Dartmouth GEISEL SCHOOL OF MEDICINE

DEPARTMENT OF BIOMEDICAL DATA SCIENCE GEISEL SCHOOL OF MEDICINE

Center for Technology and Behavioral Health
Innovate · Evaluate · Disseminate

It’s Time to Ask When: Investigating the Timing of Dynamics in Intensive Longitudinal Data

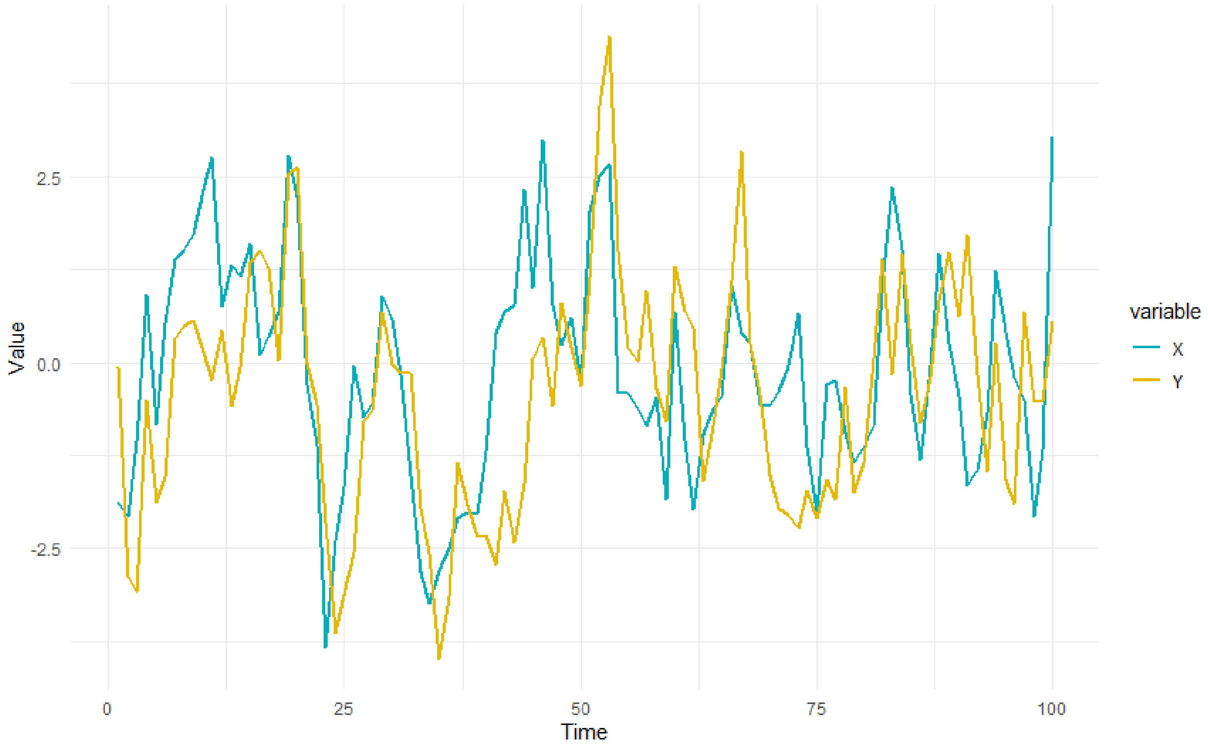
Nicholas C. Jacobson, Ph.D.

