

# Datasets y visualización

## Importar con readtable y readmatrix

```
%Readmatrix
area_mm=readmatrix("../Clase Github/Utils4SP/Datasets/areaMM.txt","Delimiter"," ")
```

```
area_mm = 1×181
    340.4277    324.3750    308.9114    293.6813    277.8073    261.0772    245.9019    232.8705 ...
```

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

```
PSD_bands = 630×13 table
```

	Dist_cm_	PSD_B1	PSD_B2	PSD_B3	PSD_B4	PSD_B5	PSD_B6	PSD_B7
1	0.5000	0.5480	0.0129	0.0052	0.0046	0.0205	1.1388	0.0107
2	0.5000	0.5042	0.0017	0.0022	0.0024	0.0057	0.5872	0.0015
3	0.5000	0.4539	0.0061	0.0016	0.0049	0.0121	0.5663	0.0060
4	0.5000	0.3205	0.0055	0.0052	0.0017	0.0082	0.6285	0.0053
5	0.5000	0.3859	0.0032	0.0026	0.0024	0.0066	0.5744	0.0035
6	0.5000	0.8591	0.0051	0.0020	0.0019	0.0073	0.6579	0.0061
7	0.5000	0.5021	0.0044	0.0020	0.0018	0.0075	0.5880	0.0027
8	0.5000	0.3402	0.0063	0.0067	0.0062	0.0097	0.5597	0.0118
9	0.5000	0.3661	0.0046	0.0020	0.0049	0.0085	0.5586	0.0043
10	0.5000	0.5464	0.0037	0.0016	0.0024	0.0075	0.5924	0.0011
11	0.5000	0.8342	0.0094	0.0025	0.0071	0.0063	0.9233	0.0042
12	0.5000	0.3870	0.0050	0.0017	0.0033	0.0124	1.1173	0.0029
13	0.5000	0.4425	0.0021	0.0005	0.0013	0.0038	0.6084	0.0020
14	0.5000	0.4923	0.0034	0.0009	0.0022	0.0069	0.5943	0.0012

⋮

## Function custom de Import File

```
%Pasa sonda Cassini
Cassini=importfile_cassini("G:\Clase Github\Utils4SP\Datasets\05358_mrdcd_sdfgmc_krtp_1s.asc")
```

## Importar audio

```
[y,fs]=audioread("G:\Clase Github\Utils4SP\Datasets\Owl.wav");
%Escuchar
sound(y,fs)
```

## Datastore

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

## Reordenando

```
%Solo 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.Properties.VariableNames=["Fecha" "Hora" "Pres_kpa" "Temp_C" "Hum_perc" "Bat_V"];
```

## Limpieza

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
    18-May-2021 06:00:00
```

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

```
T2 = datetime
    26-May-2021 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 = 1x6
    0     0     0    11    61     0
```

