Replication of ”Detection of Undocumented Changepoints: A Revision of the Two-Phase Regression Model”

by Robert Lund and Jaxk Reeves

AMS Journal of Climate: Sept 2002

Kristen L. Gore
Columbia University Statistics Department

March 10, 2011

Contents

1 Abstract

Changes in station instrumentation, location, or observer can often induce artificial discontinuities into climatic time series. This can lead to undetected changes in trends and can lead to increased variation in climate models if this ”unadjusted” data is used. The method presented in this paper uses a modified form of pointwise regression to detect undocumented changepoints in climate series. These techniques were tested on two datasets: 1) CO2 data from the Mauna Loa Observatory in Hawaii, and 2) temperature data from Chula Vista, California.

2 Introduction

3 Data/Methodology

Data:

- **Test dataset 1: Temperature Data**
  - Location: Chula Vista, CA
  - Time Span: 1919-1996
  - Yearly data
  - Instrumentation replaced in 1966. Moved in 1982. Instrumentation changed again in 1985.

- **Test dataset 2: CO_2 Data**
  - Location: Mauna Loa, Hawaii
  - Time Span: 1959-1999
  - Monthly data
  - ppm/yr

Methodology:
For the Chula Vista dataset, temperature was modeled as follows:

$$X_t = \begin{cases} \mu_1 + \alpha_1 t + \epsilon_t, & 0 < t \leq c \\ \mu_2 + \alpha_2 t + \epsilon_t, & c < t \leq n. \end{cases} \tag{1}$$

A quadratic model was used to monitor CO_2 at Mauna Loa. More on this in the appendix.

$H_o : \nexists$ changepoint vs. $H_A : \exists$ changepoint at time c

$$F_c = \frac{(SSE_{red} - SSE_{full})/2}{SSE_{Full}/(n-4)} \tag{2}$$

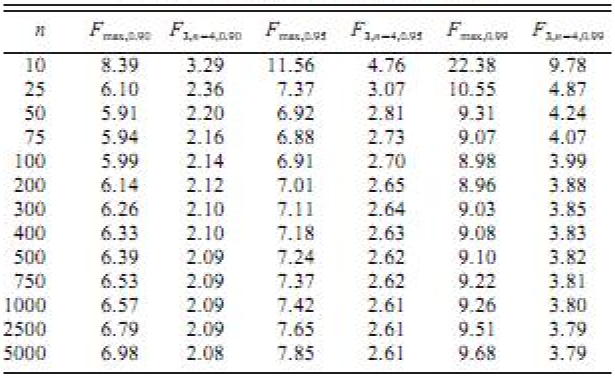
$$SSE_{Full} = \sum_{t=1}^{c}(X_t - \hat{\mu}_1 - \hat{\alpha}_1 t)^2 + \sum_{t=c+1}^{n}(X_t - \hat{\mu}_2 - \hat{\alpha}_2 t)^2 \tag{3}$$

$$SSE_{Red} = \sum_{t=1}^{n}(X_t - \hat{\mu}_{Red} - \hat{\alpha}_{Red} t)^2 \tag{4}$$

Two Test Statistic Options: 1) F^* and 2) $F_{max} = \max_{1 \leq c \leq n} F_c$

Under null, use $F^* \sim F_{3,n-4}$. If using a quadratic model, $F^* \sim F_{3,n-6}$ under the null. Use simulations to calculate quantiles (critical values) for F_{max}. Table provided below.

TABLE 1. The F_{max} and $F_{3,n-4}$ percentiles.



n	$F_{max,0.90}$	$F_{3,n-4,0.90}$	$F_{max,0.95}$	$F_{3,n-4,0.95}$	$F_{max,0.99}$	$F_{3,n-4,0.99}$
10	8.39	3.29	11.56	4.76	22.38	9.78
25	6.10	2.36	7.37	3.07	10.55	4.87
50	5.91	2.20	6.92	2.81	9.31	4.24
75	5.94	2.16	6.88	2.73	9.07	4.07
100	5.99	2.14	6.91	2.70	8.98	3.99
200	6.14	2.12	7.01	2.65	8.96	3.88
300	6.26	2.10	7.11	2.64	9.03	3.85
400	6.33	2.10	7.18	2.63	9.08	3.83
500	6.39	2.09	7.24	2.62	9.10	3.82
750	6.53	2.09	7.37	2.62	9.22	3.81
1000	6.57	2.09	7.42	2.61	9.26	3.80
2500	6.79	2.09	7.65	2.61	9.51	3.79
5000	6.98	2.08	7.85	2.61	9.68	3.79

The authors suggest using linear interpolation to find the approximate critical F_{max} value. If you know the possible changepoints, F* is preferred. However, if you have no idea where the changepoints might be, F_{max} is preferred. F_{max} is more conservative.

Once you detect a changepoint, fit piecewise model, calculate residuals, and rerun this test to see if any other changepoints exist. (Recursive process).

Note: Different from regression because it takes into account the fact that c is unknown. Drawback: Although the authors averaged over the months to remove any periodicity, this method doesn't take into account the fact that the errors are correlated. I think an autoregressive model would be more appropriate (but certainly not trivial).

4 Results

Chula Vista:
Note: I don't know how they put together the reference dataset (”regular” averaging or inverse distance weighting), and I also can't find the data for one of the sites used in the reference series, so I'm in the process of working out that situation.

