CCE_Lab1

Ryan Peckner

June 30,2021

SECTION 1 We are interested in computing the global average temperature increase above pre-industrial times in °C at a given time t. We'll start by loading the data into a dataframe called 'forcings', where we will also deposit all the results of our calculations.

```
library(readxl,quietly = T)
suppressMessages({forcings_excel = read_xlsx("CCE - Assignment 1 - Forcings.xlsx",skip = 1)})
forcings = data.frame(year = as.integer(forcings_excel[1,][-1]),co2 = as.numeric(forcings_excel[4,][-1]
                        rfOther = as.numeric(forcings_excel[5,][-1]),emissions = as.numeric(forcings_exc
head(forcings)
```

```
##   year      co2   rfOther emissions
## 1 2010 380.3290 0.8300000  7626.003
## 2 2011 381.3423 0.8241111  7709.800
## 3 2012 383.3699 0.8182222  7794.723
## 4 2013 385.4130 0.8123333  7883.785
## 5 2014 387.4731 0.8064444  7976.030
## 6 2015 389.5514 0.8005556  8071.568
```

Next, we'll fix the values of various parameters described in the handout. We'll then write functions to compute the essential variables in the model as functions of the timepoint t.

```
C_pre = 275 # Pre-industrial atmospheric CO2 level
lambda = 0.8 # Climate sensitivity
mu = 1/66 # Warming delay parameter
T_2010 = 0.8

RadiativeForcingCO2 = function(t,CO2) {
  return(5.35*log(CO2[t]/C_pre))}  # Note that R uses the natural logarithm by default

RadiativeForcingTotal = function(t,CO2) {
  RadiativeForcingCO2(t,CO2) + forcings$rfOther[t]
  }

GlobalAvgTempIncrease_ConstantRF = function(t,CO2) {
  lambda*RadiativeForcingTotal(t,CO2)
  }
```

Now, the global average temperature increase above pre-industrial times is computed via a recursive delay model. So we will use a recursive R function to compute it. Since recursion has an exponential running time, we will use memoisation to store the results of each function call.

```
library(memoise,quietly = T)
GlobalAvgTempIncrease = memoise(function(t,CO2) {
  if (t == 1) return(T_2010)
  else        return(GlobalAvgTempIncrease(t-1,CO2) +
```

```
                        mu*(GlobalAvgTempIncrease_ConstantRF(t,CO2) - GlobalAvgTempIncrease(t-1,CO2)))

})
```

Now we compute the global average temperature increase at each time point, and add these values to our dataframe containing the input variables.

```
forcings$TempIncrease = sapply(1:nrow(forcings),GlobalAvgTempIncrease,
                               CO2 = forcings$co2)
forcings$TempIncrease_ConstantCO2 = sapply(1:nrow(forcings),GlobalAvgTempIncrease,CO2 = rep(forcings$co
                                                           nrow(forcings)))
head(forcings)
```

```
##   year      co2   rfOther emissions TempIncrease TempIncrease_ConstantCO2
## 1 2010 380.3290 0.8300000  7626.003    0.8000000                0.8000000
## 2 2011 381.3423 0.8241111  7709.800    0.8190687                0.8188961
## 3 2012 383.3699 0.8182222  7794.723    0.8381210                0.8374346
## 4 2013 385.4130 0.8123333  7883.785    0.8571579                0.8556208
## 5 2014 387.4731 0.8064444  7976.030    0.8761807                0.8734601
## 6 2015 389.5514 0.8005556  8071.568    0.8951907                0.8909577
```

