2016/2017

Ingeniería basada en el conocimiento

Opiniones sobre hoteles

Héctor Rodríguez Salgado & Marta Loriente Nieves

Máster en Ingeniería Informática – ESEI – Universidad de Vigo

# Introducción

El objetivo principal del análisis de datos es la construcción y comparación de diferentes modelos de minería de datos, enfocando la discusión en sus resultados.

Se evaluarán un conjunto de datos siguiendo tres enfoques de análisis diferentes: modelos predictivos, modelos de agrupamiento y modelos de asociación.

Todos estos modelos se construirán utilizando la herramienta de minería de datos WEKA y sus algoritmos tal como se proporcionan, sin modificar ninguno de sus detalles de implementación.

# Exploración de datos

Para realizar los análisis se ha escogido un conjunto de datos de alrededor de 12.000 instancias que recogen las opiniones de los clientes que han visitado distintos hoteles. Dicha información se ha obtenido del buscador de hoteles *Tripadvisor*.

En un inicio, el *dataset* contenía la siguiente información:

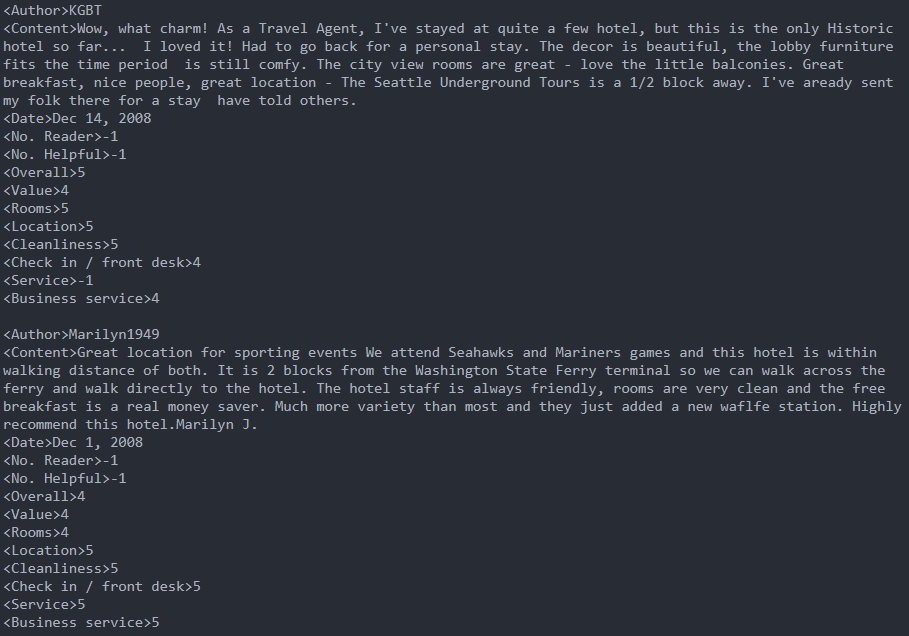


Ilustración - Dataset inicial

Para el análisis, únicamente se requería la opinión del cliente. Por tanto, se ha realizado un preprocesado de los *datasets*, obteniendo como resultado:

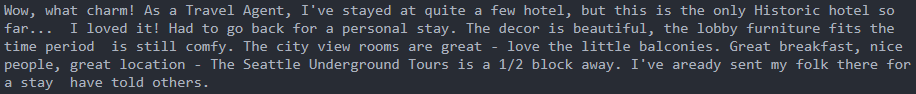


Ilustración - Dataset tras el procesado

De esta forma, cada instancia contiene la opinión de un cliente en un hotel.

Una vez realizado dicho proceso, se han escogido las 50 palabras que más se repiten en todas las opiniones para determinar si la opinión del cliente ha sido buena o mala hacia el hotel en el que se ha hospedado. Dichas palabras pasarán a ser los atributos de interés para el análisis.

A continuación, se muestra el listado de las palabras escogidas: “hotel, small, bed, parking, rooms, friendly, good, got, view, price, seattle, helpful, day, bathroom, area, like, excellent, night, comfortable, desk, didnt, breakfast, dont, little, city, best, recommend, street, hotel, place, floor, free, just, restaurant, staff, right, clean, great, room, nice, really, stay, lobby, stayed, hotels, service, nights, location, time, did, walk”.

