Datasets y visualización

Importar con readtable y readmatrix

```
%readmatrix
area_mm=readmatrix("../../Utils4SP/Datasets/areaMM.txt",'Delimiter','    ');

%readtable
PSD_bands=readtable("../../Utils4SP/Datasets/2021.10.04_IntensidadBobinas.xlsx");
```

Funcion custom de Import File

```
%Pasa sonda Cassini
%Cassini=importfile_cassini("../../Utils4SP/Datasets/05358_mrdcd_sdfgmc_krtp_1s.asc",88
Cassini=importfile_cassini("../../Utils4SP/Datasets/05358_mrdcd_sdfgmc_krtp_1s.asc",88,
```

Importar audio

```
[buho,fs_buho]=audioread("../../Utils4SP/Datasets/Owl.wav");
%Escuchar
%sound(buho,fs_buho)
```

Datastore

```
% %Le indicamos que una carpeta es un datastore
% ds=datastore("../../Utils4SP/Datasets/AtmosferaLogger_V2/");
% ds.VariableNames=["Fecha" "Hora" "Pres_kpa" "Temp_C" "Hum_perc" "Bat_V"];
% ds.TextscanFormats=["%s" "%s" "%f" "%f" "%f" "%f"];
%
% % ds.Delimiter="";
% % ds.VariableNames=["FechaHora" "Pres_kpa" "Temp_C" "Hum_perc" "Bat_V"];
% % ds.TextscanFormats=["%s" "%f" "%f" "%f" "%f"];
%
% %Leer todas las entradas
% atmosfera_part=readall(ds);
```

Primeras lecturas atmosfericas

```
ds=datastore("../../Utils4SP/Datasets/AtmosferaLogger");
ds.VariableNames=["Fecha" "Hora" "Pres_kpa" "Temp_C" "Hum_perc"];
ds.TextscanFormats=["%s" "%s" "%f" "%f" "%f"];
atmosfera=readall(ds);
```

Reordenando

```
%Sólo si importa fecha y hora en dos columnas
atmosfera.DateTime=string(atmosfera.Fecha)+" "+string(atmosfera.Hora);
```

```
%Pasar de texto a fecha-hora
atmosfera.DateTime=datetime(atmosfera.DateTime,'Format',"yyyyMMdd HH:mm:ss");
%Parche por si la tabla ya se importó
%atmosfera_part.Properties.VariableNames=["Fecha" "Hora" "Pres_kpa" "Temp_C" "Hum_perc"
```

Selección de un subconjunto de datos

Tomaremos los datos entre el 18-May y el 26 de mayo a las 6am

```
%Puntos inicial y final en el tiempo
T1=datetime(2021,5,18,6,0,0)
```

```
T1 = datetime
   210518 06:00:00
```

```
T2=datetime(2021,5,26,6,0,0)
```

```
T2 = datetime
   210526 06:00:00
```

```
%Creando criterios de selección de datos
index= atmosfera.DateTime>T1 &...
    atmosfera.DateTime<T2;

atmosfera_part=atmosfera(index,:);
```

Limpieza de NAN's

```
sum(ismissing(atmosfera_part))
```

```
ans = 1×6
     0     0     0    11    61     0
```

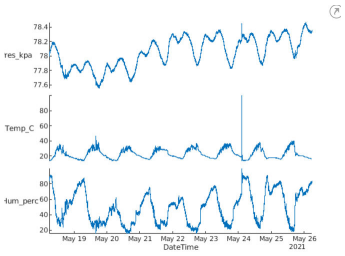
```
summary(atmosfera_part)
```

```
Variables:

    Fecha: 137522×1 cell array of character vectors

    Hora: 137522×1 cell array of character vectors

    Pres_kpa: 137522×1 double

        Values:

            Min        77.55
            Median     78.06
            Max        78.46

    Temp_C: 137522×1 double

        Values:

            Min            13.38
            Median         21.01
```

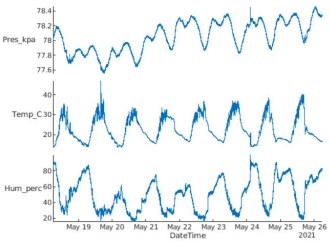
```
        Max             100
        NumMissing      11

    Hum_perc: 137522×1 double

        Values:

        Min             15.27
        Median          50.16
        Max             99.97
        NumMissing      61

    DateTime: 137522×1 datetime

        Values:

        Min       20210518 06:00:03
        Median    20210522 05:59:48
        Max       20210526 05:59:55
```

```
%Remover filas con NaNs
atmosfera_noNaNs=rmmissing(atmosfera_part);
summary(atmosfera_noNaNs)
```

```
Variables:

    Fecha: 137450×1 cell array of character vectors

    Hora: 137450×1 cell array of character vectors

    Pres_kpa: 137450×1 double

        Values:

        Min        77.55
        Median     78.06
        Max        78.46

    Temp_C: 137450×1 double

        Values:

        Min        13.38
        Median     21.01
        Max        46.86

    Hum_perc: 137450×1 double

        Values:

        Min         15.27
        Median     50.155
        Max         99.97

    DateTime: 137450×1 datetime

        Values:

        Min       20210518 06:00:03
        Median    20210522 05:57:17
        Max       20210526 05:59:55
```