```
summary(atmosfera_part)
```

Variables:

Fecha: 137522x1 cell array of character vectors

Hora: 137522x1 cell array of character vectors

Pres\_kpa: 137522x1 double

Values:

Min	77.55
Median	78.06
Max	78.46

Temp\_C: 137522x1 double

Values:

Min	13.38
Median	21.01
Max	100
NumMissing	11

Hum\_perc: 137522x1 double

Values:

Min	15.27
Median	50.16
Max	99.97
NumMissing	61

DateTime: 137522x1 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: 137450x1 cell array of character vectors

Hora: 137450x1 cell array of character vectors

Pres\_kpa: 137450x1 double

Values:

Min	77.55
Median	78.06
Max	78.46

Temp\_C: 137450x1 double

Values:

Min	13.38
Median	21.01
Max	46.86

Hum\_perc: 137450x1 double

Values:

Min	15.27
-----	-------

Median	50.155
Max	99.97

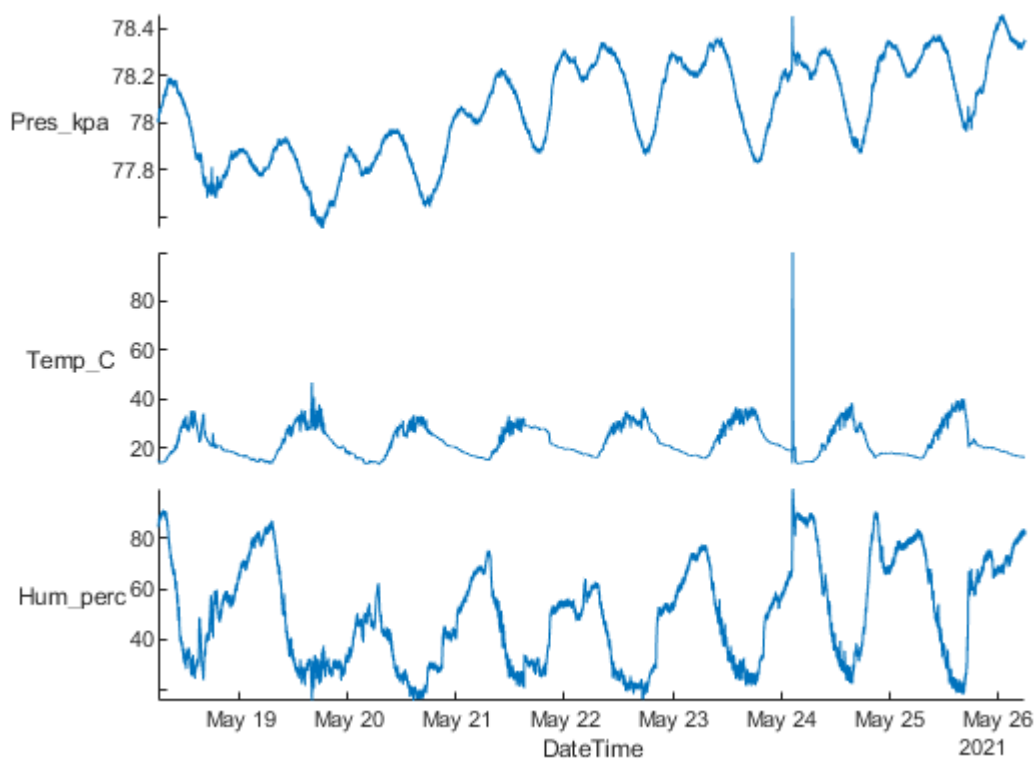
DateTime: 137450×1 datetime

Values:

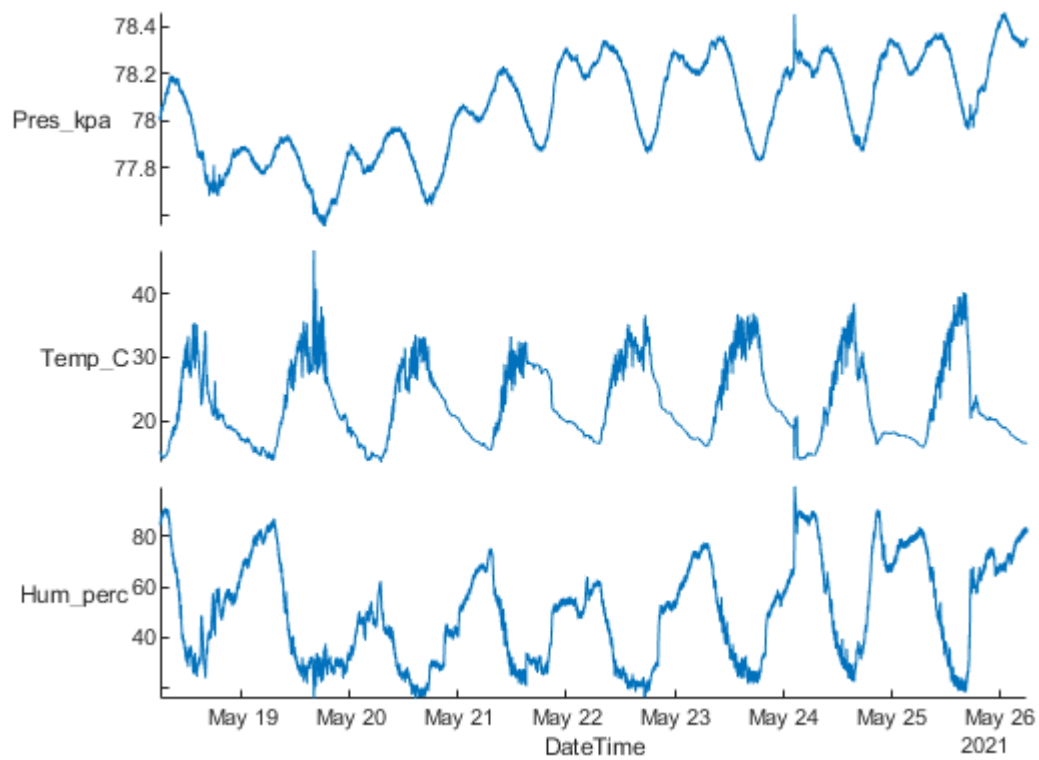
Min	20210518 06:00:03
Median	20210522 05:57:17
Max	20210526 05:59:55

## Ploteo exploratorio

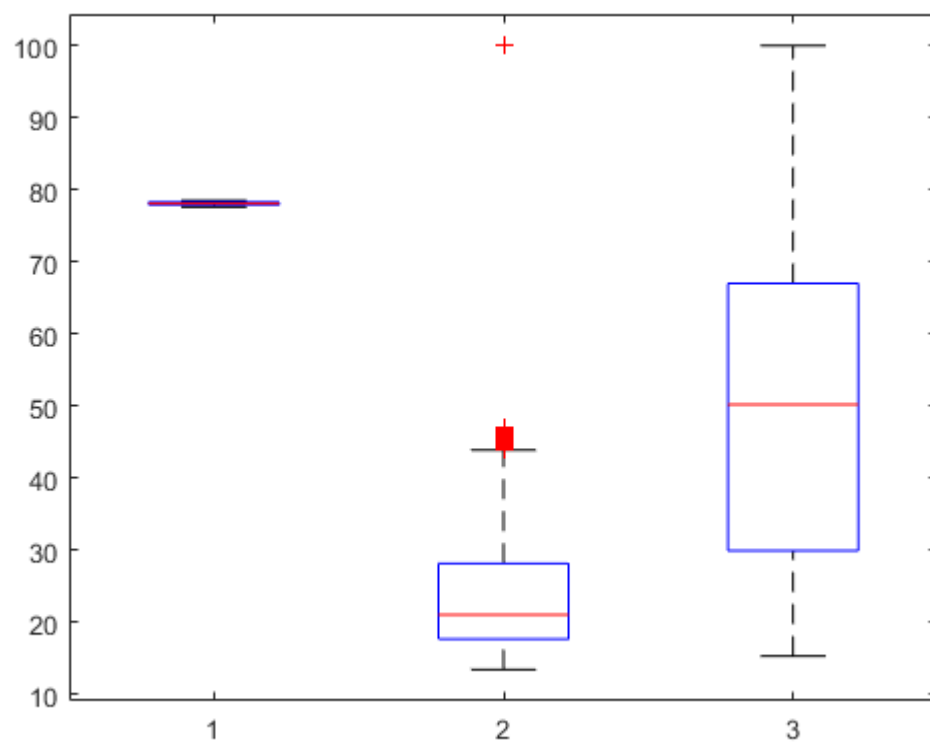
```
figure  
stackedplot(atmosfera_part,"XVariable","DateTime")
```



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

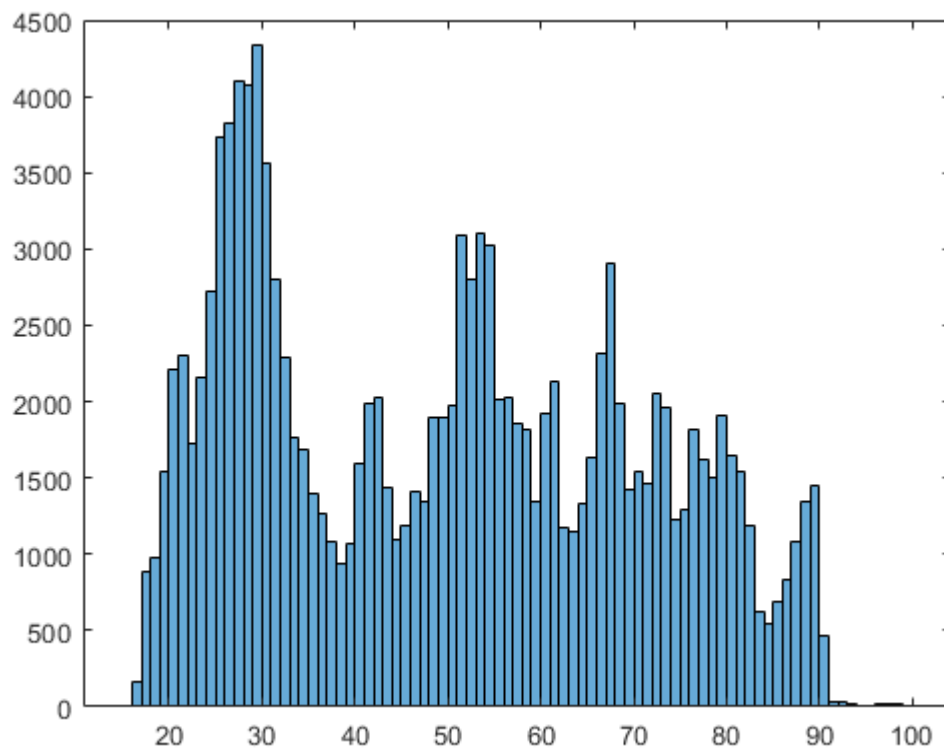


