Alan Dautov, Andres O'Connell Matt Farrell

EK 125

Lab Section C3

11 December 2019

EK125 Final Project

In recent years global temperatures have been increasing at an unprecedented rate. This phenomenon known as Global Warming is an issue in which the earth heats up due to the presence of green-house gasses in the atmosphere trapping heat from sunlight in the atmosphere that would otherwise escape back into space. Many of these green-house gasses are emitted by human technology. As humans emit more of these gasses, the global temperature increases, destabilizing the climate.

If we are able to stop Global Warming it will benefit humanity as a whole. However, it will benefit everyone in different ways. Depending on a person's location and lifestyle the effects of global warming will change.

To start those who live in more Northern Climates, the ice caps will stop melting. Allowing those ecosystems to stabilize. Those who hunt in those areas, will begin to see more stable animal populations. Those who rely on fishing will benefit greatly as migration patterns will normalize with a more constant temperature. Farmers will have more consistent growing seasons, as slight changes in temperature will not affect the viability of crops. Even those in urban areas, that are not as directly affected by a change in climate, will benefit as more consistent yields and seasons will lead to more consistent business cycles and a more stable economy. Finally, those in developing countries will benefit the most as those countries often lack the economic or structural stability to withstand radical changes to the environment.

In order to solve the issue of Global Warming, we have decided to use Average temperature of certain countries as well as the emission data of those countries. We can then establish the correlations between the gas emissions and the increase in temperature. The goal would be to reduce emissions to a level that will result in a more normalized global temperature. We can do this by finding a time range in which gas emissions correlated to a minimal increase in temperature annually. Then we through policy and public awareness we could set a goal to reduce gas emissions to the amount they were in that time period. If the emissions that cause global warming are reduced, then the change in temperature will likely be reduced.

Sources:

Average Temperature:

https://www.kaggle.com/berkeleyearth/climate-change-earth-surface-temperature-data#GlobalLandTemperaturesByMajorCity.csv

CO2 Emissions:

https://www.kaggle.com/srikantsahu/co2-and-ghg-emission-data x

In May, Google introduced an auto complete system for gmail. This system would predict what the user would want to type and autocomplete that sentence. Google calls this program “Smart Compose”. However, Google has opted not to include gendered pronouns in this new system. This was done out of fear that the machine may predict someone’s gender by using a gendered pronouns. A research scientist that worked for google realized there was an issue in January. He was writing a letter to an investigator and the program misgendered the recipient. Google’s smart compose could be good for business, the technology shows off Google’s understanding of machine learning and artificial intelligence. However, the possibility of the program misgendering someone, especially in traditionally male dominated fields such as finance, could be disastrous. This could incur bad publicity and could lead to backlash against the company. As a result of the wide set of users, can generally learn whether an investigator is more likely to be a man or a woman. Thus, the program can internalize and help perpetuate the biases of society.

Work Cited:

Dave, Paresh. “Fearful of Bias, Google Blocks Gender-Based Pronouns from New AI Tool.” Reuters, Thomson Reuters, 27 Nov. 2018, www.reuters.com/article/us-alphabet-google-ai-gender/fearful-of-bias-google-blocks-gender-based-pronouns-from-new-ai-tool-idUSKCN1NW0EF.

Editor – /Users/alandautov/Documents/MATLAB/impemis.m

impemis.m | imptemp.m | temp_uk.m | emis_uk.m

```
%% Import data from text file
% Script for importing data from the following text file:
%
%    filename: /Users/alandautov/Documents/MATLAB/emission.csv
%
% Auto-generated by MATLAB on 08-Dec-2019 23:04:57

%% Setup the Import Options and import the data
opts = delimitedTextImportOptions("NumVariables", 268);

% Specify range and delimiter
opts.DataLines = [1, Inf];
opts.Delimiter = ",";

% Specify column names and types
opts.VariableNames = ["Country", "VarName2", "VarName3", "VarName4", "VarName5", "VarName6", "VarName7", "VarName8", "VarName9", "
opts.VariableTypes = ["string", "string", "string", "string", "string", "string", "string", "string", "string", "string", "string"

% Specify file level properties
opts.ExtraColumnsRule = "ignore";
opts.EmptyLineRule = "read";

% Specify variable properties
opts = setvaropts(opts, ["Country", "VarName2", "VarName3", "VarName4", "VarName5", "VarName6", "VarName7", "VarName8", "VarName9"
opts = setvaropts(opts, ["Country", "VarName2", "VarName3", "VarName4", "VarName5", "VarName6", "VarName7", "VarName8", "VarName9"

% Import the data
emission1 = readmatrix("/Users/alandautov/Documents/MATLAB/emission.csv", opts);


%% Clear temporary variables
clear opts
```

"11420688"	"12996208"	"14443488"	"16031881"	"17755991"	"19520701"	"21314605"	"23258733"
"258905568"	"274880608"	"290921600"	"308113403"	"324911940"	"342252778"	"363857324"	"386497266"
"0"	"0"	"0"	"13578842"	"27716191"	"42295291"	"57596433"	"74005887"
"0"	"0"	"0"	"0"	"0"	"0"	"0"	"0"
"0"	"0"	"0"	"0"	"0"	"0"	"0"	"0"
"1344688"	"1623152"	"1916272"	"2329977"	"2751040"	"3157477"	"3585927"	"4018071"
"0"	"0"	"0"	"259196536"	"529017008"	"807235379"	"1099196777"	"1412293257"
"0"	"0"	"0"	"10992"	"21984"	"32976"	"51296"	"73280"
"42466159376"	"43036556240"	"43592491296"	"44139097428"	"44723117446"	"45311732247"	"45904663746"	"46508049409
"1.08E+11"	"1.10E+11"	"1.13E+11"	"1.16E+11"	"1.19E+11"	"1.22E+11"	"1.25E+11"	"1.28E+11"
"25058096"	"29491536"	"33470640"	"37858418"	"42172946"	"46289707"	"50296608"	"54611433"
"0"	"0"	"0"	"45961762"	"93813275"	"143159453"	"194948964"	"250489714"
"0"	"0"	"0"	"0"	"0"	"0"	"40304"	"73280"
"645985184"	"715575536"	"770037232"	"835186566"	"892203396"	"944083178"	"998139288"	"1054291027"
"134014464"	"134014464"	"134014464"	"134014464"	"134014464"	"134014464"	"134014464"	"134014464"
"0"	"0"	"0"	"0"	"0"	"0"	"0"	"0"
"2.72E+11"	"2.80E+11"	"2.88E+11"	"2.97E+11"	"3.07E+11"	"3.16E+11"	"3.26E+11"	"3.36E+11"
"9497088"	"12453936"	"14337232"	"16938672"	"20569696"	"23233424"	"27117264"	"30033808"
"0"	"0"	"0"	"4567766"	"8922992"	"12631948"	"16214629"	"19659112"
"93593216"	"93593216"	"93593216"	"99826414"	"105769567"	"110830845"	"115719813"	"120420197"

The emissions data file was imported to MATLAB through the import tool and was organized in a matrix of year, country, and emissions corresponded to each country, and was stored in the variable **emission1** impemis.m | imptemp.m | temp_uk.m | emis_uk.m

```matlab
%% Import data from text file
% Script for importing data from the following text file:
%
%    filename: /Users/alandautov/Documents/MATLAB/temp.csv
%
% Auto-generated by MATLAB on 08-Dec-2019 00:32:37

%% Setup the Import Options and import the data
opts = delimitedTextImportOptions("NumVariables", 1);

% Specify range and delimiter
opts.DataLines = [1, Inf];
opts.Delimiter = "";

% Specify column names and types
opts.VariableNames = "temp";
opts.VariableTypes = "string";

% Specify file level properties
opts.ExtraColumnsRule = "ignore";
opts.EmptyLineRule = "read";

% Specify variable properties
opts = setvaropts(opts, "temp", "WhitespaceRule", "preserve");
opts = setvaropts(opts, "temp", "EmptyFieldRule", "auto");

% Import the data
tbl = readtable("/Users/alandautov/Documents/MATLAB/temp.csv", opts);

%% Convert to output type
temp = tbl.temp;

%% Clear temporary variables
clear opts tbl
```

```
"2011-01-01,22.982,0.46,Zimbabwe"
"2011-02-01,23.166,0.26,Zimbabwe"
"2011-03-01,23.668000000000006,0.183,Zimbabwe"
"2011-04-01,21.759,0.319,Zimbabwe"
"2011-05-01,19.627,0.398,Zimbabwe"
"2011-06-01,16.939,0.599,Zimbabwe"
"2011-07-01,15.803,0.447,Zimbabwe"
"2011-08-01,17.883,0.328,Zimbabwe"
"2011-09-01,22.901999999999997,0.612,Zimbabwe"
"2011-10-01,24.966,0.441,Zimbabwe"
"2011-11-01,25.521,0.318,Zimbabwe"
"2011-12-01,24.013,0.35600000000000004,Zimbabwe"
"2012-01-01,23.872,0.247,Zimbabwe"
"2012-02-01,24.294,0.305,Zimbabwe"
"2012-03-01,23.596,0.354,Zimbabwe"
"2012-04-01,20.349,0.462,Zimbabwe"
"2012-05-01,19.712,0.312,Zimbabwe"
"2012-06-01,16.631,0.277,Zimbabwe"
"2012-07-01,16.048,0.7829999999999999,Zimbabwe"
"2012-08-01,18.946,1.127,Zimbabwe"
```

The temperature data file was imported to MATLAB using the import tool and was organized in a table which included the country, the year and the mont, and the average temperature during that month, and was stored in the variable **temp.**

```
count = 0;
for i = 1:size(temp)
    if contains(temp(i), "China") && ...
            ( (contains(temp(i), "-01-01")) || (contains(temp(i), "-07-01")))
        count = count + 1;
    end
end
count = count/2;
china_year_temp_jan = zeros((count),1);
china_year_temp_jul = zeros((count),1);
ind_jan = 1;
ind_jul = 1;
```

```
for j = 1:size(temp)
    if contains(temp(j), "China")
            if (contains(temp(j), "-01-01"))
                if ((strlength(temp(j))) == 18)
                        [date, rest] = strtok(temp(j), ',');
                        [avgtemp, rest2] = strtok(rest, ',');
                        [year, restdate] = strtok(date, '-');
                        [month, restdate2] = strtok(restdate, '-');
                        fixed_date = year + "." + month;
                        fixed_date1 = str2double(fixed_date);
                        china_year_temp_jan(ind_jan, 1) = fixed_date1;
                        china_year_temp_jan(ind_jan, 2) = NaN;
                        ind_jan = ind_jan + 1;
                else
                        [date, rest] = strtok(temp(j), ',');
                        [avgtemp, rest2] = strtok(rest, ',');
                        [year, restdate] = strtok(date, '-');
                        [month, restdate2] = strtok(restdate, '-');
                        fixed_date = year + "." + month;
                        fixed_date1 = str2double(fixed_date);
                        china_year_temp_jan(ind_jan, 1) = fixed_date1;
                        china_year_temp_jan(ind_jan, 2) = str2double(avgtemp);
                        ind_jan = ind_jan + 1;
                end
            end
        elseif (contains(temp(j), "-07-01"))
            if ((strlength(temp(j))) == 26)
                    [date, rest] = strtok(temp(j), ',');
                    [avgtemp, rest2] = strtok(rest, ',');
                    [year, restdate] = strtok(date, '-');
                    [month, restdate2] = strtok(restdate, '-');
                    fixed_date = year + "." + month;
                    fixed_date1 = str2double(fixed_date);
                    china_year_temp_jul(ind_jul, 1) = fixed_date1;
                    china_year_temp_jul(ind_jul, 2) = NaN;
                    ind_jul = ind_jul + 1;
            else
                    [date, rest] = strtok(temp(j), ',');
                    [avgtemp, rest2] = strtok(rest, ',');
                    [year, restdate] = strtok(date, '-');
                    [month, restdate2] = strtok(restdate, '-');
                    fixed_date = year + "." + month;
                    fixed_date1 = str2double(fixed_date);
                    china_year_temp_jul(ind_jul, 1) = fixed_date1;
                    china_year_temp_jul(ind_jul, 2) = str2double(avgtemp);
                    ind_jul = ind_jul + 1;
            end
        end
end
```

In order to organize the data by country, the number of occurrences of country (here **China**) was found in order to preallocate the matrix of temperature and year and month (January or July).