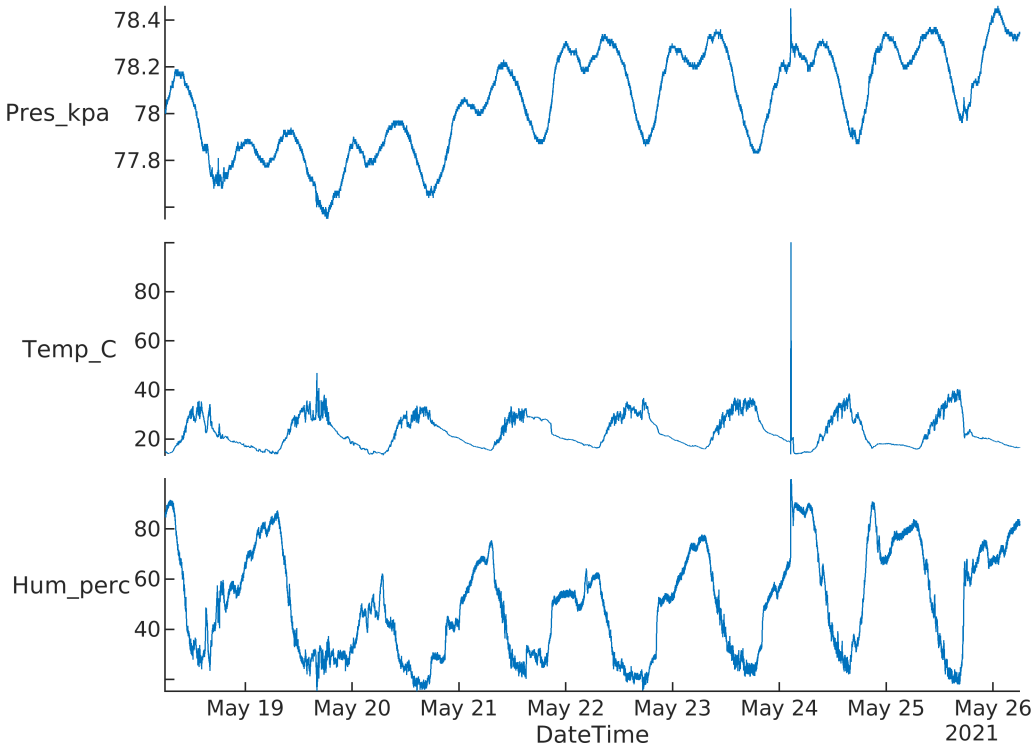
Con NaNs



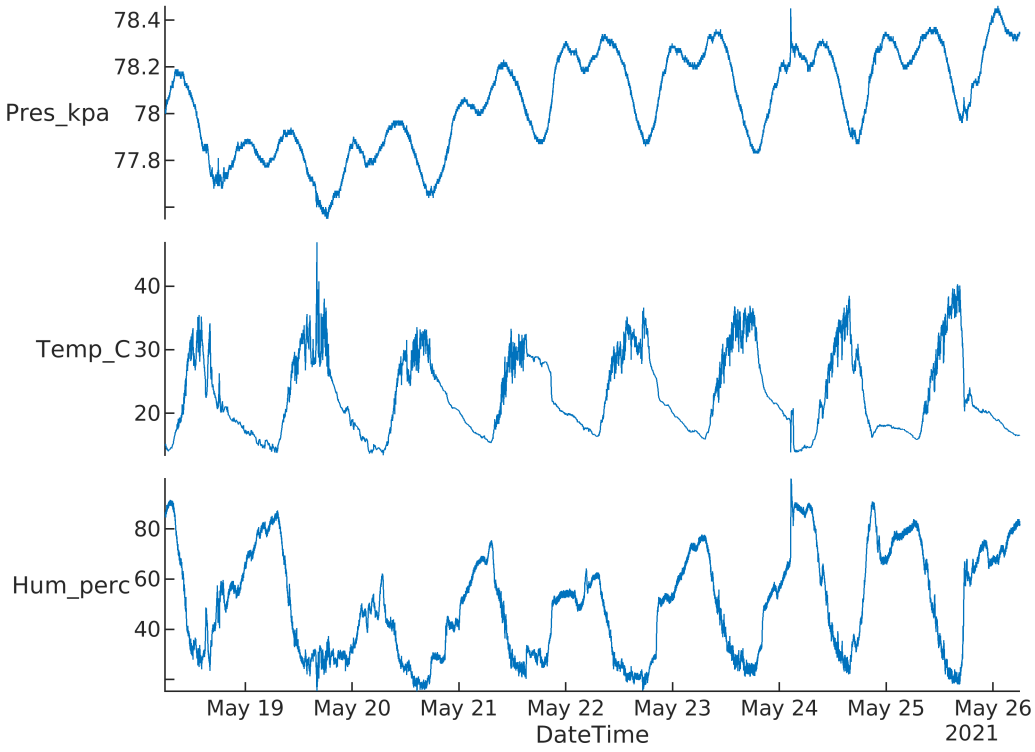
Sin NaNs

Ploteo exploratorio

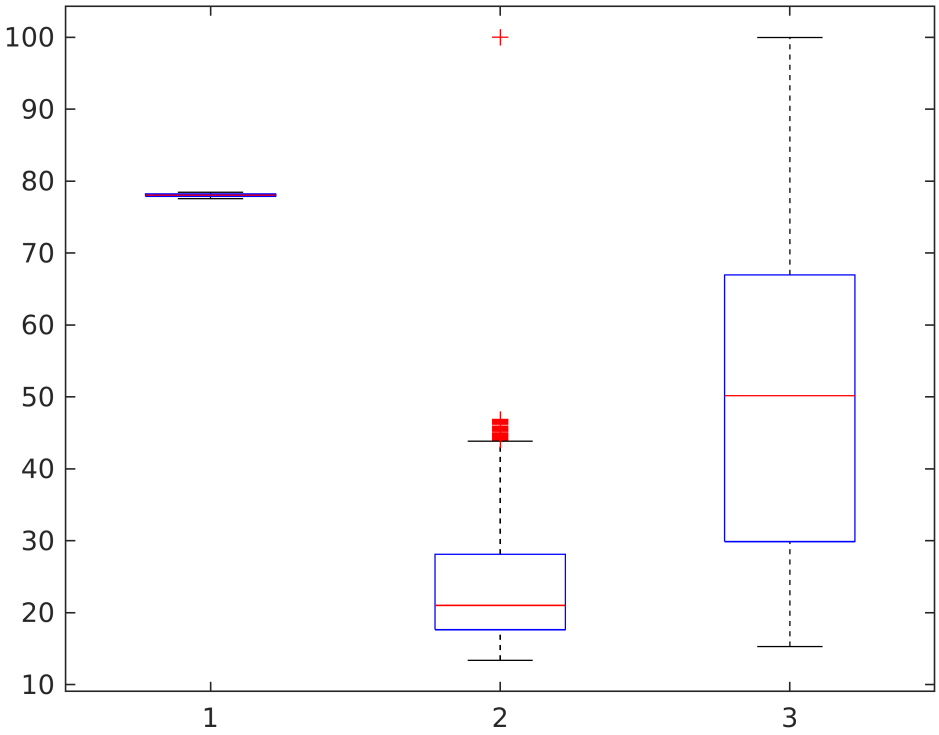
```
figure
stackedplot(atmosfera_part,'XVariable','DateTime')
```




```
figure
stackedplot(atmosfera_noNaNs,'XVariable','DateTime')
```

