Entrega Final de Prácticas APR

Francisco Fernández García
frafe18f@inf.upv.es

Alejandro Furió Agustí
juafuag@inf.upv.es

Enero de 2022

Contents

1 Mixturas Gaussianas

1.1 Ejercicio 1

Los valores de suavizado utilizados para obtener el mejor resultado posible en el clasificador gaussiano han sido $\alpha = 1 * 10^{-2}, 1 * 10^{-3}, 1 * 10^{-4}, *10^{-5}$.
Tras hacer las correspondientes iteraciones, hemos obtenido los siguientes resultados:

Las distintas curvas en D representan los errores en función de K que toma los valores: 1, 2, 5, 10, 20, 50, 100.

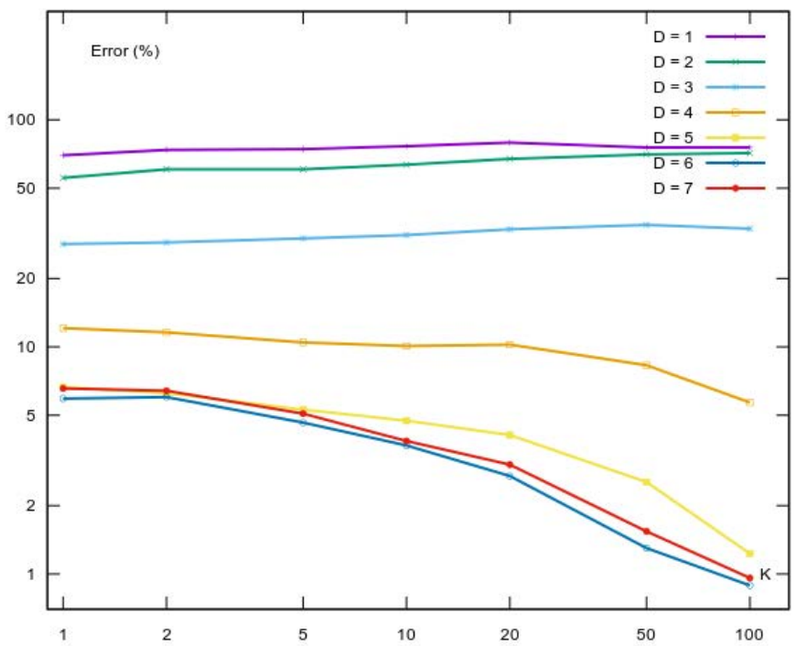


Figure 1: $\alpha = 1 * 10^{-}2$

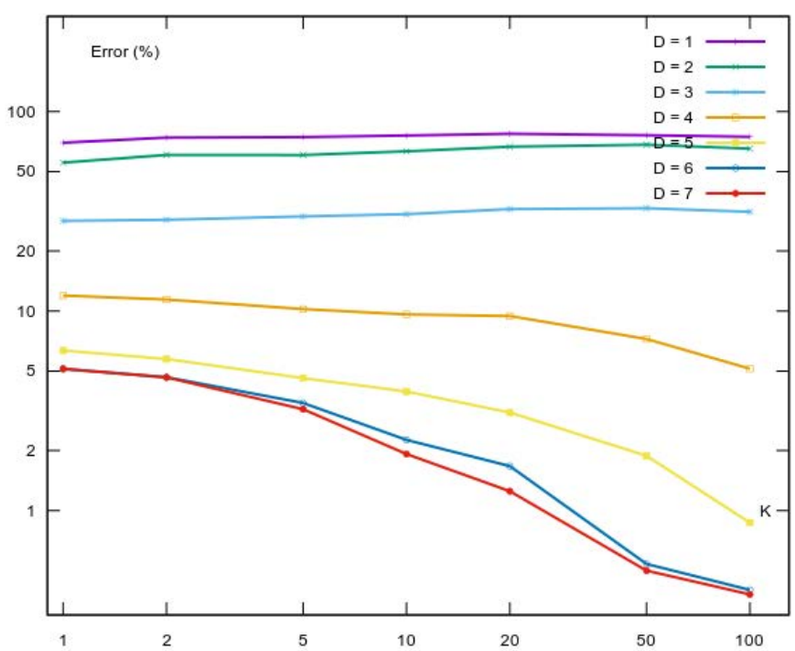


Figure 2: $\alpha = 1 * 10^{-3}$

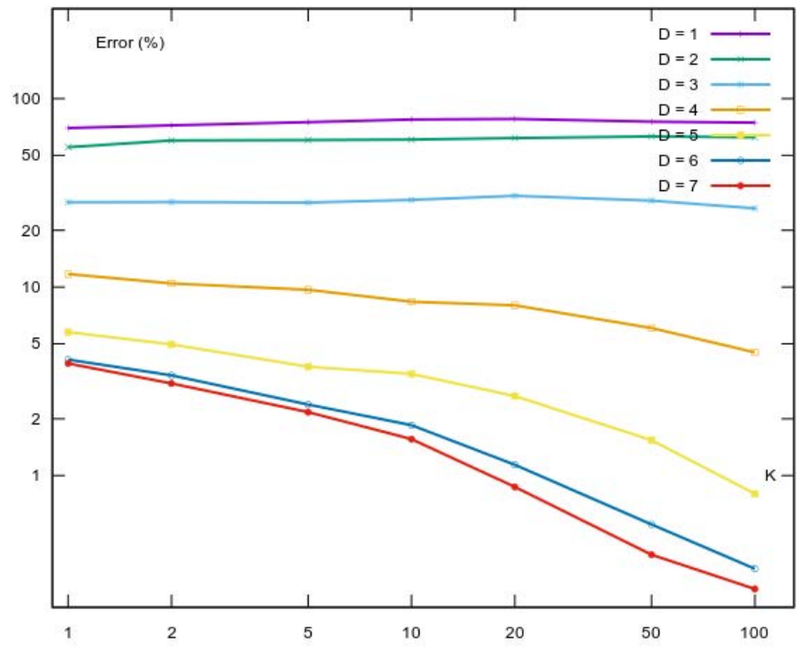


Figure 3: $\alpha = 1 * 10^{-4}$

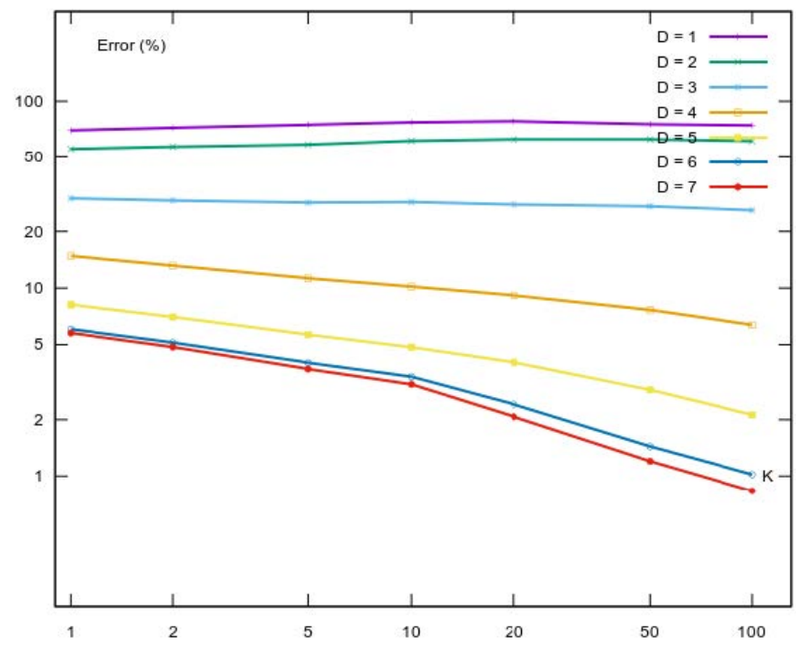


Figure 4: $\alpha = 1 * 10^{-5}$

A través de las gráficas anteriormente expuestas se puede observar que el mejor resultado se obtiene para $\alpha = 1 * 10^{-4}$, con $K = 100$ y $D = 100$, por lo que los consideraremos como los parámetros óptimos para el clasificador por mixtura de gaussianas.

