

PRÁCTICA 5 – Clasificación Bayesiana

Fernando Aliaga Ramón - 610610

2. Entrenamiento y clasificación con modelos Gaussianos regularizados

Para implementar el método de entrenamiento de las gaussianas de cada clase hemos usado las funciones que nos ofrece matlab para calcular la media y la covarianza. Para realizar las matrices según Bayes Ingenuo hemos usado la función 'diag' para dejar a ceros los elementos que no se encuentran en la diagonal de la matriz.

```
buenos = (ytr==i);
modelo{i}.N = size(find(ytr==i),1);
modelo{i}.Sigma = cov(Xtr(buenos,:));
modelo{i}.mu = mean(Xtr(buenos,:));
modelo{i}.mu = modelo{i}.mu';
modelo{i}.Sigma = modelo{i}.Sigma + landa*eye(size(modelo{i}.Sigma));
if(NaiveBayes==1)
    modelo{i}.Sigma = diag(diag(modelo{i}.Sigma));
end
```

Para la clasificación bayesiana hacemos uso de la función facilitada 'gaussLog'. Como todas las clases tienen la misma probabilidad (0.1) la función 'gaussLog' nos dirá directamente cómo de bien se ajusta cada muestra a cada clase.

```
function yhat = clasificacionBayesiana(modelo, X)
% Con los modelos entrenados, predice la clase para cada muestra X

predicciones = [];
for(i=1:10)
    predicciones = [predicciones gaussLog(modelo{i}.mu, modelo{i}.Sigma, X)];
end

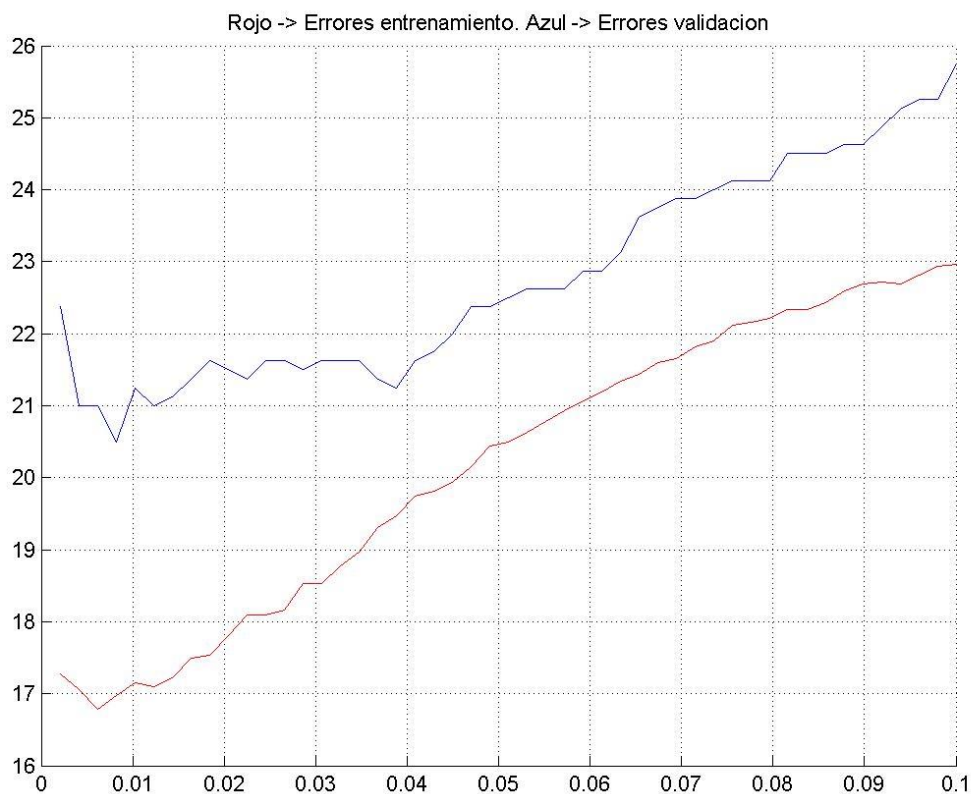
yhat=[];
for(i=1:size(X,1))
% Seleccionamos la prediccion que mas se ajusta de cada clase
    [maxval, maxindice] = max(predicciones(i,:));
    yhat = [yhat maxindice];
end

yhat = yhat';
```

3. Bayes ingenuo

Siguiendo el mismo esquema que en la práctica anterior, calculamos el mejor valor regularizador lambda mediante validación cruzada.

