Tema 3
Actividad 1

Autor
Juan José Méndez Torrero

Uso de Weka con datos

Introducción

El conjunto de datos seleccionado para esta tarea se llama credit-g. Este conjunto de datos cuenta con un total de 1000 instancias y 21 atributos. Entre los atributos, se pueden observar tanto de tipo numérico como de tipo nominal. Los atributos son los siguientes:

- Checking_status
- Duration
- Credit_history
- Purpose
- Credit_amount
- Savings_status
- Employment
- Installment_commitment
- Personal_status
- Residence_since
- Property_magnitude
- Age
- Other_payment_plans
- Housing
- Existing_credits
- Job
- Num_dependents
- Own_telephone
- Foreign_worker
- Class

El objetivo de este conjunto de datos es clasificar un tipo de crédito en bueno o malo, según los atributos anteriormente mostrados. Antes de aplicar el algoritmo Apriori para generar las reglas de asociación, se le ha aplicado un filtro ***NumericToNominal*** al conjunto de datos para convertir los valores numéricos en nominales.

Resultados

Una vez hemos transformado los datos de numérico a nominal, se ha aplicado el algoritmo Apriori, con la opción car a True, sobre el conjunto de datos. Tras ejecutar el algoritmo con varias configuraciones, se ha podido observar que el mejor soporte mínimo encontrado es de 0.5, y una confianza del 0.6. En la Figura 1 se pueden observar los resultados obtenidos. Como se puede ver, hemos limitado el número de reglas a un total de 10 reglas, en las que el consecuente sólo contiene un ítem, que en este caso es el atributo ***class***. Además, también cabe destacar que las reglas obtenidas sólo hacen referencia a la clase ***good***. Esto quiere decir que no se ha encontrado ninguna regla, entre las 10 mejores, que pueda identificar a la clase ***bad***. De estas reglas obtenidas, se puede observar que en 538 ocasiones, se puede obtener una clase good si el atributo other_parties toma el valor None y que el atributo other_payment_plans tome el valor None, con una confianza del 73%.

```
Apriori
=======

Minimum support: 0.5 (500 instances)
Minimum metric <confidence>: 0.6
Number of cycles performed: 10

Generated sets of large itemsets:

Size of set of large itemsets L(1): 5

Size of set of large itemsets L(2): 7

Size of set of large itemsets L(3): 2

Best rules found:

 1. housing=own 713 ==> class=good 527    conf:(0.74)
 2. housing=own foreign_worker=yes 685 ==> class=good 502    conf:(0.73)
 3. other_parties=none other_payment_plans=none 742 ==> class=good 538    conf:(0.73)
 4. other_payment_plans=none 814 ==> class=good 590    conf:(0.72)
 5. other_payment_plans=none num_dependents=1 698 ==> class=good 503    conf:(0.72)
 6. other_parties=none other_payment_plans=none foreign_worker=yes 718 ==> class=good 516    conf:(0.72)
 7. other_payment_plans=none foreign_worker=yes 782 ==> class=good 560    conf:(0.72)
 8. other_parties=none num_dependents=1 767 ==> class=good 539    conf:(0.7)
 9. other_parties=none 907 ==> class=good 635    conf:(0.7)
10. other_parties=none num_dependents=1 foreign_worker=yes 749 ==> class=good 524    conf:(0.7)
```

Figura 1

Además, se puede observar que las reglas obtenidas no son muy explicativas, es decir, no se han extraído suficientes reglas con la suficiente confianza como para poder clasificar un nuevo patrón introducido en el conjunto de datos.

