Méthodes d'apprentissage envisagées

Nicolas Lachiche, Agnès Braud, Cédric Wemmert, Pierre Gançarski

March 31, 2008

1 Introduction

Ce rapport vise à initier la tâche 3 du projet GeOpenSim. On s'intéresse ici à la détection et la caractérisation d'objets urbains. Ce rapport définit en terme d'apprentissage le problème posé par le projet et présente un état de l'art des techniques d'apprentissage automatique qui pourront être mises en œuvre, plus particulièrement celles utilisées dans l'équipe FDBT du LSIIT.

On distingue généralement deux grandes familles de problèmes d'apprentissage, les problèmes supervisés et non supervisés, la distinction se faisant au niveau de la connaissance du problème apportée par les données. On parle d'apprentissage supervisé quand les données permettent de savoir précisément ce qui doit être appris. Les données, appelées exemples, sont étiquetées par une classe et l'objectif est d'apprendre à classer une donnée inconnue. Dans le cas de l'apprentissage non supervisé, les données, appelées alors observations, ne sont pas étiquetées et l'objectif est de déterminer des classes.

La base commune est un ensemble de données basées sur la définition d'un langage de représentation. Ce langage introduit un deuxième critère pour classer les problèmes d'apprentissage. Le choix de ce langage est important dans la mesure où il influe considérablement sur la complexité de la phase d'apprentissage. On distingue notamment des problèmes définis dans un langage attribut-valeur, on parle alors de langage propositionnel, et dans un langage relationnel. Dans le cas propositionnel, les données sont définies en utilisant un nombre fixe d'attributs. Chaque donnée est donc définie par un ensemble ordonné de valeurs. En terme de base de données, cela revient à les stocker dans une unique table. Des techniques efficaces existent pour traiter ce type de données. Cependant de nombreuses données sont intrinsèquement structurées et représentées en termes de sous-composants. C'est le cas notamment de molécules qui sont représentées en termes d'atomes et de liaisons. On parle alors d'apprentissage relationnel. Ce dernier type d'apprentissage pose d'importants problème de complexité, dûs au fait que l'apprentissage repose sur des tests d'appariements entre données, tests très coûteux dans le cas relationnel. Une technique consiste à reformuler le problème d'apprentissage relationnel en un problème attribut-valeur. Ceci entraîne une perte d'information et il faut donc s'attacher à conserver les informations utiles à l'apprentissage. Le traitement

de données basées sur des représentations plus complexes, telles que des images, peut se ramener à ces cas.

Ce rapport est organisé comme suit. La section 2 définit le problème d'apprentissage posé dans le projet. Les sections suivantes présentent les techniques d'apprentissage dont nous disposons. La section 3 présente les travaux antérieurs utilisant des approches non-supervisées. La section 4 décrit l'apprentissage supervisé classique. La section 5 présente la propositionnalisation. La section 6 est consacrée au problème des instances multiples. La section 7 traite de l'apprentissage relationnel. La section 8 conclut en proposant les techniques que nous pensons mettre en œuvre, et en soulignant les difficultés du problème posé.

2 Le problème géographique

On s'intéresse à la détection et la caractérisation d'objets urbains. Ces objets sont définis parallèlement dans la tâche 2 du projet par les géographes. D'après le rapport RT2.1 (page 10), on distingue trois niveaux d'objets : micro qui correspondent à des objets géographiques simples, méso qui correspondent à des groupes d'objets (micro et/ou méso) qui ont un sens géorgraphique créés notamment sur des critères de proximité et de similarité, des objets macro qui correspondent à des populations d'objets (ensembles de population d'un type).

Les objets micro sont

- soit directement issus de la typologie définie dans la BD topo de l'IGN
- soit créés selon des méthodes définies par les géographes (par exemple les espaces ouverts selon une méthode décrite par Annabelle Boffet).

Le problème d'apprentissage posé vise à détecter et caractériser des objets méso. Ces objets méso sont définis par les géographes qui fournissent des exemples étiquetés de chacun. Ces exemples portent sur trois zones d'intérêt identifiées par les géographes sur Strasbourg et trois autres zones sur Orléans. Pour chaque objet les informations fournies comprennent

- Liste des objets appartenant à la classe
- une définition du domaine
- une description textuelle des critères utilisés pour décrire l'objet
- une liste d'évolution possibles
- des exemples issus des cartes topographiques à différentes dates montrant les évolutions possibles.