Para bayes ingenuo el mejor valor obtenido es 0.008163, arrojando una gráfica de errores como esta:



Reentrenamos ahora de nuevo con todos los datos y obtenemos las matrices de confusión para cada clase a la vez que la tasa de errores con los datos de test.

Error con datos de test = 19.300000

Matriz de confusion de 1

matriz_confusion =

98	29
2	871

Precision (1) = 0.771654

Recall (1) = 0.980000

Matriz de confusion de 2

matriz_confusion =

76	15
24	885

Precision (2) = 0.835165

Recall (2) = 0.760000

Matriz de confusion de 3

matriz_confusion =

72	16
28	884

Precision (3) = 0.818182

Recall (3) = 0.720000

Matriz de confusion de 4

matriz_confusion =

72	16
28	884

Precision (4) = 0.818182

Recall (4) = 0.720000

Matriz de confusion de 5

matriz_confusion =

59	9
41	891

Precision (5) = 0.867647

Recall (5) = 0.590000

Matriz de confusion de 6

matriz_confusion =

95	23
5	877

Precision (6) = 0.805085

Recall (6) = 0.950000

Matriz de confusion de 7

matriz_confusion =

```
84  4
16 896
```

Precision (7) = 0.954545

Recall (7) = 0.840000

Matriz de confusion de 8

matriz_confusion =

```
74  35
26 865
```

Precision (8) = 0.678899

Recall (8) = 0.740000

Matriz de confusion de 9

matriz_confusion =

```
88  40
12 860
```

Precision (9) = 0.687500

Recall (9) = 0.880000

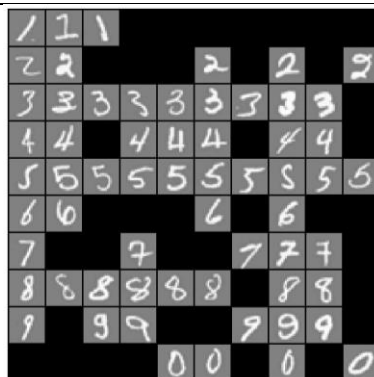
Matriz de confusion de 10

matriz_confusion =

```
89   6
11 894
```

Precision (10) = 0.936842

Recall (10) = 0.890000

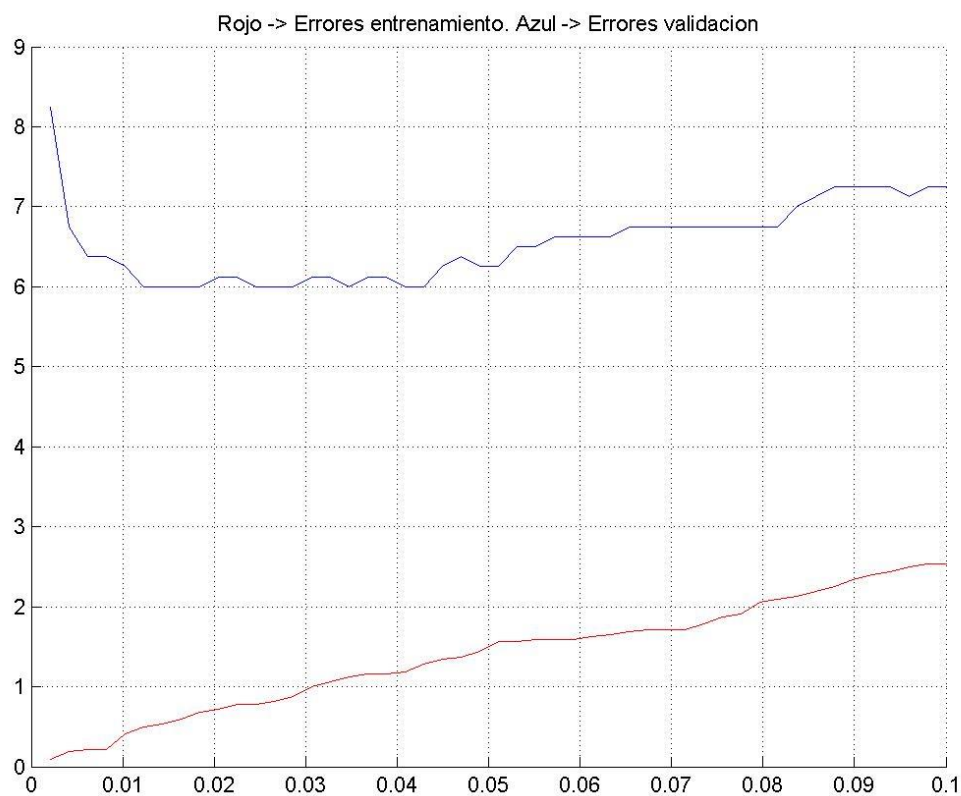


Los dígitos más problemáticos son el 5 (Recall = 0.590000) y el 8 (Precision = 0.678899).