Then, the in order to store the the temperature to the right month and year, two **if** statements were used. Also, if the temperature data was missing, then **NaN** would be stored instead of 0. Since the data was in the table as a string, the **strtok** function was used in order to separate the date and the temperature. The data for temperature in January was stored in the variable **china_year_temp_jan** and the temperature in July was stored in the variable **china_year_temp_july.** The data was in the following format:

```
china_year_temp_jan =      china_year_temp_jul =

   1.0e+03 *                  1.0e+03 *

    1.8210   -0.0075           1.8211    0.0186
    1.8220   -0.0082           1.8221    0.0198
    1.8230   -0.0091           1.8231    0.0196
    1.8240       NaN           1.8241       NaN
    1.8250       NaN           1.8251       NaN
```

Note: The data stored in these variables was written as scientific notation, but when working with data the numbers are correct.

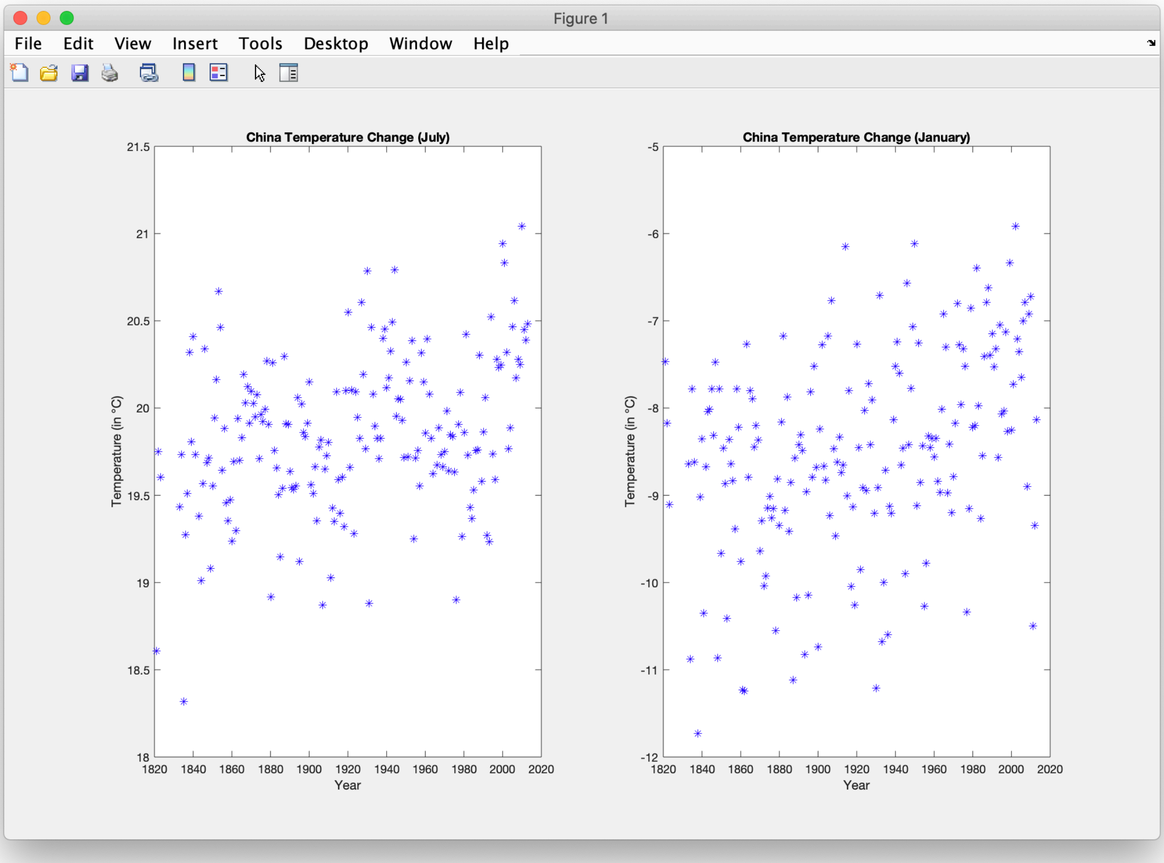
```
china_emis = zeros(2, 267);
china_emis(1, :) = emission1(1, 2:end);
china_emis(2, :) = emission1(47, 2:end);
for i = 1:267
    if (china_emis(2, i) == 0)
        china_emis(2, i) = NaN;
    end
end
```

In order to organize the emissions data the above method was used. The number of points in the data table is know, as well as the index in which the country (here China) is located. The data was organized like this:

```
Columns 169 through 180

  0.0000    0.0000    0.0000    0.0000    0.0000
  0.0027    0.0029    0.0032    0.0034    0.0038

Columns 181 through 192

  0.0000    0.0000    0.0000    0.0000    0.0000
  0.0066    0.0069    0.0074    0.0079    0.0084
```

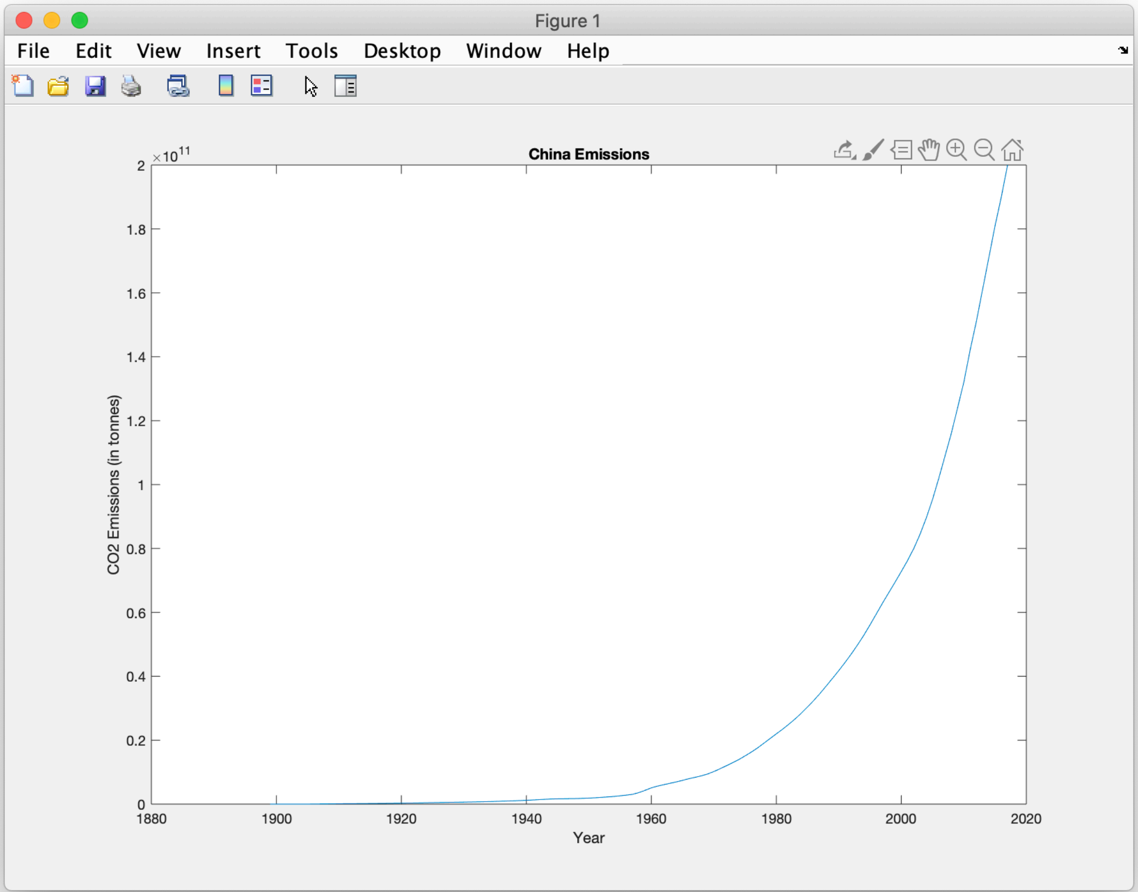
```
x = china_year_temp_jul(:, 1);
y = china_year_temp_jul(:, 2);
subplot(1,2,1)
plot(x,y, 'b*')
title('China Temperature Change (July)')
xlabel('Year')
ylabel('Temperature (in °C)')


a = china_year_temp_jan(:, 1);
b = china_year_temp_jan(:, 2);
subplot(1,2,2)
plot(a,b, 'b*')
title('China Temperature Change (January)')
xlabel('Year')
ylabel('Temperature (in °C)')
```

The data for the temperature and year has been plotted in order to see that the trend of the average temperature of the coldest month (January) and the warmest month (July) over the years:



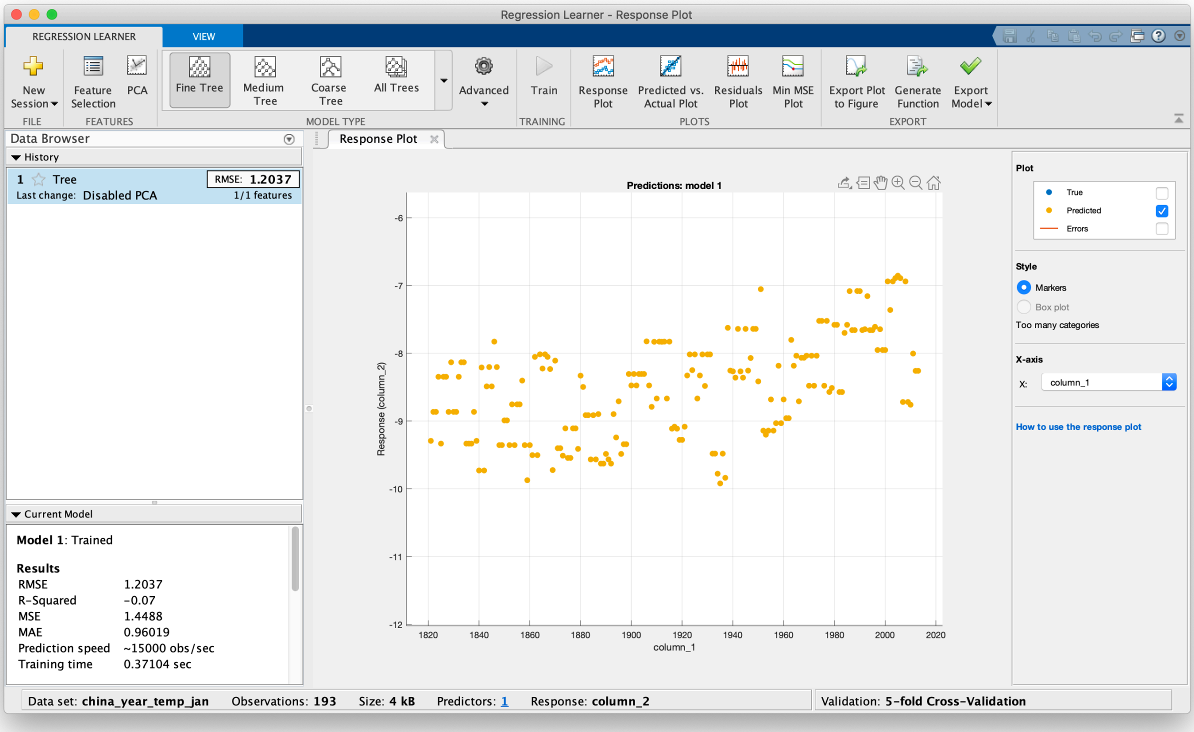
```
x = china_emis(1,:);
y = china_emis(2,:);
plot(x,y)
title('China Emissions')
xlabel('Year')
ylabel('CO2 Emissions (in tonnes)')
```

The emissions were plotted as well, in order to see the trend in emissions change over the years:

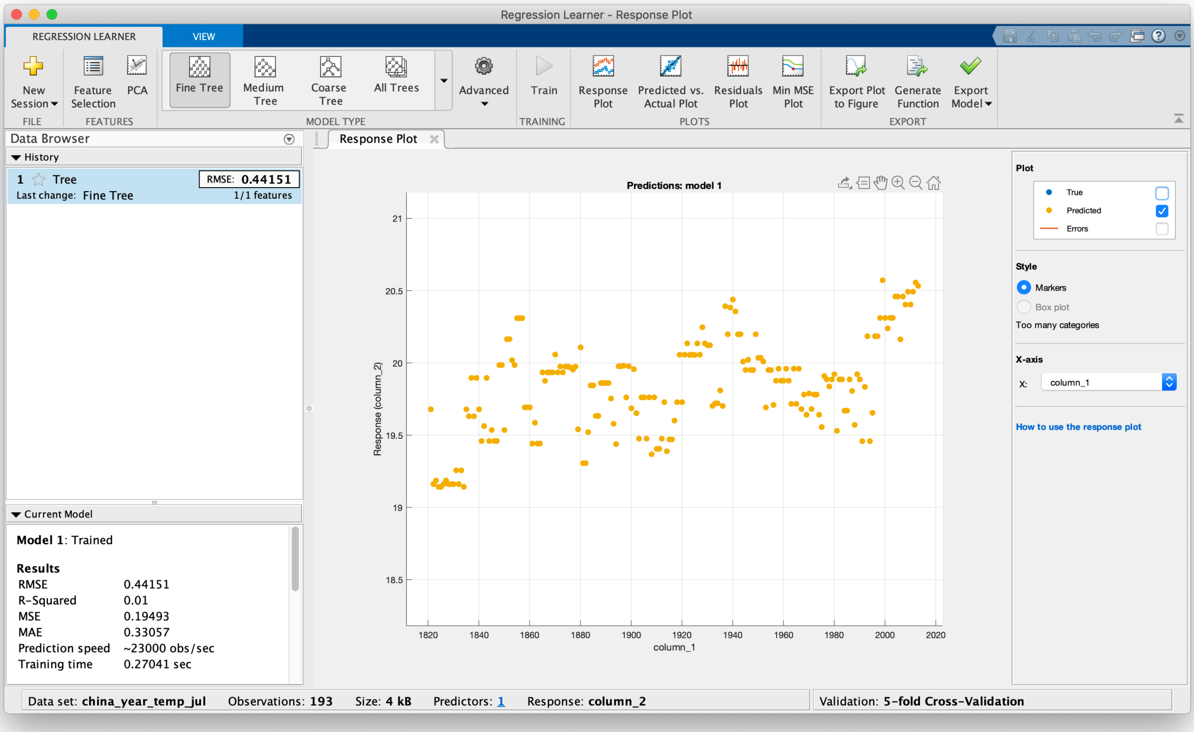


It is obvious that the emissions were only **increasing** over the years.

Regression learner for temperature in China (January):



Regression learner for temperature in China (July):



It is seen through regressions learner that there is correlation between the year and temperature: over the years the temperature in both months has only been increasing.

THE SAME CODE HAS BEEN APPLIED FOR THE FOLLOWING COUNTRIES:

Germany, India, Russia, United Kingdom, and United States. The only difference is the **if statement** that uses the name of the country (in the above code it was China) in order to organize the temperature and the month, and the emissions data for each country. The next pages are going to show the **results,** but **not the code.**

GERMANY

Date and temperature matrices:

```
germany_year_temp_jan =

   1.0e+03 *

    1.7440        NaN
    1.7450    -0.0025
    1.7460        NaN
    1.7470        NaN
    1.7480        NaN
    1.7490        NaN
    1.7500    -0.0004
```

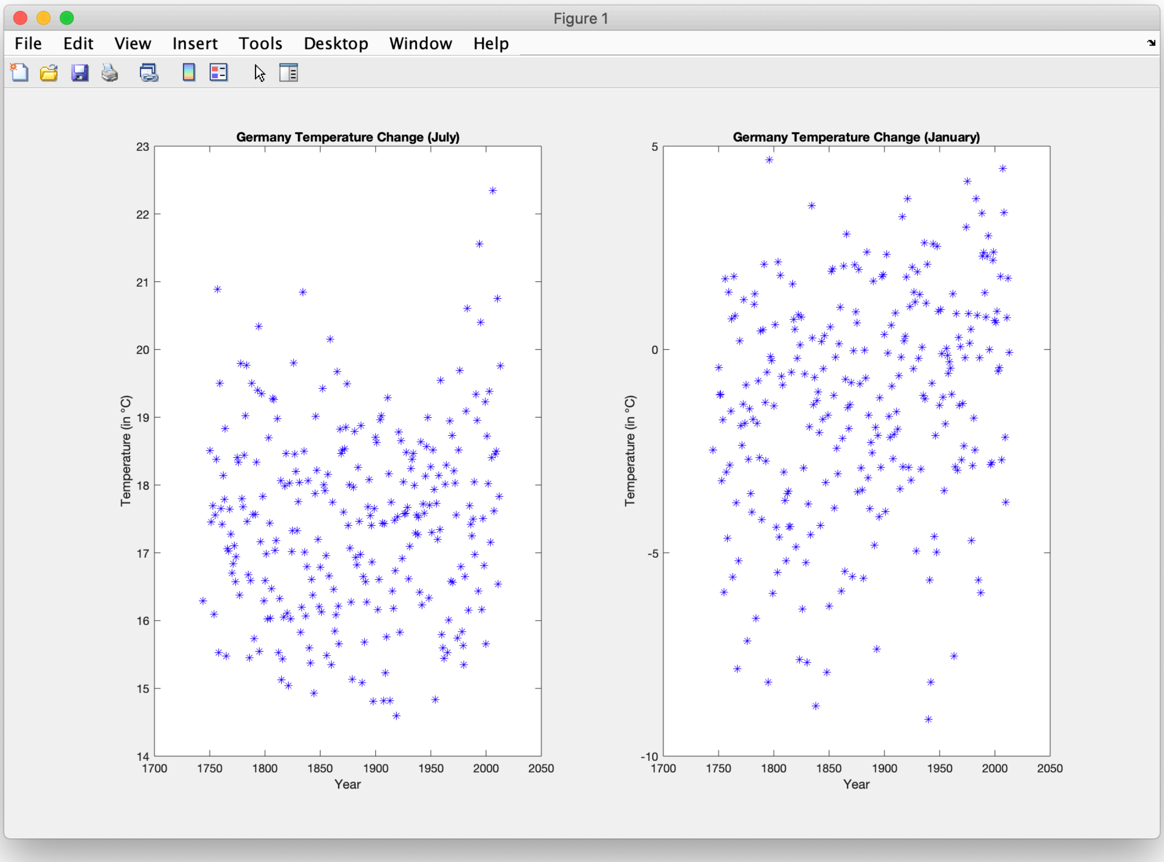
```
germany_year_temp_jul =

   1.0e+03 *

    1.7441    0.0163
    1.7451       NaN
    1.7461       NaN
    1.7471       NaN
    1.7481       NaN
    1.7491       NaN
    1.7501    0.0185
```

Emissions:

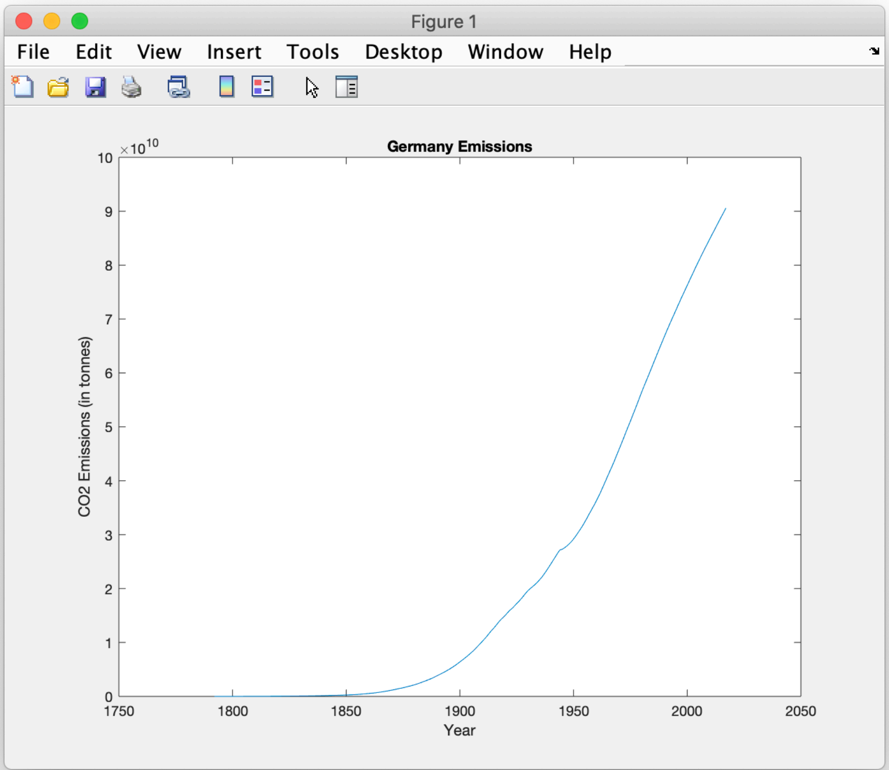
```
 Columns 73 through 84

   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
   0.0047    0.0050    0.0054    0.0059    0.0063    0.0068

 Columns 85 through 96

   0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
   0.0098    0.0103    0.0109    0.0116    0.0123    0.0132
```

Note: Command window outputs the numbers in scientific notations, but when the data is analyzed the correct values are applied.

