

José Javier Calvo Moratilla
2021/2022

Biometría

Prácticas laboratorio

Introducción:

Para la realización de los laboratorios de la asignatura se han seleccionado tres ejercicios diferentes del temario de la asignatura:

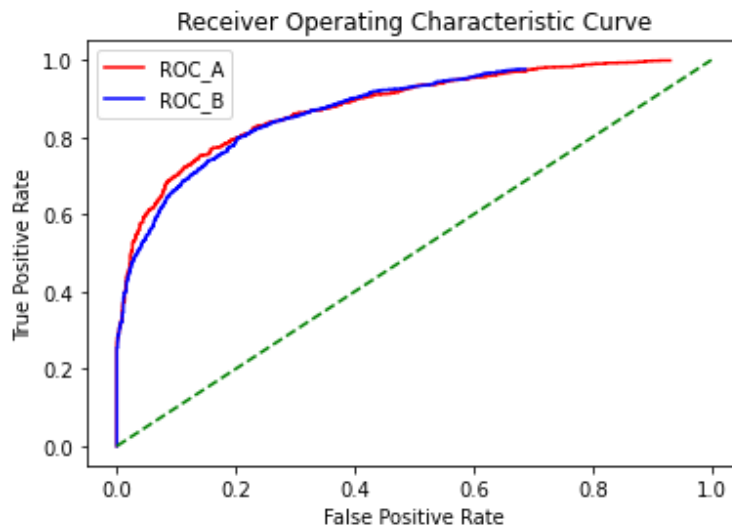
1. Ejercicio Curva ROC
2. PCA
3. LDA

1. Ejercicio Curva ROC

En el primer ejercicio se cuenta como input los scores A y B, de clientes e impostores obtenidos como salida de un sistema biométrico. Con dicha información se realizan las diferentes medidas de calidad estudiadas en la asignatura para comprobar la fiabilidad del sistema biométrico.

En primer lugar para poder obtener la curva ROC de A y B se necesita ordenar los umbrales de menor a mayor, se recorren cada uno de ellos y se obtiene para cada umbral el valor de verdaderos positivos (VP) y de falsos positivos (FP).

Una vez obtenidas las series para scores A y B se dibuja la gráfica para visualizar las dos curvas roc y así poder observar la tasa de error gráficamente.



En nuestro experimento el sistema que presenta una mayor área gráficamente, antes de calcular el área es el sistema A pero no muy lejano a B, por ello se precisan de otras aproximaciones más claras, en dicho sentido en primer lugar se opta por realizar la optimización: $FP(FN = Y)$, donde y es el umbral definido. En el ejercicio se define el umbral 0.2 obteniendo los siguientes resultados:

```
Optim FN A th = 0.2
Valores más próximos: [0.19999999999999996, 0.20069930069930075]

Lista Falsos Positivos [0.2064102564102564, 0.2064102564102564,
0.20576923076923076, 0.20512820512820512, 0.20448717948717948]

Lista th: [0.049813, 0.049877, 0.049942, 0.049971, 0.05]
```

```
Optim FN B th = 0.2
Valores más próximos: [0.19999999999999996, 0.20069930069930075]

Lista Falsos Positivos [0.2076923076923077, 0.2076923076923077,
0.20705128205128207, 0.2064102564102564, 0.20576923076923076,
0.20512820512820512, 0.20448717948717948]

Lista th: [0.041949, 0.041991, 0.042208, 0.042396, 0.042584, 0.042614,
0.042627]
```

Ahora se opta por optimizar $FN(FP = X)$, obteniendo los siguientes resultados:

```
Optim FN A th=0.2
Valores más próximos: [0.19935897435897437, 0.20064102564102565]

Lista Falsos Negativos [0.2027972027972028, 0.2034965034965035]

Lista th: [0.050378, 0.050595]
```

```

Optim FN B th=0.2
Valores más próximos: [0.19935897435897437, 0.20064102564102565]

Lista Falsos Negativos [0.2097902097902098, 0.21048951048951048,
0.21118881118881117, 0.21188811188811185, 0.21258741258741254,
0.21398601398601402, 0.2146853146853147]

Lista th: [0.0444444, 0.044465, 0.044473, 0.044643, 0.044761,
0.045517, 0.045561]

```

Por último se realiza la optimización $FP = FN$ obteniendo los siguientes resultados:

```

Optim FN A

Lista Falsos Negativos y Falsos negativos [0.2, 0.2, 0.2, 0.2, 0.2,
0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2]

Lista th: [0.05, 0.050058, 0.050114, 0.050144, 0.050248, 0.050255,
0.050307, 0.050331, 0.050333, 0.050378, 0.050401, 0.050524, 0.050595,
0.050633, 0.050678, 0.050926, 0.050932, 0.050933]

```

```

Optim FN B

Lista Falsos Negativos y Falsos negativos [0.2, 0.2, 0.2, 0.2, 0.2,
0.2, 0.2, 0.2, 0.2, 0.2, 0.2]

Lista th: [0.042627, 0.04277, 0.042802, 0.04315, 0.04343, 0.043529,
0.043767, 0.043833, 0.043884, 0.043896, 0.043931]

```

Una vez realizadas las optimizaciones se calcula el área mediante el uso de dos métodos diferenciados recogidos en el cuaderno de google colab.

Para el método nº1 se obtienen los siguientes resultados:

Sistema	AUC
A	0.8851
B	0.8755

Para el método nº2 se obtienen resultados bastante similares:

Sistema	AUC
A	0.8818
B	0.8760

Se observa claramente que el sistema A tiene una AROC mayor, por ello tiene un mejor ratio entre VP y FP.