```
figure
boxplot(atmosfera_part{:[,"Pres_kpa" "Temp_C" "Hum_perc"]})
```

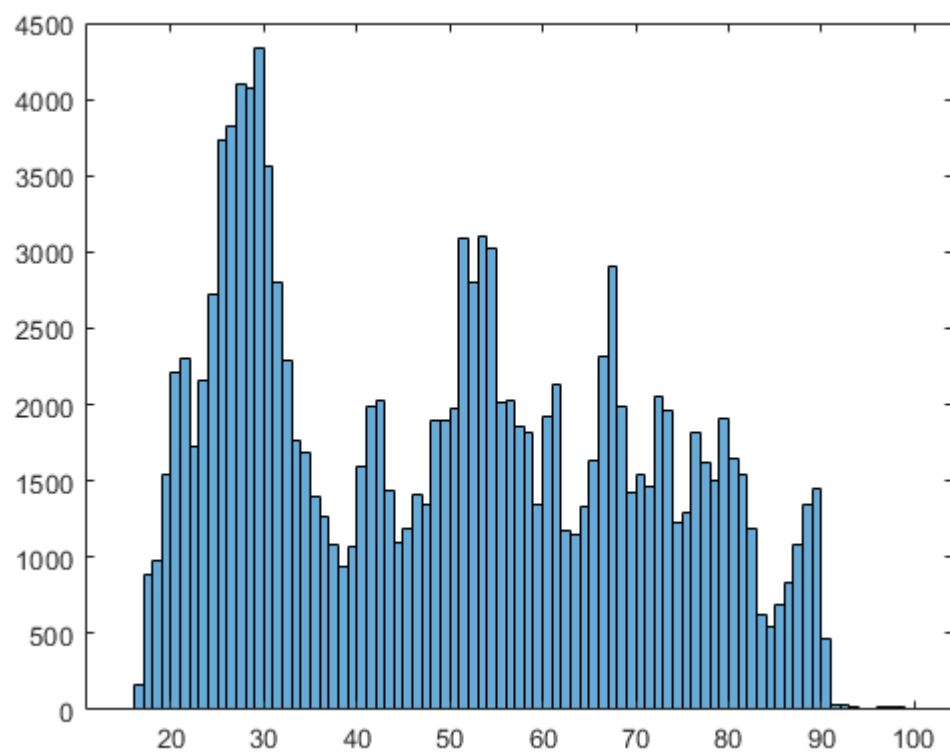


## Histograma

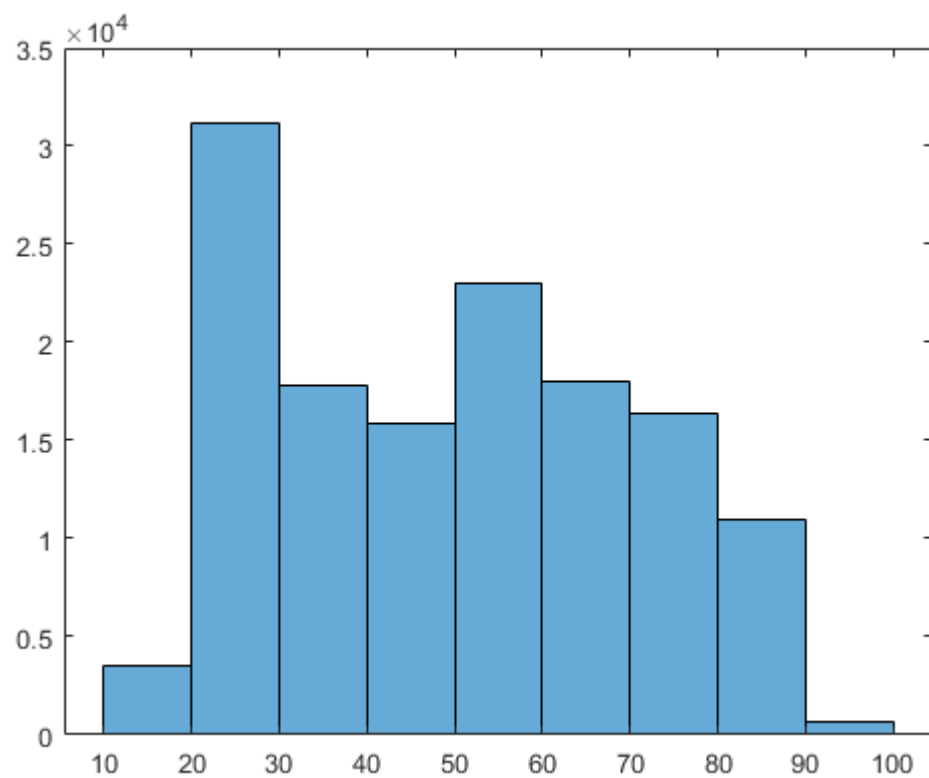
```
figure  
histogram(atmosfera_part.Hum_perc)
```



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

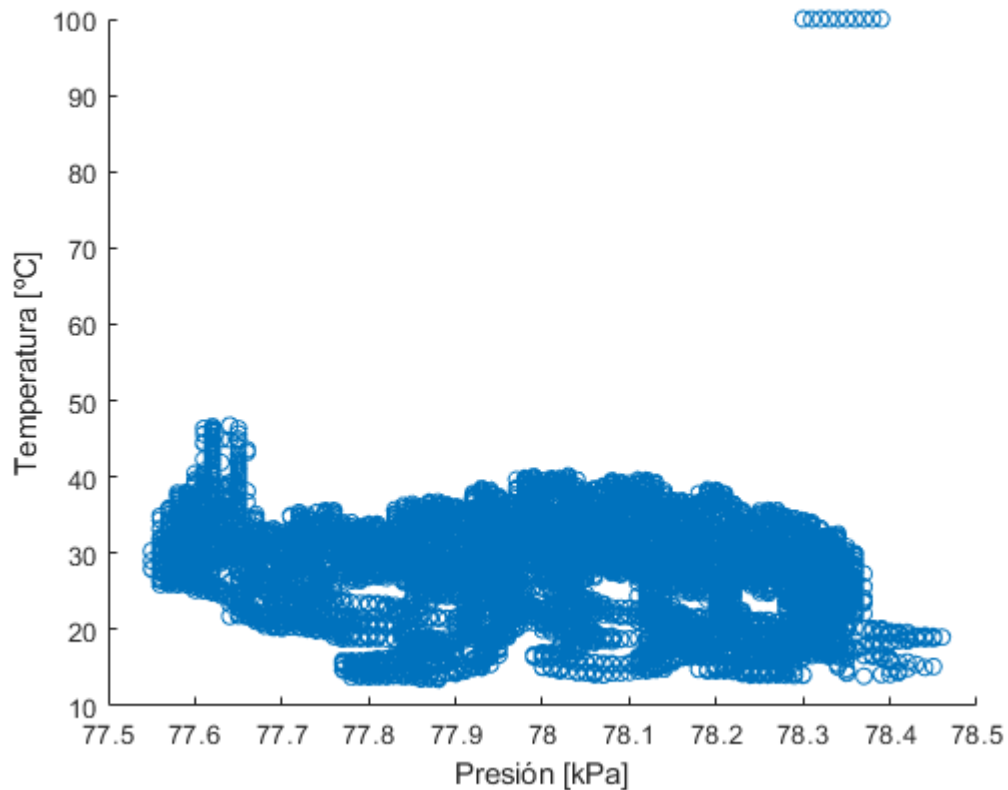


