



Reconocimiento de Patrones  
Departamento de Ciencia de la Computación  
Universidad Católica de Chile  
Prof. Domingo Mery  
<http://dmery.ing.puc.cl>

---

## Ejercicio: Selección de Características con SFS

---

Objetivo: Aprender a usar el algoritmo de selección de características SFS (Sequential Forward Selection). Los datos están en <https://github.com/domingomery/patrones> > clases > Cap03 > data

Este ejercicio tiene dos partes: 1) la primera es probar el algoritmo de clasificación KNN con  $K=1$  en un conjunto de datos en los que no se ha hecho selección de características. 2) la segunda parte es probar el mismo algoritmo pero escogiendo las 15 mejores características del mismo conjunto de datos de la primera parte. Los datos ya están normalizados.

### PARTE 1: SIN SELECCIÓN DE CARACTERÍSTICAS

- Cargar los datos del archivo DATOS1.mat. En estos datos existen las variables
  - Xtrain (matriz de 345 muestras con 299 características)
  - Ytrain (vector que contiene '1' o '0' para las 345 muestras de Xtrain)
  - Xtest (matriz de 86 muestras con 299 características)
  - Ytest (vector que contiene '1' o '0' para las 86 muestras de Xtest)
- Computar el vector Ypred como la predicción de la clase en los datos de testing usando el clasificador KNN con  $K=1$ . Para esto use los datos de entrenamiento (Xtrain, Ytrain) y los datos testing (solo Xtest).
  - PARA MATLAB: estudiar help del comando Bcl\_knn donde dice 'Training & Test together' (en caso que no funcione es porque no se ha instalado el toolbox Vlfat, en este caso usar comando Bcl\_knn\_old).
  - PARA PYTHON: estudiar KNeighborsClassifier de librería sklearn, un ejemplo de su uso está en el ejemplo visto en clase (ver código sfs.py)
- Computar el accuracy de la predicción ( $N^\circ$  muestras correctamente clasificadas /  $86 \times 100$ ). Para este computo se debe comparar Ytest con Ypred

Accuracy = \_\_\_\_\_ %

### PARTE 2: CON SELECCIÓN DE CARACTERÍSTICAS

- Cargar datos nuevos que están en el archivo DATOS2.mat, con las variables
  - X (matriz de 431 muestras con 299 características)
  - Y (vector que contiene '1' o '0' para las 435 muestras de X)
- Usando el algoritmo SFS seleccione de las 299 características, las 15 características más importantes
  - PARA MATLAB: estudiar help del comando Bfs\_sfs donde dice 'Example 1'.
  - PARA PYTHON: estudiar sfs de librería pybalu, un ejemplo de su uso está en el ejemplo visto en clase sfs.py
- Usando las variables Xtrain y Xtest de la Parte 1, defina las variables Xtrain\_new y Xtest\_new como las 15 columnas seleccionadas de Xtrain y Xtest. Al final de este paso, Xtrain\_new debe tener 345 filas y 15 columnas, y Xtest\_new 86 filas y 15 columnas.
- Computar Ypred\_new de la misma manera como se computó Ypred en la Parte 1, pero usando Xtrain\_new, Ytrain y Xtest\_new.
- Computar el accuracy comparando Ypred\_new con Ytest.

Accuracy = \_\_\_\_\_ %

---

## Ejercicio: Extracción de Características Geométricas con Balu

---

Objetivo: Diseñar un clasificador automático de flechas, que puede ser usado para identificación de señales de tránsito. Los datos están en <https://github.com/domingomery/patrones> > clases > Cap02 > images > arrows

En la carpeta arrows\_training y arrows\_testing se encuentran imágenes en formato PNG con tres clases de flechas:



1) unidireccional-curva



2) bidireccional



3) unidireccional-recta.

Usando características geométricas (momentos de Hu, momentos de Flusser y descriptores) y la técnica de selección de características SFS, diseñe un clasificador KNN (con  $K=1$ ) que clasifique correctamente las flechas de la carpeta arrows\_testing.