricco.rakotomalala@gmail.com | Paramètres | Mon compte | Aide | Déconnexion
Paramètres Google Analytics | Affichage des rapports : eric.univ-lyon2.fr/~ricco
Mes comptes Analytics : Site Ricco
Tableau de bord
Alertes Bêta
Visiteurs
Vue d'ensemble
Analyse comparative
Synthèse géographique
Nouveaux vs connus
Langues
Visiteurs - Tendances
Visites
Visiteurs uniques absolus
Exporter
E-mail
Ajouter au tableau de bord
Segments avancés : Toutes les visites
Vue d'ensemble »
Visites, tous visiteurs confondus
1 janv. 2010 - 28 févr. 2010
Visites
Graphique par :
900
450
4 janv.
11 janv.
18 janv.
25 janv.
1 févr.
8 févr.
15 févr.
22 févr.
39 262 Visites
665,46 Visites par jour


Nombre de visites par jour
800
700
600
500
400
300
200
100
0
2008
2009
2010
janv-08
févr-08
mars-08
avr-08
mai-08
juin-08
juil-08
août-08
sept-08
oct-08
nov-08
déc-08

Tanagra

Bilan (3) – Qui fréquente le site ?



	Niveau de détail : Pays/Territoire	Visites ↓
1.	France	160 649
2.	Algeria	23 853
3.	Morocco	22 943
4.	Tunisia	15 935
5.	United States	14 148
6.	Canada	8 585
7.	India	7 960
8.	Belgium	7 640
9.	Brazil	5 513
10.	United Kingdom	5 351

Quels pays ?

	Titre de la page Aucun	Pages vues ↓
1.	Portail DATA MINING	74 984
2.	Supports de cours -- Data Mining	62 583
3.	Tanagra EN	56 990
4.	Tanagra FR	56 129
5.	Cours econometrie	55 190
6.	Download Tanagra EN	37 456
7.	Download Tanagra FR	34 337
8.	Cours Excel	34 312
9.	Tutoriels Tanagra pour le Data Mining	31 985
10.	Sipina Main Menu	25 531

Quelles pages ?

Les sites des tutoriels sont récents (FR et EN)

4. Et les autres outils libres ?

Knime

Estampillé « Intelligent Data Analysis »

Université de Konstanz - Allemagne

Culture I.D.A
Code source libre C++ (méthodes)

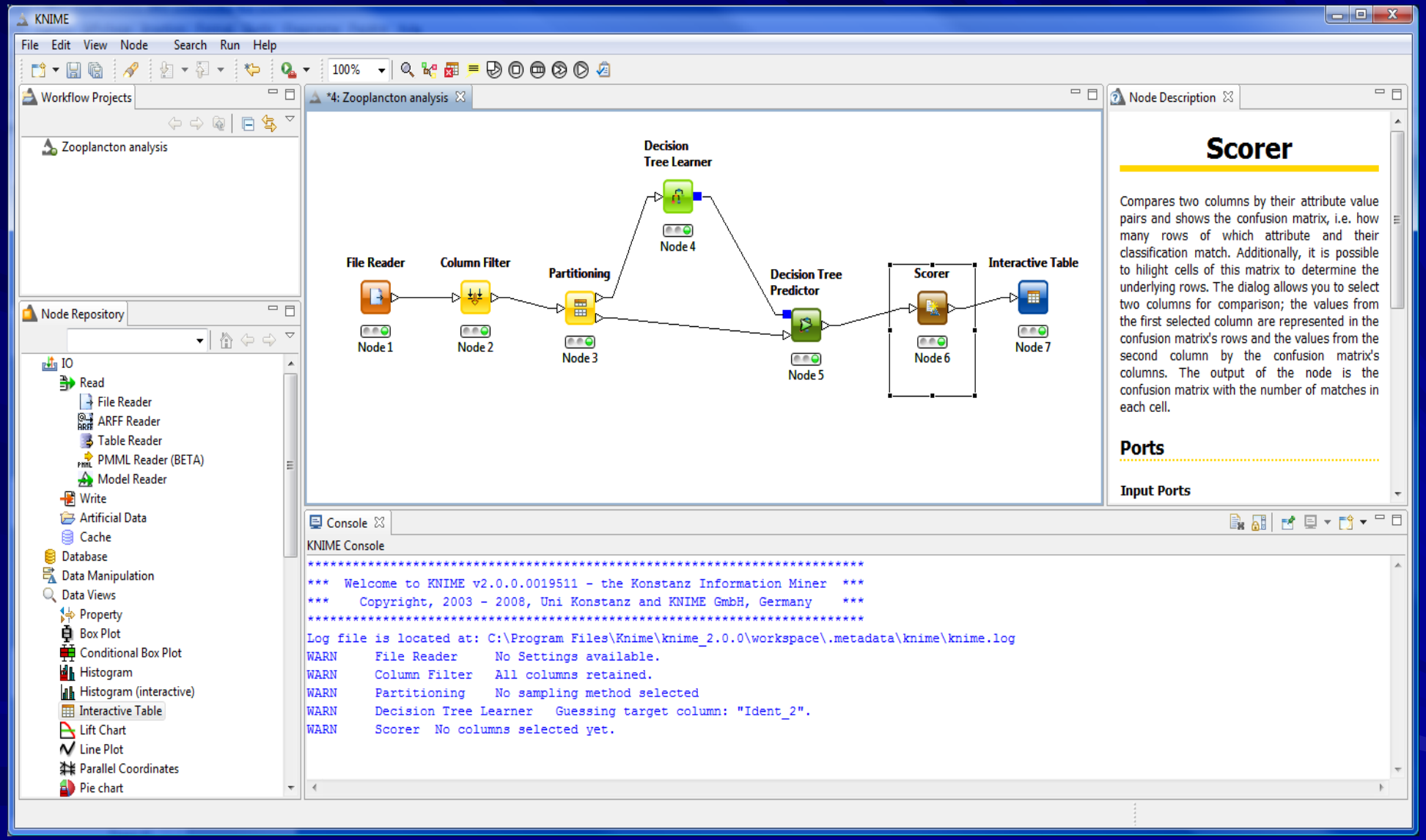
Doc sous forme de fichier d'aide intégré

Mode diagramme
Avec des fonctionnalités avancées (boucles,...)