En cambio, si el atributo usado como consecuente cambia, por ejemplo, usamos la variable personal_status, el soporte mínimo a usar disminuye, y la confianza de las reglas obtenidas disminuye en un 20% aproximadamente. En la Figura 2, se puede observar que, la confianza de las reglas obtenidas no supera el 60%, con lo que estas reglas no son muy discriminantes a la hora de clasificar una nueva instancia.

```
Apriori
=======

Minimum support: 0.35 (350 instances)
Minimum metric <confidence>: 0.3
Number of cycles performed: 13

Generated sets of large itemsets:

Size of set of large itemsets L(1): 6

Size of set of large itemsets L(2): 9

Size of set of large itemsets L(3): 3

Best rules found:

 1. other_parties=none class=good 635 ==> personal_status=male single 365    conf:(0.57)
 2. class=good 700 ==> personal_status=male single 402    conf:(0.57)
 3. housing=own 713 ==> personal_status=male single 408    conf:(0.57)
 4. other_parties=none housing=own foreign_worker=yes 625 ==> personal_status=male single 357    conf:(0.57)
 5. housing=own foreign_worker=yes 685 ==> personal_status=male single 391    conf:(0.57)
 6. other_parties=none housing=own 647 ==> personal_status=male single 368    conf:(0.57)
 7. foreign_worker=yes class=good 667 ==> personal_status=male single 379    conf:(0.57)
 8. other_parties=none 907 ==> personal_status=male single 495    conf:(0.55)
 9. foreign_worker=yes 963 ==> personal_status=male single 525    conf:(0.55)
10. other_parties=none foreign_worker=yes 880 ==> personal_status=male single 479    conf:(0.54)
```

Figura 2

```
Apriori
=======

Minimum support: 0.45 (450 instances)
Minimum metric <confidence>: 0.3
Number of cycles performed: 11

Generated sets of large itemsets:

Size of set of large itemsets L(1): 5

Size of set of large itemsets L(2): 6

Size of set of large itemsets L(3): 3

Best rules found:

 1. other_payment_plans=none num_dependents=1 foreign_worker=yes 677 ==> job=skilled 457    conf:(0.68)
 2. other_payment_plans=none num_dependents=1 698 ==> job=skilled 470    conf:(0.67)
 3. other_payment_plans=none foreign_worker=yes 782 ==> job=skilled 513    conf:(0.66)
 4. num_dependents=1 foreign_worker=yes 819 ==> job=skilled 535    conf:(0.65)
 5. other_parties=none other_payment_plans=none foreign_worker=yes 718 ==> job=skilled 469    conf:(0.65)
 6. num_dependents=1 845 ==> job=skilled 551    conf:(0.65)
 7. other_parties=none num_dependents=1 foreign_worker=yes 749 ==> job=skilled 488    conf:(0.65)
 8. other_parties=none num_dependents=1 767 ==> job=skilled 499    conf:(0.65)
 9. other_payment_plans=none 814 ==> job=skilled 529    conf:(0.65)
10. other_parties=none other_payment_plans=none 742 ==> job=skilled 480    conf:(0.65)
```

Figura 3

Por último, si volvemos a cambiar el atributo del consecuente, por ejemplo, según el trabajado (Job) que tenga una persona. De la Figura 3 se puede observar que se ha obtenido una regla en la que el antecedente cuenta con tres ítems, y cuenta con una confianza del 68%.

Conclusiones

Tras ejecutar, con distintas configuraciones, el algoritmo Apriori para extraer reglas de asociación, se puede observar que las reglas con mayor confianza son las que cuentan con un número mayor de instancias, es decir, en la Figura 1, se puede observar que en el consecuente, el atributo class sólo toma el valor ***good***. Esto es debido a la diferencia de instancias de cada una de las clases, es decir, las dos clases (good y bad), tuvieran el mismo número de instancias, las reglas obtenidas llegarían a ser más restrictivas, permitiendo al experto utilizar esas reglas para clasificar nuevas instancias que se añadan en un futuro al conjunto de datos.

Además, se ha podido observar que, incluso aumentando el número de reglas al número de instancias, no se ha podido extraer ninguna regla que clasifica una nueva instancia como clase ***bad***. Eso puede ser debido al bajo número de instancias con las que cuenta esta clase (300 de 1000).