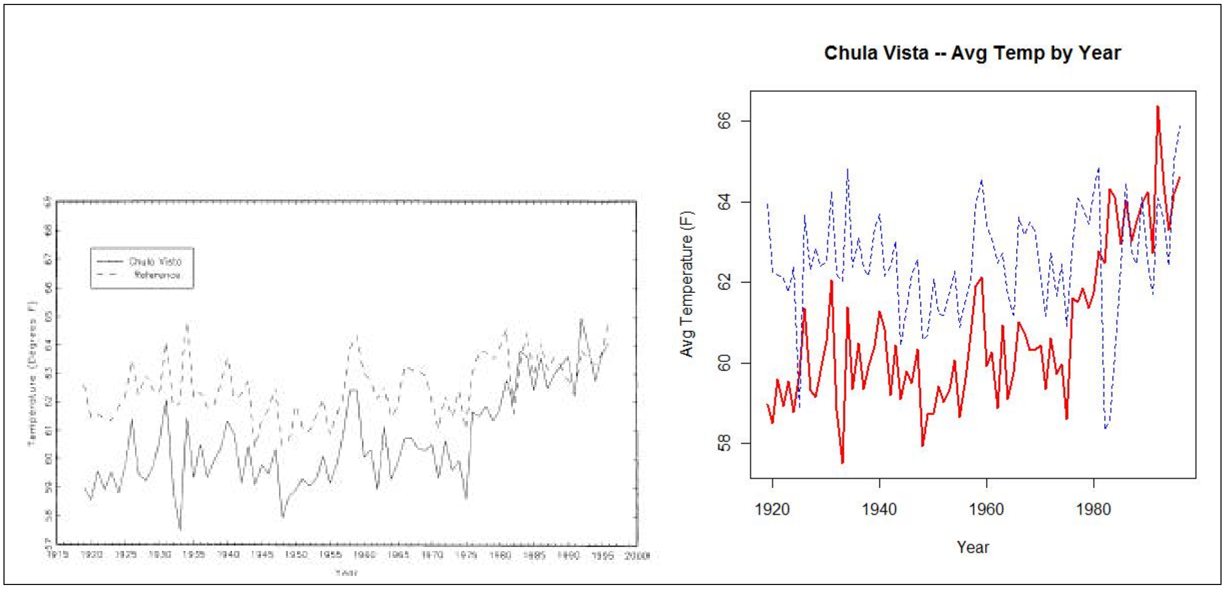


Figure 1: Temperature vs Year

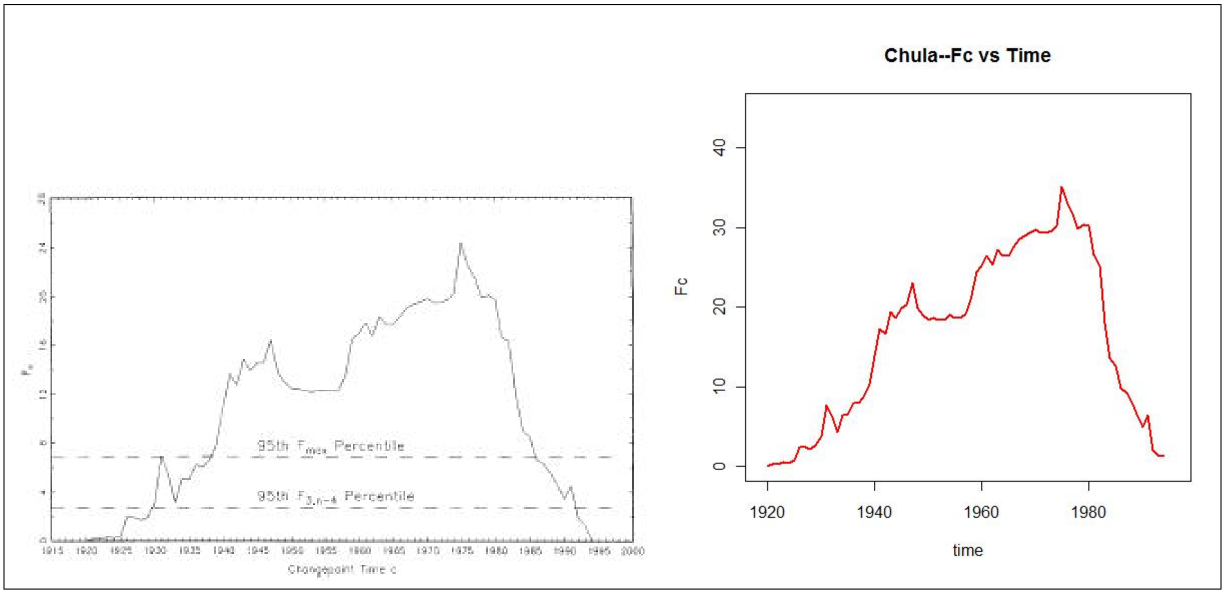


Figure 2: Fc vs Year

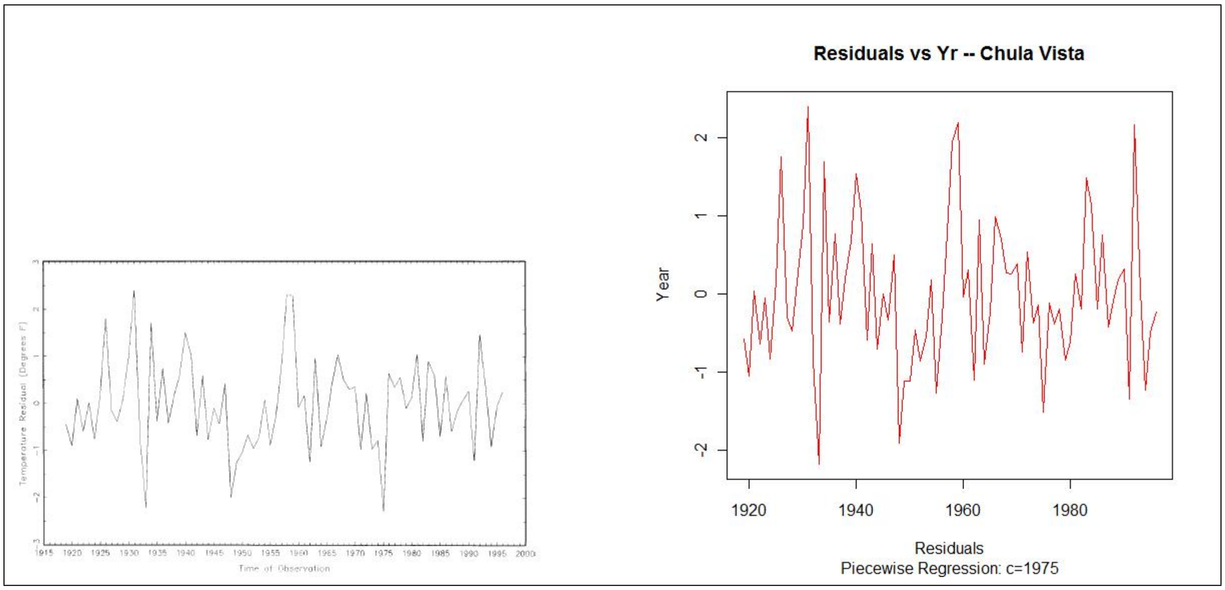


