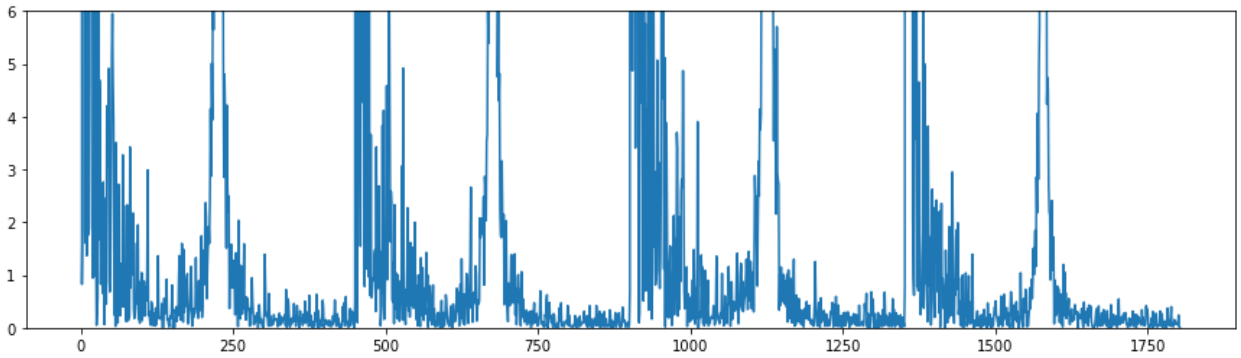
Extracción de Características y Armado de Datasets

A) Introducción:

a) Familiarizarse con la clase BCIDataset del notebook asociado al TP. Estudiar el código en detalle. Estudiar los atributos del objeto y su impacto. Estudiar los atributos generados internamente (“resultados”) en la clase en relación a lo charlado en la presentación del TP.

Se realiza un primer análisis para un sujeto y una sesión
metadata
devuelve la matriz de metadata generada por purness (pureza, t_start y t_end son el tiempo de inicio y de fin del ejemplo, el subject del cual fue extraido el ejemplo y su sesion)
son False los valores impuros que contengan mas de una etiqueta
Graficamos utilizando la función de extracción de features naif.
Se puede ver cómo están los 4 canales solapados.



Mostramos las dimensiones de los atributos de la clase:
print('dimension matriz senal ', dataset.get_X_signal().shape)
print('dimension matriz features ', dataset.get_X_features().shape)

Devuelve una matriz donde en cada fila tengo cada ejemplo y en las columnas las features extraidas
La cantidad de filas va a depender del tamaño de la ventana elegida, del overlapping y de la extensión de la señal
La cantidad de columnas de X_signal es el tamaño de la ventana por la cantidad de canales (900*4, en éste caso).
La cantidad de columnas de X_features va a depender del tamaño de la ventana por la cantidad de canales dividido 2.

dimension matriz senal (151, 3600)
dimension matriz features (151, 1804)

dataset.get_Y().shape
devuelve el vector target

dataset.__getitem__(0)

me devuelve el valor de un ejemplo:

```
{'signal': array([ -1.86,  10.77,  87.61, ...,  16.48, -32.06, -62.72]),
 'features': array([2.38843219e+04, 6.95994694e+01, 1.14199542e+01, ...,
        1.18164148e-01, 9.98628650e-02, 4.53153361e-01]),
 'label': array([99.]),
 'metadata': array(['True', '0.0', '4.495', 'AA', '0'], dtype='<U32')}
```

b) Estudiar cómo varía el número de ejemplos en el dataset y la dimensión de cada dato según la variación de la ventana de tiempo seleccionada y el criterio de solapamiento.

modificamos la dimensión del solapamiento

```
dataset_of1 = BCIDataset(csvs_path, subject='AA', session='0',overlapping_fraction=1/2)
dataset_of1.get_X_features().shape
```

disminuye la cantidad de ejemplos al aumentar el solapamiento, aunque las columnas se mantienen iguales.

modificamos la dimensión del solapamiento

```
dataset_of2 = BCIDataset(csvs_path, subject='AA', session='0',overlapping_fraction=1/4)
dataset_of2.get_X_features().shape
```

aumenta la cantidad de ejemplos al disminuir el solapamiento y las columnas se mantienen iguales.

B) Características Temporales:

a) Usando BCIDataset junto con el extractor de features básico de fft ("naif_fft_features"), analice la influencia que tiene el tamaño de la ventana en el dominio de tiempo en la resolución en frecuencia del espectrograma de potencia.

modificamos la dimensión de la ventana

```
dataset_ws1 = BCIDataset(csvs_path, subject='AA', session='0',window_size=600)
dataset_ws1.get_X_features().shape
```

aumenta la cantidad de ejemplos pero disminuye cant de features

```
Out[15]:
(226, 1204)
In [16]:
dataset_ws1.get_X_signal().shape
Out[16]:
(226, 2400)
```

Calculamos la resolución

resolucion 0.3333333333333333

modificamos la dimensión de la ventana

```
dataset_ws2 = BCIDataset(csvs_path, subject='AA', session='0',window_size=1200)
dataset_ws2.get_X_features().shape
(112, 2404)
dataset_ws2.get_X_signal().shape
(112, 4800)
```

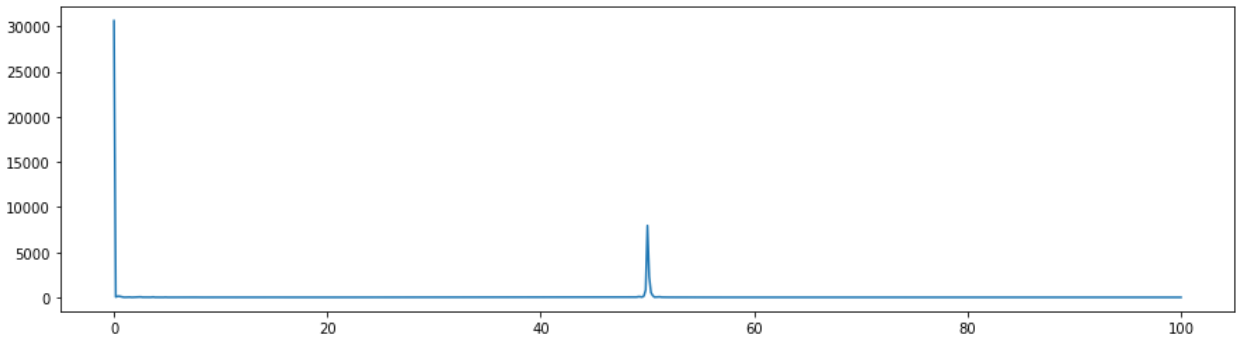
disminuye la cantidad de ejemplos pero aumentan cant de features
Calculamos la resolución
resolucion 0.16666666666666666
resolución en frecuencia = f muestreo/tamaño de la ventana.

resolución en frecuencia = 200/(tamaño de la ventana)

Al aumentar el tamaño de la ventana aumenta la resolución en frecuencia y al disminuir el tamaño de la ventana disminuye la resolución en frecuencia.