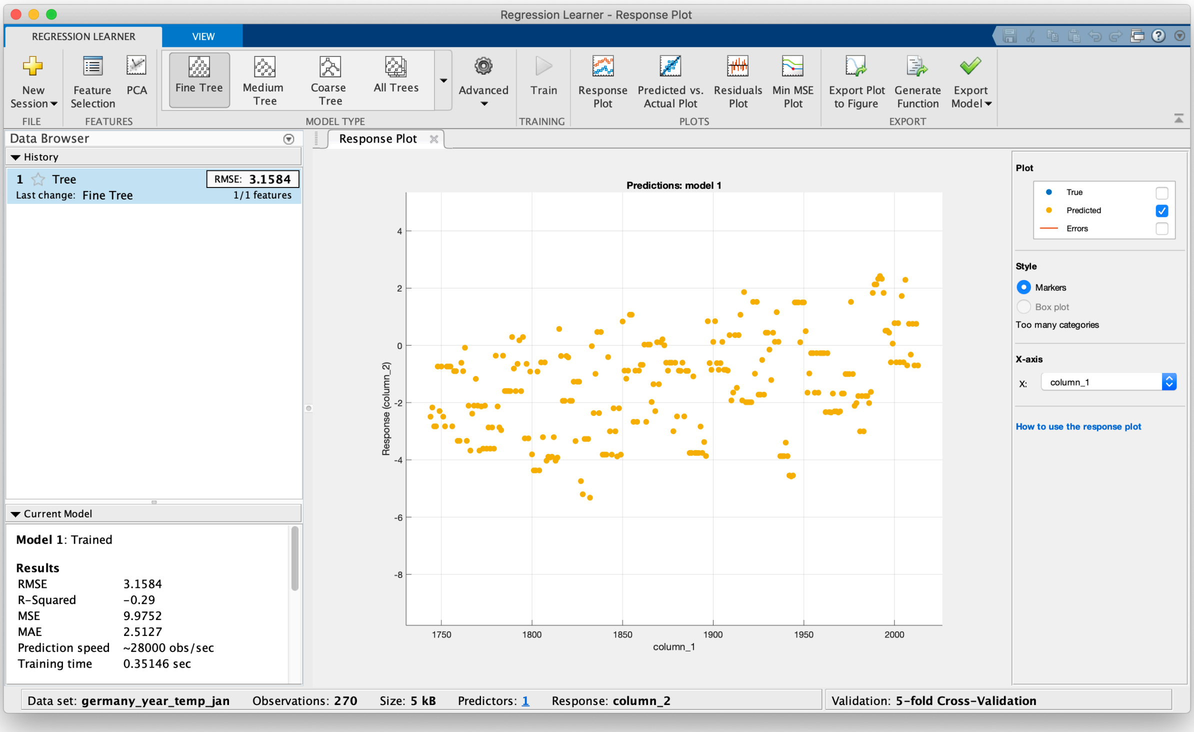
The plotted temperature change over the years:



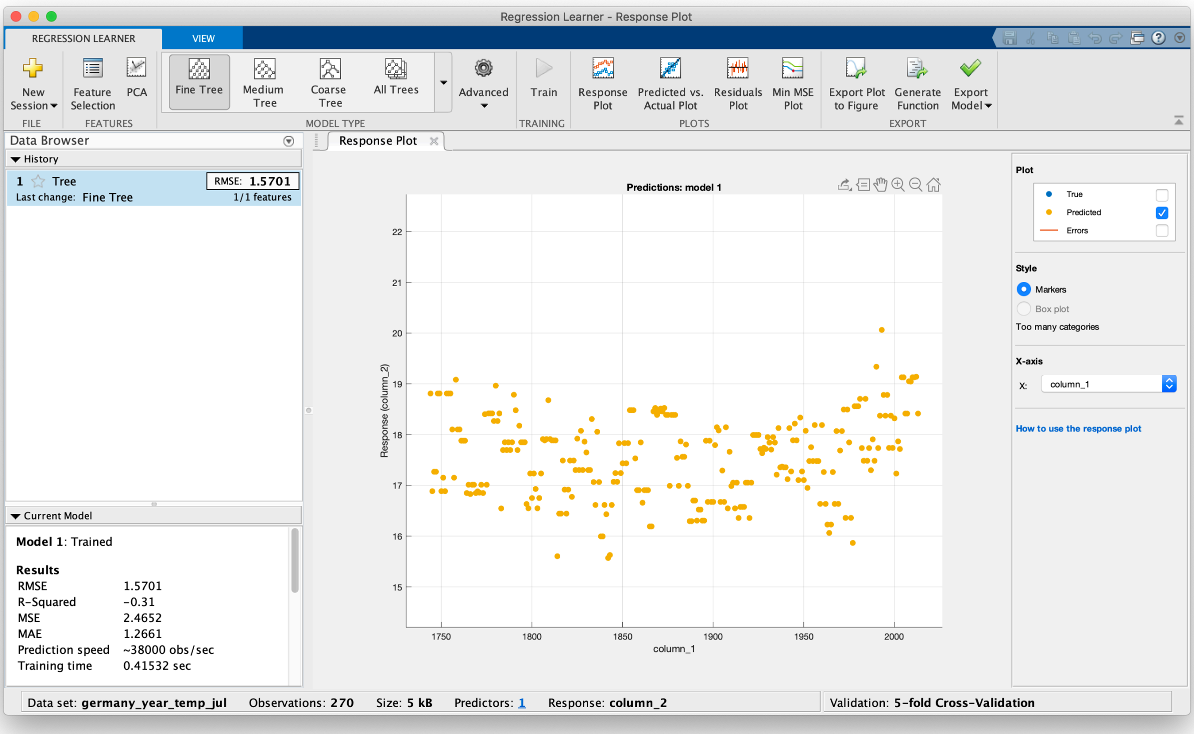
The plotted emissions change over the years:



The regression learner for temperature change in January:



The regression learner for temperature change in July:



India

Date and temperature matrices:

```
india_year_temp_jan =

   1.0e+03 *

    1.7960    0.0170
    1.7970       NaN
    1.7980    0.0172
    1.7990    0.0166
    1.8000    0.0171
```

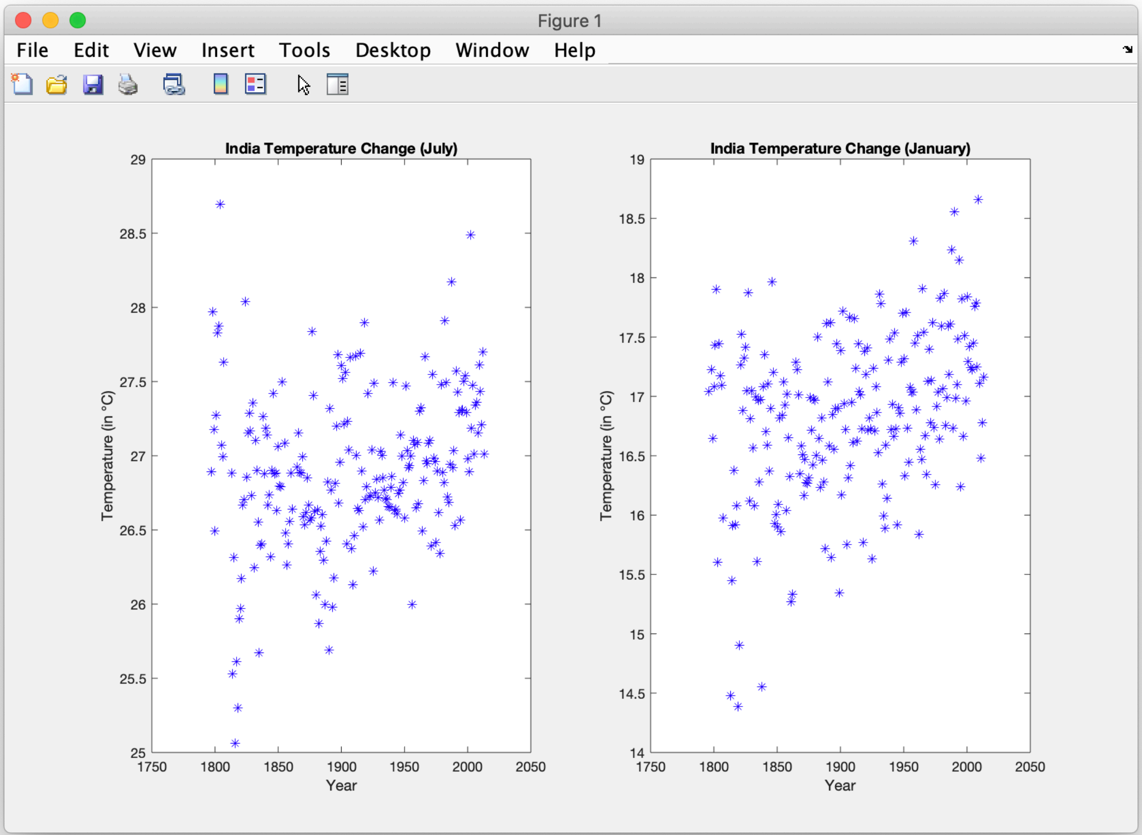
```
india_year_temp_jul =

   1.0e+03 *

    1.7961       NaN
    1.7971    0.0269
    1.7981    0.0280
    1.7991    0.0272
    1.8001    0.0265
```

Emissions:

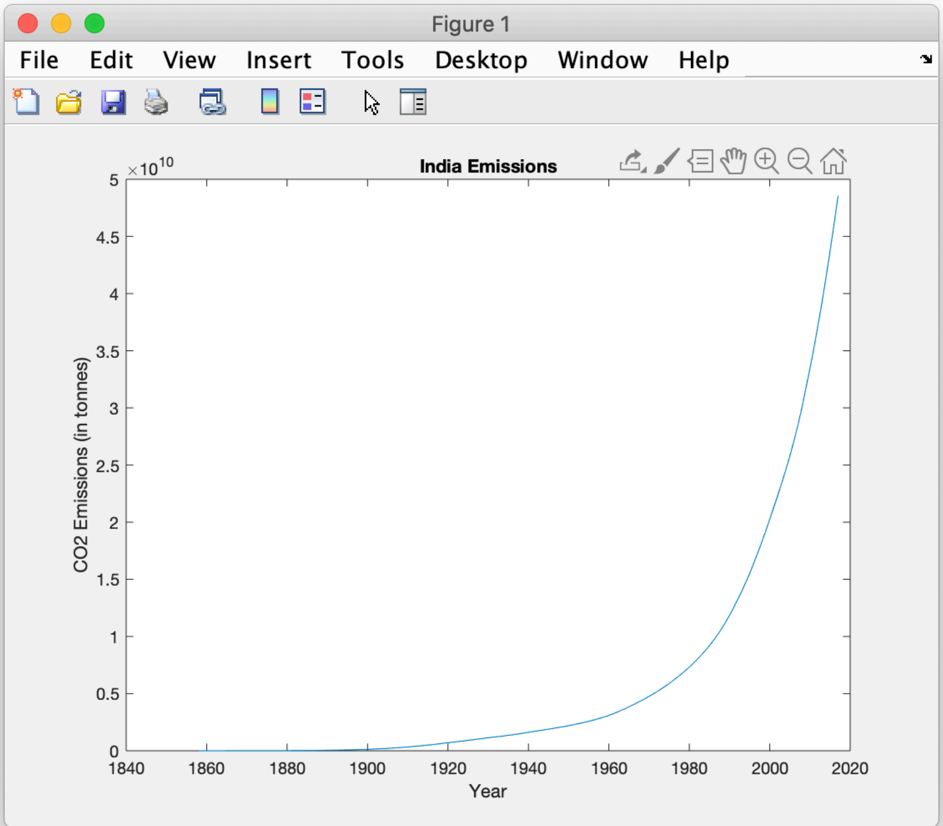
```
  Columns 121 through 132

    0.0000    0.0000    0.0000    0.0000
    0.0006    0.0006    0.0006    0.0006

  Columns 133 through 144

    0.0000    0.0000    0.0000    0.0000
    0.0019    0.0022    0.0024    0.0027
```

Note: Command window outputs the numbers in scientific notations, but when the data is analyzed the correct values are applied.