SAA
Society for Ambulatory Assessment 2020

@NC_Jacobson
nicholasjacobson.com

Intensive Longitudinal Data: A Powerful Tool

- The collection of intensive longitudinal data can access dynamic processes
- Many of our psychological theories evolve states which change over time
- Dynamic processes: Relationships one or more variables that vary over time



Design Considerations: Timing of Active Assessments

- When we don’t know the “ground truth” and want to try to sample it periodically
  - Does not apply to event contingent designs

- Options:
- **Signal contingent:** Ping people and ask that they complete the assessments
  - **Suboption**: Can choose between random (8 prompts per day at any time) or windowed random (random time between 8 – 10 AM; 10:01 AM – 12 PM, etc)
- **Interval contingent**: Ask that persons complete assessments at a specified interval (e.g. every hour; every 15 minutes, every day)

Design Considerations: Timing of Assessments

- With dynamic processes, timing matters a LOT
- If we mis-time our data collection, it has huge implications
- **Example:** Movement of helicopter blades
- Timing of assessments (i.e. shutter speed here) can make it look like they move very slowly or not at all
  - Can change the direction of the blades

Video Credit:

BrainStuff - HowStuffWorks

How do You Choose How Often to Sample?

- Theory!
  - If you have a strongly suggestive theory about timing, great!
- But…most theories do not specify ***when***…
- **Common practice!:** Researcher X, Y, & Z each used that interval before!
  - **Do not do this!**
- **Best practice!:** Try to sample your process systematically – it's much better to oversample than undersample
- Incorrect timing will **not** allow us to test our theories

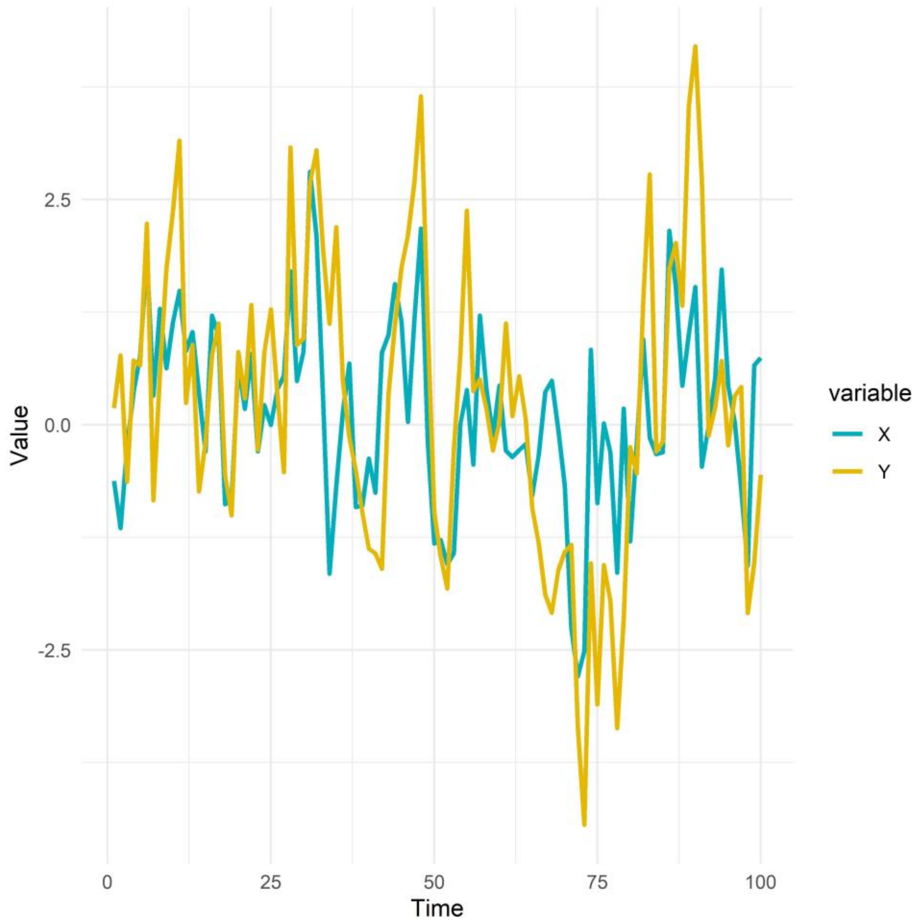
Analyzing Intensive Longitudinal Data

- **Common practice**: Look at dynamic processes concurrently or one unit in time (i.e. lag) later
  - Okay if you have a strong theory about timing
  - **Not okay** in the vast majority of intensive longitudinal data work