Esta resolución es inversa a la resolución en el tiempo ya que al aumentar la ventana se genera una menor cantidad de ejemplos temporales.
Como queremos una resolución de aproximadamente 0.25, o en su defecto menor a 0,5.
Calculamos el tamaño de la ventana para esa resolución
200/0.25=800.0

Vemos cómo los datos están influenciados por la continua y la frecuencia de 50 Hz.



b) Teniendo en cuenta el inciso anterior, las frecuencias de estimulación y la frecuencia de muestreo pertinentes, ¿cuál considera que es el número adecuado de muestras temporales que puede recortar conservando la mayor cantidad de información útil en el dominio de la frecuencia? (t).

- La resolución en frecuencia pertinente podría ser 0,5/2 ya que 0,5 Hz es el mínimo paso que nuestra senal va a expresar debido a que las frecuencias de estimulación están en 12,5 y 16,5 Hz.
- Usaríamos 0,25 Hz o un valor menor como un factor de seguridad para tomar la variación de señal correspondiente.
- Lo ideal sería tomar la resolución más pequeña posible pero eso conlleva a una mala resolución en el tiempo y por ende proporciona muy poca cantidad de muestras para realizar el análisis.

Se prueba modificando la dimensión de la ventana tal que su resolución en frec sea menor a 0,25Hz.

Al obtener un error con un window_size de 800, se debe poner un tamaño de la ventana que sea múltiplo del overlapping.

dataset_ws3 = BCIDataset(csvs_path, subject='AA', session='0',window_size=750)
dataset_ws3.get_X_features().shape
(181, 1504)
vemos una resolución aproximada a 0,25 Hz
resolucion 0.26666666666666666

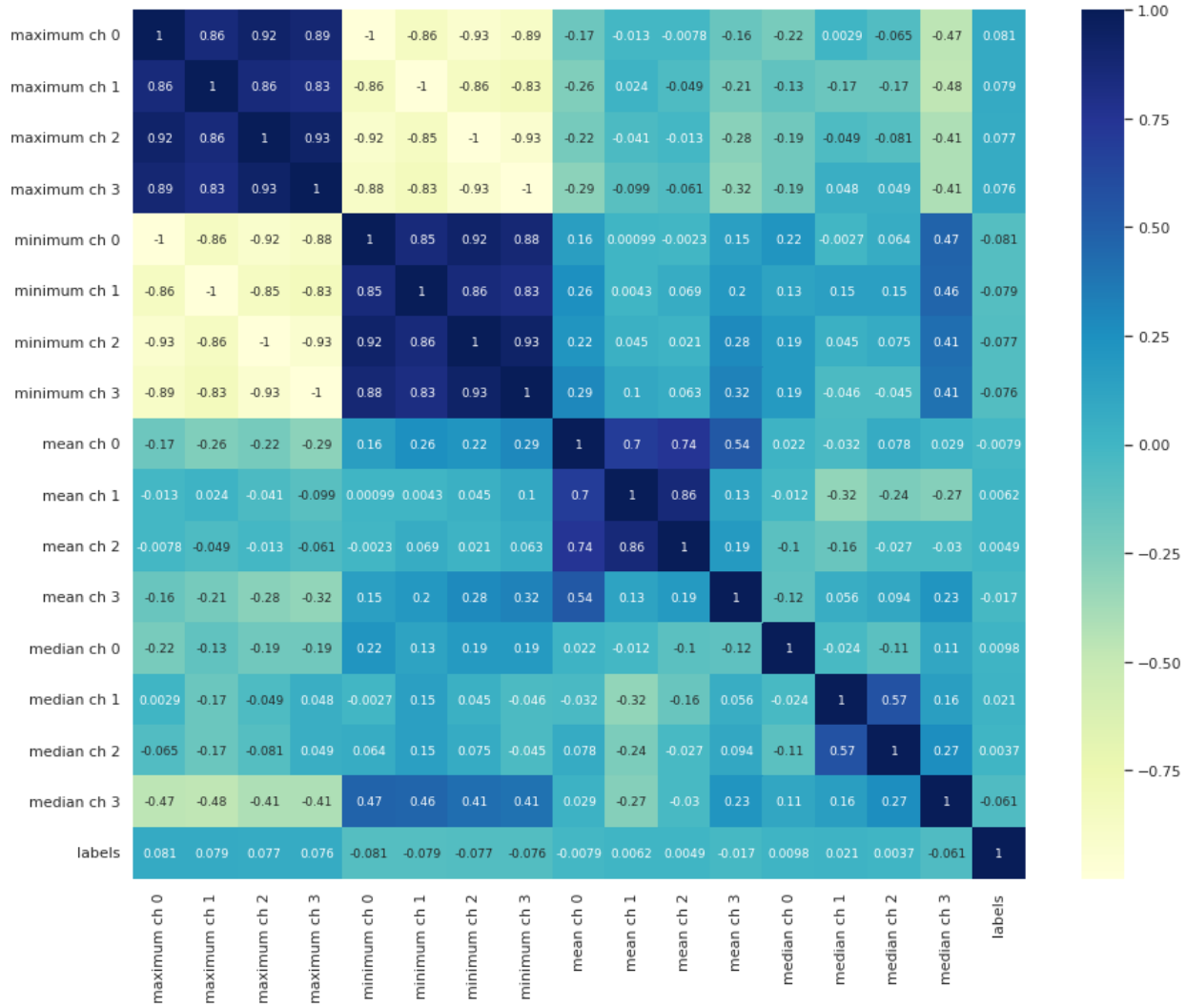
c) En adición a la serie temporal cruda -"complete_examples_signal"- (concatenada o no a lo largo de los canales, según su elección), defina una estrategia de extracción de atributos en el dominio de tiempo que opere sobre la serie cruda, ejemplo: algún criterio como la media en cada canal para el ejemplo. Sean creativos pero no dediquen mucho tiempo a este inciso, es más bien para tener un punto de comparación.

