Worksheet 2: THE *GROWTH RATE OF ATMOSPHERIC CO2*

The rate of increase of tropospheric carbon dioxide reflects the net exchange between the atmosphere and both the land and oceans, integrated over the globe. As such, it is an important measure for comparison with models and other observations.

To answer this question you will use data from the Mauna Loa Observatory in Hawaii. This high-altitude station has been measuring atmospheric carbon dioxide composition since Charles D. Keeling started them in 1958. We will assume that this data is an approximate measure of the average carbon dioxide concentration in the global free troposphere, which is a reasonable assumption at long enough time scales.

I downloaded the data from Scripps Institution of Oceanography and reformatted it to make it a little easier to read in with R. You can get my version of the CO_2 file from here:

http://waage.sr.unh.edu/~braswell/eos864/data/atm_co2.txt

Take a look. What kind of data file is this? Answer: "space delimited text". Would it be adequate for real research? Answer: no. You can look at the original file here:

http://scrippsco2.ucsd.edu/data/in_situ_co2/monthly_mlo.csv

That's better, we can see the units are *ppm* (parts per million by volume), and some additional useful information. There are a few other data columns there, but let's not bother with them at this point. Now read our CO_2 data into an R data frame. This is something you will do a lot.

```
> f = "http://waage.sr.unh.edu/~braswell/eos864/data/atm_co2.txt"
> dat = read.table(f, header=TRUE)
```

Check out the first few rows.

```
> dat[,1:5]
```

Oops, I do that all the time. What just happened?

```
> dat[1:5,]
```

Here are some other things I do almost automatically when I load a new data set.

```
> names(dat)
> dim(dat)
> summary(dat)
> sapply(dat, mode)
```

That last one was something new. The sapply() function is a way to apply a function to all the columns of a data frame. In this case we are asking what is the data type for each column. In this case, they are all numeric. What other things could we do with sapply()?

```
> sapply(dat, length)
```

But we knew this already. We'll do more complicated things with sapply() later. As an aside, let's remind ourselves what the mode and class are for data frames.

```
> mode(dat)
> class(dat)
```

Make sense? This data set really has just one physical variable, so we are limited in the kinds of things we can do, but that's alright because the question we are trying to answer really is a straightforward one we can answer with the one variable.

Remember that one way to access the columns of a data frame is by using its name and the "$" notation:

```
> dat$CO2[1:10]
> dat$Time[1:10]
```

So let's take a look at the data.

```
> plot(dat$Time, dat$CO2)
```

Like a lot of things in R, there are at least two other ways to do exactly the same thing:

```
> plot(dat$CO2 ~ dat$Time)
> plot(CO2 ~ Time, data=dat)
```

These other two use R's "formula notation" that will become more important when we do regressions and curve fitting.

Let's get back to the concentration graph. On my screen, the symbols run together and it's difficult to tell what's going on. Check out the first few years:

```
> plot(dat$Time[1:36], dat$CO2[1:36])
```

For a plot like this I like to use points and lines together:

```
> plot(dat$Time[1:36], dat$CO2[1:36], type='b')
```

The 'b' stands for 'both'. So, there's a strong seasonal cycle, which is not surprising (I know most of you are familiar with this data anyway and also understand the meaning of this pattern.) Let's look at the entire thing but using lines instead of points:

```
> plot(dat$Time, dat$CO2, type='l', main='Figure 1', xlab='Year',
+ ylab='CO2')
```

The question was, "what is the growth rate of atmospheric CO_2". We have atmospheric CO_2, but what does growth rate mean? Think about it for a minute before moving on.

Of course, growth rate is just the change over time. In R, we can easily take the first difference to calculate the monthly change in concentration in ppm/month, using the diff() function.

```
> growth = diff(dat$CO2)
> summary(growth)
```

So, the answer is that the mean growth rate is 12*0.116 = 1.39 ppm/year.

Are we done? In my opinion, no. There are far too many unanswered questions. What does the growth rate look like? Are there cycles? Is there a trend? What's happening at the annual (or other) time scales? Are there other patterns of interest? The point is, we have 50 years of data, it seems like there should be more to explore. Let's start by looking at the growth rate.

```
> plot(dat$Time, growth, type='l')
```

Error message. What happened?

```
> length(growth)
> length(dat$Time)
```

Right, since diff() returns a vector containing successive differences between the input values, it must have one fewer value than the input, and so we want a time vector that has the same length. Let's punt on this one for now, I just want to look at the growth curve.

```
> plot(growth, type='l')
```

The seasonal cycle is the dominant feature. It is important to note that the growth rate is often negative, even more often negative than positive. Is that true?

```
> hist(growth)
> hist(growth, 30)
> hist(growth, 30, prob=T)
> lines(density(growth), lty=2)
```

I pretty much always add a density plot to histograms using dens(), they work great together, but either one could be misleading on its own.