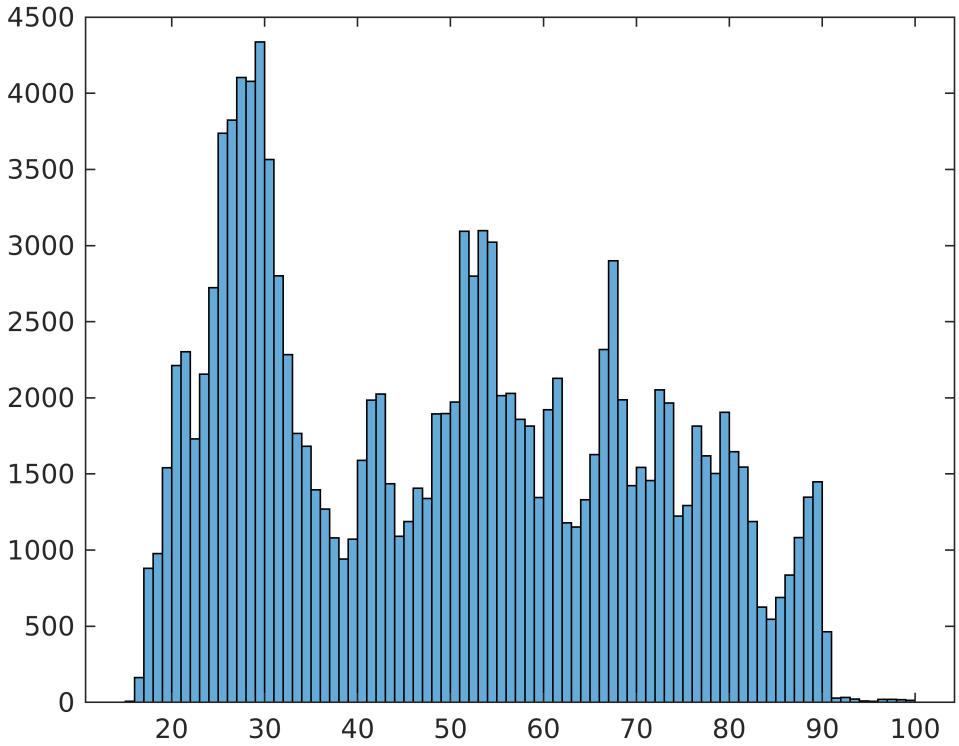



```
figure
boxplot(atmosfera_part{:,["Pres_kpa" "Temp_C" "Hum_perc"]})
```