Creamos funciones con estadísticos separados, ya que al hacerlo en una única función, el orden obtenido no era el correcto. Los mismos son:
Máxima amplitud
Mínima amplitud
Media y mediana de las amplitudes

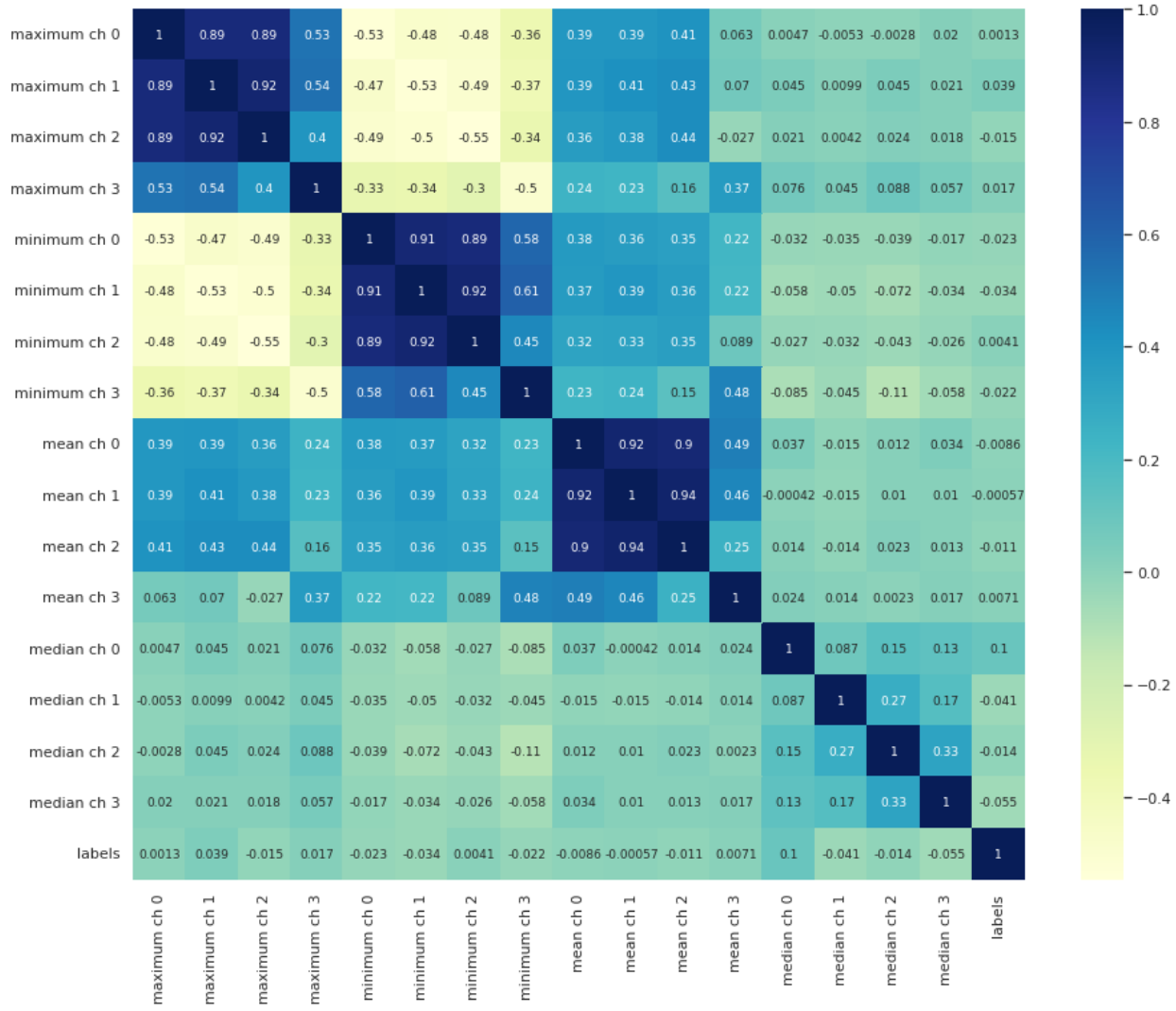
d) Guarde los datasets generados de la forma que considere conveniente.
Unificamos los features en un único dataframe.

	maximum ch 0	maximum ch 1	maximum ch 2	maximum ch 3	minimum ch 0	minimum ch 1	minimum ch 2	minimum ch 3	mean ch 0	mean ch 1	mean ch 2	mean ch 3	median ch 0	median ch 1	median ch 2	median ch 3
0	88.12	93.75	96.47	27.89	-94.32	-92.74	-99.53	-27.46	-0.11	-0.12	-0.12	-0.06	0.22	0.04	0.07	0.21
1	99.88	104.14	116.10	36.24	-85.61	-92.74	-92.58	-27.46	0.22	0.23	0.25	0.10	0.06	-0.05	-0.12	0.15
2	88.11	93.74	96.47	27.89	-85.65	-92.78	-92.62	-27.47	-0.07	-0.09	-0.08	-0.03	-0.15	0.00	-0.03	0.04
3	12.40	13.27	16.23	10.64	-55.67	-63.68	-54.84	-24.49	-0.18	-0.19	-0.17	-0.04	-0.13	0.03	0.03	0.05
4	45.04	48.70	59.06	17.05	-12.29	-17.04	-18.32	-11.00	0.21	0.20	0.22	0.06	-0.24	0.07	0.05	0.05

e) Para cada dataset, analice la contribución de información de cada feature estudiado al propósito de clasificación. Estudie la correlación entre features. Estudie la correlación entre features y etiquetas. En conjunto con la exploración de los TP anteriores, ¿considera útil estos atributos?
Para evaluar correctamente el exctrctor de features se lo debe hacer sin las etiquetas 99 ya qué estas aportan mucha variabilidad qué hace qué no se logre ver el beneficio del extractor.
Calculamos la correlación de los features calculados con el label
 todos los labels incluído el 99



sin usar labels 99



Observación:

Se visualiza una muy baja correlación entre los features calculados y los labels aunque alta entre algunos features.

C) Características Espectrales (en Frecuencia):

a) Usando BCIDataset, con el extractor de features básico de fft, genere el dataset de ejemplos utilizado como atributos el espectrograma de potencia. Determine la estrategia que considere más pertinente, si concatenar los canales o trabajar los canales en forma individual.

Se decide trabajar con todos los canales, todos los individuos y sesiones para generalizar lo mayor posible nuestro análisis.

Shape de features: (2614, 1504)

son False los valores impuros que contengan más de una etiqueta

True 1921
False 693
cantidad de etiquetas 1 481
cantidad de etiquetas 2 524
cantidad de etiquetas 99 1609
resolucion 0.26666666666666666

b) En adición, a los vectores de atributos del inciso a), generar dos estrategias adicionales de extracción de features en el dominio de la frecuencia, al menos una de ellas tiene que implicar un número de atributos <= 8. Generar los correspondientes datasets, SEAN CREATIVOS. Si reducen la información a algunas frecuencias específicas, tengan en cuenta que la frecuencia de estimulación puede no encontrarse entre los residuos de frecuencia analizados, o aún si fuera así, que la respuesta al estímulo no sea exactamente de la misma frecuencia del estímulo en sí.