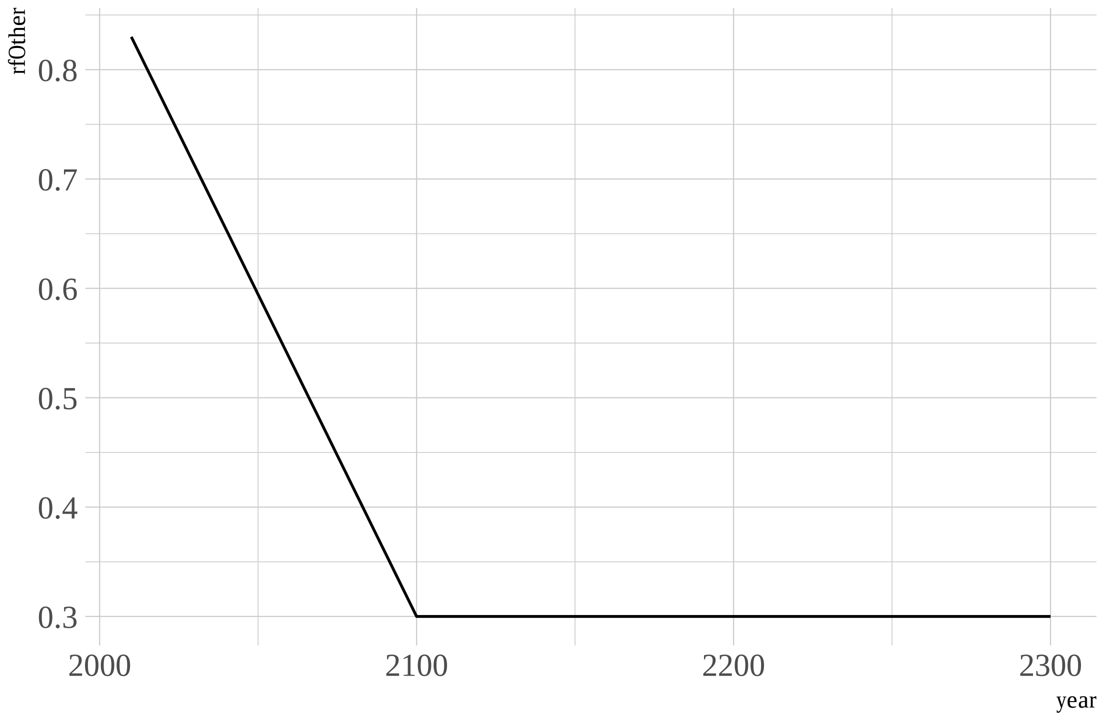
Question 1a *What happens to projected temperature if CO2 concentrations were held constant at 2010 levels in the model?*

In this case, the radiative forcing due to CO2 does not change with time, so any change in the total radiative forcing is due only to non-CO2 greenhouse gases. This non-CO2 radiative forcing decreases sharply with time and eventually stabilizes to a value of 0.3, beginning in year 2100:

```
#install.packages("extrafontdb",repos=getOption("repos"))
#devtools::install_github("hrbrmstr/hrbrthemes", force = TRUE)
library(ggplot2,quietly = T)
hrbrthemes::import_roboto_condensed()

library(hrbrthemes, quietly = T)
d <- read.csv(extrafont:::fonttable_file(), stringsAsFactors = FALSE)
d[grepl("Light", d$FontName),]$FamilyName <- font_rc_light
write.csv(d,extrafont:::fonttable_file(), row.names = FALSE)
extrafont::loadfonts()
```

```
## Registering font with R using pdfFonts(): Roboto Condensed

## Registering font with R using pdfFonts(): Roboto Condensed Light
```

```
library(gcookbook,quietly = T)

ggplot(forcings,aes(x=year,y=rfOther)) + geom_line() + theme_ipsum_rc()
```