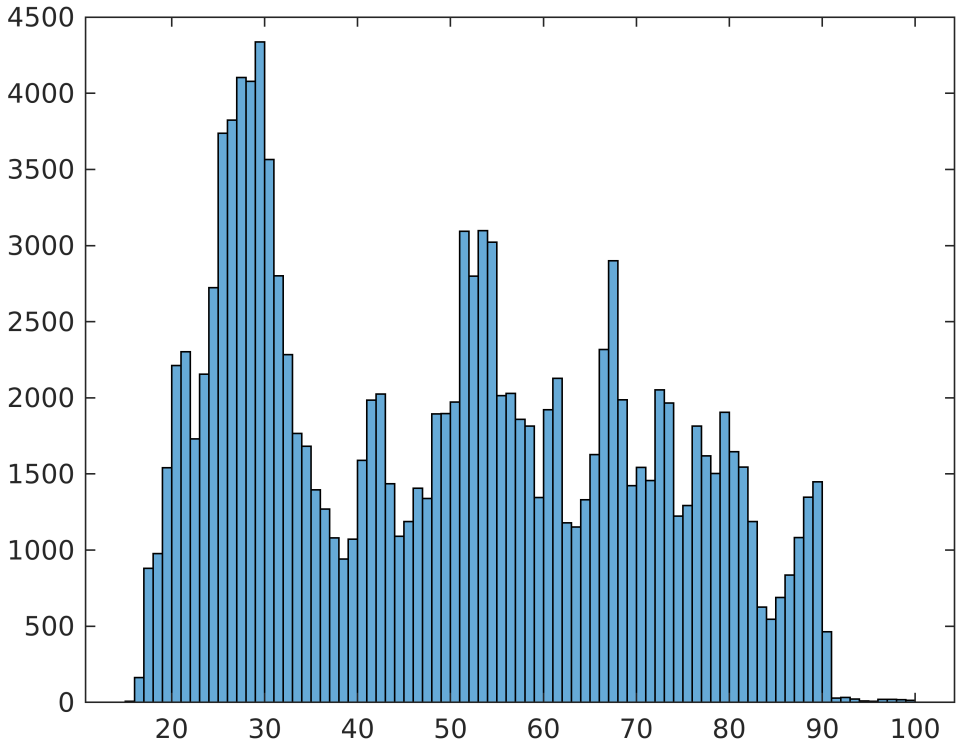


Histograma

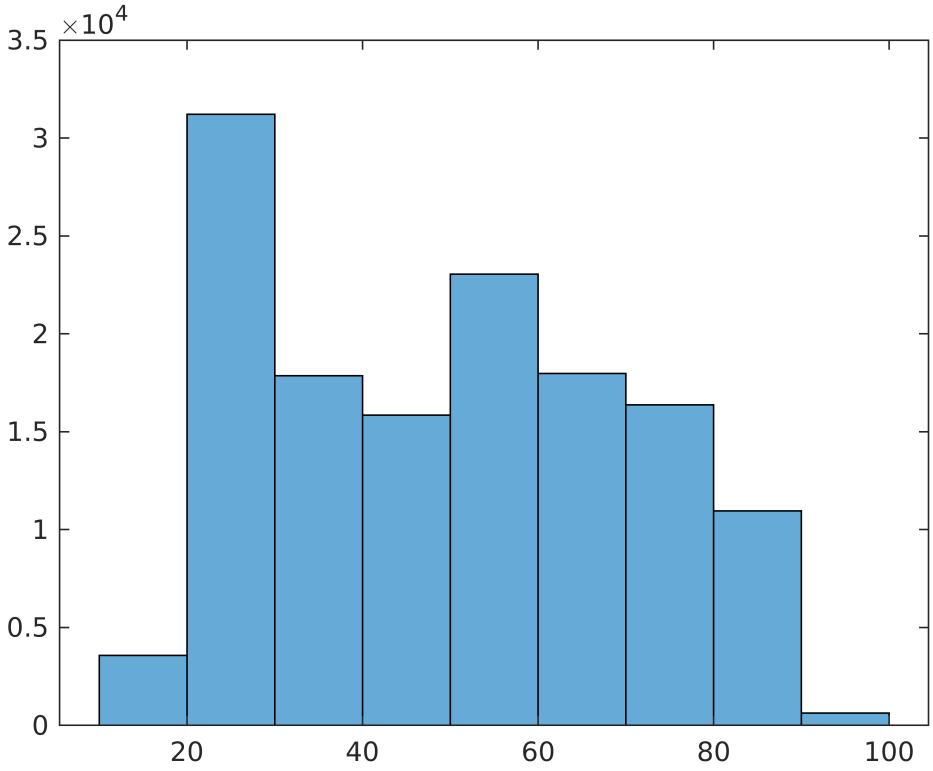
```
figure
histogram(atmosfera_part.Hum_perc)
```




```
figure
histogram(atmosfera_part.Hum_perc,'BinWidth',1)
```

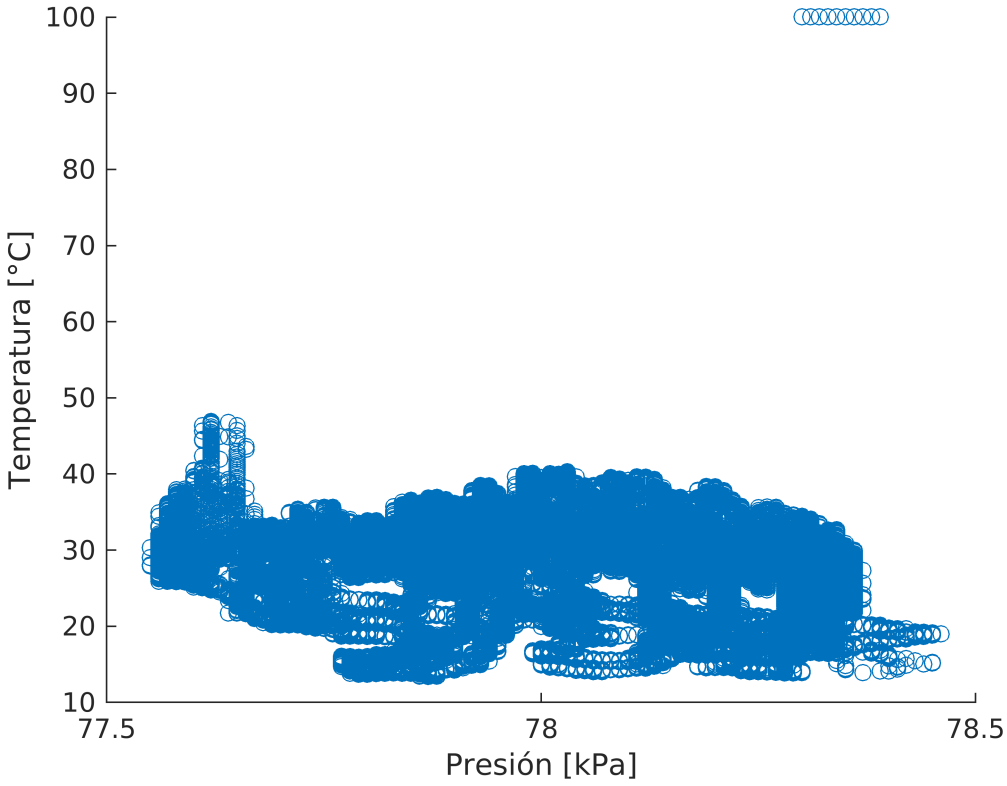



```
figure
histogram(atmosfera_part.Hum_perc,'BinWidth',10)
```