Figure 3: Residuals of Piecewise Regression Model with c=1975

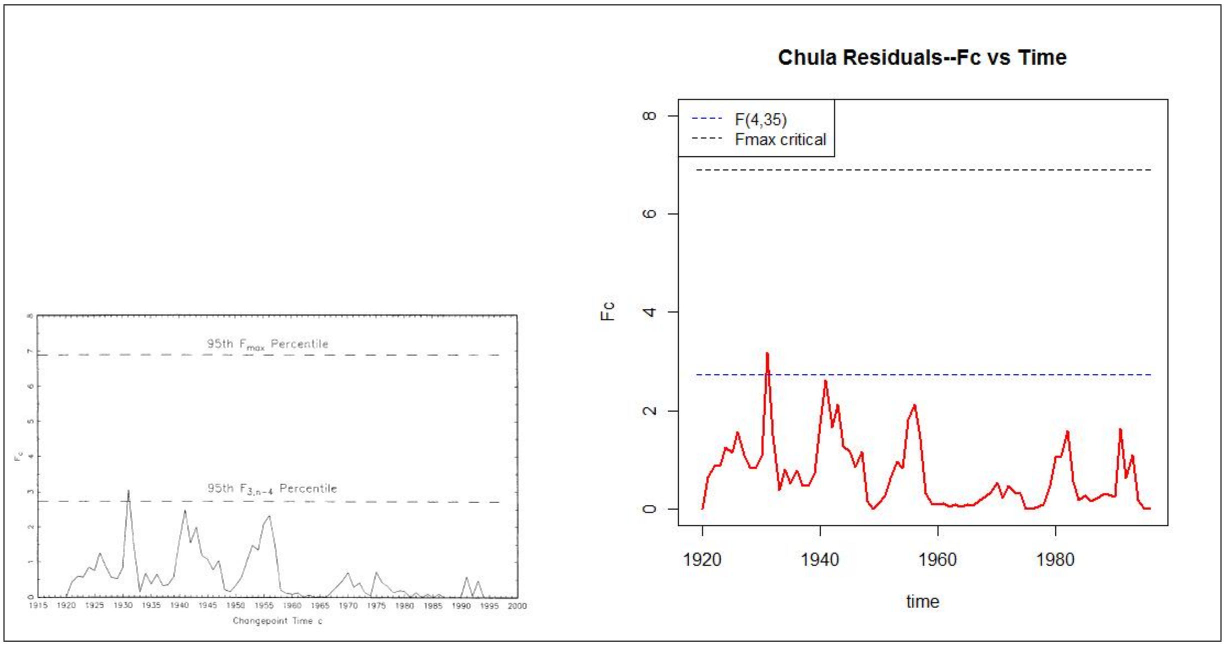


Figure 4: Residual Fc vs Year

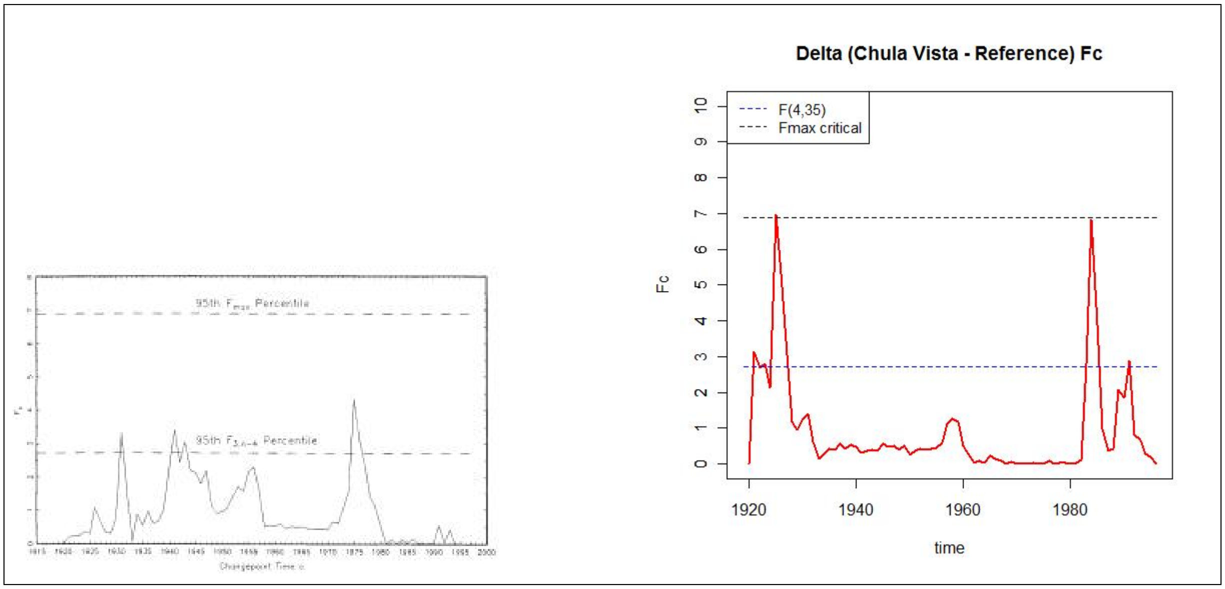
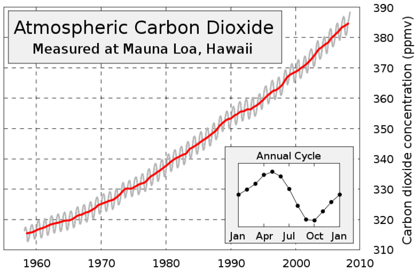


