Preprocesado de datos - Gestión de la información

Giusseppe Bervis

Agosto 2021

Filtrado y preprocesado de datos de espectrometría de masas para diagnóstico de cáncer. En el Hospital Central disponen de un espectrómetro de masas para analizar muestras de biopsias, cara a diagnosticar cáncer. Con cada muestra, si todo sale bien, se obtiene un impresionante listado de frecuencia/amplitud del espectro de la muestra, porque el aparato tiene mucha resolución.

Tu misión es recoger ese fichero de datos (podemos pactar el formato) y filtrar casos que no estén bien, así como posiblemente seleccionar alguna representación más comprimida, pero igual de útil. Para ayudarte en esa misión, uno de los ficheros es histórico donde quedó anotado si el paciente realmente tenía cáncer o no.

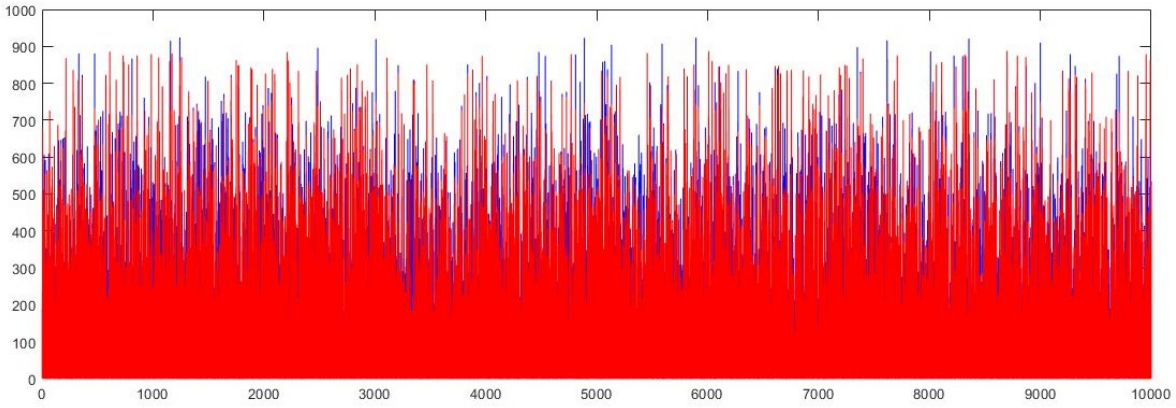
1. **Contenido de los datos:** La información proporcionada consta de 100 muestras, ubicadas en la variable «*datos*»; asimismo, cada muestra consta de 10000 puntos que corresponden con los datos arrojados por el espectómetro. También, se tiene una variable «*salida1*» que contiene la información de si la persona tenía cáncer o no. Cabe destacar que ambas variables, «*datos*» y «*salida1*», contienen datos de tipo «*double*», que es el tipo de datos numéricos por defecto en MATLAB.

   Respecto a los datos, tenemos que hay 44 muestras que son positivas y 56 muestras que son negativas para cáncer. A su vez, la matriz en la que están alojados los datos es una matriz en la que el 54 % (540941) de los datos son no nulos.

