ADA Data Reanalysis Assignment

Alana Pengilley

31/03/2021

Paper outline

The paper is an addition to the growing body of research focused on the relationship between ancient societies and rapid climatic events, specifically the 4.2 kya event (a period of increased aridity and cooling). Lawrence et al. (2021) aim to take an empirical approach at a large spatial scale to investigate trends in population and settlement organization related to the impact of the 4.2 kya event across the entirety of the Northern Fertile Crescent from 6,000 to 3,000 cal. BP.

The paper employs three different datasets (archaeological data and radiocarbon dates) from the period between 6000 and 3000 cal. BP to investigate population trends. There are two different archaeological settlement datasets used in this study. The first consists of data collected from 16 published archaeological surveys and includes 1157 sites divided into 2783 occupation phases. The second consists of urban settlement data (defined as sites which reach a size of ten hectors or more) and includes 132 urban sites divided into 283 occupation phases. The third dataset is comprised of 963 radiocarbon dates from archaeological contexts collected over a slightly broader time range, 6500 to 2500 uncal. BP, in the aim to avoid edge effects.

The archaeological settlement data (both sets) were binned into a series of 100-year time slices starting at 6,000 BP and ending at 3,000 BP. The authors applied aoristic approaches to deal with uncertainties in the dataset. Essentially, this method assumes that the total probability of an archaeological event within a given time span is 1, which indicates absolute certainty. This is then divided by the length of the sites chronological range to represent the probability of existence for each temporal block (100 years) and an aoristic weight is produced (displayed in Fig. 4 and 5). In addition, Monte Carlo methods were applied to generate randomised start of occupation periods that could be compared with other demographic proxies.

The radiocarbon dataset was processed using the R package rcarbon. The dates were calibrated and a summed probability distribution (SPD) of individual dates was produced. A KDE map of the spatial distributions of radiocarbon dates was created. they compared the summed normalised probability distributions and observed unnormalized SPDs of calibrated radiocarbon dates against a theoretical null model. They fitted a logistic growth model to the observed SPD and produced a 95% confidence envelope of 1,000 SPDs to test if the observed pattern differed from the null models (displayed in Fig. 2).

The conclusions of this study indicate support for claims of a decline in population and social complexity in relation to the 4.2 kya event. There are obvious negative trends in all three datasets between 4,300 and 3,900 cal. BP and sharp drops in the SPD and rural settlement proxies.

Specific data and analysis that will be replicated

This report will replicate almost all parts of the analyses conducted in the original paper on all three assemblages. However, it will re-frame from replicating the spatial data maps.

The archaeological datasets:

Both archaeological datasets were analysed using the same methodology. The settlement data was loaded in and the dates associated with sites were converted from BC/AD to BP for analysis and site duration was also calculated and added to the dataset. The paper split the data into two groups, above or below 300mm annual average rainfall, and so this was replicated in the analyses. Bins of 100 year time slices were created. Aoristic weights of all sites were calculated and placed into a new data frame, this was replicated with sites above 300mm or below 300mm of rainfall. The data was sumarised in order for it to be plotted, this included extracting the site counts and aoristic weights for both rainfall groups and extracting the dates of value (between 3000 - 6000 BP) and converting them into vectors that could be plotted. - Fig 4 and 5 were reproduced.

The radiocarbon dataset:

The radiocarbon data will be processed using the 'rcarbon' R package outline in the paper (and above). The goal of this will be to replicate the calibration and summed probability distributions (SPD) in order to reproduce Figure 2b, c, and d. The dataset will be divided into two groups. The first will be all dates and the second will be the dates produced from short-lived organic samples (e.g. bones, collagen, seeds, etc). These groups will be calibrated using the 'calibrate()' function and, as discussed in the paper, they will be un-normalized to avoid artificial peaks produced by normalization. SPDs of all individual dates within a time range of 6,500 to 2,500 BP are produced next. Bins of 50 years were created in order to reduced potential biases associated with SPDs. Distributions of both groups (all dates and short-lived) were produced, as well as normalized and un-normalized SPDs for comparison. Additionally, Pearsons coefficients were also calculated and correlated to 0.96 as per the paper.

Archaeological Settlement data

```
sites <- read.csv("settlements.csv", sep = ",", header = T) #load in site data
head(sites) #check first lines
```

```
##   Name     Id StartDate EndDate Longitude Latitude Rain_mm
## 1       BON_1      2500    2100  36.85311 34.75535 318.291
## 2       BON_1      1900    1600  36.85311 34.75535 318.291
## 3       BON_1      1600    1200  36.85311 34.75535 318.291
## 4       BON_1      1000     700  36.85311 34.75535 318.291
## 5      BON_11      2500    2100  36.84555 34.77783 318.291
## 6      BON_11      1900    1600  36.84555 34.77783 318.291
##                   Source
## 1 Morandi Bonacossi 2007
## 2 Morandi Bonacossi 2007
## 3 Morandi Bonacossi 2007
## 4 Morandi Bonacossi 2007
## 5 Morandi Bonacossi 2007
## 6 Morandi Bonacossi 2007
```

```
nrow(sites) #2783 occupation phases (corresponds with paper)
```

```
## [1] 2783
```

```
#dates are in BC/AD and need to be converted to BP for analysis
StartdatesBP <- (sites$StartDate) + 1950 #create new variable
EnddatesBP <- (sites$EndDate) + 1950

sites[, "StartdatesBP"] <- StartdatesBP #add variables to df
sites[, "EnddatesBP"] <- EnddatesBP

#duration of sites
Duration <- sites$StartdatesBP - sites$EnddatesBP
sites[, "Duration"] <- Duration #add to df
head(sites)
```

```
##   Name     Id StartDate EndDate Longitude Latitude Rain_mm
## 1       BON_1      2500    2100  36.85311 34.75535 318.291
```

```
## 2       BON_1      1900    1600  36.85311 34.75535 318.291
## 3       BON_1      1600    1200  36.85311 34.75535 318.291
## 4       BON_1      1000     700  36.85311 34.75535 318.291
## 5      BON_11      2500    2100  36.84555 34.77783 318.291
## 6      BON_11      1900    1600  36.84555 34.77783 318.291
##                  Source StartdatesBP EnddatesBP Duration
## 1 Morandi Bonacossi 2007         4450       4050      400
## 2 Morandi Bonacossi 2007         3850       3550      300
## 3 Morandi Bonacossi 2007         3550       3150      400
## 4 Morandi Bonacossi 2007         2950       2650      300
## 5 Morandi Bonacossi 2007         4450       4050      400
## 6 Morandi Bonacossi 2007         3850       3550      300
```

```
#the paper splits the date into two groups, above or below 300mm rainfall avg.
above300mm <- subset(sites, sites$Rain_mm > 300) #variable will be used later
below300mm <- subset(sites, sites$Rain_mm <= 300)


#basic descriptive stats of the rainfall groups
summary(above300mm$Rain_mm) #rainfall stats
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   302.9   431.8   464.8   488.1   548.9   755.5
```

```
summary(below300mm$Rain_mm)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -9999.0   250.9   258.5   186.8   286.3   299.1
```

```
summary(above300mm$Duration) #duration of site occupation in both groups
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   100.0   300.0   500.0   697.1   800.0  3800.0
```

```
summary(below300mm$Duration)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   100.0   400.0   500.0   542.8   650.0  3800.0
```

The settlement data is binned into a series of 100-year time slices and a new column is created to put the binned results in.

```
binwidth <- 100
breaks <- seq(1900, 8800, by = binwidth) #include range of dates
mids <- seq(min(breaks) + (binwidth/2), max(breaks) - (binwidth/2), by=binwidth)
newcolumns <- paste(breaks[1:(length(breaks)-1)], "start", sep = "")
```

An aoristic method is applied to the dataset. First a new dataframe needs to be created for the results, this includes a new column for site counts and one for the aoristic weights. Then a loop is used to create aoristic weights. This process assumes that the probability of site occupation occurring at each site is 1, then divide this by the length of the sites occupation to produce a 'weight' which is then fed into the df created.