Figure 5: Chula Minus Reference Fc vs Yr

Mauna Loa:



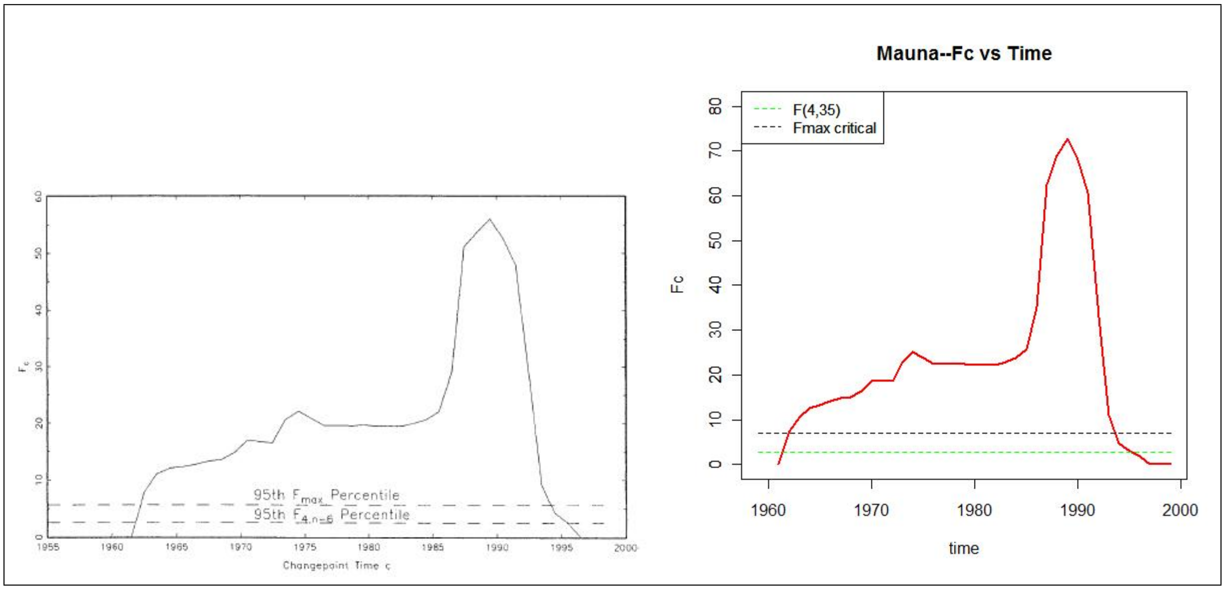


Figure 6: Fc vs Year

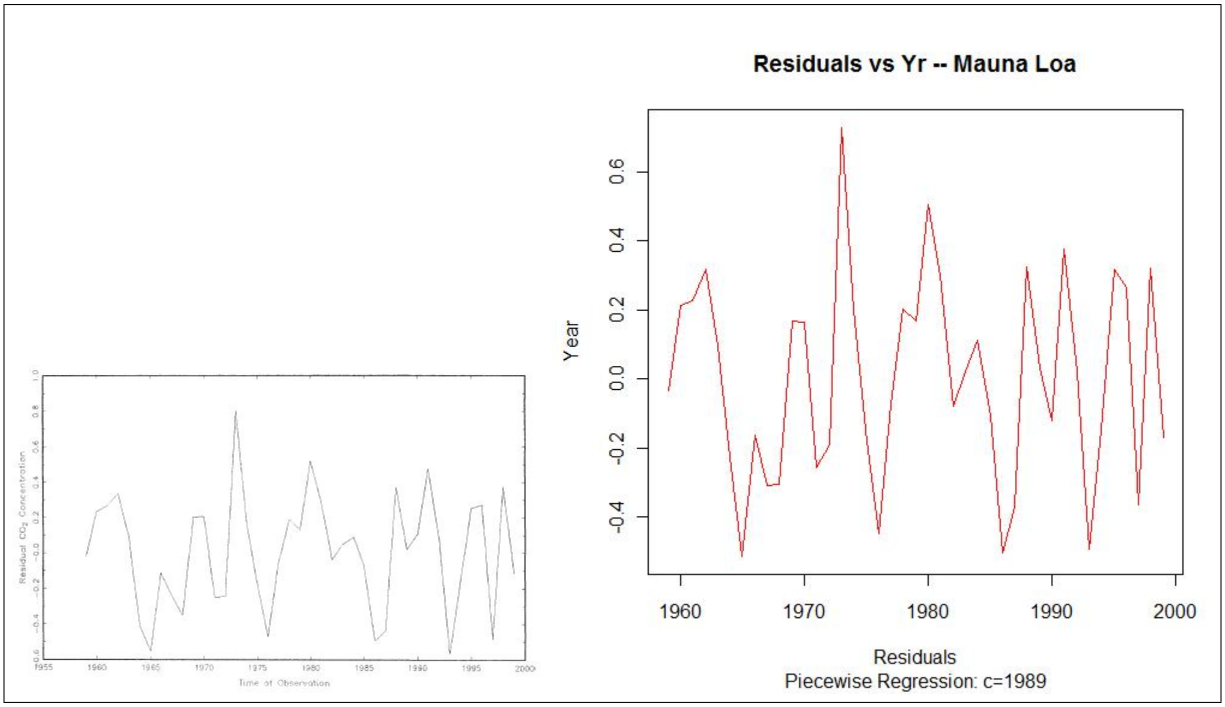


Figure 7: Residuals of Piecewise Regression with c=1989

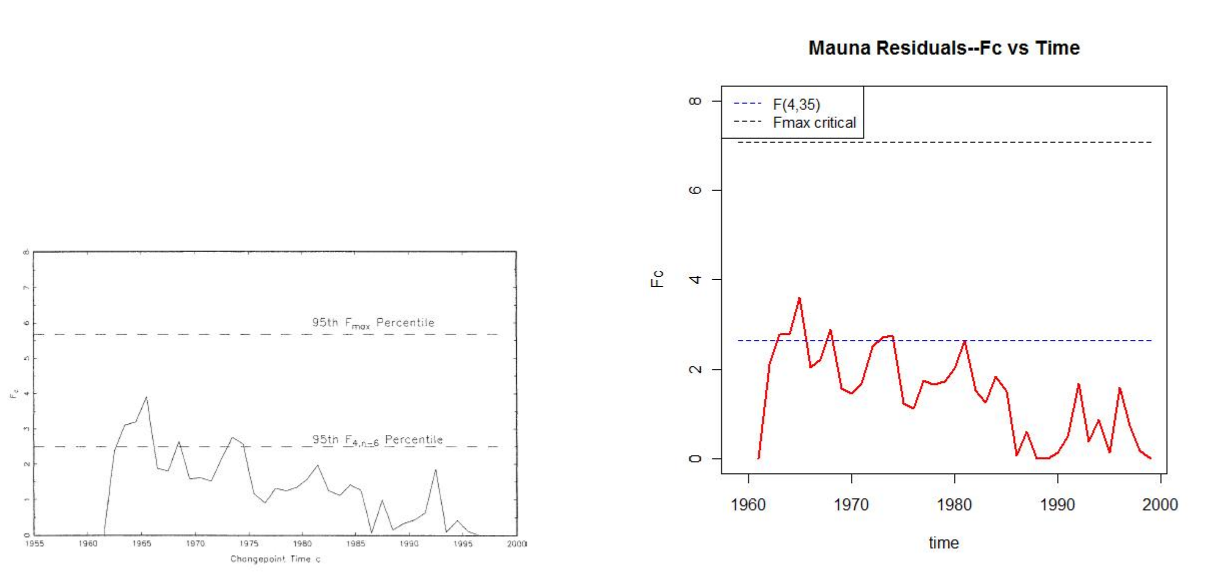


Figure 8: Residual Fc vs Year

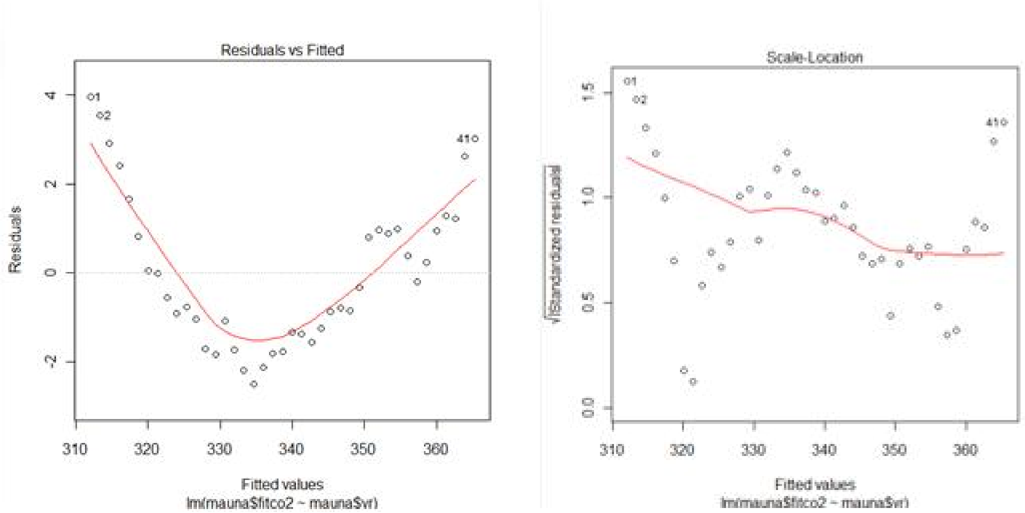
5 Conclusions

Easily replicable (for the most part). Authors used inadequate quadratic model for Mauna Loa data. Also, ata used in Chula Vista was computed via averaging over a lot of missing values (no imputation methods...EM, spline, etc). Thus, may be some bias in the data. Regression approach is somewhat inadequate because although they averaged over all of the months to remove the periodicity in the data, the yearly quantities are still correlated b/c this is climate data (= violation of iid assumption in regression-esque setting). Autocorrelation structure is much better-suited to climate data.

