



による影響を受けやすいと考えられる。そのため、将来、夏にこれまでより暑くなったと感じるよりも、冬にこれまでより暖かくなったと感じるようになるのではないかと考えられる。

最高気温と最低気温の相関関係を調べたところ、0.1050 となり、相関は見られなかった。このことから、1年の最高気温と最低気温の両方が高くなったり、低くなったりすることはなく、気温のばらつきが大きくなると考えられる。そのため、地球温暖化によって、気温のばらつきが大きくなり、体調を崩しやすくなると考えられる。

日本の猛暑日の日数の推移のグラフの青い棒グラフは1年ごとの猛暑日の日数、オレンジの直線は最小二乗法によって求めた線形回帰線である。直線の傾きが0.0150 となったので、100年で猛暑日は1~2日増えるとされる。このことから、現在の状況で地球温暖化が進めば、100年後に猛暑日は1~2日程度多くなると考えられる。

日本の真冬日の日数の推移のグラフの青い棒グラフは1年ごとの真冬日の日数、オレンジの直線は最小二乗法によって求めた線形回帰線である。直線の傾きが-0.0211 となったので、100年で真冬日は2日減るとされる。このことから、現在の状況で地球温暖化が進めば、100年後に真冬日は2日程度少なくなると考えられる。

猛暑日と真冬日のグラフより、真冬日の線形回帰線の傾きの方が大きくなった。よって、地球温暖化により、冬の方が受ける影響が大きいと考えられる。

猛暑日の日数と真冬日の日数の相関は、-0.0497 となった。よって、猛暑日と真冬日の日数の間に相関関係はないと考えられる。

日本の平均気温の推移のグラフの青い線は、年ごとの平均気温をプロットしたグラフである。オレンジの直線は最小二乗法によって求めた線形回帰線である。直線の勾配より、日本の平均気温は1年で0.0143度、100年で1.43度上昇するとされる。そのため、地球温暖化対策をとらず、現状のまま進んだ場合、100年後には今より、平均気温が1.43度高くなると考えられる。

日本の真夏日の日数の推移のグラフの青い棒グラフは1年ごとの真夏日の日数、オレンジの直線は最小二乗法によって求めた線形回帰線である。直線の傾きが0.0527 となったので、100年で真夏日は5日程度増えるとされる。このことから、現在の状況で地球温暖化が進めば、100年後に真夏日は5日程度多くなると考えられる。

日本の冬日の日数の推移のグラフの青い棒グラフは1年ごとの冬日の日数、オレンジの直線は最小二乗法によって求めた線形回帰線である。直線の傾きが-0.3068 となったので、100年で冬日は3日程度減るとされる。このことから、現在の状況で地球温暖化が進めば、100年後に冬日は3日程度減ると考えられる。

真夏日の日数と冬日の日数のグラフより、真夏日の方が線形回帰線の傾きが大きくなった。このことから、地球温暖化による影響は、最低気温0度未満の日が減ることよりも、最高気温30度以上の日が増えることに影響を与えると考えられる。また、どちらグラフも、将来の気温は今よりも高くなることを表している。

猛暑日と真夏日のグラフより、線形回帰線の傾きは真夏日の方が大きくなった。よって、将来、暑すぎる日（35度以上）が多くなるというよりも、暑い日（30度以上）が多くなると考えられる。いずれにしても、暑い日が多くなると考えられる。

真夏日の日数と冬日の日数の相関は、-0.0312 となった。よって、真夏日と冬日の日数の間に相関関係はないと考えられる。

今回得られたグラフより、地球の気温は上昇していることが分かった。今後、地球温暖化の対策をしなかったとしたら、地球の気温は上がり続ける。地球温暖化により、暑い日は増え、寒い日は減ることが考えられる。

+++++

## ○感想

130 年分の平均気温（1 日）、最高気温、最低気温を取得する時、10 年ごとにデータを取得したが、これを matlab で 1 年ごとに分けるのが難しく、苦勞した。1 年ごとに平均気温、最高気温、最低気温を分けるプログラムを書けたが、今後、もっと良い分け方がないか考えたい。今回作ったプログラムで、新潟だけでなく、日本各地や世界の気温の変化についても調べてみたい。また、気温だけでなく、株価などの経済の動きや人口の推移、地震の発生回数などについても解析すると面白いと思った。

+++++

記述欄おわり---

(ここからプログラムが始まります)

気象庁の Web サイトでは、過去の気象データの閲覧、ダウンロードサービスを提供しています。

このサービスを利用して、取得した気象データを分析してみましょう。

このプログラムでは、データの取得から MATLAB への読み込み、基本的なグラフ作成までの流れを体験できます。

## ■気象庁 Web 過去の気象データ・ダウンロード

<https://www.data.jma.go.jp/gmd/risk/obsdl/index.php>

検索条件を以下の通り設定して CSV ファイルをダウンロードします。

- 地点を選ぶ：新潟県→新潟を選択
- 項目を選ぶ：日別値の日最高気温および日最低気温
- 期間を選ぶ：1 月 1 日から 12 月 31 日の値を 2020 年から 2020 年まで表示
- 表示オプションを選ぶ：変更せず

## 過去の気象データ・ダウンロード

**重要なお知らせ** [このページの使い方](#) [よくある質問](#) [CSVファイルの形式](#)

検索条件

選択済みのデータ量 0% 100% (上限)

地点を選ぶ

項目を選ぶ

期間を選ぶ

表示オプションを選ぶ

画面に表示 ▶

CSVファイルをダウンロード ▶

選択地点・項目をクリア

選択された地点 観測項目

新潟  削除

選択された項目

日最高気温 削除

日最低気温 削除

選択された期間(日本標準時)