3. Covarianzas completas

Repetimos el apartado anterior, ahora con matrices de covarianzas completas, lo cual arroja los siguientes resultados:

Mejor lambda : 0.012245



Una vez obtenida la mejor lambda, reentrenamos con todos los datos y calculamos las matrices de confusión y errores para cada clase.

Error con datos de test = 3.500000

Matriz de confusion de 1

matriz_confusion =

98	2
2	898

Precision (1) = 0.980000

Recall (1) = 0.980000

Matriz de confusion de 2

matriz_confusion =

99 7
1 893

Precision (2) = 0.933962
Recall (2) = 0.990000

Matriz de confusion de 3
matriz_confusion =

93 5
7 895

Precision (3) = 0.948980
Recall (3) = 0.930000

Matriz de confusion de 4
matriz_confusion =

97 1
3 899

Precision (4) = 0.989796
Recall (4) = 0.970000

Matriz de confusion de 5
matriz_confusion =

93 2
7 898

Precision (5) = 0.978947
Recall (5) = 0.930000

Matriz de confusion de 6
matriz_confusion =

99 1
1 899

Precision (6) = 0.990000
Recall (6) = 0.990000

Matriz de confusion de 7
matriz_confusion =

```
97 1
3 899
```

Precision (7) = 0.989796

Recall (7) = 0.970000

Matriz de confusion de 8
matriz_confusion =

```
94 10
6 890
```

Precision (8) = 0.903846

Recall (8) = 0.940000

Matriz de confusion de 9
matriz_confusion =

```
95 4
5 896
```

Precision (9) = 0.959596

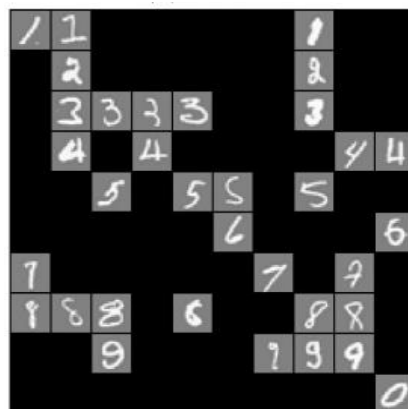
Recall (9) = 0.950000

Matriz de confusion de 10
matriz_confusion =

```
100 2
0 898
```

Precision (10) = 0.980392

Recall (10) = 1.000000



A simple vista podemos observar que los resultados obtenidos mejoran con creces los obtenidos mediante Bayes ingenuo. Esto se debe a que Bayes supone que los atributos son condicionalmente

independientes dada una clase, cosa que no ocurre en un problema de reconocimiento de dígitos manuscritos, sino todo lo contrario.

Ahora, vamos a comparar los resultados obtenidos de la práctica anterior, los obtenidos con Bayes ingenuo y por covarianzas completas.

		1	2	3	4	5
P4 RegLog	PRE	0.889908	0.867347	0.850000	0.890000	0.876289
	REC	0.970000	0.850000	0.850000	0.890000	0.850000
Bayes Ingenuo	PRE	0.771654	0.835165	0.818182	0.818182	0.867647
	REC	0.980000	0.760000	0.720000	0.720000	0.590000
Covarianzas Completas	PRE	0.980000	0.933962	0.948980	0.989796	0.978947
	REC	0.980000	0.990000	0.930000	0.970000	0.930000

		6	7	8	9	0
P4 RegLog	PRE	0.907407	0.928571	0.836735	0.905263	0.989691
	REC	0.980000	0.910000	0.820000	0.860000	0.960000
Bayes Ingenuo	PRE	0.805085	0.954545	0.678899	0.687500	0.936842
	REC	0.950000	0.840000	0.740000	0.880000	0.890000
Covarianzas Completas	PRE	0.990000	0.989796	0.903846	0.959596	0.980392
	REC	0.990000	0.970000	0.940000	0.950000	1.000000

	Errores con datos de test
P4-RegLog	10.6 %
Bayes Ingenuo	19.3 %
Covarianzas Completas	3.5 %

Como podemos ver en los resultados obtenidos mediante los 3 métodos de clasificación distintos, el mejor comportamiento lo encontramos, con mucha diferencia, en la clasificación con matrices de covarianzas completas. Los resultados de este método son excelentes, superando en todas las clases el 90% de precisión y recall.