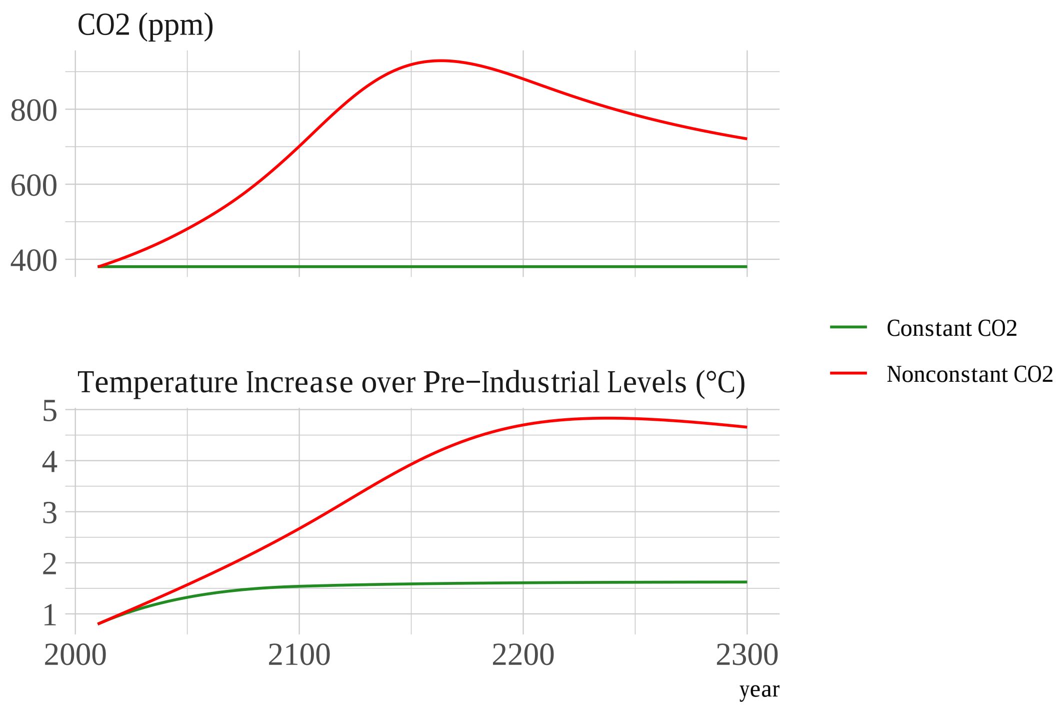
Consequently, the projected rate of temperature increase will eventually decline. Indeed, we saw above that the global average temperature increase in the case of constant radiative forcing is given by $T_t^e = \lambda \times rf_t$. Therefore we have

$$\begin{aligned} T_t &= T_{t-1} + \mu(T_t^e - T_{t-1}) && (1)\\ &= T_{t-1} + \mu(\lambda \times rf_t + T_{t-1}) && (2)\\ &= T_{t-1}(1-\mu) + \mu \times \lambda \times rf_t. && (3) \end{aligned}$$

Since $(1-\mu) < 1$ while rf_t becomes constant after 2100, we see that $T_t < T_{t-1}$ for $t > 90$.

Question 1b *Create two graphs, one for CO2 concentrations and one for temperature. The x-axis should have years on it for both graphs. Then plot each of the two cases analyzed (base case and constant concentrations) as one line.*

```
library(reshape2,quietly = T)
library(gridExtra,quietly = T)

forcings$ConstantCO2 = forcings$co2[1]
df_plot = melt(forcings[c("year","co2","ConstantCO2","TempIncrease","TempIncrease_ConstantCO2")],id.vars
df_plot$type = ifelse(grepl("Temp",df_plot$variable),"Temperature Increase over Pre-Industrial Levels (
df_plot$group = ifelse(grepl("Constant",df_plot$variable),"Constant CO2","Nonconstant CO2")

g_temp = ggplot(df_plot,aes(x=year,y=value)) +
              facet_wrap(~type,scales="free_y",ncol=1) +
              geom_line(aes(color=group)) +
            ylab("") +
            scale_color_manual(values=c("forestgreen","red"),
                               labels = c("Constant CO2","Nonconstant CO2")) +
            theme_ipsum_rc() +
            theme(legend.title = element_blank())

g_temp
```



SECTION 2 First, we will initialize the parameters given in the modeling section, then create variables representing each of the five boxes.

```
alpha = c(1,exp(-1/363),exp(-1/74),exp(-1/17),exp(-1/2))
gamma = c(0.13,0.2,0.32,0.25,0.1)
Box2010 = c(301.099,30.098,34.878,12.357,0.897)
beta = 0.00047

Box = memoise(function(i,t) {
  if (t == 1) return(Box2010[i])
  else return(alpha[i]*Box(i,t-1) + gamma[i]*beta*forcings$emissions[t])
})


AtmosphericCO2 = function(t) {
  return(Box(1,t) + Box(2,t) + Box(3,t) + Box(4,t) + Box(5,t))
}
```