Les méthodes sont des plugins
Possibilité d'importer des classes Weka
Possibilité d'intégrer des packages R

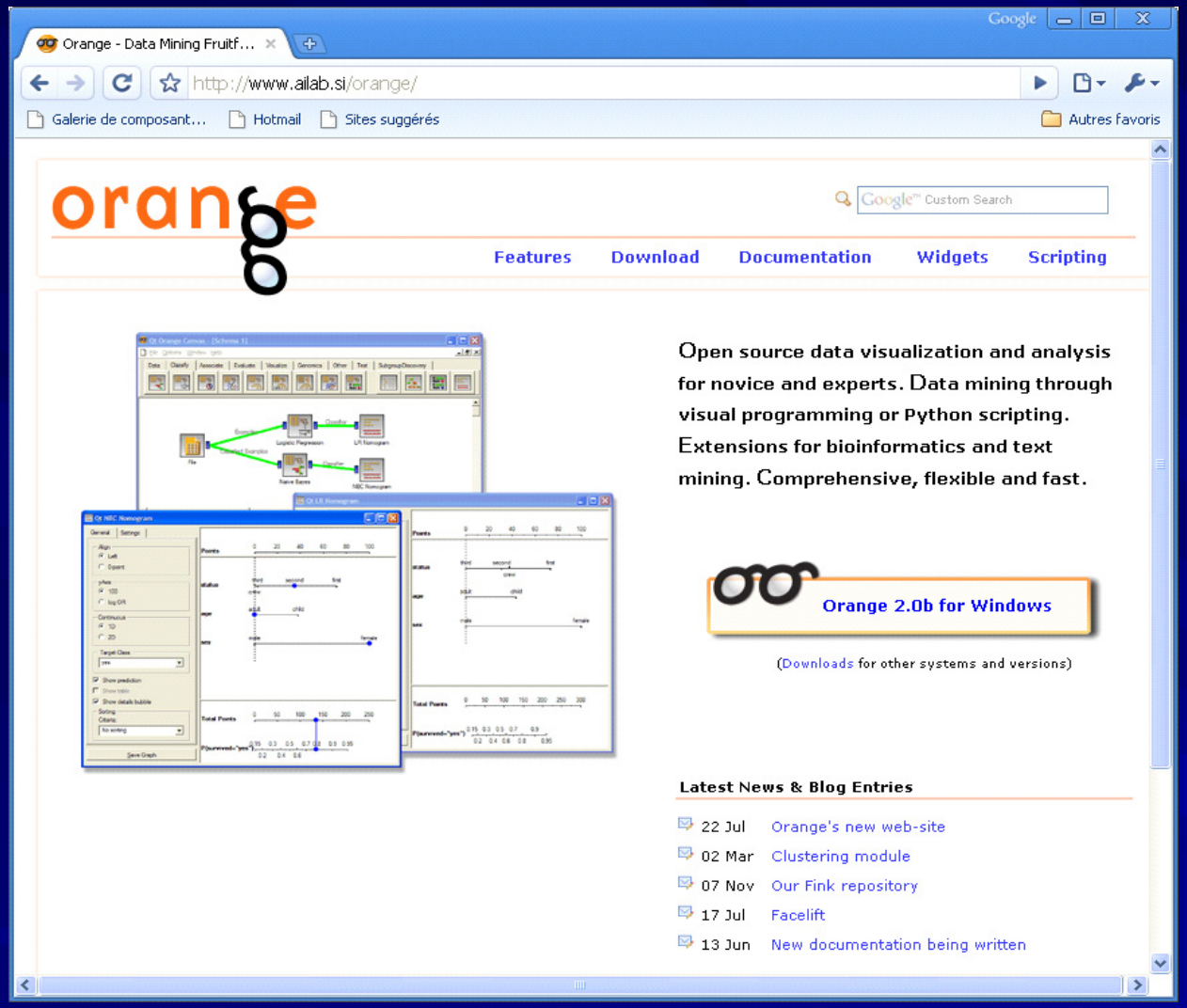
Multi-thread et possibilité de swap pour certaines méthodes, le mieux armé pour les gros volumes

Knime
Interface



Orange

Estampillé « Machine Learning »