Remember we were looking at the positive and negative parts of this cycle, and yes, it seems like the negative part is often stronger, but less frequent (*i.e.*, the CO_2 concentrations sometimes decline more steeply in the summer than they rise in the winter). We probably could have inferred this from the original data, but it's interesting to clearly see two distinct modes.

Back to the monthly growth rate.

```
> plot(growth, type='l')
```

Now we more or less know what we are working with, so let's make a strategy to finish up this question. First, when there are possibly multiple time scales involved, we should explicitly check them out. Specifically, since the seasonal cycle seems to mask everything else, let's look at the annual signal. Also, let's rethink the units. I believe that for most applications we would like to know the *amount* of CO_2 going in and out of the atmosphere.

On the issue of annual time scale, there are two primary ways to do it: (1) smooth the monthly data, and (2) average the monthly data. We don't have a really good reason to choose one over the other, so let's do both.

First, let's do the easy one: annual averages, `i.e.`, produce one value of growth rate per year.

But should we take the difference of the annual average of CO_2, or the annual average of the monthly differences? Wait, let's not make ourselves crazy, we're not doing anything nonlinear, so they should be about the same. I pick the first, and will explain why later.

```
> annco2 = tapply(dat$CO2, list(dat$Year), mean)      # help(tapply)
> anngrow = diff(annco2)                              # growth rate ppm/year
> annyear = as.numeric(names(anngrow))                # year, one per year
> plot(annyear, anngrow, type='b')                    # units: ppm/year
```

Let's now do the version based on smoothing, in case we are missing important details at the monthly scale.

```
> co2.smoo = filter(dat$CO2, rep(1,12)/12)      # smoothed monthly co2
> co2.grow = 12*diff(co2.smoo)                  # monthly growth rate ppm/yr
```

The `filter()` function applies a moving "kernel" to the time series. In this case, the kernel is just a rectangular window (the fraction 1/12 repeated 12 times). Then, I multiplied the result by 12 in order get the annul units I want. Do you see how I could have done this without so many 12's?

Skip this next part if you believe what I said about not needing to worry about whether you do the averaging or differencing first.

```
> co2.grow2 = filter(diff(dat$CO2), rep(1,12))          # do it the other way
> plot(co2.grow, co2.grow2)                             # plot the two ways
> abline(0,1)                                           # 1:1 line
> year = dat$Time[2:600]                                # diff has one less item
> plot(year, co2.grow, type='l', col='red')             # on top of one another
> lines(year, co2.grow2, type='l', col='black')          # see any difference?
```

Back to the main graph:

```
> year = dat$Time[2:600]
> plot(year, co2.grow, type='l', ylab='CO2 (ppm/yr)', xlab='Year')
> lines(annyear, anngrow, lwd=3, col='blue')
> abline(0,0, lty=2)
```

I like to show the zero line when it is within of the graphing area. In this case, it's interesting that the monthly growth rate anomaly is sometimes negative (but maybe not important).

This is getting a little bit ahead of ourselves again, but I see there's a trend (again, we probably could have anticipated this based on the curvilinear increase in concentration), and I'd like to know how much the growth rate is increasing on average.

```
> regr = lm(anngrow ~ annyear)                          # fit a linear model
> lines(annyear, predict(regr), lty=3, lwd=2, col='red')
```

We'll talk about regressions a lot in the coming weeks. For now let's just get the slope.

```
> regr          # the slope is about 0.026 ppm/year/year
> 0.026*49              # check the answer - there are 49 years in the record
```

So that's about 1.27 ppm/year over the entire record, which looks right (check the graph).

Almost done. I mentioned that we'd like to report the result in mass flux (Pg per year). So, given that *ppm* in this case means "*μmol* of CO_2 per *mol* of air", we can use the relative molecular weights, the fact that air is about 80% nitrogen and 20% oxygen, and the mass of the atmosphere (~$5x10^{18}$ kg = ~$5x10^{6}$ Pg) to estimate a conversion factor:

```
> pg.per.ppm = 1.e-6 * (12 / (0.8*28 + 0.2*32)) * 5.0e6
```

That's about 2.08 (I remember I used to use 2.13 when I was a student). Now we can report the mean growth rate, the trend, and we can show the graphs in the units we like (Pg CO_2 per year). Of course, we could have done the conversion first...

```
> mean.growth.rate = pg.per.ppm*mean(anngrow)
> growth.rate.trend = pg.per.ppm*coef(regr)[2]                    # help(coef)
> plot(year, pg.per.ppm*co2.grow, type='l', ylab='CO2 (Pg/yr)', xlab='Year')
> lines(annyear, pg.per.ppm*anngrow, lwd=3, col='blue')
> abline(0, 0, lty=2)
> lines(annyear, pg.per.ppm*predict(regr), lty=3, lwd=2, col='red')
```

This is a decent answer to the question, but in some ways we are just getting started with this data set. What might you do next if the carbon cycle was your research focus? How would you begin to explain the patterns you see in the blue line on the graph?

```
> q()
```