

HML - A Machine Learning Library in Haskell

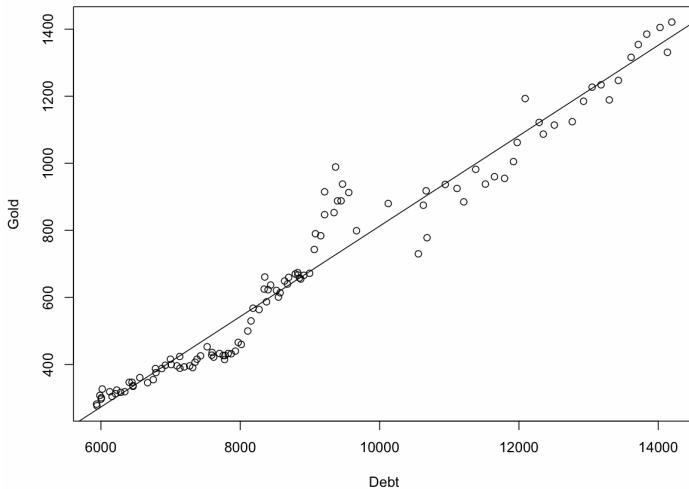
Hancel González y Kelwin Fernández

- Librería de aprendizaje de máquina implementada en Haskell.
- Que aproveche el paralelismo natural de estos algoritmos.
- Algoritmos y Máquinas implementados

- Librería de aprendizaje de máquina implementada en Haskell.
- Que aproveche el paralelismo natural de estos algoritmos.
- Algoritmos y Máquinas implementados. . . por ahora

- Librería de aprendizaje de máquina implementada en Haskell.
- Que aproveche el paralelismo natural de estos algoritmos.
- Algoritmos y Máquinas implementados. . . por ahora
 - Regresión Lineal.
 - Regresión Logística.
 - Máquina de Soporte de Vectores.
 - Redes Neuronales y Backpropagation.
 - K-Means

Regresión Lineal

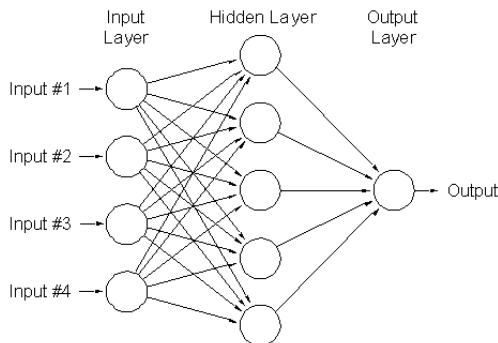


Regresión Lineal - Aspectos de Interés

- Uso de la combinación de los Monads Reader, Writer y State para configuración de los experimentos, estadísticas de progreso de aprendizaje y la máquina mutable.
- Paralelismo mediante `Control.Parallel`.
- Uso de la librería `hmatrix` de álgebra lineal.
- Uso de la librería adicional `gnuplot` para la graficación del progreso de aprendizaje.
- Algoritmos de entrenamiento mediante *gradient descent* y *normal equation*.

Regresión Lineal - Ejemplo de Uso

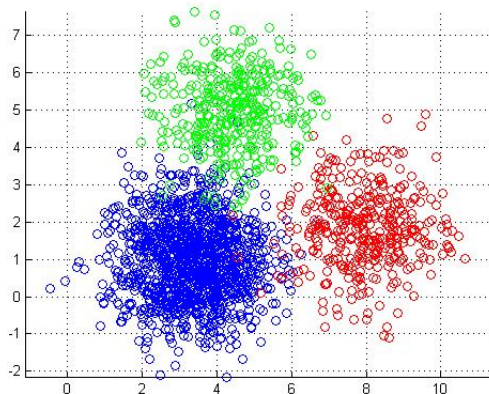
- Combinador RWS análogo a los anteriores.
- Paralelismo mediante anotaciones.
- Algoritmo de Aprendizaje: *Stochastic Backpropagation*.



Redes Neuronales - Ejemplo de Uso

K-Means

- Algoritmo de aprendizaje no supervisado.
- Agrupación de datos identificando patrones.



K-Means - Algoritmo

- 1 Crear K centroides aleatorios.
- 2 Asignar a cada punto su centroide más cercano.
- 3 Mover cada centroide a su centro de masa.
- 4 Repetir los pasos 2 y 3 hasta converger.

Paralelización de los pasos 2 y 3, recalculando cada centroide partiendo los datos de entrada en grupos y uniendo finalmente.

K-Means - Ejemplo de Uso

- 1 Comprimir imágenes.
- 2 Agrupar los píxeles en k grupos, asignar a cada píxel el color que mejor los represente a todos (el centroide, dah!).
- 3 Necesario $\log(k)$ bits para representar cada píxel.
- 4 Para 16 colores, se reduce el almacenamiento de 24 bits por píxel (RGB) a 4 bits.

Problema: no se ha encontrado una librería buena para procesamiento de las imágenes en 6.12. Actualmente se realiza con *OpenCV* en C++.

K-Means - Ejemplo de Uso (Imagen Original)



K-Means - Ejemplo de Uso (10 iteraciones después)

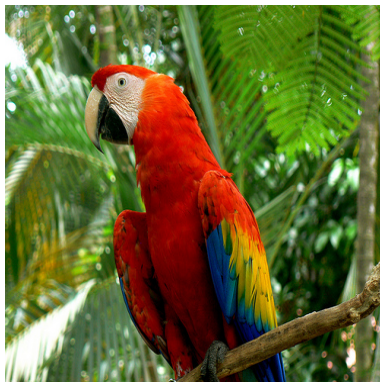


K-Means - Ejemplo de Uso (50 iteraciones después)



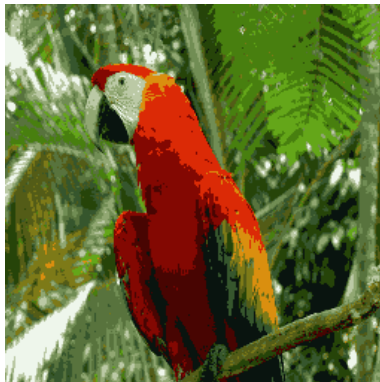
N2 (32 seg), N1 (44 seg)

K-Means - Ejemplo de Uso



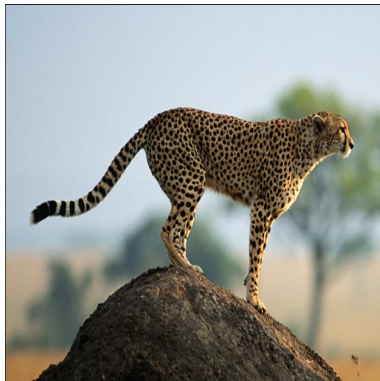
Antes

K-Means - Ejemplo de Uso



Después

K-Means - Ejemplo de Uso



Antes

K-Means - Ejemplo de Uso



Después

K-Means - Ejemplo de Uso



Antes

K-Means - Ejemplo de Uso



Después

K-Means - Ejemplo de Uso



Antes



Después