A.I. Lab – Université de Lubiana – Slovénie

Culture I.A. - Machine Learning (ICML, …)
Code source libre C++

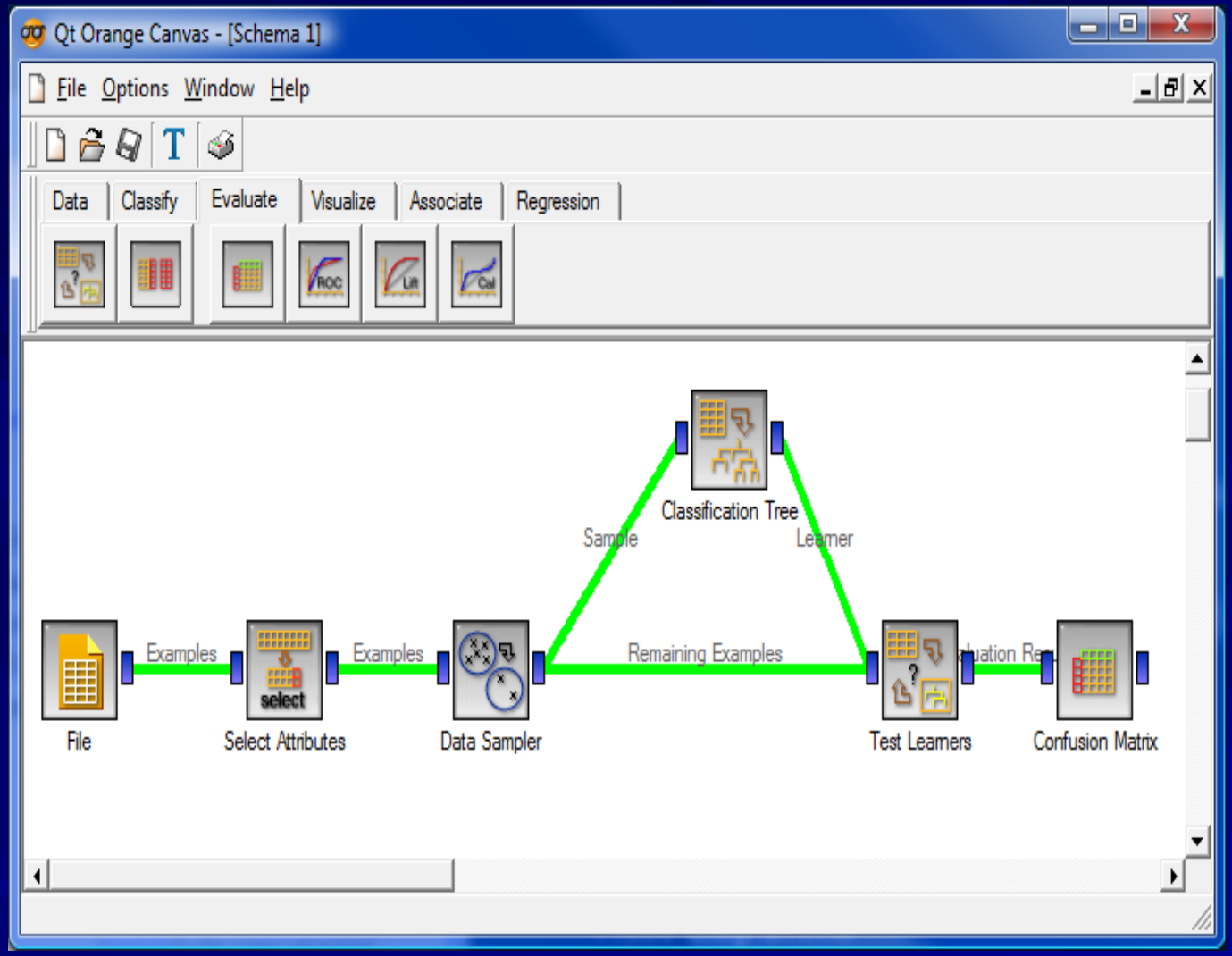
Site Web avec doc en ligne et guide

Mode diagramme
Programmation en Python

Les méthodes sont des plugins (DLL)

Très user-friendly

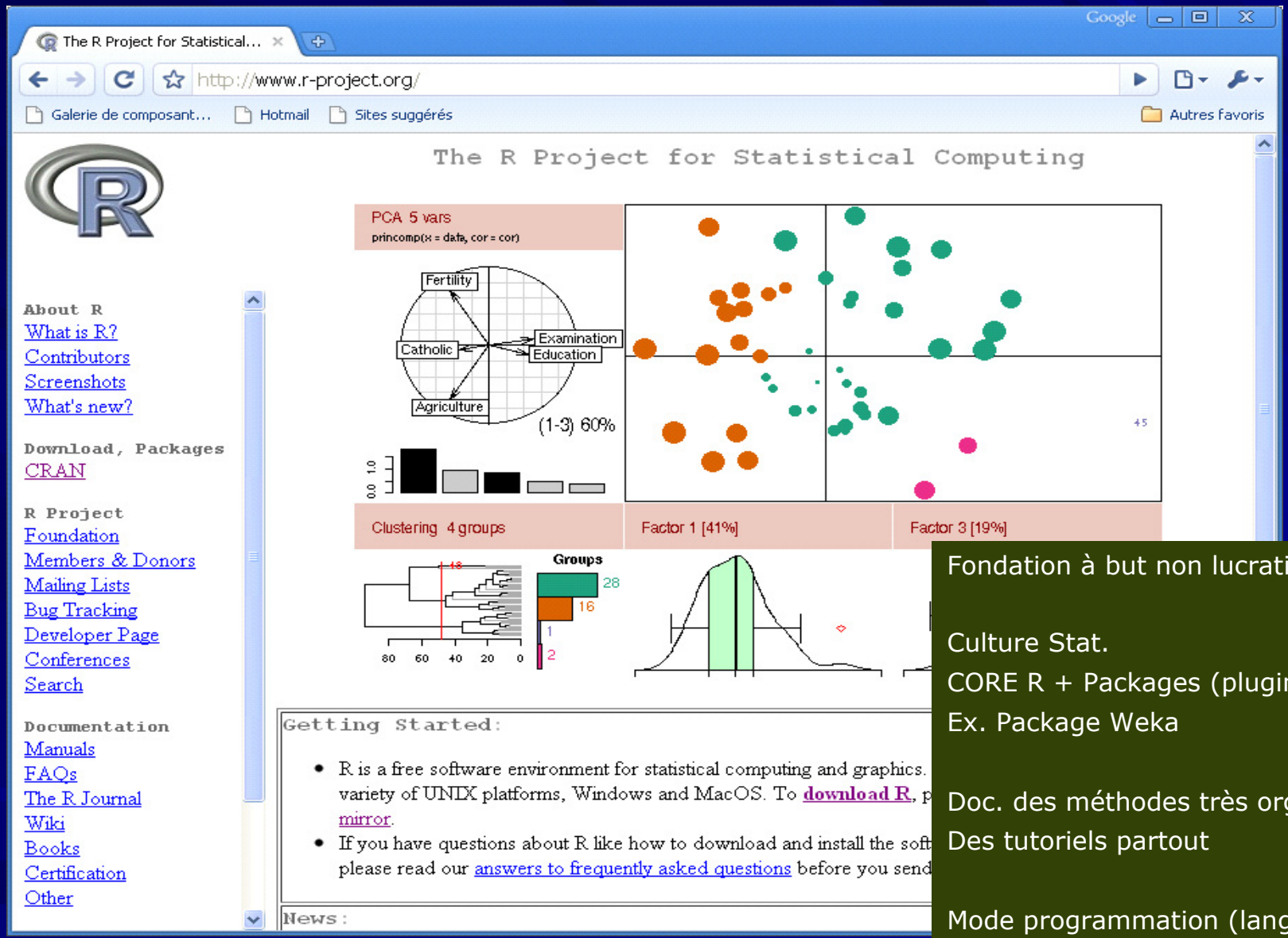
Orange
Utilisation



```
# Description: Shows how to construct an orange.ClassifierFromExampleTable
# Category:    classification, lookup classifiers, constructive induction, feature construction
# Classes:     ClassifierByExampleTable, LookupLearner
# Uses:        monk1
# Referenced:  lookup.htm

import orange

data = orange.ExampleTable("monk1")
a, b, e = data.domain["a"], data.domain["b"], data.domain["e"]

data_s = data.select([a, b, e, data.domain.classVar])
abe = orange.LookupLearner(data_s)

print len(data_s)
print len(abe.sortedExamples)

for i in abe.sortedExamples[:10]:
    print i
print

for i in abe.sortedExamples[:10]:
    print i, i.getclass().svalue
print

y2 = orange.EnumVariable("y2", values = ["0", "1"])
abe2 = orange.LookupLearner(y2, [a, b, e], data)
for i in abe2.sortedExamples[:10]:
    print i, i.getclass().svalue
print

y2 = orange.EnumVariable("y2", values = ["0", "1"])
abe2 = orange.LookupLearner(y2, [a, b], data)
for i in abe2.sortedExamples:
    print i, i.getclass().svalue
```

