

---

# SCRIPT PARA EL ANALISIS DE VOLTAJES CEREBRALES

## Table of Contents

Apertura de Archivo CSV y Ajuste de Dimensiones .....	1
Determinacion de Medidas de Dispersion .....	2
Determinacion de Variables Cartesianas .....	2
Determinacion de Polaridades Logicas .....	4
Integracion de Resultados .....	6
Glosario .....	6

## Apertura de Archivo CSV y Ajuste de Dimensiones

Apertura de archivo y asignacion de la variable *"eeg"*.

```
[Num,eeg] = uigetfile('*.csv','Selecciona los voltajes'); tic;
eeg = [eeg Num];
eeg = csvread(eeg);
```

Generacion de variable *"alfa"* de 8 celdas, cada una con los 15,000 voltajes del primer minuto de cada canal.

```
alfa = cell(1,8);
for ii = 1:8
    alfa{ii} = eeg(1:15000, ii);
end
clear ii
```

Ajuste de dimensiones de la variable *"eeg"* para contener los 500 segundos que dura el experimento, es decir, 125,000 voltajes de un registro realizado a 250 Hz.

```
eeg = eeg((end-124999:end),:);
```

Ajuste de canales en una variable *"v"* de 8 celdas, una celda por cada canal. A su vez en cada canal se distribuirán los voltajes en 10 celdas, cada una con 12,500 voltajes, o 50 segundos a 250 Hz.

```
b = 12500;
for ii = 1:8
    for jj = 1:10
        if jj == 1
            bb = 1;
        else
            bb = (b*jj)-b;
        end
        v{ii}{jj} = eeg(bb:(bb+b)-1,ii);
    end
end
clear ('b','bb','eeg','ii','jj');
```

```
fprintf('Importacion de archivo %s y ajuste de dimensiones realizada  
correctamente.\n', Num);  
clear Num;
```

## Determinacion de Medidas de Dispersion

Determinacion de la media aritmetica, asignandola a la variable "*mv*". 8 canales con 10 bloques, cada cual conteniendo la media aritmetica de los 12,500 voltajes de su bloque.

```
for ii = 1:8  
    for jj = 1:10  
        mv{ii}{jj} = mean(v{ii}{jj});  
    end  
end  
clear('ii','jj');
```

Determinacion de la varianza, asignandola a la variable "*vv*". 8 canales con 10 bloques, cada cual conteniendo la varianza de los 12,500 voltajes de su bloque.

```
for ii = 1:8  
    for jj = 1:10  
        vv{ii}{jj} = var(v{ii}{jj});  
    end  
end  
clear('ii','jj');
```

Determinacion de la desviacion estandar, asignandola a la variable "*sv*". 8 canales con 10 bloques, cada cual conteniendo la desviacion estandar de los 12,500 voltajes de su bloque.

```
for ii = 1:8  
    for jj = 1:10  
        sv{ii}{jj} = std(v{ii}{jj});  
    end  
end  
clear('ii','jj');
```

```
fprintf('Medidas de dispersion (media, varianza y desviacion estandar)  
calculadas correctamente.\n');
```

## Determinacion de Variables Cartesianas

Calculo de diferencias de numeros contiguos, asignandolos a la variable "*dv*", en el formato  $x\{y\}\{z\}$ , en donde "*x*" es la variable; "*y*" el canal del 1 al 8, y "*z*" el bloque del 1 al 10 cada cual con 12,500 digitos.

```
for ii = 1:8  
    for jj = 1:10  
        dv{ii}{jj} = diff(v{ii}{jj});  
    end  
end  
clear('ii','jj')
```

  

```
for ii = 1:8  
    for jj = 1:10
```

```
        dv{ii}{jj}(12500) = (v{ii}{jj}(12500)) - (v{ii}{jj}(12499));  
    end  
end  
clear('ii','jj')
```

Determinacion de la desviacion estandar de las diferencias de voltaje, asignandolas a la variable "*sdv*" en el formato general *x{y}{z}*.

```
for ii = 1:8  
    for jj = 1:10  
        sdv{ii}{jj} = std(dv{ii}{jj});  
    end  
end  
clear('ii','jj');
```

