

Aula 22

Representação de
dados no tempo

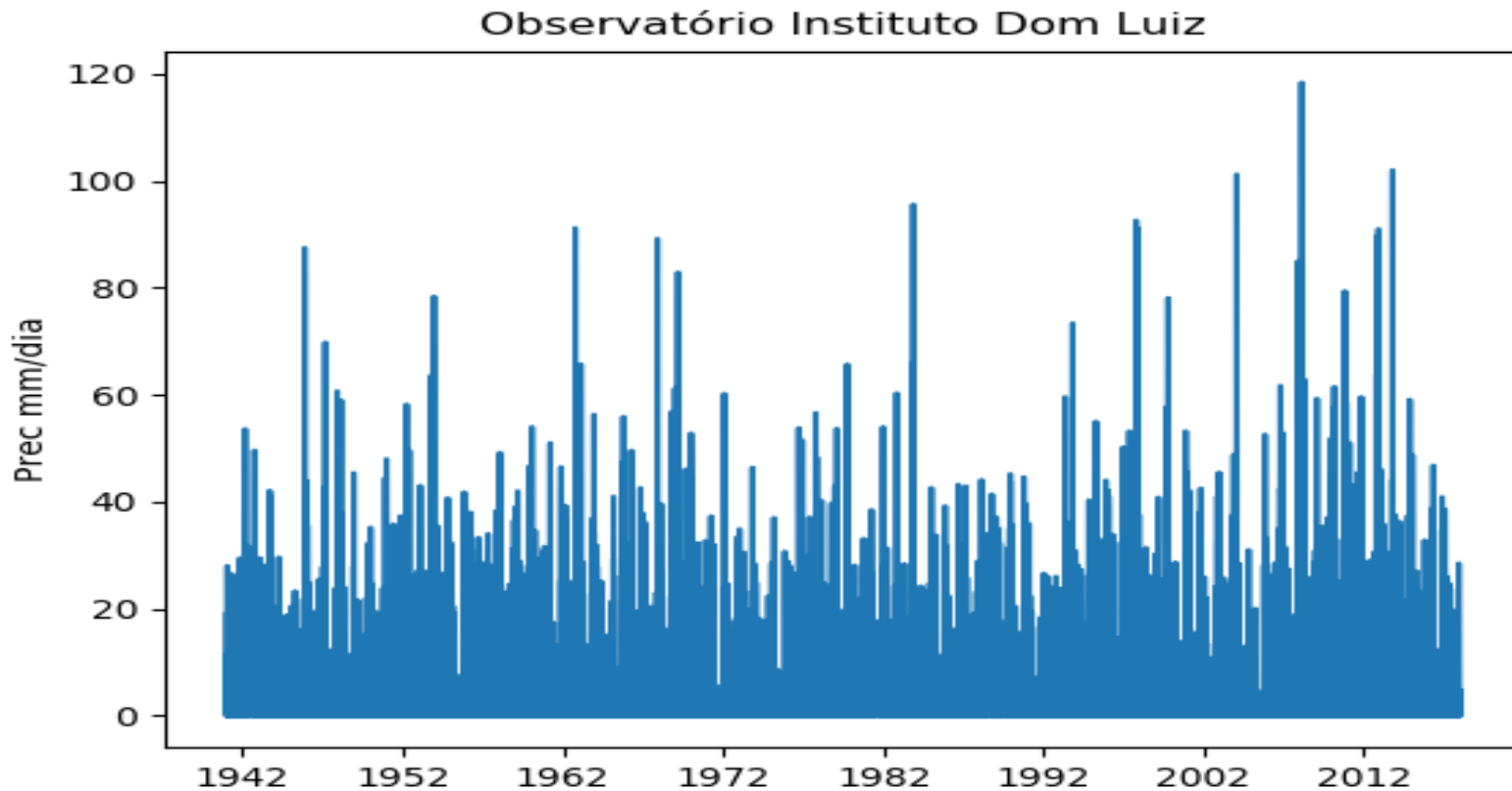
Leitura de série temporal

```
import numpy as np
import datetime
import matplotlib.pyplot as plt
dados=np.loadtxt('prec24h_535_2.dat')
ano=np.array(dados[:,0],dtype=int);
mes=np.array(dados[:,1],dtype=int);
dia=np.array(dados[:,2],dtype=int);
prec=dados[:,3]
del dados
tempo=[]
for kd in range(len(prec)):
    tempo.append(datetime.datetime\
        (ano[kd],mes[kd],dia[kd]))

plt.plot(tempo,prec)
plt.ylabel('Prec mm/dia')
plt.title('Instituto Dom Luiz')
plt.savefig('IDL_Prec_1941_2017.png')
```