- $\alpha = 1 * 10^{-4}$
- $K = 100$
- $D = 100$

1.2 Ejercicio 2

A la vista de los resultados obtenidos en el Ejercicio 1, ejecutamos el script $mixgaussian - eva.m$ con el conjunto de entrenamiento $train - images$, conjunto de evaluación $t10k - images$, $\alpha = 1 * 10^{-4}$, con $K = 100$ y $D = 100$:

```
./mixgaussian-eva.m train-images-idx3-ubyte.mat.gz
train-labels-idx1-ubyte.mat.gz t10k-images-idx3-ubyte.mat.gz
t10k-labels-idx1-ubyte.mat.gz 1e-4 100 100
```

Se obtiene un error en el conjunto de evaluación del **1'67%**

1.3 Ejercicio 3

Se ha optado por implementar la ampliación de finalización temprana en la función mixgaussian, para lo cual se ha añadido una variable que representa el error en la iteración anterior inicializada a 100:

```
lastee = 100
```

Además, se han añadido las siguientes líneas al finalizar el cálculo de los errores en cada iteración.

```
if (tee >= lastee)
    tee = lasttee;
    break;
endif

lasttee = tee
```

2 SVM

2.1 Ejercicio 1

Tarea mini: dos conjuntos de datos de entrenamiento 2D

- trSep.dat, trSeplabels.dat: linealmente separable (sin kernel)
- tr.dat, trlabels.dat: no linealmente separable Para cada uno de los dos conjuntos de la tarea mini:

Para cada uno de los dos conjuntos de la tarea mini:

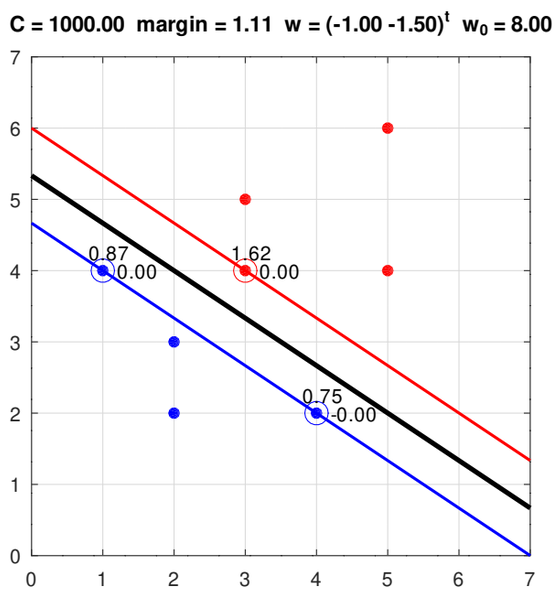
- Obtén los SVM sin kernel con C grande ($C = 1000$).
- Halla los multiplicadores; vectores soporte; vector de pesos y umbral de la discriminante lineal; y margen correspondiente.
- Calcula los parametros de la frontera lineal de separación.

Para el conjunto no-separable, prueba valores relevantes de C y:

- Determina los valores de tolerancia de margen, ζ.
- Marca los vectores soporte "erróneos" en la gráfica.

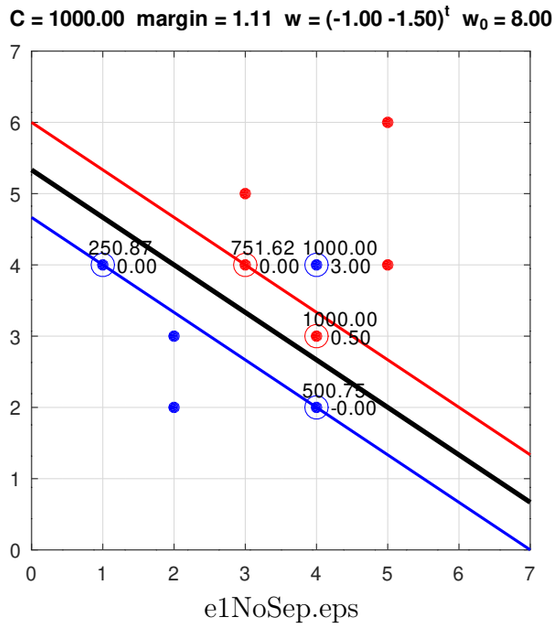
2.1.1 Caso Separable

Multiplicadores de Lagrange = [0.8747, 0.7499, -1.6246]
Vectores soporte:
$(1,1) \rightarrow 1$ $(2,1) \rightarrow 4$ $(3,1) \rightarrow 3$ $(2,2) \rightarrow 2$ $(3,2) \rightarrow 4$
Vector de pesos = [-0.9995, -1.4998]



