SOP 90: Analyzing Climate Trends with R

Marc Los Huertos

July 6, 2017

Contents

Abstract

Trend and Time Series Analyses are very important in environmental monitoring. For our purposes, developing methods to analyze climate data can using a range of tools, from relatively simple methods to advanced statistical modelling.

This document is designed to introduce a few tools to analyze regularly (i.e. daily or monthly) collected data. First, we will use a standard regression model, mixed-effects models, and finally more advanced time-series modelling approaches.

1 Introduction

Trend analysis... versus time series...

the Mann-Kendall Trend Test, seasonal Mann-Kendall Test, correlated seasonal Mann-Kendall Test, partial Mann-Kendall Trend test, (Seasonal) Sen's slope, partial correlation trend test and change-point test after Pettitt.

Detailed knowledge of the statistical methods used in analysis is beyond the scope of MFPH Part A, but methods include:

Chi-square test for linear trend Regression analysis More detailed consideration of analysis is available here

Time series analysis

Time series analysis refers to a particular collection of specialised regression methods that use integrated moving averages and other smoothing techniques to illustrate trends in the data. It involves a complex process that incorporates information from past observations and past errors in those observations into the estimation of predicted values.

Moving averages provide a useful way of presenting time series data, highlighting any long-term trends whilst smoothing out any short-term fluctuations. They are also commonly used to analyse trends in financial analysis. The calculation of moving averages is described in more detail here.

Presentation of trend data

Presentations of time-trend data should usually include the following:

Graphical plots displaying the observed data over time Comment on any statistical methods used to transform the data Report average percent change An interpretation of the trends seen Interpretation of trend data

The results of all ecological studies, including time-series designs should be interpreted with caution:1

Data on exposure and outcome may be collected in different ways for different populations Migration of populations between groups during the study period may dilute any difference between the groups Such studies usually rely on routine data sources, which may have been collected for other purposes Ecological studies do not allow us to answer questions about individual risks References

Bailey L, Vardulaki K, Langham J, Chandramohan D. Introduction to Epidemiology. Open University Press, 2005.

A time series is a sequence of data points, measured typically at successive points in time spaced at uniform time intervals. Examples of time series are the daily closing value of the Dow Jones index and the annual flow volume of the Nile River at Aswan. Time series are very frequently plotted via line charts. Time series are used in statistics, signal processing, pattern recognition, econometrics, mathematical finance, weather forecasting, earthquake prediction, electroencephalography, control engineering, astronomy, and communications engineering.

Time series analysis comprises methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data. Time series forecasting is the use of a model to predict future values based on previously observed values. While regression analysis is often employed in such a way as to test theories that the current values of one or more independent time series affect the current value of another time series, this type of analysis of time series is not called "time series analysis."

Time series data have a natural temporal ordering. This makes time series analysis distinct from other common data analysis problems, in which there is no natural ordering of the observations (e.g. explaining people's wages by reference to their respective education levels, where the individuals' data could be entered in any order). Time series analysis is also distinct from spatial data analysis where the observations typically relate to geographical locations (e.g. accounting for house prices by the location as well as the intrinsic characteristics of the houses). A stochastic model for a time series will generally reflect the fact that observations close together in time will be more closely related than observations further apart. In addition, time series models will often make use of the natural one-way ordering of time so that values for a given period will be expressed as deriving in some way from past values, rather than from future values (see time reversibility.)

2 Understanding the Data: The First Step of Data Analysis

2.1 Data Source and Metadata

2.2 Evaluating the Structure of Data

This is analogous to selection a column or row of numbers in Excel to find the mean and you can usually find it by just looking at your spreadsheet to find the data of interest. In R you have to think a bit about what you want. Using the `str` command is good start, but we could also just look at the top of the observations to see which variables are of interest. To this we use the function `head()`, which is short for header, which shows the variable names and the first six observations.

```
head(Thailand)
```

```
##             STATION STATION_NAME     DATE  PRCP TAVG TMAX
## 1 GHCND:TH000048456 DON MUANG TH 19430101 -9999 27.6 33.9
## 2 GHCND:TH000048456 DON MUANG TH 19430102 -9999 26.8 31.7
## 3 GHCND:TH000048456 DON MUANG TH 19430103 -9999 27.2 32.8
## 4 GHCND:TH000048456 DON MUANG TH 19430104 -9999 27.3 33.3
## 5 GHCND:TH000048456 DON MUANG TH 19430105 -9999 27.8 32.2
## 6 GHCND:TH000048456 DON MUANG TH 19430106 -9999 27.1 32.8
##   TMIN    NewDate
## 1   NA 1943-01-01
## 2 21.7 1943-01-02
## 3 21.1 1943-01-03
## 4 21.1 1943-01-04
## 5 21.1 1943-01-05
## 6 21.1 1943-01-06
```

2.3 Evaluating for Completeness

NA is the R symbol for missing data and R requires the user to be fairly intentional about how to deal with missing data. Missing data usually mean the dataset is biased. In contrast to many software packages, R forces you to acknowledge the implications of missing data, which can be annoying, like a parent reminding you to clean your room or brush your teeth or take a shower once in the while. But the trade is worth it: you have dealt explicitly with missing data.

2.4 Evaluating the Central Tendencies

One of the first things you should do with your data is determine some of the central tendencies. For example, the mean, median, and standard deviation. Also some graphing of the data is also important. For example, what does the distribution of the data look like?

Let's start with the easy stuff. We want to get the mean of the maximum temperatures. That means we need to get the values, named TMAX from the data frame.

Okay, so we want "average." But typing average by itself doesn't show us anything except an error. Let's try `str` again. Notice the dollar symbols. These symbols are use to signify a list of values inside the data frame. To access this list, we type

```
Thailand$TMAX
```

So, now we can get the number of observations, i.e. the length of the vector, by typing

```
length(Thailand$TMAX)

## [1] 27026
```

Okay, let's calculate the mean. In this case, it requires caution. Notice there are NAs in the data.

Typing `mean(Thailand$TMAX)` gives an a ambiguous result, `NA`. Try it. R is basically saying that the mean can not be calculated because of missing values, thus the mean is also missing. So, can we not calculate the mean when data are missing? No, we just have to tell R what to do with missing data. In this case, we tell R to remove them, with the argument `na.rm="TRUE"`, where True can be abbreviated to T. na.rm="TRUE" roughly translates to 'please remove all the NAs.'