R

Estampillé « Statistique »



Fondation à but non lucratif

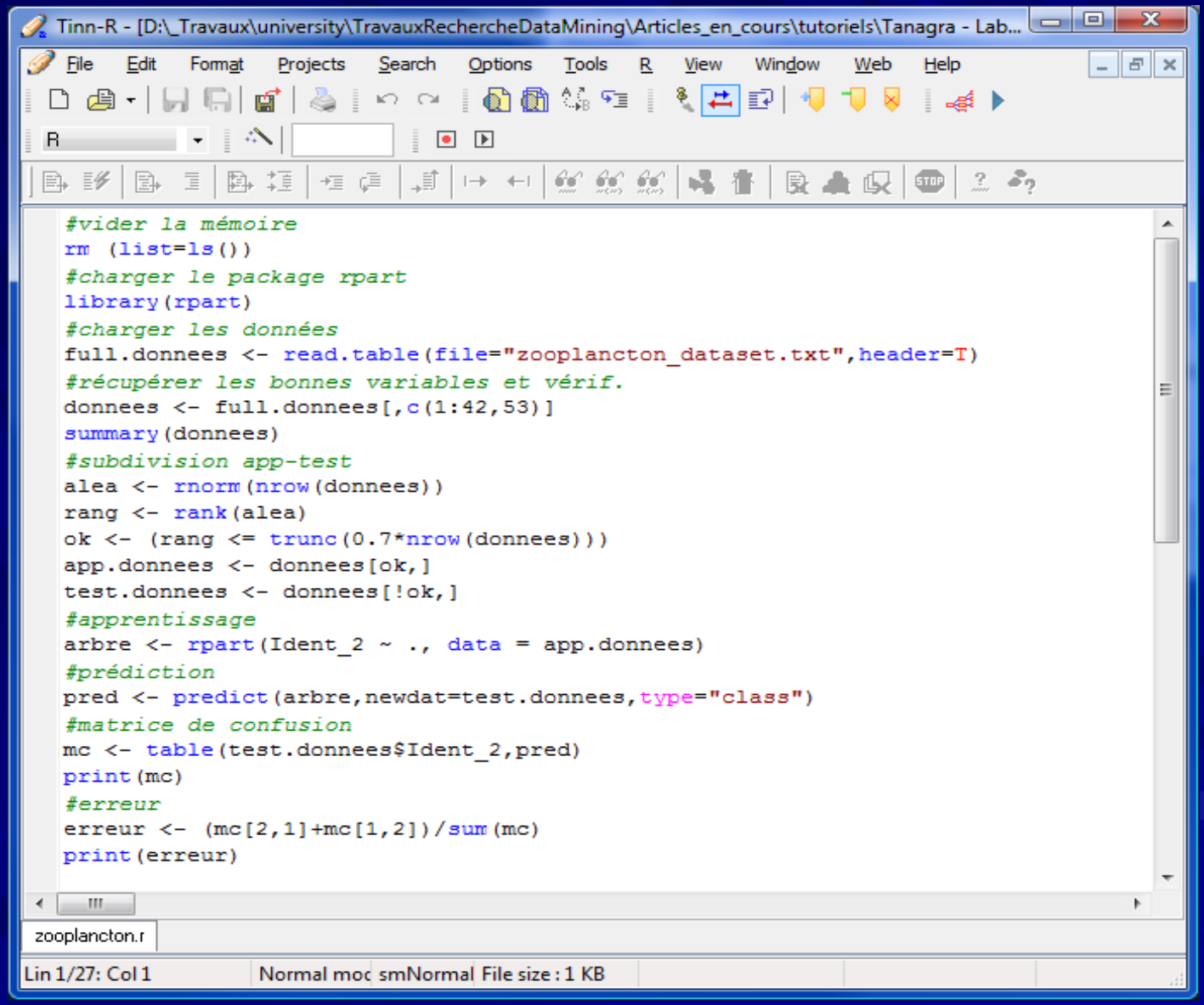
Culture Stat.
CORE R + Packages (plugins)
Ex. Package Weka