Now, we will add a column to our dataframe containing the outputs from this carbon cycle model.

```
forcings$CarbonCycleModelCO2 = sapply(1:nrow(forcings),AtmosphericCO2)
head(forcings)
```

```
##   year      co2   rfOther emissions TempIncrease TempIncrease_ConstantCO2
## 1 2010 380.3290 0.8300000  7626.003    0.8000000                0.8000000
## 2 2011 381.3423 0.8241111  7709.800    0.8190687                0.8188961
## 3 2012 383.3699 0.8182222  7794.723    0.8381210                0.8374346
## 4 2013 385.4130 0.8123333  7883.785    0.8571579                0.8556208
## 5 2014 387.4731 0.8064444  7976.030    0.8761807                0.8734601
## 6 2015 389.5514 0.8005556  8071.568    0.8951907                0.8909577
##   ConstantCO2 CarbonCycleModelCO2
## 1     380.329            379.3290
```

```
## 2     380.329             381.3428
## 3     380.329             383.3703
## 4     380.329             385.4134
## 5     380.329             387.4735
## 6     380.329             389.5518
```

Question 1c *What happens to projected temperature if CO2 emissions were held constant at 2010 levels in the model?*

If CO2 emissions are held constant,

Question 1d *By how much % would we need to reduce emissions in each year to keep global warming below 2° over the next 300 years?*

For this, we will rewrite the model to accept arbitrary emission values instead of those supplied in the Excel sheet.

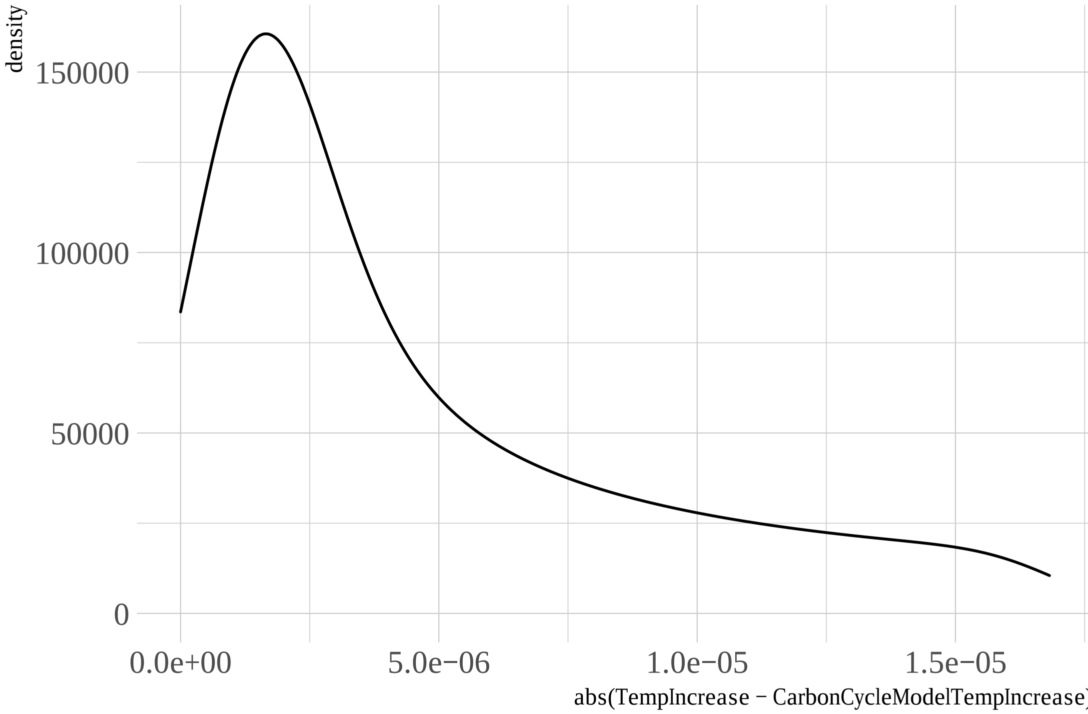
```
Box = memoise(function(i,t,emissions) {
  if (t == 1) return(Box2010[i])
  else return(alpha[i]*Box(i,t-1,emissions) + gamma[i]*beta*emissions[t])
})


AtmosphericCO2 = function(t,emissions) {
  return(Box(1,t,emissions) + Box(2,t,emissions) + Box(3,t,emissions) + Box(4,t,emissions) + Box(5,t,emi
}
```