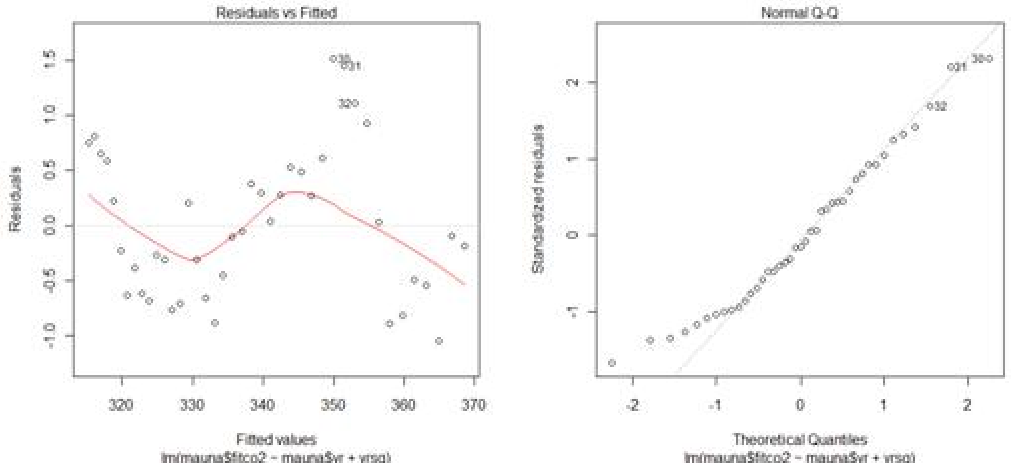
6 Appendix

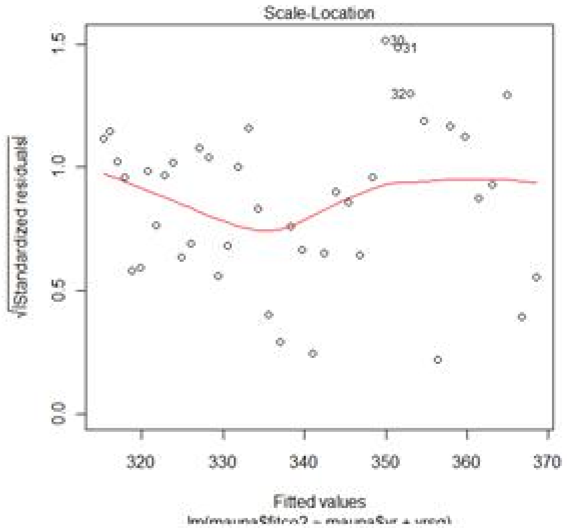
Quadratic Model for Mauna Loa: To understand why the authors chose to fit a quadratic model to the Mauna Loa data, I first fit the first-order model used in the Chula Vista case. The results of the diagnostic tests are found below:

First-Order Model Diagnostics:



Second-Order Model Diagnostics:





Clearly the second-order model is *better*, but these results suggest that perhaps the quadratic model isn't fully adequate either. If time allows, I'll go back, find a better model to fit this data, and try to see if this method still yields the same (or similar) results.