| | Ano | Mês | Dia | Prec |
|------|-------------------------|-----|-----|-------------|
| 0000 | * * * Top of File * * * | | | |
| 0001 | 1941 | 1 | 1 | 19.00000000 |
| 0002 | 1941 | 1 | 2 | 9.19999981 |
| 0003 | 1941 | 1 | 3 | 0.00000000 |
| 0004 | 1941 | 1 | 4 | 0.00000000 |
| 0005 | 1941 | 1 | 5 | 0.00000000 |
| 0006 | 1941 | 1 | 6 | 0.00000000 |
| 0007 | 1941 | 1 | 7 | 0.00000000 |
| 0008 | 1941 | 1 | 8 | 11.50000000 |
| 0009 | 1941 | 1 | 9 | 0.00000000 |
| 0010 | 1941 | 1 | 10 | 5.69999981 |
| 0011 | 1941 | 1 | 11 | 2.09999990 |
| 0012 | 1941 | 1 | 12 | 4.90000010 |
| 0013 | 1941 | 1 | 13 | 9.39999962 |
| 0014 | 1941 | 1 | 14 | 0.400000006 |
| 0015 | 1941 | 1 | 15 | 2.59999990 |
| 0016 | 1941 | 1 | 16 | 5.50000000 |
| 0017 | 1941 | 1 | 17 | 1.20000005 |
| 0018 | 1941 | 1 | 18 | 0.100000001 |
| 0019 | 1941 | 1 | 19 | 5.40000010 |
| 0020 | 1941 | 1 | 20 | 11.3999996 |
| 0021 | 1941 | 1 | 21 | 28.0000000 |
| 0022 | 1941 | 1 | 22 | 25.6000004 |
| 0023 | 1941 | 1 | 23 | 9.30000019 |
| 0024 | 1941 | 1 | 24 | 17.5000000 |
| 0025 | 1941 | 1 | 25 | 0.00000000 |
| 0026 | 1941 | 1 | 26 | 6.40000010 |
| 0027 | 1941 | 1 | 27 | 1.79999995 |
| 0028 | 1941 | 1 | 28 | 4.40000010 |
| 0029 | 1941 | 1 | 29 | 5.40000010 |
| 0030 | 1941 | 1 | 30 | 0.800000012 |
| 0031 | 1941 | 1 | 31 | 11.3000002 |
| 0032 | 1941 | 2 | 1 | 3.29999995 |
| 0033 | 1941 | 2 | 2 | 0.00000000 |
| 0034 | 1941 | 2 | 3 | 13.1999998 |
| 0035 | 1941 | 2 | 4 | 0.00000000 |
| 0036 | 1941 | 2 | 5 | 0.00000000 |