This process is then repeated for sites above and below 300mm of annual rainfall.

```
#aoristic weights for all sites
#create df for results
aoristicweights <- sitecounts <- sites
sitecounts[newcolumns] <- NA
sitecounts<-sitecounts[,newcolumns]
```

```r
aoristicweights[newcolumns]<-NA
aoristicweights<-aoristicweights[,newcolumns]
aoristicweights<- sitecounts

for (i in 1:nrow(sites)){
  cat(paste(i,"; ",sep=""))
  sitestart <- sites$StartdatesBP[i]
  siteend <- sites$EnddatesBP[i]
  siteyears <- seq(siteend, sitestart, by = 1)
  siteyearshist <- hist(siteyears, breaks=breaks, plot = F)
  timeweights <- siteyearshist$counts/binwidth
  sitecounts[i,] <- timeweights #counts
  aoristicweights[i,] <- timeweights/sum(timeweights) #aoristic weights
}
```

```
## 1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19; 20; 21; 22; 23; 24; 25; 26; 27; 28
```

```r
#aoristic weights for all sites with rainfall above 300mm
aoristicweights_above300mm <- sitecounts_above300mm <- above300mm
sitecounts_above300mm[newcolumns] <- NA
sitecounts_above300mm<-sitecounts_above300mm[,newcolumns]
aoristicweights_above300mm[newcolumns]<-NA
aoristicweights_above300mm<-aoristicweights_above300mm[,newcolumns]
aoristicweights_above300mm<- sitecounts_above300mm

for (i in 1:nrow(above300mm)){
  cat(paste(i,"; ",sep=""))
  sitestart <- above300mm$StartdatesBP[i]
  siteend <- above300mm$EnddatesBP[i]
  siteyears <- seq(siteend,sitestart,by=1)
  siteyearshist <- hist(siteyears, breaks=breaks, plot=FALSE)
  timeweights <- siteyearshist$counts/binwidth
  sitecounts_above300mm[i,] <- timeweights #counts
  aoristicweights_above300mm[i,] <- timeweights/sum(timeweights) #aoristic weights
}
```

```
## 1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19; 20; 21; 22; 23; 24; 25; 26; 27; 28
```

```r
#aoristic weights for all sites with rainfall below 300mm
aoristicweights_below300mm <- sitecounts_below300mm <- below300mm
sitecounts_below300mm[newcolumns] <- NA
sitecounts_below300mm<-sitecounts_below300mm[,newcolumns]
aoristicweights_below300mm[newcolumns]<-NA
aoristicweights_below300mm<-aoristicweights_below300mm[,newcolumns]
aoristicweights_below300mm<- sitecounts_below300mm

for (i in 1:nrow(below300mm)){
  cat(paste(i,"; ",sep=""))
  sitestart <- below300mm$StartdatesBP[i]
  siteend <- below300mm$EnddatesBP[i]
  siteyears <- seq(siteend,sitestart,by=1)
  siteyearshist <- hist(siteyears, breaks=breaks, plot=FALSE)
  timeweights <- siteyearshist$counts/binwidth
  sitecounts_below300mm[i,] <- timeweights #counts
  aoristicweights_below300mm[i,] <- timeweights/sum(timeweights) #aoristic weights
```

```
}
```

```
## 1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19; 20; 21; 22; 23; 24; 25; 26; 27; 28
```

Combine and summarize the data into a format that can be graphed, colSum() will change the data.frame to numeric variables - this is done for both rainfall groups and all data.

```
#summaries of groups
sum_sitecounts_above300mm <- colSums(sitecounts_above300mm)
sum_aoristicweights_above300mm <- colSums(aoristicweights_above300mm)

sum_sitecounts_below300mm <- colSums(sitecounts_below300mm)
sum_aoristicweights_below300mm <- colSums(aoristicweights_below300mm)

sum_sitecounts<- colSums(sitecounts)
sum_aoristicweights <- colSums(aoristicweights)
```

Then need to extract the dates of use and convert into a format that can be compared.

```
#above 300mm (counts and weights)
sum_above300mm <- sum_sitecounts_above300mm[12:41] #extract values between 3000-6000BP, count
x <- sum_above300mm
sum_above300mm_norm<-((x-min(x))/(max(x) - min(x)))

sum_above300mm_aoristicwts <- sum_aoristicweights_above300mm[12:41] #weights
x <- sum_above300mm_aoristicwts
sum_above300mm_aoristicwts_norm<-((x-min(x))/(max(x) - min(x)))

#below 300mm(counts and weights)
sum_below300mm <- sum_sitecounts_below300mm[12:41] #counts
x <- sum_below300mm
sum_below300mm_norm<-((x-min(x))/(max(x) - min(x)))

sum_below300mm_aoristicwts <- sum_aoristicweights_below300mm[12:41] #weights
x <- sum_below300mm_aoristicwts
sum_below300mm_aoristicwts_norm<-((x-min(x))/(max(x) - min(x)))
```

Figure 4a replication

```
#figure 4a (Raw count)
plot(mids[12:41],sum_above300mm_norm, lty = "solid", col = "white", cex.axis = 0.5, xlab = "cal BP", yla
lines(mids[12:41],sum_above300mm_norm, lty="solid", col="green")
lines(mids[12:41],sum_below300mm_norm, lty="solid", col="red")
rect(4300, -1, 3900, 1.5, col = rgb(128,128,128, alpha = 120, maxColorValue = 255), border = NA)
text(x = 4100, 0.90, labels = "4.2 k", font = 2, col = "black")
abline(v=seq(5800, 3200, -200), lty = 3, col= "grey86")
text(x = 5860, y = 0.6, labels = "Raw count", font = 2, cex = 0.5)
text(x = 5880, y = 0.55, label = paste("site-phase=", nrow(above300mm)," ", "sites=", length(unique(abo
text(x = 5880, y = 0.50, label = paste("site-phase=", nrow(below300mm)," ", "sites=", length(unique(bel
legend(x=5960,0.47, legend=c("Above 300mm","Sbelow 300mm"), lty=c("solid","solid"), lwd=c(0.5,0.5), col
```

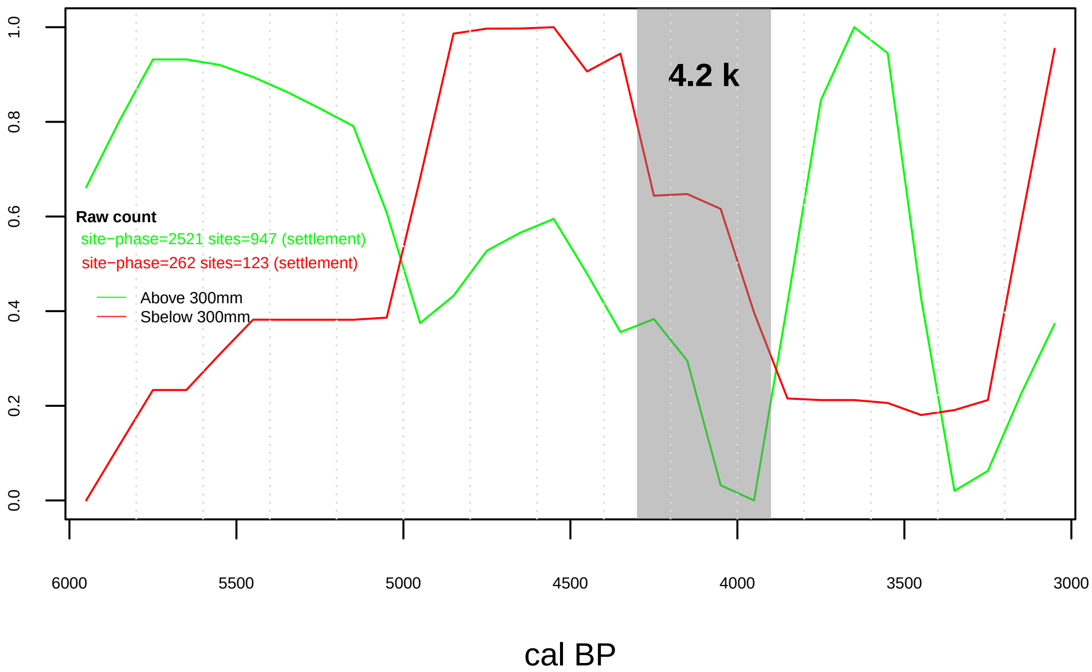


Figure 4b replication

```
#figure 4b (Aoristic wts)
plot(mids[12:41], sum_above300mm_aoristicwts_norm, lty = "solid", col = "white", cex.axis = 0.5, xlab =
lines(mids[12:41], sum_above300mm_aoristicwts_norm, lty = "dashed", col = "green")
lines(mids[12:41], sum_below300mm_aoristicwts_norm, lty = "dashed", col = "red")
rect(4300, -1, 3900, 1.5, col = rgb(128,128,128, alpha = 120, maxColorValue = 255), border = NA)
text(x = 4100, 0.90, labels = "4.2 k", font = 2, col = "black")
abline(v=seq(5800, 3200, -200), lty = 3, col= "grey86")
text(x = 5800, y = 0.6, labels = "Aoristic Weights", font = 2, cex = 0.5)
text(x = 5880, y = 0.55, label = paste("site-phase=", nrow(above300mm)," ", "sites=", length(unique(abo
text(x = 5880, y = 0.50, label = paste("site-phase=", nrow(below300mm)," ", "sites=", length(unique(bel
legend(x=5960,0.47, legend=c("Above 300mm","Below 300mm"), lty=c("solid","solid"), lwd=c(0.5,0.5), col=
```