We want to ensure that global warming doesn't exceed 2°C over the next 300 years. We'll calculate the emissions threshold that would need to be maintained to prevent this increase. For this, supply the output of the carbon cycle model as the input to the climate dynamics model from Section 1.

```
CarbonCycleModelCO2 = function(emissions) {
  sapply(1:nrow(forcings),AtmosphericCO2,emissions=emissions)
}

GlobalAvgTempIncrease_FromEmissions = memoise(function(t,emissions) {
  GlobalAvgTempIncrease(t,CarbonCycleModelCO2(emissions))
})
```

As a sanity check, let's make sure that the average temperature increase calculated in this fashion agrees with the results from the model when we input the original emissions. Since it isn't reasonable to expect exact agreement given the approximations involved in each approach, let's instead look at the distribution of the magnitudes of the differences between the results for each timepoint.

```
forcings$CarbonCycleModelCO2 = CarbonCycleModelCO2(forcings$emissions)
forcings$CarbonCycleModelTempIncrease = sapply(1:nrow(forcings),
                                              GlobalAvgTempIncrease_FromEmissions,
                                              emissions=forcings$emissions)
ggplot(forcings,aes(abs(TempIncrease - CarbonCycleModelTempIncrease))) + geom_density() + theme_ipsum_r
```



We see that the differences are extremely small relative to temperature, so the carbon cycle model does seem to accurately predict the temperature effects of emissions.

Let's introduce some functions to calculate the relevant parameters of the model - in particular, the effect on temperature increase of a given percent reduction in CO2 emissions.

```
ReducedEmissions = memoise(function(t,percent_reduction) {
  if (t == 1) return(forcings$emissions[1])
  else return(ReducedEmissions(t-1,percent_reduction)*(1-percent_reduction))
})

ReducedEmissionsProfile = function(percent_reduction) {
  sapply(1:nrow(forcings),ReducedEmissions,percent_reduction=percent_reduction)
}

TempIncrease_ReducedEmissions = function(t,percent_reduction) {
  return(GlobalAvgTempIncrease_FromEmissions(t,ReducedEmissionsProfile(percent_reduction)))
}

MaxTempIncrease = function(percent_reduction) {
  return(max(sapply(1:nrow(forcings),TempIncrease_ReducedEmissions,percent_reduction)))
}
```

Now, we wish to calculate the minimum annual percent reduction in emissions that will keep the temperature increase below 2°C for all timepoints. We'll compute this with a simple root-finding algorithm.

```
RequiredPercentReduction = uniroot(function(p) MaxTempIncrease(p) - 2,interval=c(0,1))
print(RequiredPercentReduction$root)
```

```
## [1] 0.01318409
```

We see that we would need an annual emissions reduction of around **1.3%** (beginning in 2011) to maintain a temperature increase below 2°C until 2300. Let's add the reduced emissions profile and associated atmospheric carbon concentrations and temperature increases to our master dataframe for future reference.

```r
forcings$ReducedEmissions2010 = ReducedEmissionsProfile(RequiredPercentReduction$root)
forcings$AtmosphericCO2_ReducedEmissions2010 = CarbonCycleModelCO2(forcings$ReducedEmissions2010)
forcings$TempIncrease_ReducedEmissions2010 = sapply(1:nrow(forcings),
                                                     TempIncrease_ReducedEmissions,
                                                     percent_reduction=RequiredPercentReduction$root)
```

Question 1e *If emissions stay as they are specified in base case until the year 2049, and are then reduced by a fixed percent each year, how much would they have to be reduced in percent in each year to keep global warming below 2° over the next 300 years?*

