Austria Warming

J.A.Hartigan Yale University September 2013

1 Global Warming

from <u>http://en.wikipedia.org/wiki/Global_warming</u>

Global warming is the rising average temperature of Earth's atmosphere and oceans since the late 19th century and its projected continuation. Since the early 20th century, Earth's average surface temperature has increased by about 0.8 °C (1.4 °F), with about two thirds of the increase occurring since 1980. Warming of the climate system is unequivocal, and scientists are more than 90% certain that most of it is caused by increasing concentrations of greenhouse gases produced by human activities such as deforestation and the burning of fossil fuels. .. Climate model projections .. indicate that during the 21st century the global surface temperature is likely to rise a further 1 to $6^{\circ}C$. The ranges of these estimates arise from the use of models with differing sensitivity to greenhouse gas concentrations. Although our output of 29 gigatons of CO2 is tiny compared to the 750 gigatons moving through the carbon cycle each year, it adds up because the land and ocean cannot absorb all of the extra CO2. About 40% of this additional CO2 is absorbed. The rest remains in the atmosphere, and as a consequence, atmospheric CO2 is at its highest level in 15 to 20 million years (Tripati 2009). (A natural change of 100ppm normally takes 5,000 to 20,000 years. The recent increase of 100ppm has taken just 120 years). An increase in global temperature will cause sea levels to rise ... Warming is expected to be strongest in the Arctic and would be associated with continuing retreat of glaciers, permafrost and sea ice. If global mean temperature increases to 4°C above pre-industrial levels, the limits for human adaptation are likely to be exceeded in many parts of the world, while the limits for adaptation for natural systems would largely be exceeded throughout the world. Hence, the ecosystem services upon which human livelihoods depend would not be preserved.

2 Worldwide weather station locations

from

http://cdiac.ornl.gov/epubs/ndpictures/ndp041/graphics/ndp041.temp.gif

picture("pictures/ndp041_temp.png")

GHCN Version 1 Temperature Stations



There are at present about 40000 weather stations worldwide, concentrated in the U.S., Europe, and east Asia. There are few long established stations, and the more recent stations offer no evidence that the present increase in temperature is unusual compared to, say, 19th century values. There were perhaps only 100 weather stations in 1850. Berlin had one in 1701, Vienna had one in 1775, and Austria had 10 in 1850. We will look at what happened at these Austrian stations. We find data and plots from many long established European stations at:

http://www.zamg.ac.at/histalp/Statmod_AT_T01.html

3 Vienna and long record German station

Temperatures increased about two degrees C^o over the last century. picture("pictures/g3.png")



picture("pictures/g4.png")



The two stations have similar patterns over time.

4 Location of Stations

See the data preparation section for the source of the file "temp.csv":
temp <- read.csv(file="data/temp.csv", header=T,
as.is=T)</pre>

Remove an unwanted X from the variable names: names(temp) <- gsub("X", "", names(temp)) temp[, 1:6]

	loc	long	lat	alt :	1859 :	1860
1	Graz - Universität	15.45	47.08	377	9.0	7.7
2	Innsbruck-Universität	11.38	47.26	609	8.6	7.4
3	Klagenfurt-Flughafen	14.32	46.65	459	8.0	6.4
4	Kremsmünster	14.13	48.05	389	8.5	6.9
5	Linz-Stadt	14.29	48.30	263	8.6	7.2
6	Obergurgel-Vent	11.03	46.87	1938	1.1	-0.3
7	Salzburg-Flughafen	13.00	47.80	450	8.4	6.8
8	St. Andrä im Lavanttal	14.83	46.76	402	7.6	6.1
9	Villacher Alpe	13.67	46.60	2160	0.0	-1.6
10	Wien-Hohe Warte	16.36	48.25	209	10.2	8.9

```
tiff("pictures/Station Locations.tif",w=1200,h=680)
par(mar=c(2,2,5,2))
boundary <- read.table(
paste("data/", "boundary.txt", sep=""), header=F)
temp$lat[9] <- 46.55 # fix a bad latitude
plot(boundary[,2], boundary[, 3],type="l",lwd=3,
main="Long record Austrian temperature stations"
,cex.main=3, col="red", axes=F,xlab="", ylab="")
for( st in 1:10)
lines(temp$long[st]+c(0, temp$alt[st]/2000),
        temp$lat[st]+c(0, temp$alt[st]/2000),
        col="blue", lwd=3)
text(temp$long, temp$lat, temp$loc, cex=2, pos=4)
dev.off()</pre>
```



The heights of stations are indicated by the blue lines . The two high stations are in the Alps.