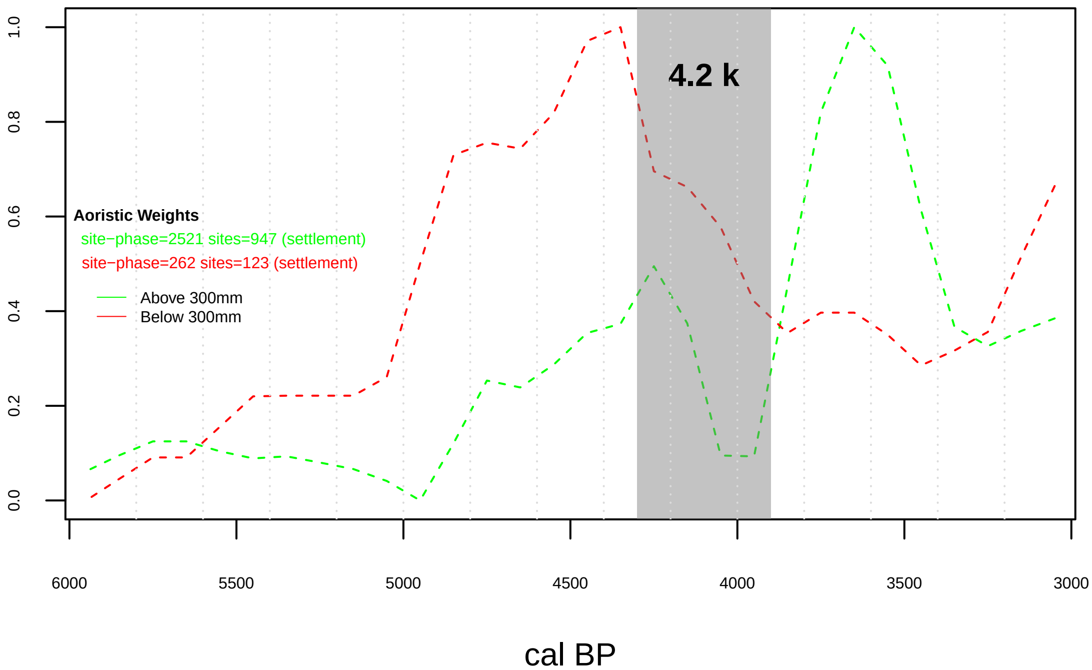
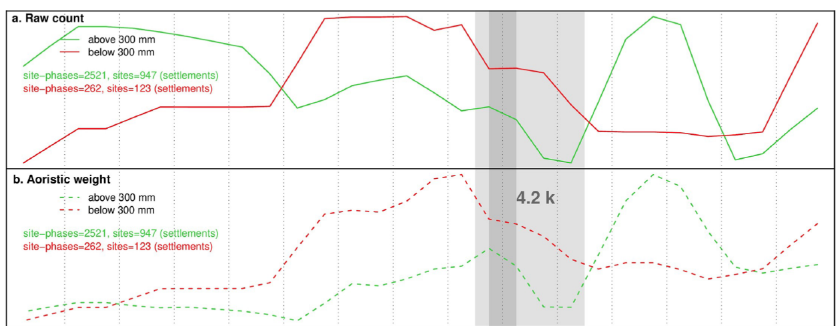


Figure 4 from paper (for comparison)



Archaeological settlement data (Urban)

This dataset will be processed the same way as the other settlement dataset.

```
urban_sites <- read.csv("urban_sites.csv", sep = ",", header = T)
head(urban_sites)
```

```
##                    Name  Id Rainfall StartDate EndDate   SizeHa Longitude
## 1             Abu Harba 195  186.243      2500    1600 10.60000  37.38201
## 2              Abu Kula 190  437.290      2550    1500 10.00000  42.32779
## 3              Abu Kula 190  437.290      3100    2550  8.00000  42.32779
## 4                Agilah 221  255.632      1900    1650 31.87901  39.63867
## 5              Ain Dara  68  453.873      1600    1000 22.00000  36.85222
## 6 Al-Hirbat al-Garbiyya 193  185.758      1800    1600 22.00000  37.35318
##   Latitude
## 1 35.44246
## 2 36.66782
## 3 36.66782
## 4 36.59434
## 5 36.46018
```

```
## 6 35.25884
##
## 1
## 2 Wilkinson. T. J., Tucker. D. J., (1995) Settlement and Development in the North Jazira, Iraq: A stu
## 3 Wilkinson. T. J., Tucker. D. J., (1995) Settlement and Development in the North Jazira, Iraq: A stu
## 4
## 5                                                                                             Stone. E
## 6
```

```
nrow(urban_sites) #283 occupation phases (corresponds with paper)
```

```
## [1] 283
```

```
#converted from BC/AD to BP
StartdatesBP_urban <- (urban_sites$StartDate) + 1950
EnddatesBP_urban <- (urban_sites$EndDate) +1950

urban_sites[, "StartdatesBP"] <- StartdatesBP_urban
urban_sites[, "EnddatesBP"] <- EnddatesBP_urban

#duration
Duration_urban <- urban_sites$StartdatesBP - urban_sites$EnddatesBP
urban_sites[, "Duration"] <- Duration_urban
head(urban_sites)
```

```
##                   Name  Id Rainfall StartDate EndDate   SizeHa Longitude
## 1            Abu Harba 195  186.243      2500    1600 10.60000  37.38201
## 2             Abu Kula 190  437.290      2550    1500 10.00000  42.32779
## 3             Abu Kula 190  437.290      3100    2550  8.00000  42.32779
## 4               Agilah 221  255.632      1900    1650 31.87901  39.63867
## 5             Ain Dara  68  453.873      1600    1000 22.00000  36.85222
## 6 Al-Hirbat al-Garbiyya 193  185.758      1800    1600 22.00000  37.35318
##   Latitude
## 1 35.44246
## 2 36.66782
## 3 36.66782
## 4 36.59434
## 5 36.46018
## 6 35.25884
##
## 1
## 2 Wilkinson. T. J., Tucker. D. J., (1995) Settlement and Development in the North Jazira, Iraq: A stu
## 3 Wilkinson. T. J., Tucker. D. J., (1995) Settlement and Development in the North Jazira, Iraq: A stu
## 4
## 5                                                                                             Stone. E
## 6
##   StartdatesBP EnddatesBP Duration
## 1         4450       3550      900
## 2         4500       3450     1050
## 3         5050       4500      550
## 4         3850       3600      250
## 5         3550       2950      600
## 6         3750       3550      200
```

```
#split into rainfall groups
urban_above300mm <- subset(urban_sites, urban_sites$Rainfall > 300)
```

```r
urban_below300mm <- subset(urban_sites, urban_sites$Rainfall <= 300)

#basic descriptive stats of the rainfall groups
summary(urban_above300mm$Rainfall) #rainfall stats
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   304.2   350.4   422.9   425.5   474.4   735.9
```

```r
summary(urban_below300mm$Rainfall)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   120.7   228.4   253.2   242.6   267.4   299.1
```

```r
summary(urban_above300mm$Duration) #duration of site occupation in both groups
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    50.0   200.0   500.0   660.6   900.0  3000.0
```

```r
summary(urban_below300mm$Duration)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      50     350     550     730    1000    3000
```

```r
#bin into 100-year time slots
#reused same variables as above (binwidth, breaks, mids)
newcolumns_urban <- paste(breaks[1:(length(breaks)-1)], "start", sep = "")


#aoristic weights for urban sites (all)
#create df for results
aoristicweights_urban <- sitecounts_urban <- urban_sites
sitecounts_urban[newcolumns_urban] <- NA
sitecounts_urban<-sitecounts_urban[,newcolumns_urban]
aoristicweights_urban[newcolumns_urban]<-NA
aoristicweights_urban<-aoristicweights_urban[,newcolumns_urban]
aoristicweights_urban<- sitecounts_urban

for (i in 1:nrow(urban_sites)){
  cat(paste(i,"; ",sep=""))
  sitestart <- urban_sites$StartdatesBP[i]
  siteend <- urban_sites$EnddatesBP[i]
  siteyears <- seq(siteend, sitestart, by = 1)
  siteyearshist <- hist(siteyears, breaks=breaks, plot = F)
  timeweights <- siteyearshist$counts/binwidth
  sitecounts_urban[i,] <- timeweights #counts
  aoristicweights_urban[i,] <- timeweights/sum(timeweights) #aoristic weights
}
```

```
## 1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19; 20; 21; 22; 23; 24; 25; 26; 27; 2
```

```r
#aoristic weights for all sites with rainfall above 300mm
aoristicweights_urban_above300mm <- sitecounts_urban_above300mm <- urban_above300mm
sitecounts_urban_above300mm[newcolumns_urban ] <- NA
sitecounts_urban_above300mm <-sitecounts_urban_above300mm[,newcolumns_urban ]
aoristicweights_urban_above300mm[newcolumns_urban]<-NA
aoristicweights_urban_above300mm<-aoristicweights_urban_above300mm[,newcolumns_urban ]
aoristicweights_urban_above300mm<- sitecounts_urban_above300mm
```

```r
for (i in 1:nrow(urban_above300mm)){
  cat(paste(i,"; ",sep=""))
  sitestart <- urban_above300mm$StartdatesBP[i]
  siteend <- urban_above300mm$EnddatesBP[i]
  siteyears <- seq(siteend,sitestart,by=1)
  siteyearshist <- hist(siteyears, breaks=breaks, plot =F)
  timeweights <- siteyearshist$counts/binwidth
  sitecounts_urban_above300mm[i,] <- timeweights #counts
  aoristicweights_urban_above300mm[i,] <- timeweights/sum(timeweights) #aoristic weights
}
```

```
## 1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19; 20; 21; 22; 23; 24; 25; 26; 27; 28
```

```r
#aoristic weights for all sites with rainfall below 300mm

aoristicweights_urban_below300mm <- sitecounts_urban_below300mm <- urban_below300mm
sitecounts_urban_below300mm[newcolumns_urban ] <- NA
sitecounts_urban_below300mm <-sitecounts_urban_below300mm[,newcolumns_urban ]
aoristicweights_urban_below300mm[newcolumns_urban]<-NA
aoristicweights_urban_below300mm<-aoristicweights_urban_below300mm[,newcolumns_urban ]
aoristicweights_urban_below300mm<- sitecounts_urban_below300mm

for (i in 1:nrow(urban_below300mm)){
  cat(paste(i,"; ",sep=""))
  sitestart <- urban_below300mm$StartdatesBP[i]
  siteend <- urban_below300mm$EnddatesBP[i]
  siteyears <- seq(siteend,sitestart,by=1)
  siteyearshist <- hist(siteyears, breaks=breaks, plot =F)
  timeweights <- siteyearshist$counts/binwidth
  sitecounts_urban_below300mm[i,] <- timeweights #counts
  aoristicweights_urban_below300mm[i,] <- timeweights/sum(timeweights) #aoristic weights
}
```