We will recycle the same code from above, but supply the model with the base case emissions levels until 2049.

```r
ReducedEmissions = memoise(function(t,percent_reduction) {
  if (t <= 39) return(forcings$emissions[t])
  else return(ReducedEmissions(t-1,percent_reduction)*(1-percent_reduction))
})

ReducedEmissionsProfile = function(percent_reduction) {
  sapply(1:nrow(forcings),ReducedEmissions,percent_reduction=percent_reduction)
}

TempIncrease_ReducedEmissions = function(t,percent_reduction) {
  return(GlobalAvgTempIncrease_FromEmissions(t,ReducedEmissionsProfile(percent_reduction)))
}

MaxTempIncrease = function(percent_reduction) {
  return(max(sapply(1:nrow(forcings),TempIncrease_ReducedEmissions,percent_reduction)))
}
```

Again, we use root-finding to identify the desired minimal percent reduction.

```r
RequiredPercentReduction = uniroot(function(p) MaxTempIncrease(p) - 2,interval=c(0,1))
print(RequiredPercentReduction$root)
```

```
## [1] 0.2307607
```

We see that we would need an annual emissions reduction of around **23%** to maintain a temperature increase below 2°C until 2300 if the base case emissions are released until 2049. Again, we add this information to our dataframe for reference.

```r
forcings$ReducedEmissions2050 = ReducedEmissionsProfile(RequiredPercentReduction$root)
forcings$AtmosphericCO2_ReducedEmissions2050 = CarbonCycleModelCO2(forcings$ReducedEmissions2050)
forcings$TempIncrease_ReducedEmissions2050 = sapply(1:nrow(forcings),
                                                     TempIncrease_ReducedEmissions,
                                                     percent_reduction=RequiredPercentReduction$root)
```

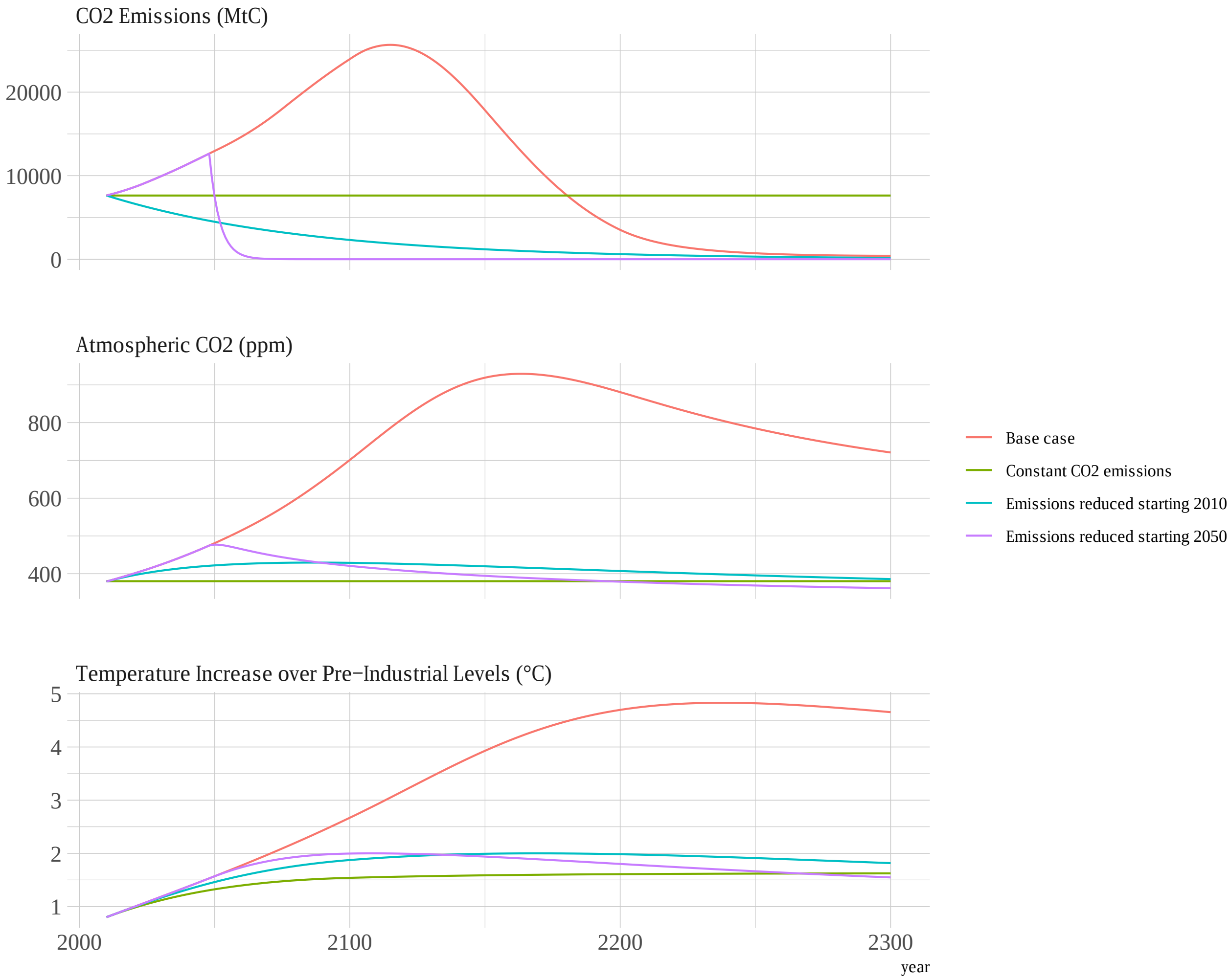
Question 1f *Create three graphs, one for CO2 emissions, one for CO2 concentrations and one for temperature. The x-axis should have years on it for all three graphs. Then plot each of the four cases analyzed (base case, constant emissions, reduction in emissions starting now, reduction in emissions starting in 2050) as one line.*

```r
library(dplyr,quietly = TRUE)
```