5 All stations yearly temperatures

```
Plot average temperature for each station for each year:
year <- 1859:2008
# Separate temperatures by 0.1C to avoid
overstrikes.
tempS <- temp
for(cols in 1:150)
tempS[,cols+4]<-</pre>
Separate(unlist(tempS[,cols+4]),0.1)
colors <- rainbow(10) [rank(tempS[,5])]</pre>
mt <- apply(temp[, -(1:4)], 2, mean)</pre>
tiff("pictures/Annual Temperatures.tif", w=1200,
h=1200)
Grid( c(1750, seq(1850, 2000, 50), 2008), seq(-2, 12, 2),
ylab="AustrianYearlyTemperatures, 1859-2008/Degrees
C'', at=c(1860,1748), cex=c(3, 2))
for( i in 1:10)
points(year,tempS[i,-(1:4)],col=colors[i],pch=16)
text(pos=4, 1760, Separate(temp[,5], 0.4),temp$loc,
cex=2.5, col=colors, xpd=T)
text( 1850, 5, "MEAN", cex=2)
lines(year, mt, lwd=4)
dev.off()
```



The station names have been separated. The 10 station values for each year have been separated. The pattern of increasing temperatures is similar in all stations, with the two mountain stations shifted about 8°C lower than the plains stations.

6 Residuals from the all Station mean

```
rt <- t( t( temp[,-(1:4)]) - mt)
# separate values for each year by 0.1C
rtS <- rt
for(cols in 1:150)
rtS[, cols]<- Separate(unlist(rtS[,cols]),0.1)</pre>
colors <- rainbow(10) [rank(rtS[,1])]</pre>
tiff("pictures/Differences from mean.tif", w=1200,
h=1200)
Grid( c(1750, seq(1850, 2000, 50), 2008), seq(-8, 4, 2),
ylab="Residuals from Yearly Mean, 1859-2008/Degrees
C'', at=c(1850,1745), cex=c(2.5,2))
for( i in 1:10)
points(year, rt[i, ], col=colors[i], pch=16)
text(pos=4, 1760, Separate(rt[,1], 0.4), temp$loc,
cex=2.5, col=colors, xpd=T)
text( 1850, 0, "mean", cex=2)
lines(year, rep(0,length(year)), lwd=4)
dev.off()
```

Degrees C



The residuals are obtained by subtracting the mean over stations for each year from each station's value for that year. The residuals for each station vary randomly about a constant value, indicating similar trend behaviour for each station over time.

7 Residuals from additive model

Fit the additive model in which a temperature for a particular year and station is represented as a sum of terms consisting of a year value, a station value, and a residual:

```
tiff("pictures/Residuals adjusting for year and
stations.tif", w=1200, h=1200)
ms <- apply(rt, 1, mean)</pre>
rt <- rt - ms
rtS <-rt
for(cols in 1:150)
rtS[, cols]<- Separate(unlist(rt[,cols]),0.05)</pre>
colors <- rainbow(10)[rank(rtS[,1])]</pre>
Grid( c(1750, seq(1850, 2000, 50), 2008), seq(-1,1,0.5),
ylab="Residuals from Additive model/ Degrees C",
at=c(1900, 1745), cex=1.9)
for( i in 1:10)
points(year, rt[i, ], col=colors[i], pch=16)
text(pos=4, 1760, Separate(rt[,1], 0.05), temp$loc,
cex=2.5, col=colors, xpd=T)
dev.off()
```

Residuals from Additive model Degrees C Wien-Hohe Warte 0.5 Salzburg-Flughafen Innsbruck-Universität Graz - Universität agenfurt-Flughafen 0 Kremsmünster illacher Alpe Andrä im Lavanttal Linz-Stadt **Obergurgel-Vent** -0.5 1950 2000 1850 1900

```
cat("residual sd", round(sd(as.vector(rt)),2),
"\n")
```

```
residual sd 0.24
```

The residuals from the additive model are respectably small, sd = .24. There are mean temperature differences between the different stations, but they have a similar generally increasing pattern over time.