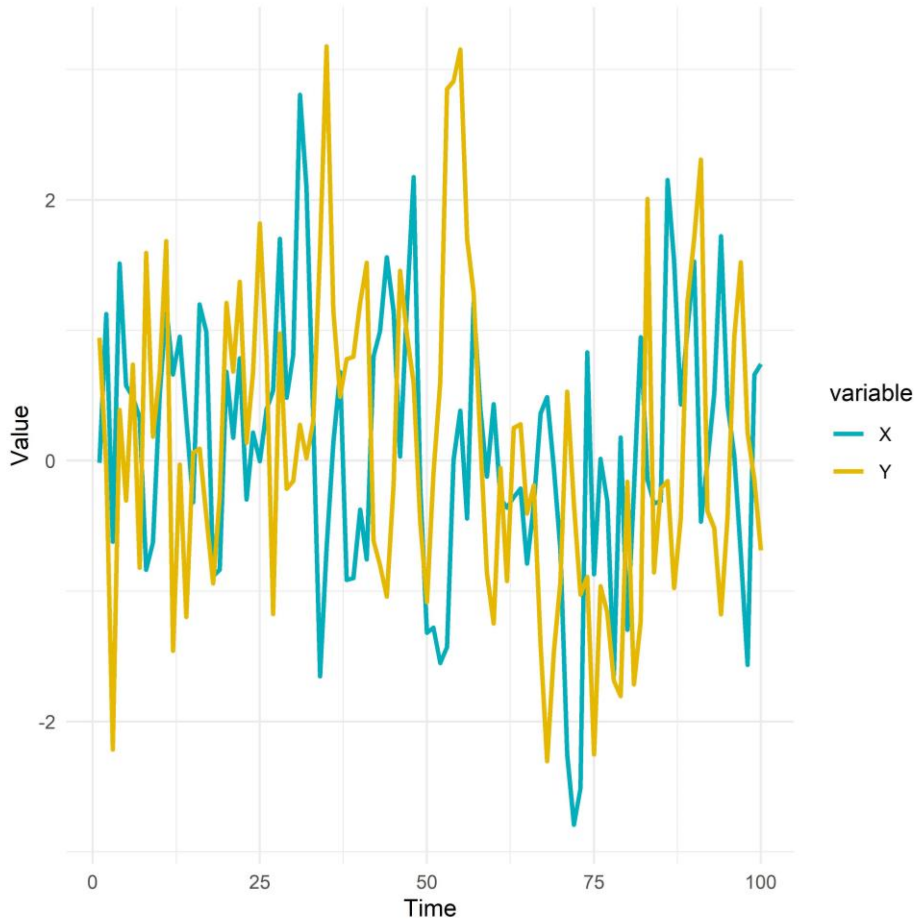
- **Best practice**: Investigate the times in which processes optimally predict one another

I would just see it in my data, right?

- Yes with small lag:
- Lag 1



- **<u>No</u>** with large lags:
- Lag 10



Prior Analytic Tools to Identify Higher Order Lags

- Manually investigate higher-order lags (1,2,3…X)
  - ***High chance*** of false positives
- Continuous time structural equation modeling
  - Okay option, but inflexible
  - Restricted to a few functional forms
  - Unless you use higher-order stochastic differential equations, but then will likely be uninterpretable
- Fractals
  - Potentially good alternative, but less accessible to most & much harder to interpret with precision
- Auto and cross-correlation functions: not designed for missingness and not designed for multiple people

New Alternative:
The Differential Time-Varying Effect Model

- Automated R package which fits smooth curves to the data to explore potential higher order lags, strongly backed by simulation studies

- Next: subsequently performs confirmatory analyses using gold-standard state-space modeling routines

- Usages: One to many predictors, one to many outcomes, one to many people
- Uses full information maximum likelihood for missing data

Behavior Research Methods
https://doi.org/10.3758/s13428-018-1101-0

CrossMark

The Differential Time-Varying Effect Model (DTVEM): A tool for diagnosing and modeling time lags in intensive longitudinal data