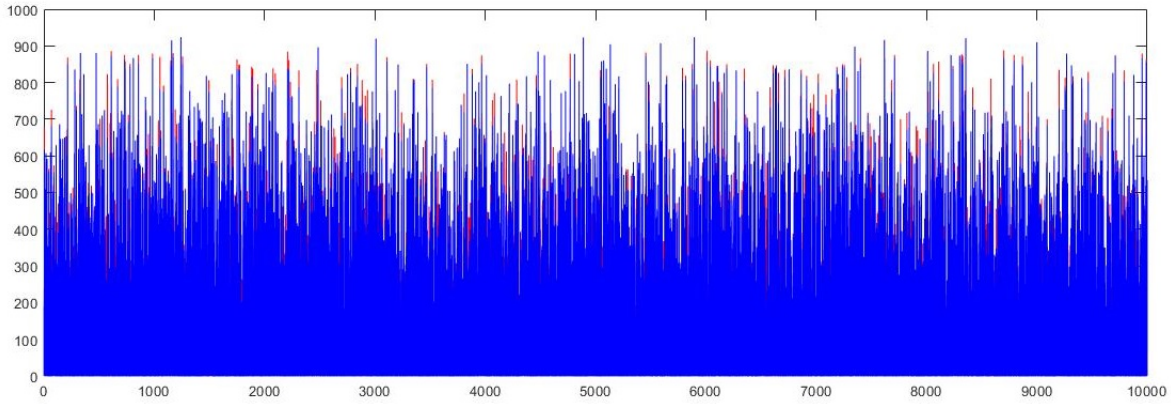
2. **Visualización de los datos:** Para obtener acceso a los datos, primero estos se deben de cargar: `load('arcene.mat')`. Para la visualización de los datos, generamos una sucesión de 10000 valores, de 1 a 10000, y grafícamos independientemente las muestras positivas «datosP» (en color rojo) y las negativas «datosN» (color azul), datos que han sido previamente extraídos de «*datos*». Dicho proceso queda recogido en el siguiente código.

   ```
   t = 1:10000;
   n = 1:3;
   plot(t,datosN,'color','blue')
   hold on
   plot(t,datosP,'color','red')
   ```

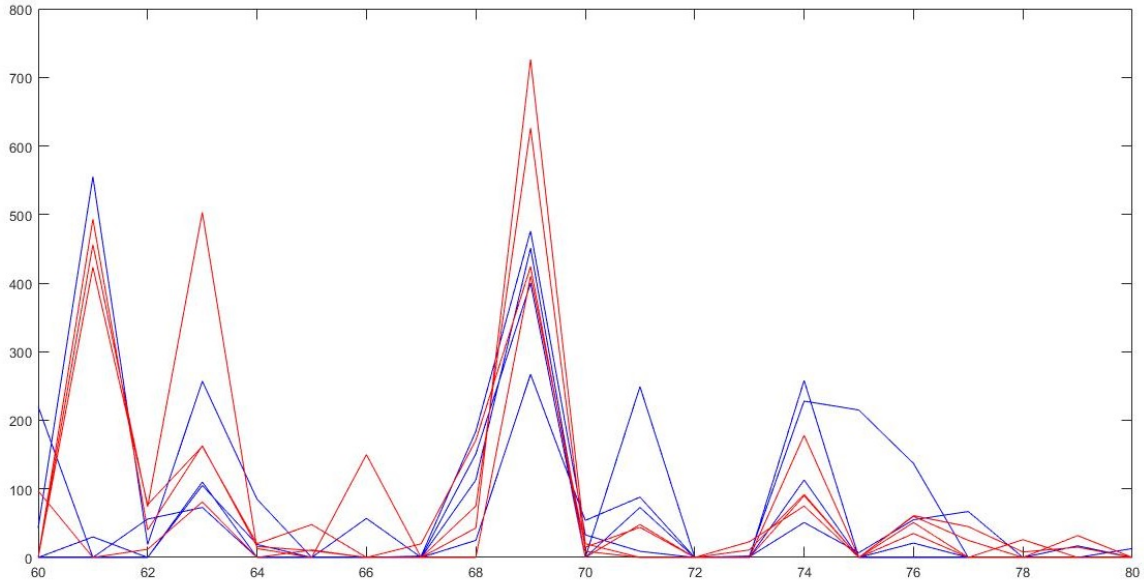
   Cuyo resultado es:



A primera impresión, pareciera que los valores positivos (en rojo) toman valores más bajos en la mayoría de los valores de la espectometría. En la siguiente imagen se superponen los valores negativos a los positivos, lo que nos da la siguiente imagen.



Aunque, evidentemente, lo anterior, es una aproximación visual que no se debe cumplir por regla general, como en la siguiente imagen (en el punto 69).