# Análisis de predicción de atributos

La primera tarea es determinar si es posible predecir si la opinión de un cliente es buena o mala en base a las palabras que contiene su opinión. Para ello se han ejecutado diferentes algoritmos predictivos y se ha observado el resultado.

## Lazy learning

El primer algoritmo que se ha decidido utilizar ha sido de tipo “lazy learning: (k-) nearest-neighbour”, conocido como IBK en WEKA. Se ha decidido realizar una predicción para cada atributo del dataset, variando el número de vecinos en cada análisis. La tendencia en cada uno de los análisis es que a más vecinos, mejor es la predicción realizada, ya que el porcentaje de acierto es mayor.

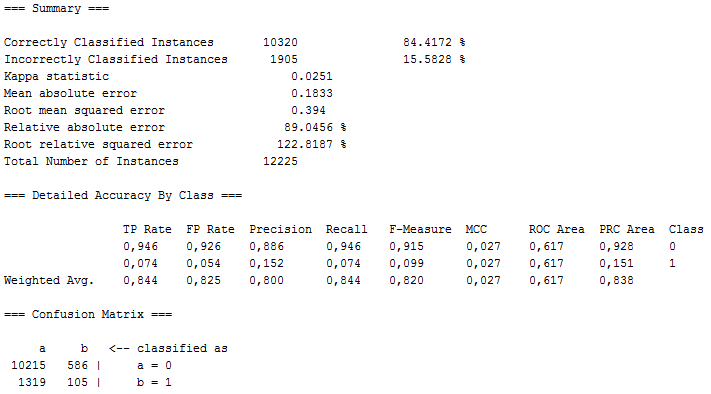


Ilustración - IB1

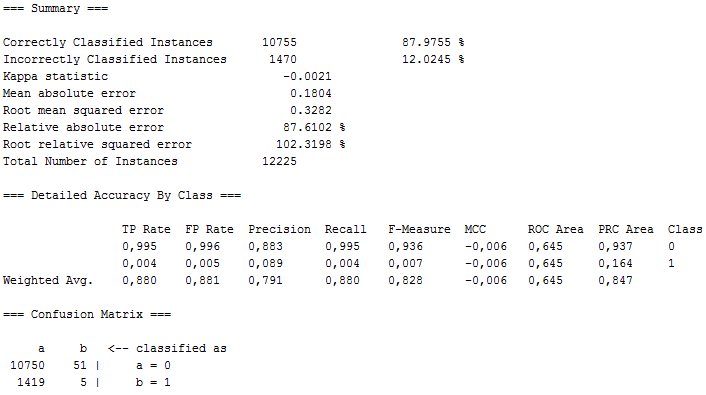


Ilustración - IB5

Como se puede observar en los resultados anteriores, a mayor número de vecinos, mayor es el porcentaje de acierto de este algoritmo. Aunque observando los aciertos obtenidos, se reduce considerablemente el número de instancias clasificadas correctamente para el tipo ‘b’ y, en cambio, aumenta para el tipo ‘a’. Esto indica que no siempre que se obtenga un porcentaje de acierto mayor, el análisis va a ser mejor, como ocurre en este caso, ya que dicho porcentaje, a efectos prácticos, sólo se refleja en el tipo ‘a’, no en el tipo ‘b’.

## Decision trees

El concepto principal detrás de aprendizaje árbol de decisión es que, a partir de los datos de entrenamiento, se va a construir un modelo predictivo que se asigna a una estructura de árbol. El objetivo es lograr la clasificación perfecta con el número mínimo de la decisión, aunque no siempre es posible debido al ruido o inconsistencias en los datos.

Dentro de este tipo de algoritmos se ha decidido comenzar con el tipo J48.