2.1.2 Caso no Separable

Multiplicadores: [250.87, 500.75, 1000.00, -751.62, -1000.00]
Vectores Soporte:
$(1,1) \rightarrow 1$ $(2,1) \rightarrow 4$ $(3,1) \rightarrow 4$ $(4,1) \rightarrow 3$ $(5,1) \rightarrow 4$ $(1,2) \rightarrow 4$ $(2,2) \rightarrow 2$ $(3,2) \rightarrow 4$ $(5,2) \rightarrow 3$
Vector de pesos: -0.9995 -1.4998



e1NoSep.eps

2.2 Ejercicio 2

Estima el error de las SVMs en MNIST, en funcion de C (-c 1, 10, 100) y el kernel (-t 0, 1, 2, 3), con sus parametros específicos:

- Polinómico (-t 1): grado del polinomio (-d 1, 2, 3, 4, 5, ...)
- Gaussiano (-t 2): gamma (-g 1e-1, 1e-2, 1e-3, 1e-4, 1e-5, ...)
- Sigmoide (-t 3): gamma (-g 1e-1, 1e-2, 1e-3, 1e-4, 1e-5, ...)

Recomendaciones:

- Normaliza los datos en [0, 1] (dividiendo por 255).
- Como en mixturas, haz un script que explore los parametros indicados arriba para unos pocos valores de $PCA(D = 50, 100, 200)$.
- Por eficiencia, utiliza una particion entrenamiento-validación reducida; por ejemplo, $90\% - 10\%$ con 6000 muestras en total.

Describe brevemente los resultados obtenidos con base en una representación adecuada de los mismos, gráfica o tabular.

Tipo de Kernel	Dimensión PCA	Coste	Gamma	Precisión
0	50	1	0e+00	90.83
0	50	10	0e+00	90.67
0	50	100	0e+00	89.83
0	100	1	0e+00	91.17
0	100	10	0e+00	90.33
0	100	100	0e+00	90.17
0	200	1	0e+00	91.00
0	200	10	0e+00	90.67
0	200	100	0e+00	90.67

Table 1: Kernel Lineal

Tipo de Kernel	Dimensión PCA	Coste	Grado	Precisión
1	50	1	1	92.67
1	50	1	2	95.50
1	50	1	3	95.17
1	50	1	4	93.17
1	50	1	5	91.50
1	50	10	1	91.67
1	50	10	2	95.00
1	50	10	3	94.50
1	50	10	4	93.17
1	50	10	5	91.67
1	50	100	1	90.33
1	50	100	2	95.00
1	50	100	3	93.50
1	50	100	4	92.00
1	50	100	5	91.17
1	100	1	1	92.50
1	100	1	2	93.67
1	100	1	3	91.50
1	100	1	4	86.83
1	100	1	5	78.50
1	100	10	1	94.00
1	100	10	2	95.83
1	100	10	3	94.67
1	100	10	4	92.33
1	100	10	5	90.00
1	100	100	1	91.17
1	100	100	2	94.83
1	100	100	3	93.50
1	100	100	4	92.83
1	100	100	5	90.50
1	200	1	1	92.00
1	200	1	2	91.33
1	200	1	3	81.33
1	200	1	4	61.33
1	200	1	5	38.50
1	200	10	1	93.00
1	200	10	2	95.50
1	200	10	3	92.17
1	200	10	4	83.67
1	200	10	5	69.33
1	200	100	1	92.33
1	200	100	2	95.33
1	200	100	3	94.67
1	200	100	4	92.00
1	200	100	5	84.83