3 Approches non-supervisées

Dans le cadre de nos travaux antérieurs, nous nous sommes intéressés au problème de la classification automatique d'images de télédétection. Plusieurs méthodes ont été testées sur des images à haute résolution et ont montré de bons

résultats. Cependant, avec l'arrivée des images à très haute résolution, les approches de fouille et d'interprétation d'images consistant à considérer les pixels de façon indépendantes ont montré leurs limites pour l'analyse d'images complexes. Pour résoudre ce problème, de nouvelles méthodes s'appuient sur une segmentation préalable de l'image qui consiste en une agrégation des pixels connexes afin de former des régions homogènes au sens d'un certain critère.

Cependant le lien est souvent complexe entre la connaissance de l'expert sur les objets qu'il souhaite identifier dans l'image et les paramètres nécessaires à l'étape de segmentation permettant de les identifier. Nous avons proposé de modéliser la connaissance de l'expert sur les objets présents dans une image sous la forme d'une ontologie. Cette ontologie est utilisée pour guider un processus de classification non supervisée. Cette approche, qui ne nécessite pas d'exemple, permet de réduire le *fossé sémantique* entre l'interprétation automatique de l'image et la connaissance de l'expert. Plusieurs travaux ont été menés sur la représentation et l'utilisation de connaissances pour la classification d'images.

Tout d'abord, dans le cadre du projet Fodomust, une représentation de la connaissance experte sous la forme d'une ontologie de concepts a été proposée avec une définition des concepts sous la forme d'attributs représentés par des intervalles de valeurs acceptables. Un mécanisme d'appariement et de navigation permet à l'issue d'une segmentation, d'identifier la sémantique de certaines régions. Cet appariement compare les propriétés des régions avec les caractéristiques des concepts et permet d'assigner un degré d'appartenance des régions aux concepts de l'ontologie.

Ces travaux ont mis en évidence que l'étape de segmentation préalable a une forte importance sur l'identification par l'ontologie. En effet, si les objets sont mal construits, ils ne pourront pas être identifiés. Nous avons donc proposé un moyen de créer une segmentation guidée par les connaissances présentes dans l'ontologie. Dans le cadre du projet Ecosgil, nous avons créé une ontologie utilisant les mêmes mécanismes que pour celle créée dans le projet Fodomust. Dans cette optique, une première hiérarchie de concepts a été proposée. D'autre part, l'expert géographe nous a fourni un ensemble de connaissances pour améliorer l'interprétation des images. Un modèle enrichi de connaissances supplémentaires a été développé.

Des tests ont été effectués sur un extrait d'image à partir d'une segmentation. Les premiers résultats montrent que l'identification par l'ontologie "spectrale" (sans information contextuelle) présente de nombreuses erreurs. La *scoremap* permet de juger la qualité de l'identification des régions (score calculé par rapport à la vérité terrain fournie par l'expert). En utilisant des règles contextuelles (par exemple les voisins directs), plusieurs erreurs peuvent être corrigées (reconstruction du shore, élimination du bruit dans les dunes) mais certaines restent présentes.

L'introduction de données d'altitude (Lidar) sur la zone permet de réduire les erreurs de classification. En effet, la prise en compte de ces données permet de détecter des incohérences de labellisation. Au regard des degrés d'appartenance

des concepts identifiés par rapport à la connaissance de leurs altitude, des incohérences flagrantes sont détectées (régions sombres dans les dunes).

Enfin, nous avons voulu enrichir notre base de connaissances (ontologie spectrale, spatiale et contextuelle) en y insérant des connaissances sur des objets composites. Ces objets sont des compositions d'objets identifiés par l'ontologie. Pour cela, nous avons inséré des règles de construction dans la base de connaissances à partir de trois types de relations topologiques : *contains*, *surrounded by* et *connected to*. Cela nous a permis d'identifier dans une image THR, à la fois des objets simples, mais aussi des objets macro (compositions d'objets simples respectant certaines règles).

Dans le cadre du projet GeOpenSim, les objets des bases ne sont pas à construire car ils sont 'donnés' par la base de données BDTopo. Cela devrait permettre d'utiliser plus facilement des connaissances expertes pour les identifier en fonction de leur forme et contexte. De plus, dans la thèse d'Annabelle Boffet, de nombreuses approches pour la construction des objets macro ou méso sont présentées. On peut donc supposer que nous diposerons de ces objets et que nos travaux devront se focaliser sur l'identification automatique plutôt que sur la construction proprement dite.

