## Clasificación de blocks de página

### Descripción

Los 5,473 ejemplos provienen de 54 documentos distintos. Cada observación se refiere a un bloque. Todos los atributos son numéricos.

### Información de atributos

A continuación se presentan los atributos y metadatos que describen a la base de datos.

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Dominio |
| Altura | Entero | 1 - 187 |
| Longitud | Entero | 1 - 340 |
| Área | Entero | 1 - 1224 |
| Excentricidad | Real | 0 - 9.111 |
| Porcentaje de pixeles negros | Real | 0 - 0.374 |
| Percentaje de pixeles negros RLSA | Real | 0 - 0.866 |
| Media de transisiones blanco-negro | Real | 0 - 132 |
| Pixeles negros en mapa de bits original | Entero | 0 - 303 |
| Pixeles negros en mapa de bits RLSA | Entero | 0 - 829 |
| Transiciones blanco-negro en original | Entero | 0 - 173 |

### Tiempos del algoritmo

El registro de sucesos de Weka, indicó lo siguiente cuando **KNN** (K-Vecinos) fue igual a **1** (1 Grupo):

15:28:34: Empezó el viernes, 25 abril 2014

15:29:51: Empezó weka.classifiers.lazy.IBk

15:29:51: Comando:weka.classifiers.lazy.IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

15:30:02: Terminó weka.classifiers.lazy.IBk

El registro de sucesos de Weka, indicó lo siguiente cuando **KNN** (K-Vecinos) fue igual a **3** (3 Grupos):

15:28:34: Empezó el viernes, 25 abril 2014

15:32:06: Empezó weka.classifiers.lazy.IBk

15:32:06: Comando:weka.classifiers.lazy.IBk -K 3 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

15:32:18: Terminó weka.classifiers.lazy.IBk

El registro de sucesos de Weka, indicó lo siguiente cuando **KNN** (K-Vecinos) fue igual a 5 (5 Grupos):

15:28:34: Empezó el viernes, 25 abril 2014

15:32:06: Empezó weka.classifiers.lazy.IBk

15:32:06: Comando:weka.classifiers.lazy.IBk -K 3 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

15:32:18: Terminó weka.classifiers.lazy.IBk

### Instancias

Informe de instancias para **KNN=1** (1 grupo)

Instancias clasificadas correctamente 5255 96.0168 %

Instancias clasificadas incorrectamente 218 3.9832 %

Informe de instancias para **KNN=3** (3 grupos)

Instancias clasificadas correctamente 5257 96.0534 %

Instancias clasificadas incorrectamente 216 3.9466 %

Informe de instancias para **KNN=5** (5 grupos)

Instancias clasificadas correctamente 5250 95.9255 %

Instancias clasificadas incorrectamente 223 4.0745 %

### Matriz de confusión

Matriz de confusión obtenida con Weka cuando **KNN** (K-Vecinos) fue igual a **1** (1 Grupo):

a b c d e <-- Clasificado como

4836 28 3 10 36 | a = 1

49 276 0 2 2 | b = 2

9 0 18 0 1 | c = 3

18 2 0 66 2 | d = 4

54 1 0 1 59 | e = 5

Matriz de confusión obtenida con Weka cuando **KNN** (K-Vecinos) fue igual a **3** (3 Grupo):

a b c d e <-- Clasificado como

4863 20 3 7 20 | a = 1

68 257 0 2 2 | b = 2

10 0 17 0 1 | c = 3

21 1 0 65 1 | d = 4

58 2 0 0 55 | e = 5

Matriz de confusión obtenida con Weka cuando **KNN** (K-Vecinos) fue igual a **5** (5 Grupo):

a b c d e <-- Clasificado como

4863 19 5 4 22 | a = 1

75 252 0 2 0 | b = 2

11 0 16 0 1 | c = 3

19 3 0 64 2 | d = 4

57 3 0 0 55 | e = 5

### Gráfica de resultados

A continuación se presenta un gráfico que muestra la relación de lo obtenido. Se muestran los registros clasificados correcta e incorrectamente, así como el tiempo que llevó a Weka hacer la clasificación.

La siguiente gráfica muestra los resultados obtenidos para cuando **KNN** (K-Vecinos) fue igual a **1** (1 Grupo):

La siguiente gráfica muestra los resultados obtenidos para cuando **KNN** (K-Vecinos) fue igual a **3** (3 Grupos):

La siguiente gráfica muestra los resultados obtenidos para cuando **KNN** (K-Vecinos) fue igual a 5 (5 Grupos):

### Conclusiones

Como se puede observar en las gráficas, entre mayor sea el número de grupos a clasificar (valor de KNN, K en el algoritmo) mayor será la clasificación incorrecta, resulta curioso ver que aun cuando la clasificación se hace hacia un solo grupo, el algoritmo tira un margen de error. Por la velocidad del procesamiento del algoritmo, puede calificarse como bueno, o por lo menos para el presente ejemplo.