8 Parobolic fit to mean temperature

We keep track of the various model fits with a version of the akaike information criterion: AIC2 = 2 number of parameters - log likelihood.

```
AIC2 <- function(s)
2*s$df[1]+length(s$res)*log(max(1-s$r.sq,.00001))/2</pre>
```

This criterion accepts an additional parameter if the likelihood ratio test rejects the null hypothesis omitting the parameter at the 5% significance level.

Subtract mean from square term to get near orthogonal effects in regression:

```
sqyear <- (year-mean(year))^2
linsq <- lm(mt ~ year + sqyear)
tiff("pictures/Fitting curves.tif", w=1200, h=800)
Grid(c(1840, seq(1850,2000, 50),2008),seq(4, 10,1),
ylab=" Austrian Yearly Temperatures/Parabolic
Fit/Degree C", at=c(1925, 1848, 1925),
cex=c(2.5, 1.8, 1.8))
points(year, mt, col="green", pch=16)
lines(year, mt)
lines(year, linsq$fit, col="blue", lwd=3)
dev.off()
```



print(round(summary(linsq)\$coef[-1, -(3:4)], 5))

Estimate Std. Error year 0.01007 0.00114 sqyear 0.00014 0.00003

```
cat("AIC2:", round(AIC2(summary(linsq)),1), "\n")
```

AIC2: -32.9

Significant linear and parobolic coefficients, but the fit isn't all that good. The standard errors are misleading because the neighbouring observations are correlated. Don't trust the projection into the future.

9 Parabola with lag

The lag model adds the previous year"s temperature to the set of predicting variables in the model.

```
mtlag1 <- c(mt[1], mt[ - length(mt)])
linsqlag <- lm(mt ~ year + sqyear + mtlag1)
tiff("pictures/lag added.tif", w=1200, h=800)
Grid(c(1840,seq(1850,2000,50),2008),seq(4,10,1),
ylab="LaggedParabolic Fit, Austrian Yearly
Temperatures/Degree C", at=c(1920, 1848),
cex=c(2.5, 2))
lines(year, mt)
points(year, mt, pch=16, col= "green")
lines(year, linsq$fit, col="lgreen")
lines(year, linsq$fit, col="red", lwd=3)
lines(year, linsqlag$fit, col="red", lwd=3)
```



```
print(round( summary(linsqlag)$coef[-1, -(3:4)],
5))
```

```
Estimate Std. Erroryear0.008850.00138sqyear0.000120.00003mtlag10.127590.08215
```

cat("AIC2:", round(AIC2(summary(linsqlag)),1), "")

```
AIC2: -32.1
```

We see that the lagged effect is insignificant, and we don't need to use it. The lag adds the little red wibble wobble to the parabolic fit.

10 Fit a change point

For each year as a possible change point fit a different slope before and after the year, and then select that year with the smallest AIC2.

```
options(digits=10)
aic <- rep(0,146)
for( change in 1861:2006) {
# Construct a slope variable for each segment.
y^2 <- pmax(year - change, 0)
y1 <- year - y2
aic[change-1860] <- AIC2(summary(lm(mt \sim y1 + y2))
))
}
tiff("pictures/aic for different change
points.tif", w=1200, h=800)
Grid(c(1859, seq(1860, 2000, 20), 2006), seq(-40, -20, 5),
ylab="Change point for linear fit/aic2", at=c(1900,
1859, cex=c(3, 2.5))
points(1861:2006, aic, pch=16)
# Draw 95% interval
low <- min(which(aic <=min(aic)+2)) + 1860
high <- \max(which(aic <= min(aic)+2)) + 1860
lines(c(low,high),rep(min(aic)+2,2),
    col="red",lwd=3)
text(1930, min(aic)+ 1, "95% confidence region",
   cex=2, col="red")
arrows(1980, min(aic) + 1, 1980, min(aic))
dev.off()
```