Doc. des méthodes très organisée
Des tutoriels partout

Mode programmation (langage S)

Quelques tentatives de création d'interfaces plus conviviales

R
Utilisation



RapidMiner
« Machine Learning »

Entreprise commerciale
Community Edition – Gratuite

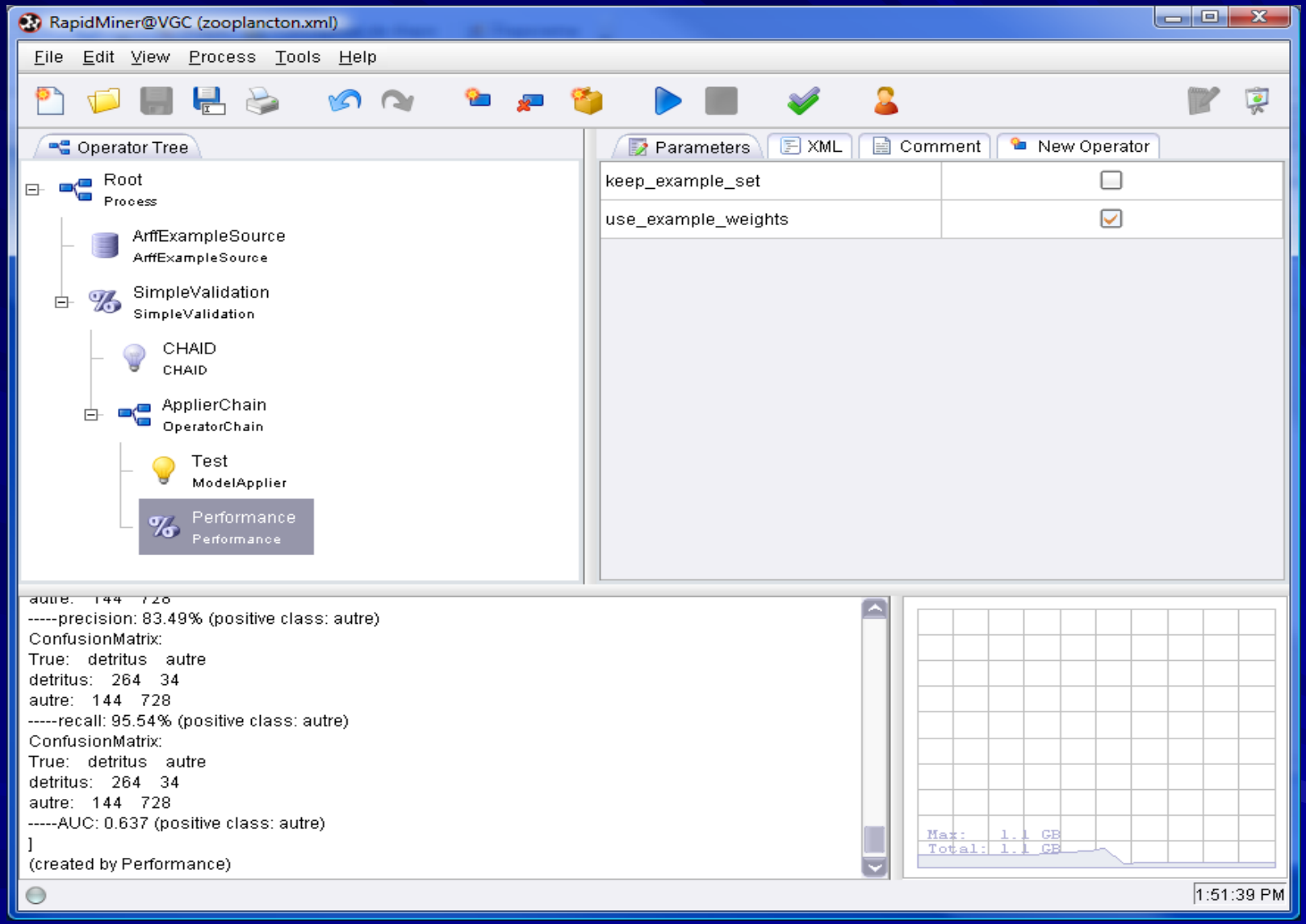
Dérivée de Yale (Licence GNU)
Il existe une version commerciale, sans code source
Code de calcul Weka, mais s'en démarque de plus en plus

Pas de documentation
Mais une multitude d'exemples « pré-programmées »