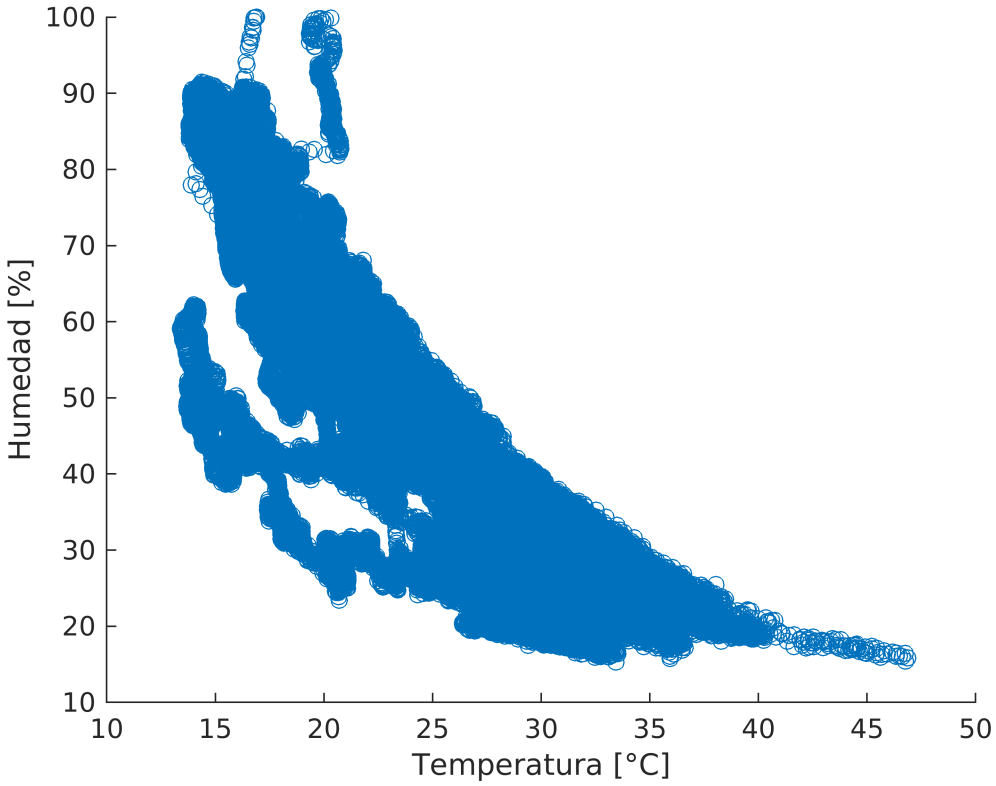


Dispersión

```
figure
scatter(atmosfera_part.Pres_kpa,atmosfera_part.Temp_C)
xlabel("Presión [kPa]")
ylabel("Temperatura [°C]")
```

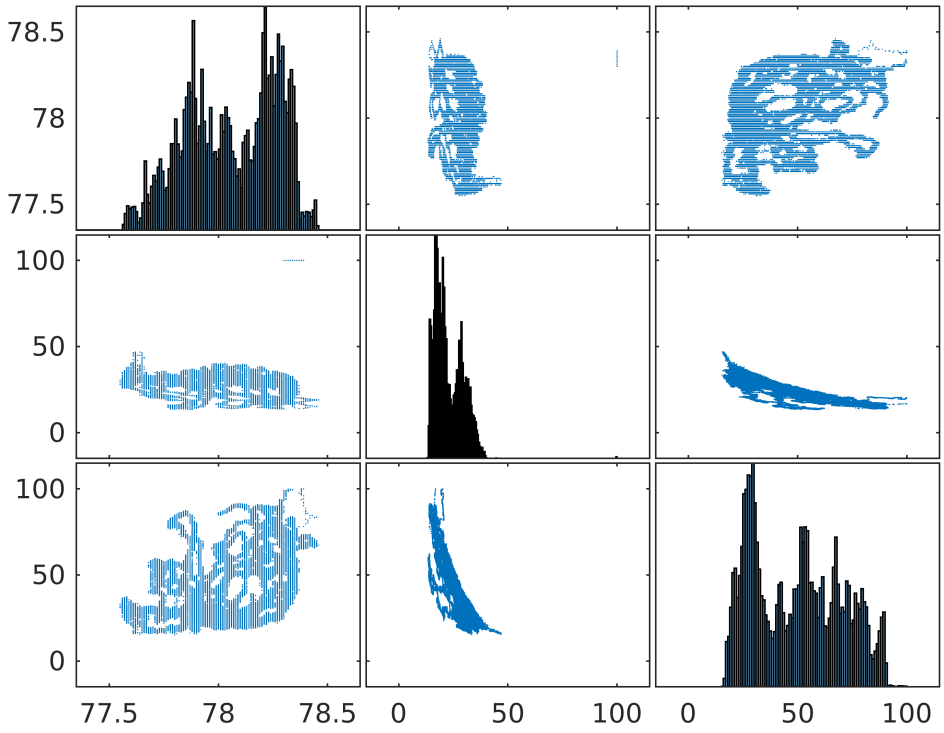



```
figure
scatter(atmosfera_part.Temp_C,atmosfera_part.Hum_perc)
xlabel("Temperatura [°C]")
ylabel("Humedad [%]")
```