The change point is at: which (aic==min(aic)) + 1860

[1] 1980

The 95% confidence interval is: c(low, high)

[1] 1965 1988

11 Fitted change point lines

```
y2 <- pmax(year - which(aic==min(aic)) -1860, 0)
y1 <- year - y2
reg <- lm( mt ~ y1 + y2 )
tiff("pictures/Fit Change Point Lines.tif", w=1200,
h=800)
Grid( c(1859, seq(1860, 2000, 20), 2008), seq(4,
10, 1),
ylab=" Lines fitted with a changepoint at
1980/Degree C", cex=c(3,2), at=c(1920, 1858))
points(year, mt , pch=16, col="green")
lines( year, col = "red", lwd = 2, reg$fit)
lines( year, col = "black", lwd = 1, mt)
dev.off()
```



This is the famous "hockey stick" pattern for climate change (1999, Mann, Bradley and Hughes)

```
print(round(summary(reg)coef[-1, -(3:4)], 4))
```

```
Estimate Std. Error
y1 0.0056 0.0014
y2 0.0532 0.0085
```

```
cat("AIC2:", round(AIC2(summary(reg)), 1), "")
```

AIC2: -35.1

We see an increase in temperature of .005 degrees per year up till 1980, and then a rate of increase of .05 degrees per year after 1980. This rate is often reported as 5 degrees per century, but really we don't know what will happen in the next century. Certainly, up 5 degrees in the next century would leave the world cooked.

12 Change in the individual stations

```
Find the best change point, at each station:
Set up data frame to include change points and slopes:\f2
chp <- data.frame( matrix(0, nrow=10, ncol=5))</pre>
names(chp)<-
c("changept","slope1","se1","slope2", "se2")
rownames(chp) <- substr(temp$loc, 1, 15)</pre>
# run over different stations
for (st in 1:10) {
mt <- unlist(temp[st, -(1:4)])</pre>
aic <- rep(0, 146)
for ( change in 1861:2006) {
  y^2 <- pmax(year - change, 0)
  y1 <- year - y2
  aic[change-1860]<-AIC2(summary(lm(mt~y1+ y2)))</pre>
}
changept <- which (aic == min(aic)) + 1860
y2 <- pmax(year - changept, 0)</pre>
y1 <- year - y2
reg <- lm( mt ~ y1 + y2 )
chp[st, 1] <- changept
chp[st,2:5]<-
    unlist(t(summary(reg)coef[-1, -(3:4)]))
}
print(round(chp,3))
              changept slope1 se1 slope2
                                            se2
Graz - Universi
                   1980 0.006 0.001 0.066 0.009
                  1978 0.006 0.002 0.054 0.008
Innsbruck-Unive
Klagenfurt-Flug
                   1985 0.004 0.001 0.063 0.011
Kremsmünster
                   1980 0.005 0.002 0.048 0.010
Linz-Stadt
                  1980 0.007 0.002 0.059 0.009
               1978 0.007 0.002 0.042 0.008
Obergurgel-Vent
Salzburg-Flugha
                  1980 0.006 0.002 0.056 0.010
St. Andrä im La
                  1985 0.005 0.001 0.057 0.011
Villacher Alpe
                   1979 0.006 0.002 0.046 0.009
```

Good agreement, in change points and slopes, except for Vienna!

1888 -0.036 0.009 0.016 0.002

Wien-Hohe Warte

13 Why is Vienna different?

Plot the aic values for Vienna to see why the change point is so different:

```
tiff("pictures/What happened to Vienna.tif",
w=1200, h=600)
```

```
Grid(c(1859,seq(1860,2000,20),2006),seq(-30,-18,1),
ylab="Change point for Vienna/aic2", at=c(1900,
1859), cex=c(3, 2.5))
```

```
points(1861:2006, aic, pch=16, col="blue")
dev.off()
```



The aic values show two local minima, one about 1886 and the other about 1968 (or maybe 1980). this suggests that the vienna series has two change points.