Determinacion del espacio en el que transcurren las diferencias de voltaje, de  $\pm 3$  desviaciones estandares de la diferencia de sus voltajes, asignandolos a la variable "*fsdv*" en el formato general.

```
for ii = 1:8  
    for jj = 1:10  
        fsdv{ii}{jj} = (6).*(sdv{ii}{jj});  
    end  
end  
clear('ii','jj');
```

Determinacion de la media aritmetica de la diferencia de voltajes, asignada a la variable "*mdv*"

```
mdv = cell(1,8);  
for ii = 1:8  
    mdv{ii} = zeros(1,10);  
end  
clear ii;  
  
for ii = 1:8  
    for jj = 1:10  
        mdv{ii}(jj) = mean(dv{ii}{jj});  
    end  
end  
clear('ii','jj');
```

Determinacion de la potencia angular, partiendo de la distribucion de  $\pm 90$  grados de  $\pm 3$  desviaciones estandares de las diferencias de voltaje, considerando ese rango como el hemicirculo propio de la distribucion anterograda del voltaje a travez del vector del tiempo. La variable "*ang*" contendra en el formato general, la potencia angular calculada como  $\pm 90$  grados entre 6 desviaciones estandares de la diferencia voltaica.

```
for ii = 1:8  
    for jj = 1:10  
        ang{ii}{jj} = (fsdv{ii}{jj}) / 90;  
    end  
end  
clear('ii','jj');
```

Determinacion del vector resultante de la multiplicacion de la diferencia de voltajes y la potencia angular, asignandoles la variable "*vdv*" en el formato general.

```
vdv = cell(1,8);
```

```
for ii = 1:8
    vdv{ii} = cell(1,10);
end
clear('ii','jj');

for ii = 1:8
    for jj = 1:10
        for kk = 1:12500
            vdv{ii}{jj}(kk) = dv{ii}{jj}(kk) .* (ang{ii}{jj});
        end
    end
end
clear('ii','jj');

fprintf('Variables cartesianas determinadas correctamente.\n')
```

## Determinacion de Polaridades Logicas

Determinacion de valores contiguos superiores a 0, asignandolos a la variable "*loa*" en el formato general.

```
for ii = 1:8
    for jj = 1:10
        ploa{ii}{jj} = vdv{ii}{jj} > 0;
        ploa{ii}{jj}(1) = 0;
    end
end
clear('ii','jj')

loa = cell(1,8);
for ii = 1:8
    loa{ii} = cell(1,10);
end
clear('ii','jj');

for ii = 1:8
    for jj = 1:10
        for kk = 1:12500
            if ploa{ii}{jj}(kk) == 1
                loa{ii}{jj}(kk) = ploa{ii}{jj}(kk) + loa{ii}{jj}(kk-1);
            else
                loa{ii}{jj}(kk) = 0;
            end
        end
    end
end
clear('ii','jj','kk','ploa')
```

Determinacion de valores contiguos inferiores a 0, asignandolos a la variable "*lod*" en el formato general.

```
for ii = 1:8
    for jj = 1:10
        plod{ii}{jj} = vdv{ii}{jj} < 0;
        plod{ii}{jj}(1) = 0;
    end
end
```

```
end
clear('ii','jj')

lod = cell(1,8);
for ii = 1:8
    lod{ii} = cell(1,10);
end
clear('ii','jj');

for ii = 1:8
    for jj = 1:10
        for kk = 1:12500
            if plod{ii}{jj}(kk) == 1
                lod{ii}{jj}(kk) = plod{ii}{jj}(kk) + lod{ii}{jj}(kk-1);
            else
                lod{ii}{jj}(kk) = 0;
            end
        end
    end
end
clear('ii','jj','kk','plod')
```

Asignacion de variables **"loam"** y **"lodm"** con los valores maximos de elementos contiguos mayores y menores a 0 respectivamente.

```
loam = cell(1,8);
lodm = cell(1,8);
for ii = 1:8
    loam{ii} = zeros(1,10);
    lodm{ii} = zeros(1,10);
end
clear ii

for ii = 1:8
    for jj = 1:10
        loam{ii}(jj) = max(loa{ii}{jj});
        lodm{ii}(jj) = max(lod{ii}{jj});
    end
end
clear('ii','jj')
```

Asignacion de variable **"loas"** y **"lods"** para la suma de valores contiguos totales, determinando la totalidad de *hemi-ondas* positivas y negativas de donde  $1 = 1/250$  segundos, tomando en cuenta un registro a 250 Hz.