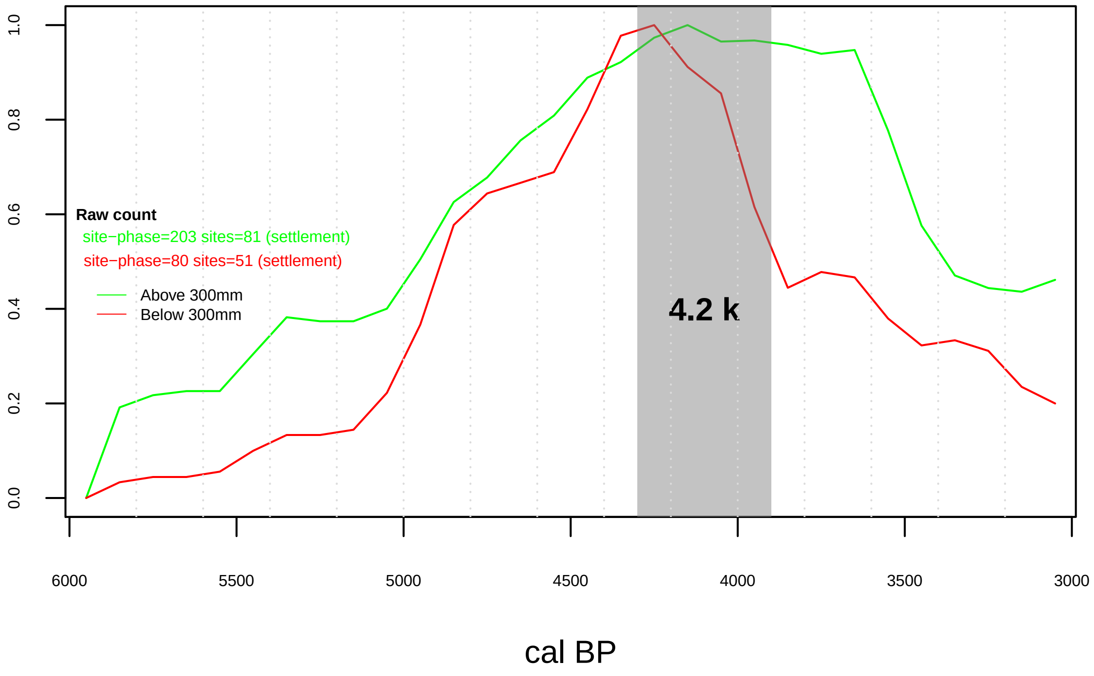
```
## 1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19; 20; 21; 22; 23; 24; 25; 26; 27; 28
```

```r
#summaries of rainfall groups

sum_urbancounts <- colSums(sitecounts_urban)
sum_urbanaoristicweights <- colSums(aoristicweights_urban)

sum_urbancounts_above300mm <- colSums(sitecounts_urban_above300mm)
sum_urbanaoristicweights_above300mm <- colSums(aoristicweights_urban_above300mm)

sum_urbancounts_below300mm <- colSums(sitecounts_urban_below300mm)
sum_aoristicweights_below300mm <- colSums(aoristicweights_urban_below300mm)


#all urban sites (counts and weights converted)
urbancounts <- sum_urbancounts[12:41]
x <- urbancounts
urbancounts_norm <- ((x-min(x))/(max(x) - min(x)))

urbanwieghts <- sum_urbanaoristicweights[12:41]
x <- urbanwieghts
urbanwieghts_norm <- ((x-min(x))/(max(x) - min(x)))
```

```
#above 300mm (counts and weights converted)
urbansum_above300mm <- sum_urbancounts_above300mm[12:41] #extract values between 3000-6000BP
x <- urbansum_above300mm
urbansum_above300mm_norm<-((x-min(x))/(max(x) - min(x)))

urbansum_above300mm_aoristicwts <- sum_urbanaoristicweights_above300mm[12:41] #weights
x <- urbansum_above300mm_aoristicwts
urbansum_above300mm_aoristicwts_norm<-((x-min(x))/(max(x) - min(x)))

#below 300mm(counts and weights converted)
urbansum_below300mm <- sum_urbancounts_below300mm[12:41] #counts
x <- urbansum_below300mm
urbansum_below300mm_norm<-((x-min(x))/(max(x) - min(x)))

urbansum_below300mm_aoristicwts <-sum_aoristicweights_below300mm[12:41] #weights
x <- urbansum_below300mm_aoristicwts
urbansum_below300mm_aoristicwts_norm<-((x-min(x))/(max(x) - min(x)))
```

##Replication of Figure 5a

```
plot(mids[12:41],urbansum_above300mm_norm, lty = "solid", col = "white", cex.axis = 0.5, xlab = "cal BP
lines(mids[12:41],urbansum_above300mm_norm, lty="solid", col="green")
lines(mids[12:41],urbansum_below300mm_norm, lty="solid", col="red")
rect(4300, -1, 3900, 1.5, col = rgb(128,128,128, alpha = 120, maxColorValue = 255), border = NA)
text(x = 4100, 0.4, labels = "4.2 k", font = 2, col = "black")
abline(v=seq(5800, 3200, -200), lty = 3, col= "grey86")
text(x = 5860, y = 0.6, labels = "Raw count", font = 2, cex = 0.5)
text(x = 5880, y = 0.55, label = paste("site-phase=", nrow(urban_above300mm)," ", "sites=", length(uniq
text(x = 5880, y = 0.50, label = paste("site-phase=", nrow(urban_below300mm)," ", "sites=", length(uniq
legend(x=5960,0.47, legend=c("Above 300mm","Below 300mm"), lty=c("solid","solid"), lwd=c(0.5,0.5), col=
```



Replication of Figure 5b

```
#figure 5c (Aoristic wts)
plot(mids[12:41], urbansum_above300mm_aoristicwts_norm, lty = "solid", col = "white", cex.axis = 0.5, xl
lines(mids[12:41], urbansum_above300mm_aoristicwts_norm, lty = "dashed", col = "green")
lines(mids[12:41], urbansum_below300mm_aoristicwts_norm, lty = "dashed", col = "red")
rect(4300, -1, 3900, 1.5, col = rgb(128,128,128, alpha = 120, maxColorValue = 255), border = NA)
text(x = 4100, 0.90, labels = "4.2 k", font = 2, col = "black")
abline(v=seq(5800, 3200, -200), lty = 3, col= "grey86")
text(x = 5800, y = 0.6, labels = "Aoristic Weights", font = 2, cex = 0.5)
text(x = 5880, y = 0.55, label = paste("site-phase=", nrow(urban_above300mm)," ", "sites=", length(uniq
text(x = 5880, y = 0.50, label = paste("site-phase=", nrow(urban_below300mm)," ", "sites=", length(uniq
legend(x=5960,0.47, legend=c("Above 300mm","Below 300mm"), lty=c("solid","solid"), lwd=c(0.5,0.5), col=c
```

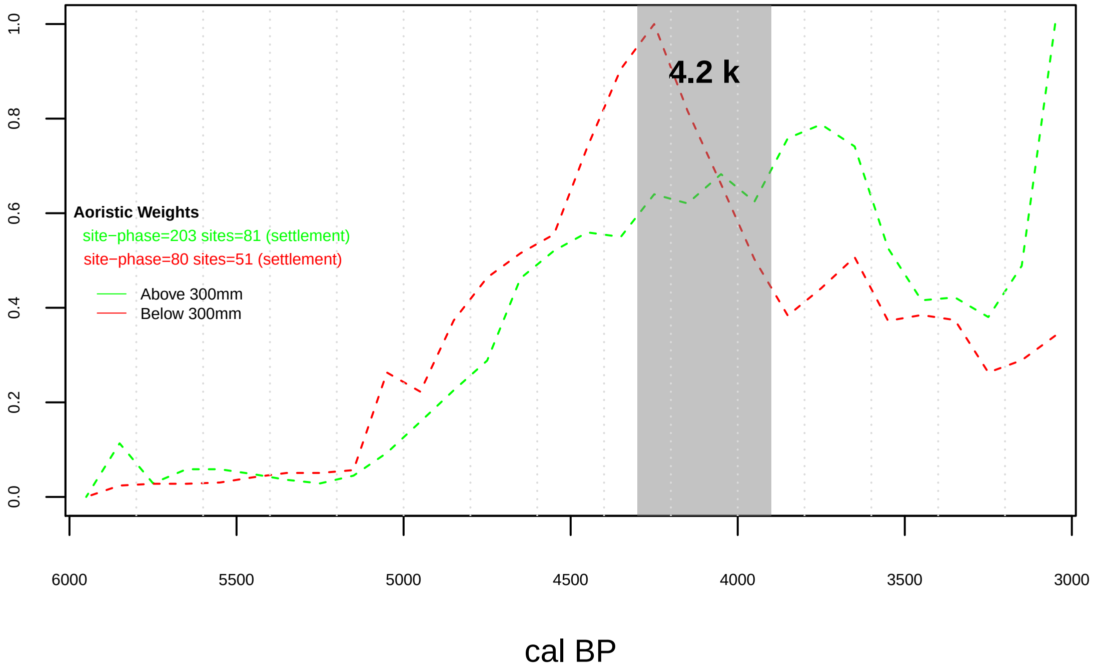


Figure 5 from paper (for comparison)