Série temporal diária

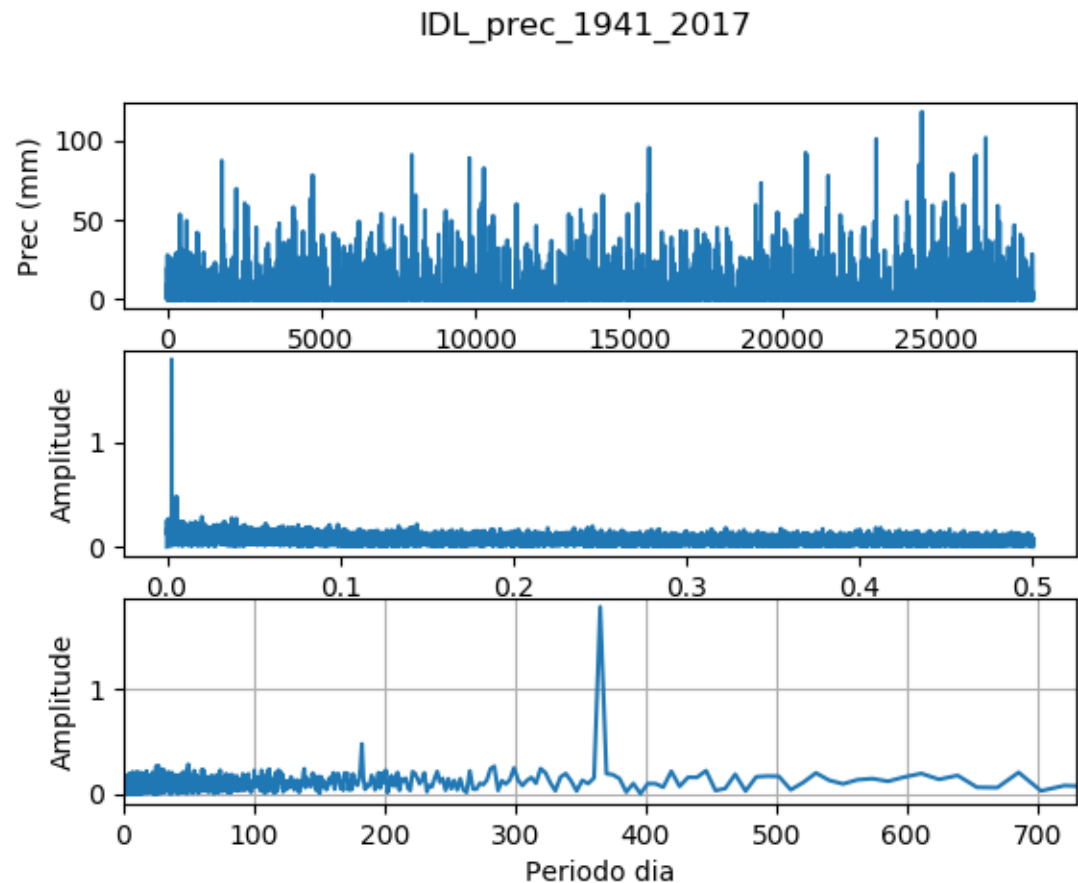


Espectro

```
def ampSPEC(s,dt,titulo=' ',unitx='s',unity='',graph='',maxPER=0):
    n=len(s); S=np.fft.fft(s-np.mean(s))
    fNyq=1./(2*dt)
    df=fNyq/(n//2)
    freq=np.arange(0,fNyq+df/2,df)
    if len(graph)!=0:
        plt.subplot(3,1,1); plt.plot(np.arange(0,n*dt,dt),s)
        plt.ylabel(unity); plt.suptitle(titulo)
        plt.subplot(3,1,2)
        plt.plot(freq,np.abs(S[0:n//2+1]/(n//2)))
        plt.ylabel('Amplitude');
        plt.xlabel('f '+unitx+r'$^{-1}$')
        plt.subplot(3,1,3);
        plt.plot(1/freq[1:],np.abs(S[1:n//2+1]/(n//2)))
        plt.xlabel('Periodo '+unitx)
        if maxPER>0:
            plt.xlim(0,maxPER)
        plt.ylabel('Amplitude'); plt.grid(); plt.savefig(graph+'.png')
    return S,freq
```

```
PREC, freq=ampSPEC (prec, 1, titulo='IDL_prec_1941_2017',  
, unitx='dia', unity='Prec (mm)', graph='t', maxPER=730)
```

Notar que a maior parte dos dias prec=0!

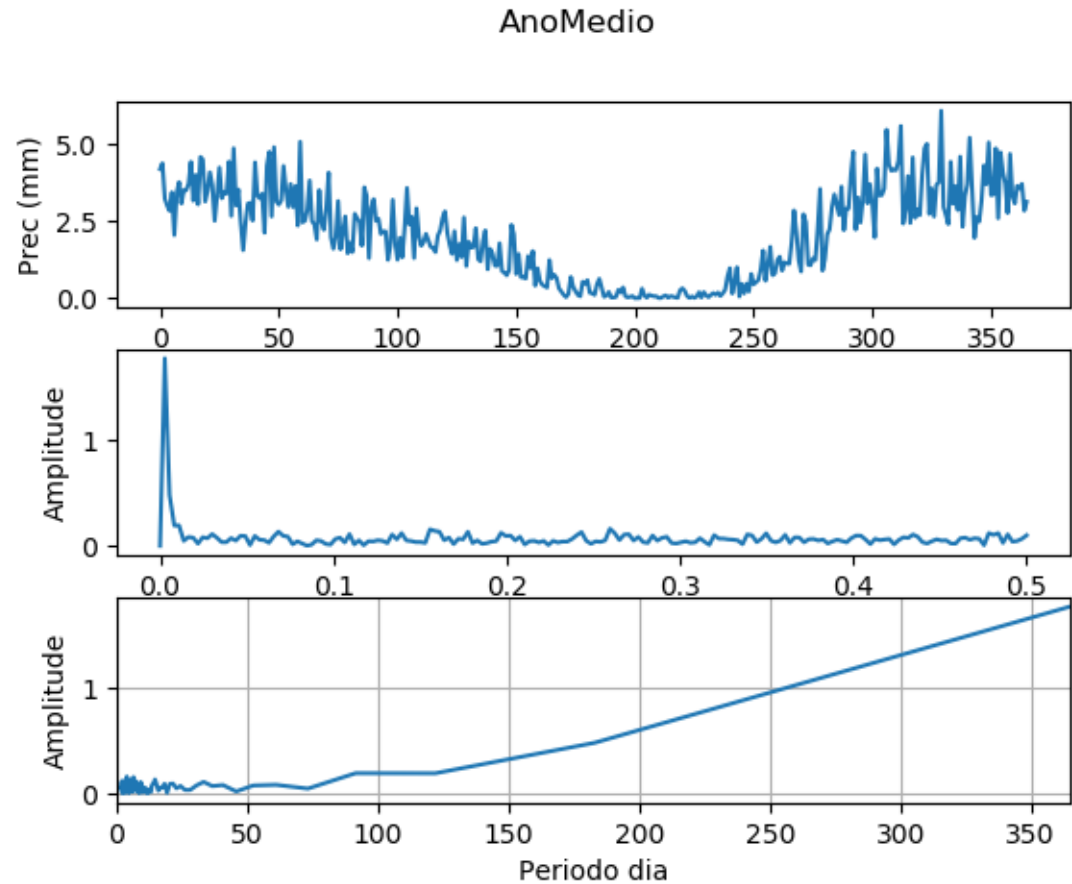
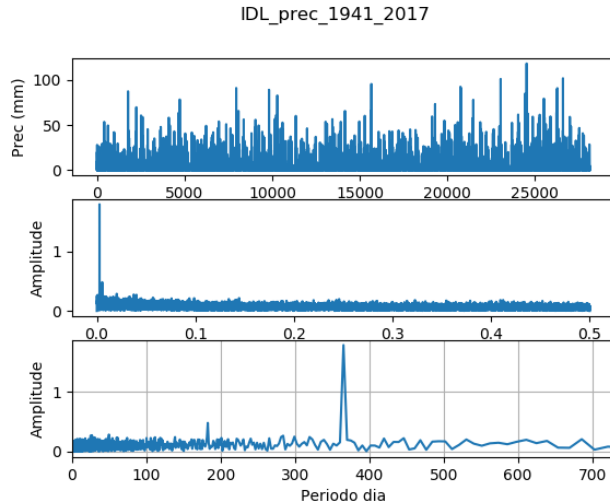