Se crean las siguientes funciones extractoras de features en la frecuencia.
La siguiente función calcula la ubicación en frecuencia del valor de amplitud máxima.
Calc_max_filtered

Esta función devuelve un intervalo de la señal filtrada en frecuencia.
Filtered_range

Usamos el método Principal Component Analysis en la señal filtrada
Filtered_pca

Analizamos todos los sujetos con la función de extracción de features 'filtered_fft_features' brindada como guia.
shape senal cruda (2614, 3000)
shape senal filtrada (2614, 1504)
No utilizamos ésta función para analizar correlación, ya que devuelve una gran cantidad de features de frecuencia que no son de interés (es por ello qué se creo la función filtered_range) .

c) Guarde los datasets generados de la forma que considere conveniente.
Armamos dataframes con cada uno de los features calculados.
Df qué devuelve la función calc_max

	max freq ch 0	max freq ch 1	max freq ch 2	max freq ch 3
0	11.733333	11.733333	11.733333	11.733333
1	10.666667	10.666667	11.200000	12.533333
2	10.666667	10.400000	10.400000	12.533333
3	13.066667	11.466667	13.600000	18.400000
4	12.266667	9.866667	16.800000	12.266667

Luego de obtener el df de la función filtered range se decide agrupar con estadisticos las frecuencias de interes para cada label (entre 12 Hz y 13 Hz para label 1 y entre 16 Hz y 17 Hz para el label 2)
Realizamos estadísticos de media mediana y moda, de los valores de amplitud de cada canal devuelta por la fft para las frecuencias de interés de etiqueta 1 y 2.
Estos estadisticos tiene en cuenta los cuatro canales en conjunto.

freq_label_1_mean	freq_label_2_mean	freq_label_1_median	freq_label_2_median	freq_label_1_mode	freq_label_2_mode
11.652037	4.432589	12.096649	4.709343	0.509747	0.396103
7.032170	6.963447	4.222540	4.095578	0.617204	0.079872
10.077106	6.446917	7.786847	5.418278	0.431159	0.223331
1.266582	0.607291	0.624375	0.436857	0.008577	0.068263
1.279458	1.494135	0.544339	0.954417	0.002034	0.024173

Df generado por el extractor de pca:

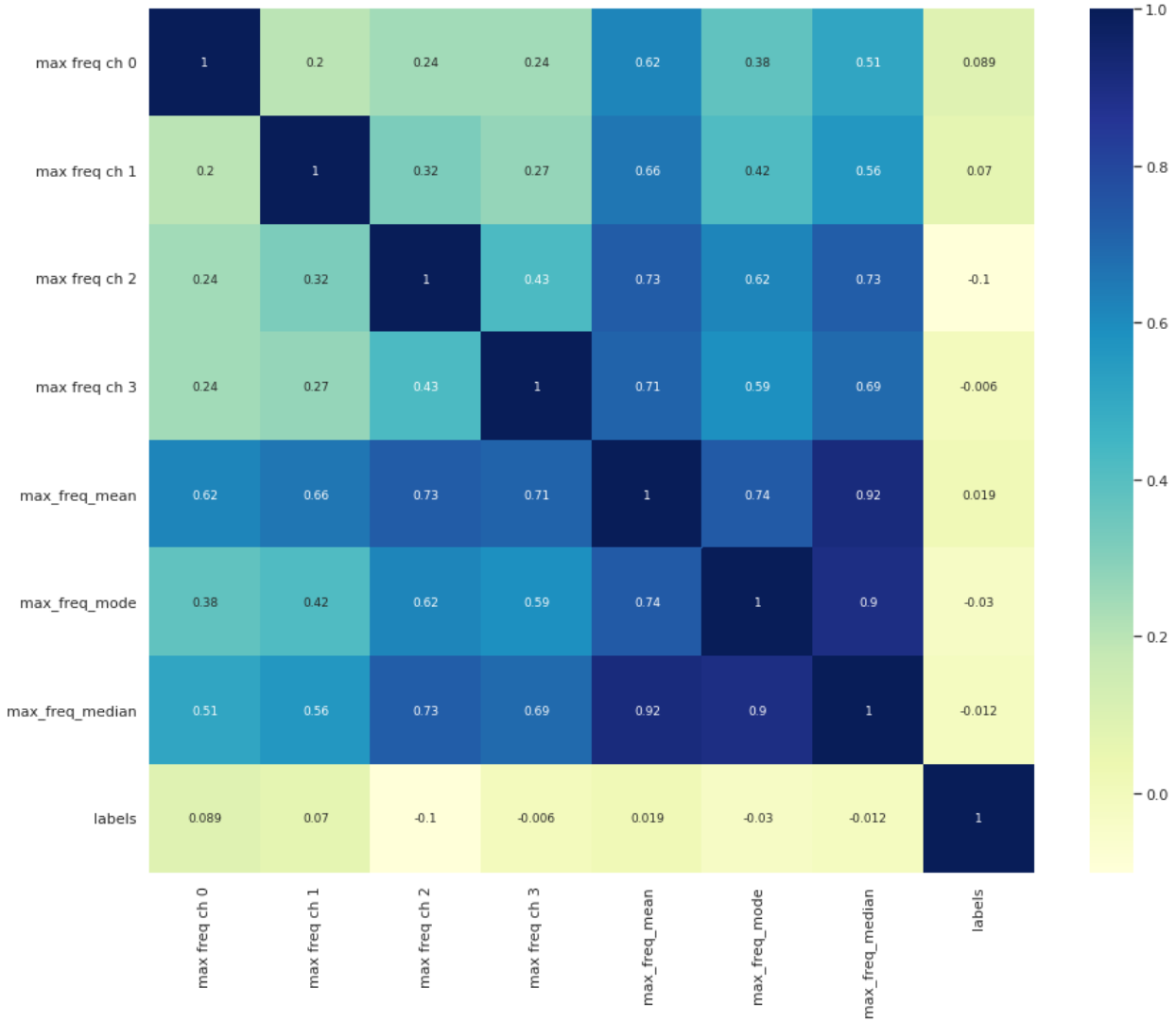
	pca 0 ch 0	pca 1 ch 0	pca 2 ch 0	pca 0 ch 1	pca 1 ch 1	pca 2 ch 1	pca 0 ch 2	pca 1 ch 2	pca 2 ch 2	pca 0 ch 3	pca 1 ch 3	pca 2 ch 3
0	68.916774	-6.344447	2.809746	94.768989	-20.329217	-7.354996	72.589494	-17.699877	75.810609	27.099455	-12.686523	-3.969462
1	24.892492	18.257177	1.609137	35.544948	10.759736	17.261752	42.258027	12.107581	15.804575	0.902452	1.423004	-1.819431
2	19.861768	21.282315	-1.297874	25.812412	17.380757	3.863574	33.809526	16.855151	-0.037189	-2.024077	2.017040	-1.655416
3	-6.344068	-2.756816	-0.167940	-9.643496	0.505119	-1.211254	-12.409630	-0.743190	-4.067499	-4.791792	-1.683306	0.310231
4	-5.127254	-3.527783	0.229703	-8.952149	0.569159	0.612894	-9.867748	0.019406	-3.500792	-3.988550	-1.461725	0.018502

d) Repita el inciso e) del apartado B). En el caso de los vectores de pocos features realice un pairplot para visualizar en baja dimensionalidad el problema de clasificación.