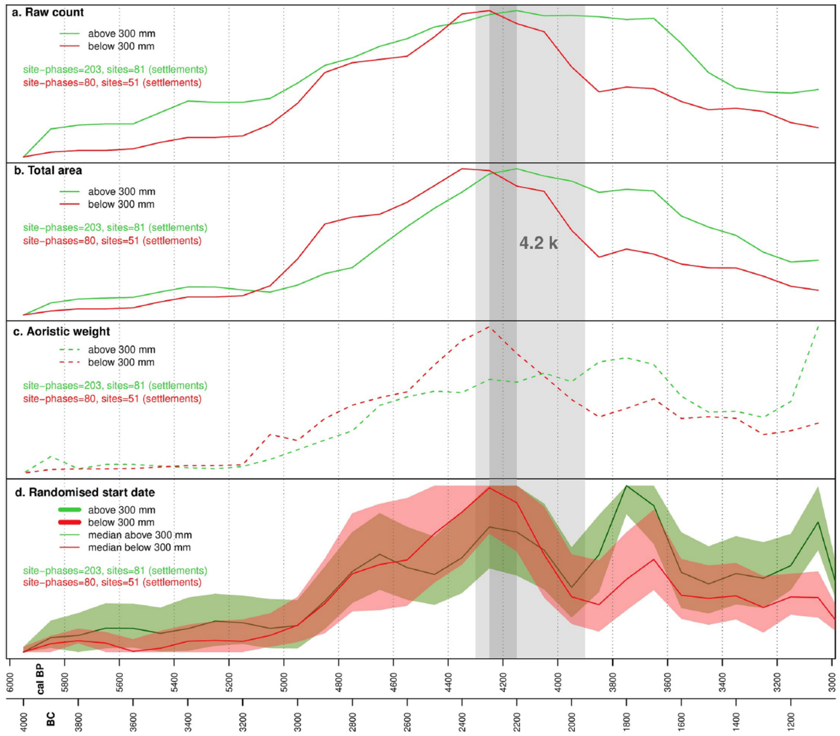


Fig 5. Urban settlements dataset trends. a: Raw site counts. b: Total area (summed). c: Aoristic weight (of counts). d: Randomized start dates (of counts).

Radiocarbon data analysis

First we load in the radiocarbon dataset and subset all values associated with short-lived dates as described in the paper.

```
library(rcarbon)
library(parallel)

#Load in radiocarbon data set and view first few rows
dates <- read.csv("dates.csv", sep = ",", header = T)
head(dates)
```

```
##   DateID     LabID OthLabID Problems  CRA Error DC13         Material
## 1   6771    OxA-882                  6100   120    0 grain (charred)
## 2   5075 Ly-12506                    4070    40    0         charcoal
## 3   5076 Ly-12507                    4090    35    0             seed
## 4   5077 Ly-12508                    3990    40    0             seed
## 5   5078 Ly-12509                    3990    40    0             seed
## 6   5079 Ly-12510                    4020    40    0             seed
##                Species SiteID    SiteName
## 1 Triticum boeoticum    1060 Abu Hureyra
## 2                       6606    Al Rawda
## 3                       6606    Al Rawda
## 4                       6606    Al Rawda
## 5                       6606    Al Rawda
## 6                       6606    Al Rawda
##                                                                    SiteContext
## 1 Level 324 -basal pit-; Trench E, phase 1, basal pit; Trench E, level 468, pit
```

```
## 2                                                           LOCUS 1084 - SECT. 1
## 3                                                           LOCUS 1115 - SECT. 1
## 4                                                         LOCUS 2036 - SECT. 2 A
## 5                                                         LOCUS 2049 - SECT. 2 A
## 6                                                         LOCUS 2063 - SECT. 2 A
##     SiteType Country Longitude Latitude    UTM_X   UTM_Y LocQual
## 1 settlement      SY   38.3933  35.8683 445228.5 3969511       B
## 2 settlement      SY   37.6330  35.1810 375530.3 3893971       A
## 3 settlement      SY   37.6330  35.1810 375530.3 3893971       A
## 4 settlement      SY   37.6330  35.1810 375530.3 3893971       A
## 5 settlement      SY   37.6330  35.1810 375530.3 3893971       A
## 6 settlement      SY   37.6330  35.1810 375530.3 3893971       A
##                                      Source Comments
## 1 Moore et al. 1986 ;CalPal; Flohr et al. 2016
## 2                                  BANADORA
## 3                                  BANADORA
## 4                                  BANADORA
## 5                                  BANADORA
## 6                                  BANADORA
```

```
#subset short lived dates (defined as a separate group in text)
shortlived_dates <- dates[grep(paste(c("antler","bone","coprolite","seed","fruit","carpal","tooth", "gr
```

Create parameters to work with for the analyses, including creating a available for bins of 50 years and creating a broader time span to avoid edge effects (as discussed in the paper).

```
detectCores()#detect number of cores in usage
```

```
## [1] 8
```

```
#Parameters for analyses
bin <- 50
runm <- 50 #for smoothing (number stipulated in text)
ncores <- 8
realstartBP <- 6000 #original time span as stipulated in paper
realendBP <- 3000
workingstartBP <- realstartBP + 500 #broader time span to avoid edge effect
workingendBP <- realendBP - 500
```

Then need to calibrate the radiocarbon dates using the calibrate() function from the 'rcarbon' package. The dates were calibrated as unnormalised as discussed previously. Although normalization (or not) does not have an impact on the shape of each individual dates calibrated probability distribution, it does influence the shape of SPDs. Additionally, a 'IntCal20' calibration curves was used as direct in the paper.

```
#Calibrate dates (un-normalised)
cal_dates <- calibrate(x = dates$CRA, errors = dates$Error, calCurves = "intcal20", normalised = F, nco
```

```
## [1] "Calibrating radiocarbon ages..."
## [1] "Running in parallel (standard calibration only) on 1 workers..."
##   |                                                                              |
```

```
cal_shortlived_dates <- calibrate(x = shortlived_dates$CRA, errors = shortlived_dates$Error, calCurves =
```

```
## [1] "Calibrating radiocarbon ages..."
## [1] "Running in parallel (standard calibration only) on 8 workers..."
##   |                                                                              |
```

```r
#summary of calibrated dates for both groups
summary(cal_dates[1:10])
```

```
##    DateID MedianBP OneSigma_BP_1 OneSigma_BP_2 OneSigma_BP_3 OneSigma_BP_4
## 1       1     6973  7157 to 7094  7082 to 6849  6813 to 6802      NA to NA
## 2       2     4563  4786 to 4766  4616 to 4589  4587 to 4517  4477 to 4446
## 3       3     4598  4793 to 4762  4689 to 4682  4622 to 4523  4457 to 4457
## 4       4     4469  4518 to 4468  4447 to 4417      NA to NA      NA to NA
## 5       5     4469  4518 to 4468  4447 to 4417      NA to NA      NA to NA
## 6       6     4484  4522 to 4458  4456 to 4436  4432 to 4424      NA to NA
## 7       7     4165  4235 to 4195  4187 to 4142  4129 to 4092      NA to NA
## 8       8     4285  4400 to 4370  4355 to 4325  4300 to 4234  4197 to 4185
## 9       9     4285  4400 to 4370  4355 to 4325  4300 to 4234  4197 to 4185
## 10     10     4369  4422 to 4348  4333 to 4295      NA to NA      NA to NA
##    OneSigma_BP_5 TwoSigma_BP_1 TwoSigma_BP_2 TwoSigma_BP_3 TwoSigma_BP_4
## 1       NA to NA  7256 to 6724  6708 to 6676      NA to NA      NA to NA
## 2       NA to NA  4803 to 4757  4698 to 4673  4649 to 4423      NA to NA
## 3       NA to NA  4813 to 4753  4713 to 4666  4654 to 4515  4481 to 4444
## 4       NA to NA  4571 to 4542  4534 to 4388  4372 to 4353  4328 to 4299
## 5       NA to NA  4571 to 4542  4534 to 4388  4372 to 4353  4328 to 4299
## 6       NA to NA  4782 to 4768  4614 to 4595  4583 to 4411      NA to NA
## 7       NA to NA  4347 to 4335  4293 to 4076  4041 to 3990      NA to NA
## 8   4165 to 4161  4409 to 4223  4206 to 4155      NA to NA      NA to NA
## 9   4165 to 4161  4409 to 4223  4206 to 4155      NA to NA      NA to NA
## 10      NA to NA  4514 to 4481  4444 to 4284  4276 to 4247      NA to NA
```

```r
summary(cal_shortlived_dates[1:10])
```

```
##    DateID MedianBP OneSigma_BP_1 OneSigma_BP_2 OneSigma_BP_3 OneSigma_BP_4
## 1       1     6973  7157 to 7094  7082 to 6849  6813 to 6802      NA to NA
## 2       2     4598  4793 to 4762  4689 to 4682  4622 to 4523  4457 to 4457
## 3       3     4469  4518 to 4468  4447 to 4417      NA to NA      NA to NA
## 4       4     4469  4518 to 4468  4447 to 4417      NA to NA      NA to NA
## 5       5     4484  4522 to 4458  4456 to 4436  4432 to 4424      NA to NA
## 6       6     4165  4235 to 4195  4187 to 4142  4129 to 4092      NA to NA
## 7       7     5454  5580 to 5503  5489 to 5444  5406 to 5326      NA to NA
## 8       8     6445  6537 to 6516  6502 to 6310      NA to NA      NA to NA
## 9       9     7352  7426 to 7272      NA to NA      NA to NA      NA to NA
## 10     10     7378  7504 to 7267      NA to NA      NA to NA      NA to NA
##    TwoSigma_BP_1 TwoSigma_BP_2 TwoSigma_BP_3 TwoSigma_BP_4
## 1   7256 to 6724  6708 to 6676      NA to NA      NA to NA
## 2   4813 to 4753  4713 to 4666  4654 to 4515  4481 to 4444
## 3   4571 to 4542  4534 to 4388  4372 to 4353  4328 to 4299
## 4   4571 to 4542  4534 to 4388  4372 to 4353  4328 to 4299
## 5   4782 to 4768  4614 to 4595  4583 to 4411      NA to NA
## 6   4347 to 4335  4293 to 4076  4041 to 3990      NA to NA
## 7   5658 to 5277  5172 to 5132  5104 to 5060      NA to NA
## 8   6667 to 6280      NA to NA      NA to NA      NA to NA
## 9   7557 to 7544  7508 to 7237  7221 to 7166      NA to NA
## 10  7612 to 7157  7106 to 7075  7035 to 7035      NA to NA
```