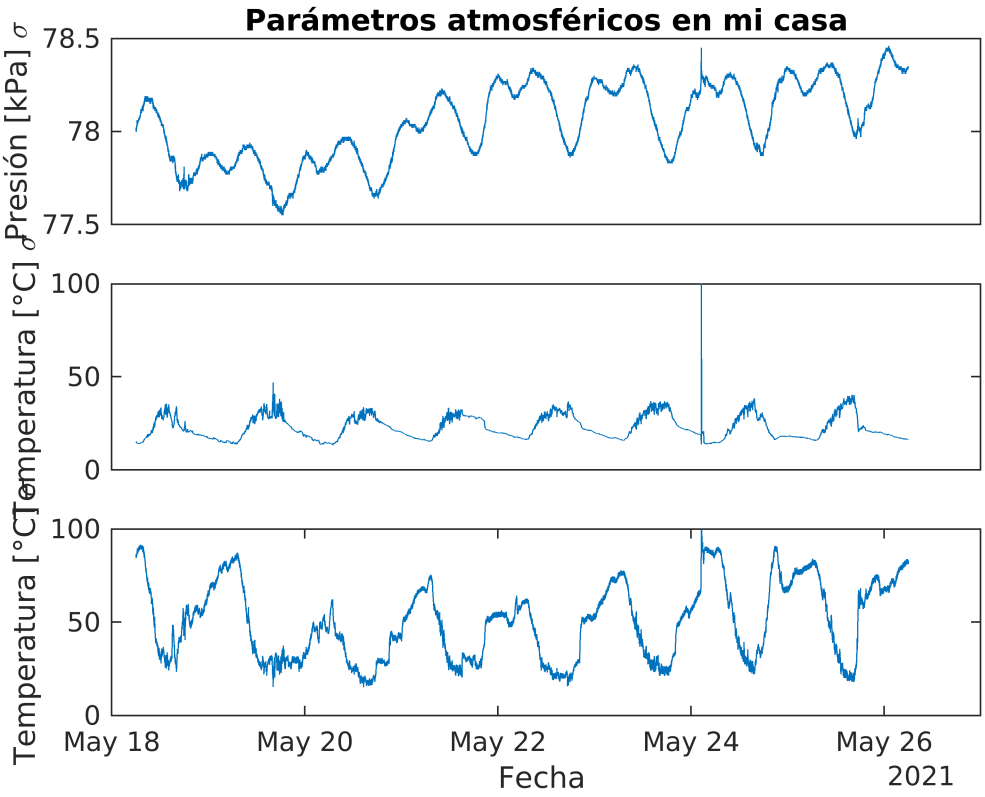
Plot matrix

```
figure
plotmatrix(atmosfera_part{:,["Pres_kpa" "Temp_C" "Hum_perc"]})
```



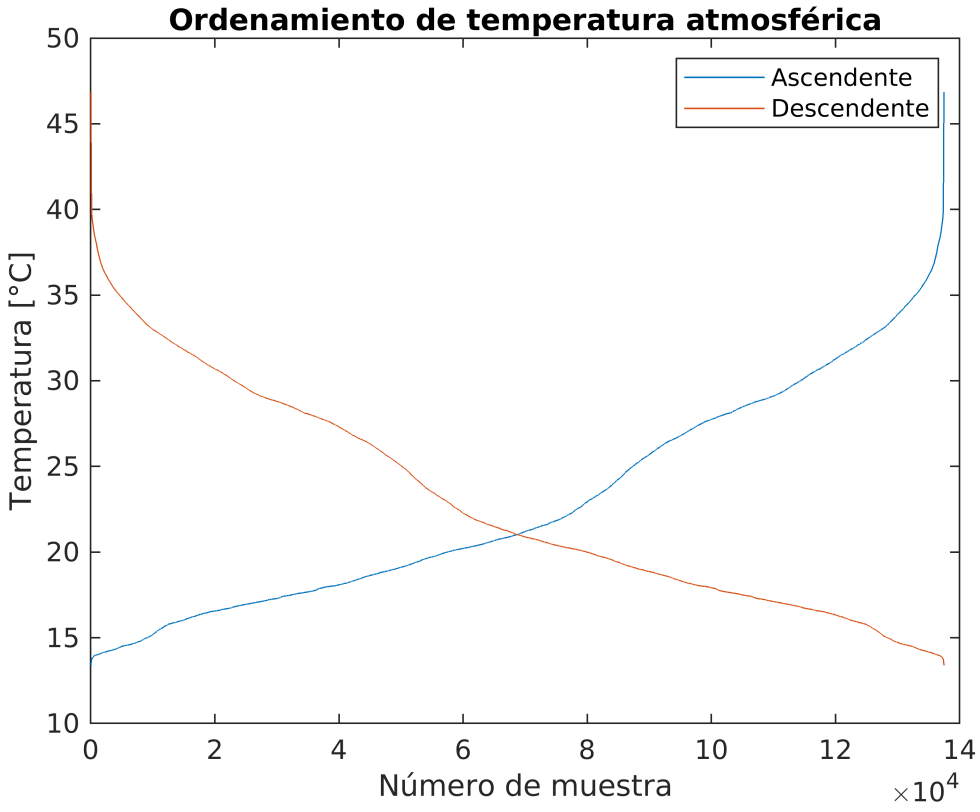
Tiledlayout

```
%tiledlayout("flow")
tiledlayout(3,1)
nexttile
plot(atmosfera_part.DateTime,atmosfera_part.Pres_kpa)
ylabel("Presión [kPa] \sigma")
%xlabel("Fecha ")
title("Parámetros atmosféricos en mi casa")
set(gca,'xtick',[])

nexttile
plot(atmosfera_part.DateTime,atmosfera_part.Temp_C)
ylabel("Temperatura [°C] \sigma")
%xlabel("Fecha ")
set(gca,'xtick',[])

nexttile
plot(atmosfera_part.DateTime,atmosfera_part.Hum_perc)
ylabel("Temperatura [°C] \sigma")
xlabel("Fecha ")
```