El resultado que se ha ofrecido utilizando un testeo de cross-validation de 10 folds es el siguiente:

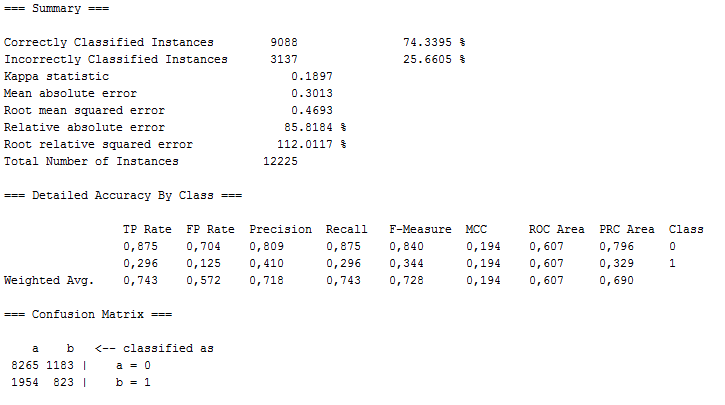


Ilustración - J48

tomando como clase el atributo *good*. Con este atributo se obtuvo un porcentaje de aciertos del 74%. Si se mira la matriz de confusión se ve que hay más errores cuando se predice que una opinión es *good* cuando en realidad no lo es. Con ello se puede decir con certeza que este algoritmo no es bueno para realizar un estudio para este caso.

## Rules

Dentro de este tipo de algoritmos el primero que se va a utilizar será JRIP cuyo objetivo es podar para cortar la reducción de errores, obteniendo como resultado en base al atributo *good* como clase del algoritmo:

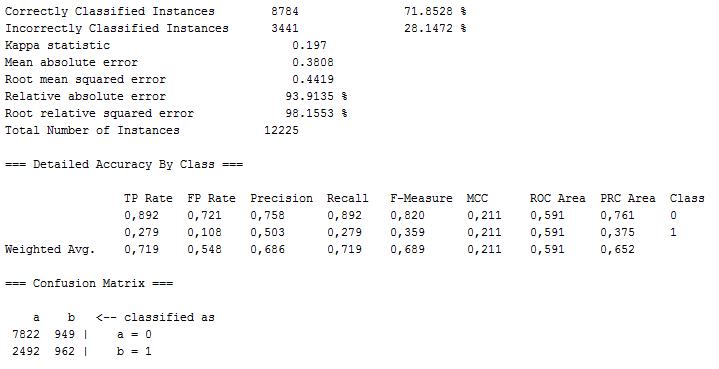


Ilustración - JRIP

Se obtiene casi un 72% de acierto, pero si se observa detenidamente la matriz de confusión se produce más errores al predecir los atributos good que sean positivos como negativos. Por lo tanto, en base a esta última afirmación se puede decir que es un mal algoritmo de predicción para este caso en concreto.

El siguiente algoritmo de este tipo será OneR y ZeroR manteniendo el atributo *good* como clase. En base a eso obtenemos exactamente los mismos resultados que con el algoritmo JRIP. Sin embargo, con el algoritmo PART se obtienen peores como se muestra a continuación:

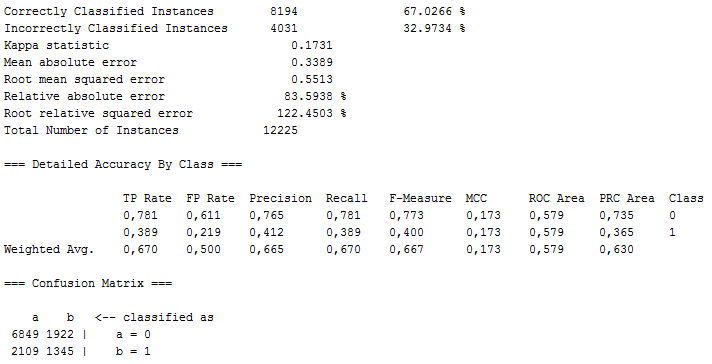


Ilustración - Part

Tiene un menor porcentaje de acierto con respecto a los anteriores algoritmos, pero se sigue manteniendo que se obtienen más errores en la matriz de confusión.

Con respecto a los árboles de decisión se puede afirmar que este tipo de algoritmo no nos aporta mejores resultados, con lo cual queda eliminado.