3. **Detectando datos no númericos:** Es posible que los datos tengan valores no numéricos, por lo que hay que confirmar si existen dichos valores en nuestros datos. Para esto, se usó el siguiente código, y se muestra su resultado.

```
>> sum(sum(isnan(datos)'))
ans = 0
```

   Como puede verse no hay presencia de dichos valores en la muestra.

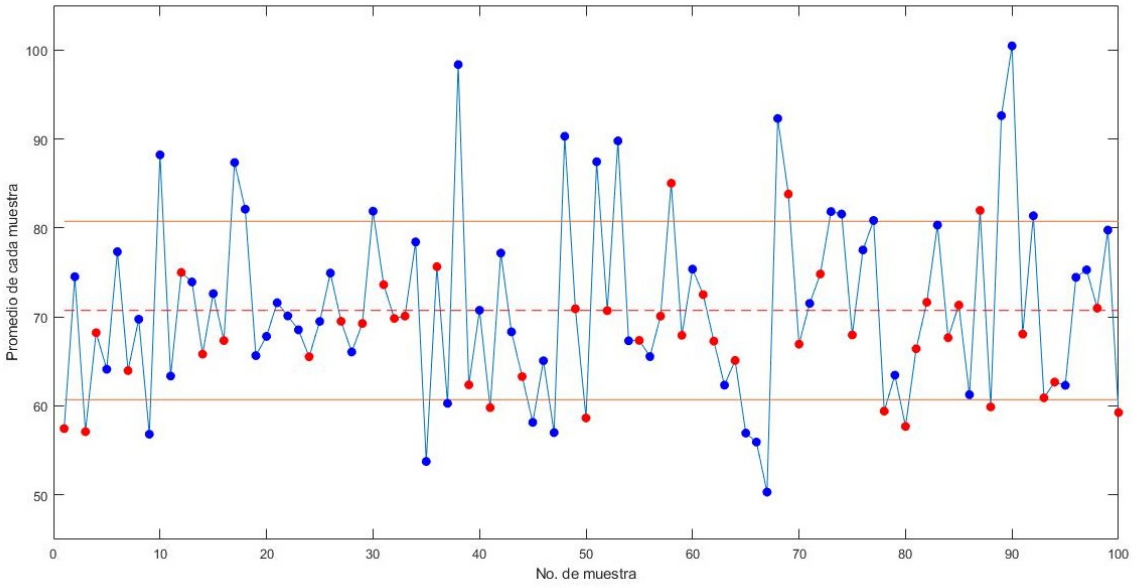
4. **Estadística de los datos:** A continuación se presentan las estadísticas de los datos.

Summary de «datos»

Mínimo:	0
Máximo:	924
Media :	70.7267
Desviación típica (avg.) :	124.5897

```
>> nsignals = 1:100;
>> avg = mean(datos,2);

>> min(avg)
ans = 50.3170

>> max(avg)
ans = 100.4485

>> mean(avg)
ans = 70.7267
```

```
>> std(avg)
ans = 10.0148

plot(nsignals,avg)
```