1月1日から12月31日までの日別値を

2020年から2020年まで表示

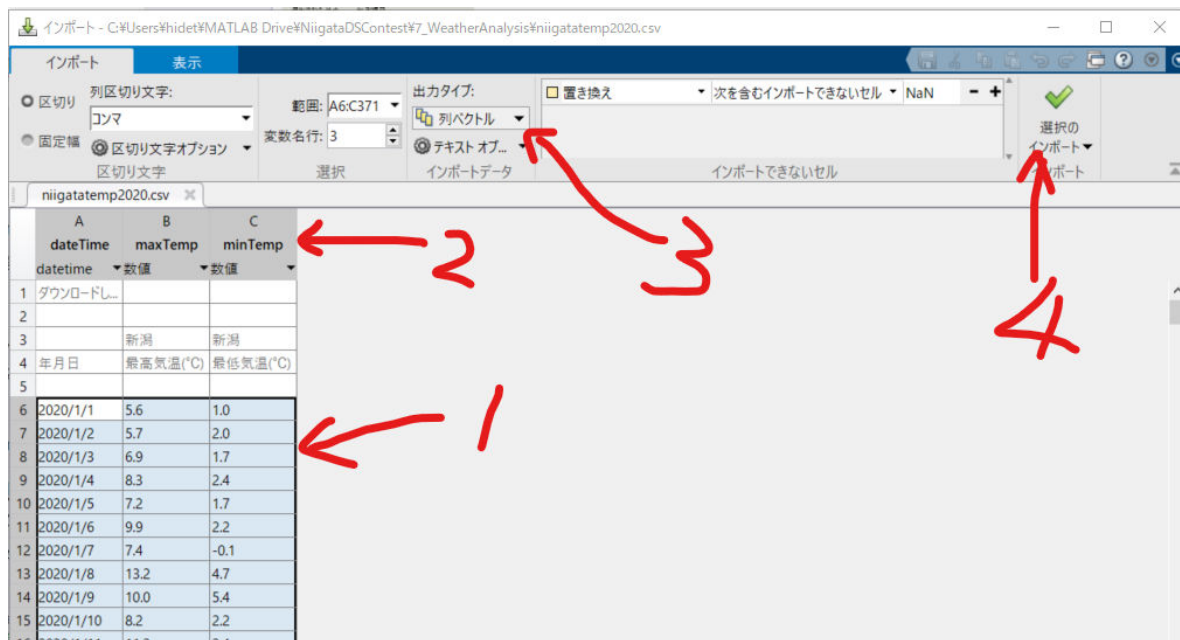
同梱のサンプルファイル"niigatatemp2020.csv"は、上記の手順でダウンロードしたファイルです。

(ダウンロード後に名前を変更しています)

## ファイルからデータを読み込む

MATLAB の現在のフォルダーに表示されている niigatatemp2020.csv をダブルクリックして、インポートツールを表示します。

下の手順に従って、ファイルから必要なデータを読み込みます。

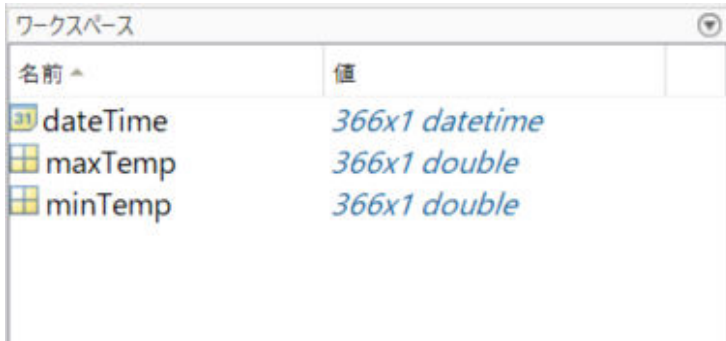


	A	B	C
	dateTime	maxTemp	minTemp
	datetime	数値	数値
1	ダウンロードし...		
2			
3		新潟	新潟
4	年月日	最高気温(°C)	最低気温(°C)
5			
6	2020/1/1	5.6	1.0
7	2020/1/2	5.7	2.0
8	2020/1/3	6.9	1.7
9	2020/1/4	8.3	2.4
10	2020/1/5	7.2	1.7
11	2020/1/6	9.9	2.2
12	2020/1/7	7.4	-0.1
13	2020/1/8	13.2	4.7
14	2020/1/9	10.0	5.4
15	2020/1/10	8.2	2.2
16	2020/1/11	11.3	2.4

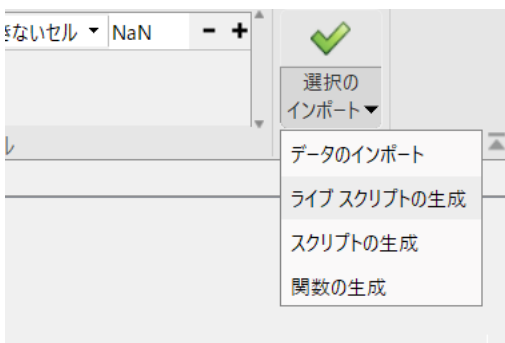
1. スプレッドシートでデータの入ったセルのみ選択されていることを確認

2. A, B, C それぞれ dateTime, maxTemp, minTemp に変更(これが変数の名前になります)
3. 出力タイプを列ベクトルに変更
4. 選択のインポートを実行

ワークスペース上に、dateTime に日付、maxTemp に最高気温、minTemp に最低気温の変数が作成されたことを確認してください。



Tip: インポートツール上で行った操作をプログラム化する機能があります。これにより、同じ種類のファイルを毎回インポートツールを使わずにプログラムを使って自動的に読み込むことができます。インポートツールの[選択のインポート]からメニューでライブスクリプトの生成を選択すると、ライブスクリプト形式のプログラムが自動生成されます。ぜひお試しください。



