Climate Change - Greenhouse Gases Effects

Project submission for edX *Data Science: Capstone* course

Justyna Piątyszek

2022-04-20

Contents

Introduction

This is the final project submission for HarvardX Professional Certificate in Data Science program.

The aim of this project is to analyze the influence of several greenhouse gases on the changes in average global temperature and to build a model that accurately predicts the temperature changes based on the observed gas concentrations.

We will use a data set based on data coming from NOAA Earth System Research Laboratory (ESRL), SOLARIS-HEPPA, NASA GISS and the Climatic Research Unit of the University of East Anglia.

The data set can be found on Kaggle under the following link:
https://www.kaggle.com/datasets/econdata/climate-change

We will start the analysis with Exploratory Data Analysis (EDA). Based on noticed data properties, we will try out several models including Linear Regression, Random Forest, K-Nearest Neighbors and Ensemble model. Before training the models and making predictions, we will partition the data set into train and test sets. To evaluate our models, we will compare R-squared, Root Mean Squared Error and Mean Absolute Error.

We start by loading the needed libraries (please note that this process could take a couple of minutes):

```
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
if(!require(corrplot)) install.packages("corrplot", repos = "http://cran.us.r-project.org")
if(!require(data.table)) install.packages("data.table", repos = "http://cran.us.r-project.org")
if(!require(dlookr)) install.packages("dlookr", repos = "http://cran.us.r-project.org")
if(!require(dplyr)) install.packages("dplyr", repos = "http://cran.us.r-project.org")
if(!require(randomForest)) install.packages("randomForest", repos = "http://cran.us.r-project.org")
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")

library(caret)
library(corrplot)
library(data.table)
library(dlookr)
library(dplyr)
library(randomForest)
library(tidyverse)
```

Loading the .csv file into R:

```
url <- "https://raw.githubusercontent.com/justpiat/Climate_Change/bd216a156291c993fc4f7145eba9979b416d4b
dat <- read.csv(url)
names(dat)[7] <- 'CFC-11'
names(dat)[8] <- 'CFC-12'
```

Exploratory Data Analysis

We can see that our data set is a tidy data frame with 308 rows and 11 columns. Each line represents measured global values on a given month between May 1983 and December 2008.

```
head(dat)
```

Year	Month	MEI	CO2	CH4	N2O	CFC-11	CFC-12	TSI	Aerosols	Temp
1983	5	2.556	345.96	1638.59	303.677	191.324	350.113	1366.102	0.0863	0.109
1983	6	2.167	345.52	1633.71	303.746	192.057	351.848	1366.121	0.0794	0.118
1983	7	1.741	344.15	1633.22	303.795	192.818	353.725	1366.285	0.0731	0.137
1983	8	1.130	342.25	1631.35	303.839	193.602	355.633	1366.420	0.0673	0.176
1983	9	0.428	340.17	1648.40	303.901	194.392	357.465	1366.234	0.0619	0.149
1983	10	0.002	340.30	1663.79	303.970	195.171	359.174	1366.059	0.0569	0.093

The variables include:

- `Year` - the year of the observation
- `Month` - the month of the observation
- `MEI` - multivariate ENSO index, characterizing the intensity of El Niño Southern Oscillation (an irregular weather event in the Pacific Ocean that affects global temperatures)
- `CO2` - atmospheric concentrations of carbon dioxide
- `CH4` - atmospheric concentrations of methane
- `N2O` - atmospheric concentrations of nitrous oxide
- `CFC-11` - atmospheric concentrations of trichlorofluoromethane (CCI3F), commonly referred to as CFC-11 or Freon-11

- `CFC-12` - atmospheric concentrations of dichlorodifluoromethane (CCI2F2), commonly referred to as CFC-12 or Freon-12
- `TSI` - Total Solar Irradiance in W/m2 (the solar power over all wavelengths per unit area)
- `Aerosols` - mean stratospheric aerosol optical depth at 550 nm (indication of how much direct sunlight is prevented from reaching the ground by various particles, e.g. from a volcanic eruption)
- `Temp` - the difference in degrees Celsius between the average global temperature in the given month and a reference value

CO2, CH4 and N20 concentrations are expressed in ppmv (parts per million by volume), whereas CFC-11 and CFC-12 in ppbv (parts per billion by volume).

