



# Escuela superior de cómputo Instituto politécnico nacional

## Practica 01

Materia: Reconocimiento de voz

Profesor: Carmona García Enrique Alfonso

Alumno: Sánchez Mederos Yael Alexandr

Grupo: 7BM1

## Modificaciones

En el primer programa *02\_regresion\_1.py*

Se guardaron las matrices en un archivo data frame y se almacena en un archivo csv 'datos'

```
# Crear el dataframe con las dos matrices
df = pd.DataFrame(data=X, columns=['col'+str(i+1) for i in range(10)])
df['label'] = y

# Guardar el dataframe en un archivo csv
df.to_csv('datos.csv', index=False)
```

Datos guardados en el archivo csv

col1	col2	col3	col4	col5	col6	col7	col8	col9	col10	label
0.61473631	0.02420839	0.99641856	0.43835753	0.83754877	0.69576271	0.28927897	0.01539918	0.49934924	0.18841906	3
0.28656859	0.95358146	0.32505432	0.74774137	0.00416492	0.97217468	0.72581339	0.61000445	0.57815791	0.33648233	1
0.3866371	0.78744794	0.10231964	0.01941954	0.82199612	0.56215958	0.58922733	0.20686702	0.36077201	0.96741988	2
0.35911709	0.48374377	0.2633531	0.78142089	0.44471711	0.1746881	0.9413813	0.03579237	0.24914558	0.52935697	2
0.91751593	0.23261426	0.488386	0.68758187	0.84636499	0.03373385	0.44929238	0.24284455	0.4241412	0.10831963	3
0.04123459	0.71957014	0.54807294	0.85379857	0.50116476	0.57356733	0.63700901	0.28542518	0.98932164	0.06684131	4
0.18379524	0.32716234	0.73981776	0.20750822	0.89151004	0.32335538	0.98144644	0.5905578	0.10926843	0.92646901	1
0.28858274	0.58056552	0.555635	0.40239624	0.76253606	0.0761625	0.52353998	0.83989159	0.30258426	0.20183	0
0.69894841	0.84835154	0.87008299	0.34745894	0.49472628	0.71817824	0.28565323	0.05486778	0.67250209	0.02393493	4
0.74154545	0.64076825	0.33917101	0.1284635	0.33792	0.73454688	0.86269867	0.53598186	0.1292777	0.94832618	1
0.22564248	0.19905237	0.94179281	0.11008568	0.06607336	0.38528548	0.40502169	0.34699624	0.0229588	0.75175991	2
0.30134909	0.45904852	0.79208399	0.16682646	0.24016152	0.64035856	0.38889343	0.80474761	0.72257913	0.72115515	3
0.36856086	0.15280934	0.07525061	0.70019876	0.85278623	0.18848283	0.17731034	0.75385758	0.97017107	0.24263969	3
0.55935756	0.8693494	0.83943013	0.03637574	0.51377507	0.69019768	0.92892208	0.23031137	0.48937211	0.30840397	1
0.62833339	0.89756656	0.69733284	0.5324026	0.01425676	0.09678065	0.48687679	0.63544422	0.39520645	0.6403272	2
0.9073498	0.58862876	0.35021158	0.86762608	0.43083355	0.51459886	0.66589883	0.71989937	0.96973347	0.84306527	2
0.65236554	0.6740218	0.88937114	0.27791164	0.54468263	0.9303274	0.58771487	0.70080364	0.79794945	0.67443208	0
0.77507596	0.54074821	0.16586088	0.65222458	0.28596978	0.08688528	0.1603471	0.40133003	0.45284664	0.56255131	0
0.52206687	0.25544363	0.7350044	0.91928239	0.37047988	0.93909413	0.6726992	0.96582691	0.65082454	0.58535907	3
0.31675546	0.54182463	0.92267493	0.90646388	0.33628622	0.04375908	0.49608659	0.40876172	0.58640216	0.65807944	0

Resultados obtenidos:

```
0.49608659 0.40876172 0.58640216 0.65807944]]
Distribución de vector de etiquetas de clases: [3 1 2 2 3 4 1 0 4 1 2 3 3 1 2 2 0 0 3 0]
y: [3 1 2 2 3 4 1 0 4 1 2 3 3 1 2 2 0 0 3 0]
Señal generada: [[0.35845959]
 [0.86123884]
 [0.14197788]
 [0.95822978]
 [0.3436164 ]
 [0.30689892]
 [0.26623579]
 [0.28086492]
 [0.02667643]
 [0.30645904]]
Etiqueta de salida predicha: 2
```

En el segundo programa 02\_knn\_1.py

Para cargar los datos del archivo csv se usó pandas

En el arreglo 'X' se usaron todos los datos excepto la ultima columna y la ultima columna se guardo en la variable 'y'

```
df = pd.read_csv('datos.csv')

# Convertir los datos de X a un arreglo de tipo float
X = np.array([list(map(float, row)) for row in df.iloc[:, :-1].values])

# Cargar los datos de y en un arreglo separado
y = np.array(df.iloc[:, -1].values)
```

Resultados obtenidos:

```
[[0 0 0 1]
 [0 0 0 1]
 [1 0 0 0]
 [1 0 0 0]]
```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	1.0
1	0.00	0.00	0.00	1.0
2	0.00	0.00	0.00	1.0
3	0.00	0.00	0.00	1.0
accuracy			0.00	4.0
macro avg	0.00	0.00	0.00	4.0
weighted avg	0.00	0.00	0.00	4.0