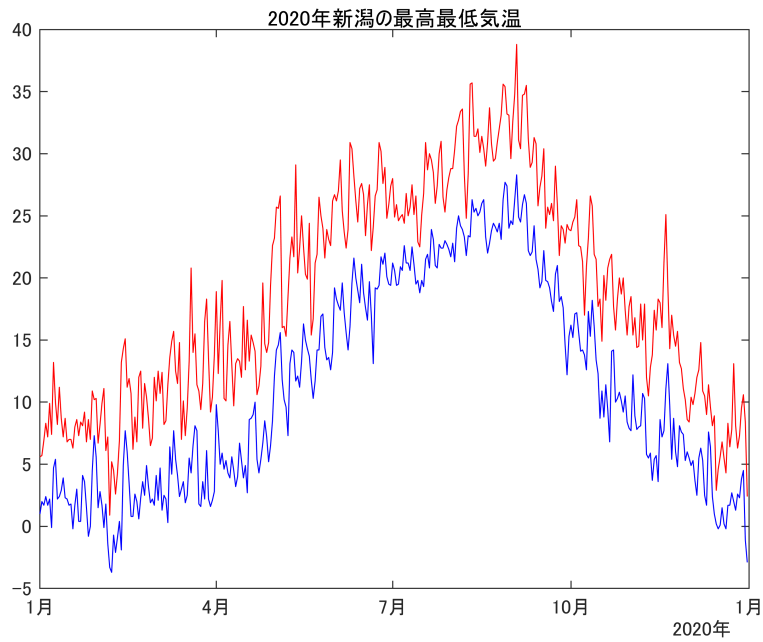
## プログラムを実行してグラフを確認

データを MATLAB に読み込んだら、このプログラムを実行してグラフを描いてみましょう。

### グラフに出力(日別最高最低気温の推移)

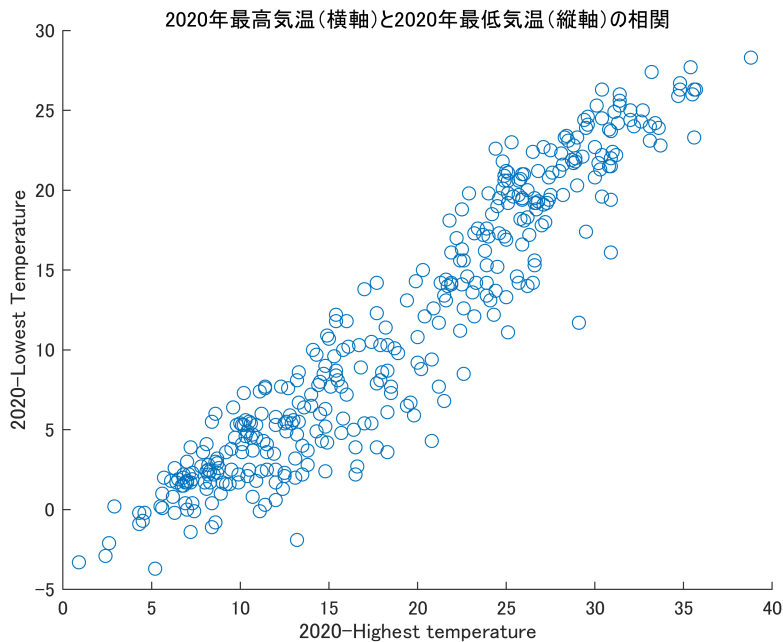
まずは日付を横軸、最高気温、最低気温を縦軸にして 1 年間の気温推移をグラフにします。

```
figure(1)
plot(dateTime, maxTemp, 'r') %最高気温を赤で表示
hold on %同じグラフ追記する時のコマンド
plot(dateTime, minTemp, 'b') %最低気温を青で表示
title('2020 年新潟の最高最低気温')
hold off
```



### %日別最高最低気温の相関関係

```
scatter(maxTemp,minTemp)
coMaxmin = corrcoef(maxTemp,minTemp);
xlabel('2020-Highest temperature')
ylabel('2020-Lowest Temperature')
title('2020年最高気温（横軸）と2020年最低気温（縦軸）の相関')
```



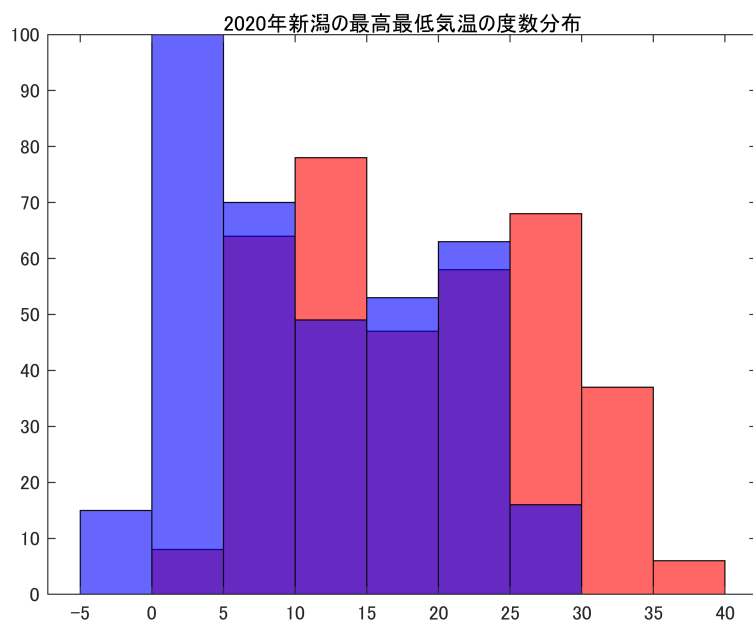
## グラフに出力(ヒストグラム)