Okay as of July 6, 2017, the average is 32.9461248[1]. It will change next month when May 2010 is added to the data set. Now let's determine the median and standard deviation.

```
median(Thailand$TMAX, na.rm=T)

## [1] 33

sd(Thailand$TMAX, na.rm=T)

## [1] 2.336892
```

If you would like a summary of each of the variables, the function is pretty easy to remember–but the output is not exceptionally pleasing.

```
summary(Thailand)

##                STATION            STATION_NAME        DATE
##  GHCND:TH000048456:27026   DON MUANG TH:27026   Min.   :19430101
##                                                 1st Qu.:19610848
##                                                 Median :19800265
##                                                 Mean   :19797426
##                                                 3rd Qu.:19980830
##                                                 Max.   :20170630
##
##       PRCP              TAVG              TMAX            TMIN
##  Min.   :-9999.0   Min.   :-9999.0   Min.   :19.30   Min.   : 2.40
##  1st Qu.:    0.0   1st Qu.:   27.1   1st Qu.:31.60   1st Qu.:23.00
##  Median :    0.0   Median :   28.4   Median :33.00   Median :24.50
##  Mean   :-1532.3   Mean   : -255.2   Mean   :32.95   Mean   :24.02
##  3rd Qu.:    1.0   3rd Qu.:   29.6   3rd Qu.:34.40   3rd Qu.:25.60
##  Max.   :  484.1   Max.   :   34.4   Max.   :40.80   Max.   :30.10
```

[1]How many significant figures should you report? Have I reported this correctly?

```
##                                         NA's   :5066    NA's   :7760
##     NewDate
##  Min.   :1943-01-01
##  1st Qu.:1961-08-31
##  Median :1980-02-29
##  Mean   :1980-03-04
##  3rd Qu.:1998-08-29
##  Max.   :2017-06-30
##
```

Nevertheless, the output gives you a really good idea regarding the central tendencies of the entire data set. Granted typing code might seem like a major step backwards in the computer world, but after a few weeks you will appreciate not having the search through arcane menus to find which button to push–even worse, in these push-button software systems, it often hard to figure out what they are doing. In the case of R, you have a really good idea of what it did, but were much more engaged in the process.

2.5 Evaluating Spread

When the mean and median diverge, it means that the distribution is skewed in some way. Let's see what the distribution looks like by creating a histogram.

```
hist(Thailand$TMAX)
```

The one you have made probably does not look that pretty, but with some more advanced coding, this is what it might look like (Figure 1).

Congratulations, you have made it through the next step in R! You now know how to do an exploratory analysis and even generate a basic histogram to view the distribution of a data set. Next, we use a standard statistical technique to determine the slope of the line and weather the line is statistically significant.

3 Simple Regression

3.1 What is Linear Regression?

Linear regression is the most basic and commonly used predictive analysis. Regression estimates are used to describe data and to explain the relationship between one dependent variable and one or more independent variables. At the center of the regression analysis is the task of fitting a single line through a scatter plot. The simplest form with one dependent and one independent variable is defined by the formula:

$$y = a + b * x. \tag{1}$$

Sometimes the dependent variable is also called the response. The independent variables are also predictor variables. However, Linear Regression Analysis

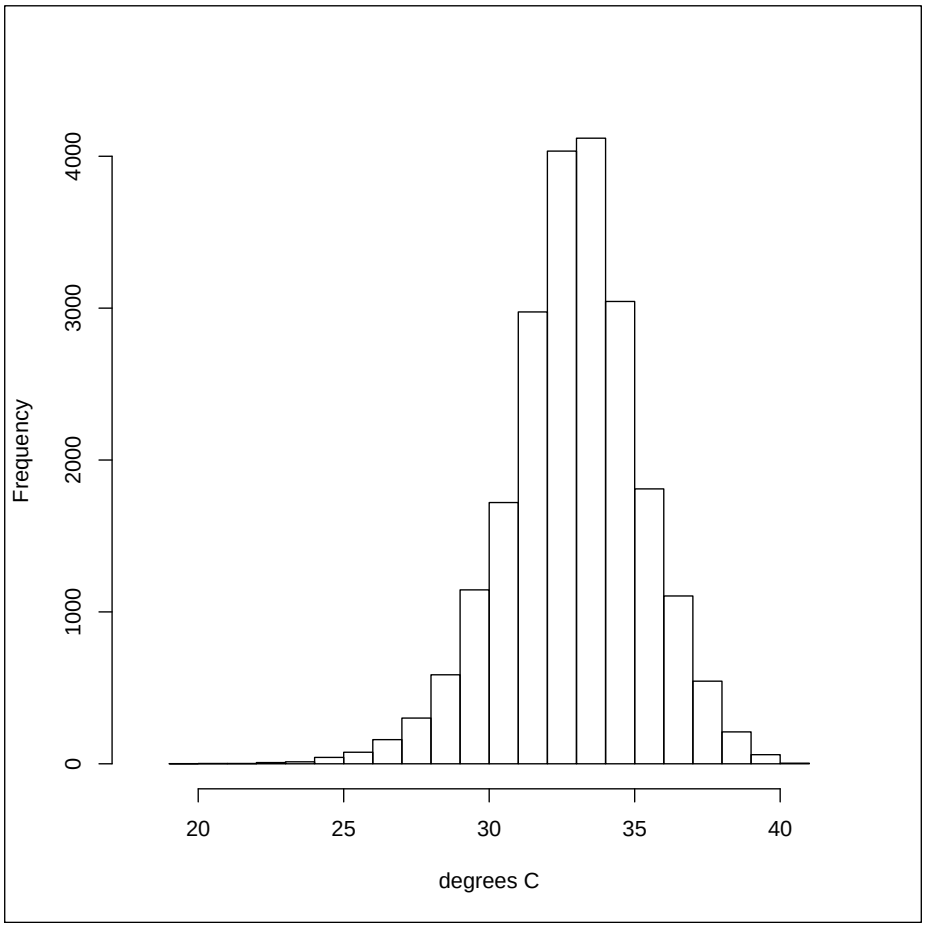


Figure 1: Histogram of Maximum Temperatures, Bangkok, Thailand.

consists of more than just fitting a linear line through a cloud of data points. It consists of 3 stages:

1. analyzing the correlation and directionality of the data,
2. estimating the model, i.e., fitting the line, and
3. evaluating the validity and usefulness of the model.

There are three major uses for Regression Analysis: 1) causal analysis, 2) forecasting an effect, 3) trend forecasting. Other than correlation analysis, which focuses on the strength of the relationship between two or more variables, regression analysis assumes a dependence or causal relationship between one or more independent and one dependent variable.

Firstly, it might be used to identify the strength of the effect that the independent variable(s) have on a dependent variable. Typical questions are what is the strength of relationship between dose and effect, sales and marketing spending, age and income.

Secondly, it can be used to forecast effects or impacts of changes. That is, regression analysis helps us to understand how much the dependent variable will change when we change one or more independent variables. Typical questions are, How much additional Y do I get for one additional unit of X?.

Thirdly, regression analysis predicts trends and future values. The regression analysis can be used to get point estimates. Typical questions are, "What will the price for gold be 6 month from now?" "What is the total effort for a tasks?"

3.2 Regression and Climate Change

For our purposes, we are accessing the impacts of change, however, we may want to extend our projection out 20 years and make a prediction too.

$y = mx + b$, where x is time.

$y\ x$

$$y = mx + b + \epsilon \tag{2}$$

4 Linear Models in R

The use of the linear model is the cornerstone of statistics. So ubiquitous it is rarely explained coherently. The linear model can be summarized at the equation for a line, but with the addition of error. You are probably familiar with the equation for a line where,

$$y = m * x + b \tag{3}$$

This equation defines a line, where m is the slope, b is the y-intercept, and the x and y are coordinates. The linear model is based on this form and is usually written as

$$y \sim \alpha + \beta * x + \epsilon \tag{4}$$

The order is usually changed, where the intercept is first, followed by the slope and x variable and the addition of error or noise. The error is usually symbolized as ϵ. In general, in a statistical model, Greek letters are used and instead of an equals sign, we use a tilde, meaning that that left side of the equation is a function of the right side. Luckily, this is the approximate form that R expects, so if you understand this, you will have a pretty good idea of how to code a linear model in R.