Code:

```
## Kristen Gore
## Changept Detection Replication
################################################################################
rm(list = ls(all = TRUE)) ## clear saved workspace
library(segmented)  ## package for piecewise regression
##setwd("/hpc/scratch/stats/users/klg2130/reprod/")
##.libPaths("/hpc/scratch/stats/users/klg2130/rpackages/")

setwd("C:/Users/Kristen/Documents/Reprod Special Course/")

## FUNCTIONS ###################################################################

f.alphahat=function(c,ds){ ## for the full model
    n=nrow(ds)
    t1=c(1:c)
    x1=ds[1:c,2]
    t2=c((c+1):n)
    x2=ds[(c+1):n,2]
    num1=sum((t1-mean(t1))*(x1-mean(x1)))
    num2=sum((t2-mean(t2))*(x2-mean(x2)))
    den1=sum((t1-mean(t1))^2)
    den2=sum((t2-mean(t2))^2)
    a1=num1/den1
    a2=num2/den2
    return(c(a1,a2))    ## alphahat1, alphahat2
}

f.muhat=function(c,ds){   ## will return mu1 and mu2
    n=nrow(ds)
    t1=c(1:c)
    x1=ds[1:c,2]
    t2=c((c+1):n)
    x2=ds[(c+1):n,2]
    muhat1=mean(x1)-f.alphahat(c,ds)[1]*mean(t1)
    muhat2=mean(x2)-f.alphahat(c,ds)[2]*mean(t2)
    return(c(muhat1,muhat2))
}

f.alphared=function(ds){
    n=nrow(ds)
    t=ds[,1]
    x=ds[,2]
    alpha=12*sum((x-mean(x))*t)/(n*(n+1)*(n-1))
    return(alpha)
}
```

```
f.muhatred=function(ds){
    n=nrow(ds)
    t=ds[,1]
    x=ds[,2]
    return(mean(x-f.alphared(ds)*t))
}

f.SSE=function(c,ds,type){ ##type="f" or "r" (full or reduced)
    SSE=NA
    n=nrow(ds)
    if(type=="f" | type=="full"){ ## FULL MODEL SSE
        ds1=ds[1:c,]
        ds2=ds[(c+1):n,]
        mod1=lm(ds1[,2]~1+ds1[,1])
        mod2=lm(ds2[,2]~1+ds2[,1])
        SSE1=sum(resid(mod1)^2) ## extract SSE
        SSE2=sum(resid(mod2)^2)
        SSE=SSE1+SSE2
    }
    if(type=="r" | type=="reduced"){
        ##t=ds[,1]
        ##x=ds[,2]
        ##SSE=sum((x-f.muhatred(ds)-f.alphared(ds)*t)^2) ## <--alt way to calculate SSE
        mod=lm(ds[,2]~1+ds[,1])
        SSE=sum(resid(mod)^2)
    }
    if(is.nan(SSE) | is.na(SSE)){SSE=NA}
    return(SSE)
}

f.Fc=function(c,ds=mauna2){
    n=nrow(ds)
    SSE.r=f.SSE(c,ds,"r")
    SSE.f=f.SSE(c,ds,"f")
    Fc=((SSE.r-SSE.f)/2)/(SSE.f/(n-4))
    if(is.na(Fc) | is.nan(Fc)){Fc=NA}
    return(Fc)
}

f.Fcvec=function(ds){
    fc=c()
    fc[1]=0
    for(i in 2:(nrow(ds)-1)){
        fc[i]=f.Fc(i,ds)
    }
    fc[nrow(ds)]=0
    ##print(length(fc))
    return(fc)
```

```
}

f.Fmax=function(Fc){
        return(c(max(Fc),which(Fc==max(Fc))))
}


f.plotfc=function(ds,dsname,title,y=80){
    fc=f.Fcvec(ds)
    if(dsname=="mauna2"){
        x=mauna[,1] ## years for mauna
        fc[1:2]=NA ## b/c using quadratic instead of first-order model
        fc[3]=0
        }
    if(dsname!="mauna2"){
        x=chula[,1]
        fc[1]=NA
        fc[2]=0
        }
    plot(x,fc,type="l",lwd=2,xlab="time",ylab="Fc",main=title,col="red",ylim=c(0,y))
}

f.jpgplot=function(ds,dsname,title){
        jpeg(paste(dsname,"fc.jpg",sep=""))
        f.plotfc(ds,dsname,title)
        dev.off()
}

##f.plotfc(chula3,"chula3","Chula--Fc vs Time")


################################################################################
## CHULA VISTA
################################################################################

## REFERENCE DATASET
avalon=read.table("avalon.txt",sep=",",header=T); avalon$date=as.Date(avalon$date,format=’%m/
cuyamaca=read.table("cuyamaca.txt",sep=",",header=T); cuyamaca$date=as.Date(cuyamaca$date,for
indio=read.table("indio.txt",sep=",",header=T); indio$date=as.Date(indio$date,format=’%m/%d/%
redlands=read.table("redlands.txt",sep=",",header=T); redlands$date=as.Date(redlands$date,for

ref1=merge(cuyamaca, indio, by = "date",all.x=T)
ref2=merge(ref1, avalon, by = "date",all.x=T)
ref3=merge(ref2, redlands, by = "date",all.x=T)

## since ’merge’ keeps crashing R, I have to code this manually...~sigh~
## brace yourself...it’s 4am, and I’m coding...it’s really bad...view with caution...
```

```
tempa=rep(NA,times=nrow(cuyamaca))
tempi=rep(NA,times=nrow(cuyamaca))
tempr=rep(NA,times=nrow(cuyamaca))

for(i in 1:nrow(avalon)){
    tempa[which(avalon$date[i]==cuyamaca$date)]=avalon[i,2]
}
for(i in 1:nrow(indio)){
    tempi[which(indio$date[i]==cuyamaca$date)]=indio[i,2]
}
for(i in 1:nrow(avalon)){
    tempr[which(redlands$date[i]==cuyamaca$date)]=redlands[i,2]
}

ref.month=cbind(cuyamaca,tempa,tempi,tempr)
ref.month$yr=as.factor(format(ref.month$date,’%Y’))
reftemp=c()
for(i in 1:78){
    a=as.matrix(ref.month[ref.month$yr==(i+1918),2:5])
    reftemp[i]=mean(a,na.rm=T)
}

refds=as.data.frame(cbind(levels(ref.month$yr),reftemp))
names(refds)=c("yr","temp")
refds$yr=as.numeric(as.character(refds$yr)); refds$temp=as.numeric(as.character(refds$temp))

## ACTUAL CHULA DS #########################################

chulaold=read.csv("chulavista.csv",header=TRUE,col.names=c("date","avgtemp","tmin","tmax")) #
chulaold$date=as.Date(chulaold[,1],format=’%m/%d/%Y’)
chulaold$yr=as.factor(format(chulaold$date,’%Y’))
chulaold2=chulaold[,c(5,2)]  ## just yr and avg temp

temp=c()
for(i in 1:78){
    a=chulaold2[chulaold2$yr==(i+1918),2]
    temp[i]=mean(a,na.rm=T)
}

chula=as.data.frame(cbind(c(1919:1996),temp))
names(chula)=c("yr","temp")
chula$temp[78]=64.6083
## this record was incomplete in the SCO database, so I had to find another source (NCDC). sa
## MNTM is monthly temperature
chula2=chula
chula2$yr=c(1:nrow(chula))

## Plotting Temp vs Yr for CV and Reference ds
```

```
plot(chula$yr,chula$temp,type="l",main="Chula Vista -- Avg Temp by Year",lwd=2,col="red",xlab
        points(refds,type="l",lty="dashed",col="blue") ## add the reference time series

f.plotfc(chula2,"chula2","Chula--Fc vs Time")

