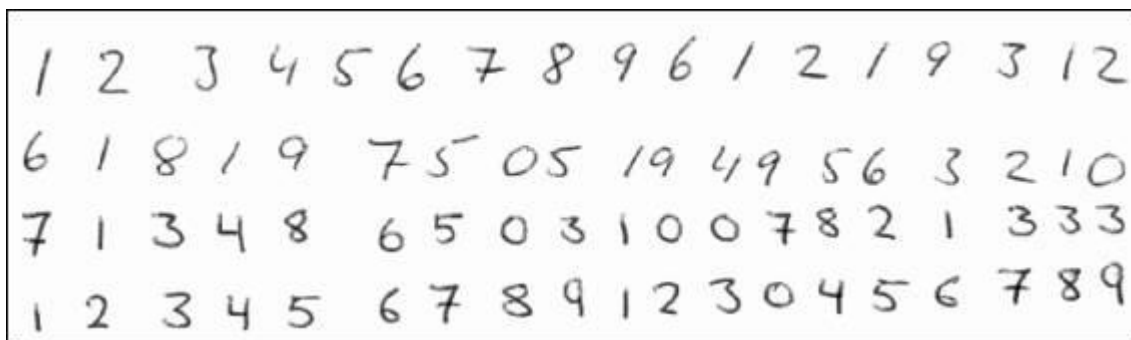




## Pràctica 2.

### Reconeixement automàtic de dígit manuscris



#### Objectiu:

L'objectiu de la pràctica és reconèixer els dígit presents en una sèrie d'imatges.

Es facilita un dataset d'imatges de dígit manuscrits, juntament amb les seves etiquetes. El dataset és enorme (60.000 imatges). Podeu treballar amb un subconjunt d'aquestes imatges per estalviar-vos problemes de memòria i capacitat de càlcul.

#### Etales:

1. Per a cada imatge, obtenir les característiques que us semblin adients per a descriure el dígit.
2. Seleccionar un subconjunt d'imatges que serviran per a fer l'aprenentatge (un 80% del total). A aquestes imatges les anomenarem '**imatges de mostra**', a la resta les anomenarem '**imatges de test**'.
3. Entrenar un classificador amb les imatges de mostra.
4. Classificar cada imatge de test en la classe més similar en funció del seu vector de característiques, i identificar-la amb el nombre corresponent.

Es demana:

1. Programa MATLAB que entreni i validi el sistema de reconeixement proposat. Sempre és recomenable repetir les etapes 2,3 i 4 usant diferents *folds* i promitjar els resultats (*cross-validation*) .
2. Programa MATLAB que implementi el reconeixement automàtic de dígit manuscrits:
  - Entrada: imatge que volem processar
  - Sortida: valor numèric del dígit present en la imatge
3. Documentació. Cal que lliureu un informe contenint:
  - Descripció de les característiques utilitzades per a modelitzar els dígit.
  - Descripció dels classificadors utilitzats
  - Descripció dels experiments realitzats
  - Resultats obtinguts. No oblideu la matriu de confusió.
  - Descripció de les funcions utilitzades, deixant ben clar quines són les implementades per vosaltres. Tot el software provinent d'altres fonts, ha d'estar correctament referenciat i comentat, EN CAS CONTRARI ES CONSIDERARÀ PLAGI.
  - L'última secció de l'informe serà un annex amb tot el codi

Es valorarà :

- Justificació de l'elecció dels descriptors i els classificadors
- Qualitat dels descriptors utilitzats per a modelitzar els dígit
- Resultats obtinguts
- Rigor en l'experimentació. Presentació formal de resultats, usant els indicadors estàndar d'avaluació de classificació
- Un anàlisi de significància (i no-redundància) de les característiques emprades.

Notes importants:

- Podeu utilitzar les funcions de la toolbox 'image processing' de MATLAB i altres funcions contribuïdes, però s'exigirà que tingueu un coneixement propi del que fan aquestes funcions i del que volen dir els seus paràmetres. Qualsevol contribució d'altri ha d'estar correctament referenciada i documentada en la memòria.
- Actualment, no existeix cap solució capaç d'assolir un rati del 100% en la classificació de dígit manuscrits. Òbviament no cal assolir un rati d'error del 0% per a obtenir la qualificació de 10.

### Informació sobre el dataset:

( *The MNIST database of handwritten digits*. Yann LeCun, Corinna Cortes, Christopher J.C. Burges)

The original black and white (bilevel) images were size normalized to fit in a 20x20 pixel box while preserving their aspect ratio. The resulting images contain grey levels as a result of the anti-aliasing technique used by the normalization algorithm. the images were centered in a 28x28 image by computing the center of mass of the pixels, and translating the image so as to position this point at the center of the 28x28 field.

### **TRAINING SET LABEL FILE (train-labels-idx1-ubyte):**

[offset]	[type]	[value]	[description]
0000	32 bit integer	2049	magic number (MSB first)
0004	32 bit integer	60000	number of items
0008	unsigned byte	??	label
0009	unsigned byte	??	label
.....			
xxxx	unsigned byte	??	label

The labels values are 0 to 9.

### **TRAINING SET IMAGE FILE (train-images-idx3-ubyte):**

[offset]	[type]	[value]	[description]
0000	32 bit integer	2051	magic number
0004	32 bit integer	60000	number of images
0008	32 bit integer	28	number of rows
0012	32 bit integer	28	number of columns
0016	unsigned byte	??	pixel
0017	unsigned byte	??	pixel
.....			
xxxx	unsigned byte	??	pixel

Pixels are organized row-wise. Pixel values are 0 to 255.

Per llegir les imatges, podeu utilitzar la funció *readMNIST.m* que trobareu junt amb les imatges (i la corresponent llicència a *license.txt*). La funció retorna imatges de 20x20 píxels amb nivells de gris entre 0 i 1.