The function to build a linear model is `lm()`. This function is extremely powerful and can be easily implemented, but this is a good time to see what the help menus look like in R.

```
help(lm)
```

I am not showing it here, but you should see a long complex looking help page window pop up. All help files in R are structured the same way, so in spite of the uninterpretable text, written by and for computer programmers, the structure will become familiar. Beginning with the description, the help screen describes the function, how to use it, and give some examples. Admittedly, I rarely understand much of the text, but I find the examples to be very useful! In fact, I suggest you paste the example into R and see what happens, I find this one of the best ways to learn R. Use an example that I know works, then change it to make it do what I want it to do.

Using the linear model, we can analyze several types of data, when the response variable is continuous. If the have a predictor variable that is categorical, then we often analyze the data using the method known as analysis of variance or ANOVA. If the predictor variable is continuous, then we often analyze data using a regression analysis.

Okay, let's see if we can do this for our Mauna Loa data. Let's test if there is a significant change of carbon dioxide concentrations with time. Since both the predictor and response variables are continuous data, this analysis will be a linear regression, but the form and function of the linear model are exactly the same. The linear model would look something like this

$$TMAX \sim \alpha + \beta * time + \epsilon \tag{5}$$

Translating this in R will take some additional tricks besides just getting the code figured out. First, we need to identify the predictor variable in the data frame. There are three variables associated with time: year, month, and decimal.date. Because these data are in a time series, they are serially correlated, meaning that the June sample will be more like the July sample than the August sample. In addition, the June 2010 sample will be similar to the June 2009 sample. These correlation violate the assumption of independence, but for today, we will ignore this violation and just create a linear model in bliss. So, let's use year as the predictor variable and assume there might be some error because each month have slightly different concentrations. For the response variable, we will use the monthly averages, "'average"'. Remember there are some missing data, it will be interesting to note how R deals with that.

First, let's create a plot of data using `plot()`, whose format is `plot(x, y)` or `plot(y  x)`. We will use the later for now,

```
#plot(TMAX ~ NewDate, data=Thailand)
```

Finally, there is one important difference between the linear model that we used in the `aov()` function. This time we use the `lm()` function that arrange the results more in-line with a regression model. This syntax is still pretty straight forward,

Figure 2: Carbon dioxide concentrations sampled for each month at Mauna Loa, Hawaii.

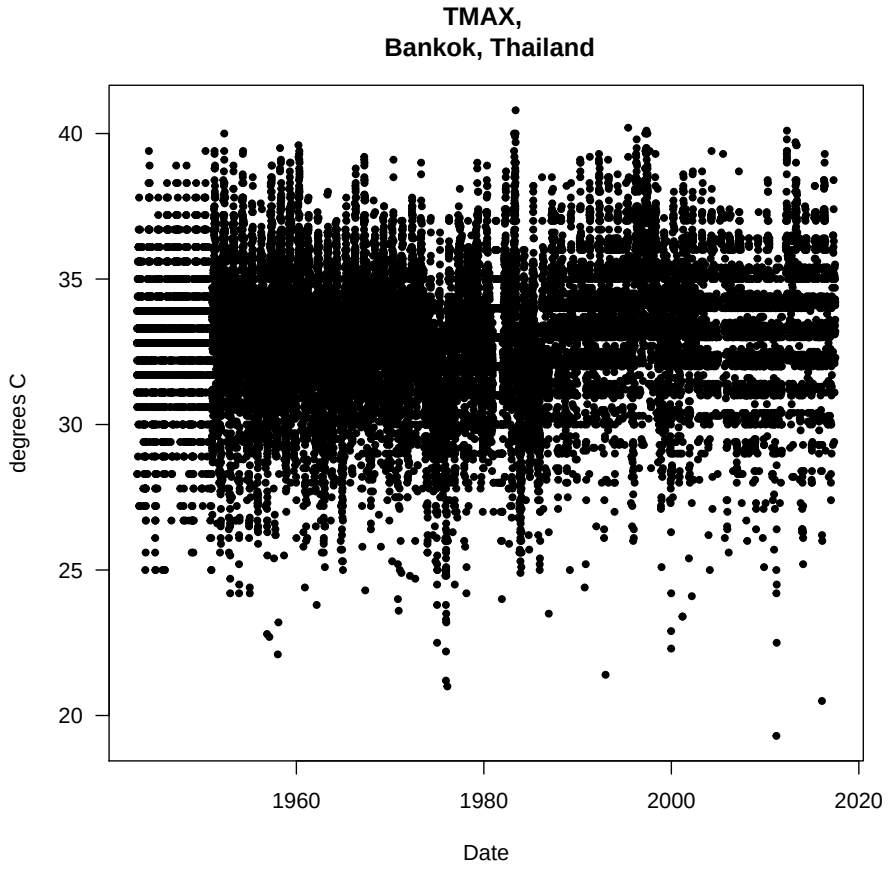
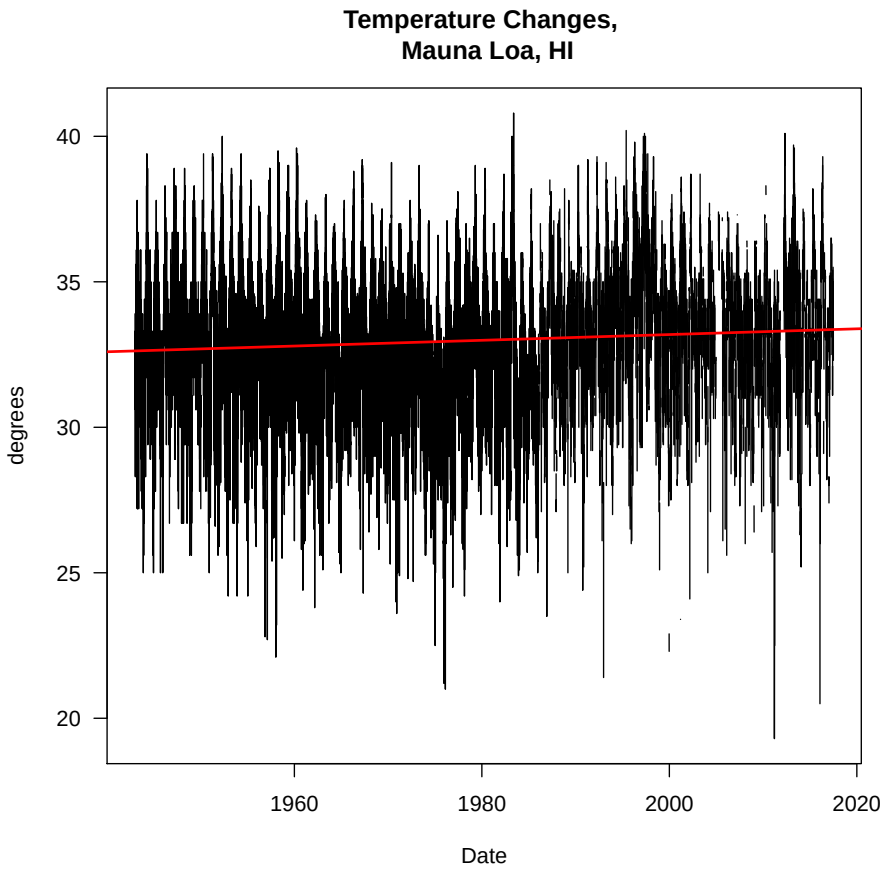


Figure 3: TMAX in Bankok, Thailand.




```
#lm(average ~ year, data=maunaloa)
```

From this model, we learn that the change in CO_2 is ppm $year^{-1}$.[2] Figure **??** shows the increasing concentrations, but also the seasonal variation. Statisticians have more advanced methods to analyze these data then what we have done, but for our purposes the implications are the same. Greenhouse gas emissions are increasing and the estimated rates suggest an increasing rate.