Summed probability distributions were calculated next. The function spd() aggregates calibrated radiocarbon dates within a defined chronological range (2,500 - 6,500 BP). To mitigate biases that can results from strong inter-site variability in sample size, artificial bins are implemented using the binPrep() function.

Normalised and non-normalised SPDs for both groups of dates were created in order to compare in Fig. 2c.

```
#Summed Probability Distributions
#SPD as normalised and non-normalised for comparison
bins <- binPrep(sites = dates$SiteID, age = dates$CRA, h = bin) #create bins
spdnorm <- spd(x = cal_dates, bins = bins, timeRange = c(workingstartBP, workingendBP), datenormalised =
```

```
## [1] "Extracting and aggregating..."
##   |                                                                              |
```

```
spd <- spd(x = cal_dates, bins = bins, timeRange = c(workingstartBP, workingendBP), datenormalised = F,
```

```
## [1] "Extracting and aggregating..."
##   |                                                                              |
```

```
#short-lived dates SPD
bins_shortlived <- binPrep(sites = shortlived_dates$SiteID, age = shortlived_dates$CRA, h = bin)
spdnorm_shortlived <- spd(x = cal_shortlived_dates, bins = bins_shortlived, timeRange = c(workingstartB
```

```
## [1] "Extracting and aggregating..."
##   |                                                                              |
```

```
spd_shortlived <- spd(x = cal_shortlived_dates, bins = bins_shortlived, timeRange = c(workingstartBP, wo
```

```
## [1] "Extracting and aggregating..."
##   |                                                                              |
```

Pearsons coefficient is then calculated. The Pearsons coefficient matches that described in the paper, r = 0.96 for both date groups.

```
#Calculate pearsons coefficient, both correlated to 0.96 as per the paper
#Pearsons coefficient for SPD of all dates (normalised and not)
(cor.test(spd$grid$PrDens, spdnorm$grid$PrDens, method = "pearson"))
```

```
## 
##  Pearson's product-moment correlation
## 
## data:  spd$grid$PrDens and spdnorm$grid$PrDens
## t = 247.17, df = 3999, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9668321 0.9706431
## sample estimates:
##       cor
## 0.9687948
```

```
#Pearsons coefficient for SPD of short-lived dates (normalised and not)
(cor.test(spd_shortlived$grid$PrDens, spdnorm_shortlived$grid$PrDens, method = "pearson"))
```

```
## 
##  Pearson's product-moment correlation
## 
## data:  spd_shortlived$grid$PrDens and spdnorm_shortlived$grid$PrDens
## t = 235.17, df = 3999, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
```

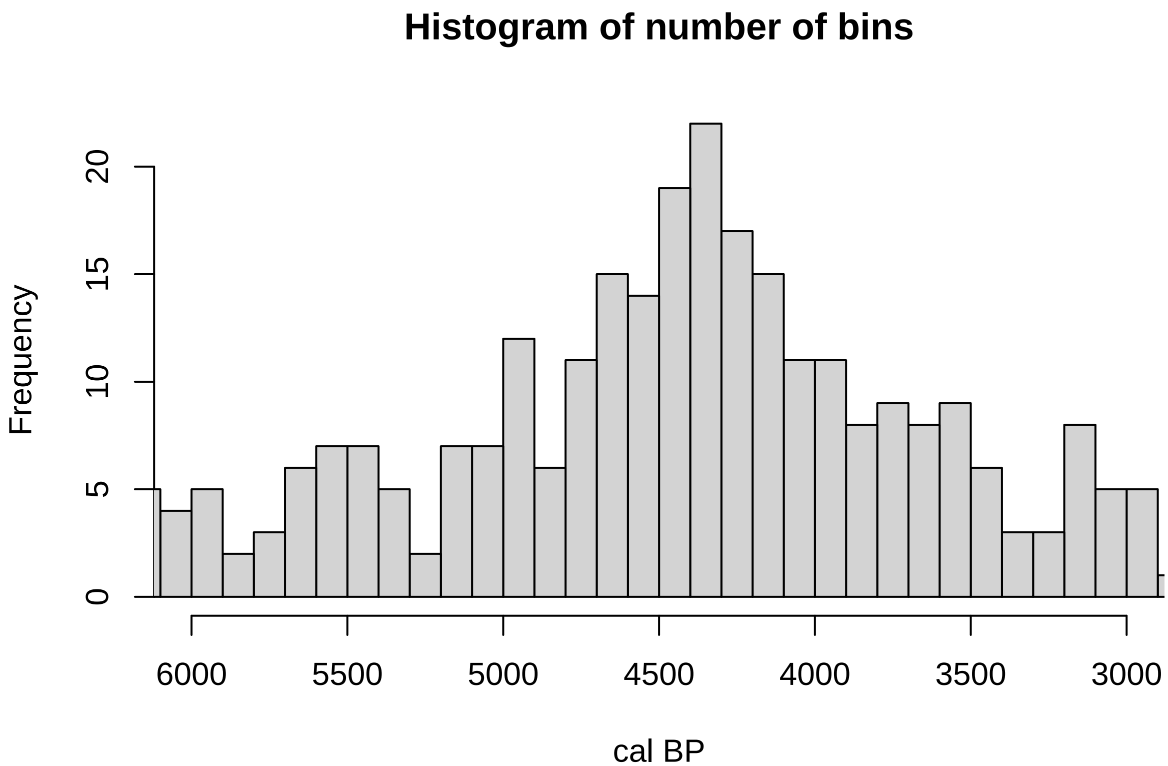
```
## 95 percent confidence interval:
##  0.9635404 0.9677234
## sample estimates:
##       cor
## 0.9656945
```

Replication of Figure 2b

```r
#Visualizing Bins (figure 2b)
dates_bins_med <- binMed(x = cal_dates, bins = bins)
```

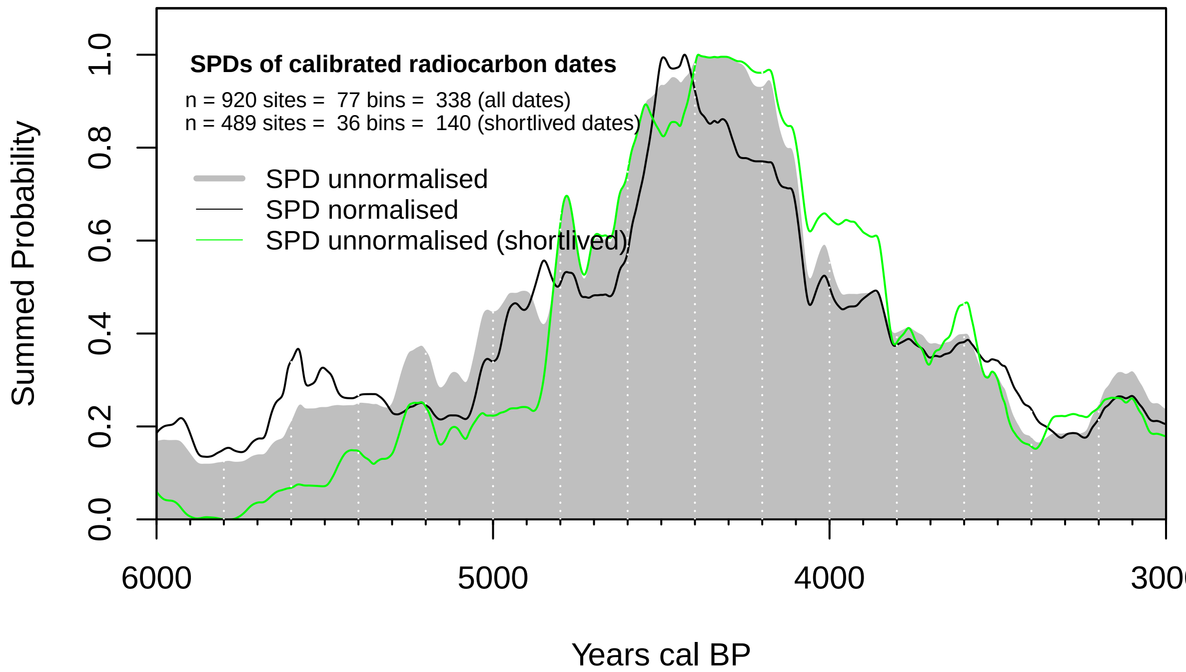
```
## [1] "Extracting and aggregating..."
##   |                                                                               |
```

```r
hist(dates_bins_med, 50, main="Histogram of number of bins", xlab = "cal BP", xlim = c(6000, 3000))
```