Pour cela, l'approche de classification guidée par des connaissances que nous proposons d'utiliser, consiste en la succession de plusieurs étapes :

- construction des objets composites macro et méso (règles ou construction implicite)
- caractérisation des objets (forme et contexte)
- classification multiobjective semi-supervisée (guidée par une ontologie ou des exemples)

4 Apprentissage supervisé classique

Nous nous intéressons à présent à l'apprentissage supervisé, c'est-à-dire l'apprentissage d'une fonction, dans le cadre le plus courant, celui d'une représentation attribut-valeur. La littérature propose de nombreux algorithmes pour traiter ce problème. Néanmoins il n'existe aucun algorithme qui se comporte mieux que tous les autres sur tous les problèmes. Les différences se font notamment sur les types de données traités, la résistance au bruit contenu dans les données, l'intelligibilité de ce qui est appris.

On distingue notamment

- les techniques d'apprentissage à base d'instances. Aucune connaissance n'est vraiment apprise dans ce type d'apprentissage. La classification est faite par comparaison entre la nouvelle donnée et l'ensemble d'exemples disponibles, en partant de l'hypothèse que des données similaires ont la même classe.

- les arbres de décision. Ils construisent des hypothèses sous la forme d'arbres. Ces arbres permettent de représenter des ensembles de règles concluant sur différentes valeurs de classes. Ils résistent assez bien au bruit et produisent des modèles simples à interpréter.

- l'apprentissage d'ensembles de règles. Les règles sont représentées sous une forme logique, compréhensible, ce qui permet à l'utilisateur d'avoir une explication du résultat.

- les classeurs bayésiens. L'apprentissage s'appuie sur des statistiques. Il est rapide et de bonne précision, mais fournit peu d'explications.

- les réseaux neuronaux. Ce sont des approches de type boîte-noire pour lesquelles un résultat est produit sans que l'utilisateur n'ait d'explication. Ces approches sont théoriquement capables d'apprendre des fonctions très difficiles, mais sont un peu lente à entraîner.

- les machines à vecteur support. Ce sont des techniques relativement efficaces, mais peu compréhensibles.

- les techniques de régression. Elles permettent de prédire des attributs numériques. La plus connue est la régression linéaire.

- les arbres de régression. Ce sont des variantes des arbres de décision dédiées à l'apprentissage d'une fonction à valeur continue. Certaines offrent une relative compréhensibilité.

- SVR. C'est une variante des SVM pour la régression.

La mise au point d'un algorithme d'apprentissage adéquat est une étape importante d'un processus d'extraction de connaissances, mais ce n'est pas le seul. Il faut avant toute chose formuler le problème d'apprentissage, souvent réaliser un pré-traitement des données (normalisation, discrétisation, sélection d'attributs pertinents), il y a ensuite l'évaluation des résultats, le post-traitement.

Différentes plate-formes proposent des implantations de tout ou partie du processus et des algorithmes d'apprentissage, par exemple TANAGRA, SIPINA, R-project et ORANGE. Parmi elles, la plus populaire est sans doute Weka (The Waikato Environment for Knowledge Analysis) qui est proposée sous licence GNU GPL (http://www.cs.waikato.ac.nz/ ml/weka/). Cette plate-forme très complète intègre les différentes étapes de l'extraction de connaissances : pré-traitement des données, sélection d'attributs, fouille de données (clustering, classification, régression), visualisation.

L'équipe FDBT possède une solide expérience dans le domaine de l'apprentissage supervisé, notamment relationnel, qu'elle a pu appliquer à des données réelles complexes telles que des données chimiques, images du cerveau... utilisant différentes techniques parmi celles citées (classeurs bayésiens, arbres de régression, approches génétiques, réseaux neuronaux).

5 Propositionnalisation

Toutes les données ne sont pas disponibles directement sous la forme d'une table attribut-valeur. C'est le cas de beaucoup d'applications nouvelles, en particulier dans le domaine scientifique. C'est le cas des données géographiques que nous considérons, pour lesquelles la représentation des objets micro, en fait les attributs, n'est pas complètement définie. C'est encore pire pour les objets méso, que l'on ne sait pas encore représenter.

Citons une autre source de données non propositionnelles. Toutes les données déjà stockées dans des bases de données relationnelles, par exemple dans les systèmes d'information des entreprises, ne sont pas disponibles directement sous la forme d'une seule table attribut-valeur.