Now let's ask if this value is significant, by putting the linear model into a ANOVA-like table. There are a number of functions that do this and we have seen the `anova()` function above. For linear regression, however, the `summary()` function gives a more complete output.

[2]When I first made this handout the rate of increase was 1.441. Why do you think this rate has changed?

```
summary(lm(TMAX ~ NewDate, data=Thailand))

##
## Call:
## lm(formula = TMAX ~ NewDate, data = Thailand)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.9974  -1.3193   0.0782   1.4717   7.7771
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.289e+01  1.631e-02 2016.35   <2e-16 ***
## NewDate     2.702e-05  2.140e-06   12.62   <2e-16 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.329 on 21958 degrees of freedom
##   (5066 observations deleted due to missingness)
## Multiple R-squared:  0.007202,Adjusted R-squared:  0.007157
## F-statistic: 159.3 on 1 and 21958 DF,  p-value: < 2.2e-16
```

Here we find the that the slope and intercept are highly significant, we have some information on the residuals, and R^2 estimates, etc.

4.1 Creating Monthly Averages

So, let's figure out how to see how changes happen for individual months.

```
#Get months
#Thailand£Month = months(LosAngeles£NewDate) # Creates problems.
Thailand$Month = format(as.Date(Thailand$NewDate), format = "%m")
Thailand$Year = format(Thailand$NewDate, format="%Y")
```

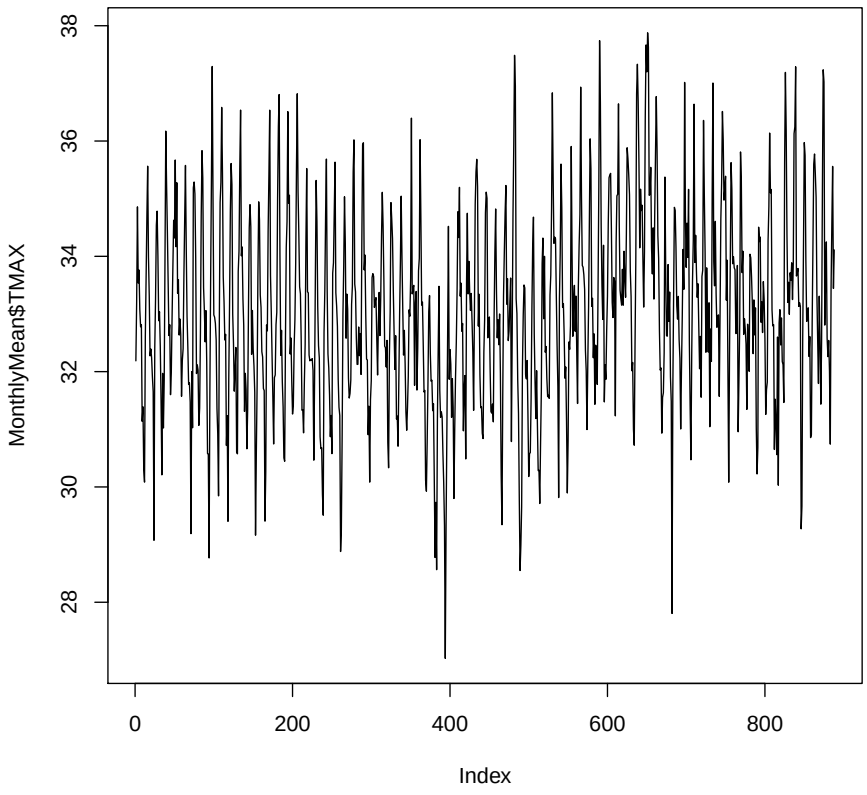
4.2 Creating Monthly Means

```
MonthlyMean = aggregate(TMAX ~ Month + Year, Thailand, mean)

MonthlyMean$YEAR = as.numeric(MonthlyMean$Year)
MonthlyMean$MONTH = as.numeric(MonthlyMean$Month)


MonthlySD = aggregate(TMAX ~ Month + Year, Thailand, sd)
```

```r
MonthlySD$YEAR = as.numeric(MonthlySD$Year)
MonthlySD$MONTH = as.numeric(MonthlySD$Month)
MonthlySD$NewDate = MonthlySD$YEAR + (MonthlySD$MONTH - 1)/12

head(MonthlySD)
```

```
##   Month Year     TMAX YEAR MONTH  NewDate
## 1    01 1943 1.523317 1943     1 1943.000
## 2    02 1943 1.668648 1943     2 1943.083
## 3    03 1943 1.969395 1943     3 1943.167
## 4    04 1943 2.521970 1943     4 1943.250
## 5    05 1943 2.100818 1943     5 1943.333
## 6    06 1943 1.041763 1943     6 1943.417
```

```r
plot(MonthlyMean$TMAX, ty='l')
```



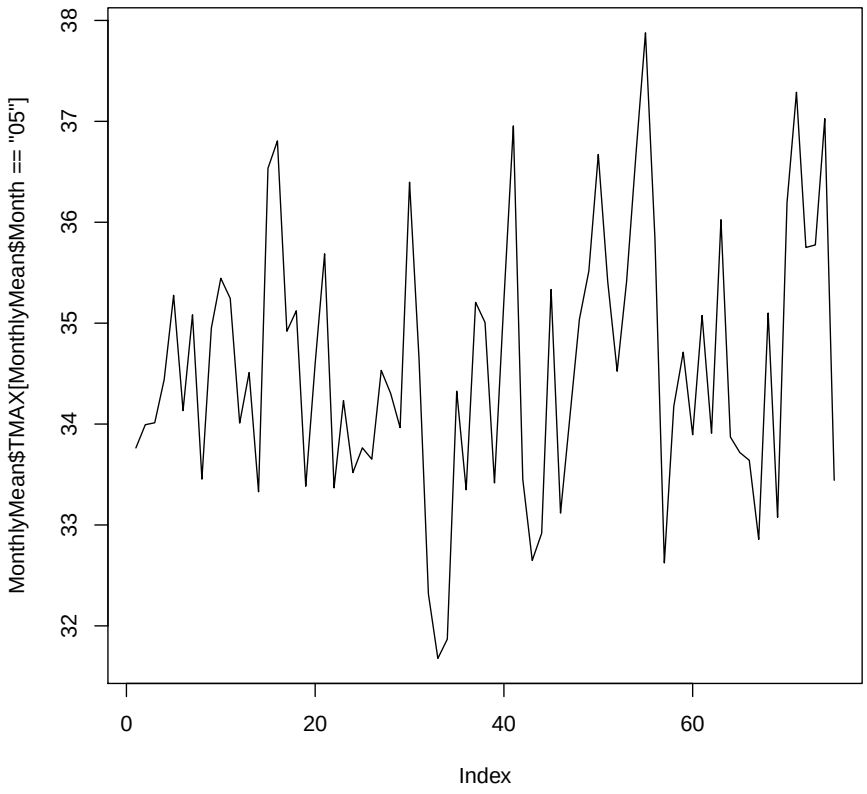
Below is Standard Deviation

```
#plot(MonthlySD£TMAX, ty='l')

#plot(TMAX~ NewDate, data=MonthlySD, ty='l')
#SD.lm <- lm(TMAX~NewDate, data=MonthlySD)
#summary(SD.lm)

#abline(coef(SD.lm), col="red")
```

Selecting for 1 Month – May

Perhaps, we can get a better handle on this stuff if we analyze for just one month at a time – certainly easier to visualize!