## FITTING A PIECEWISE LINEAR REGRESSION MODEL WITH c=1975
cv.1=chula[chula$yr<=1975,]
cv.model1=lm(cv.1$temp~cv.1$yr)
cv.2=chula[chula$yr>1975,]
cv.model2=lm(cv.2$temp~cv.2$yr)
cvresids=c(resid(cv.model1),resid(cv.model2))
plot(chula$yr,cvresids,xlab="Residuals",type="l",ylab="Year",col="red",main="Residuals vs Yr

cvresidsds=as.data.frame(cbind(c(1:length(cvresids)),cvresids))

f.plotfc(cvresidsds,"cvresidsds","Chula Residuals--Fc vs Time",8)
legend("topleft", c("F(4,35)","Fmax critical"),col=c("blue","black"),cex=0.9,lty=c("dashed","
points(c(1919,1996),c(F.crit,F.crit),type="l",lty="dashed",col="blue")
points(c(1919,1996),c(Fmax.crit,Fmax.crit),type="l",lty="dashed",col="black")


## Delta Ref & Chula ##############################

delta=as.data.frame(cbind(c(1:nrow(chula)),chula$temp-refds$temp))
names(delta)=c("yr","temp")
delta1=delta
delta1$yr=chula[,1]
f.plotfc(delta,"delta","Chula Minus Reference Fc",60)
legend("topleft", c("F(4,35)","Fmax critical"),col=c("blue","black"),cex=0.9,lty=c("dashed","
points(c(1919,1996),c(F.crit,F.crit),type="l",lty="dashed",col="blue")
points(c(1919,1996),c(Fmax.crit,Fmax.crit),type="l",lty="dashed",col="black")

## to find the yr in which the (relative) changept occurs:
f.Fmax(f.Fcvec(delta)) ##Fmax=52.1055 | occurs at yr index 63 (yr=1981)

## Critical Values:
f.interp=function(n){  ##linear interpolation for Fmax critical value
        b1=(6.88-6.91)/(75-100)
        b0=6.91-100*b1
        y=b1*n+b0
        return(y)
}  ## Note: only good for 75<=n<=100
F.crit=qf(0.95,3,78-4) ## = 2.72828
Fmax.crit=f.interp(78) ## = 6.8836


## want the residual of the delta Fc statistics
delta.1=delta1[delta1$yr<=1981,]
delta.model1=lm(delta.1$temp~delta.1$yr)
delta.2=delta1[delta1$yr>1981,]
delta.model2=lm(delta.2$temp~delta.2$yr)
deltaresids=c(resid(delta.model1),resid(delta.model2))

deltaresidsds=as.data.frame(cbind(c(1:length(deltaresids)),deltaresids))
f.plotfc(deltaresidsds,"deltaresidsds","Delta (Chula Vista - Reference) Fc",10)
axis(2,at=seq(0,10,1),labels=T)
legend("topleft", c("F(4,35)","Fmax critical"),col=c("blue","black"),cex=0.9,lty=c("dashed","
points(c(1919,1996),c(F.crit,F.crit),type="l",lty="dashed",col="blue")
points(c(1919,1996),c(Fmax.crit,Fmax.crit),type="l",lty="dashed",col="black")


#plot(chula$yr,deltaresids,xlab="Residuals",type="l",ylab="Year",col="red",main="Delta (Chula

f.Fcvec
################################################################################
################################################################################

################################################################################
## MAUNA
################################################################################
f.SSE = function(c,ds,type){  ## Quadratic Model for Mauna Loa
    n=nrow(ds)
    SSE=NA
    if(type=="f" | type=="full"){ ## FULL MODEL SSE
        ds1=ds[1:c,]
        ds2=ds[(c+1):n,]
        timesq1=ds1[,1]^2
        timesq2=ds2[,1]^2

        mod1=lm(ds1[,2]~1+ds1[,1]+timesq1)  ## quadratic model
        mod2=lm(ds2[,2]~1+ds2[,1]+timesq2)
        SSE1=sum(resid(mod1)^2) ## extract SSE
        SSE2=sum(resid(mod2)^2)
        SSE=SSE1+SSE2
    }
    if(type=="r" | type=="reduced"){ ## REDUCED MODEL SSE (no changept)
        timesq=ds[,1]^2
        mod=lm(ds[,2]~1+ds[,1]+timesq)
        SSE=sum(resid(mod)^2)
    }
    if(is.nan(SSE) | is.na(SSE)){SSE=NA}
    SSE
}

f.interp=function(n){  ##linear interpolation for Fmax critical value
        b1=(6.92-7.37)/(50-25)
        b0=6.92-50*b1
        y=b1*n+b0
        return(y)
}  ## Note: only good for 25<=n<=50


## END OF MAUNA FUNCTIONS
###########################



## MAUNA #################################
mauna=read.csv("ml-5.csv",header=TRUE)
plot(mauna)
mauna2=mauna
mauna2$yr=c(1:nrow(mauna2))
 ## NOTE: will use a quadratic model for this data.  X_t=mu+alpha*t+beta*t^2


## calculating critical value Fmax* for F dist with 3 num df and n-6=41-6=35 df:
F.crit=qf(0.95,4,35) ## = 2.64
Fmax.crit=f.interp(41)

## PLOTTING THE MAUNA F STATISTICS:
f.plotfc(mauna2,"mauna2","Mauna--Fc vs Time")
axis(2,at=seq(0,80,10),labels=T)
legend("topleft", c("F(4,35)","Fmax critical"),col=c("blue","black"),cex=0.9,lty=c("dashed","
points(c(1959,1999),c(F.crit,F.crit),type="l",lty="dashed",col="green")
points(c(1959,1999),c(Fmax.crit,Fmax.crit),type="l",lty="dashed",col="black")
## Comments: The overall pattern matches perfectly, but my Fmax is higher than Lund's.
##           Fortunately it doesn't change the results of the analysis.  Data discrepancy?

## FITTING A PIECEWISE LINEAR REGRESSION MODEL WITH c=1989
yrsq=mauna$yr^2
mauna.model=lm(mauna$fitco2~mauna$yr+yrsq) #note that 0 as the true model has not interc

mauna.1=mauna[mauna$yr<=1989,]
yrsq.1=mauna.1$yr^2
mauna.model1=lm(mauna.1$fitco2~mauna.1$yr+yrsq.1)
mauna.2=mauna[mauna$yr>1989,]
yrsq.2=mauna.2$yr^2
mauna.model2=lm(mauna.2$fitco2~mauna.2$yr+yrsq.2)
maunaresids=c(resid(mauna.model1),resid(mauna.model2))
plot(mauna$yr,maunaresids,xlab="Residuals",type="l",ylab="Year",col="red",main="Residuals vs

##piecewise.mauna=segmented(mauna.model,seg.Z=~mauna$yr,psi=1989)

#########################
## Fc analysis on Residuals
```

```
mauna.iteration2=cbind(mauna2$yr,maunaresids)
row.names(mauna.iteration2)=c(1:nrow(mauna.iteration2))
mauna.iteration2=as.data.frame(mauna.iteration2)
names(mauna.iteration2)=c("yr","res")

f.plotfc(mauna.iteration2,"mauna2","Mauna Residuals--Fc vs Time",8)
axis(2,at=seq(0,80,10),labels=T)
legend("topleft", c("F(4,35)","Fmax critical"),col=c("blue","black"),cex=0.9,lty=c("dashed","
points(c(1959,1999),c(F.crit,F.crit),type="l",lty="dashed",col="blue")
points(c(1959,1999),c(Fmax.crit,Fmax.crit),type="l",lty="dashed",col="black")
```