Lorsque l'on ne dispose que de programmes d'apprentissage propositionnels, comme la plupart des algorithmes implémentés dans weka, et que les données ne sont pas encore représentées en attribut-valeur, la solution la plus courante est de transformer manuellement les données en attribut-valeur. La transformation de données relationnelles en données propositionnelles s'appelle la propositionnalisation. C'est la solution par défaut quand on ne connaît pas d'approche plus expressive. Nous verrons plus loin qu'ils existent des techniques dédiées à des données relationnelles, mais celles-ci sont encore peu connues et donc peu utilisées.

La plupart du temps les données ne sont même pas disponibles sous une forme relationnelle, et elles sont directement modélisées en attribut-valeur. Remarquons qu'un avantage est que le choix des attributs est fait par un expert du domaine, et qu'ainsi les attributs sont généralement plus pertinents et des propriétés peuvent être introduites qui ne pourraient être automatiquement, en tous cas pas facilement, être dérivées d'une représentation relationnelle. Cependant, inversement, il y a souvent une perte d'information en propositionnalisant, caractérisé par le fait que l'on ne pourrait pas reconstruire la représentation relationnelle à partir de la représentation propositionnelle. De plus, la génération des attributs constitue une étape de l'extraction de connaissances, et cette étape peut être assistée par des programmes. Enfin, comme nous allons le voir, il existe quelques erreurs fréquentes lors de la propositionnalisation.

La propositionnalisation est définie à partir de données relationnelles, c'est-à-dire représentées par plusieurs tables attribut-valeur. Prenons l'exemple d'un patient, représenté par une table patient comportant des attributs constants tels que sa date de naissance, et par une table examen représentant les examens des patients avec des attributs tels que l'identifiant du patient, la date de l'examen, son poids, son pouls, indice de masse corporelle, etc.

Une technique simple pour passer de plusieurs tables à une seule table est de faire une jointure entre toutes les tables. Cette technique n'a de sens que si elle conduit à compléter la table "principale", c'est-à-dire celle correspondant à l'individu qui nous intéresse. Ainsi en cas de jointure entre la table patient et la table examen, c'est la table examen qui est complétée. Cela n'a un sens que si l'on s'intéresse aux examens, pas si l'on s'intéresse aux patients en général. Il ne faut pas oublier qu'en attribut-valeur, une ligne correspond à un individu et

réciproquement.

Une autre technique moins orthodoxe consiste à concaténer les colonnes de plusieurs tables relatives à l'individu principal. Cela peut avoir un sens si les informations relatives à l'individu sont réparties dans plusieurs bases (dossier scolaire, médical, employeur, logement,etc.). Mais cela n'a pas toujours de sens. Par exemple, si l'on considère les examens des patients, il y de fait un "premier" examen, un "second" examen, etc. pour chaque patient. On peut être tenté d'en faire autant de colonnes de la table patient... Pourtant le numéro d'ordre de l'examen n'a aucune signification : si le patient a un examen en plus au début, toutes les informations sont décalées et le pouls à la première consultation devient le pouls à la seconde consultation. En attribut-valeur, chaque colonne a un sens pour toutes les lignes.

Une troisième technique consiste à concaténer toutes les valeurs de chaque attribut, par exemple l'ensemble des valeurs du pouls de tous les examens du patient. Cette technique n'a un sens que si l'ensemble des valeurs devient LA valeur du nouvel attribut "ensemble des pouls", c'est-à-dire que le nombre d'ensembles possibles est limité et qu'un même ensemble apparaît chez plusieurs patients. Bref, on crée un nouvel attribut catégoriel. Sinon aucun algorithme propositionnel ne peut le traiter. En attribut-valeur, un attribut prend une valeur par ligne, pas un ensemble de valeurs.

Les trois techniques précédentes n'ont pas toujours un sens, mais elles peuvent s'appliquer dans des cas précis. Nous allons présenter des techniques qui s'appliquent dans tous les cas. Elles consistent à ramener les informations à l'individu principal.

La première approche s'appuie sur un quantificateur existentiel, "Est-ce qu'il existe un examen du patient tel que ... ?" La difficulté est de lister les conditions sur l'examen : "tel que son indice de masse corporelle est supérieur à 25", "tel que son indice de masse corporelle est supérieur à 25 et son pouls supérieur à 120", etc.