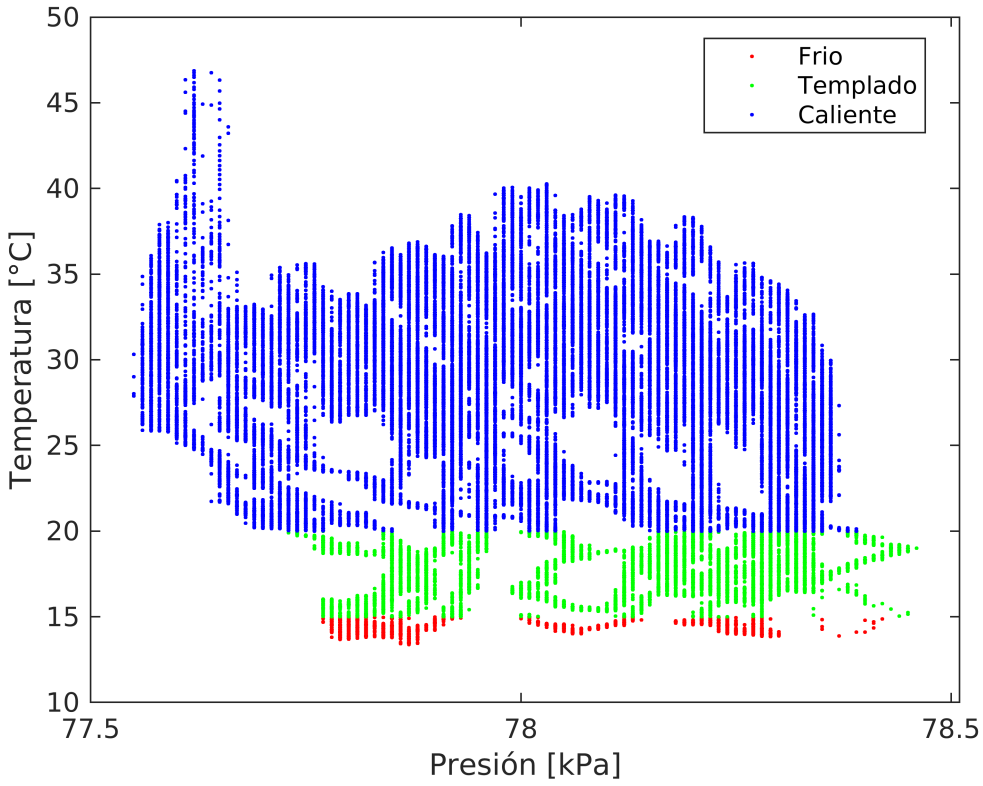
Agrupación y orden

```
% Darle categorías a temperatura
%Fronteras de las categorías
edges=[-30 15 20 55];

%Categorías
categorias=["Frio" "Templado" "Caliente"];

%Clasifico las temperatura con un categórico
temp_cats=discretize(atmosfera_noNaNs.Temp_C,edges,'categorical',categorias);

%La agrego a tabla original
atmosfera_noNaNs.temp_cats=temp_cats;

%Ordenamiento
temp_Descend=sort(atmosfera_noNaNs.Temp_C,'descend');
temp_Ascend=sort(atmosfera_noNaNs.Temp_C,'ascend');

figure
plot([temp_Ascend temp_Descend])
ylabel("Temperatura [°C]")
xlabel("Número de muestra")
legend("Ascendente","Descendente")
title("Ordenamiento de temperatura atmosférica")
```

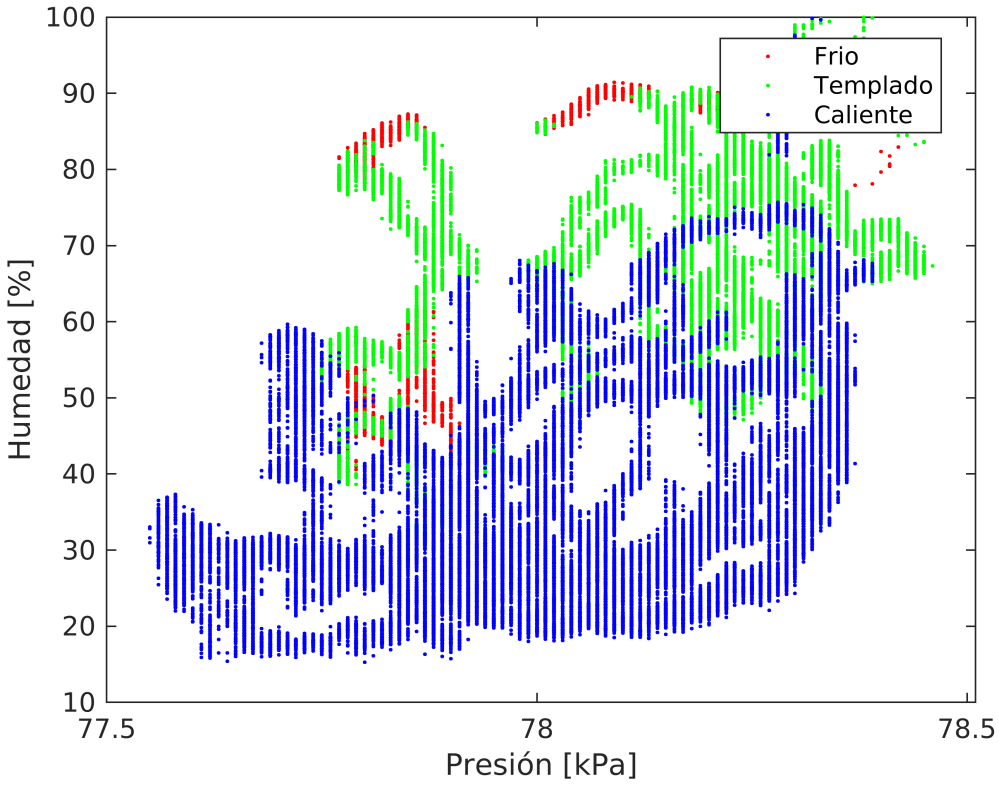


G scatter y pareto chart

```
figure
gscatter(atmosfera_noNaNs.Pres_kpa,atmosfera_noNaNs.Temp_C,atmosfera_noNaNs.temp_cats)
xlabel("Presión [kPa]")
ylabel("Temperatura [°C]")
```

