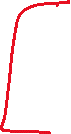
**Example data**

What is time series analysis?



We are going to examine data that has been observed over time. Typically, there is dependence between successive observed data values which limits our ability to use “conventional” statistical analysis methods that rely on independent observations.



In this course, we will

* Learn how to identify this dependence
* Use the dependence to construct models
* Forecast the future!

Most of our course focuses on modeling ONE series of observations without any explanatory variables. There will be some deviations from it.

Does this include longitudinal data? Yes and no.

We will not examine data observed only for a small number of time periods. For example, the type of data that we will not examine includes observations for subjects in medical studies, such as:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Subject** | **Time 1** | **Time 2** | **Time 3** | **Time 4** |
| 1 | 10 | 12 | 11 | 15 |
| 2 | 32 | 21 | 25 | 30 |
|  |  |  |  |  |

Example: OSU enrollment data (osu\_enroll.R, osu\_enroll.csv)

Partial listing of the data:



|  |  |  |  |
| --- | --- | --- | --- |
| **t** | **Semester** | **Year** | **Enrollment** |
| 1 | Fall | 1989 | 20,110 |
| 2 | Spring | 1990 | 19,128 |
| 3 | Summer | 1990 | 7,553 |
| 4 | Fall | 1990 | 19,591 |
| 5 | Spring | 1991 | 18,361 |
| 6 | Summer | 1991 | 6,702 |
|  |  |  |  |
| 32 | Spring | 2000 | 19,835 |
| 33 | Summer | 2000 | 7,202 |
| 34 | Fall | 2000 | 21,252 |
| 35 | Spring | 2001 | 20,004 |
| 36 | Summer | 2001 | 7,558 |
| 37 | Fall | 2001 | 21,872 |
| 38 | Spring | 2002 | 20,992 |
| 39 | Summer | 2002 | 7,868 |
| 40 | Fall | 2002 | 22,992 |

Define xt to represent the enrollment at time t. Thus, x1 = 20,110, x2 = 19,128, …, x40 = 22,992.

> osu.enroll <- read.csv(file = "OSU\_enroll.csv",

stringsAsFactors = TRUE)

> head(osu.enroll)

t Semester Year Enrollment date

1 1 Fall 1989 20,110 8/31/1989

2 2 Spring 1990 19,128 2/1/1990

3 3 Summer 1990 7,553 6/1/1990

4 4 Fall 1990 19,591 8/31/1990

5 5 Spring 1991 18,361 2/1/1991

6 6 Summer 1991 6,702 6/1/1991

> tail(osu.enroll)

t Semester Year Enrollment date

35 35 Spring 2001 20,004 2/1/2001

36 36 Summer 2001 7,558 6/1/2001

37 37 Fall 2001 21,872 8/31/2001

38 38 Spring 2002 20,922 2/1/2002

39 39 Summer 2002 7,868 6/1/2002

40 40 Fall 2002 22,992 8/31/2002

> #One way to do plot

> dev.new(width = 8, height = 6, pointsize = 10)

> plot(x = x, ylab = "OSU Enrollment",

xlab = "t (time)", type = "l", col = "red",

main = "OSU Enrollment from Fall 1989 to Fall 2002",

panel.first = grid(col = "gray", lty = "dotted"))

> points(x = osu.enroll$Enrollment, pch = 20, col = "blue")



When only x is specified in the plot() function, R puts this on the y-axis and uses the observation number on the x-axis. Compare this to the next plot below where both x and y arguments are specified.

> #More complicated plot

> fall <- osu.enroll[osu.enroll$Semester == "Fall",]

> spring <- osu.enroll[osu.enroll$Semester == "Spring",]

> summer <- osu.enroll[osu.enroll$Semester == "Summer",]

> plot(y = fall$Enrollment, x = fall$t,

ylab = "OSU Enrollment", xlab = "t (time)", col =

"blue", main = "OSU Enrollment from Fall 1989 to Fall

2002", panel.first = grid(col = "gray", lty =

"dotted"), pch = 1, type = "o", ylim = c(0,

max(osu.enroll$Enrollment)))

> lines(y = spring$Enrollment, x = spring$t, col = "red",

type = "o", pch = 2)

> lines(y = summer$Enrollment, x = summer$t, col =

"darkgreen", type = "o", pch = 3)