Using dlookr package, we can see that all variables are numeric and that the data set has mostly unique values (except for `Year` and `Month`). We can see that there are no missing values for any of the variables:

```
diagnose(dat)
```

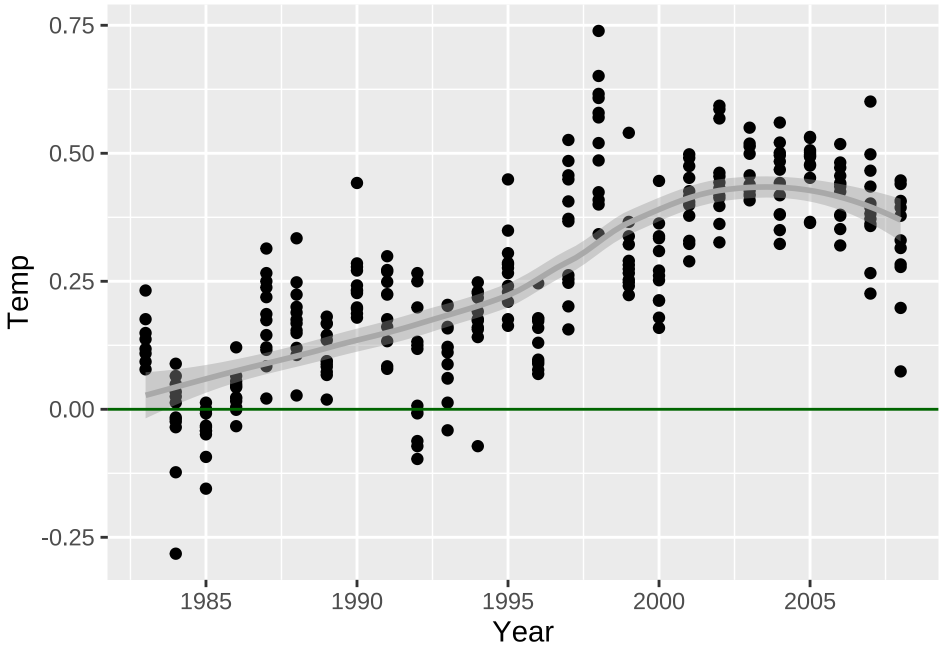
variables	types	missing_count	missing_percent	unique_count	unique_rate
Year	integer	0	0	26	0.0844156
Month	integer	0	0	12	0.0389610
MEI	numeric	0	0	294	0.9545455
CO2	numeric	0	0	298	0.9675325
CH4	numeric	0	0	303	0.9837662
N2O	numeric	0	0	304	0.9870130
CFC-11	numeric	0	0	307	0.9967532
CFC-12	numeric	0	0	307	0.9967532
TSI	numeric	0	0	302	0.9805195
Aerosols	numeric	0	0	155	0.5032468
Temp	numeric	0	0	242	0.7857143

The following tibble gives us basic statistics of the variables:

```
describe(dat, MEI:Temp) %>% select(variable, mean:kurtosis)
```