# Clustering

El *clustering* o agrupamiento es la tarea de agrupar un conjunto de objetos de tal manera que los miembros del mismo grupo (llamado clúster) sean más similares, en algún sentido u otro. ES la tarea principal de la minería de datos exploratoria y es una técnica común en el análisis de datos estadísticos.

## K-Means

Es un método de agrupamiento, que tiene como objetivo la participación de un conjunto de *n* observaciones en *k* grupos en el que cada observación pertenece al grupo cuyo valor medio es más cercano.

Analizamos con el número de clúster igual a 10. También ponemos el atributo displayStdDevs a True (atributo que nos mostrará información más detallada de los clústers).

Vamos a ejecutar el .arff con dos el modo de testeo *use training set*.

A continuación, se explica paso a paso las partes que se han obtenido al ejecutar este dataset. La primera captura nos muestra el número de iteraciones junto con el error de la suma cuadrática, que se ha comprobado que cuánto mayor número de clúster es menor. La siguiente sección muestra qué puntos aleatorios se pusieron al inicio para saber si son iguales, distintos…

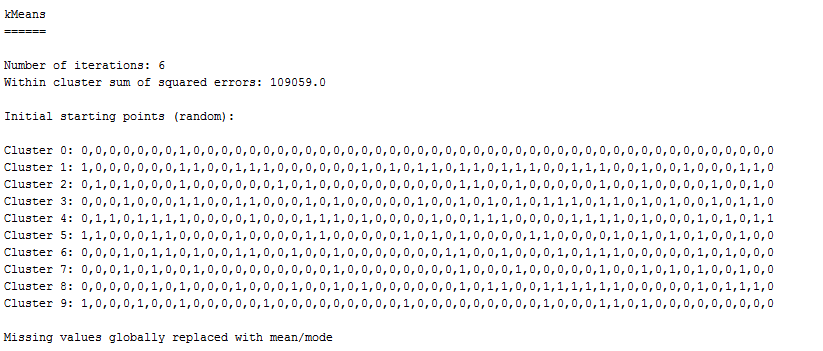


Ilustración - SimpleKMeans

Se muestran ahora los clústers que se han realizado, y la aparición de las palabras en cada uno de ellos.

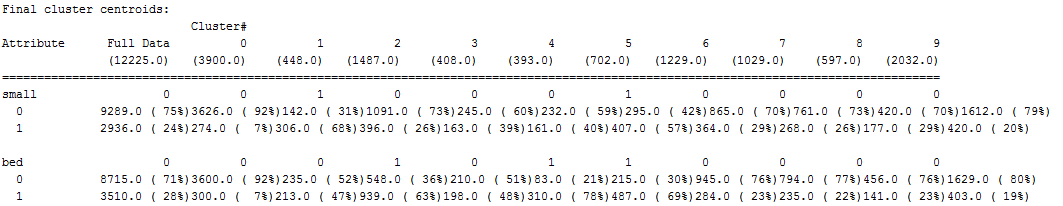
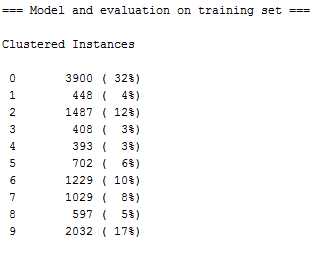


Ilustración - SimpleKMeans

Tal y como se muestra en la ilustración anterior se han realizado 10 agrupaciones, con tantos miembros que se indica justo debajo del número del clúster. Seguidamente se toma cada una de las palabras del dataset y se detalla la aparición en cada clúster. Este nivel de detalle es gracias al atributo mencionado anteriormente *displayStdDevs*.

Para finalizar se interpretan los clusters obtenidos:

Como se puede ver se ha obtenido el número de instancias para cada uno de los clusters. Es decir, 3900 ocurrencias se han producido en el cluster 0, lo que es un 32% del total.

## EM

El algoritmo de expectativa de maximización es un método iterativo para la búsqueda de la máxima verosimilitud, teniendo en cuenta variables latentes.