> #Another way to do plot with actual dates

> plot(y = osu.enroll$Enrollment,

x = as.Date(osu.enroll$date, format = "%m/%d/%Y"),

xlab = "Time", type = "l", col = "red",

main = "OSU Enrollment from Fall 1989 to Fall 2002",

ylab = “OSU Enrollment”)

> points(y = osu.enroll$Enrollment,

x = as.Date(osu.enroll$date, format = "%m/%d/%Y"), pch

= 20, col = "blue")

> #Create own gridlines

> abline(v = as.Date(c("1990/1/1", "1992/1/1", "1994/1/1",

"1996/1/1", "1998/1/1", "2000/1/1", "2002/1/1")),

lty = "dotted", col = "lightgray")

> abline(h = c(10000, 15000, 20000), lty = "dotted", col =

"lightgray")



Questions of interest:

1. What patterns are there over time?
2. How can the dependence among observations be used to help model the data?
3. Can future enrollment be predicted using this data?
4. Most of the time, we will only use past values in the series to predict future values. However, in this case, what explanatory variables (independent variables, covariates) may be useful to use to predict enrollment?
5. Why is modeling enrollment and predicting future enrollment important?

5-8-01 O’Colly article: “$1.8 million loss attributed to slight enrollment decline”

<https://www.ocolly.com/1-8-million-loss-attributed-to-slight-enrollment-decline/article_e3116026-d8d5-5d89-ad1f-9136459d84ac.html>



Example: S&P500 Index (SP500.R, SP500weekly.csv)

Source: Yahoo! Finance

This index measures the performance of the 500 largest US companies.

> SP500 <- read.csv(file = "SP500weekly.csv",

stringsAsFactors = TRUE)

> head(SP500)

WeekStart Open High Low Close AdjClose Volume

1 1/1/1995 459.21 462.49 457.20 460.68 460.68 1199080000

2 1/8/1995 460.67 466.43 458.65 465.97 465.97 1627330000

3 1/15/1995 465.97 470.43 463.99 464.78 464.78 1667400000

4 1/22/1995 464.78 471.36 461.14 470.39 470.39 1628110000

5 1/29/1995 470.39 479.91 467.49 478.65 478.65 1888560000

6 2/5/1995 478.64 482.60 478.36 481.46 481.46 1579920000

> tail(SP500)

WeekStart Open High Low Close AdjClose Volume

1395 9/19/2021 4402.95 4465.40 4305.91 4455.48 4455.48 15697030000

1396 9/26/2021 4442.12 4457.30 4288.52 4357.04 4357.04 15555390000

1397 10/3/2021 4348.84 4429.97 4278.94 4391.34 4391.34 14795520000

1398 10/10/2021 4385.44 4475.82 4329.92 4471.37 4471.37 13758090000

1399 10/17/2021 4463.72 4559.67 4447.47 4544.90 4544.90 13966070000

1400 10/24/2021 4553.69 4608.08 4537.36 4605.38 4605.38 16206040000

> x <- SP500$Close

> #One way to do plot

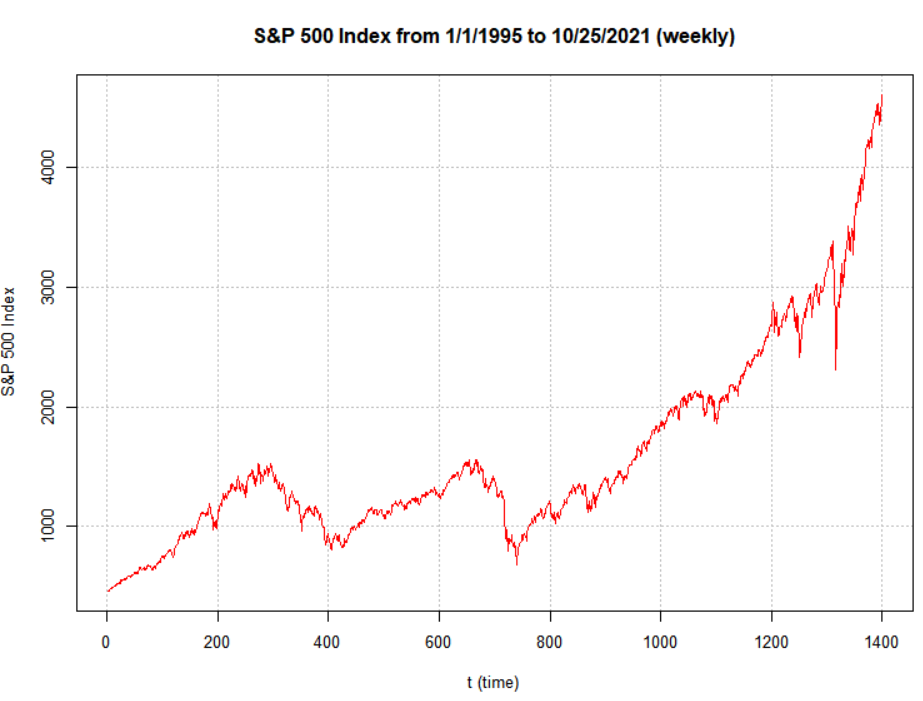
> dev.new(width = 8, height = 6, pointsize = 10)