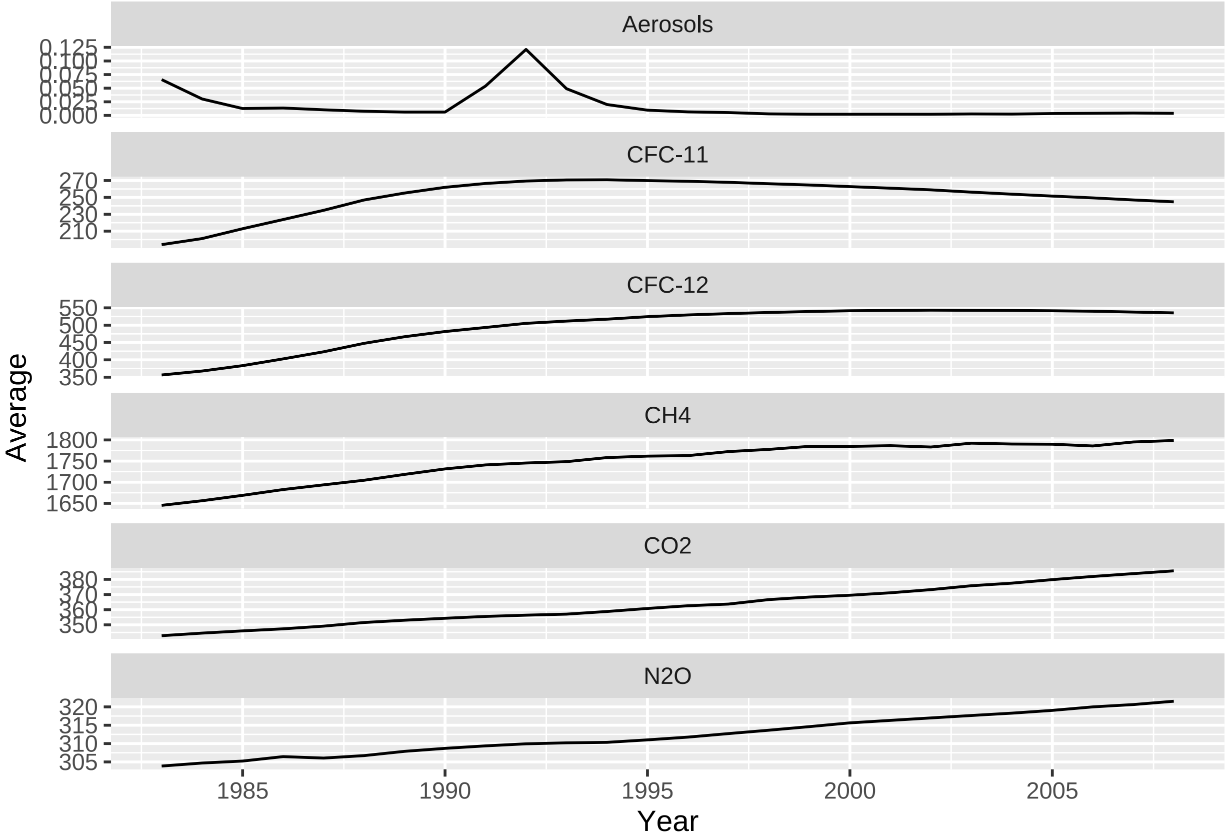
variable	mean	sd	se_mean	IQR	skewness	kurtosis
MEI	0.2755552	0.9379185	0.0534429	1.22925	0.5393020	0.1608255
CO2	363.2267532	12.6471249	0.7206368	20.43500	0.1786759	-1.0668743
CH4	1749.8245130	46.0516782	2.6240379	64.70250	-0.8266609	-0.3434096
N2O	312.3918344	5.2251307	0.2977295	8.86750	0.1453155	-1.1951099
CFC-11	251.9730682	20.2317832	1.1528128	20.73550	-1.4637386	1.3377354
CFC-12	497.5247825	57.8268988	3.2949934	68.11350	-1.2277239	0.1328412
TSI	1366.0707591	0.3996095	0.0227699	0.64620	0.7160308	-0.3981607
Aerosols	0.0166571	0.0290496	0.0016553	0.00980	2.9805525	8.6921707
Temp	0.2567760	0.1790898	0.0102046	0.28550	-0.0265399	-0.6304036

The following graph presents the temperature change over time:

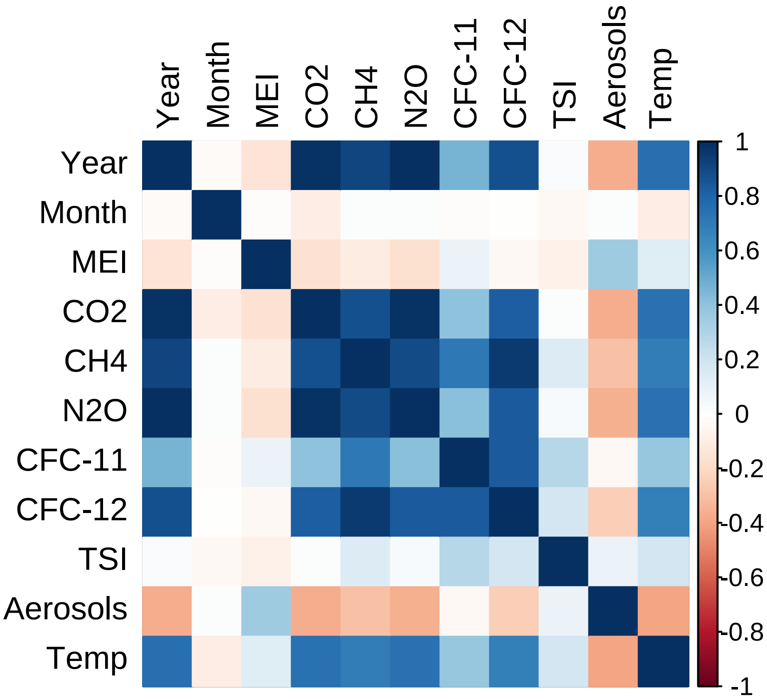


The last negative value of the global temperature change was observed in 1994, with the highest increases from 1997 onward. The temperature values show a clear positive trend for most of the analyzed period with a slight downward trend starting in 2005.

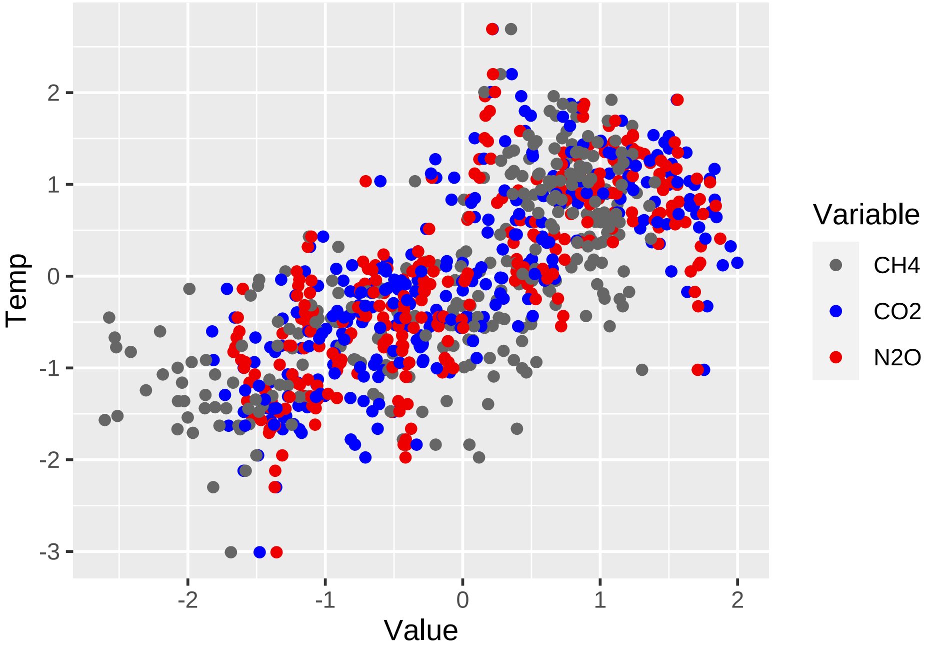
From the graphs below, we can see that the yearly average concentrations of CH4, CO2 and NO3 follow a very similar pattern, with visibly increased emissions at the beginning of the 21st century. CFC-11 and CFC-12 concentrations seem to notice a slight decline at the end of the analyzed period. Aerosols levels seem to be mostly constant with a concentration spike in the 1990s.



The correlation matrix shows that Year, CH4, CO2, NO3 and CFC-12 have a high positive correlation with one another and with the temperature change. The `Aerosols` variable shows a negative correlation to other gases concentrations and temperature. `TSI` and `Month` show no significant correlation with any of the variables and `MEI` has a slight positive correlation with Aerosols levels.



CH4, CO2 and N2O concentrations may prove to be the strongest predictors of the change in temperature. We can see that the standardized values of those gases align with the z-scores for temperature values:



Methodology - Prediction models

To train our models, we will divide the data set into train and test sets:

```
set.seed(1, sample.kind = "Rounding")
test_index <- createDataPartition(y = dat$Temp, times = 1, p = 0.2, list = FALSE)
train_set <- dat[-test_index,]
test_set <- dat[test_index,]

train_x <- train_set[1:10]
train_y <- train_set$Temp

test_x <- test_set[1:10]
test_y <- test_set$Temp
```