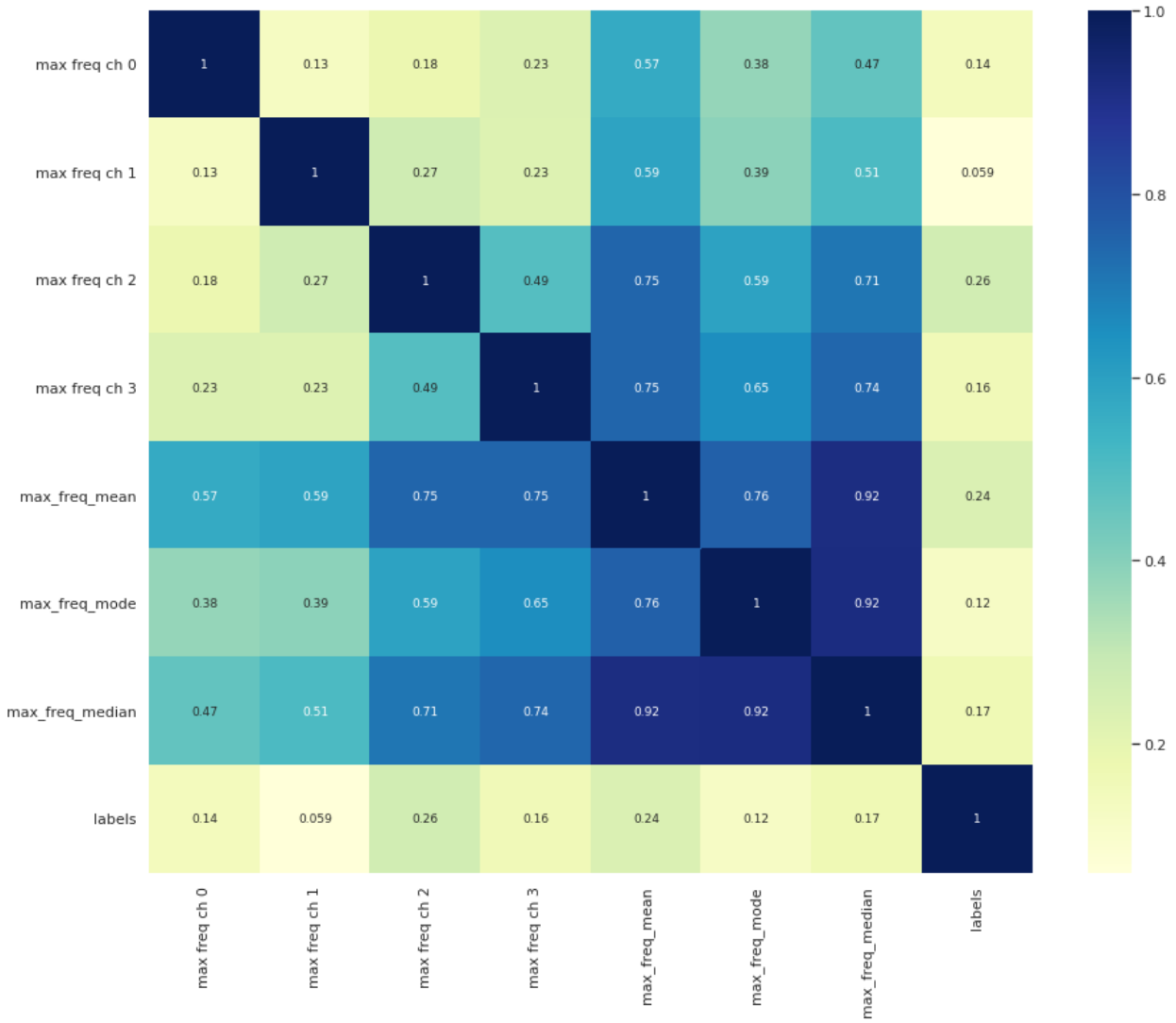
Mostramos el df de calc_max en el cuál se incluyo también la media, la moda y la mediana de los valores de frecuencia de cada canal en conjunto.

	max freq ch 0	max freq ch 1	max freq ch 2	max freq ch 3	max_freq_mean	max_freq_mode	max_freq_median	labels
59	15.733333	16.533333	16.533333	12.533333	15.333333	16.533333	16.133333	2.0
60	12.533333	16.533333	16.533333	16.533333	15.533333	16.533333	16.533333	2.0
61	9.600000	16.533333	16.533333	16.533333	14.800000	16.533333	16.533333	2.0
62	11.200000	11.733333	11.733333	11.733333	11.600000	11.733333	11.733333	2.0
63	12.000000	12.000000	16.800000	16.800000	14.400000	12.000000	13.200000	2.0
64	9.866667	10.400000	16.800000	16.800000	13.466667	16.800000	15.133333	2.0
65	11.733333	11.733333	11.733333	11.733333	11.733333	11.733333	11.733333	2.0
66	12.266667	10.666667	16.800000	11.733333	12.866667	10.666667	12.000000	2.0
72	12.266667	13.333333	13.333333	10.933333	12.466667	13.333333	12.900000	1.0
73	9.600000	13.866667	16.800000	9.866667	12.533333	9.600000	11.200000	1.0
74	10.666667	12.000000	10.666667	13.066667	11.600000	10.666667	11.133333	1.0
75	10.666667	12.266667	10.666667	10.666667	11.066667	10.666667	10.666667	1.0
76	11.466667	11.466667	11.466667	10.666667	11.266667	11.466667	11.466667	1.0
77	10.933333	10.933333	10.933333	10.933333	10.933333	10.933333	10.933333	1.0
78	11.200000	10.933333	11.200000	10.933333	11.066667	10.933333	11.000000	1.0
79	12.266667	10.933333	12.533333	13.600000	12.333333	10.933333	12.300000	1.0
86	12.000000	12.266667	16.800000	16.800000	14.466667	16.800000	15.633333	2.0