次はヒストグラムを作成してみましょう。

```

figure(2)
histogram(maxTemp, 'FaceColor', 'red') %最高気温のヒストグラムを赤で表示
hold on
histogram(minTemp, 'FaceColor', 'blue') %最低気温のヒストグラムを青で表示
title('2020年新潟の最高最低気温の度数分布')
hold off

```



## 自分で気象データを取得して、データサイエンスしてみましょう

ここまで、データの取得から基本的なグラフの出力まで一通り作業しました。

作業の流れを掴んだところで、自分で気象庁 Web サイトから好きなデータをダウンロードして、MATLAB で解析や将来予測をしてください。

年間最高気温と最低気温をプロットする。

```

%それぞれの年ごとのデータを取り出す.
yearnum = (1891:1:2020)'; %change

data1891 = readmatrix('data1891.csv');
data1901 = readmatrix('data1901.csv');
data1911 = readmatrix('data1911.csv');
data1921 = readmatrix('data1921.csv');
data1931 = readmatrix('data1931.csv');
data1941 = readmatrix('data1941.csv');
data1951 = readmatrix('data1951.csv');
data1961 = readmatrix('data1961.csv');
data1971 = readmatrix('data1971.csv');
data1981 = readmatrix('data1981.csv');
data1991 = readmatrix('data1991.csv');
data2001 = readmatrix('data2001.csv');

```

```

data2011 = readmatrix('data2011.csv');
dataall = [data1891;data1901;data1911;...
           data1921;data1931;data1941;data1951;data1961;...
           data1971;data1981;data1991;data2001;data2011];
yeardata130 = cell(1,length(yearnum)); %change
monthdata130 = cell(1,length(yearnum)); %change
daydata130 = cell(1,length(yearnum)); %change
meantemp130 = cell(1,length(yearnum)); %change
Mtemp130 = cell(1,length(yearnum)); %change
mtemp130 = cell(1,length(yearnum)); %change
for a = 1891:1:2020 %change
    yeara = zeros(length(dataall),1);
    montha = zeros(length(dataall),1);
    daya = zeros(length(dataall),1);
    meantemp = zeros(length(dataall),1);
    Mtemp = zeros(length(dataall),1);
    mtemp = zeros(length(dataall),1);

    for b = 1:length(dataall)
        if dataall(b,1) == a
            yeara(b,1) = dataall(b,1);
            montha(b,1) = dataall(b,2);
            daya(b,1) = dataall(b,3);
            meantemp(b,1) = dataall(b,4);
            Mtemp(b,1) = dataall(b,5);
            mtemp(b,1) = dataall(b,6);
        end
    end
end

yeardata130{a-1890} = yeara(:,1); %change(年数を変えた場合)
monthdata130{a-1890} = montha(:,1); %change
daydata130{a-1890} = daya(:,1); %change
meantemp130{a-1890} = meantemp(:,1); %change
Mtemp130{a-1890} = Mtemp(:,1); %change
mtemp130{a-1890} = mtemp(:,1); %change
end

```

%要素が0である年の行を削除する.

```

for c = 1:130 %change
    idx = yeardata130{c} == 0;
    yeardata130{c} = yeardata130{c}(~idx,:);
    monthdata130{c} = monthdata130{c}(~idx,:);
    daydata130{c} = daydata130{c}(~idx,:);
    meantemp130{c} = meantemp130{c}(~idx,:);
    Mtemp130{c} = Mtemp130{c}(~idx,:);
    mtemp130{c} = mtemp130{c}(~idx,:);
end

```

%年ごとの気温の最大値, 最小値を歳出する.

```

M = zeros(130,1); %change
m = zeros(130,1); %change
for d = 1:130 %change
    M(d,1) = max(Mtemp130{d});