To evaluate our models, we will create functions calculating R-squared (`R2`), Root Mean Squared Error (`RMSE`) and Mean Absolute Error (`MAE`):

```
R2 <- function(test_y, y_hat){
  sqrt(cor(test_y, y_hat))
}

RMSE <- function(test_y, y_hat){
  sqrt(mean((y_hat - test_y)^2))
}

MAE <- function(test_y, y_hat){
  mean(abs(test_y - y_hat))
}
```

Linear Regression

First, we will make predictions using the variables that seem to have the biggest influence on the temperature according to the correlation matrix, i.e. Year, CH4, CO2, N2O, CFC-12 and Aerosols:

```
train_x <- train_set[c(1,4:6,8,10)]
test_x <- test_set[c(1,4:6,8,10)]

set.seed(1, sample.kind = "Rounding")
fit_lm1 <- train(train_x, train_y, method = "lm")
y_hat_lm1 <- predict(fit_lm1, newdata = test_x)
R2_lm1 <- R2(test_y, y_hat_lm1)
RMSE_lm1 <- RMSE(test_y, y_hat_lm1)
MAE_lm1 <- MAE(test_y, y_hat_lm1)
print(R2_lm1)
```

```
## [1] 0.9082323
```

```
print(RMSE_lm1)
```

```
## [1] 0.09205927
```

```
print(MAE_lm1)
```

```
## [1] 0.07175545
```

In the second model, we will train our linear regression model using all variables to predict temperature change:

```
train_x <- train_set[1:10]
test_x <- test_set[1:10]

set.seed(1, sample.kind = "Rounding")
fit_lm2 <- train(train_x, train_y, method = "lm")
y_hat_lm2 <- predict(fit_lm2, newdata = test_x)
R2_lm2 <- R2(test_y, y_hat_lm2)
RMSE_lm2 <- RMSE(test_y, y_hat_lm2)
MAE_lm2 <- MAE(test_y, y_hat_lm2)
print(R2_lm2)
```

```
## [1] 0.93946
```

```
print(RMSE_lm2)
```

```
## [1] 0.0825516
```

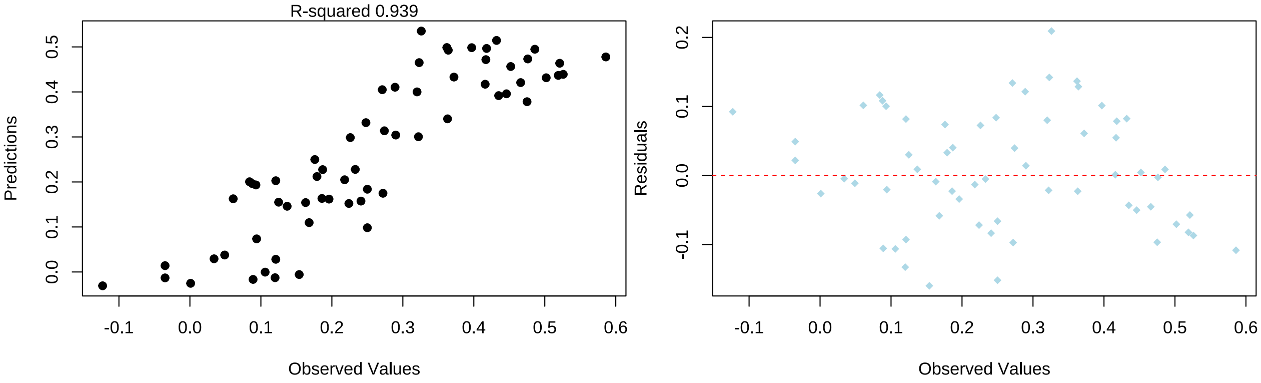
```
print(MAE_lm2)
```

```
## [1] 0.06832185
```

We can see that the linear model with all 10 predictors gives us slightly better predictions:

```
## integer(0)
```

```
## integer(0)
```