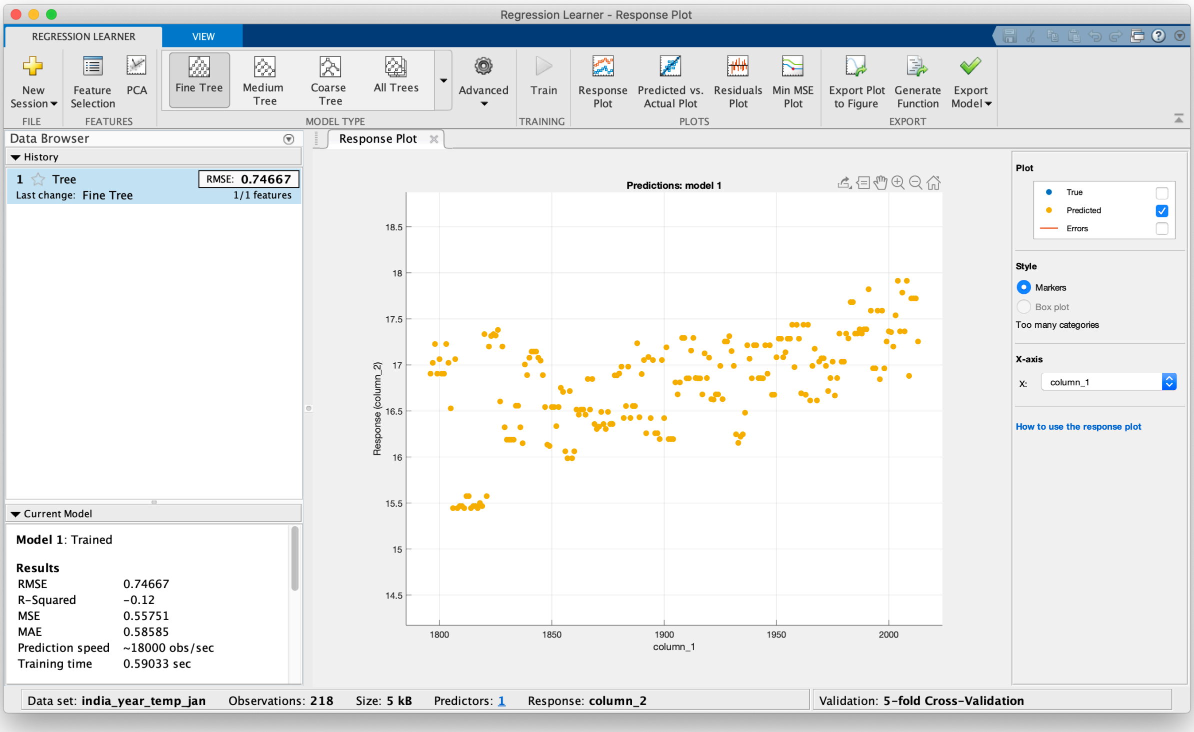
The plotted temperature change over the years:



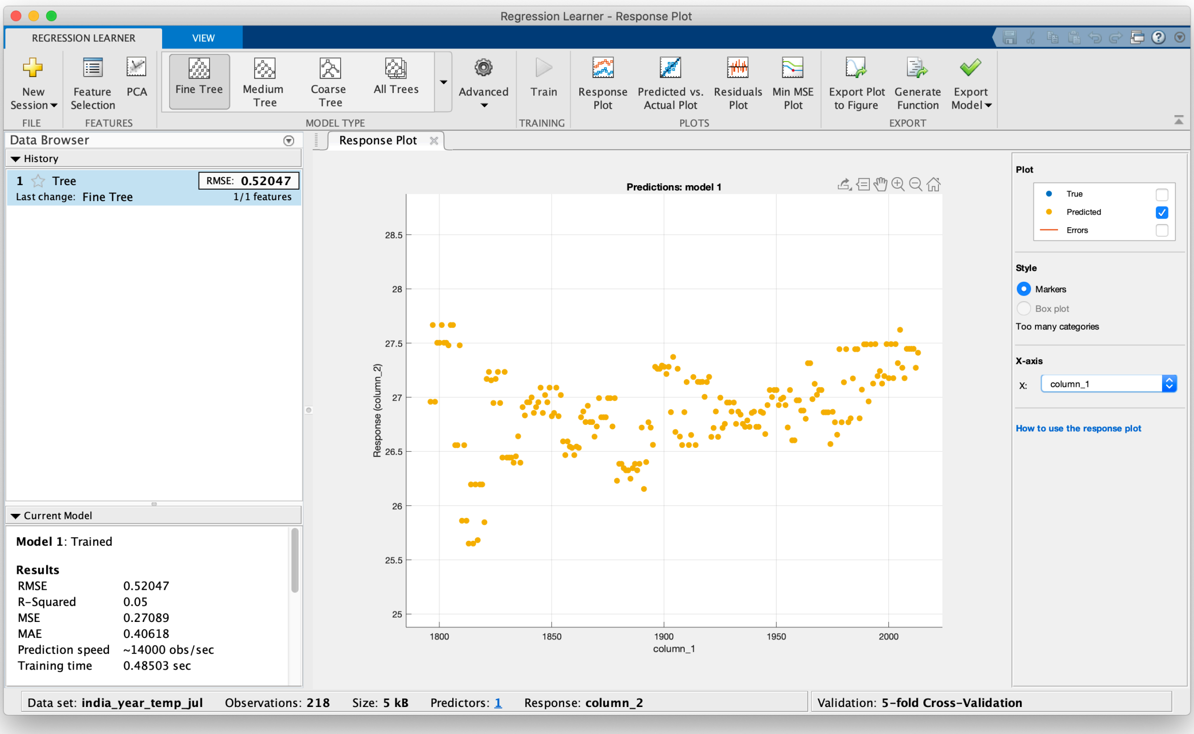
The plotted emissions change over the years:



The regression learner for temperature change in January:



The regression learner for temperature change in July:



Russia

Date and temperature matrices:

```
russia_year_temp_jan =

   1.0e+03 *

    1.8120   -0.0271
    1.8130   -0.0304
    1.8140   -0.0284
    1.8150       NaN
    1.8160       NaN
```

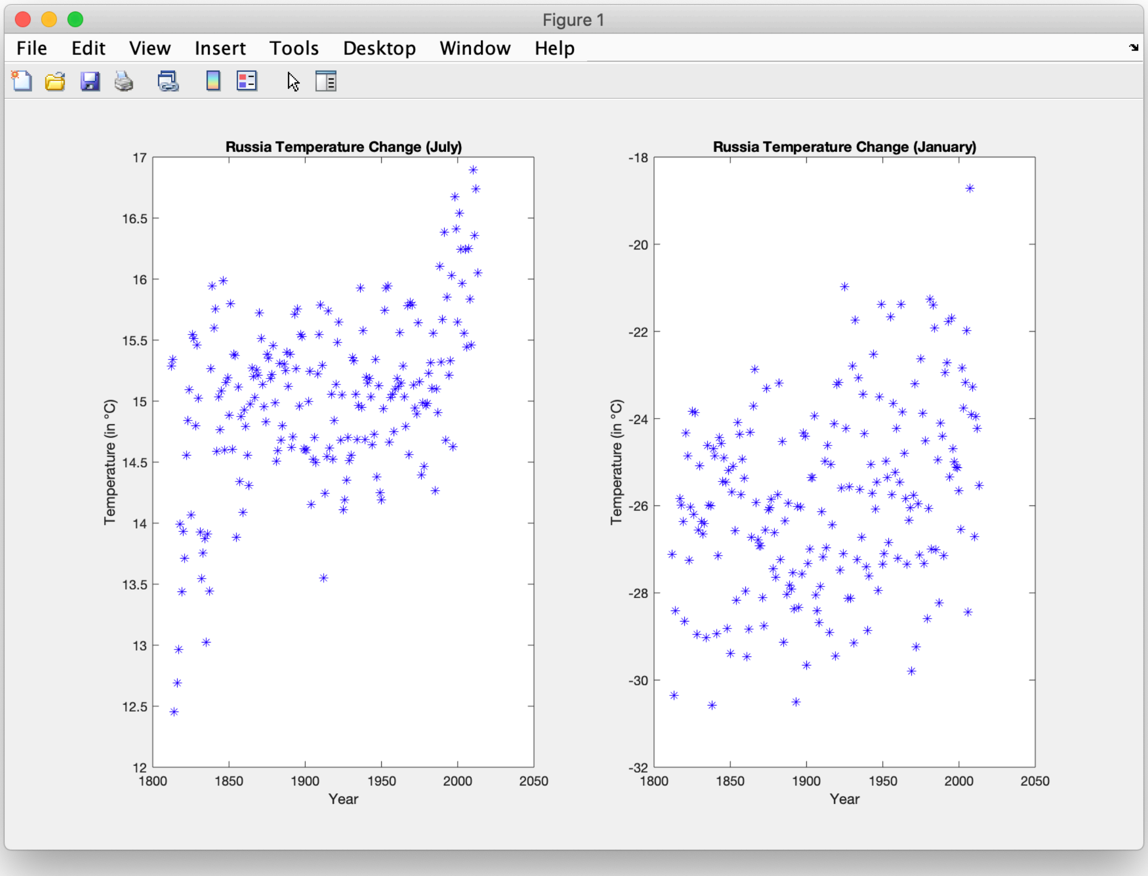
```
russia_year_temp_jul =

   1.0e+03 *

    1.8121    0.0153
    1.8131    0.0153
    1.8141    0.0125
    1.8151       NaN
    1.8161    0.0127
```

Emissions:

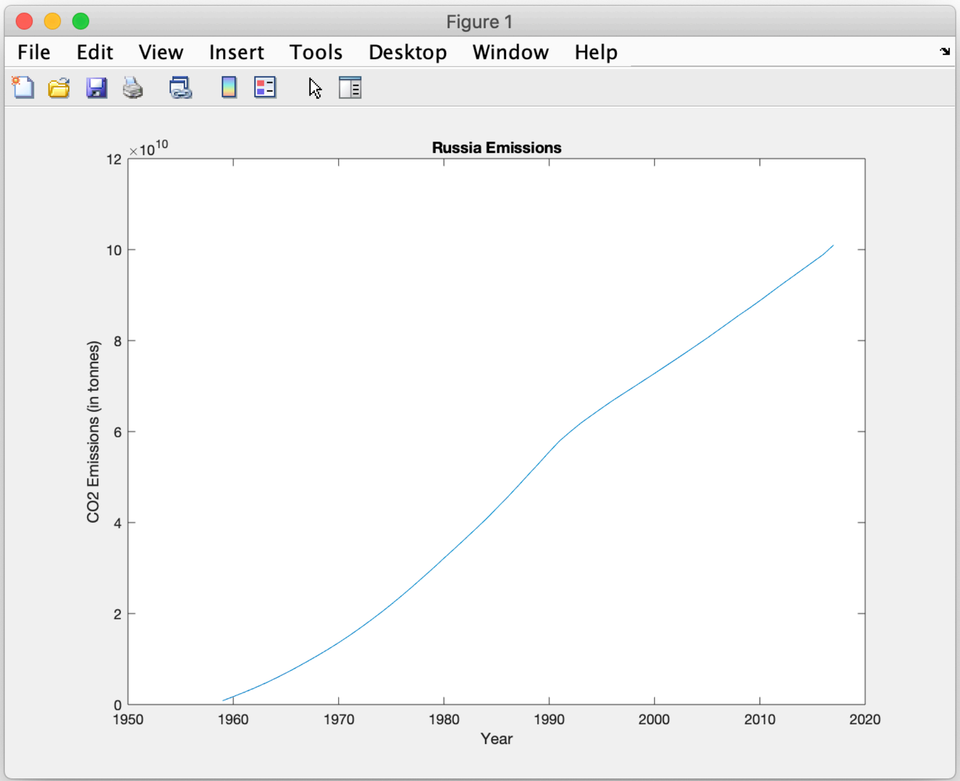
```
  Columns 229 through 240

    0.0000    0.0000    0.0000    0.0000
    0.3002    0.3216    0.3426    0.3640

  Columns 241 through 252

    0.0000    0.0000    0.0000    0.0000
    0.5799    0.5998    0.6187    0.6354
```

Note: Command window outputs the numbers in scientific notations, but when the data is analyzed the correct values are applied.

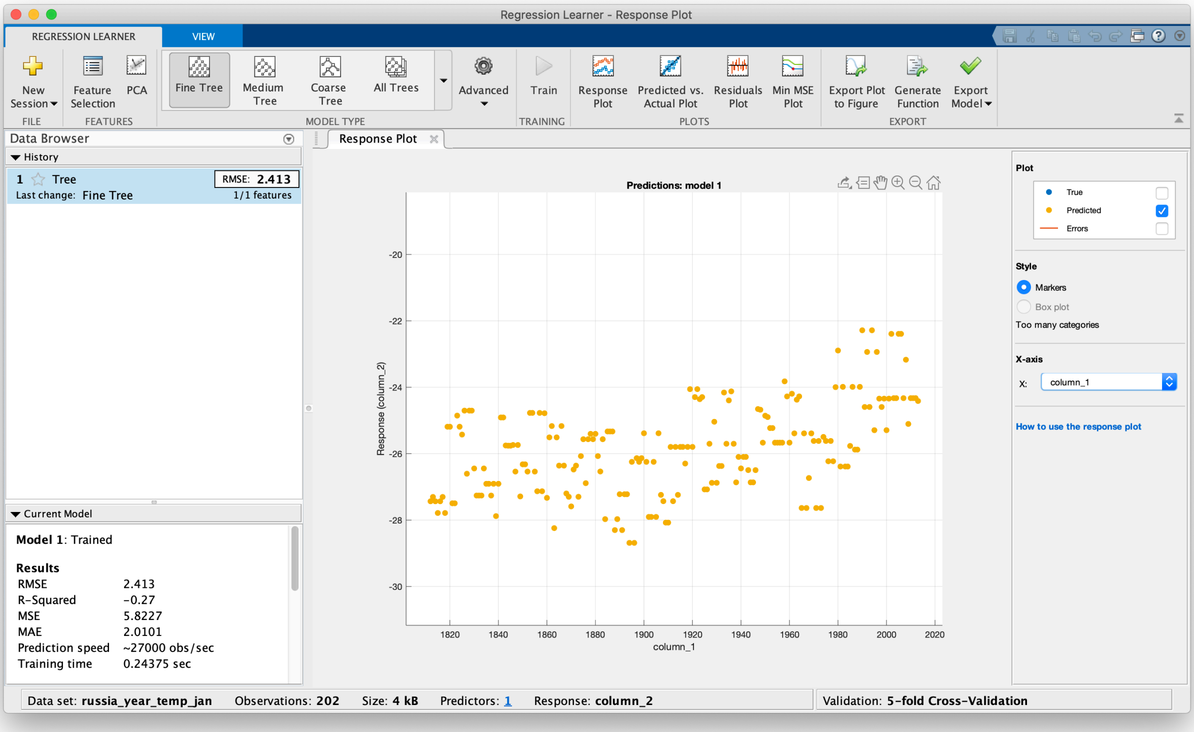
The plotted temperature change over the years:



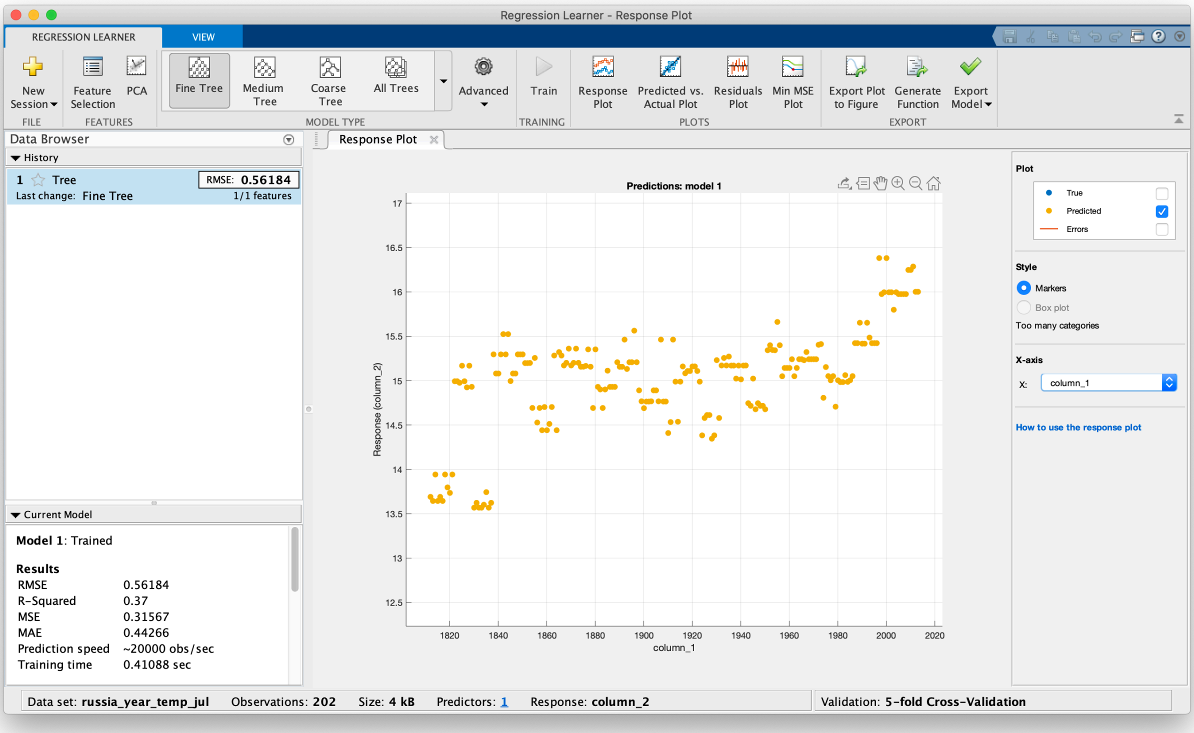
The plotted emissions change over the years:



The regression learner for temperature change in January:



The regression learner for temperature change in July:



United Kingdom

Date and temperature matrices:

```
uk_year_temp_jan =

   1.0e+03 *

    1.7440       NaN
    1.7450    0.0028
    1.7460       NaN
    1.7470       NaN
    1.7480       NaN
    1.7490       NaN
    1.7500    0.0035
```

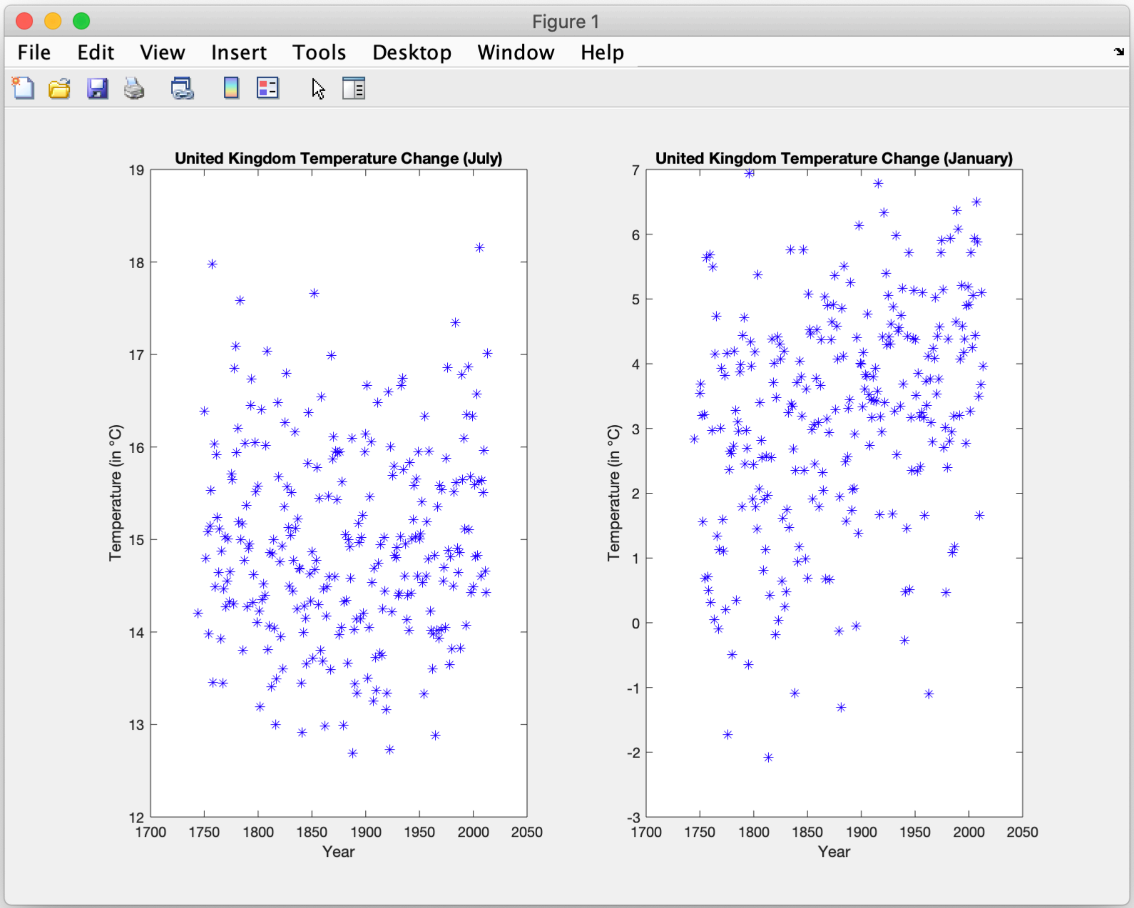
```
uk_year_temp_jul =

   1.0e+03 *

    1.7441    0.0142
    1.7451       NaN
    1.7461       NaN
    1.7471       NaN
    1.7481       NaN
    1.7491       NaN
    1.7501    0.0164
```