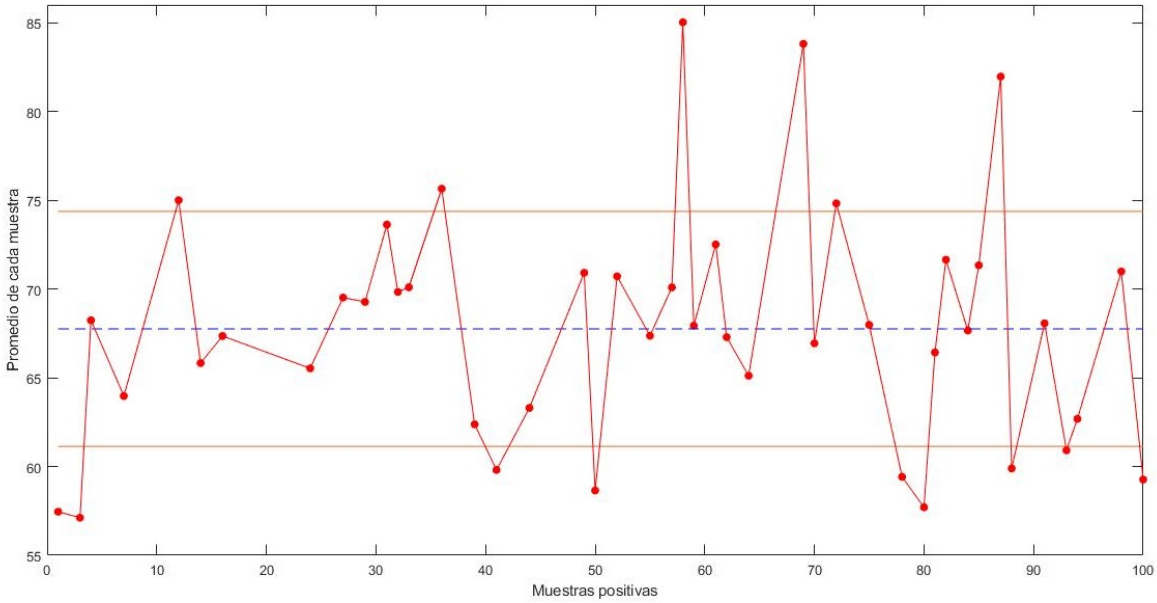


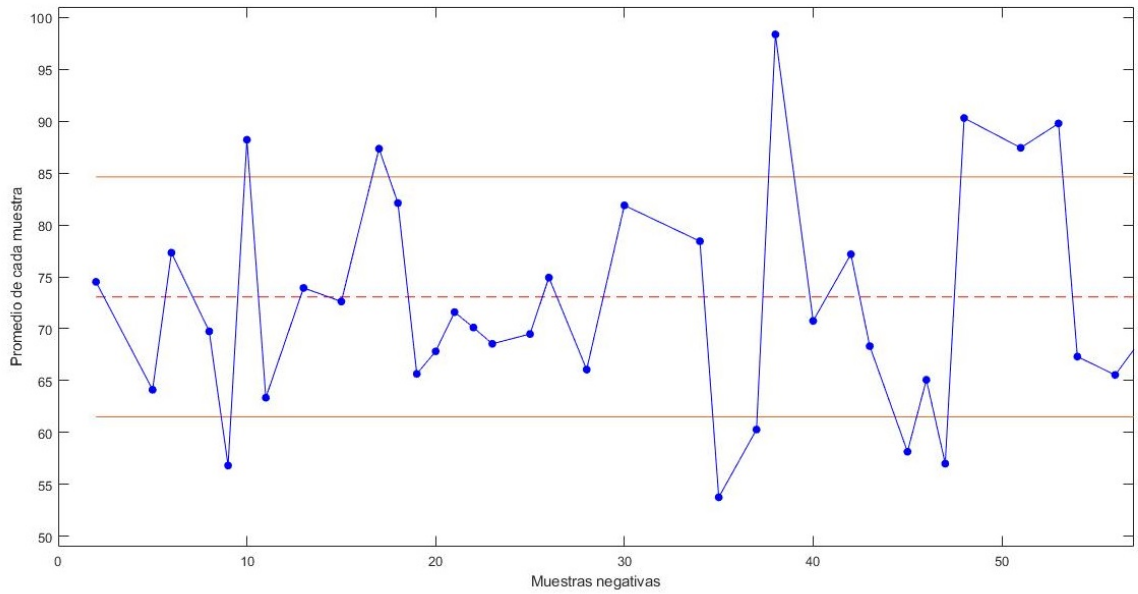
Puede verse que los picos más altos se dan en las muestras 38 y 90, así como los picos más bajos en las muestras 35 y 67, estos son las muestras con mayores y menores medias. Cabe destacar que todas estas muestras son negativas para cáncer.