Correlación con todas las etiquetas



Correlación sólo con etiquetas 1 y 2



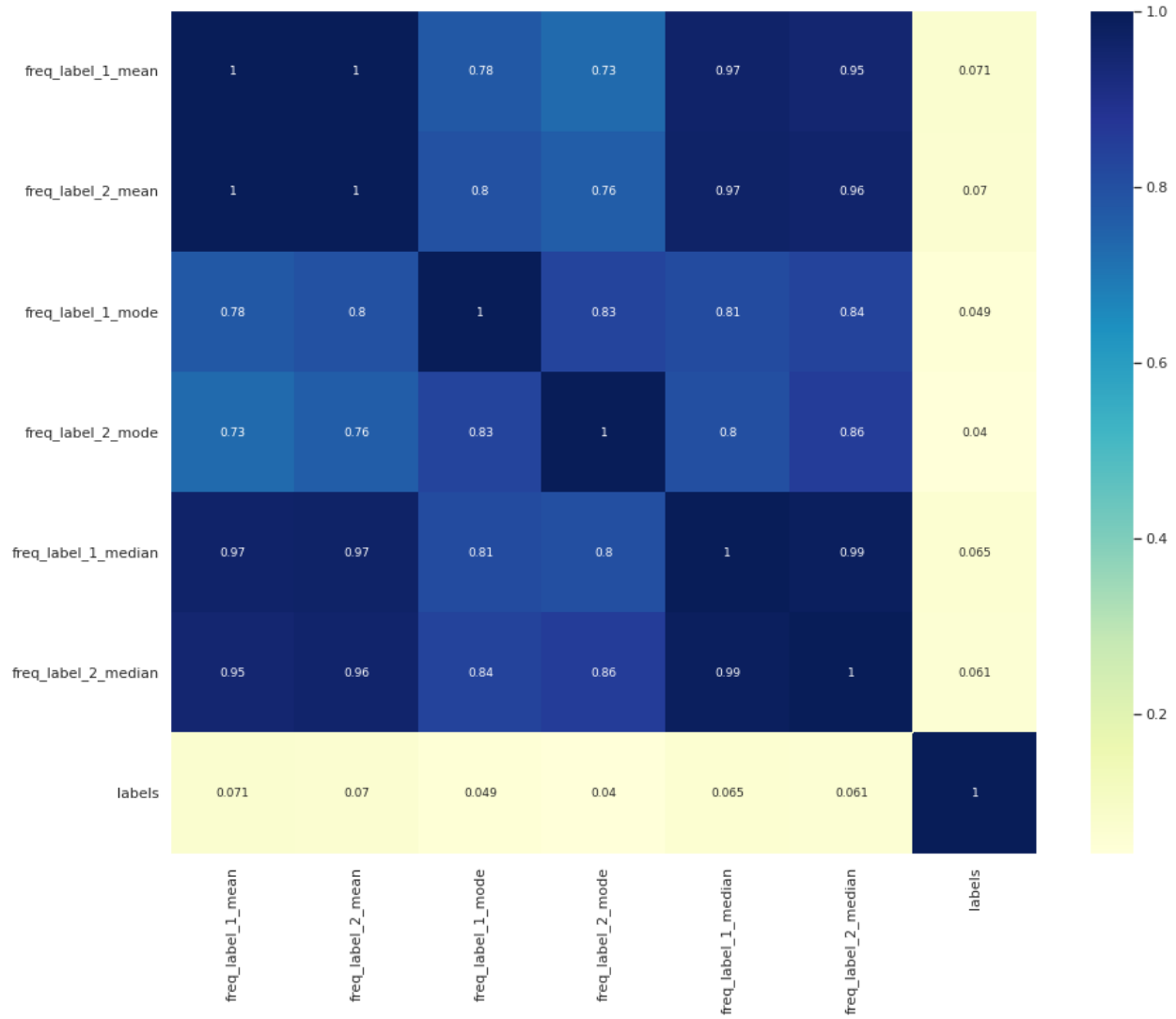
Visualmente se valida la hipótesis de que los valores de etiqueta 1 están cercanos a la frecuencia 12,5 Hz y la etiqueta 2 a 16,5 Hz. Sin embargo ésto no es siempre igual, debido a las condiciones de adquisición de datos.

Visualmente vemos que hay correlación entre la frecuencia con amplitud máxima y la etiqueta, pero matemáticamente la correlación es baja por la variabilidad mencionada.