Emissions:

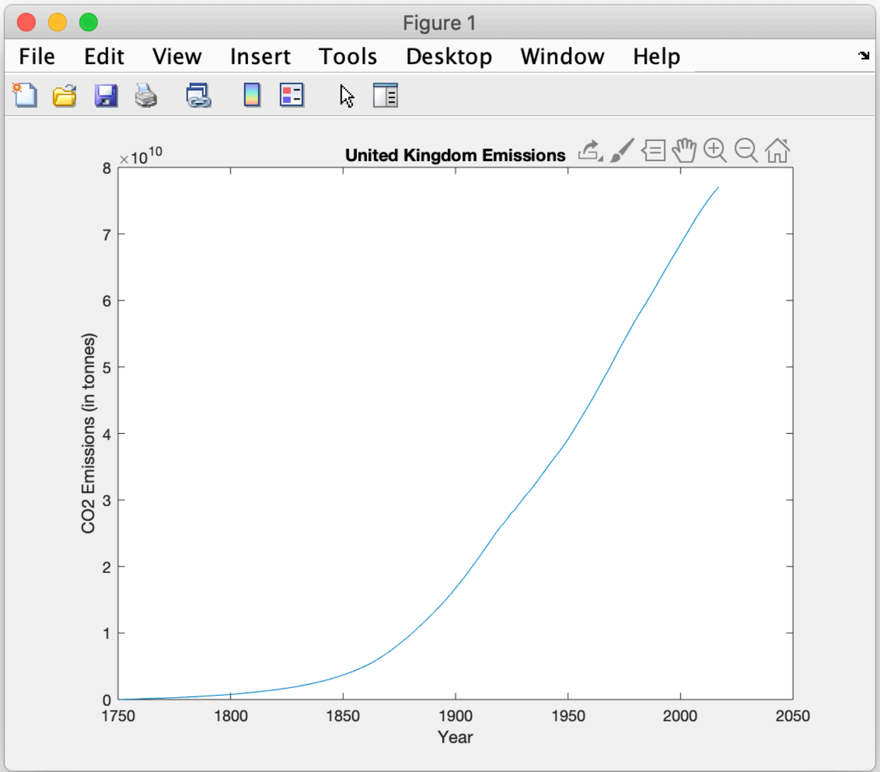
```
  Columns 229 through 240

    0.0000    0.0000    0.0000    0.0000
    5.6476    5.7055    5.7616    5.8164

  Columns 241 through 252

    0.0000    0.0000    0.0000    0.0000
    6.3300    6.3894    6.4473    6.5049

  Columns 253 through 264
```

Note: Command window outputs the numbers in scientific notations, but when the data is analyzed the correct values are applied.

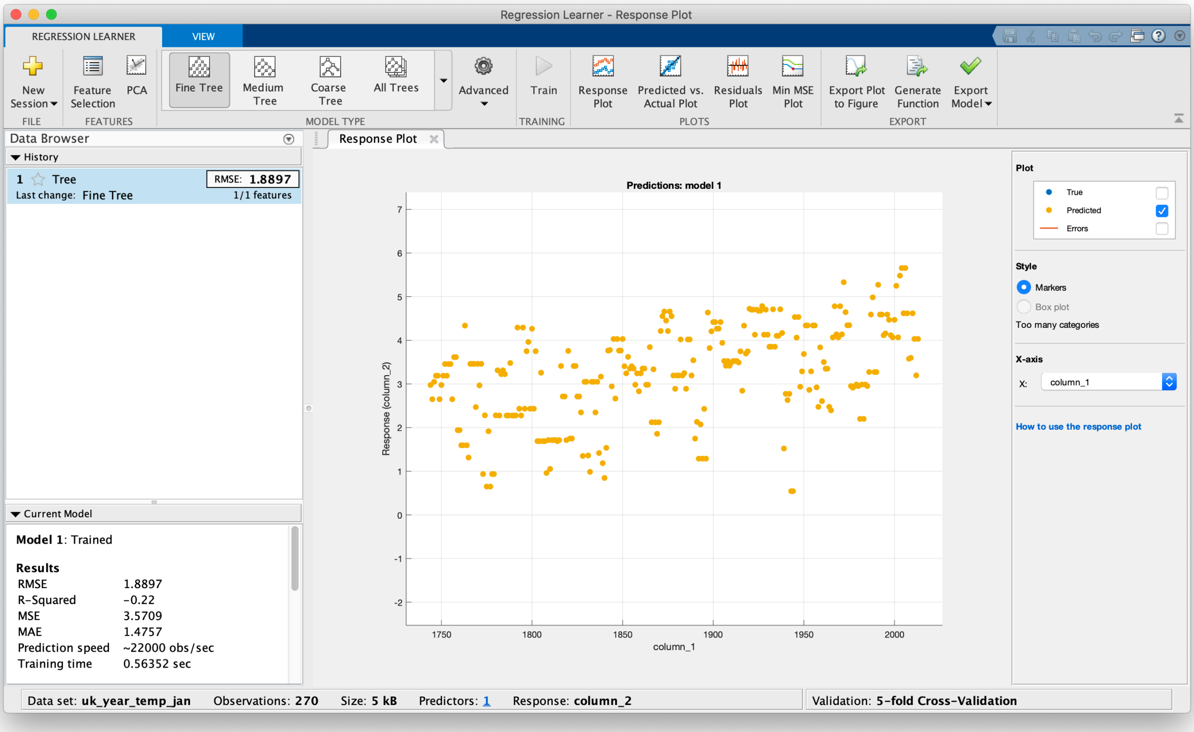
The plotted temperature change over the years:



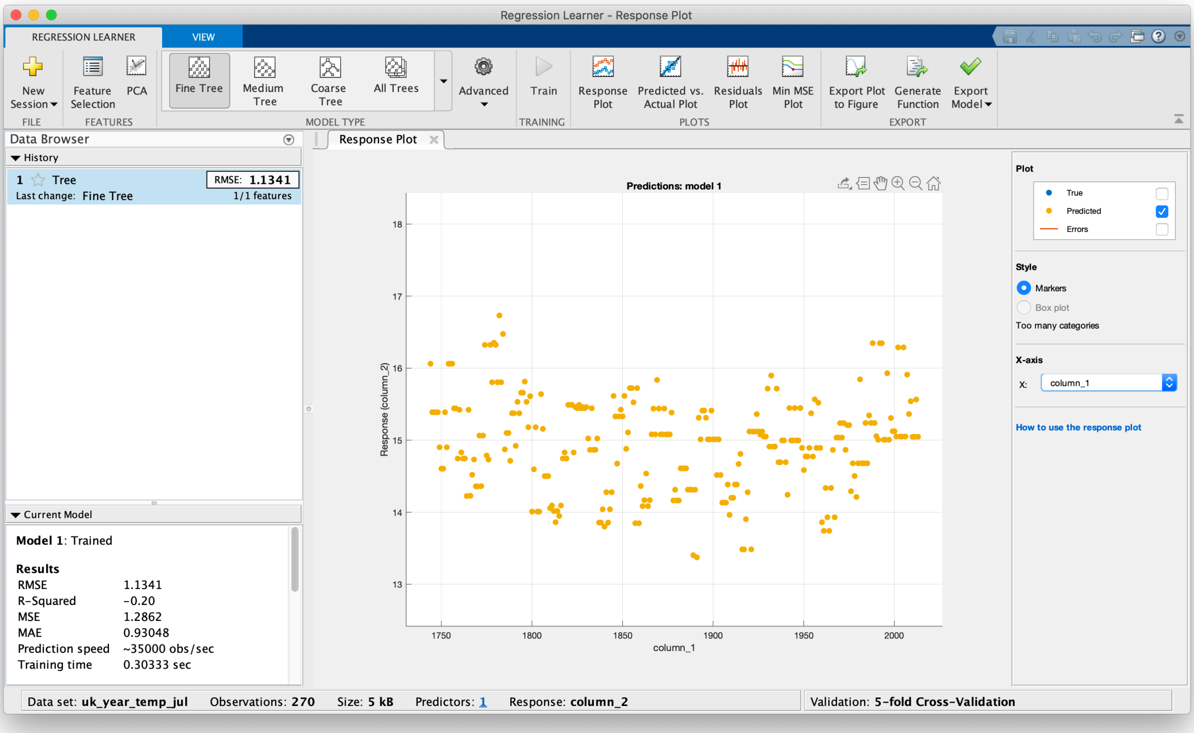
The plotted emissions change over the years:



The regression learner for temperature change in January:



The regression learner for temperature change in July:



United States

Date and temperature matrices:

```
us_year_temp_jan =

   1.0e+03 *

    1.7690   -0.0040
    1.7700       NaN
    1.7710       NaN
    1.7720       NaN
    1.7730       NaN
    1.7740       NaN
    1.7750   -0.0016
```

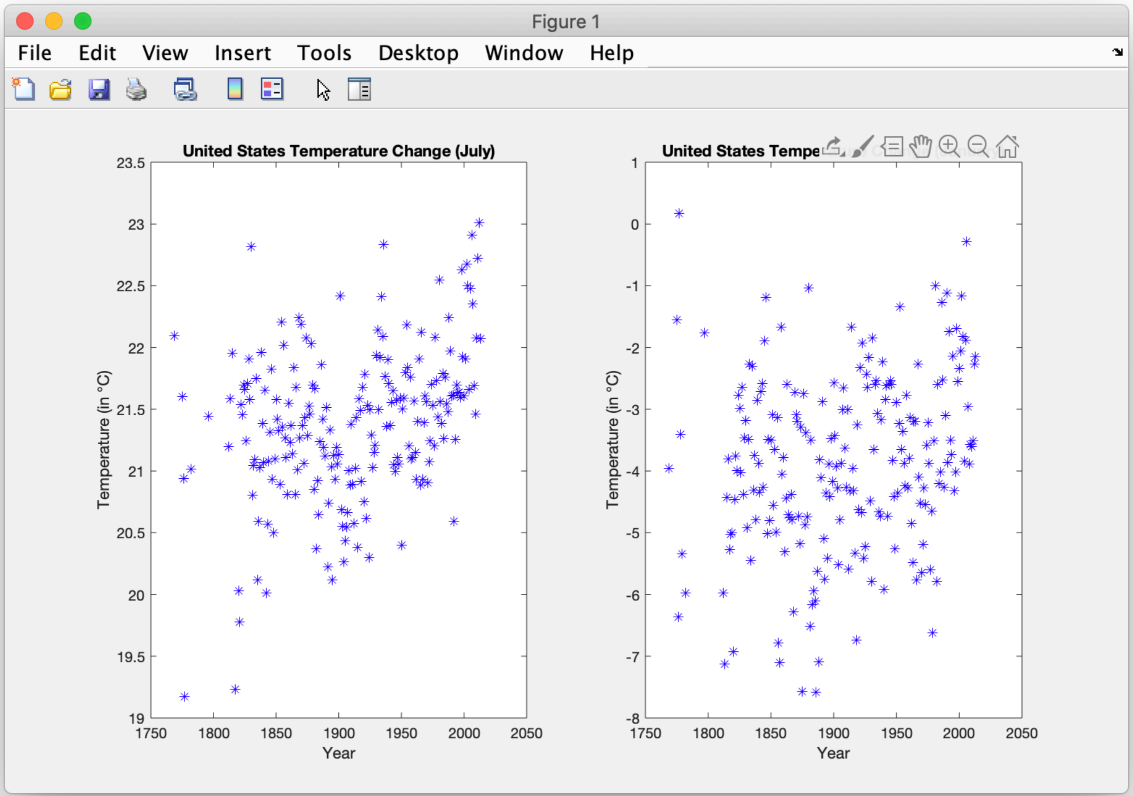
```
us_year_temp_jul =

   1.0e+03 *

    1.7691    0.0221
    1.7701       NaN
    1.7711       NaN
    1.7721       NaN
    1.7731       NaN
```