14 Three segments for Vienna

Construct a slope variable for the each segment, constant on the other segments.

```
y2 <- pmax(year - 1886, 0)
y1 <- year - y2
y3 <- pmax(year - 1968, 0)
y23 <- y2 - y3
Regress with 2 segments:
reg2 < -lm(mt ~ y1 + y2)
print(round(summary(reg2)coef[-1, -(3:4)], 4))
 Estimate Std. Error
y1 -0.0407 0.0103
y2 0.0153 0.0016
cat("AIC2:", round(AIC2(summary(reg2)), 1), "")
AIC2: -29
Regress with 3 segments:
reg3 < -lm(mt ~ y1 + y23 + y3)
print(round(summary(reg3)coef[-1, -(3:4)], 4))
  Estimate Std. Error
y1 -0.0303
               0.0105
              0.0025
y23 0.0092
     0.0351 0.0063
y3
cat("AIC2:", round(AIC2(summary(reg3)), 1), "")
```

AIC2: -32.3

The 95% significance threshold for adding a parameter to the model is to have the aic2 increase by 2, which it does here. True, we are selecting change points corresponding to the two lowest modal values on the aic curve, which invalidates the usual tail probability calculations.

15 Three segment fit

```
Compare the two and three segment fits to Vienna:
tiff("pictures/ Two and Three Segments fitted.tif",
w=1200, h=700)
Grid( c(1859, seq(1860, 2000,20),2008),seq(6,12,1),
ylab="Fitting two and three segments to Wien/Degree
C", cex=c(3,2), at=c(1920, 1857))
```

```
points(year, temp[10,-(1:4)],pch=16,col="green")
lines(year, temp[10,-(1:4)])
lines(year,reg2$fit, lwd=3)
lines(year,reg3$fit, lwd=3, col = "red")
```

dev.off()



The estimated final slope is .35 which corresponds to .035 degrees centigrade per year. The three segment fit captures the rapidly declining values before 1880 without much affecting the estimate of slope of the recent rapid increase.

16 Three segments for mean temperatures

```
Compute the AIC2 for for each choice of two change points, store in
the matrix aic:
mt <- apply(temp[, -(1:4)], 2,mean)</pre>
aic <- data.frame(matrix(0, 147, 147))</pre>
names(aic) <- 1860:2006
row.names(aic) <- names(aic)</pre>
for ( c1 in 1860:2005) {
for (c2 in (c1+1):2006) {
  y2 <- pmax(year - c1, 0)
  y1 <- year - y2
  y3 <- pmax(year - c2, 0)
  y23 <- y2 - y3
  aic[c1-1859,c2-1859]<-
  AIC2(summary(lm(mt \sim y1 + y23 + y3)))
}
}
```

16.1 The two change points that give the minimum aic.

```
location <- which(aic==min(aic))</pre>
```

The aic matrix is converted into a vector by concatenating the columns.

The location of the minimum is in this concatenated vector. The first change point is the remainder after dividing the location by 147:

11 <- location %% 147 c1 <- 11 + 1859

The second change point is 1 plus the number of times 147 divides into the location:

12 <- 1+ location %/% 147
c2 <- 12 + 1859</pre>

To express the change points in years we add 1859. round (min (aic), 3)

```
[1] -36.854
c(c1, c2)
[1] 1879 1980
```

The change points are at 1879 and 1980 with the minimum AIC2 equal to -36.854.

16.2 Fitting the two change point model

```
y2 <- pmax(year - c1, 0)
y1 <- year - y2
y3 <- pmax(year - c2, 0)
y23 <- y2 - y3
reg <- lm(mt ~ y1 + y23 + y3)
print(round(summary(reg3)$coef[-1, -(3:4)], 4) )
Estimate Std. Error
```

	bernate bea.	DITOI		
y1	-0.0303	0.0105		
y23	0.0092	0.0025		
у3	0.0351	0.0063		

cat("AIC2:", round(AIC2(summary(reg)), 1), "")

