Dynamical Systems Tutorial

# Dynamical Systems Tutorial

## Error: cannot change working directory

## Prepping the Data

The first thing that needs to be done is get the data in the correct format, which means creating a lagged version of the variable we are interested in. The data that I am using is course enrollment data from 2000-2014.

First read in the data and look at the first few lines:

# Read in tab delimited text data with headers  
d01 <- read.delim("Query1.txt", header = TRUE, sep = "\t")  
head(d01)

## SNum Days CurrentEnroll Status LowerDiv Year Semester Location Time2  
## 1 2 M,W,F 200 ? 1 2000 spring main Morning  
## 2 3 M,W,F 69 A 1 2000 spring main Morning  
## 3 10 H 57 A 1 2000 spring main Evening  
## 4 1 M,W,F 70 ? 1 2000 spring main Morning  
## 5 2 T,H 57 P 1 2000 spring main Morning  
## 6 3 T,H 66 P 1 2000 spring main Morning  
## Days2 CNum Area CredHours NumAttributes CrossList Req  
## 1 3 1010 None 4 0 0 1  
## 2 3 1010 None 4 0 0 1  
## 3 1 1010 None 4 0 0 1  
## 4 3 1020 None 3 0 0 0  
## 5 2 1020 None 3 0 0 0  
## 6 2 1020 None 3 0 0 0

This shows every section of every course, we want to look at the change in total course enrollment over time so the data needs to be manipulated a bit. First we need to aggregate all the different sections of the same course. Then we need to create a lagged variable (the enrollment from one year prior t0 and t+1) which will help us understand the change in enrollment from year to year.

library(dplyr)  
library(DataCombine)  
  
# Create new dataframe with only the Year, Course Number and Total  
# Enrollment  
d02 <- d01 %.% group\_by(CNum, Year, Semester) %.% summarise(total = sum(CurrentEnroll)) %.%   
 arrange(desc(total))  
  
# Order the data by Course number and year  
attach(d02)  
d02 <- d02[order(CNum, Year), ]  
  
# Create lagged variable for each course  
d02 <- slide(d02, Var = "total", GroupVar = "CNum", NewVar = "LagSem", slideBy = -1) # Semester to semester  
d02 <- slide(d02, Var = "LagSem", GroupVar = "CNum", NewVar = "LagYear", slideBy = -1) # Fall to Fall

## Exploratory Analysis

Now we have a small data set (d02) with a lagged variable for every course. Now we can plot the total number of course enrollments by the lagged number of course enrollments. If a scatter plot is created with the course enrollments are plotted on the x-axis and lagged enrollments (number enrolled in the same class one year later) we get a graph that looks like this.

library(ggplot2)  
sp01 <- ggplot(d03, aes(total, LagSem))

## Error: object 'd03' not found

p1 <- sp01 + geom\_point() + labs(title = "Course Enrollment by Enrollment the Previous Semester")

## Error: object 'sp01' not found

p1

## Error: object 'p1' not found

If there is no change year over year the data points should line up on a perfect 45 degree line. We can add that 45 degree line and add a smoother line to see where the set points are.

p1 + geom\_abline(intercept = 0, slope = 1) + stat\_smooth()

## Error: object 'p1' not found

Another way to look at this is to create a difference score, which will rotate the graph so no change is a horizontal line.

d02$DiffScoreS <- d02$total - d02$LagSem  
  
sp02 <- ggplot(d02, aes(total, DiffScoreS))  
p2 <- sp02 + geom\_point() + labs(title = "Course Enrollment by Difference 1 year later",   
 x = "Enrollment", y = "Change in Enrollment") + geom\_abline(intercept = 0,   
 slope = 0) + stat\_smooth()  
p2

plot of chunk sp02

plot of chunk sp02

# Same thing for year to year change  
d02$DiffScoreY <- d02$total - d02$LagYear  
  
sp03 <- ggplot(d02, aes(total, DiffScoreY))  
p3 <- sp03 + geom\_point() + labs(title = "Course Enrollment by Difference 1 year later",   
 x = "Enrollment", y = "Change in Enrollment") + geom\_abline(intercept = 0,   
 slope = 0) + stat\_smooth()  
p3

plot of chunk sp02

plot of chunk sp02

If the smoothed slope (the blue line) crosses the horizontal line it indicates that this is were there is no change from one year to the next, a set point. Set points come in two flavors when looking at a single variable like this, they are either “attractor” or “repellents”. Attractors are stable points, and it is expected that values will converge to these set points over time. In other words the attractors are attractive and pull values close to them.

Repellors are the opposite, they push things away, repelling values away over time.

Telling which set points are attractors and which are repellors is pretty easy, attractors have a negative slope, and repellors have a positive slope. Looking at the difference plot above there seems to be an attractive set point (attractor) at the 400 mark, and a repulsive set point (repellor) at the 825 mark.

With this data it is hard to see what is going on where there are a lot of points. In the data course 1010 is always highly enrolled and is skewing all of the rest of the data. Lets remove 1010 and re-run the analysis.

d03 <- d02[which(d02$CNum != 1010), ]  
sp04 <- ggplot(d03, aes(total, DiffScoreY))  
sp04 + geom\_point() + labs(title = "Course Enrollment by Difference 1 year later \n (without 1010)") +   
 geom\_abline(intercept = 0, slope = 0) + stat\_smooth()

plot of chunk wo1010

plot of chunk wo1010

Now it looks like there is a repellor around an enrollment of 200. Looking closely at the blue line it appears to parallel the horizontal line for quite a while before crossing it. The slope of the line not only tells us if a set point is an attractor or repellor, but it also can tell us how strong the attraction/repulsion is. In this case is looks like the repellor is very weak until enrollment reaches 200, and then enrollment shoots off to higher numbers.

The next step is to figure out exactly where these set points are and how strong they are.

## Regression Analysis

We can use basic polynomial regression to figure out exactly where the set points are and how strong they are. Based on the graphs above we can see 1 repellor at an enrollment of 200, an attractor at about 400, and another repellor at about 825. This line should be able to be fit witha polynomial function (X3).

In a regression analysis we could say that the year to year difference in enrollment is equal to an intercept (b0) plus enrollment raised to a power of three (X3) plus a error term (e), or \[ \Delta x = \beta\_0 + \beta\_1x^3 + e \].

In regression if there is a polynomial (X3) all lower order terms must also be included (X2 , X). So our equation turns into \[ \Delta x = \beta\_0 + \beta\_1x^3 + \beta\_2x^2 + \beta\_3x^3 + e \].

lm01 <- lm(DiffScoreS ~ poly(total, 3), data = d02)  
summary(lm01)

##   
## Call:  
## lm(formula = DiffScoreS ~ poly(total, 3), data = d02)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -189.0 -15.1 4.5 13.8 190.1   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.19 1.14 -1.04 0.30   
## poly(total, 3)1 194.21 34.43 5.64 2.3e-08 \*\*\*  
## poly(total, 3)2 27.41 34.55 0.79 0.43   
## poly(total, 3)3 346.01 34.30 10.09 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 33.5 on 861 degrees of freedom  
## (86 observations deleted due to missingness)  
## Multiple R-squared: 0.137, Adjusted R-squared: 0.134   
## F-statistic: 45.5 on 3 and 861 DF, p-value: <2e-16