```
figure
histogram(atmosfera_part.Hum_perc, "BinWidth", 10)
```



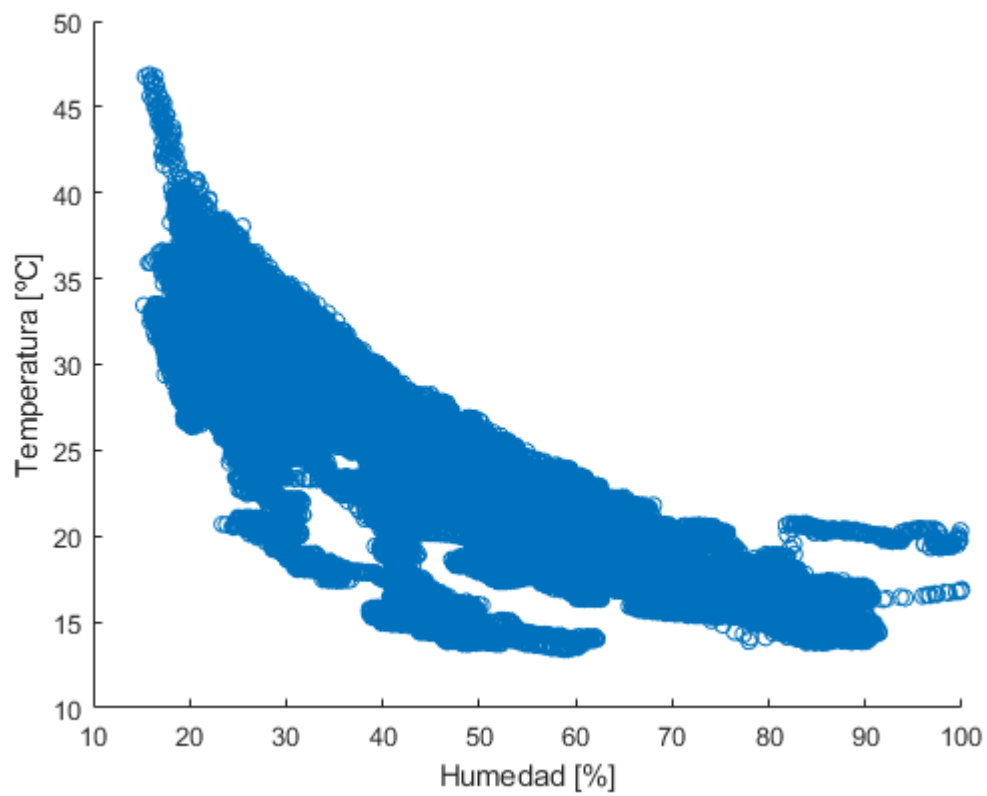
## Dispersión

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



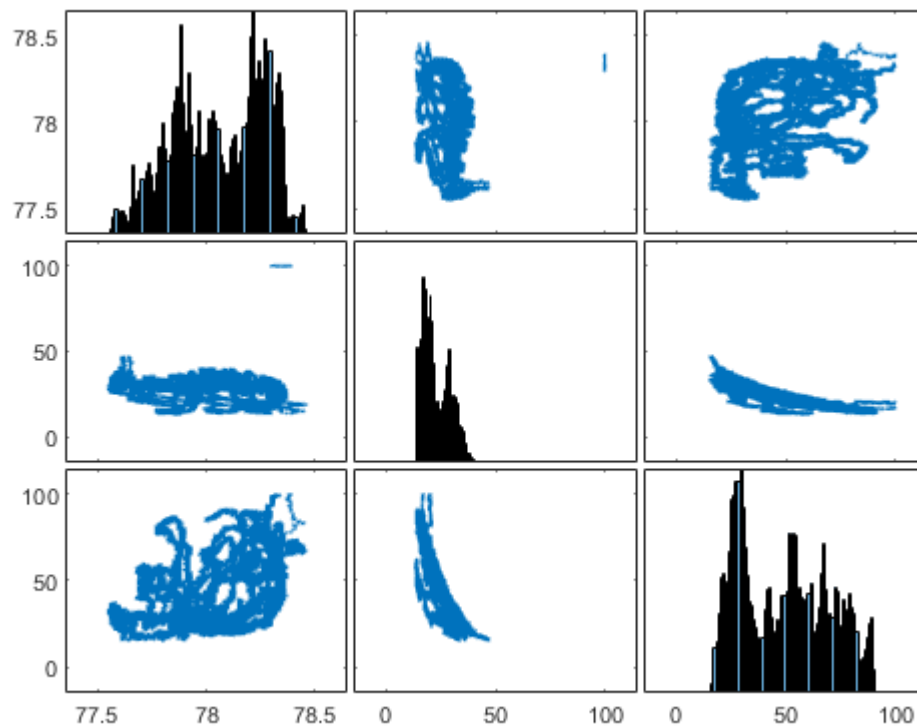
```