AIC2: -36.9

17 Three segment fit plotted

```
tiff("pictures/Fit with three segments.tif",
w=1200, h=600)
```

```
Grid( c(1859, seq(1860,2000,20), 2008),seq(5,9, 1),
ylab="Three segment fit to mean Austrian
Temperatures/Degree C",cex=c(3,2),at=c(1932, 1857))
```

```
points(year, mt ,pch=16, col="green")
lines(year, col="red", lwd=3, lm(mt~y1+y23+y3)$fit)
lines(year, mt)
dev.off()
```



The three segment solution is a considerable improvement over the two segment solution. However the aic2 only decreases from -35.1 to -36.9, so the improvement is not significant at the 95% standard level. The three segment solution for the average curve suggests an increase of 3.5C per century if the slope of the last segment continues (it won't). The general story is of decreasing temperatures up to 1879, a small rate of increase up to 1980, a large rate of increase since then.

18 Extrapolate integrated brownian motion

Extrapolate the observed Austrian mean temperature series x using the integrated brownian motion model. This is a particular case of the Kalman filter. The underlying temperature series, say, the "true" mean temperatures for all of Austria, is assumed be an integrated brownian motion z. Assume:

(1) $x|z \sim N(z, v1)$

(2) diff(diff(z)) ~ N(0, v2).

Here z is the fitted value to x, and is expected to be smoother than x. The integrated Brownian motion is characterised by the second differences of z being independent normal.

An observed straight line x would be fitted exactly by z under this model, with the v2 chosen to be zero. If x consisted of continuous straight line segments z would also closely approximate x.

All the extrapolated values lie on the straight line through the last two fitted values, but the posterior variance of z given the observed x increases markedly for distant projections.

```
temp <- read.csv("data/temp.csv", header=T,
as.is=T)
data <- temp[, -(1:4)]
mt <- apply(data, 2, mean)</pre>
```

```
Use the function extrapolateIBM to look 20 years ahead:
f <- extrapolateIBM(mt, extra=20)
```

```
tiff("pictures/Extrapolate next twenty years.tif",
w=1200, h=580)
```

```
Grid( c(1859,seq(1860,2020,20),2028),seq(4,12,1),
ylab="Extrapolate Integrated Brownian Motion to
2028/Degree C", cex=c(3,2), at=c(1940, 1857)
)
```

```
lines(1859:2028, f$est)
lines(1859:2028, f$est + 2*f$se, col="red", lwd=2)
lines(1859:2028, f$est - 2*f$se, col="blue", lwd=2)
for(y in 5:12)
lines(1859:2028,rep(y,170),col="grey")
points(1859:2008, mt, pch=16)
dev.off()
```



I prefer this extrapolation to the 3 segment extrapolation, which relies heavily on the accuracy of the break points, and suggests that the slope of the last segment is the one to be sustained. After a small initial decrease, the smoothed curve increases over the last 100 years, and more rapidly in the last 30 years. The suggested future increase is 4C in the next 100 years, with a very wide error band about that prediction. Similar extrapolations occur for the individual stations.

19 Conclusion

Austria is warming up. The worldwide data suggests that the 40-50 degrees latitude regions, full of population, production, and pollution, are producing lots of carbon dioxide, and warming up faster than other latitudes.

20 Data preparation

We use data archived by HISTALP - HISTORICAL INSTRUMENTAL CLIMATOLOGICAL SURFACE TIME SERIES OF THE GREATER ALPINE REGION. The data sets are available at the website:

"http://www.zamg.ac.at/histalp/Statmod_AT_T01.html#dataset"

We select 10 Austrian temperature series dating from to 1859. They have been copied from the above website into the the data subdirectory of the present directory :

```
filelist <- list.files("data/")
print(filelist)</pre>
```

```
[1] "boundary.txt" "HISTALP-T01-GRA-hom.txt"
[3] "HISTALP-T01-INN-hom.txt" "HISTALP-T01-KLA-hom.txt"
[5] "HISTALP-T01-KRE-hom.txt" "HISTALP-T01-LIN-hom.txt"
[7] "HISTALP-T01-OBG-hom.txt" "HISTALP-T01-SAL-hom.txt"
[9] "HISTALP-T01-SAN-hom.txt" "HISTALP-T01-VIA-hom.txt"
[11] "HISTALP-T01-WIE-hom.txt" "temp.csv"
```