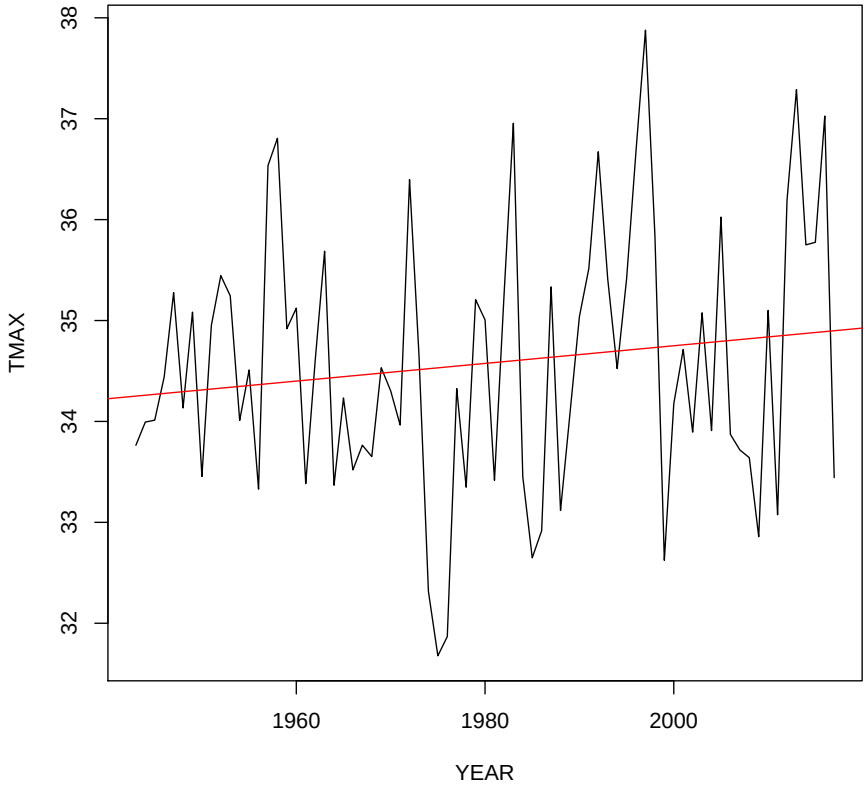
```
plot(MonthlyMean$TMAX[MonthlyMean$Month=="05"], ty='l')
```



```r
plot(TMAX~YEAR, data=MonthlyMean[MonthlyMean$Month=="05",], ty='l')
May.lm <- lm(TMAX~YEAR, data=MonthlyMean[MonthlyMean$Month=="05",])
summary(May.lm)
```

```
##
## Call:
## lm(formula = TMAX ~ YEAR, data = MonthlyMean[MonthlyMean$Month ==
##     "05", ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.85376 -0.93210 -0.04633  0.81231  3.15347
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 17.238621  13.753642   1.253    0.214
## YEAR         0.008756   0.006946   1.261    0.211
##
## Residual standard error: 1.302 on 73 degrees of freedom
## Multiple R-squared:  0.0213,Adjusted R-squared:  0.007897
## F-statistic: 1.589 on 1 and 73 DF,  p-value: 0.2115
```

```r
abline(coef(May.lm), col="red")
```

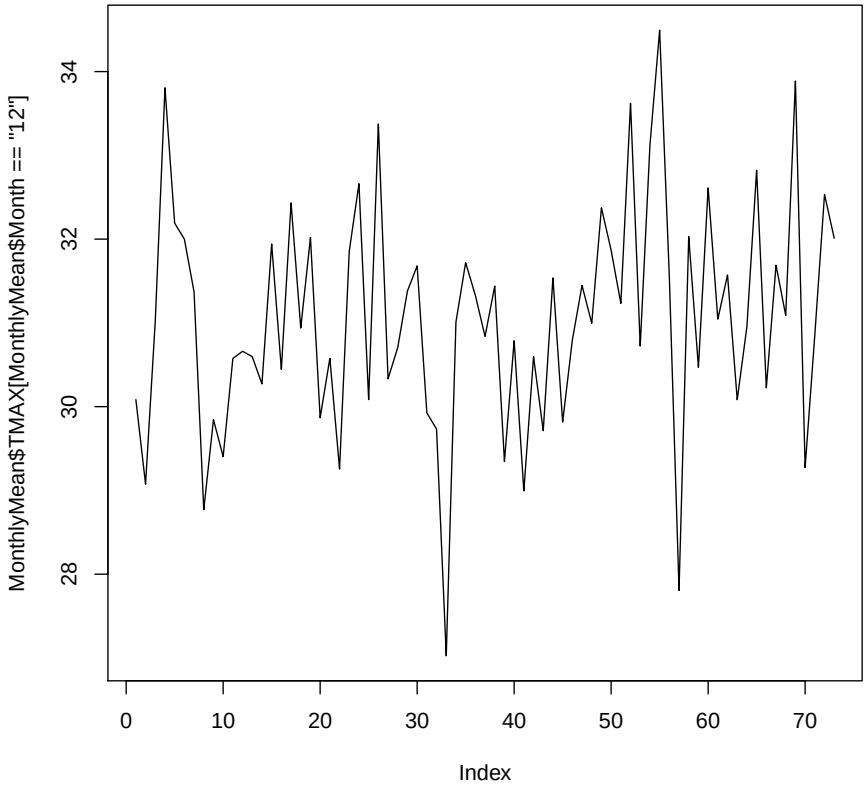


Now, the change is 0.0088 degress C/year or 0.876 degress C/100 years with a probability of 0.2115. Although we can't reject the null hypothesis, we find the method to be fairly straightforward!