Replication of Figure 2c

```r
#plot unnormalised and normalised SPD (Figure 2c)
plot(spd, xlim=c(6000,3000),rescale = T) #spd data
plot(spdnorm, xlim=c(6000,3000), xaxt='n', yaxt='n', type='simple', col="black", rescale = T, add = T)#
plot(spd_shortlived, xlim=c(6000,3000), xaxt='n', yaxt='n', type='simple', col="green", rescale = T, ad
abline(v=seq(5800, 3200, -200), lty = 3, col= "white") #mark every 200 years with dotted line
text(x = 5900, 1, labels="SPDs of calibrated radiocarbon dates", font=2, cex=0.75, adj=c(0,1)) #subtitl
text(x = 5800, 0.90, labels = paste("n =", nrow(dates), "sites = ", length(unique(dates$SiteID)), "bins
text(x = 5780, 0.85, labels = paste("n =", nrow(shortlived_dates), "sites = ", length(unique(shortlived
legend(x=5950,0.80, legend=c("SPD unnormalised","SPD normalised","SPD unnormalised (shortlived)"), lty=
```



A logitsic growth model is then fitted to the observed SPD and a 95% confidence envelope of 1,000 SPDs is randomly generated. This will statistically test if the observed pattern differs from the null models. As discussed in the paper, deviations outside of the 95% confidence limits indicate periods of population growth or decline that are greater than expected according to a logistic model of population growth.

The text did not stipulate which function to use to get a logistic model. I ran the the genetic logistic model test using the function modelTest() but the results did not compare to the paper. I then tried a custom logistic model and was successful.

The p-value is the same as that detailed in the paper (<0.01).

```
#testing against theoretical growth model
#Generate Logistic model
set.seed(1)
nsim <- 1000 #from text

#logistic_model <- modelTest(cal_dates, errors = dates$Error, bins = bins, nsim = nsim, timeRange = c(6
#tried this but didn't match the papers results so did a custom Log mod

#Custom Logistic model
spd.smoothed <- spd(cal_dates, timeRange = c(6000, 3000), bins = bins, runm = runm) #generate a smoothe
```

```
## [1] "Extracting and aggregating..."
##   |                                                                         |
```

```
logFit <- nls(PrDens~SSlogis(calBP, Asym, xmid, scale), data = spd.smoothed$grid, control = nls.control
#generate a df containing the fitted values
logFitDens=data.frame(calBP=spd.smoothed$grid$calBP,PrDens=SSlogis(input=spd.smoothed$grid$calBP,Asym=c
# Use the modelTest function
Log_model <- modelTest(cal_dates, errors = dates$Error, bins = bins, nsim = nsim, timeRange = c(6000,300
```

```
## [1] "Aggregating observed dates..."
## [1] "Monte-Carlo test..."
## [1] "Running in parallel on 8 workers..."
```

```
## |                                                                               |
```

```
# Retrieve p-values
p <- Log_model$pval
```

Replication of Figure 2d

```
#plot log model (figure 2d)
plot(Log_model, xlim=c(6000,3000))
lines(Log_model$fit$calBP, Log_model$fit$PrDens, col = "black", lty = 2) #regression line
abline(v=seq(5800, 3200, -200), lty = 3, col= "white") #mark every 200 years with dotted line
abline(v=c(4300, 3900), col = c("red", "red"), lwd = c(2, 2)) #4.2k marker
text(x = 4100, 0.26, labels = "4.2 k", font = 2, col = "red")
text(x = 5900, 0.27, labels="Logistic Fit", font=2, cex=0.75, adj=c(0,1)) #subtitle text
legend(x=5950,0.26, legend=c("SPD (dates not normalised)", "Logistic Model", "95% MC envelope", "positi
text(x = 5700, 0.16, cex = 0.6, labels = substitute(paste("p value = ", x), list(x = round(Log_model$pva
```

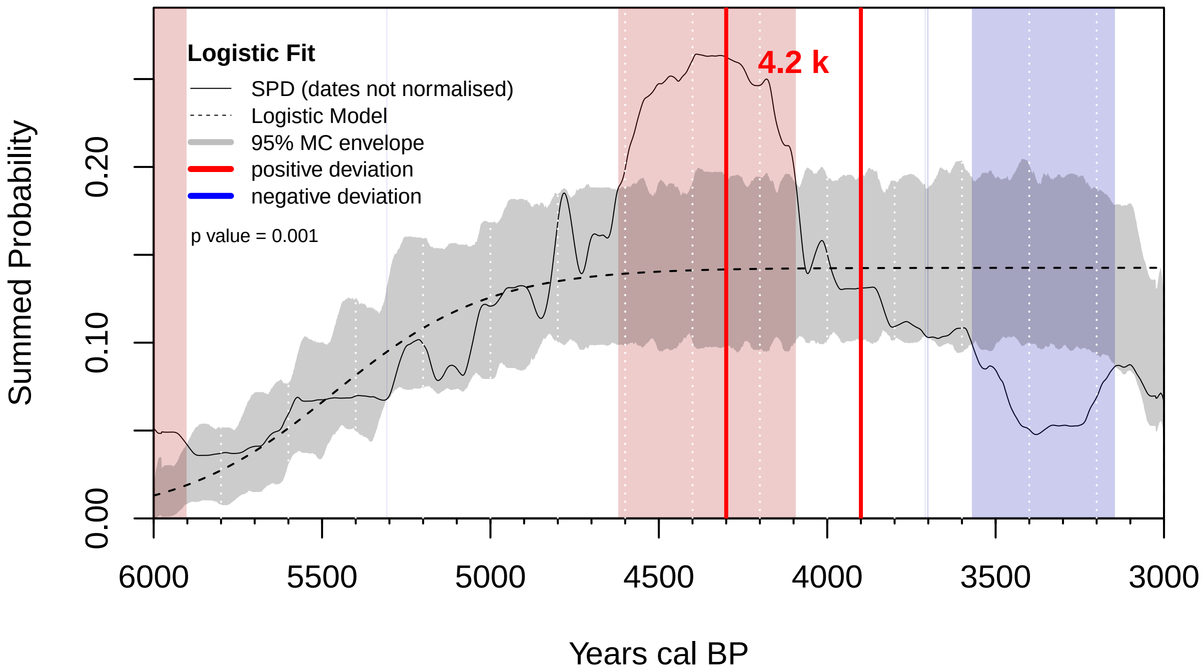


Figure 2 from paper (for comparison)

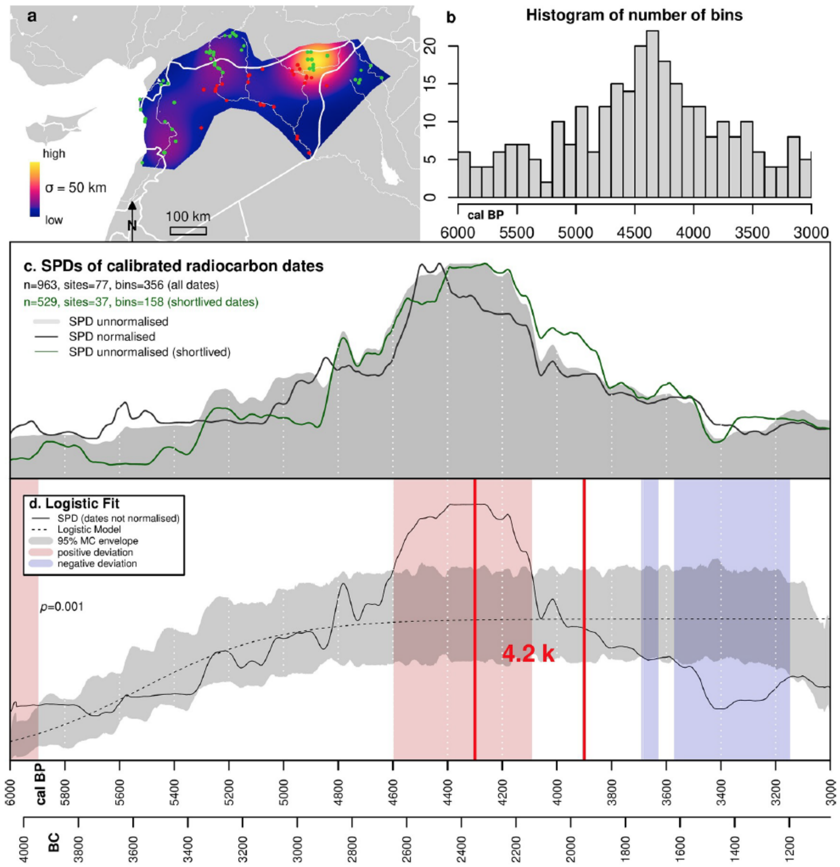


Fig 2. Summed probability distribution dataset. a: Kernel Density Estimate (50km bandwidth) of radiocarbon dates used. b. Histogram of bins and numbers of dates per bin. c: Summed Probability Distributions of calibrated radiocarbon dates. d: Summed Probability Distribution of dates superimposed on Logistic population model. Country outlines in map (a) from the GADM website (https://gadm.org/).

https://doi.org/10.1371/journal.pone.0244871.g002

Pearsons coefficient values

Table 3 comparison (all dates from 6000 - 3000k)

The replicated values are almost identical to those produced in the paper. There is a slight difference in the pearsons value between SPD when compared to the count and aoristic, although this number is only .01 of a position off in both cases and can still be interpreted in the same light as the paper.

```
x <- c(spd$grid[3501:502,2]) #group values (late to recent)
SPD <- colSums(matrix(x, nrow = 100, ncol = 30)) #create matrix
count_1 <- sum_sitecounts[12:41] #extract dates of value
aoristic_1 <- sum_aoristicweights[12:41] #extract dates of value
proxies_1 <- data.frame(cbind(count_1, aoristic_1, SPD)) #combine variables into df
(table_3_replication <- cor(proxies_1, method = "pearson")) #peasons correlation between variables
```

```
##                 count_1 aoristic_1        SPD
## count_1      1.00000000 0.03816917 -0.1705716
## aoristic_1   0.03816917 1.00000000  0.2636534
## SPD         -0.17057162 0.26365336  1.0000000
```

Table 3 from paper (for comparison)

Table 3. Pearson correlation coefficient r-value matrix between the archaeo-demographic proxies (all sites) from 6,000 to 3,000 cal BP.