Mode diagramme arborescent

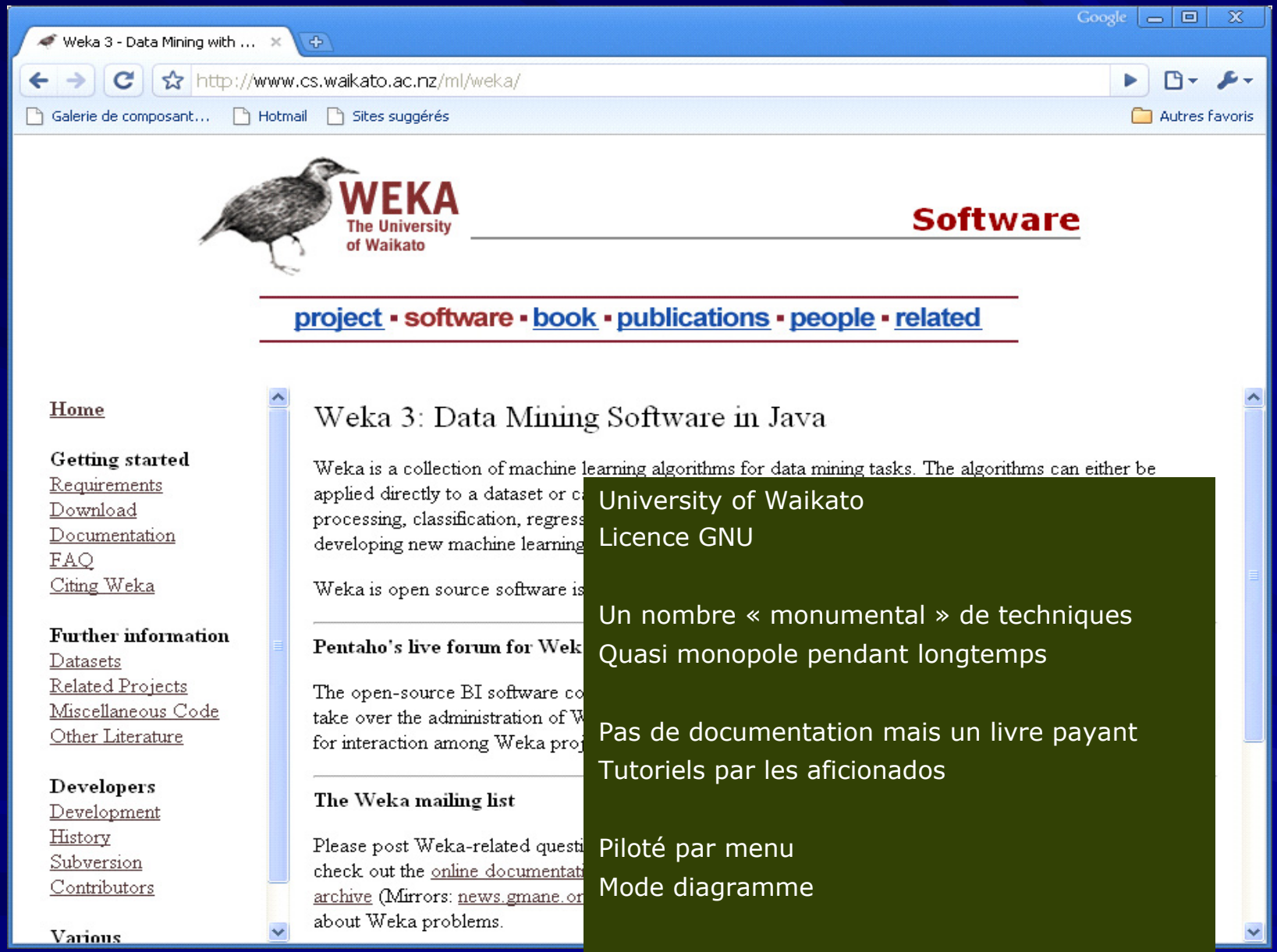
Une « profusion » de techniques de data mining

RapidMiner
Utilisation



Weka

« Machine Learning »