Ano médio

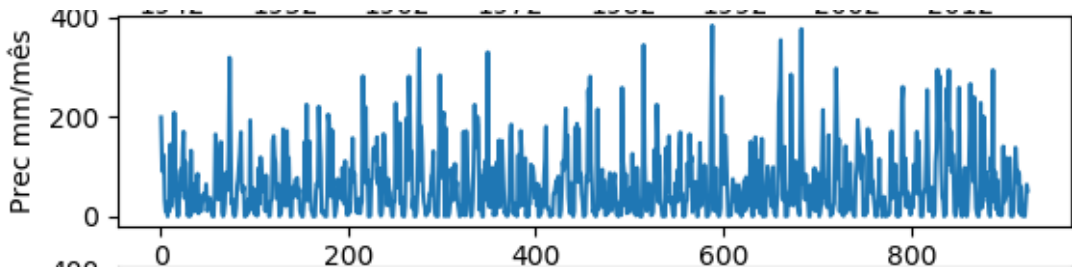
```
precMED=np.zeros(366)
nMED=np.zeros(366)
for kd in range(n):
    doy=int(tempo[kd].strftime('%j'))
    precMED[doy-1]+=prec[kd]
    nMED[doy-1]+=1
precMED=precMED/nMED
```

```
PRECa ,freqa=ampSPEC (precMED,1,titulo='AnoMedio',unit  
x='dia',unity='Prec (mm)',graph='teste',maxPER=365)
```

Consistente!



Total mensal



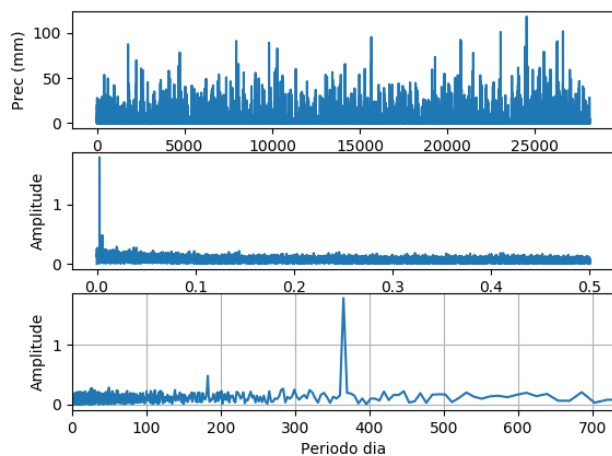
```
ndMES=[31,28,31,30,31,30,31,31,30,31,30,31]
anoFIRST=np.min(ano); anoLAST=np.max(ano)
numMES=12*(anoLAST-anoFIRST+1) #número total de meses
precM=np.zeros((numMES),dtype=float)
mJ=-1; dJ=-1
for yy in range(anoFIRST,anoLAST+1):
    for mm in range(1,12+1):
        mJ=mJ+1
        ndm=ndMES[mm-1]
        if mm==2 and (yy%4==0 and (yy%100!=0 or yy%400==0)) :
            ndm=ndm+1
        for dd in range(ndm):
            dJ=dJ+1
            precM[mJ]=precM[mJ]+prec[dJ]
plt.subplot(3,1,2)
plt.plot(np.linspace(1,numMES,numMES),precM)
plt.ylabel('Prec mm/mês')
```