scatter(atmosfera_part.Hum_perc,atmosfera_part.Temp_C)
xlabel("Humedad [%]")
ylabel("Temperatura [°C]")
```





## Plot Matrix

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

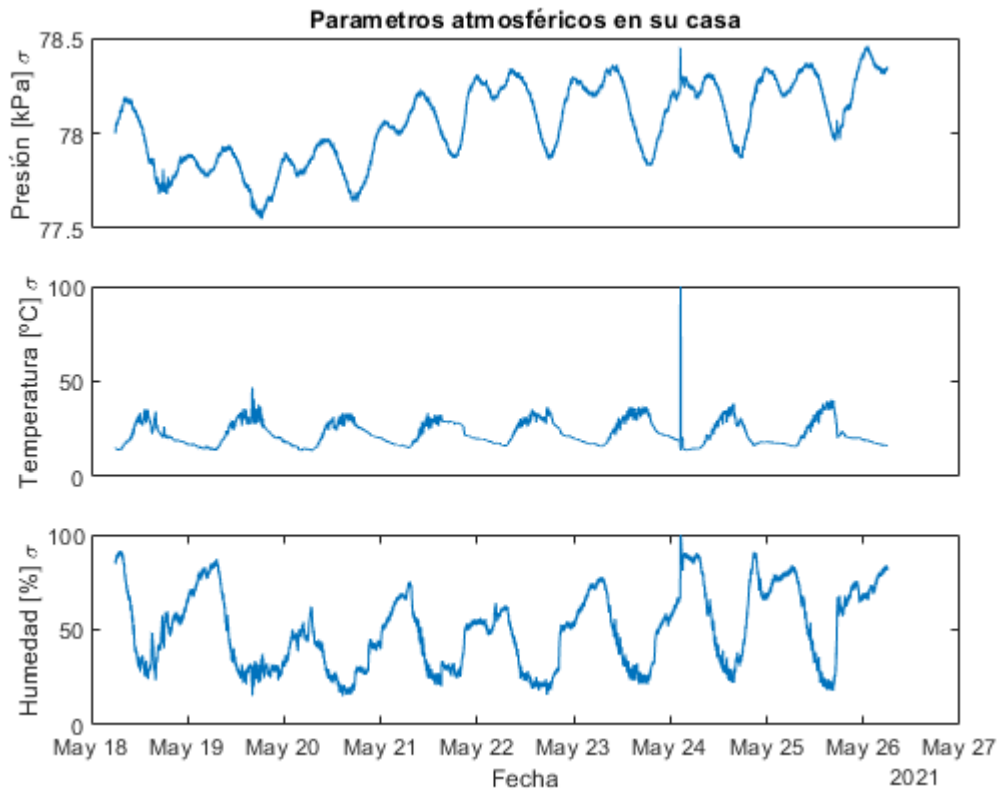


## Tiledlayout