```

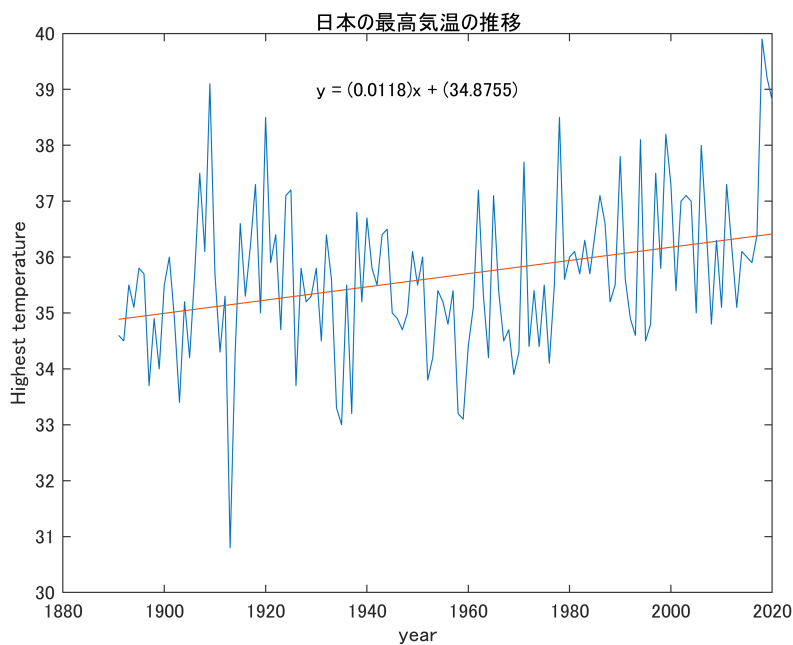


```

    m(d,1) = min(mtemp130{d});
end
figure
plot(yearnum,M)
%ylim([30 40])

%最高気温の線形回帰線  $y = px+a$  (最小二乗法によって求めた線形回帰線)
hold on
yearnum130 = (1:1:130)';
p1 = polyfit(yearnum130,M,1);
y1 = polyval(p1,yearnum130);
plot(yearnum,y1)
s1 = sprintf('y = (%.4f)x + (%.4f)',p1(1),p1(2));
text(1930,39,s1)
hold off
xlabel('year')
ylabel('Highest temperature')
title('日本の最高気温の推移')

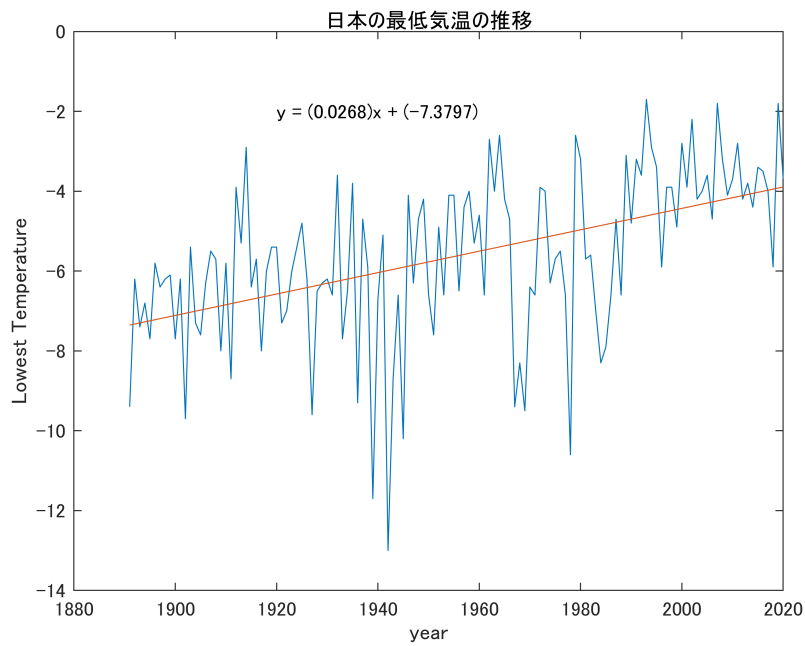
```



```

plot(yearnum,m)
%最低気温の線形回帰線  $y = px+a$  (最小二乗法によって求めた線形回帰線)
hold on
p2 = polyfit(yearnum130,m,1);
y2 = polyval(p2,yearnum130);
plot(yearnum,y2)
s2 = sprintf('y = (%.4f)x + (%.4f)',p2(1),p2(2));
text(1920,-2,s2)
hold off
xlabel('year')
ylabel('Lowest Temperature')
title('日本の最低気温の推移')

```



%最高気温と最低気温の相関係数，線形回帰線（最小二乗法）

```
%p3 = polyfit(M,m,1);
```

```
%y3 = polyval(p2,M);
```

```
%b3 = M\m;
```

```
%y3 = M*b3;
```

```
scatter(M,m)
```

```
hold on
```

```
%plot(M,y3)
```

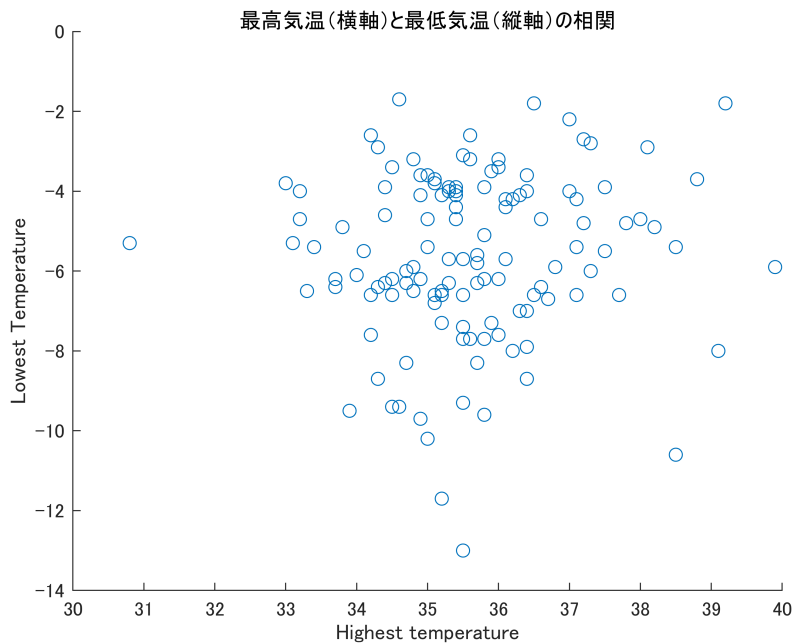
```
hold off
```

```
coMandm = corrcoef(M,m);
```

```
xlabel('Highest temperature')
```

```
ylabel('Lowest Temperature')
```

```
title('最高気温（横軸）と最低気温（縦軸）の相関')
```



%年ごとの猛暑日（最高気温 35 度以上）と真冬日（最高気温 0 度未満）の日数を算出する。

```
temp35num = cell(1, length(yearnum));
temp0num = cell(1, length(yearnum));
for e = 1:130 %change
    for f = 1:length(Mtemp130{e})
        if 35 <= Mtemp130{e}(f,1)
            temp35num{e}(f,1) = Mtemp130{e}(f,1);
        end
        if Mtemp130{e}(f,1) < 0
            temp0num{e}(f,1) = Mtemp130{e}(f,1);
        end
    end
end
```

%要素が 0 である年の行を削除する。

```
for f = 1:130 %change
    idx35 = temp35num{f} == 0;
    temp35num{f} = temp35num{f}(~idx35,:);

    idx0 = temp0num{f} == 0;
    temp0num{f} = temp0num{f}(~idx0,:);
end
```

