Introduction à Weka (algorithme Apriori)

ING2 GI EISTI 2017-2018

PRÉSENTATION ET OBJECTIF

Dans ce TP nous allons découvrir la plateforme WEKA, qui offre une API d'algorithmes de data mining en Java, ainsi qu'un GUI pour la tester. **Ce TP est à faire sous Linux.**

INSTALLATION

WEKA est un ensemble de classes JAVA regroupées dans un JAR (comme JUNIT). Il suffit donc d'importer ce JAR dans une classe JAVA, ou d'utiliser directement les classes en ligne de commandes pour l'utiliser. Ce JAR est exécutable, donc il est également possible d'accéder directement à un GUI de manipulation avec la commande `java -jar weka.jar` .

L'ensemble des classes (JAR) ainsi que des fichiers d'exemples (ARFF) sont disponibles en téléchargement gratuitement à l'adresse www.cs.waikato.ac.nz/ml/weka

PREMIERS PAS

1. Lancez le GUI de WEKA puis sélectionner l'explorer
2. Vous avez le choix entre 6 onglets
    a. Preprocess: préparation des données
    b. Classify: algorithmes de classification (supervisés)
    c. Cluster: algorithmes de segmentation
    d. Associate: algorithmes de règles d'associations
    e. Select Attributes: choix des "meilleurs attributs"
    f. Visualize: affichage 2D des données
3. Sélectionnez l'onglet Preprocess et chargez le fichier `weather.nominal.arff`
4. Sélectionnez ensuite l'onglet Associate et choisissez l'algorithme Apriori que vous

avez étudié au premier semestre

5. Lancez l'algorithme sur le jeu de données avec le bouton start, sans modifier les paramètres.
6. Utilisez le bouton Log pour trouver quelle classe de l'API le GUI a utilisé.
7. Reproduisez le résultat en ligne de commande via la commande `java -cp weka.jar …` (vous avez accès à la complétion automatique) . N'oubliez pas l'argument `-t` pour choisir le fichier d'entrée.

MODIFICATION DES PARAMÈTRES

En cliquant sur le bouton droit dans la fenêtre face au bouton Choose de l'onglet Associate, vous pouvez modifier les paramètres de l'algorithme. Le bouton More explique le rôle de chacun des paramètres.

1. delta: fait décroître le support minimal tant qu'on a pas atteint le nombre de règles demandées ou que la valeur minimale du support n'est pas atteinte.
2. lowerBoundMinSupport: valeur minimum du support
3. metricType: mesure pour classer les règles produites
    a. Confidence (confiance)
    b. Lift (amélioration)
    c. Leverage (proportion d'exemples concernés par uniquement la partie droite ou gauche de la règle)
    d. Conviction (comme amélioration pour les exemples où la partie droite de la règle n'est pas respectée)
4. minMetric: valeur minimale de la mesure pour considérer une règle
5. numRules: nombre de règles maximum à produire
6. removeAllMissingCols: supprime les valeurs manquantes
7. significanceLevel: test statistique (non utilisé dans ce TP)
8. upperBoundMinSupport: valeur initiale du support

Faites varier les différents paramètres et comparez les résultats obtenus.

UTILISATION EN PRODUCTION

L'utilisation du GUI est très pratique d'un point de vue pédagogique, mais évidemment pas adaptée à une utilisation en production

1. Utilisez la commande l'API WEKA et gnuplot pour tracer des graphes afin de

comparer le temps d'exécution de l'algorithme Apriori en fonction du seuil du support et du seuil de confiance. Pour utiliser gnuplot, il suffit de créer un fichier contenant les valeurs (une ligne par point de la forme x,y), puis sous gnuplot de l'afficher via la commande `plot nomfichier with line`.

2. A partir du jeu de données retail_small.dat (la version complète retail.dat nécessite trop de RAM pour Weka sur vos PC), créez une application (en JAVA ou dans le langage de votre choix permettant l'exécution de commandes externes) permettant de :
    a. créer un fichier arff compatible avec l'algorithme Apriori implémenté dans WEKA
    b. lancer l'algorithme Apriori de WEKA
    c. interpréter les résultats (parsing des règles)
    d. écrire une fonctionnalité de prédiction issue du résultat de l'algorithme, permettant de saisir un panier de courses et de proposer des articles susceptibles d'être intéressants