La seconde approche consiste à compter le nombre de fois où les conditions précédentes sont réalisées, par exemple "le nombre d'examens où l'indice de masse corporelle est supérieur à 25". On passe ainsi d'un attribut booléen à un attribut entier. L'expressivité est plus élevée, mais il faut disposer d'algorithmes capables de la gérer.

La troisième approche consiste à agréger les données du niveau inférieur, à l'aide de fonctions d'agrégation classiques (maximum, minimum, moyenne, mode, etc.) Ainsi, on cherchera les valeurs minimale, maximale et moyenne du pouls du patient, etc.

Ces approches se généralisent aisément au cas de plusieurs tables. Imaginons par exemple qu'à chaque examen, le patient puisse effectuer d'autres tests (nombre de globules blancs, échographie, etc.) enregistrés dans une table test dont les colonnes sont l'identifiant du patient, la date de l'examen, le nom du test et son résultat (positif ou négatif). On peut chercher "s'il existe un examen dont un test est négatif", "le nombre d'examens dont un test est négatif", mais également combiner les deux quantifications, "s'il existe un examen dont le nombre de tests négatifs est supérieur à 3".

Il est encore possible de construire d'autres attributs, par exemple, "le nombre d'années pendant lesquelles le pouls du patient est toujours supérieur à 120".

Il existe des programmes qui propositionnalisent des données relationnelles, le plus souvent à la volée, pour utiliser l'information en interne pour construire leur modèle. Les données propositionnelles sont rarement exportées dans un fichier. Dans tous les cas, les programmes s'appuient sur une des approches précédentes, mais ne les combinent pas en général. Evidemment, personne n'est capable de générer automatiquement tous les attributs imaginables.

La propositionnalisation fournit des algorithmes aider l'utilisateur dans la génération des attributs les plus pertinents possibles. Une bonne connaissance de ces approches peut inspirer de nouvaux attributs, et accessoirement permet d'éviter des erreurs de modélisation.

6 Instances multiples

Le problème des instances multiples est à mi-chemin entre le propositionnel et le relationnel. Les exemples sont des ensembles d'instances, où chaque instance est un vecteur attribut-valeur classique. Nous sommes en apprentissage supervisé. Chaque ensemble est étiqueté positif ou négatif.

Le problème a été introduit pour un problème chimique, MUSK, dans lequel on essaye de prédire si une molécule est active ou non. On sait qu'une molécule est active si elle peut se fixer sur un récepteur, qu'une molécule peut prendre un certain nombre de formes stables (appelées conformations), et donc une molécule est active si une de ses conformations peut se fixer au récepteur. On ne connait bien sûr pas la forme du récepteur, ni quelle conformation est la bonne, mais on peut mesurer expérimentalement si la molécule est active. Ainsi c'est l'ensemble des conformations, c'est-à-dire la molécule, qui est étiqueté active, si une des conformations est active. Une molécule est inactive si aucune de ses conformations n'est active. La prédiction d'activité est un des domaines où on rencontre des instances multiples.

Un autre domaine d'application est la recherche d'images par le contenu (content-based image retrieval), par exemple on recherche des images contenant un tigre. Une image est positive si elle contient un tigre, négative sinon, et une image est décrite par l'ensemble de ses régions, elles-même décrites par des vecteurs attribut-valeur (couleur, texture, forme).

La difficulté provient du fait que les instances ne sont pas étiquetées. Il n'est pas possible d'étiqueter comme positives toutes les instances d'un ensemble positif. En effet, sur l'exemple des images, cela reviendrait à considérer comme "tigre" toutes les régions d'une image contenant un tigre. On comprend que cela perturbe fortement l'apprentissage. Il n'est donc pas possible d'utiliser des algorithmes propositionnels directement.

De nombreux algorithmes ont été proposés, dérivés d'algorithmes propositionnels ou non. Certains s'appuient sur une propositionnalisation, en général une agrégation des propriétés des instances, et se ramènent à un problème