Haciendo un análisis de las dos clases de datos, tenemos:

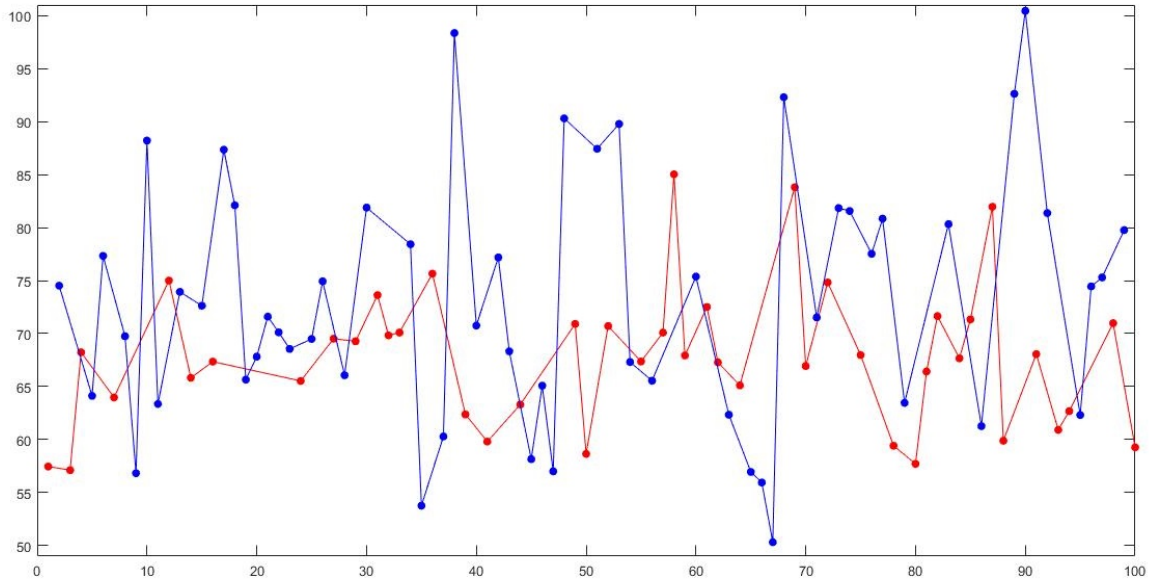
Tipo de muestras	**Mín**	**Media**	**Máx.**	**Desv. Stándar**
Positivas	57.1133	67.7496	85.0222	6.6149
Negativas	50.3170	73.0659	100.4485	11.5612

De aquí se deduce, lo que se había comentado líneas anteriores, las muestras negativas tienen valores más altos en la espectometría, en comparación con las muestras positivas.





A continuación se presenta una comparación visual entre los valores de las muestras positivas y negativas.



5. **Verificando la presencia de «*Outliers*»:** Visualmente no se nota presencia de valores que no concuerden con el resto de los datos. En todo caso, se considerarán outliers aquellos valores que se encuentren a más de 10 veces la desviación típica de la media. Debido a la gran cantidad de valores 0, se optó por 10 veces la desviación típica, cuya medida abarca (en promedio) el rango de espectrometría de masas y, así, cualquier valor fuera de dicho rango sería considerado un «*Outliers*». Esto que se recoge en el siguiente código:

```
>> sum(sum(isoutlier(datos,'mean','ThresholdFactor',10)))
ans = 0
```

   De esto se deduce que no hay presencia de «*Outliers*», en los datos, y por lo tanto no hay muestras que remover.

6. **Reducción de dimensión:** Se usará el análisis de componente principales para realizar la reducción de la dimensión de los datos. Se hará uso del paquete «*drtoolbox*», primeramente, para determinar la cantidad de dimensiones a usar.