```

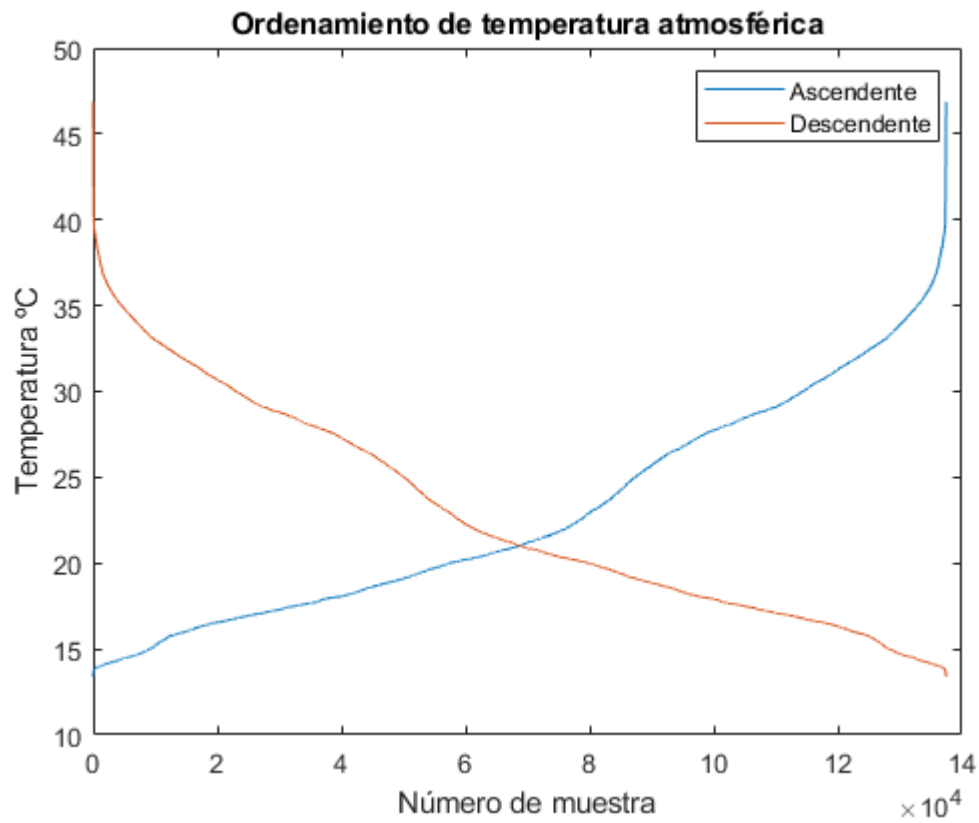
tiledlayout(3,1)
nexttile
plot(atmosfera_part.DateTime,atmosfera_part.Pres_kpa)
ylabel("Presión [kPa] \sigma")
%xlabel("Fecha")
title("Parametros atmosféricos en su casa")
set(gca,"xtick",[])
nexttile
plot(atmosfera_part.DateTime,atmosfera_part.Temp_C)
ylabel("Temperatura [°C] \sigma")
%xlabel("Fecha")
set(gca,"xtick",[])
nexttile
plot(atmosfera_part.DateTime,atmosfera_part.Hum_perc)
ylabel("Humedad [%] \sigma")
xlabel("Fecha")

```



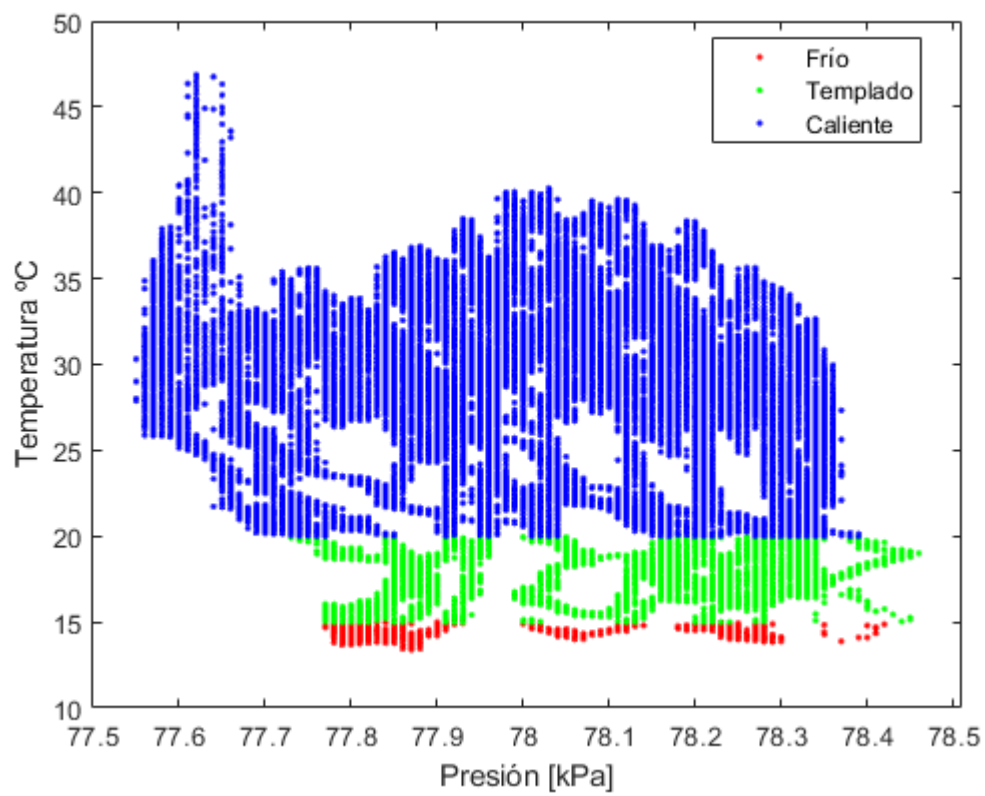
## Agrupación y orden