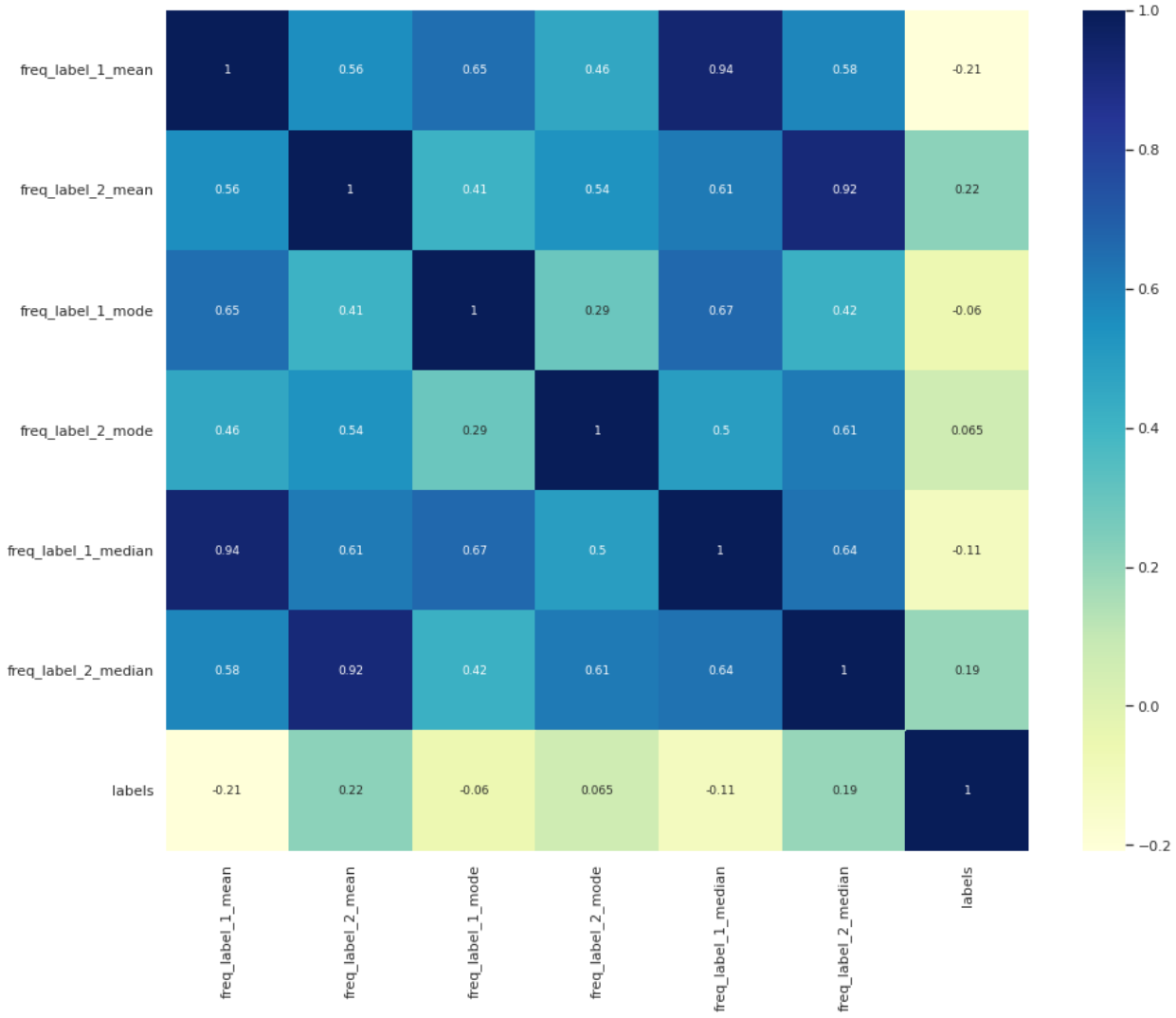
Frequency range

Hacemos solo la correlación con los estadísticos calculados de las frecuencias de interés.