4.3 Next Steps

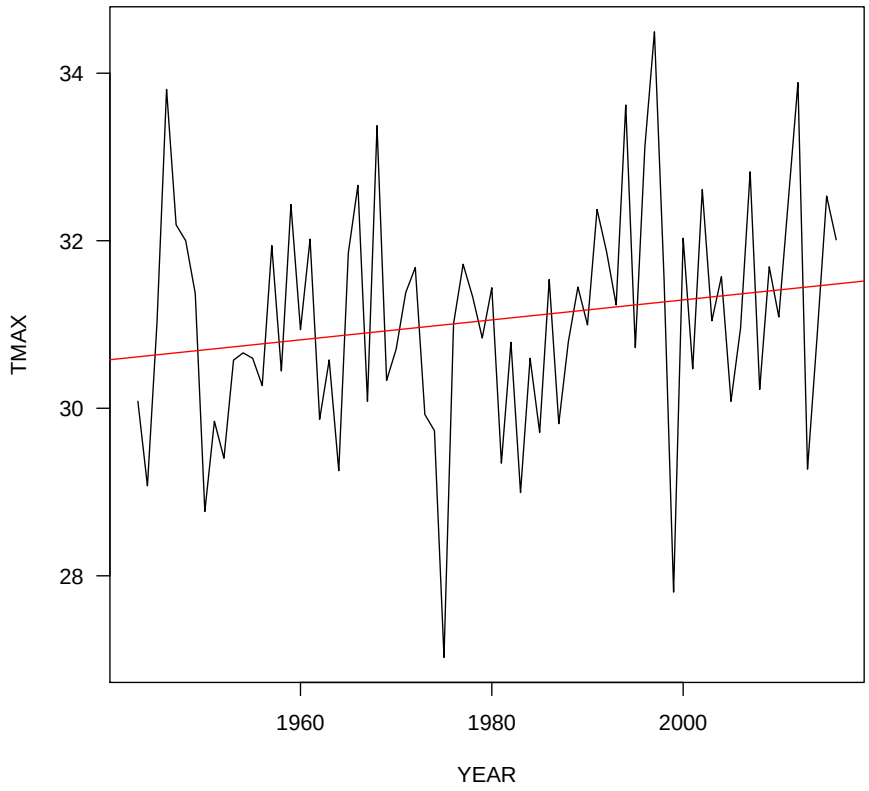
I think you should evaluate every month and see what happens. You might also consider looking at the TMIN as well. Could be important!

Lets try Dec




```
##
## Call:
## lm(formula = TMAX ~ YEAR, data = MonthlyMean[MonthlyMean$Month ==
##     "12", ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9696 -0.8163 -0.1591  0.7210  3.2366
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.465036  14.852894   0.503    0.617
## YEAR         0.011914   0.007505   1.588    0.117
##
## Residual standard error: 1.358 on 71 degrees of freedom
## Multiple R-squared:  0.03428,Adjusted R-squared:  0.02068
```

```
## F-statistic:  2.52 on 1 and 71 DF,  p-value: 0.1168
```



4.3.1 TMIN

1. We create a monthly TMIN mean for each month.

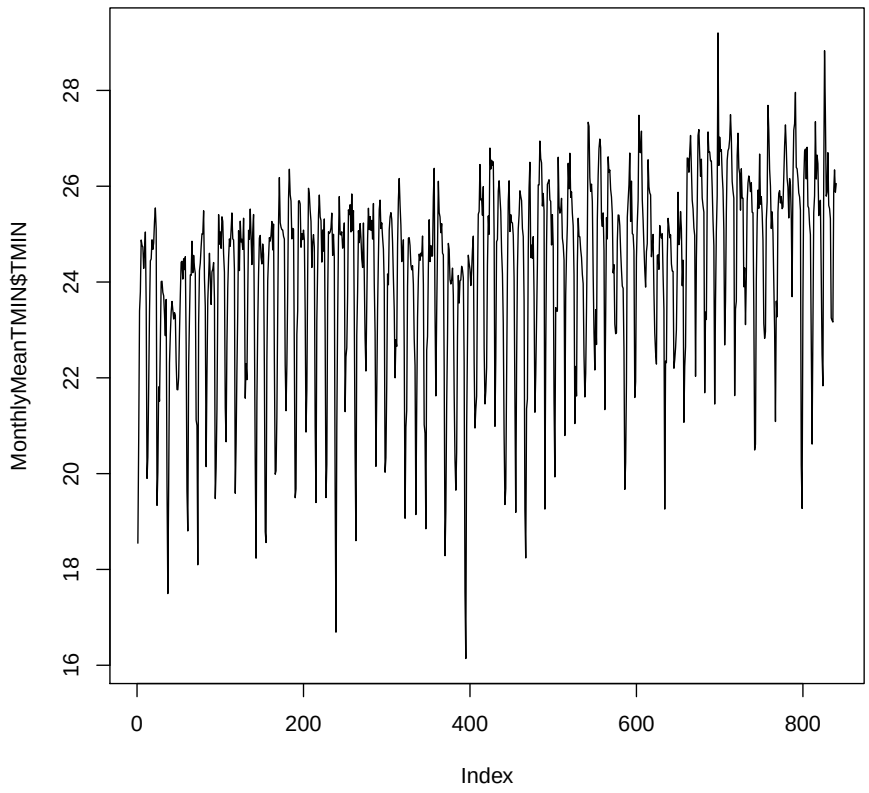
```
MonthlyMeanTMIN = aggregate(TMIN ~ Month + Year, Thailand, mean)

MonthlyMeanTMIN$YEAR = as.numeric(MonthlyMeanTMIN$Year)
head(MonthlyMeanTMIN)

##   Month Year     TMIN YEAR
## 1    01 1943 18.54828 1943
## 2    02 1943 20.73077 1943
## 3    03 1943 23.39655 1943
## 4    04 1943 23.79259 1943
```

```
## 5    05 1943 24.87692 1943
## 6    06 1943 24.76429 1943
```