> plot(x = x, ylab = "S&P 500 Index", xlab = "t (time)",

type = "l", col = "red", main = "S&P 500 Index from

1/1/1995 to 10/25/2021 (weekly)",

panel.first = grid(col = "gray", lty = "dotted"))



> #Another way to do plot with actual dates

> plot(y = x, x = as.Date(x, format =

"%m/%d/%Y"), xlab = "Time", type = "l", col = "red", main

= "S&P 500 Index from 1/1/1995 to 10/25/2021 (weekly)",

ylab = "S&P 500 Index")

> #Create own gridlines

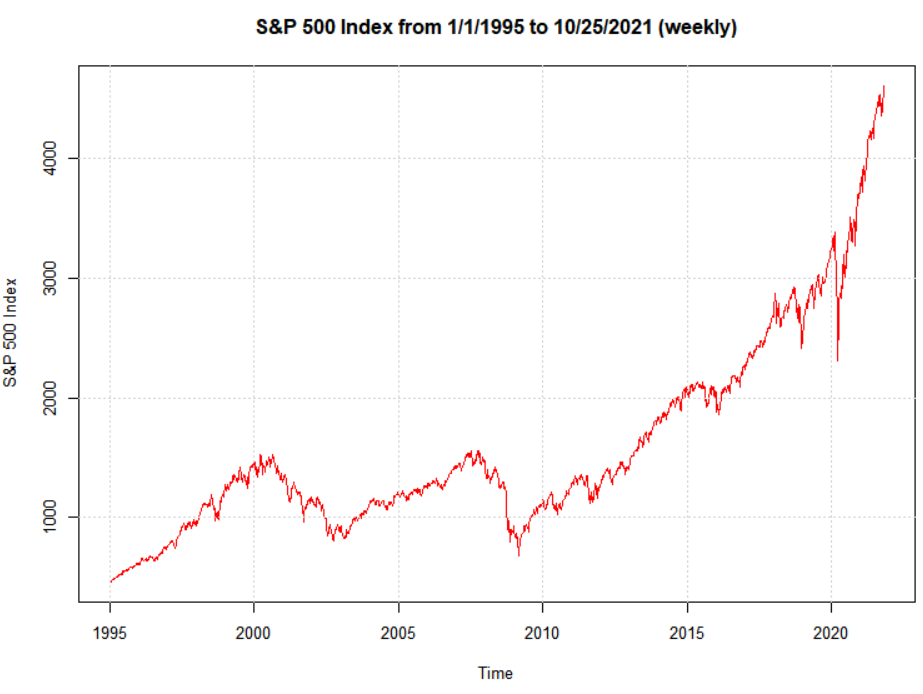
> abline(v = as.Date(c("1995/1/1", "2000/1/1", "2005/1/1",

"2010/1/1", "2015/1/1", "2020/1/1")), lty = "dotted",

col = "lightgray")

> abline(h = seq(from = 0, to = 5000, by = 1000), lty =

"dotted", col = "lightgray")



> # One more way with fine control of the dates

> plot(y = x, x = as.Date(SP500$WeekStart, format =

"%m/%d/%Y"), xlab = "Time", type = "l", col = "red",

main = "S&P 500 Index from 1/1/1995 to 10/25/2021

(weekly)", ylab = "S&P 500 Index", xaxt = "n")