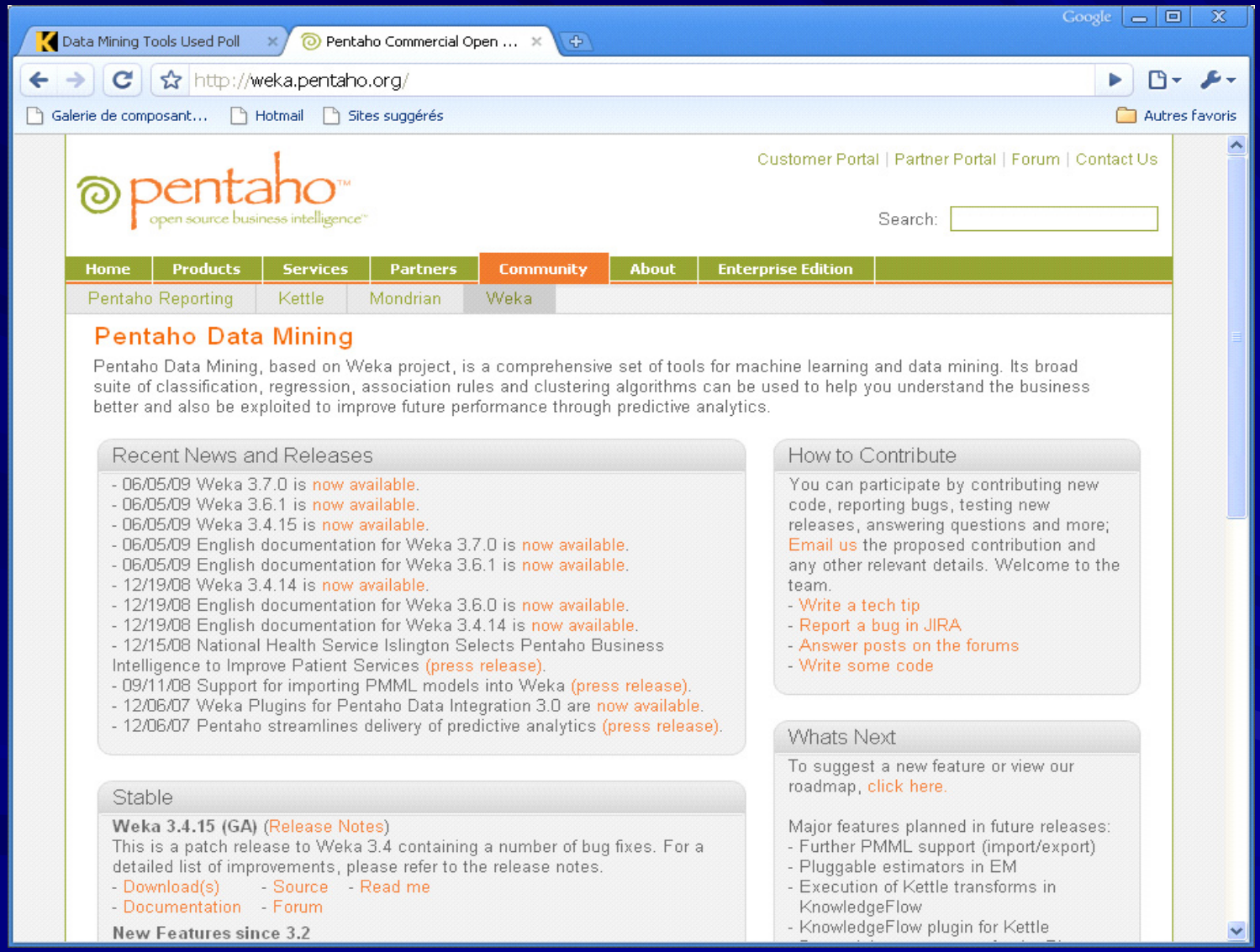
University of Waikato
Licence GNU

Un nombre « monumental » de techniques
Quasi monopole pendant longtemps

Pas de documentation mais un livre payant
Tutoriels par les aficionados

Piloté par menu
Mode diagramme

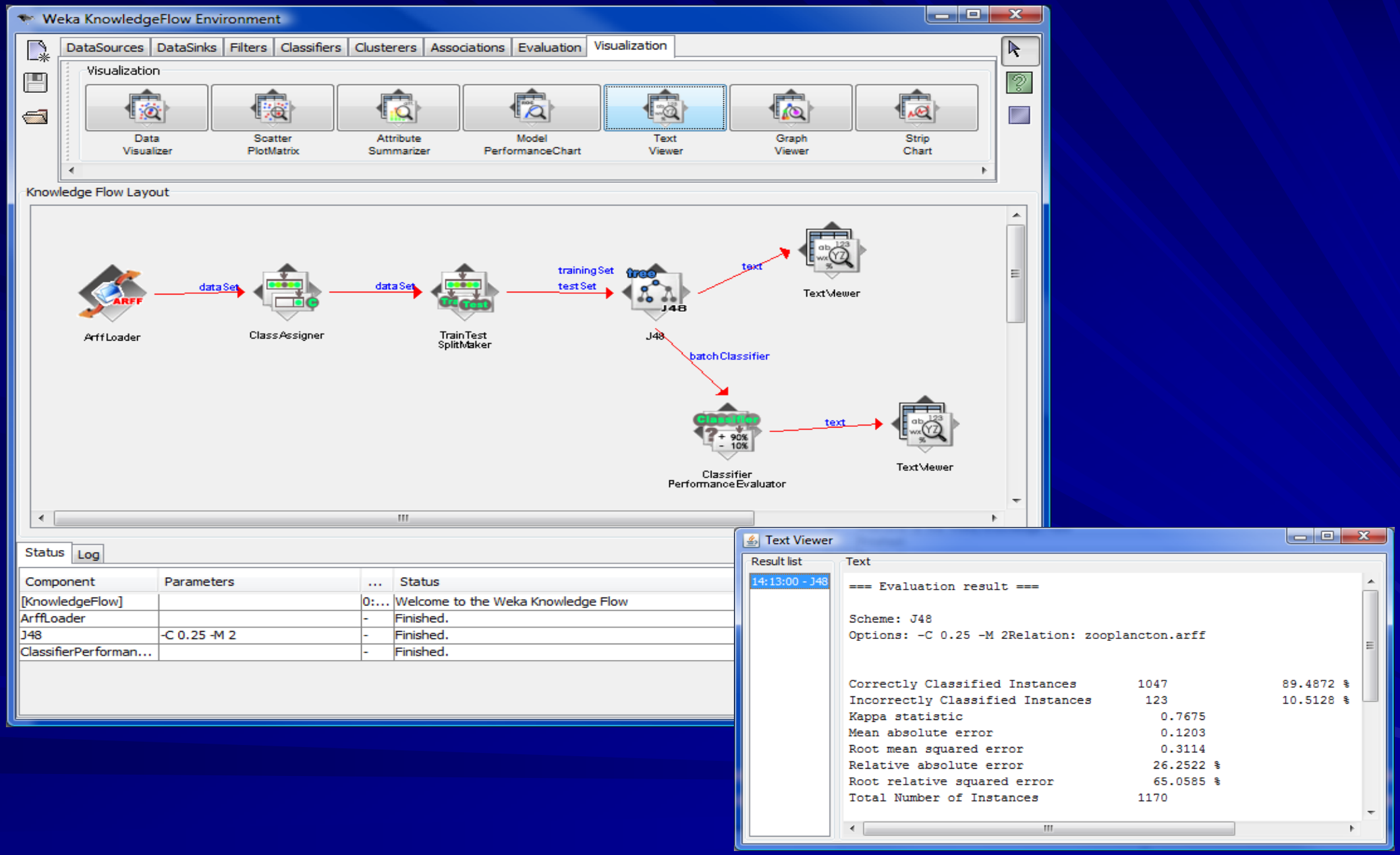
Mais quel avenir ? cf. version Pentaho


Data Mining Tools Used Poll
Pentaho Commercial Open ...
http://weka.pentaho.org/
Galerie de composant...
Hotmail
Sites suggérés
Autres favoris
Customer Portal | Partner Portal | Forum | Contact Us
pentaho
open source business intelligence
Search:
Home
Products
Services
Partners
Community
About
Enterprise Edition
Pentaho Reporting
Kettle
Mondrian
Weka
Pentaho Data Mining
Pentaho Data Mining, based on Weka project, is a comprehensive set of tools for machine learning and data mining. Its broad suite of classification, regression, association rules and clustering algorithms can be used to help you understand the business better and also be exploited to improve future performance through predictive analytics.
Recent News and Releases
- 06/05/09 Weka 3.7.0 is now available.
- 06/05/09 Weka 3.6.1 is now available.
- 06/05/09 Weka 3.4.15 is now available.
- 06/05/09 English documentation for Weka 3.7.0 is now available.
- 06/05/09 English documentation for Weka 3.6.1 is now available.
- 12/19/08 Weka 3.4.14 is now available.
- 12/19/08 English documentation for Weka 3.6.0 is now available.
- 12/19/08 English documentation for Weka 3.4.14 is now available.
- 12/15/08 National Health Service Islington Selects Pentaho Business Intelligence to Improve Patient Services (press release).
- 09/11/08 Support for importing PMML models into Weka (press release).
- 12/06/07 Weka Plugins for Pentaho Data Integration 3.0 are now available.
- 12/06/07 Pentaho streamlines delivery of predictive analytics (press release).
How to Contribute
You can participate by contributing new code, reporting bugs, testing new releases, answering questions and more; Email us the proposed contribution and any other relevant details. Welcome to the team.
- Write a tech tip
- Report a bug in JIRA
- Answer posts on the forums
- Write some code
Stable
Weka 3.4.15 (GA) (Release Notes)
This is a patch release to Weka 3.4 containing a number of bug fixes. For a detailed list of improvements, please refer to the release notes.
- Download(s) - Source - Read me
- Documentation - Forum
New Features since 3.2
Whats Next
To suggest a new feature or view our roadmap, click here.
Major features planned in future releases:
- Further PMML support (import/export)
- Pluggable estimators in EM
- Execution of Kettle transforms in KnowledgeFlow
- KnowledgeFlow plugin for Kettle


Data Mining Tools Used Poll
Discover why Pentaho Dat...
http://www.pentaho.com/products/data_mining/discover_data_mining.php
Galerie de composant...
Hotmail
Sites suggérés
Autres favoris
Products | Support & Services | Partners | Community
pentaho
open source business intelligence
Pentaho BI Suite
Enterprise Edition
Download
Pentaho Data Mining Enterprise Edition
Gain insight into hidden patterns and relationships in your data to discover indicators of future performance.
Discover
Try
Buy
Pentaho Data Mining