20.1 Read data and select desired years

We make a data frame 10 x 154 to contain the 10 stations, with 4 descriptive variables, and 150 yearly temperatures:

```
temp <- data.frame(matrix(0,10,154) )
names(temp) <-c("loc", "long", "lat", "alt",
1859:2008)
Run through files containing HISTALP:
nf <- 0
for ( ff in 1:length(filelist) ){
   if ( grepl("HISTALP", filelist[ff]) ){
      nf <- nf + 1
      tf <-scan(paste("data/", filelist[ff], sep=""),
           what="", sep="\n", quiet=T)
      if (nf==1) print(head(tf))</pre>
```

```
# Get station name, title, lat long altitude:
  temp[nf, 1] <- substr(tf[3], 1, 25)</pre>
  temp[nf,2] <- as.numeric(substr(tf[3], 36, 40))
  temp[nf,3] <- as.numeric(substr(tf[3], 42, 46))
  temp[nf,4] <- as.numeric(substr(tf[3], 48, 52))</pre>
# Select range of 150 years 1859-2008:
  year <- as.numeric(substr(tf, 1, 4))</pre>
  use <- !is.na(year) & year < 2009 & year >1858
  year <- year[use]</pre>
# Extract year data and divide by 10 to convert to
Centigrade:
  temp[nf,5:154]<-
     as.numeric(substr(tf[use],116,118))/10
}
}
[1] "homogenised mean temperature
2009-07-06 09:15"
[2] "
                                   Y
                       ABBR.
                             X
                                       Z"
[3] "Graz - Universität
                     AT
                         GRA 15.45 47.08 377"
        JAN
            FEB MAR
                    APR
[4] "
                         MAY
                             JUN
                                 JUL
                                     AUG
SEP OCT NOV DEC 3-5 6-8 9-11 12-2 4-9 10-3
1-12"
[5]
911
[6]
9"
```

The early years" temperatures, are missing(999999), as are the last months in 2009.

temp[,1:6]

	loc	long	lat	alt	1859	1860
1	Graz - Universität	15.45	47.08	377	9.0	7.7
2	Innsbruck-Universität	11.38	47.26	609	8.6	7.4
3	Klagenfurt-Flughafen	14.32	46.65	459	8.0	6.4
4	Kremsmünster	14.13	48.05	389	8.5	6.9
5	Linz-Stadt	14.29	48.30	263	8.6	7.2
6	Obergurgel-Vent	11.03	46.87	1938	1.1	-0.3
7	Salzburg-Flughafen	13.00	47.80	450	8.4	6.8
8	St. Andrä im Lavanttal	14.83	46.76	402	7.6	6.1
9	Villacher Alpe	13.67	46.60	2160	0.0	-1.6
10	Wien-Hohe Warte	16.36	48.25	209	10.2	8.9

The first line is for Graz, longitude 15.45, latitude 47.08, altitude 377 metres, and then 9.0C, 7.7C,... for temperatures in 1859, 1860.

The data prepartion was easy because the format was a "flat" format. Each line of the data sets, except the first four, carried the same information about a different year. So we just needed to pick out the years we wished to include in the study.

Write to file so you don't have to extract again:

write.csv(temp, file="data/temp.csv",row.names=F)