Ano bissexto

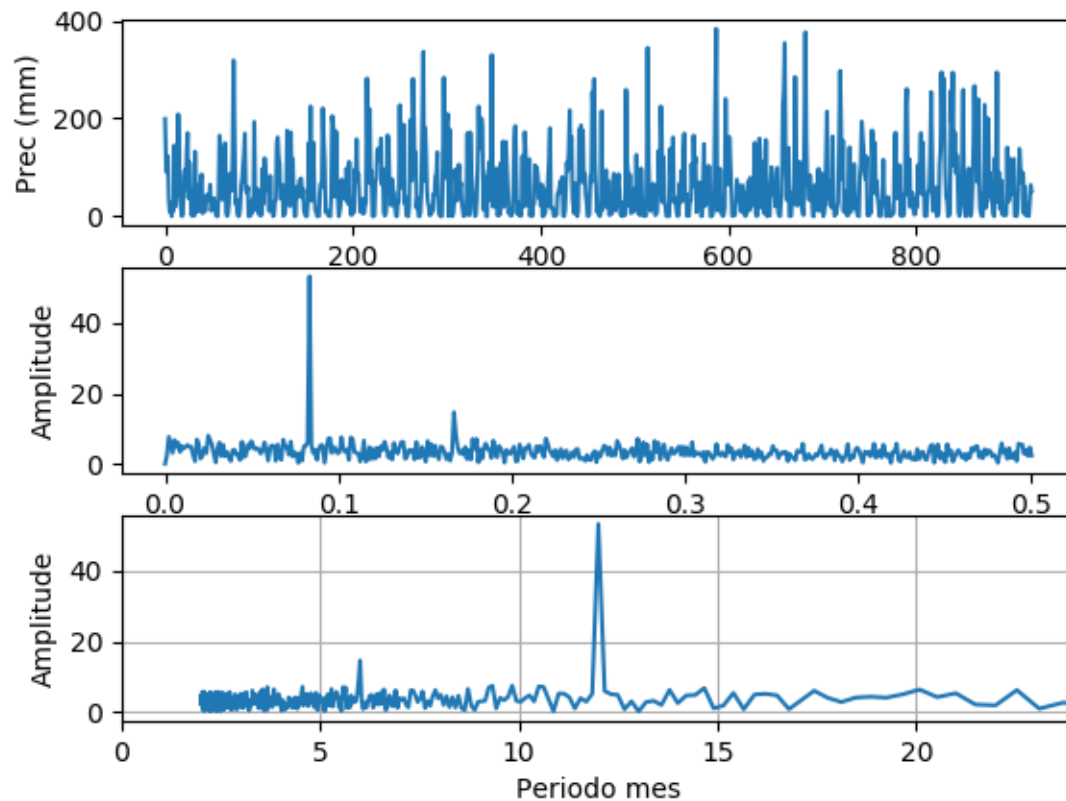

```
PRECm, freqm=ampSPEC (precM,1, titulo='AnoMedio', unitx='mes', unity='Prec (mm)', graph='testeM', maxPER=24)
```

A série mensal quase não tem zeros

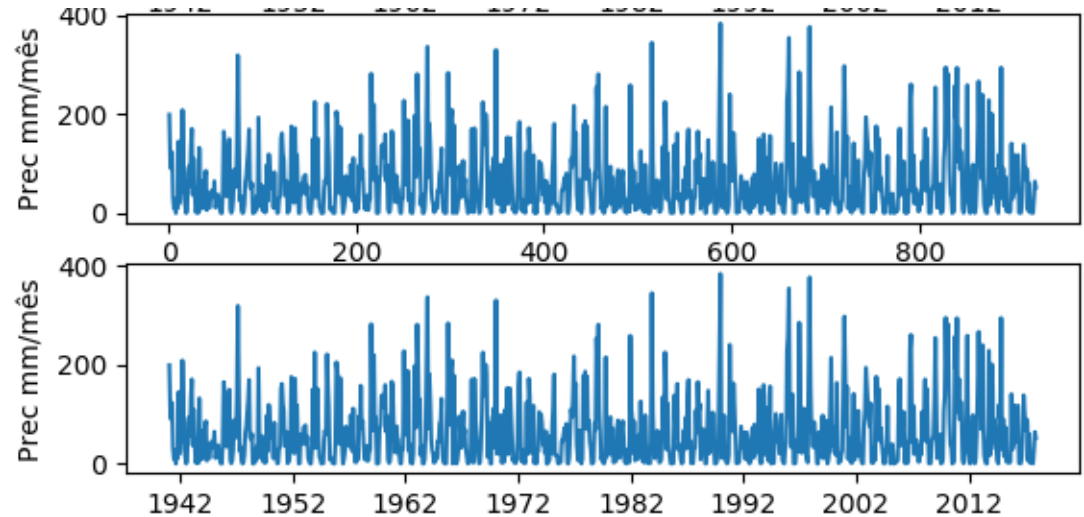
IDL_prec_1941_2017



AnoMedio



Total mensal c/datetime



```
mesLIST=[]  
for yy in range(anoFIRST,anoLAST+1):  
    for mm in range(1,13):  
        mesLIST.append(datetime.datetime(yy,mm,15))  
plt.subplot(3,1,3)  
plt.plot(mesLIST,precM)  
plt.ylabel('Prec mm/mês')
```

Datas julianas

Contagem de **dias sucessivos** desde uma data de referência.