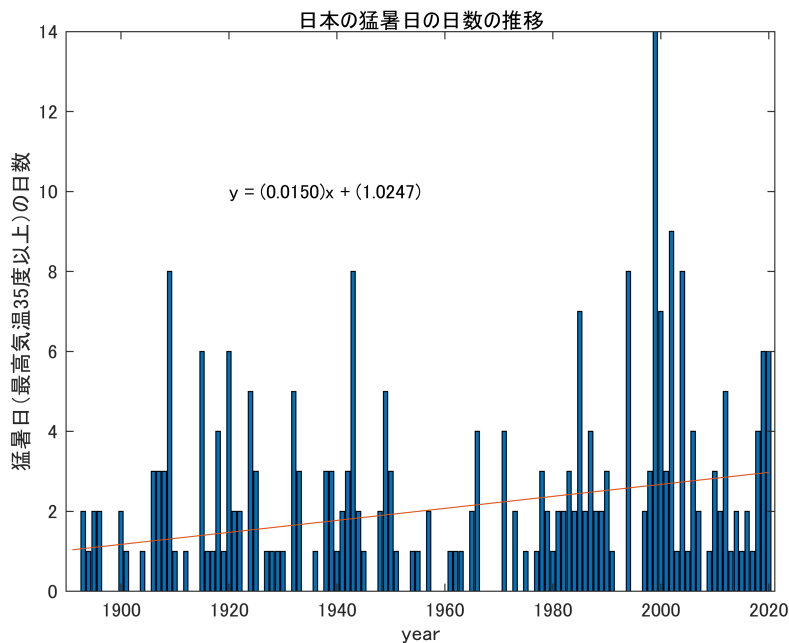
%猛暑日と真冬日の日数を算出する。

```
T35num = zeros(130,1); %change
T0num = zeros(130,1); %change

for g = 1:130 %change
    T35num(g,1) = length(temp35num{g});
    T0num(g,1) = length(temp0num{g});
end
bar(yearnum, T35num)
```

```
%猛暑日の日数の線形回帰線  $y = px+a$ (最小二乗法によって求めた線形回帰線)
```

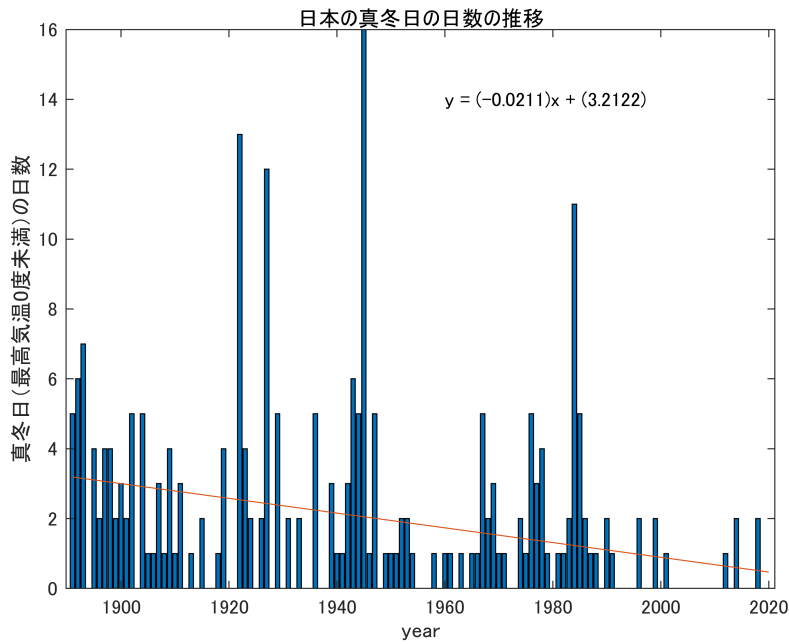
```
hold on  
p7 = polyfit(yearnum130,T35num,1);  
y7 = polyval(p7,yearnum130);  
plot(yearnum,y7)  
s7 = sprintf('y = (%.4f)x + (%.4f)',p7(1),p7(2));  
text(1920,10,s7)  
hold off  
xlabel('year')  
ylabel('猛暑日(最高気温35度以上)の日数')  
title('日本の猛暑日の日数の推移')
```



```
bar(yearnum,T0num)
```

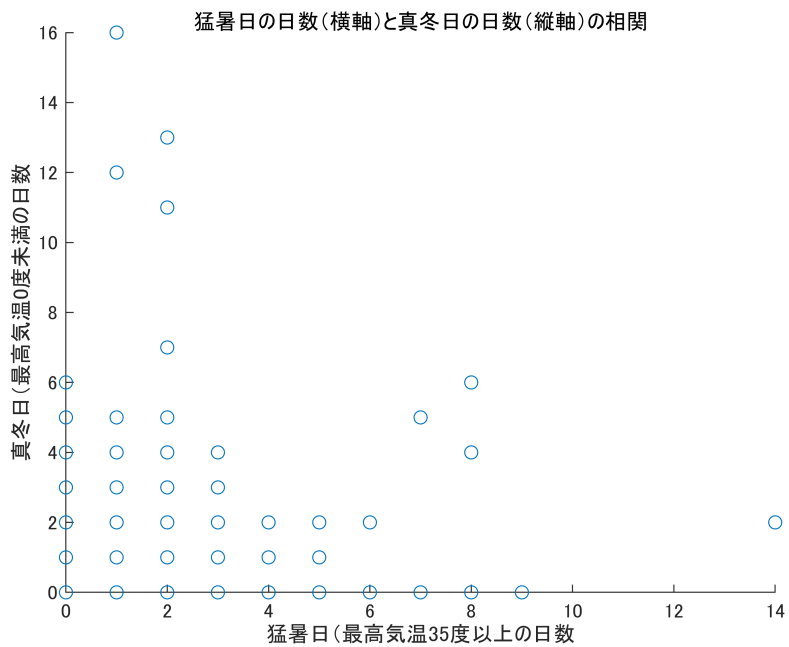
```
%真冬日の日数の線形回帰線  $y = px+a$ (最小二乗法によって求めた線形回帰線)
```

```
hold on  
p8 = polyfit(yearnum130,T0num,1);  
y8 = polyval(p8,yearnum130);  
plot(yearnum,y8)  
s8 = sprintf('y = (%.4f)x + (%.4f)',p8(1),p8(2));  
text(1960,14,s8)  
hold off  
xlabel('year')  
ylabel('真冬日(最高気温0度未満)の日数')  
title('日本の真冬日の日数の推移')
```