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:gridExtra':
##
```

```
##     combine

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
forcings$ConstantEmissions = forcings$emissions[1]
forcings = rename(forcings,BaseCaseEmissions = emissions,BaseCaseAtmosphericCO2 = co2, BaseCaseTempIncre
df_plot = melt(forcings[c("year","BaseCaseEmissions","ConstantEmissions","ReducedEmissions2010","Reduced
                          "BaseCaseAtmosphericCO2","ConstantCO2","AtmosphericCO2_ReducedEmissions2010",
                          "BaseCaseTempIncrease","TempIncrease_ConstantCO2",
                          "TempIncrease_ReducedEmissions2010","TempIncrease_ReducedEmissions2050")],
                        id.vars="year")

df_plot$type = "CO2 Emissions (MtC)"
df_plot$type[grep("CO2",df_plot$variable)] = "Atmospheric CO2 (ppm)"
df_plot$type[grep("Temp",df_plot$variable)] = "Temperature Increase over Pre-Industrial Levels (°C)"


df_plot$type = factor(df_plot$type,levels=c("CO2 Emissions (MtC)","Atmospheric CO2 (ppm)","Temperature I
 # Make sure emissions get plotted first

df_plot$reduction = "Base case"
df_plot$reduction[grep("Constant",df_plot$variable)] = "Constant CO2 emissions"
df_plot$reduction[grep("ReducedEmissions2010",df_plot$variable)] = "Emissions reduced starting 2010"
df_plot$reduction[grep("ReducedEmissions2050",df_plot$variable)] = "Emissions reduced starting 2050"

g_temp = ggplot(df_plot,aes(x=year,y=value)) +
              facet_wrap(~type,scales="free_y",ncol=1) +
              geom_line(aes(group=variable,color=reduction)) +
            ylab("") +
            theme_ipsum_rc() +
            theme(legend.title = element_blank())

g_temp
```


CO2 Emissions (MtC)
20000
10000
0
Atmospheric CO2 (ppm)
800
600
400
Temperature Increase over Pre−Industrial Levels (°C)
5
4
3
2
1
2000
2100
2200
2300
year
Base case
Constant CO2 emissions
Emissions reduced starting 2010
Emissions reduced starting 2050