Pode ser referido a um ano:

$$\text{dia_juliano} \in [1,365 \text{ ou } 366]$$

Ou a um periodo anos. Em **datetime** os dias julianos são contados a partir de 0001-01-01 (1º dia do calendário moderno)

```
X=datetime.datetime(2018,3,16,12,00,35,89)
```

```
print(X,X.toordinal())
```

```
>>2018-03-16 12:00:35.000089 736769
```

```
X=datetime.datetime(1,1,1,12,00,35,89)
```

```
print(X,X.toordinal())
```

```
>>0001-01-01 12:00:35.000089 1
```

datetime

calendar

Dados sintéticos

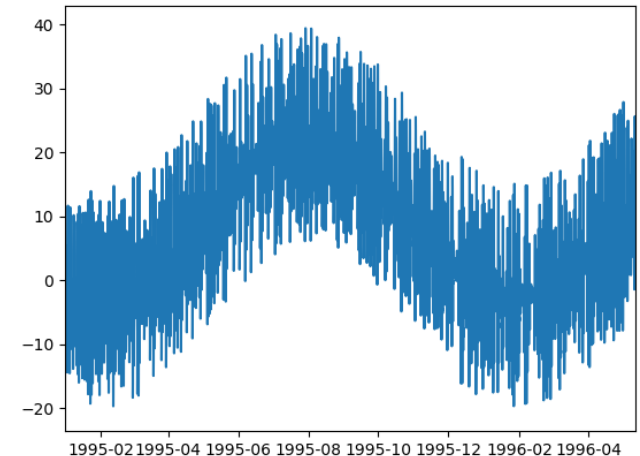
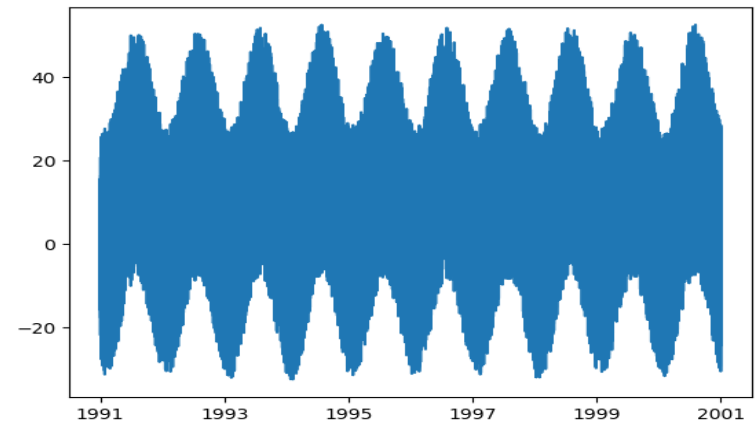
```
import numpy as np
import datetime;import calendar
ySTART=1991;yEND=2000
dataSTA=datetime.datetime(ySTART,1,1,0);
dataEND=datetime.datetime(yEND,12,31,23)
numOBS=(dataEND.toordinal()-dataSTA.toordinal()+1)*24
T=np.zeros((numOBS),dtype=float);kh=-1
for ano in range(ySTART,yEND+1):
    AmpA=12+2*(np.random.rand()-0.5) #AmpA∈[11,13] media 12
    if calendar.isleap(ano):
        ndays=366
    else:
        ndays=365
    for julian in range(1,ndays+1):
        AmpD=10+10*(np.random.rand()) #AmpD∈[10,30] media 20
        for hour in range(0,24):
            kh=kh+1
            T[kh]=10+AmpD*np.sin(2*hour*np.pi/24.+4*np.pi/3)\
                +AmpA*np.sin(2*julian*np.pi/ndays+4*np.pi/3)
```

Datetime

.timedelta

```
import matplotlib.pyplot as plt
time0=dataSTA
dateList = [] #empty list
for hs in range(0, numOBS):
    dateList.append(time0+datetime.timedelta(hours=hs))
plt.plot(dateList,T)

plt.figure()
plt.plot(dateList,T)
plt.xlim(datetime.datetime(1995,1,1),\
          datetime.datetime(1996,5,12))
plt.savefig('serie_t_extrato.png')
```

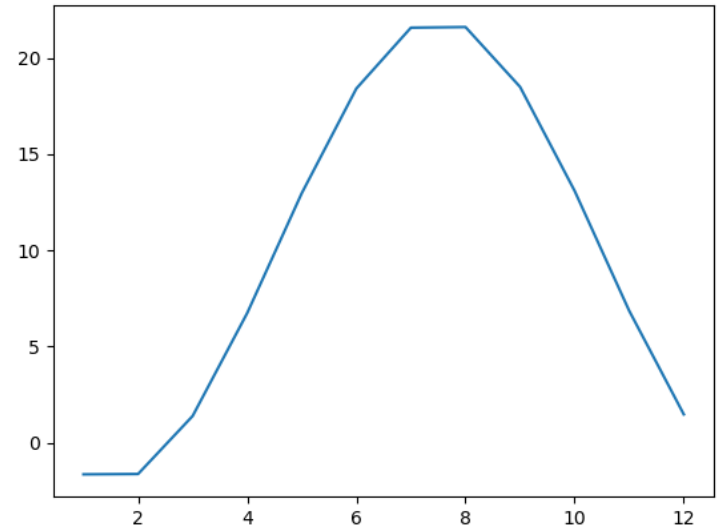


Ano médio

```
Tmes=np.zeros((12),dtype=float)
Nmes=np.zeros((12),dtype=int)
for kh in range(len(T)):
    mes=dateList[kh].month-1
    Tmes[mes]=Tmes[mes]+T[kh]
    Nmes[mes]=Nmes[mes]+1
```