Debido a que para existir valores  $>1$ , previamente deben existir valores  $1 \dots "n-1"$ , cuando  $"n"$  es  $>1$  se restara al valor  $"n-1"$  el valor de  $"n"$ . Para obtener asi los valores de  $"n"$ , propios de  $"n"$  y no necesariamente, los precursores de  $"n+1"$ .

```
loas = cell(1,9);
lods = cell(1,9);
for ii = 1:8
    for jj = 1:10
        for kk = 1:8
            loas{ii}{jj}{kk} = sum(loa{ii}{jj} == kk);
```

```
        lods{ii}{jj}{kk} = sum(lod{ii}{jj} == kk);  
        if kk > 1  
            loas{ii}{jj}{kk - 1} = (loas{ii}{jj}{kk - 1}) - (loas{ii}  
{jj}{kk});  
            lods{ii}{jj}{kk - 1} = (lods{ii}{jj}{kk - 1}) - (lods{ii}  
{jj}{kk});  
        end  
    end  
end  
clear('ii','jj','kk')
```

## Integracion de Resultados

Con el proposito de la visualizacion de los resultados de las distintas variables, en el formato general de resultados, se ha incorporado una celda  $x/9$  que contiene la secuencia concatenada de resultados, en donde  $x$  es una variable previamente determinada y 9 es su posicion.

En la variable "*loas*" y "*lods*", la celda 9 contiene ordenados los resultados en una secuencia en la cual se han agrupado en columnas los numeros de diferencias de voltaje del mismo sentido ( $\pm$ ). y en filas los momentos. siendo la fila del 1 al 10 los 10 momentos de canal 1, la del 11 al 20 los del segundo canal, etc.

```
for ii = 1:8  
    for jj = 1:10  
        for kk = 1:8  
            ll = jj;  
            if ii > 1  
                ll = ll + ((10*ii)-10);  
            end  
            loas{9}(ll, kk) = loas{ii}{jj}{kk};  
        end  
    end  
end  
clear('ii','jj','kk','ll')  
  
for ii = 1:8  
    for jj = 1:10  
        for kk = 1:8  
            ll = jj;  
            if ii > 1  
                ll = ll + ((10*ii)-10);  
            end  
            lods{9}(ll, kk) = lods{ii}{jj}{kk};  
        end  
    end  
end  
clear('ii','jj','kk','ll')
```

## Glosario

Glosario de variables calculadas.

```
fprintf('Glosario:\n')
```

```
fprintf('alfa: valores del primer minuto con ondas alfa provocadas.\n')
fprintf('ang:  valores de potencia angular  $\pm 90(6$  Desviaciones
    estandares de la diferencia de voltajes).\n')
fprintf('dv:   resultado de la diferencia de voltajes contiguos "n-
(n-1)".\n')
fprintf('fsdv: espacio que transcurre entre 6 desviaciones estandares
    ( $\pm 3$ ).\n')
fprintf('loa:  valores logicos cuya diferencia de voltajes es
    ascendente (positiva).\n')
fprintf('loam: numero maximo de valores logicos contiguos que son
    ascendente, de acuerdo a loa.\n')
fprintf('loas: sumatoria de valores logicos ascendentes contiguos
    individuales de acuerdo a loa.\n')
fprintf('lod:  valores logicos cuya diferencia de voltajes es
    descendente (negativa).\n')
fprintf('lodm: numero maximo de valores logicos contiguos que son
    descendentes, de acuerdo a lod.\n')
fprintf('lods: sumatoria de valores logicos descendentes contiguos
    individuales de acuerdo a lod.\n')
fprintf('mv:   media aritmetica de los voltajes.\n')
fprintf('mdv:  media aritmetica de la diferencia de voltajes.\n')
fprintf('sdv:  desviacion estandar de la diferencia de voltajes
    contiguos.\n')
fprintf('sv:   desviacion estandar de los voltajes.\n')
fprintf('v:    voltajes registrados en 8 canales, con 10 bloques cada
    canal (formato general).\n')
fprintf('vdv:  vector resultante de la diferencia de voltajes
    multiplicada por la potencia angular.\n')
fprintf('vv:   varianza de los voltajes.\n')
```

toc % Intentando llevar una noción adecuada de los tiempos.

Código creado por **Miguel Angel Santos Saldivar** un Julio de 2017.

*Published with MATLAB® R2016a*