%相関係数 (猛暑日と真冬日)

```
scatter(T35num,T0num)
co35and0 = corrcoef(T35num,T0num);
xlabel('猛暑日 (最高気温 35 度以上の日数)')
ylabel('真冬日 (最高気温 0 度未満の日数)')
title('猛暑日の日数 (横軸) と真冬日の日数 (縦軸) の相関')
```



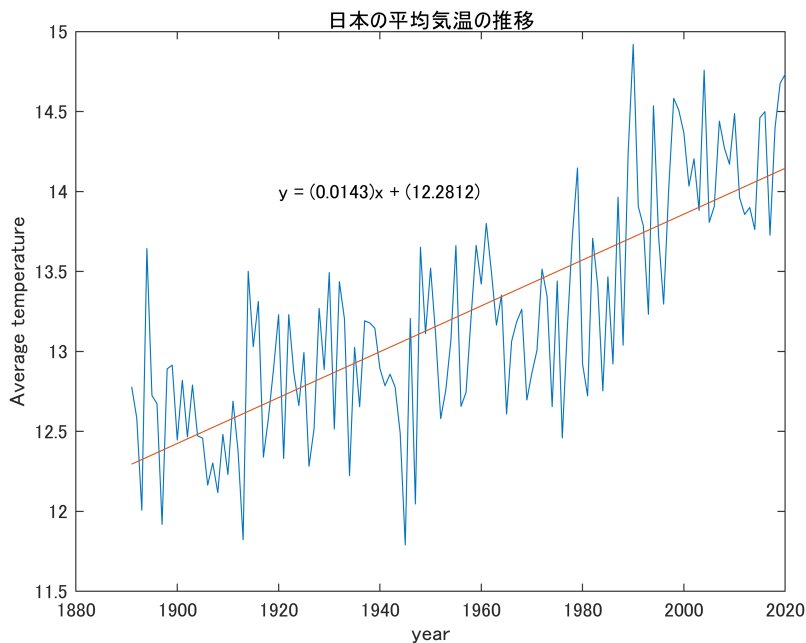
%年平均気温 (1日の平均気温で1年間の平均を取る。)

```
meanyeartemp = zeros(130,1);
for h = 1:130
```

```

meanyeartemp(h,1) = mean(meantemp130{h});
end
plot(yearnum,meanyeartemp)
%年平均気温の線形回帰線 y = px+a(最小二乗法によって求めた線形回帰線)
hold on
p4 = polyfit(yearnum130,meanyeartemp,1);
y4 = polyval(p4,yearnum130);
plot(yearnum,y4)
s4 = sprintf('y = (%.4f)x + (%.4f)',p4(1),p4(2));
text(1920,14,s4)
hold off
xlabel('year')
ylabel('Average temperature')
title('日本の平均気温の推移')

```



%年ごとの真夏日（最高気温 30 度以上）と冬日（最低気温 0 度未満）の日数を算出する。

```

temp30num = cell(1, length(yearnum));
tempm0num = cell(1, length(yearnum));
for e = 1:130 %change
    for f = 1:length(Mtemp130{e})
        if 30 <= Mtemp130{e}(f,1)
            temp30num{e}(f,1) = Mtemp130{e}(f,1);
        end
        if mtemp130{e}(f,1) < 0
            tempm0num{e}(f,1) = mtemp130{e}(f,1);
        end
    end
end
end

```

%要素が 0 である年の行を削除する。

```

for f = 1:130 %change
    idx35 = temp30num{f} == 0;
    temp30num{f} = temp30num{f}(~idx35,:);
end

```

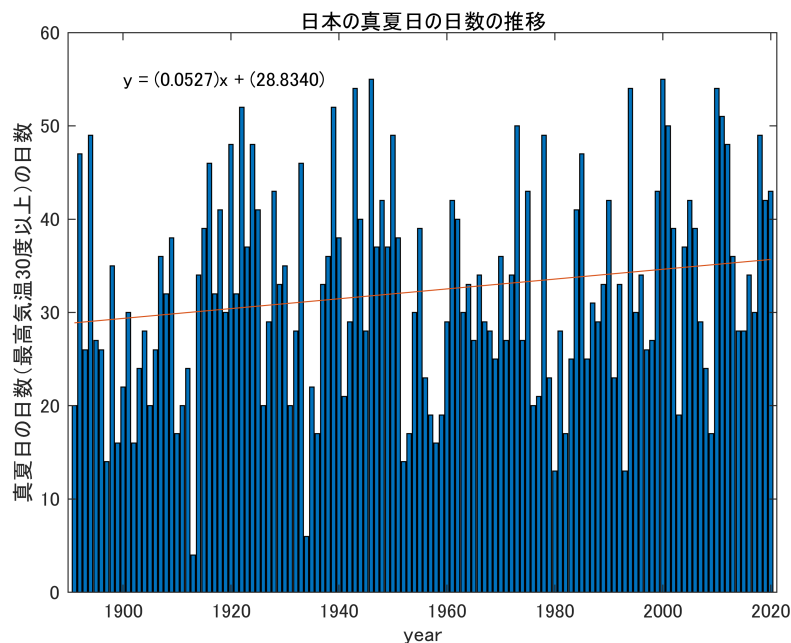
```

    idx0 = tempm0num{f} == 0;
    tempm0num{f} = tempm0num{f}(~idx0,:);
end

%真夏日と冬日の日数を算出する.
T30num = zeros(130,1); %change
Tm0num = zeros(130,1); %change

for g = 1:130 %change
    T30num(g,1) = length(temp30num{g});
    Tm0num(g,1) = length(tempm0num{g});
end
bar(yearnum,T30num)
%真夏日の日数の線形回帰線 y = px+a(最小二乗法によって求めた線形回帰線)
hold on
p5 = polyfit(yearnum130,T30num,1);
y5 = polyval(p5,yearnum130);
plot(yearnum,y5)
s5 = sprintf('y = (%.4f)x + (%.4f)',p5(1),p5(2));
text(1900,55,s5)
hold off
xlabel('year')
ylabel('真夏日の日数 (最高気温 30 度以上) の日数')
title('日本の真夏日の日数の推移')