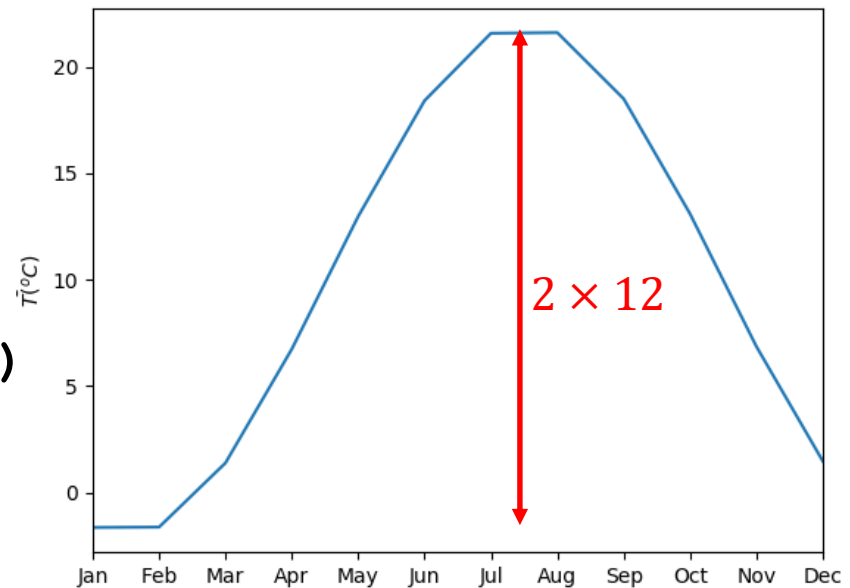
Tmes=Tmes/Nmes

```
plt.plot(np.linspace(1,12,12),Tmes)
plt.xlim(1,12)
plt.xlabel('mês')
plt.ylabel(r'$T\bar{ } (^{\circ}C)$')
```



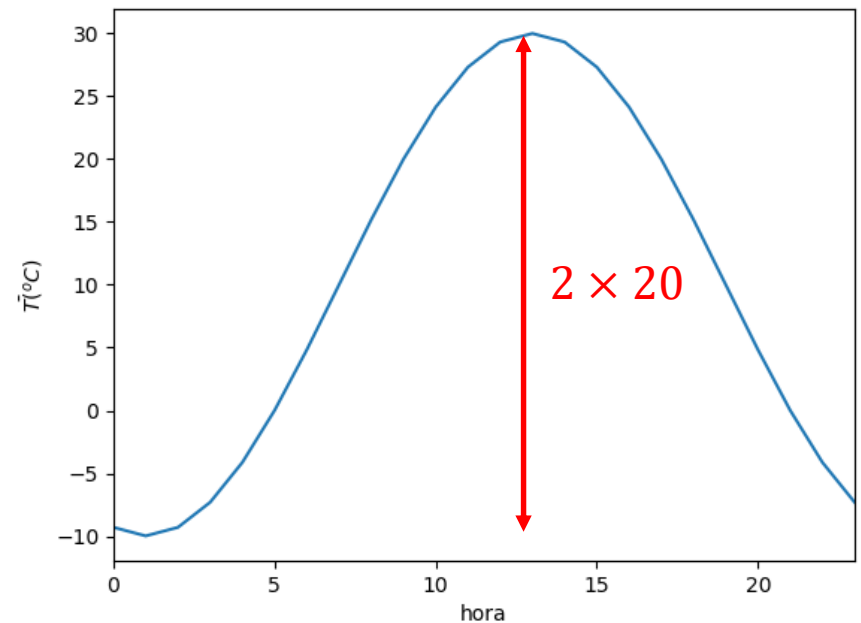
Ano médio versão 2

```
Tmes=np.zeros((12),dtype=float)
Nmes=np.zeros((12),dtype=int)
for kh in range(len(T)):
    mes=dateList[kh].month-1
    Tmes[mes]=Tmes[mes]+T[kh]
    Nmes[mes]=Nmes[mes]+1
Tmes=Tmes/Nmes
my_xticks = ['Jan', 'Feb', 'Mar', 'Apr', 'May', \
             'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec']
plt.xticks(np.linspace(1,12,12), my_xticks)
plt.plot(np.linspace(1,12,12),Tmes)
plt.xlim(1,12)
plt.ylabel(r'$\bar{T}$ (°C)')
```



Dia médio

```
Th=np.zeros((24),dtype=float)
Nh=np.zeros((24),dtype=int)
for kh in range(len(T)):
    hora=dateList[kh].hour-1
    Th[hora]=Th[hora]+T[kh]
    Nh[hora]=Nh[hora]+1
Th=Th/Nh
plt.figure()
plt.plot(np.linspace(0,23,24),Th)
plt.xlim(0,23)
plt.xlabel('hora')
plt.ylabel(r'$T\bar{\text{ }}$ (^oC)$')
```