Table 2: Kernel Polinomial

Tipo de Kernel	Dimensión PCA	Coste	Gamma	Precisión
2	50	1	1e-01	94.17
2	50	1	1e-02	95.17
2	50	1	1e-03	90.17
2	50	1	1e-04	75.67
2	50	1	1e-05	11.67
2	50	10	1e-01	94.17
2	50	10	1e-02	96.50
2	50	10	1e-03	93.17
2	50	10	1e-04	90.17
2	50	10	1e-05	76.00
2	50	100	1e-01	94.17
2	50	100	1e-02	96.00
2	50	100	1e-03	93.00
2	50	100	1e-04	92.83
2	50	100	1e-05	90.17
2	100	1	1e-01	89.83
2	100	1	1e-02	94.67
2	100	1	1e-03	90.17
2	100	1	1e-04	76.17
2	100	1	1e-05	11.67
2	100	10	1e-01	90.17
2	100	10	1e-02	96.00
2	100	10	1e-03	93.50
2	100	10	1e-04	89.67
2	100	10	1e-05	76.17
2	100	100	1e-01	90.17
2	100	100	1e-02	96.17
2	100	100	1e-03	94.67
2	100	100	1e-04	93.17
2	100	100	1e-05	89.67
2	200	1	1e-01	87.00
2	200	1	1e-02	94.67
2	200	1	1e-03	90.50
2	200	1	1e-04	76.17
2	200	1	1e-05	11.67
2	200	10	1e-01	87.17
2	200	10	1e-02	96.00
2	200	10	1e-03	93.33
2	200	10	1e-04	90.50
2	200	10	1e-05	76.50
2	200	100	1e-01	87.17
2	200	100	1e-02	96.00
2	200	100	1e-03	94.17
2	200	100	1e-04	92.17
2	200	100	1e-05	90.50

Table 3: Kernel Radial

Tipo de Kernel	Dimensión PCA	Coste	Gamma	Precisión
3	50	1	1e-01	18.00
3	50	1	1e-02	88.83
3	50	1	1e-03	87.83
3	50	1	1e-04	61.17
3	50	1	1e-05	11.67
3	50	10	1e-01	16.00
3	50	10	1e-02	83.00
3	50	10	1e-03	92.00
3	50	10	1e-04	87.83
3	50	10	1e-05	61.17
3	50	100	1e-01	16.33
3	50	100	1e-02	79.00
3	50	100	1e-03	92.33
3	50	100	1e-04	92.00
3	50	100	1e-05	87.83
3	100	1	1e-01	16.33
3	100	1	1e-02	88.50
3	100	1	1e-03	88.67
3	100	1	1e-04	61.83
3	100	1	1e-05	11.67
3	100	10	1e-01	17.50
3	100	10	1e-02	83.67
3	100	10	1e-03	92.50
3	100	10	1e-04	88.67
3	100	10	1e-05	61.83
3	100	100	1e-01	17.50
3	100	100	1e-02	80.17
3	100	100	1e-03	93.67
3	100	100	1e-04	92.50
3	100	100	1e-05	88.67
3	200	1	1e-01	16.50
3	200	1	1e-02	88.50
3	200	1	1e-03	88.83
3	200	1	1e-04	61.67
3	200	1	1e-05	11.67
3	200	10	1e-01	17.67
3	200	10	1e-02	85.17
3	200	10	1e-03	91.50
3	200	10	1e-04	88.83
3	200	10	1e-05	61.67
3	200	100	1e-01	18.17
3	200	100	1e-02	80.67
3	200	100	1e-03	92.67
3	200	100	1e-04	91.50
3	200	100	1e-05	88.83

Table 4: Kernel Sigmoide

2.3 Ejercicio 3

Kernel 0

Valores óptimos: D = 100, C = 1
Intervalo: [90.16, 91.30]

Kernel 1

Valores óptimos: D = 100, C = 10, d = 2
Intervalo: [94.90-5.72]

Kernel 2

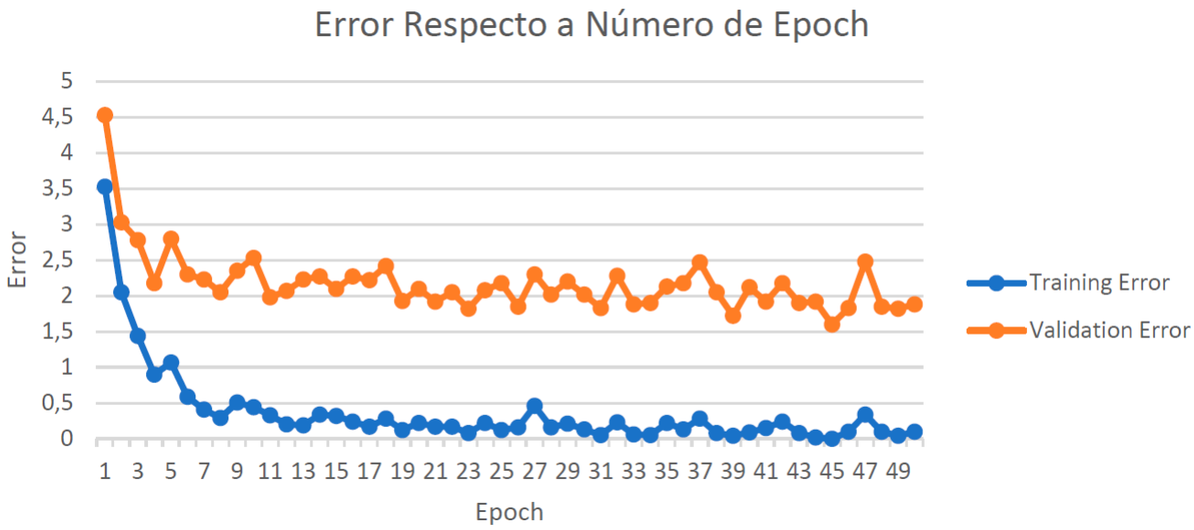
Valores óptimos: D = 50, C = 100, g = 1e-02
Intervalo: [95.46, 96.24]