Una vez obtenidas las AUC se obtiene los resultados de D-Prime:

Sistema	D-Prime
A	0.7590
B	0.8608

2. PCA

El ejercicio de PCA consiste en implementar y aplicar dicha técnica para clasificar cara mediante el uso de un modelo de k-vecinos.

En primer lugar se cargan los datos, se identifican 40 sujetos donde se tienen 5 imágenes para el conjunto de entrenamiento y otras 5 para el conjunto de test.

Las imágenes tienen un shape de (1127, 92), por lo que se obtienen en total 10304 características (Features) de cada imagen.

Una vez procesadas todas las imágenes se almacena la información en un array de Numpy, obteniendo un conjunto de entrenamiento con unas dimensiones:

X Train Shape: (200, 10304)

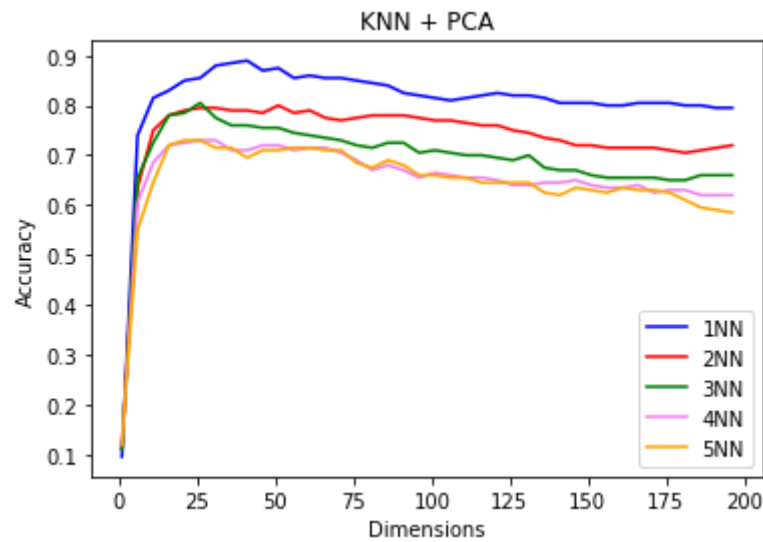
Y Train Shape: (200,)

Para realizar el proceso de PCA se siguen los siguientes pasos:

1. Se obtiene la media de los datos
2. Se obtiene la matriz A
3. Se obtiene la matriz de covarianzas a partir de la matriz A
4. Se obtiene Delta-prima y B_prima con la función de eigenvectors
5. Con dichas matrices se obtienen los eigenVectors y eigenValues
6. Se normalizan los eigenVectors

7. Se ordenan los eigenVectors de mayor valor de eigenValues a menor.
8. Una vez obtenidos los mejores eigenVectors se multiplican por la matriz de datos para reducir la dimensionalidad. Se reducen tanto los conjuntos de entrenamiento como el de test
9. Por último se entrena y evalúa k_vecinos con diferentes parámetros como número de vecinos y valor de dimensionalidad.

Después de realizar la ejecución el mejor resultado obtenido es:



1 Vecino, d= 41, Score: 0.89

Una vez obtenido el mejor resultado para PCA se realiza otra técnica llamada LDA.

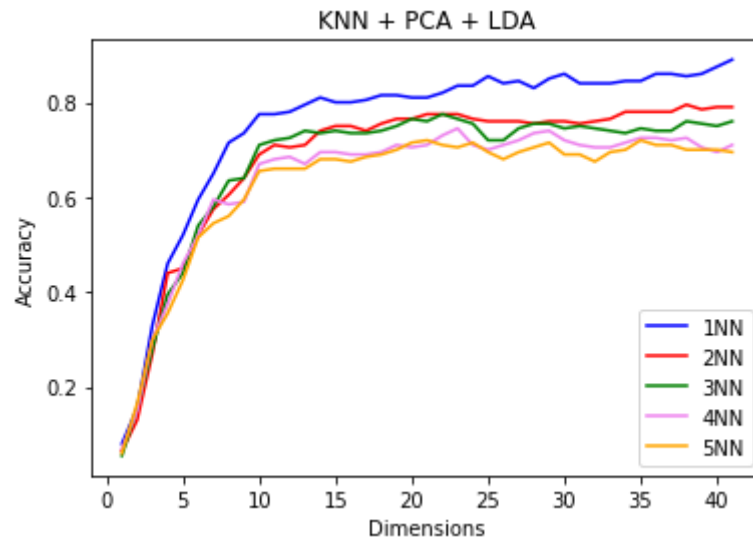
3. LDA

Para realizar el ejercicio de LDA primero se ha aplicado PCA sobre los datos, que partiendo de los resultados se profundiza con los parámetros obtenidos.

Para realizar LDA se han seguido los siguientes pasos:

1. Se obtiene la media de los datos a los que se les ha aplicado PCA.
2. Se definen las matrices SB y SW
3. Para cada label diferente se obtienen la matriz de clase correspondiente, su media y se calcula su respecto SB y SW, cuyos valores se van acumulando en las dos matrices
4. Se obtiene la matriz C
5. Se obtienen los eigenValues y eigenVectors de la matriz C
6. Se ordenan los eigenVectors de mayor a menor valor de eigenValues

7. La matriz B de eigenVectors obtenida se utiliza para transformar los datos ya transformados por la PCA.
8. Dichos datos transformados se utilizan de nuevo en el modelo k-vecinos probando los parámetros de dimensionalidad y número de vecinos, obteniendo los siguientes resultados:



1 Vecino, d= 41, Score: 0.89

No se consigue mejorar el resultado obtenido por PCA, ello implica que la dimensión no se puede reducir en mayor medida, por ello el valor óptimo para realizar la reducción de dimensionalidad, perdiendo el menor número de información discriminativa es con un valor $d=41$, utilizando el modelo con 1-vecinos.