propositionnel au niveau des ensembles. D'autres utilisent une approche semi-supervisée et prédisent les étiquettes des instances. Une variante consiste à agréger les prédictions faites au niveau des instances. A ce moment, on somme parfois les contributions de la totalité des instances. Nous ne sommes plus dans le cadre classique où une instance explique la classe de l'ensemble, mais dans un nouveau cadre, dit collectif, où toutes les instances contribuent à la classe. Des tests récents montrent que les approches collectives obtiennent des résultats comparables à l'approche classique sur plusieurs jeux de test usuels. Cela s'explique probablement par la nature des données : peut-être que la classe est réellement fonction de la totalité des instances et qu'une seule ne suffit pas. L'exemple du patient et de ses examens peut techniquement être considéré comme un problème d'instance multiple. Imaginons qu'un patient soit étiqueté malade. On ne sait pas si un examen suffit à expliquer l'étiquette, par exemple le fait que sa tension soit supérieure à un seuil à un examen, ou si c'est le fait que sa tension soit supérieure à ce seuil sur tous les examens qui explique qu'il soit étiqueté malade. C'est justement un des objectifs de l'extraction de connaissances.

Le problème des instances multiples ne mérite plus vraiment son nom puisqu'il ne s'agit plus de trouver quelle instance est positive (et surtout pourquoi) dans un ensemble étiqueté positif. Il reste un cas particulier d'apprentissage relationnel où deux tables ont une association un à plusieurs. Les techniques d'instances multiples ne savent pas gérer plus de deux tables ou d'autres associations.

7 Relationnel

L'apprentissage relationnel concerne les données représentées sous la forme de plusieurs tables (et leurs associations) et plus généralement les données dont la représentation naturelle n'est pas une seule table attribut-valeur, par exemple les molécules sont représentées plus naturellement par une structure 2D, voire 3D.

Les techniques proposées par la programmation logique inductive s'appliquent aux données relationnelles puisque les prédicats sont équivalents à des tables. De plus ces techniques permettent de manipuler des connaissances du domaine sous la forme de règles. Ces systèmes génériques sont les seuls capables de gérer n'importe quel domaine représenté par un ensemble de faits et de règles. la contrepartie est qu'ils ne sont pas spécialisés, et souvent plus lents ou plus limités dans le volume de données qu'ils peuvent manipuler ou dans la complexité des hypothèses qu'ils peuvent explorer. Ils fournissent donc un bon moyen d'explorer plusieurs représentations complexes d'un même domaine, quitte à développer un système dédié une fois la représentation choisie.

La plupart des approches propositionnelles ont été généralisées au relationnel, y compris les approches s'appuyant sur des distances comme le clustering ou les plus-proches voisins. Nous avons nous-même développé des algorithmes généralisant les règles d'association, l'apprentissage d'ensembles de règles, de classeurs bayésiens, la sélection d'attributs, des réseaux neuronaux, des arbres

de classification et de régression.

Ces outils n'attendent que des données et des experts pour les utiliser.

8 Conclusion

Nous pensons distinguer deux étapes : la construction des régions méso et l'étiquetage des régions obtenues.

8.1 Construction

L'étape de construction des régions ne correspond pas à un problème d'apprentissage type. Nous envisageons d'en faire un pré-traitement indépendant. Dans ce cas les pistes que nous proposons sont :

- utiliser les réseaux de communication
- croissance de région
- autres approches non-supervisées

Ces pistes ne sont pas exclusives. Elles peuvent être combinées entre elles. Mais il est probable que la construction des régions ne soit pas dissociable de l'étape suivante. Par exemple dans la croissance de région, qui dit quand une région doit être découpée ou quand deux régions doivent être fusionnées ?

8.2 Etiquetage

L'étiquetage est *a priori* un problème typique d'apprentissage supervisé. Il s'agit de prédire l'étiquette d'une région.

Puisque cela ressemble à de la reconnaissance d'images, nous pensons nous ramener à des instances multiples (objet méso composé d'un ensemble d'objets micro) au besoin propositionaliser ou utiliser une approche collective pour construire des attributs globaux. Une autre piste, non exclusive, est d'intégrer des connaissances, par exemple les ontologies du LIV, en spécialisant des techniques de programmation logique inductive.

8.3 Intégration

La construction et l'étiquetage ne sont sûrement pas indépendants. L'étiquette d'une zone dépend de ce qu'elle contient. Faire croître la région "risque" de changer son étiquette.Comment le prendre en compte ?

Un problème complémentaire est la collecte d'exemples négatifs. Celle-ci est naturellement liée aux condiions d'application des algorithmes de caractérisation : si l'algorithme doit classer des régions construites automatiquement, il faut l'entrainer sur des données similaires. Il ne suffit pas de lui donner des régions construites à main et correspondant à d'autres classes !

La première étape de la fouille de données reste l'identification du problème.