Kernel 3

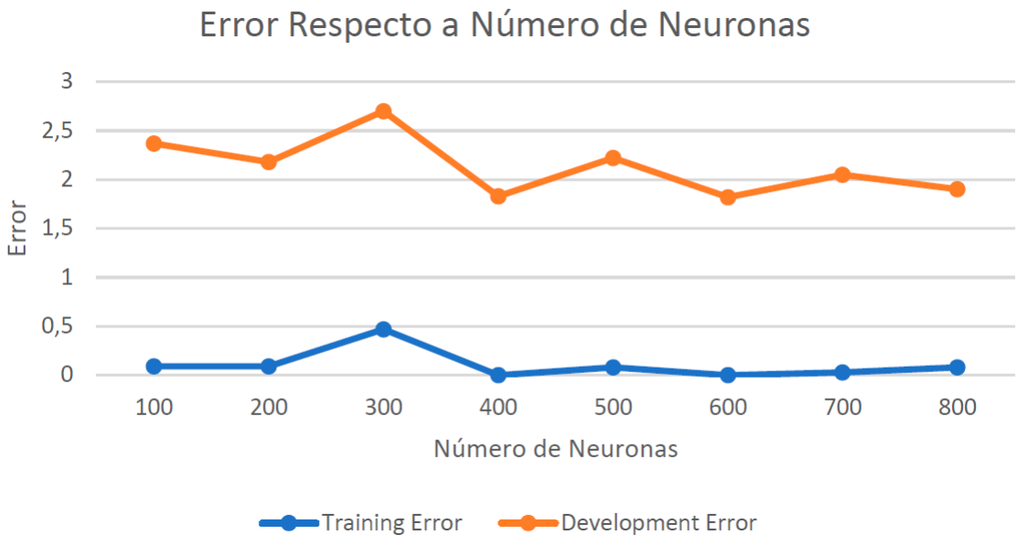
Valores óptimos: D = 100, C = 100, g = 1e-03
Intervalo: [91.82 - 92.86]

3 Redes Neuronales

3.1 Ejercicio 1



Se puede apreciar que el error converger en torno a los 20 epochs, por lo que para el resto de la práctica se usará esta confuguración.



3.2 Ejercicio 2

Epoch	Training Error	Validation Error
1	3.53%	4.53%
2	2.05%	3.03%
3	1.44%	2.78%
4	0.90%	2.18%
5	1.07%	2.80%
6	0.59%	2.30%
7	0.41%	2.23%
8	0.29%	2.05%
9	0.51%	2.35%
10	0.44%	2.53%
11	0.33%	1.98%
12	0.20%	2.07%
13	0.19%	2.23%
14	0.34%	2.27%
15	0.32%	2.10%
16	0.24%	2.27%
17	0.17%	2.22%
18	0.28%	2.42%
19	0.12%	1.93%
20	0.22%	2.10%
21	0.17%	1.92%
22	0.17%	2.05%
23	0.08%	1.82%
24	0.22%	2.08%
25	0.12%	2.18%
26	0.16%	1.85%
27	0.46%	2.30%
28	0.16%	2.02%
29	0.21%	2.20%
30	0.13%	2.02%
31	0.05%	1.83%
32	0.23%	2.28%
33	0.06%	1.88%
34	0.05%	1.90%
35	0.22%	2.13%
36	0.13%	2.18%
37	0.28%	2.47%
38	0.08%	2.05%
39	0.04%	1.72%
40	0.09%	2.12%
41	0.15%	1.92%
42	0.24%	2.18%
43	0.08%	1.90%
44	0.02%	1.92%
45	0.00%	1.60%
46	0.10%	1.83%
47	0.34%	2.48%
48	0.10%	1.85%
49	0.04%	1.82%
50	0.10%	1.88%

Table 5: Tasas de Error respecto a número de Epoch.