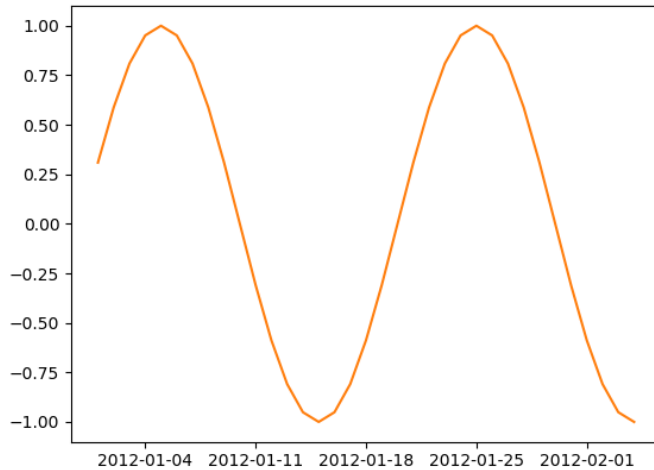
Nota: o ruído desapareceu na média

datas excel

```
import numpy as np
import datetime
import openpyxl as pyxl
import matplotlib.pyplot as plt
wb=pyxl.load_workbook('data_xls.xlsx',\
    data_only=True)
ws=wb['Sheet1']
rows=ws.max_row
datas=[] #lista
valor=np.zeros((rows-1))
for r in range(2,rows+1):
    datas.append(ws.cell(row=r,column=1).value)
    valor[r-2]=ws.cell(row=r,column=3).value
for r in range(len(datas)):
    print(r,datas[r])
plt.plot(datas,valor)
```

| | A | B | C | D |
|----|-----------|-----|----------|---|
| 1 | data | num | valor | |
| 2 | 01-Jan-12 | 1 | 0.309017 | |
| 3 | 02-Jan-12 | 2 | 0.587785 | |
| 4 | 03-Jan-12 | 3 | 0.809017 | |
| 5 | 04-Jan-12 | 4 | 0.951057 | |
| 6 | 05-Jan-12 | 5 | 1 | |
| 7 | 06-Jan-12 | 6 | 0.951057 | |
| 8 | 07-Jan-12 | 7 | 0.809017 | |
| 9 | 08-Jan-12 | 8 | 0.587785 | |
| 10 | 09-Jan-12 | 9 | 0.309017 | |
| 11 | 10-Jan-12 | 10 | 1.23E-16 | |
| 12 | 11-Jan-12 | 11 | -0.30902 | |
| 13 | 12-Jan-12 | 12 | -0.58779 | |
| 14 | 13-Jan-12 | 13 | -0.80902 | |
| 15 | 14-Jan-12 | 14 | -0.95106 | |
| 16 | 15-Jan-12 | 15 | -1 | |
| 17 | 16-Jan-12 | 16 | -0.95106 | |
| 18 | 17-Jan-12 | 17 | -0.80902 | |
| 19 | 18-Jan-12 | 18 | -0.58779 | |

```
plt.plot(datas, valor)
```



```
for r in range(len(datas)):
```

```
    print(r, datas[r])
```

```
0 2012-01-01 00:00:00
```

```
1 2012-01-02 00:00:00
```

```
2 2012-01-03 00:00:00
```

```
3 2012-01-04 00:00:00
```

```
...
```

```
34 2012-02-04 00:00:00
```

```
35 2012-02-05 00:00:00
```

| | A | B | C | D |
|----|-----------|-----|----------|---|
| 1 | data | num | valor | |
| 2 | 01-Jan-12 | 1 | 0.309017 | |
| 3 | 02-Jan-12 | 2 | 0.587785 | |
| 4 | 03-Jan-12 | 3 | 0.809017 | |
| 5 | 04-Jan-12 | 4 | 0.951057 | |
| 6 | 05-Jan-12 | 5 | 1 | |
| 7 | 06-Jan-12 | 6 | 0.951057 | |
| 8 | 07-Jan-12 | 7 | 0.809017 | |
| 9 | 08-Jan-12 | 8 | 0.587785 | |
| 10 | 09-Jan-12 | 9 | 0.309017 | |
| 11 | 10-Jan-12 | 10 | 1.23E-16 | |
| 12 | 11-Jan-12 | 11 | -0.30902 | |
| 13 | 12-Jan-12 | 12 | -0.58779 | |
| 14 | 13-Jan-12 | 13 | -0.80902 | |
| 15 | 14-Jan-12 | 14 | -0.95106 | |
| 16 | 15-Jan-12 | 15 | -1 | |
| 17 | 16-Jan-12 | 16 | -0.95106 | |
| 18 | 17-Jan-12 | 17 | -0.80902 | |
| 19 | 18-Jan-12 | 18 | -0.58779 | |

Objetos datetime (hh:mm:ss a zero)