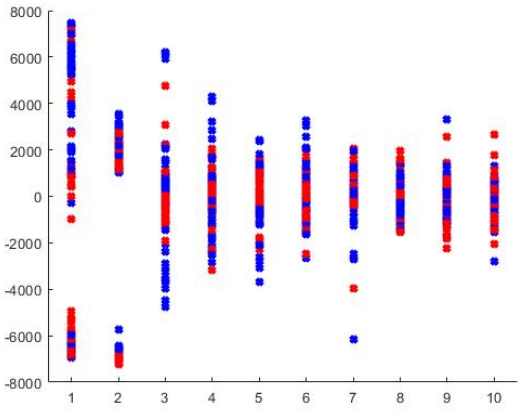
```
% Analisis de las PC
intrinsic_dim(datos, 'EigValue')
ans = 10
```

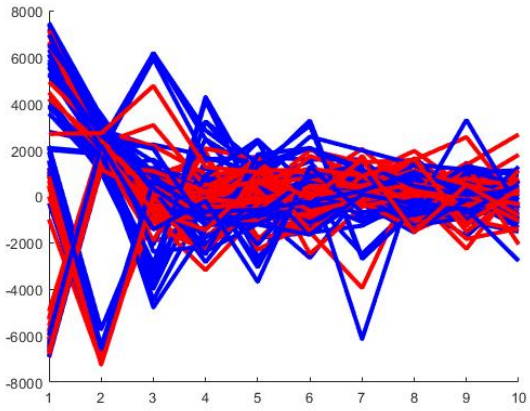
   De lo que se tiene que la cantidad mínima de componentes principales para poder representar los datos es 10.

   PCA con 10 PC

```
Ndatos = compute_mapping(datos,'PCA',10);
t = 1:10;
figure, hold on
```

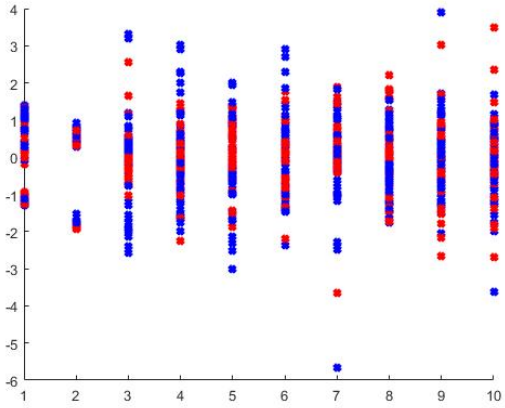
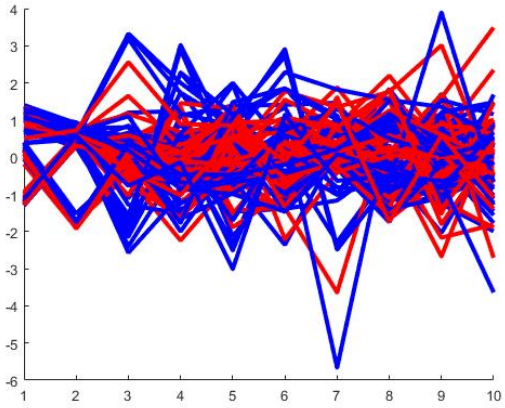
```
for i = 1:size(Ndatos,1)
    if(salida1(i) == 1)
        plot(t,Ndatos(i,:),'rx','LineWidth',3);
    elseif(salida1(i) == 0)
        plot(t,Ndatos(i,:),'bx','LineWidth',3);
    end
end
xlim([0.5,10.5])
```





7. **Normalización de los datos:** Se normalizarán los datos, de modo que tengan media 0 y desviación estándar 1. Esto es posible ya que, por lo menos visualmente, la distribución de los datos en la reducción de dimensión siguen distribuciones que aparentan ser normales.

```
Z = normalize(Ndatos);
figure, hold on
for i = 1:size(Z,1)
    if(salida1(i) == 1)
        plot(t,Z(i,:),'r','LineWidth',3);
    elseif(salida1(i) == 0)
        plot(t,Z(i,:),'b','LineWidth',3);
    end
end
```

8. **Algunas pruebas de agrupación:**

   - **Agrupando los datos originales:** Al aplicar «*kmeans*» a los datos originales tenemos la siguiente agrupación de datos:

	Grupo 1	Grupo 2
Positivos	22	22
Negativos	16	40

   - **Agrupando los datos normalizados:** Se hicieron 100 iteraciones para probar el agrupamiento, teniendo como promedio los siguientes resultados.

	Grupo 1	Grupo 2
Positivos	20	24
Negativos	26	30