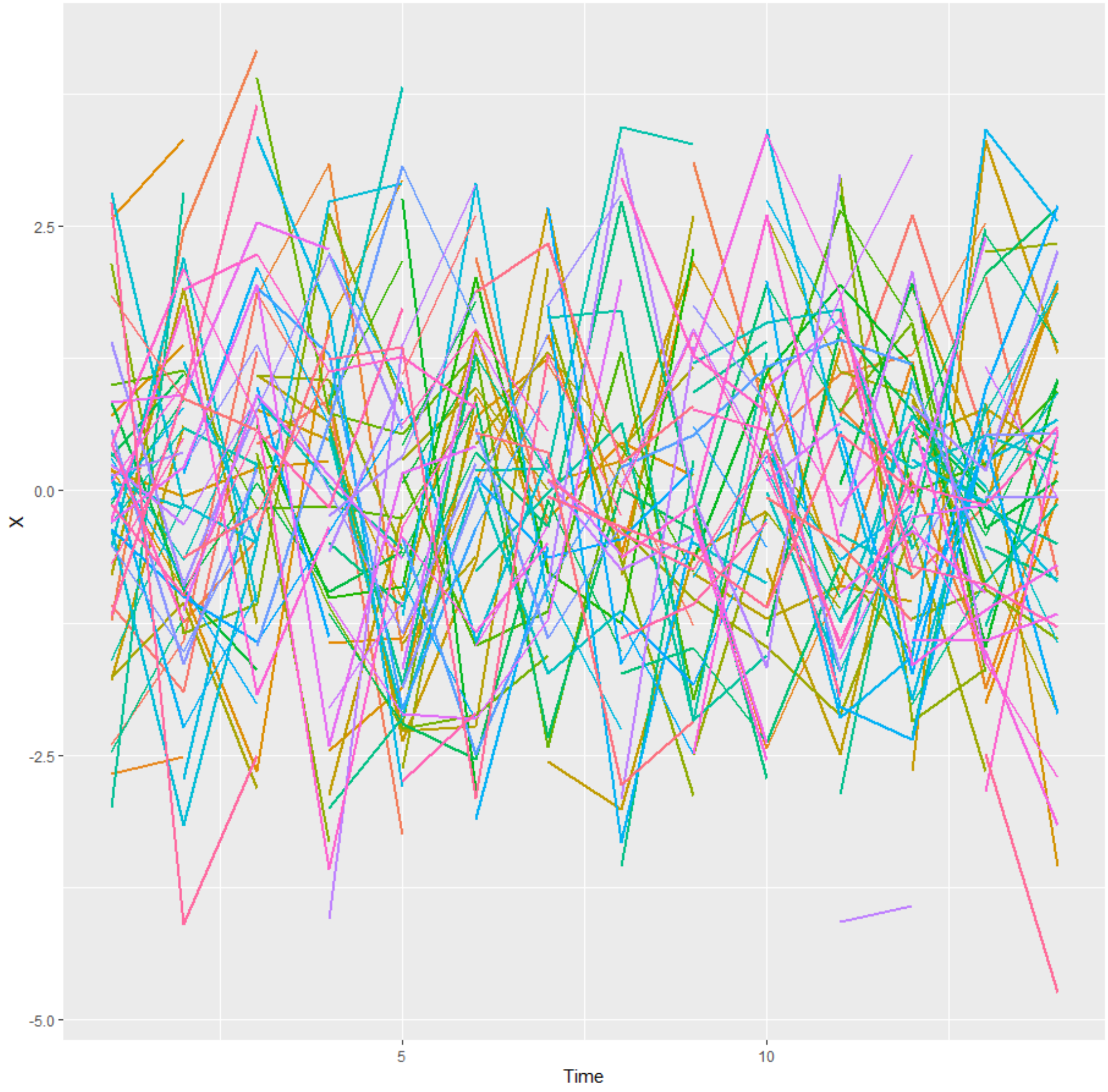
Nicholas C. Jacobson[1] · Sy-Miin Chow[1] · Michelle G. Newman[1]