```
%Dar categorías a temperatura
%Fronteras de las categorías
edges=[-30 15 20 55];
%Categorías
categorias=["Frío" "Templado" "Caliente"];
%Clasifico las temperaturas con un categórico
temp_cats=discretize(atmosfera_noNaNs.Temp_C,edges,"categorical",categorias);
%Agrego la categorización a la tabla
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")
```

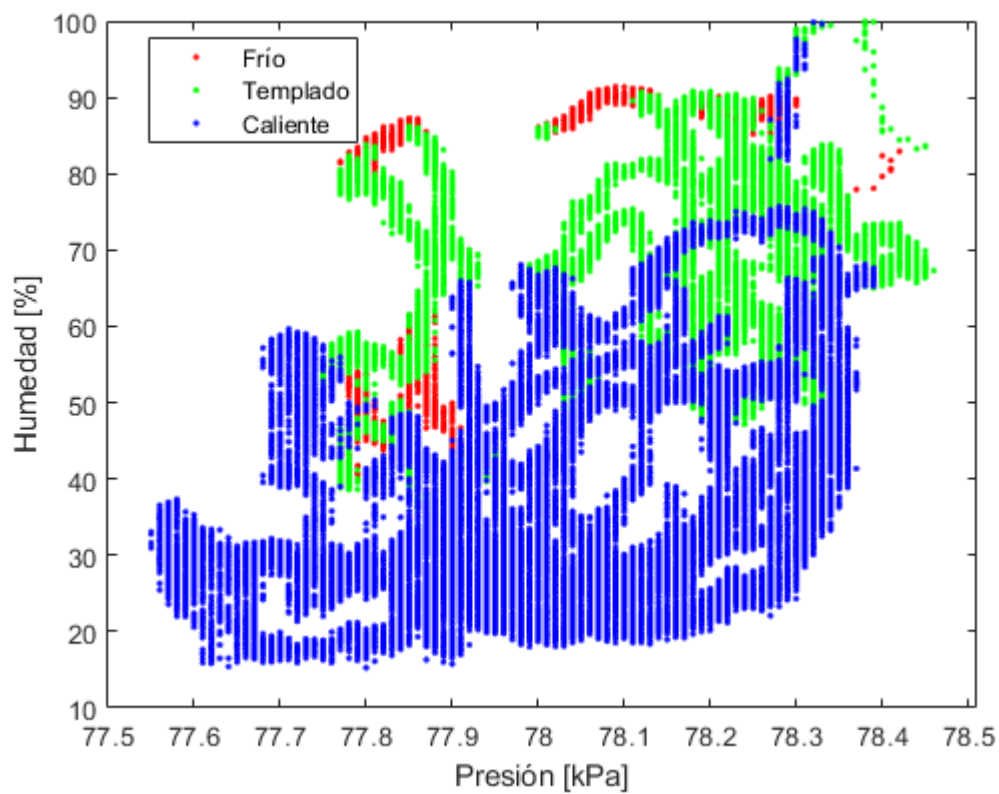


## G scatter

```
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 chart

```
%Hasta la siguiente clase
%figure
%pareto(atmosfera_noNaNs.Hum_perc)
```