> axis.Date(side = 1, at = seq(from = as.Date("1995/1/1"),

to = as.Date("2021/12/31"), by = "years"), labels =

format(x = seq(from = as.Date("1995/1/1"), to =

as.Date("2021/12/31"), by = "years"), format = "%b%y"),

las = 2) #las changes orientation of labels

> #Create own gridlines

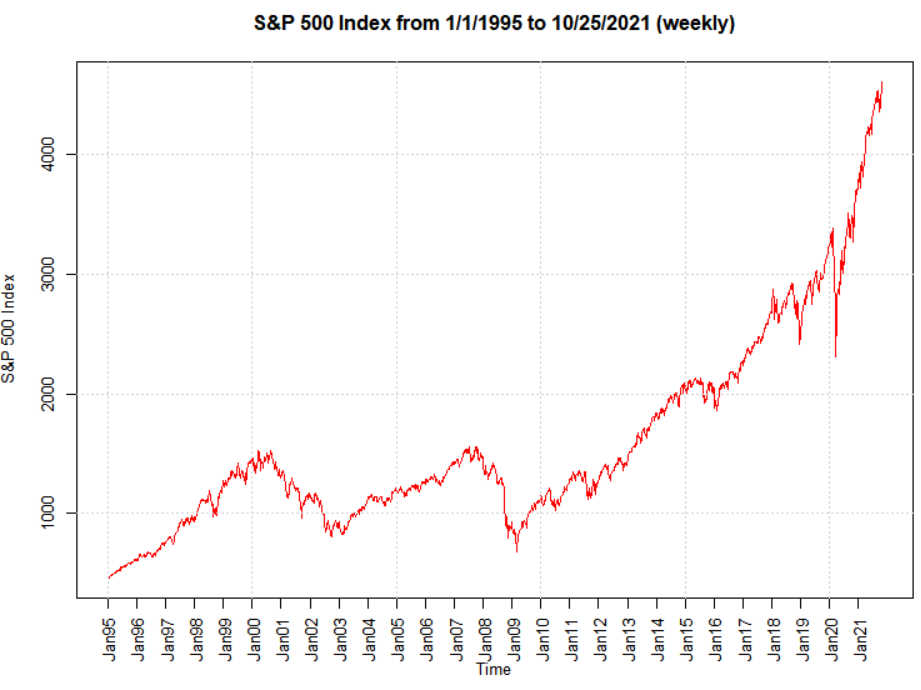
> abline(v = as.Date(c("1995/1/1", "2000/1/1", "2005/1/1",

"2010/1/1", "2015/1/1", "2020/1/1")), lty = "dotted",

col = "lightgray")

> abline(h = seq(from = 0, to = 5000, by = 1000), lty =

"dotted", col = "lightgray")



Questions of interest:

1. What patterns are there over time?
2. How can the dependence among observations be used to help model the data?
3. Can future index values be predicted using this data?
4. Why is modeling the index and predicting future index values important?

Example: Sunspots (sunspots.R, SN\_y\_tot\_V2.0.csv)

Source: <https://wwwbis.sidc.be/silso/home>

Number of sunspots per year on the sun from 1700-2020.

> sunspots <- read.table(file = "SN\_y\_tot\_V2.0.csv", sep =

";", col.names = c("Mid.year", "Mean.total",

"Mean.SD.total", "Numb.obs.used", "Definitive"))

> head(sunspots)

Mid.year Mean.total Mean.SD.total Numb.obs.used Definitive

1 1700.5 8.3 -1 -1 1

2 1701.5 18.3 -1 -1 1

3 1702.5 26.7 -1 -1 1

4 1703.5 38.3 -1 -1 1

5 1704.5 60.0 -1 -1 1

6 1705.5 96.7 -1 -1 1

> tail(sunspots)

Mid.year Mean.total Mean.SD.total Numb.obs.used Definitive

316 2015.5 69.8 6.4 8903 1

317 2016.5 39.8 3.9 9940 1

318 2017.5 21.7 2.5 11444 1

319 2018.5 7.0 1.1 12611 1

320 2019.5 3.6 0.5 12884 1

321 2020.5 8.8 4.1 14440 1

> dev.new(width = 8, height = 6, pointsize = 10)