21 Functions

```
Separate <- function(xx, d) {</pre>
# finds y as close as possible (in ssq) to xx but
with differences between y always exceeding d.
x < - sort(xx)
lx <- length(x)
if(lx == 1 || d \le 0) return(x)
# initialise y, dd;divide by d to prevent rounding
error
y < -x/d
dd <- rep(1/2, lx)
# collapse neighbouring intervals to achieve d
for (it in 1:lx) {
 try <- 1
 for(iter in 1:lx) {
   if(try == length(y)) break
# amalgamate neighbours if contradict space rule
   if (y[try]+dd[try] \ge y[try+1] - dd[try+1])
     y[try]<-(y[try]*dd[try] +y[try+1]*dd[try+1])/</pre>
            (dd[try] + dd[try+1])
     dd[try] <- dd[try] + dd[try+1]
# remove bad neighbour and back up try
     y <- y[-try-1]
     dd <- dd[-try-1]
     if(try > 1) try <- try-1
   }
 else try <- try+1
 }
# stop once you have reached the end of y
 if(try == length(y)) break
}
# reconstruct yy every d apart within y blocks
yy < -rep(0,0)
```

```
for( i in 1:length(y))
yy <- c(yy, d*y[i]+d*seq(-dd[i]+1/2, dd[i]-1/2,1))
return(yy[rank(xx, ties.method="first")])
}
extrapolateIBM <- function(x, extra =0, iter=200){</pre>
# Extrapolate the time series x using integrated
brownian motion model.
# predictors for IBM ; begin with assumed small
variance for IBM
lx <- length(x)
mx \leftarrow matrix(0, nrow=2*lx-2, ncol = lx)
v1 <- mean(lm(x \sim seq(1:lx)))$res^2)
v2 < - v1/10
# predictor matrix
mx[1:lx, 1:lx] <- diag(rep(.0001, lx))
for(i in 1:(1x-2)) mx[i+1x, i + 0:2] <- c(1, -2, 1)
# Rotate x and z so that the rotated variables are
independent.
# svd holds for all v1 and v2
s <- svd(mx)
sd[(1x-1):1x] <- 0
d <- s$d
vx <- t(s$v) %*% x
# iterate using EM to estimate variances v1 and v2
# the new estimate of v1 is the expected value of
(vx-vz)^2,
# the new estimate of v2 is the expected value of
d^2 vz^2,
# when vz the posterior from vx~N(vz,v1), 0~N(vz*d)
, v2)
\# vx = norm(vx/(1 + d^2*v1/v2, 1/v1 + d^2/v2))
d2 <- 1/d[-c(lx-1, lx)]^2
```

```
vx2 <- vx[-c(lx-1, lx)]^2
oldlike <- 0
# begin iterations
for(it in 1:iter){
 vz <- diag(1/(1 + d^2 * v1/v2))  %*% vx
 vv <- 1/(1/v1 + d^2/v2)
 mv < -v1 + v2*d2
 loglike <-0.5*(sum(log(mv) + vx2/mv))
 if(it==1 |loglike > oldlike +.001) {
   oldlike <- loglike
#EM updates; accelerate a little
   acc <-0
   if(it>5) acc <- 0.75
   if(it>10) acc <- 1
 v2 <-((1+acc)*sqrt(sum( d^2*vx^2*(vv/v1)^2+d^2*vv
      )/(1x-2)) - acc*sqrt(v2))^2
   vv < -1/(1/v1 + d^2/v2)
   vz <- diag(1/(1 + d^2 * v1/v2))  %*% vx
   v1 <-((1+acc)*sqrt(mean(vx^{2}*(1-vv/v1)^{2}+vv)))
        )) - acc * sqrt(v1) )^2
 }
}
# two additional parameters for fitting a mean and
a variance
# the third term is minus log like for fitting a
mean and variance
aic2 <-4-oldlike + length(x) * (log(var(x)) + 1)/2
z <- s$v %*% diag( 1/(1 + d^2 * v1/v2) ) %*% vz
# do the regression with the given v1, v2 to get
estimates and compute se for the present and future
z"s
mx[1:lx, 1:lx] <- diag(1, lx)
w \le rep(1/v1, 2*(lx-1))
```

```
w[(lx+1) : (2*(lx-1))] <- 1/v2
xx <- c(x, rep(0, lx-2))
cov <- summary(lm(xx ~ -1+mx, weights=w))$cov
se <- sqrt(diag(cov))</pre>
```

```
# extrapolate with se
k <- 1:extra
zz <- z[lx] + k * (z[lx] - z[lx-1])
see<-sqrt((k+1)^2*cov[lx,lx]-2*k*(k+1)*cov[lx,lx-1]
+ (k+1)^2*cov[lx-1, lx-1])</pre>
```

```
return(list(est=c(z,zz),se=c(se,see),aic2 = aic2 ))
}
```