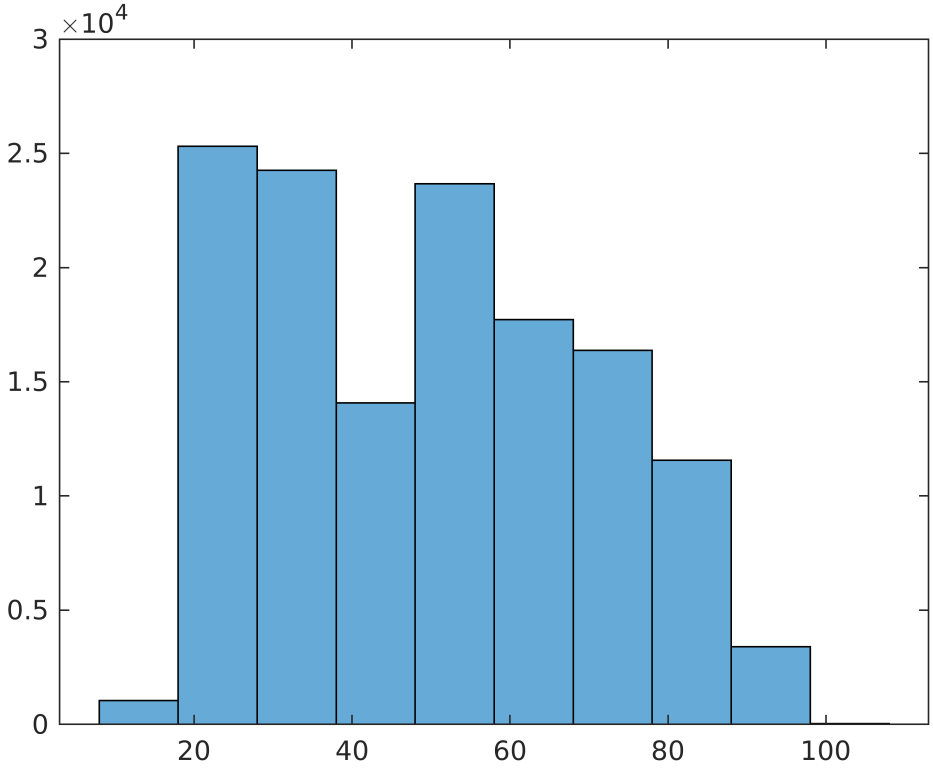



```
figure
gscatter(atmosfera_noNaNs.Pres_kpa,atmosfera_noNaNs.Hum_perc,atmosfera_noNaNs.temp_cats
xlabel("Presión [kPa]")
ylabel("Humedad [%]")
```

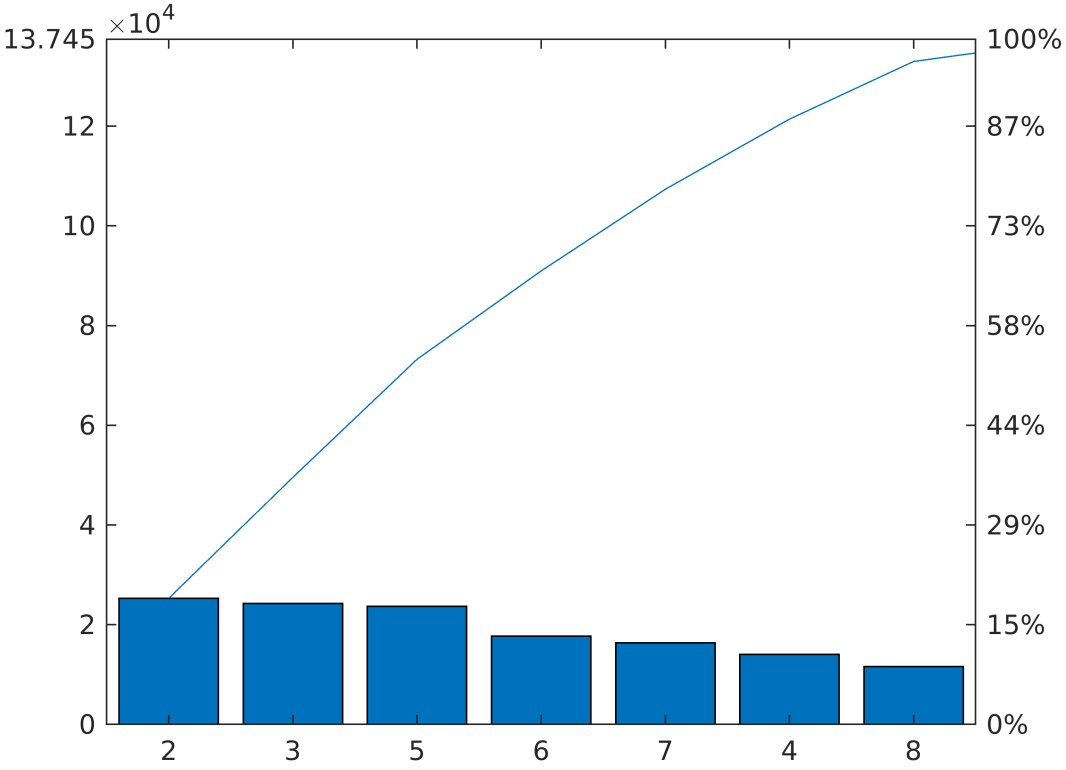


Pareto

```
h=histogram(atmosfera_noNaNs.Hum_perc,10);
```




```
cuentas=h.BinCounts;

figure
pareto(cuentas)
```


13.745
×10^4
12
10
8
6
4
2
0
100%
87%
73%
58%
44%
29%
15%
0%
2
3
5
6
7
4
8