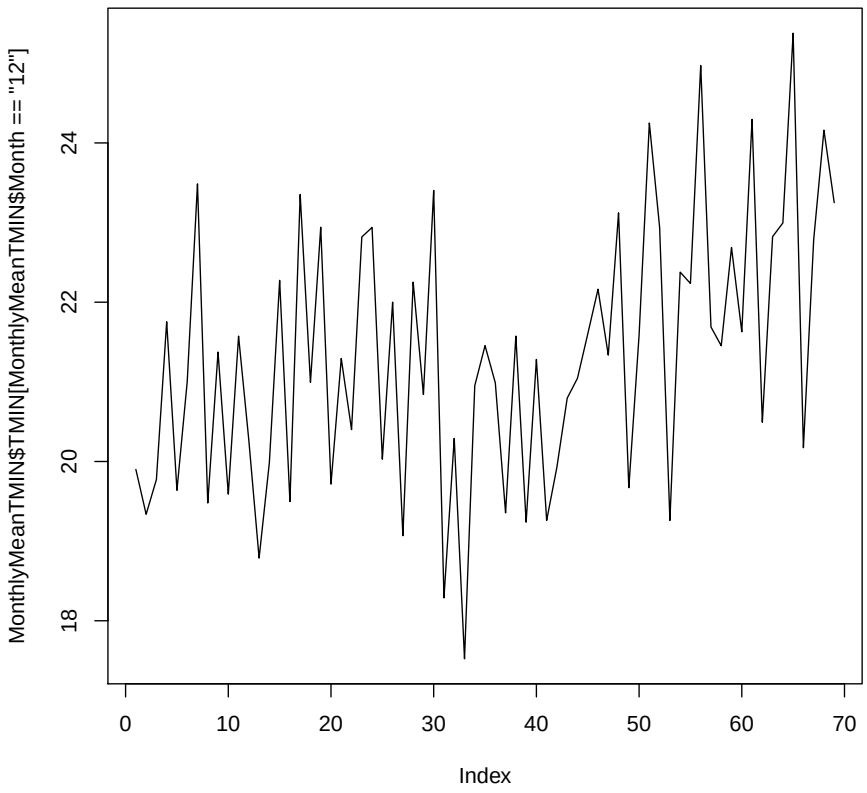
4.4 TMIN

2. Now we plot the mins, and again, find tons of scatter.

```
plot(MonthlyMeanTMIN$TMIN, ty='l')
```

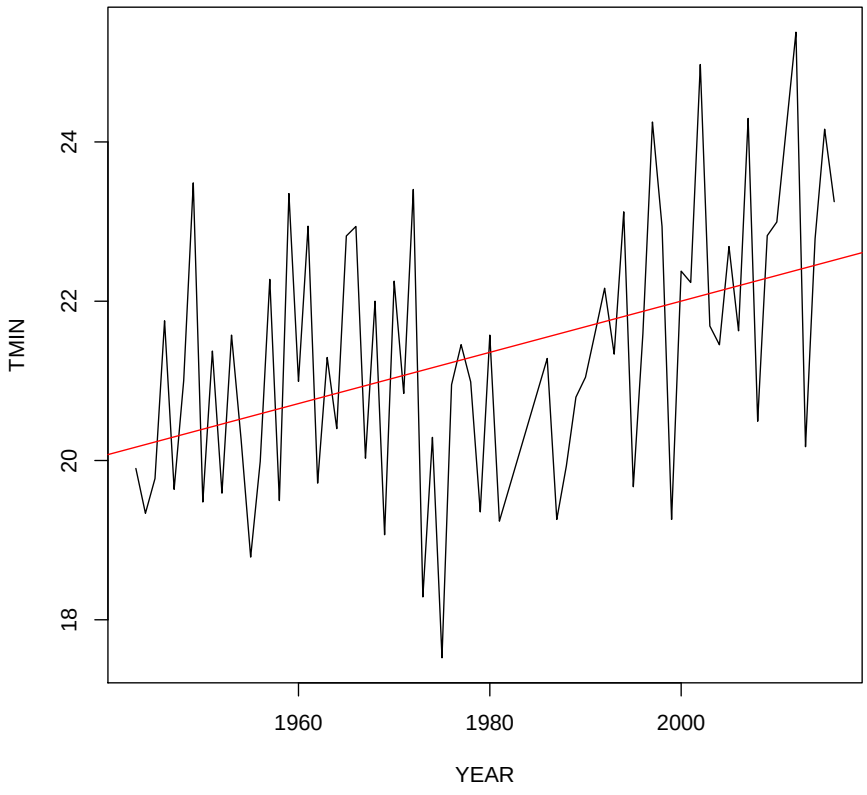



```
plot(MonthlyMeanTMIN$TMIN[MonthlyMeanTMIN$Month=="12"], ty='l')
```




```
plot(TMIN~YEAR, data=MonthlyMeanTMIN[MonthlyMeanTMIN$Month=="12",], ty='l')
Dec.lm <- lm(TMIN~YEAR, data=MonthlyMeanTMIN[MonthlyMeanTMIN$Month=="12",])
summary(Dec.lm)

##
## Call:
## lm(formula = TMIN ~ YEAR, data = MonthlyMeanTMIN[MonthlyMeanTMIN$Month ==
##     "12", ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6746 -0.8672 -0.2573  1.0020  3.1247
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -42.320213  16.700175  -2.534 0.013617 *
```

```
## YEAR          0.032161   0.008439   3.811 0.000303 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.525 on 67 degrees of freedom
## Multiple R-squared:  0.1782,Adjusted R-squared:  0.1659
## F-statistic: 14.52 on 1 and 67 DF,  p-value: 0.0003034

abline(coef(Dec.lm), col="red")
```



4.5 TMIN

3. In this case, we get a slope, 0.0321607 degress/year and a probability of 3.034×10^{-4} and an r-sqaured of 0.178. Cool! As we might expect, the a small amount of the variance is explained by the "Month." Many things predict

temerpature, that year is one, is quite problematic.

4. What we have not determined is the cause. So, be careful when you describe the results, cause and effect cannot be analyzed using this method.

4.5.1 Departure from Mean

```
#PRCP_mean = mean(LosAngeles£PRCP, na.rm=T)

#plot(PRCP~NewDate, data=LosAngeles)
#abline(h=PRCP_mean, col="blue")
```

4.5.2 Experimental Portion — Precipitation

Precipiation might depend more on the departure from the mean (often referred as as normal, whatever that means!). I think it's worth pursuing, but haven't finished the analysis yet.

First, we need a "mean" – The IPCC uses 1961-1990 as a norm, I don't know what is the standard for California, so we should look that up.

Second, we need to remove the missing values and evalaute which years have complete years. If you are missing rainy months, then the whole year should be thrown out – but what about partial years in the drought season?

Third, we will need to decide what level of aggredation – monthly, yearly, etc.

Fourth, in CA the water year starts in Oct 1. Should we follow the same convention?

```
#LosAngeles£PRCP[LosAngeles£PRCP==-9999] <- NA
#YearlySum = aggregate(PRCP ~ Year, LosAngeles, sum)
#YearlySum£YEAR = as.numeric(YearlySum£Year)
#YearlyMean = mean(YearlySum£PRCP)
```

A yearly mean, based on the annual sum for the entire records. Not sure this is appropriate.

Figure has points of the yearly sum of rainfall and the blue line mean. The greenline is the trend and red line is a five year running average, I think! I am still trying to understand what the code is doing.

```
#plot(PRCP~YEAR, data=YearlySum, las=1, ty="p")
#abline(h=YearlyMean, col="blue")
#YearlySum.lm = lm(PRCP~YEAR, data=YearlySum)
#abline(coef(YearlySum.lm), col="green")

#n <- 5
#k <- rep(1/n, n)
```

```
#k

#y_lag <- stats::filter(YearlySum£PRCP, k, sides=1)
#lines(YearlySum£YEAR, y_lag, col="red")
```

```
#summary(YearlySum.lm)
```

4.6 Problems with a Simple Regression Model

Regression models, like all statistics, rely on certain assumptions. Violations of these assumptions reduces the validity of the model. If the violations are serious, then the model could be misleading or even incorrect.

Here is a list of assumptions to produce a valid regression model:

Homogeneity of Variance

something else

Assumptions about e , the error term: t i. E(et) = 0, zero mean ii. E(et) = s , constant variance 2 2 iii. E(et , Xt) = 0, no correlation with X iv. E(e , e) , no autocorrelation. t t-1 v. e Normally distributed (for hypothesis testing). t 3. Assumption four is especially important and most likely not to be met when using time series data.

Autocorrelation. 1. It is not uncommon for errors to track themselves; that is, for the error a time t to depend in part on its value at t - m, where m is a prior time period.

4.6.1 Model Diagnostics

With every statistical test done, researchers validate their model in some way or anther. Often this entails the use of diagnostics, a standardize battery of procedures to check to see if the data are following the assumptions.

In R four plots are created by default. To see them all at the same time, we need to change the graphical parameters so the graphics window expects four panels, in this case a 2 rows and two columns.