	Count	Aoristic weight	Random	SPD
Count	1.00			
Aoristic weight	0.04	1.00		
Random	0.14	**0.86**	1.00	
SPD	-0.18	0.25	0.16	1.00

Significant correlations indicated by bold numbers (p-value < 0.05).

https://doi.org/10.1371/journal.pone.0244871.t003

Table 4 comparison (all dates from 6000 - 4000k)

Again, the replicated values are almost identical to those produced in the paper. There is a slight difference in the pearsons value between SPD when compared to the count and aoritsic, although this difference is slight and the signfiaince of these values compare to those presented in the paper.

```
x <- c(spd$grid[2501:502,2])
SPD <- colSums(matrix(x, nrow = 100, ncol = 20))
count_2 <- sum_sitecounts[22:41]
aoristic_2 <- sum_aoristicweights[22:41]
proxies_2 <- data.frame(cbind(count_2, aoristic_2, SPD))
(table_4_replication <- cor(proxies_2, method = "pearson"))
```

```
##               count_2 aoristic_2        SPD
## count_2     1.0000000 -0.3894488 -0.5911249
## aoristic_2 -0.3894488  1.0000000  0.9176673
## SPD        -0.5911249  0.9176673  1.0000000
```

```
#check that p value for one of the bold numbers in table 4, i.e correlation between SPD and aoristic.
#the p-value is <0.05 and is significant
(cor.test(SPD, aoristic_2, method = "pearson"))
```

```
##
##  Pearson's product-moment correlation
##
## data:  SPD and aoristic_2
## t = 9.7983, df = 18, p-value = 1.221e-08
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8000270 0.9673571
## sample estimates:
##       cor
## 0.9176673
```

Table 4 from paper (for comparison)

Table 4. Pearson correlation coefficient r-value matrix between the archaeo-demographic proxies (all sites) from 6,000 to 4,000 cal BP.

	Count	Aoristic weight	Random	SPD
Count	1.00			
Aoristic weight	-0.39	1.00		
Random	-0.26	**0.87**	1.00	
SPD	-0.61	**0.91**	**0.70**	1.00

Significant correlations indicated by bold numbers (p-value < 0.05).

https://doi.org/10.1371/journal.pone.0244871.t004

Table 5 comparison (urban dates from 6,000-3,000k)

The replicated values are almost identical to those produced in the paper. There is are very minute difference in the pearsons value between SPD when compared to the count and aoritsic, however this correlation can be interpreted in the same way as the paper.

```
x <- c(spd$grid[3501:502,2])
SPD <- colSums(matrix(x, nrow = 100, ncol = 30))
count_3 <- sum_urbancounts[12:41]
aoritsic_3 <- sum_urbanaoristicweights[12:41]
proxies_3 <- data.frame(cbind(count_3, aoritsic_3, SPD))
(Table_5_replication<- cor(proxies_3, method = "pearson"))
```

```
##              count_3 aoritsic_3       SPD
## count_3    1.0000000  0.8770329 0.8555891
## aoritsic_3 0.8770329  1.0000000 0.6669123
## SPD        0.8555891  0.6669123 1.0000000
```

```
#check that p value for one of the bold numbers in table 5, i.e correlation between SPD and count.
#the p-value is <0.05 and is significant
(cor.test(SPD, count_3, method = "pearson"))
```

```
##
##  Pearson's product-moment correlation
##
## data:  SPD and count_3
## t = 8.7459, df = 28, p-value = 1.699e-09
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7160309 0.9293830
## sample estimates:
##       cor
## 0.8555891
```

Table 5 from paper (for comparison)

Table 5. Pearson correlation coefficient r-value matrix between the archaeo-demographic proxies (only urban sites) from 6,000 to 3,000 cal BP.

	Count	Area	Aoristic weight	Random	SPD
Count	1.00				
Area	**0.97**	1.00			
Aoristic weight	**0.88**	**0.91**	1.00		
Random	**0.88**	**0.89**	**0.96**	1.00	
SPD	**0.85**	**0.78**	**0.66**	**0.66**	1.00

Significant correlations indicated by bold numbers (p-value < 0.05).

https://doi.org/10.1371/journal.pone.0244871.t005

Table 6 comparison (urban dates from 6,000-4,000k)

The replicated values are identical to the those presented in the paper.

```
x <- c(spd$grid[2501:502,2])
SPD <- colSums(matrix(x, nrow = 100, ncol = 20))
count_4 <- sum_urbancounts[22:41]
aoritsic_4 <- sum_urbanaoristicweights[22:41]
proxies_4 <- data.frame(cbind(count_4, aoritsic_4, SPD))
(Table_5_replication<- cor(proxies_4, method = "pearson"))
```

```
##              count_4 aoritsic_4       SPD
## count_4    1.0000000  0.9767146 0.9490895
## aoritsic_4 0.9767146  1.0000000 0.9416295
```

```
## SPD        0.9490895  0.9416295 1.0000000
```

```r
#check that p value for one of the bold numbers in table 6, i.e correlation between SPD and count.
#the p-value is <0.05 and is significant
(cor.test(SPD, count_4, method = "pearson"))
```

```
##
##  Pearson's product-moment correlation
##
## data:  SPD and count_4
## t = 12.783, df = 18, p-value = 1.813e-10
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8733820 0.9800129
## sample estimates:
##       cor
## 0.9490895
```

Table 6 from paper (for comparison)

Table 6. Pearson correlation coefficient r-value matrix between the archaeo-demographic proxies (only urban sites) from 6,000 to 4,000 cal BP.

	Count	Area	Aoristic weight	Random	SPD
Count	1.00				
Area	**0.98**	1.00			
Aoristic weight	**0.98**	**0.99**	1.00		
Random	**0.97**	**0.98**	**0.98**	1.00	
SPD	**0.95**	**0.93**	**0.94**	**0.93**	1.00

Significant correlations indicated by bold numbers (p-value < 0.05).

https://doi.org/10.1371/journal.pone.0244871.t006

Conclusion/discussion

Overall, the analyses that was replicated in this report was successful. Both archaeological datasets were successfully loaded in and converted into a format that was then used for analyses. Basic descriptive stats for duration and rainfall variables in each dataset were calculated. The data was successfully binned into the time slices outlined in the paper and aoristic weights were successfully calculated using a loop. The process of calculating these weights using r code was not detailed in the paper which made it challenging to replicate. However, the authors did explain the theory behind getting aoristic weights of each site, enough so, that creating a loop to calculate these values was achievable. Replication of parts of the archaeological dataset figures (Fig. 4 and 5) was fairly straight forward and although each figure consisted of a lot of elements, it was not challenging to replicate.

While the paper has sufficient information on processing of the radiocarbon dataset, it was still a challenge to replicate. To assist in this data processing, I found the rcarbon vignette extremely useful (https://cran.r-project.org/web/packages/rcarbon/vignettes/rcarbon.html) and as a result, replication was successful.

The main challenges I faced was trying to process the spatial datasets for analyses. I was able to load the various shp files using the {rgdal} package, however further processing of the files was unclear, and I was unsuccessful in my attempt. A further challenge related to this issue was converting the datasets into an object that could be spatial mapped. While I think the next step would be to extract the coordinates of each site to then spatially map, there was no instruction of how this could be achieved in r. The issue of spatial mapping came up in the radiocarbon part of the paper, in which the dates are divided into rainfall groups. The authors appear to group these dates using the rainfall_zone.shp file, however, no further detail was provided on the specific steps to achieve this and I was unsuccessful in this part of the replication.

Apart for these challenges, all other aspects of replication was successful. The only difference I could spot between the original and replicated results were the very slight differences in the pearson coefficient values (Table 3 and 4). However, I think these differences were not significance enough to consider a failed result. I speculate they might have occurred due to outliers that the original methodology might have excluded.

Overall, the results I achieved seemed to align with those presented in the paper. Given more time (and knowledge on the subject) it would also be interesting to replicate the spatial analyses.

References

Lawrence D, Palmisano A, de Gruchy MW (2021) Collapse and continuity: A multi-proxy reconstruction of settlement organization and population trajectories in the Northern Fertile Crescent during the 4.2kya Rapid Climate Change event. PLoS ONE 16(1): e0244871. https://doi.org/10.1371/journal.pone.0244871