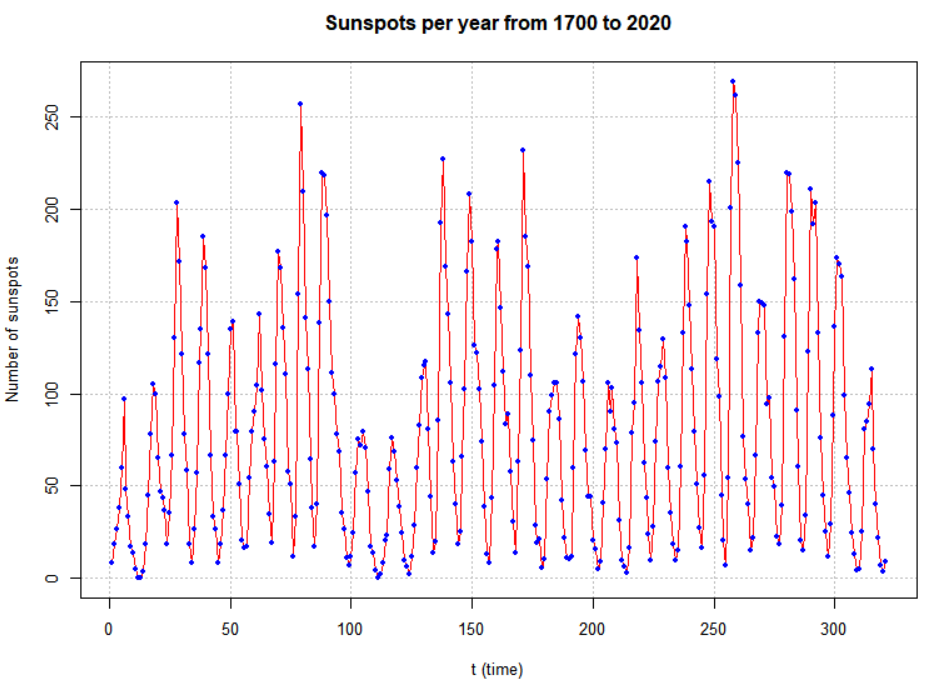
> plot(x = sunspots$Mean.total, ylab = "Number of

sunspots", xlab = "t (time)", type = "l", col = "red",

main = "Sunspots per year from 1700 to 2020",

panel.first = grid(col = "gray", lty = "dotted"))

> points(x = sunspots$Mean.total, pch = 20, col = "blue")



> # Include dates

> plot(y = sunspots$Mean.total, x = sunspots$Mid.year, ylab

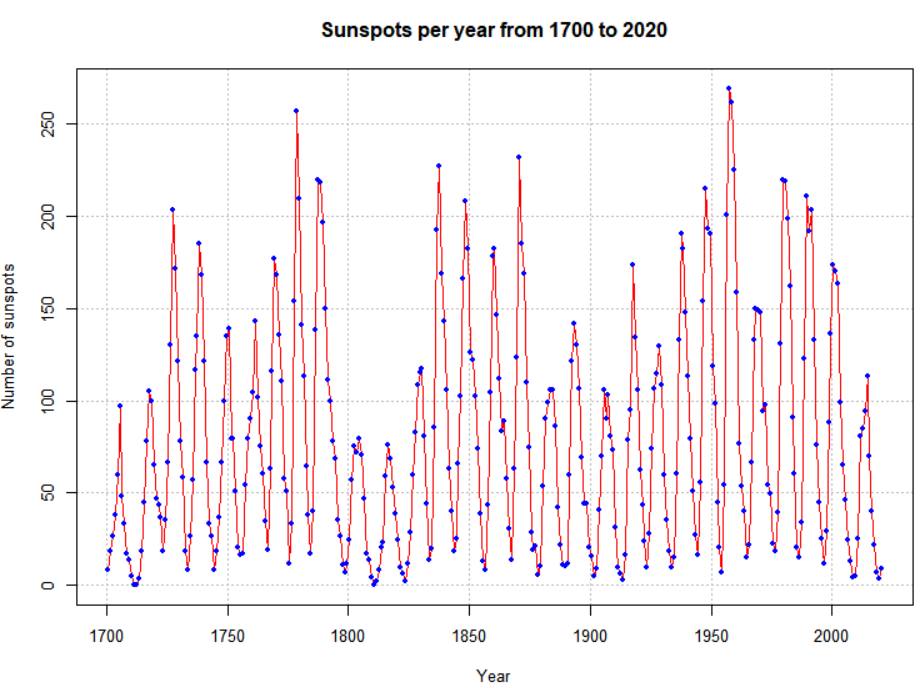
= "Number of sunspots", xlab = "Year", type = "l", col