```



```

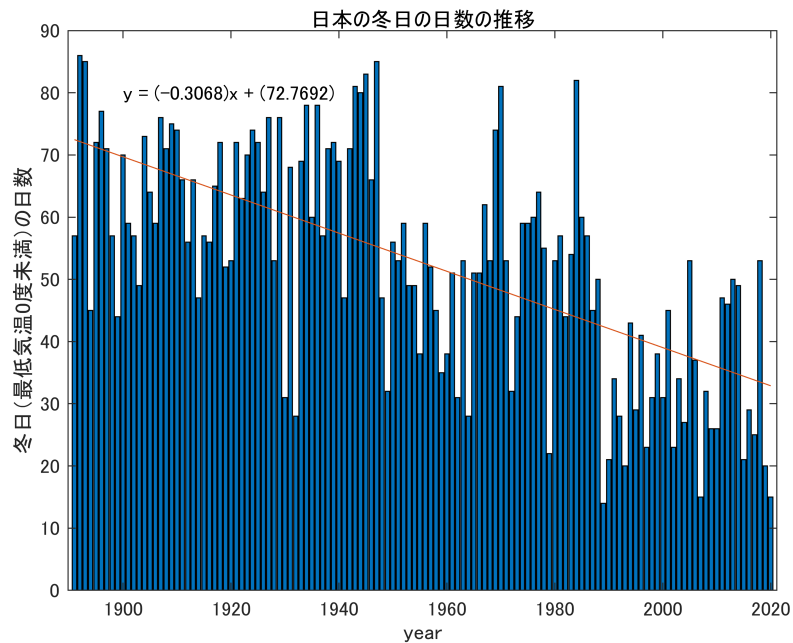
bar(yearnum,Tm0num)
%冬日の日数の線形回帰線 y = px+a(最小二乗法によって求めた線形回帰線)
hold on
p6 = polyfit(yearnum130,Tm0num,1);
y6 = polyval(p6,yearnum130);
plot(yearnum,y6)
s6 = sprintf('y = (%.4f)x + (%.4f)',p6(1),p6(2));

```

```

text(1900,80,s6)
hold off
xlabel('year')
ylabel('冬日（最低気温0度未満）の日数')
title('日本の冬日の日数の推移')

```

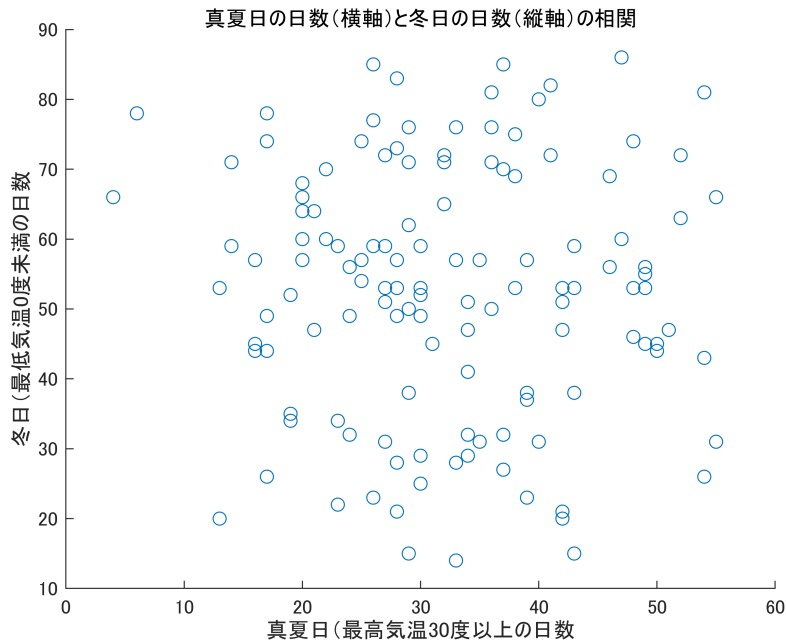


```

%相関係数（真夏日と冬日）
scatter(T30num,Tm0num)
co30and0 = corrcoef(T30num,Tm0num);
xlabel('真夏日（最高気温30度以上の日数）')
ylabel('冬日（最低気温0度未満の日数）')
title('真夏日の日数（横軸）と冬日の日数（縦軸）の相関')

```





## 考察:

実行結果に対して、冒頭の考察記述欄に考察を自由に記述してみましょう。

## 保存(レポート化):

ここまでの作業お疲れさまでした。最後に作業の結果を残しておきましょう。

このプログラムは実行結果とともにレポート出力できます。

[ライブエディター]タブの[エクスポート]ボタンを押して、PDFを選んでエクスポートしてください。

保存されたPDF形式のファイルをコンテスト窓口に送付してください。

Copyright 2021 The MathWorks, Inc.