Tutorial

- One variable collected
- Interested in X predicting itself

- 100 people
- 14 day daily diary study

- Code:
- library(DTVEM)
- data(exampledat1)



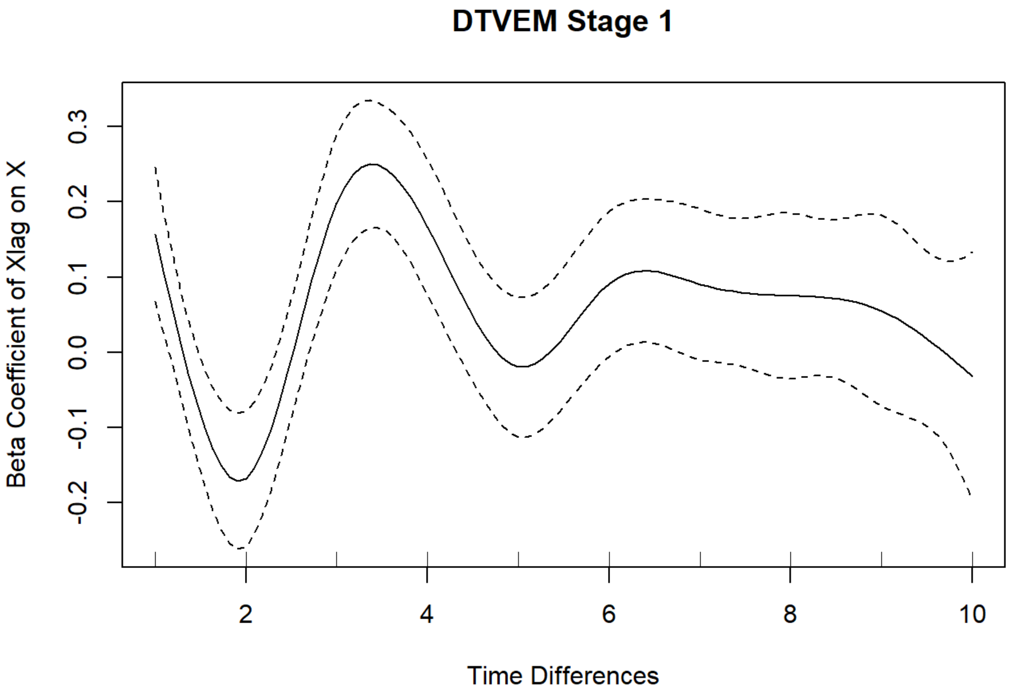
Variable Predicting Itself Over Time

- A single line of code:

- out=LAG("X",differntialtimevaryingpredictors=c("X"),outcome=c("X"), data=exampledat1,ID="ID",Time="Time",k=9,standardized=FALSE,predictionstart = 1,predictionsend = 10,predictionsinterval = 1)