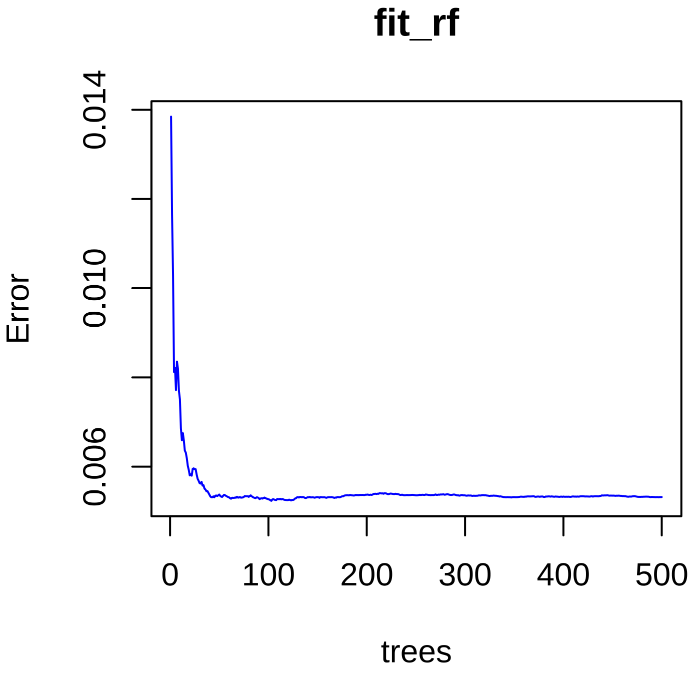


Random Forest

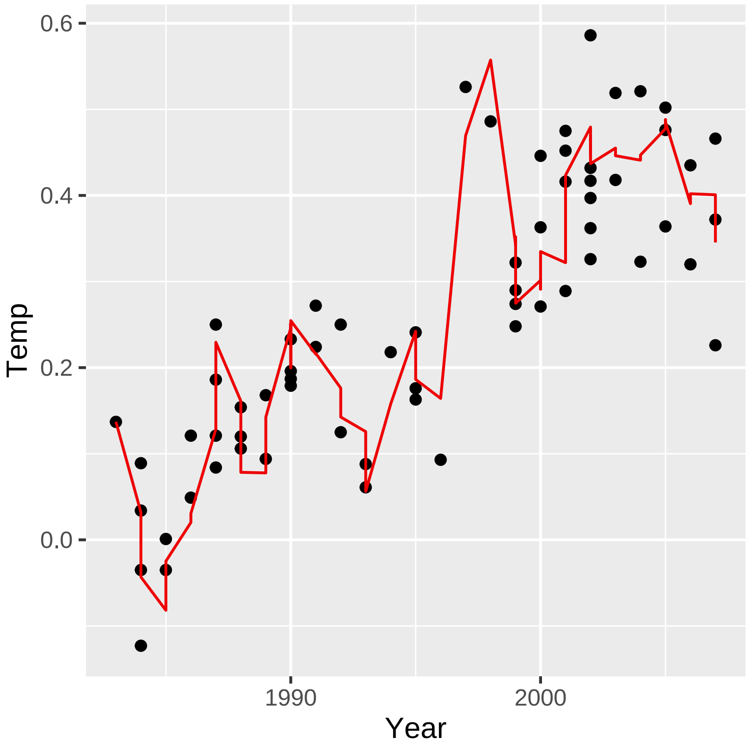
We will use all 10 predictors and the `randomForest` function in the randomForest package for our next model:

```
set.seed(1, sample.kind = "Rounding")
fit_rf <- randomForest(Temp~., data = train_set)
```

We can see that the model's accuracy stabilizes at around 50 trees:



The red line shows us the resulting estimate for this random forest:



And we can see it gives us lower errors than the linear models:

```
## [1] 0.9640556
```

```
## [1] 0.06122336
```

```
## [1] 0.04991073
```

The most important variables in this model for predicting temperature change are CFC-12, N2O and Year:

```
varImp(fit_rf) %>% arrange(desc(.))
```

	Overall
CFC12	1.8758941
N2O	1.4070008
Year	1.1217458
Aerosols	1.0630168
CO2	0.6514738
CFC11	0.5575853
MEI	0.4887184
CH4	0.4005024
TSI	0.2262212
Month	0.1706905

K-Nearest Neighbors

For the kNN algorithm, we will check a sequence of k from 1 to 10, since we do not have many data points:

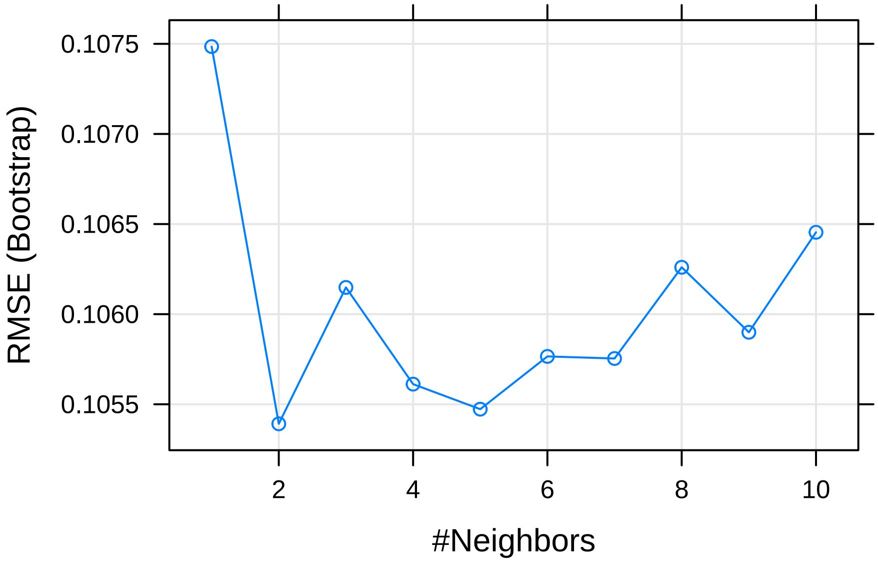
```
set.seed(1, sample.kind = "Rounding")
fit_knn <- train(Temp ~ ., method = "knn",
                   data = train_set,
                 tuneGrid = data.frame(k = seq(1:10)))
```

We get the best estimates with the parameter k=2:

```
fit_knn$bestTune
```

	k
2	2

```
plot(fit_knn)
```



```
y_hat_knn <- predict(fit_knn, test_set)
```

We can see that we get better results than from the linear models, but slightly worse than with the RF algorithm:

```
R2_knn <- R2(test_y, y_hat_knn)
RMSE_knn <- RMSE(test_y, y_hat_knn)
MAE_knn <- MAE(test_y, y_hat_knn)
print(R2_knn)
```

```
## [1] 0.9368189
```

```
print(RMSE_knn)
```

```
## [1] 0.08490523
```

```
print(MAE_knn)
```

```
## [1] 0.06451563
```

Ensemble

For our final model, we will check if we can improve the final results by combining the results of the two previous algorithms. We will combine the random forest and knn models and create new predictions by taking the average of the two models:

```
y_hat_ensemble <- (y_hat_rf + y_hat_knn)/2
R2_en <- R2(test_y, y_hat_ensemble)
RMSE_en <- RMSE(test_y, y_hat_ensemble)
MAE_en <- MAE(test_y, y_hat_ensemble)
print(R2_en)
```

```
## [1] 0.9584728
```

```
print(RMSE_en)
```

```
## [1] 0.06648744
```

```
print(MAE_en)
```

```
## [1] 0.05425139
```

Results

We will store all results in the following data frame:

model	R2	RMSE	MAE
Linear I	0.9082323	0.0920593	0.0717554
Linear II	0.9394600	0.0825516	0.0683219
Random Forest	0.9640556	0.0612234	0.0499107
K-Nearest Neighbors	0.9368189	0.0849052	0.0645156
Ensemble	0.9584728	0.0664874	0.0542514

By trying out different models, we improved the R-squared from **0.908** to **0.964**, the RMSE from **0.0921** to **0.0612** and the Mean Absolute Error from **0.0718** to **0.0499**. We got the best predictions from the **Random Forest algorithm**. We can also see that combining predictions from the knn and the random forest models greatly improved the results compared to the knn model alone.

Conclusion

Despite relatively few data points, the trained models turned out to provide satisfactory predictions of global temperature change. A similar analysis could be conducted with an updated data set including new observations after the year 2008 until now. To get a more detailed picture, we could also use emission values per country/region.
The caret package provides many more algorithms which could be used in future work to improve predictions. The models can be useful to predict the scale of global warming and to single out the factors that influence global temperature the most. This can provide insight into which gas concentrations should be reduced to most effectively slow down the rise in global temperature.