Once you've got analysis, reporting, and dashboards deployed, it's time to take your business intelligence (BI) to the next level by adding data mining and advanced analytics. This is a level of BI excellence that many organizations never manage to evolve to, however the importance of pushing ahead with advanced capabilities cannot be underestimated - they can provide a truly sustainable competitive advantage and enable your organization to maximize both its efficiency and effectiveness.
Data Mining is the process of running data through sophisticated algorithms to uncover meaningful patterns and correlations that may otherwise be hidden. These can be used to help you understand the business better and also exploited to improve future performance through predictive analytics. For example, data mining can warn you there's a high probability a specific customer won't pay on time based on an analysis of customers with similar characteristics.
Explore and Learn
Resources
Data Sheet
User Forum
Download
Deploying Data Mining Models
PMML Support
Request a Quote
Popular Links
BI Economics White Paper
More Links
Version 3
Pentaho BI Suite Enterprise Editic
User Friendly
Cloud Ready
Community Powere
Pentaho BI Suite 3.5
Design. Deploy. Escape.

Weka

Utilisation en mode « Knowledge flow »



5. Démonstration

Arbres de décision

Construction de l’arbre sur un échantillon d’apprentissage
Évaluation sur un échantillon test

Règles d’association

Construction et organisation des règles d’association

Traitement des gros volumes

Tutoriel Arbre + Comparatif diapo suivante

Performances comparées

Gros volumes

Logiciel	Temps de traitement (secondes)		Occupation mémoire (Mo)			
	Importation	Induction arbre	Avant lancement	Après importation	Pic traitement	Après induction
KNIME	47	270	92.6	160.4	245.8	245.8
ORANGE	90	130	24.9	259.5	795.7	795.7
R (package rpart)	24	157	18.8	184.1	718.9	718.9
RAPIDMINER	7	298	136.3	228.1	1274.4	1274.4
SIPINA	25	122	7.8	67.1	539.9	539.9
TANAGRA	11	33	7.0	53.1	121.6	73.5
WEKA	10	338	52.3	253.2	699.6	699.6

Arbres de décision, 500.000 obs., 21 descripteurs

Logiciel	Temps de traitement (sec.)		Taux d'erreur en validation croisée (%)	Occupation mémoire (Mo)			
	Importation	Calcul		Au lancement	Avec les données	Durant le traitement	Durant la validation croisée
ORANGE	95	690	4% (6/135)	25	118	317	406
RAPIDMINER JMySVMLearner	5	29	11% (15/135)	124	210	338	608
RAPIDMINER C-SVC (LIBSVM)	5	9	2% (3/135)	124	210	442	870
TANAGRA - SVM	12	130	4% (6/135)	7	337	393	393
TANAGRA C-SVC(LIBSVM)	12	11	4% (6/135)	7	337	406	406
WEKA - SMO	11	12	3% (4/135)	54	243	489	595

SVM, 135 obs., 31809 descripteurs

Les outils se tiennent, tout dépend des méthodes et des caractéristiques des données !!!

6. Conclusion

Conclusion

Quel logiciel pour quel contexte ?

Recherche (Data Mining)

Développer de nouvelles techniques

Les intégrer dans un environnement opérationnel

Pour des comparaisons à grande échelle

Les diffuser simplement et largement

→ Logiciel R

Avec les Packages

Utilisateur (Ou Recherche autre que Data Mining)

Contexte d'exploration des données

i.e. appliquer les techniques à des données,

Les faire coopérer (ces techniques)

Interpréter et publier les résultats

Enseignement

→ Les outils se valent

Critères de différenciation

Manipulation des données - texte/tableur/sgbd

Pouvoir les enchaîner (tous)

Traitement des très gros volumes (Knime ?)

Profusion des techniques (oui et non)

Outils graphiques (Knime, Orange)

Notoriété (Weka)

Et TANAGRA ?

Culture francophone du traitement des données

Machine Learning + Analyse de données et statistique

Un effort constant sur la documentation !!!