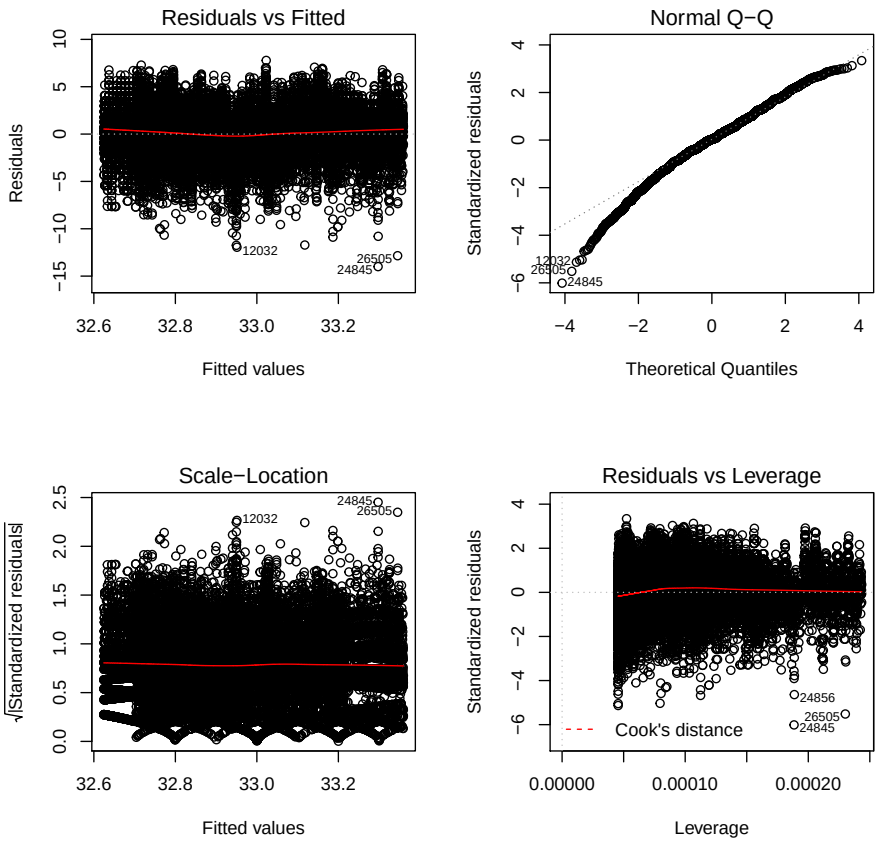
```
par(mfrow=c(2,2))
```

Try not to get bogged down in the code at this point. But it is a useful thing to remember.

To determine the validity of linear model assumptions (e.g. normality or heterogeneity of variance), you have probably used statistical tests; in contrast statisticians almost exclusively look at diagnostic plots. Why? When assumptions are violated the tests to determine violations do not perform well. So, let's see how to look at these assumptions graphically with these diagnostic plots.

Figure 4: Default diagnostic plots for a linear model in R.

```
par(mfrow=c(2,2))
plot(lm(TMAX ~ NewDate, data=Thailand))
```



Linear models should have diagnostic plots that do not have any obvious structure or pattern. In this case, Figure 4.6.1 should show a great deal remaining structure in the residuals. Although for today, we are not going to try to interpret these figures, but you should notice there is a ton of unaccounted structure, i.e. variance, in the model. This is due, in part, to a violation of independence; these data are serially correlated and the model does not account for that and is inappropriate because of this. It also appears that a straight-line model does not fit well and a curvilinear should be investigated.

A properly specified model is shown in

5 Relaxing Model Assumptions

5.1 Using Sources of Error in the Model

Instead of letting autocorrelation be 'hidden' problem in the data, we can incorporate the correlation structure into the model and use it to our advantage – create a better, i.e. unbiased estimate of the model parameters.

5.2 Generalized Least Square (GLS) and Autocorrelation

```
library(nlme)
TMAX.gls = gls(TMAX ~ NewDate, data = Thailand, na.action=na.omit)
summary(TMAX.gls)

## Generalized least squares fit by REML
##   Model: TMAX ~ NewDate
##   Data: Thailand
##        AIC      BIC    logLik
##   99477.04 99501.03 -49735.52
##
## Coefficients:
##                Value  Std.Error   t-value p-value
## (Intercept) 32.89085 0.01631207 2016.3503       0
## NewDate      0.00003 0.00000214   12.6213       0
##
##  Correlation:
##         (Intr)
## NewDate -0.268
##
## Standardized residuals:
##         Min          Q1         Med          Q3         Max
## -6.01131472 -0.56658497  0.03357611  0.63203004  3.33992877
##
## Residual standard error: 2.328514
## Degrees of freedom: 21960 total; 21958 residual
```

```
TMAX.gls2 = gls(TMAX ~ NewDate, data = Thailand, correlation = corAR1(form=~1), na.action=na
summary(TMAX.gls2)

## Generalized least squares fit by REML
##   Model: TMAX ~ NewDate
##   Data: Thailand
##        AIC      BIC    logLik
##   81141.43 81173.42 -40566.72
##
## Correlation Structure: AR(1)
##  Formula: ~1
##  Parameter estimate(s):
##     Phi
## 0.75252
##
## Coefficients:
##                Value  Std.Error  t-value p-value
## (Intercept) 32.89103 0.04341180 757.6518       0
## NewDate      0.00003 0.00000569   4.7312       0
##
##  Correlation:
##         (Intr)
## NewDate -0.268
##
## Standardized residuals:
##         Min          Q1         Med          Q3         Max
## -6.00952180 -0.56664256  0.03361602  0.63184878  3.33922871
##
## Residual standard error: 2.329056
## Degrees of freedom: 21960 total; 21958 residual

anova(TMAX.gls, TMAX.gls2)

##           Model df      AIC      BIC    logLik   Test
## TMAX.gls      1  3 99477.04 99501.03 -49735.52
## TMAX.gls2     2  4 81141.43 81173.42 -40566.72 1 vs 2
##           L.Ratio p-value
## TMAX.gls
## TMAX.gls2 18337.61  <.0001
```

5.3 Adding Seasonality

6 More Sophisticated Approaches

Methods for time series analyses may be divided into two classes: frequency-domain methods and time-domain methods. The former include spectral analysis and recently wavelet analysis; the latter include auto-correlation and cross-correlation analysis. In time domain correlation analyses can be made in a filter-like manner using scaled correlation, thereby mitigating the need to operate in frequency domain.

Whether or not you wish to forecast or not has nothing whatsoever to do with correct time series analysis. Time series methods can develop a robust model which can be used simply to characterize the relationship between a dependent series and a set of user-suggested inputs (a.k.a. user-specified predictor series) and empirically identified omitted variables be they deterministic or stochastic.Users at their option can then extend the "signal" into the future i.e. forecast with uncertainties based upon the uncertainty in the coefficients and the uncertainty in the future values of the predictor . Now these two kinds of empirically identified "omitted series" can be classified as 1) deterministic and 2) stochastic. The first type are simply Pulses, Level Shifts , Seasonal Pulses and Local Time Trends whereas the second type is represented by the ARIMA portion of your final model. When one omits one or more stochastic series from the list of possible predictors, the omission is characterized by the ARIMA component in your final model. Time series modelers refer to ARIMA models as a "Poor Man's Regression Model" because the past of the series is being used as a proxy for omitted stochastic input series.

7 Advanced Methods

You may want to examine the GAM package in R, as it can be adapted to do some (or all) of what you are looking for. The original paper (Hastie & Tibshirani, 1986) is available via OpenAccess if you're up for reading it.

Essentially, you model a single dependent variable as being an additive combination of 'smooth' predictors. One of the typical uses is to have time series and lags thereof as your predictors, smooth these inputs, then apply GAM.

This method has been used extensively to estimate daily mortality as a function of smoothed environmental time series, especially pollutants. It's not OpenAccess, but (Dominici et al., 2000) is a superb reference, and (Statistical Methods for Environmental Epidemiology with R) is an excellent book on how to use R to do this type of analysis.