Aquí observamos un valor óptimo de 45 Epochs para obetener el mínimo error.

Neurons	Training Error	Development Error
100	0.09%	2.37%
200	0.09%	2.18%
300	0.47%	2.70%
400	0.00%	1.83%
500	0.08%	2.22%
600	0.00%	1.82%
700	0.03%	2.05%
800	0.08%	1.90%

Table 6: Tasas de Error respecto a número de Neuronas con optimizador Adam, función de activación ReLU y 1 capa oculta.

600 neuronas en la capa oculta nos proporcionan el mínimo error.

Neurons l1	Neurons l2	Training Error	Develpoment Error
300	300	0.37%	2.33%
300	400	0.42%	2.18%
300	500	0.22%	2.08%
300	600	0.16%	1.98%
400	300	0.27%	2.25%
400	400	0.20%	1.85%
400	500	0.15%	1.93%
400	600	0.34%	2.23%
500	300	0.31%	2.28%
500	400	0.18%	2.02%
500	500	0.25%	2.07%
500	600	0.17%	1.75%
600	300	0.32%	2.00%
600	400	0.14%	1.85%
600	500	0.39%	2.20%
600	600	0.21%	2.03%

Table 7: Tasas de Error respecto a número de Neuronas con optimizador Adam, función de activación ReLU y 2 capas ocultas.

El clasificador con dos capas ocultas que tiene menor "Training Error" es el de 600 neuronas en L1 y 400 en L2. El que tiene menor "Development Error" es el de 500 neuronas en L1 y 600 en L2.

Neurons l1	Neurons l2	Neurons l3	Training Error	Develpoment Error
300	300	300	0.35%	2.43%
300	300	400	0.35%	2.25%
300	300	500	0.22%	2.28%
300	400	300	0.28%	2.02%
300	400	400	0.29%	2.17%
300	400	500	0.27%	2.28%
300	500	300	0.62%	2.53%
300	500	400	0.24%	2.18%
300	500	500	0.23%	1.82%
400	300	300	0.47%	2.20%
400	300	400	0.16%	2.05%
400	300	500	0.18%	1.88%
400	400	300	0.13%	2.10%
400	400	400	0.21%	1.88%
400	400	500	0.38%	2.37%
400	500	300	0.18%	1.85%
400	500	400	0.11%	1.72%
400	500	500	0.26%	2.05%
500	300	300	0.19%	1.93%
500	300	400	0.11%	1.70%
500	300	500	0.08%	1.88%
500	400	300	0.25%	2.28%
500	400	400	0.39%	2.32%
500	400	500	0.34%	2.18%
500	500	300	0.12%	1.93%
500	500	400	0.07%	1.80%
500	500	500	0.12%	1.87%

Table 8: Tasas de Error respecto a número de Neuronas con optimizador Adam, función de activación ReLU y 3 capas ocultas.

El clasificador con tres capas ocultas que tiene menor "Training Error" es el de 500 neuronas en L1, 500 en L2 y 400 neuronas en L3. El que tiene menor "Development Error" es el de 500 neuronas en L1, 300 en L2 y 400 neuronas en L3.

3.3 Nota Aclaratoria

Por cuestiones de computabilidad, se ha limitado la experimentación de 1 a 3 capas con 600 neuronas como máximo.

El mejor clasificador es el de tres capas ocultas con 500 neuronas en L1, 300 en L2 y 400 neuronas en L3 dando un "Development Error" del 1.70%

4 Conclusiones Finales

De las alternativas con las que se ha experimentado, la que da mejor resultados es el clasificador de mixturas de gaussianas (con parámetros $\alpha = 1 * 10^{-4}$, $K = 100$, $D = 100$), aunque a la vista de los resultados de la web oficial este error se podría mejorar con redes neuronales más complejas (con más capas y neuronas)