Output

Exploratory

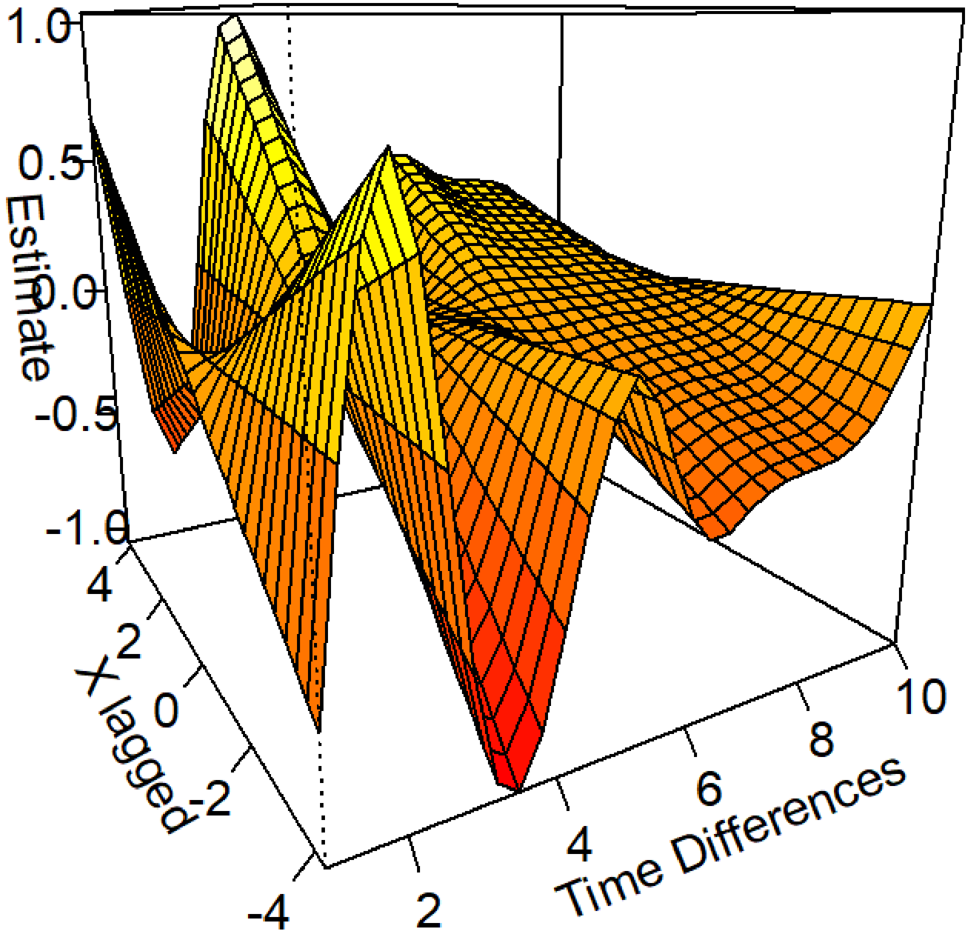


Confirmatory

```
##          name matrix row col   Estimate  Std.Error lbound ubound lboundMet
## 1 XlagonXlag1   s1.A   1   1  0.2360088 0.04461888     NA     NA     FALSE
## 2 XlagonXlag2   s1.A   1   2 -0.2485319 0.04220480     NA     NA     FALSE
## 3 XlagonXlag3   s1.A   1   3  0.2835748 0.04300307     NA     NA     FALSE
##   uboundMet     tstat       pvalue  sig
## 1     FALSE  5.289438 1.226928e-07 TRUE
## 2     FALSE -5.888712 3.892167e-09 TRUE
## 3     FALSE  6.594293 4.272871e-11 TRUE
```

A second line of code to visualize

- vis.gam(out$stage1out$mod,xlab="Time Differences",ylab="X lagged",zlab="Estimate",theta=-30,ticktype="detailed")



It’s Time to Ask When Taking this into Practice

- **Design Recommendations**:
- Do: test theories if they are available (usually not)
- Do: When entering the unknown try to oversample a bit (better to catch a fast-moving process), you can always look at a higher lag, you can’t look at data that doesn’t exist

- **Analysis Recommendations:**
- Do: Test lags if they are theorized (usually not)
- Do: Explore higher-order lags using freely available tools

Where do I get this?

- http://www.nicholasjacobson.com/project/dtvem/

Thank You to My Lab!

Graduate students/Residents:

- Michael Heinz, M.D.
- Matthew Nemesure, M.S.
- Shuai Jiang, M.S.
- Damien Lekkas, M.S.
- Curtis Petersen, M.S.
- Jason Kim, B.S.
- Lili Liu, B.S.
- Mengdie Ge, B.S.

Undergraduate Research Assistants:

- Colin Goodbred
- Brandon Feng
- Alexandra Limb
- Iain Sheerin
- Mia Seiner
- MK Song
- Rapael Huang
- Vernoica Abreu
- William Chen

Postbac Research Assistant

- Natalie Thomas

Software Engineer

- William Hasslet, Ph.D.