= "red", main = "Sunspots per year from 1700 to 2020",

panel.first = grid(col = "gray", lty = "dotted"))

> points(y = sunspots$Mean.total, x = sunspots$Mid.year,

pch = 20, col = "blue")



> #Convert to an object of class "ts"

> x <- ts(data = sunspots$Mean.total,, start = 1700, frequency

= 1)

> class(x)

[1] "ts"

> class(sunspots$Mean.total)

[1] "numeric"

> x

Time Series:

Start = 1700

End = 2020

Frequency = 1

[1] 8.3 18.3 26.7 38.3 60.0 96.7 48.3 33.3 16.7

[10] 13.3 5.0 0.0 0.0 3.3 18.3 45.0 78.3 105.0

EDITED

[307] 24.7 12.6 4.2 4.8 24.9 80.8 84.5 94.0 113.3

[316] 69.8 39.8 21.7 7.0 3.6 8.8

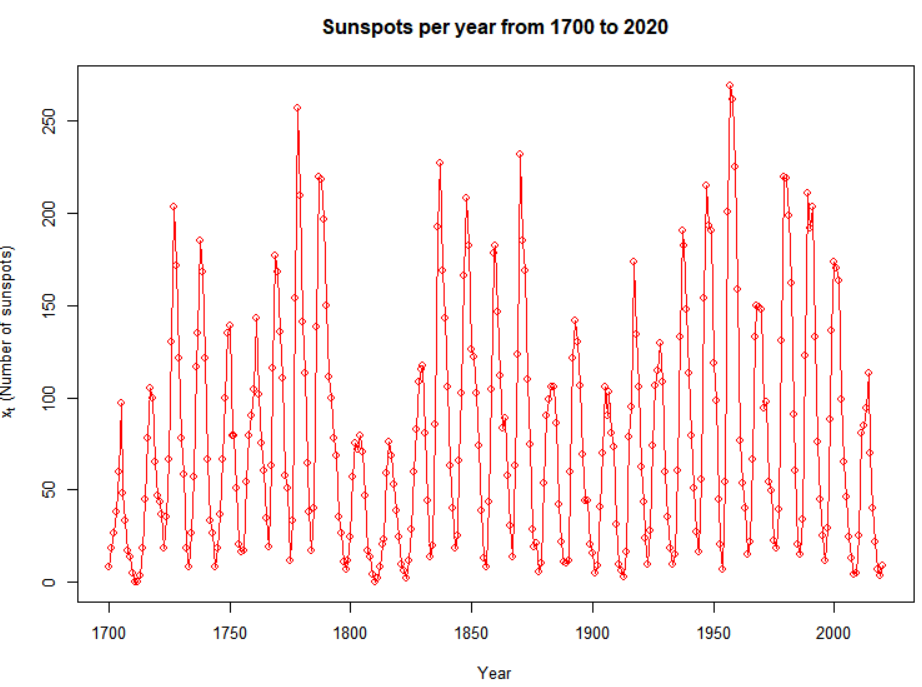
> # plot() is a generic function - uses the plot.ts() method

function

> plot(x = x, ylab = expression(paste(x[t], " (Number of

sunspots)")), xlab = "Year", type = "o", col = "red", main

= "Sunspots per year from 1700 to 2020")



Notes:

* The sunspot values are not necessarily integers.
* Every object in R has a class. For time series data, it is sometimes useful to use a ts class type with it.

Questions of interest:

1. What patterns are there over time? 10-12 year cycles
2. How can the dependence among observations be used to help model the data?
3. Can future sunspots be predicted using this data?
4. Why is modeling the number of sunspots and predicting future values important?

There are many more examples of time series data from a diverse set of areas!