Emissions:

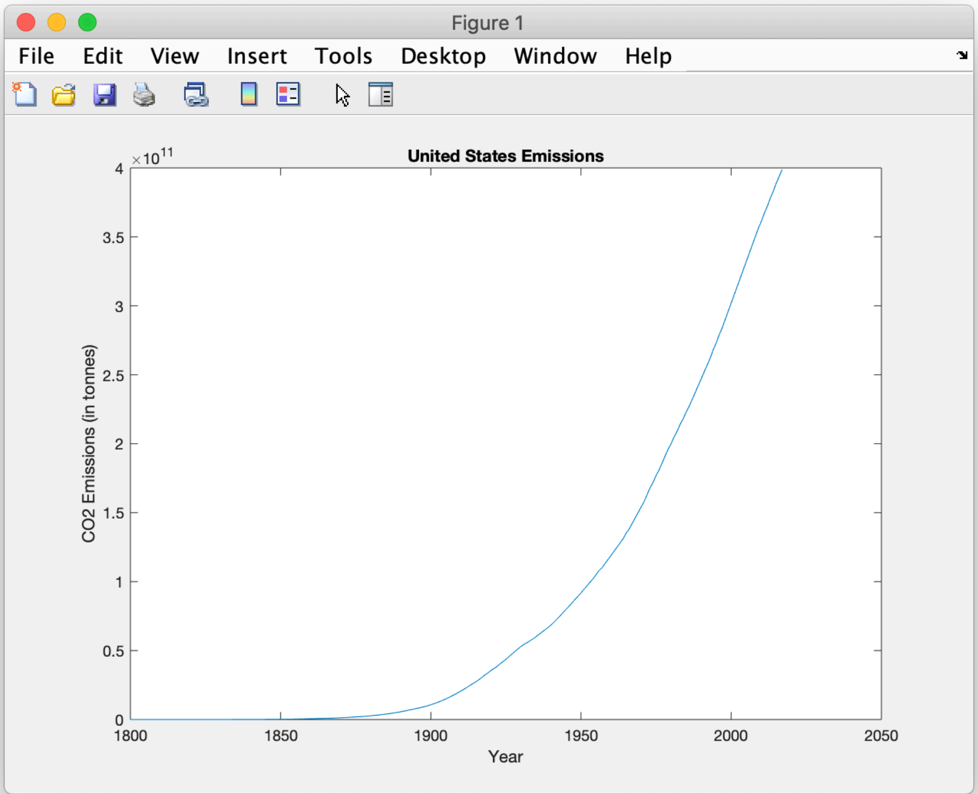
```
  Columns 241 through 252

    0.0000    0.0000    0.0000    0.0000
    2.5200    2.5700    2.6200    2.6800

  Columns 253 through 264

    0.0000    0.0000    0.0000    0.0000
    3.2000    3.2600    3.3200    3.3800
```

Note: Command window outputs the numbers in scientific notations, but when the data is analyzed the correct values are applied.

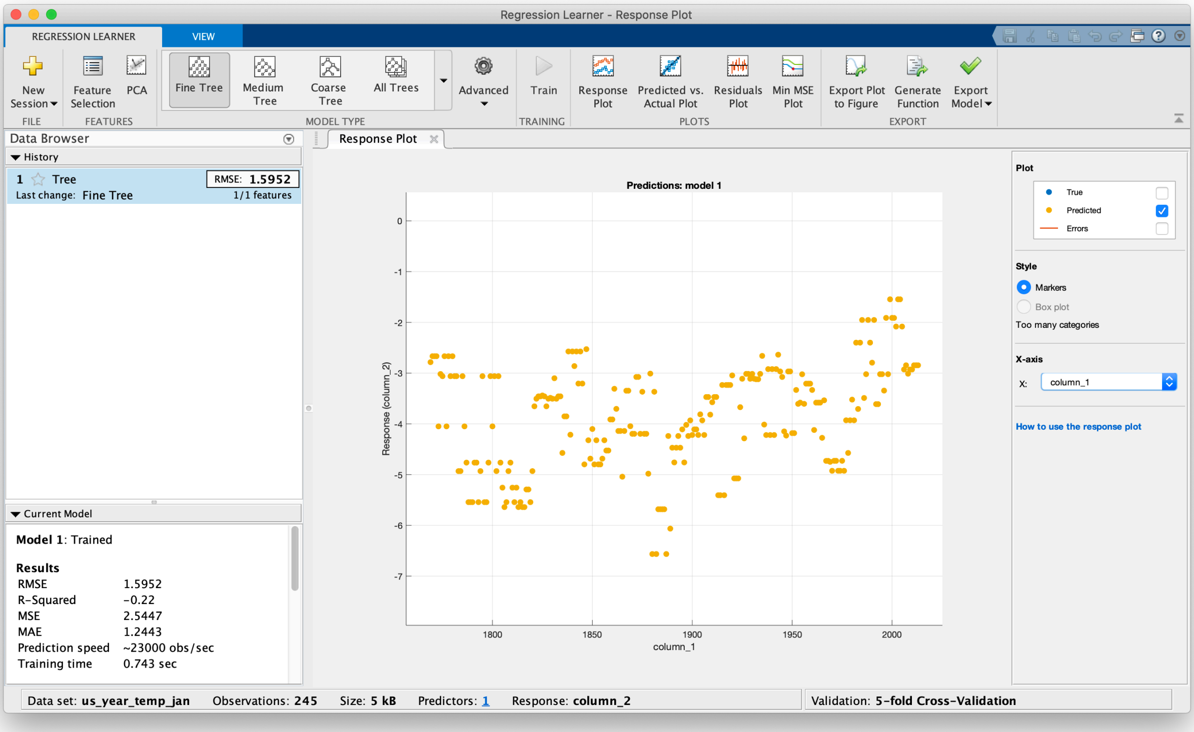
The plotted temperature change over the years:



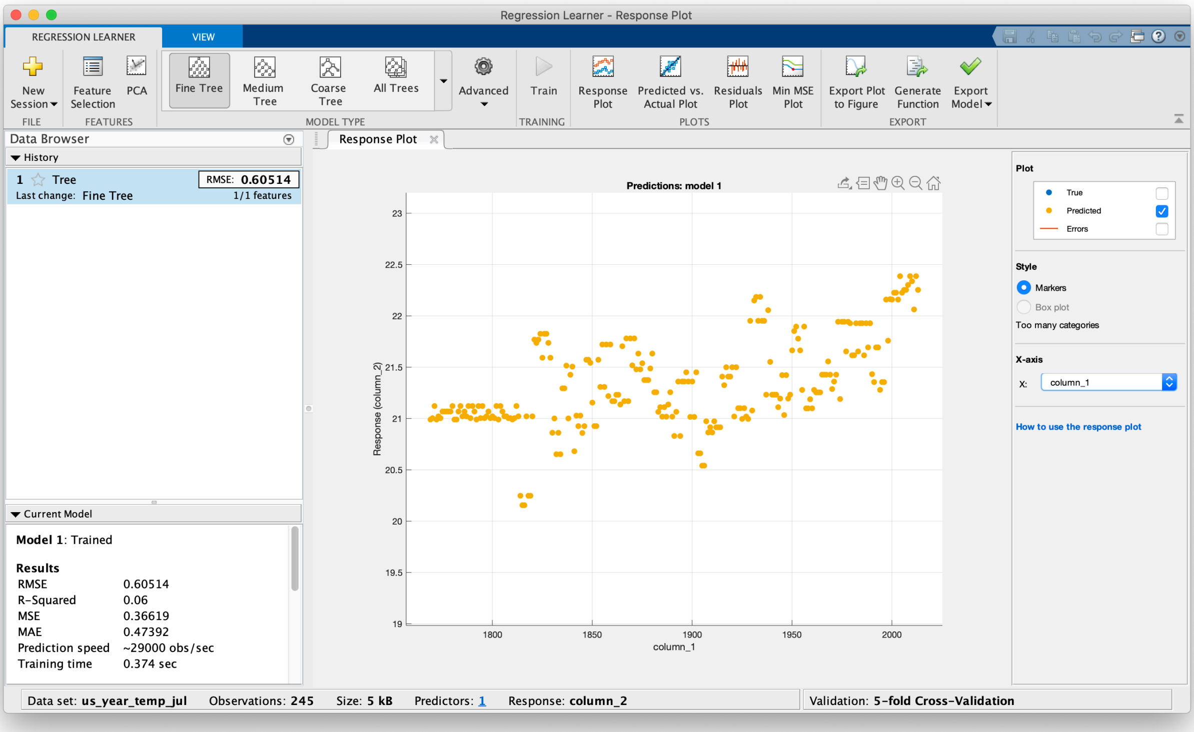
The plotted emissions change over the years:



The regression learner for temperature change in January:



The regression learner for temperature change in July:



Conclusion

According to data, the emissions for every countries were only increasing over the years. The regression learner shows that there's correlation between the year and the temperature change: every year the temperature is increasing. In all of the countries the average temperate of both January and July during the 19th century was lower than the temperature during the mid 20th and beginning of 21st centuries. It happened due to a huge increase in the emissions, for the reason that during that times the industrialization was happening. Even the 2 degrees temperature change can affect the environment. Obviously, it is not possible to decrease emissions to 0, so our solution is to try to find the optimal emissions level which didn't cause a significant temperature change. From the data it looks like the period of the time when the emissions were quite existing but the temperature change wasn't happening is from 1850s to 1950s. After 1950s the significant temperature change stated, therefore 1950s was the threshold for the temperature change and the emissions during that year are optimal.

For **China** it is: 1871475936 tons

For **Germany** it is: 29189721328 tons

For **India** it is: 2194578448 tons

For **Russia** it is: 854787568 tons

For **United Kingdom** it is: 39149444288 tons

For **United States** it is: 91806763184 tons

Of course, in order to stop the climate change all countries should go to the level of 1950s emissions, not only these six countries. However, this project could be the start of the emissions level reduction for the these six countries. Every country should decrease their emissions because we only have **one** planet.