Correlación con todas las etiquetas



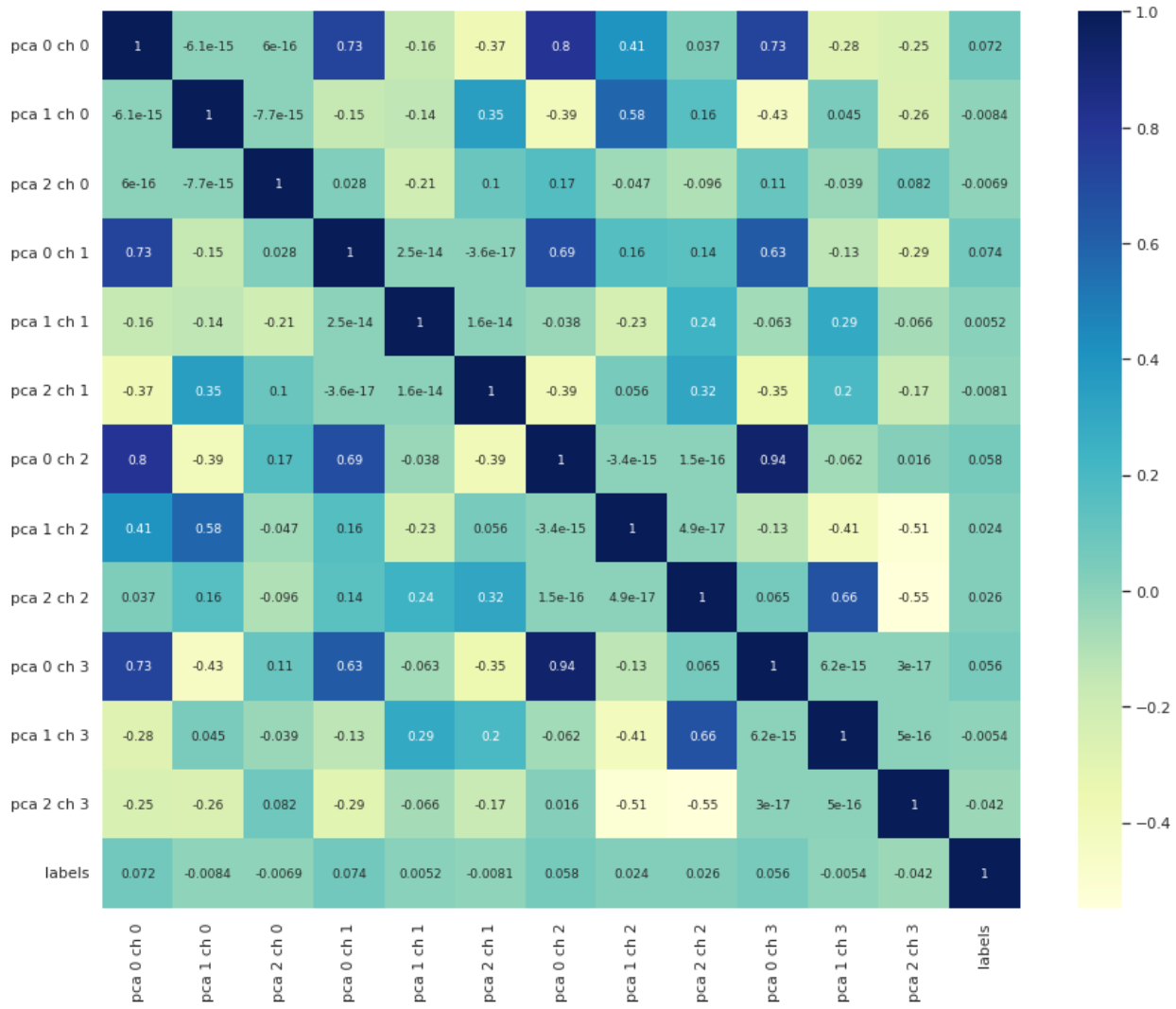
Correlación sólo con etiquetas 1 y 2



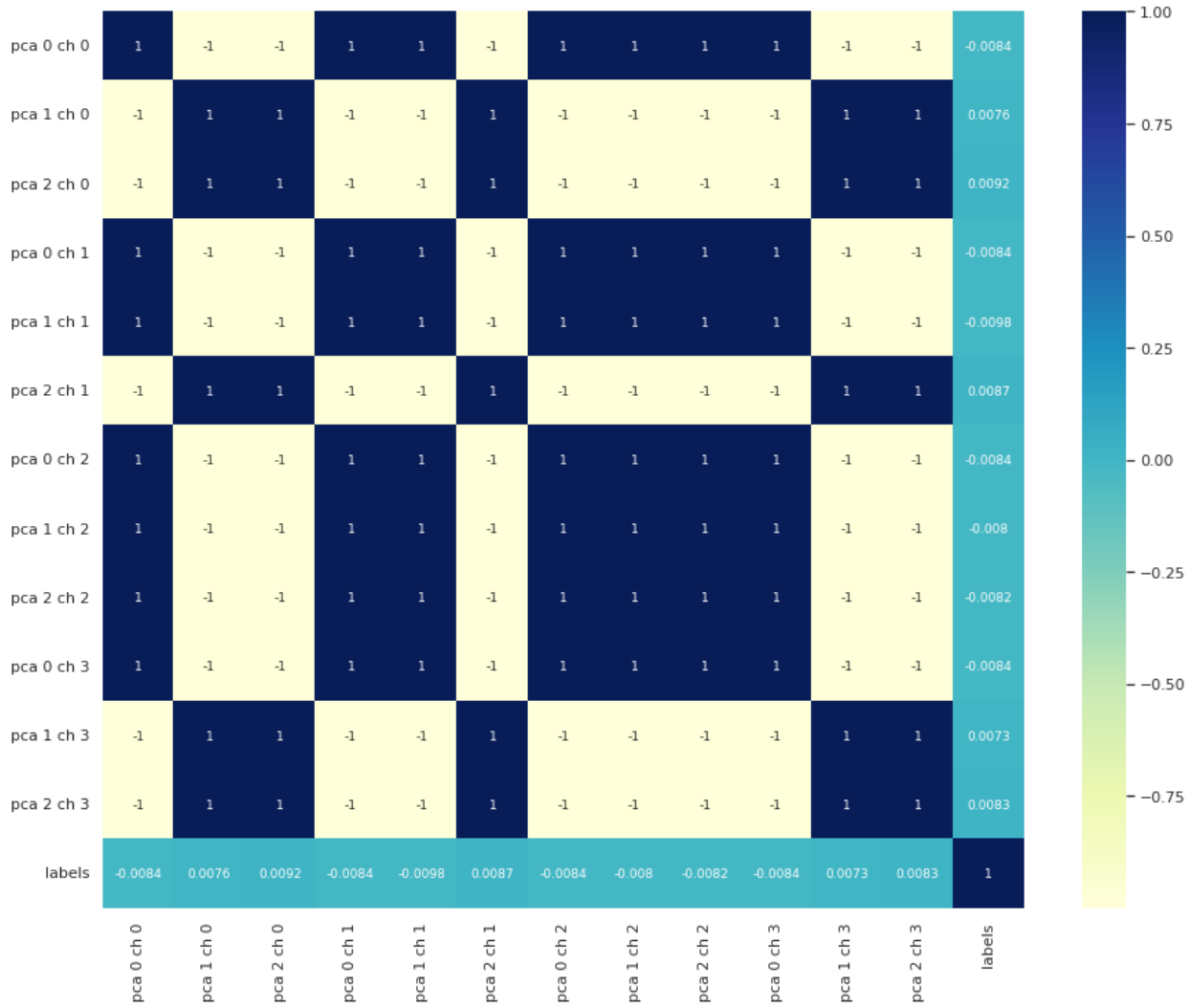
Se ven pequeñas correlaciones con los estadísticos y los labels.

Principal Component Analysis

Correlación con todas las etiquetas



Correlación sólo con etiquetas 1 y 2



No logramos ver correlación con estos features.

Conclusion general:

Para los features calculados se observa una baja correlación con los labels debido a la variabilidad del problema, sin embargo se extraen a continuación los features con mayor correlación observada.

e) Analice la distribución del dataset, número de ejemplos por clase, balance entre clases, número de ejemplos “puros” y ejemplos “impuros”.

Armamos un nuevo df con los features que presentaron mejor correlación

	max freq ch 0	max freq ch 1	max freq ch 2	max freq ch 3	max_freq_mean	max_freq_mode	max_freq_median	freq_label_1_mean	freq_label_2_mean
0	11.733333	11.733333	11.733333	11.733333	11.733333	11.733333	11.733333	11.652037	4.432589
1	10.666667	10.666667	11.200000	12.533333	11.266667	10.666667	10.933333	7.032170	6.963447
2	10.666667	10.400000	10.400000	12.533333	11.000000	10.400000	10.533333	10.077106	6.446917
3	13.066667	11.466667	13.600000	18.400000	14.133333	11.466667	13.333333	1.266582	0.607291
4	12.266667	9.866667	16.800000	12.266667	12.800000	12.266667	12.266667	1.279458	1.494135

freq_label_1_median	freq_label_2_median	purness	labels
12.096649	4.709343	1	99.0
4.222540	4.095578	1	99.0
7.786847	5.418278	1	99.0
0.624375	0.436857	1	99.0
0.544339	0.954417	1	99.0

Distribución de la pureza
1 1921
0 693
Los valores 1 son los True y 0 los False. Se los cambia a valor numéricos para luego poder reducirlos en los algoritmos de predicción.
Distribución de las etiquetas
99.0 1609
2.0 524
1.0 481

Observación:

Luego de no tener valores esperados con lo planteado se realizan pruebas aumentando la cantidad de datos disminuyendo el tamaño de la ventana (empeorando la resolucion a 0,5) y aumentando el overlapping. Sin embargo no se obtienen mejores resultados que los mostrados (se retira del notebook para no hacer muy extenso el análisis). Se decide dejar los resultados obtenidos previamente.

Se cree igual qué esto sirve para evaluar la calidad del feature. Sin embargo si se desea realizar un análisis de ML se podría agrandar más el dataset con lo mencionado previamente.

D) Particionado del dataset.

a) A partir de los datasets generados en los apartados B) y C), definir un esquema de particionado apropiado para el entrenamiento de un algoritmo de clasificación. Tener en cuenta:

i) Frecuencias de cada una de las clases y balanceo.
La etiqueta 99 es casi 3 a 1 con respecto a las otras
Visualizamos las proporciones de cada clase en los conjuntos de train, val y test:
y_train:
99.0 1158

2.0 377
1.0 346
Y_val:
99.0 161
2.0 53
1.0 48
Y_test:
99.0 290
2.0 94
1.0 87
Vemos que se mantiene en cada grupo la proporción casi 3 a 1 de la etiqueta 99 con respecto a las otras.

ii) Inclusión de etiquetas puras e impuras.
Proporción aproximada 3 a 1
Visualizamos las proporciones de pureza en los conjuntos de train, val y test:
X_train
1 1380
0 501
X_val
1 191
0 71
X_test
1 350
0 121

Vemos que se mantiene en cada grupo la proporción casi 3 a 1 del valor 1 con respecto al 0 en la pureza.

iii) Influencia del solapamiento en la independencia de los conjuntos.

El solapamiento puede influir en que en un mismo conjunto (train, val o test) se muestre la misma señal repetida o también, si estuviesen en distintos conjuntos, en val o test tendríamos información del dataset con el que se entrenó el modelo. Las muestras no son completamente independiente debido al solapamiento.

iv) Porcentaje de la base total que se considera en cada conjunto.

Train: 71.96 %
Val: 10.02 %
Test: 18.02 %

v) Tener en cuenta la influencia de los diferentes pacientes y sesiones. ¿Qué estrategia prioriza? Utilizar todos los pacientes y validar/testear sobre conjuntos definidos teniendo en cuenta esa variable. O tener un conjunto de validación/testeo completamente independiente formado por otro paciente.

Hemos utilizado la primera estrategia, ya que en el data frame final, se utilizan datos de todos los pacientes y sesiones. Creemos que de ésta manera tendríamos a generalizar más el modelo, debido a la mayor variabilidad de los datos.

vi) Fundamente todas las decisiones tomadas.