Practical Interpretation and Insights with Recurrence Quantification Analysis for Decision Making Research

Erin McCormick, Leslie Blaha

Carnegie Mellon University
Air Force Research Laboratory

July 26, 2021

Welcome!

Structure of the tutorial today:

Tutorial lectures

- Live in Zoom
- Slides available in the tutorial zip at https://tinyurl.com/5ej4zu7b

Most lectures include exercises

- During lectures, have handy "cogsci2021_tutorial_exercises.HTML"

All lectures and Q&A on Zoom

- Meeting ID: 927 7068 5564, passcode: 128076

Welcome!

Structure of the tutorial today:

Tutorial lectures

- Live in Zoom
- Slides available in the tutorial zip at https://tinyurl.com/5ej4zu7b

Most lectures include exercises

- During lectures, have handy "cogsci2021_tutorial_exercises.HTML"

All lectures and Q&A on Zoom

- Meeting ID: 927 7068 5564, passcode: 128076

Course materials are provided in "PracticalRQA.ZIP"

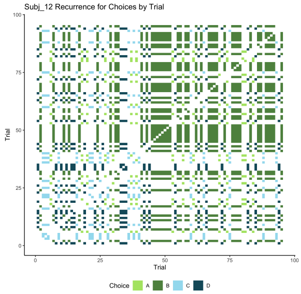
This ZIP archive includes:

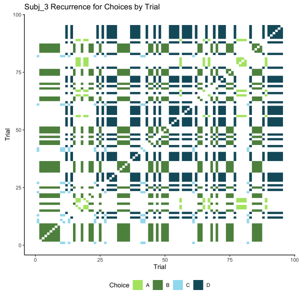
- Proceedings Description (PDF)
- Exercises (HTML, RMD, data folder)
- Lecture slides (PDF)
- Helper code (R)
- Helper code demonstration file (RMD, HTML)
- Interactive Shiny App (R)

Resources:

You can post questions in the chat or submit to https://forms.gle/ipkr1pJBPsCnc1Jf9

You can download the tutorial materials in PracticalRQA.zip at: https://tinyurl.com/5ej4zu7b





Additional Resources

Our tutorial will emphasize *discrete RQA* and its interpretation for the study of decision making and choice sequence data.

More great tutorial resources for anyone interested in reading more or delving deeper into analysis of continuous variables:

- Coco, M. I., & Dale, R. (2014). Cross-recurrence quantification analysis of categorical and continuous time series: An R package. *Frontiers in Psychology, 5*, 510.
  - https://www.frontiersin.org/articles/10.3389/fpsyg.2014.00510/full
  - CRQA Package: https://cran.r-project.org/web/packages/crqa/index.html
  - We'll rely on this R package in this tutorial. Thanks to the Co-Mind Lab for making it available!
- Wallot, S., & Leonardi, G. (2018). Analyzing multivariate dynamics using cross-recurrence quantification analysis (CRQA), diagonal-cross-recurrence profiles (DCRP), and multidimensional recurrence quantification analysis (MDRQA)—a tutorial in R. *Frontiers in Psychology, 9*, 2232.
  - https://www.frontiersin.org/articles/10.3389/fpsyg.2018.02232/full?report=reader

Topics covered

Section 1

- Why use recurrence quantification analysis?
- Technical foundation of discrete recurrence quantification analysis

Section 2

- Interpreting recurrence plots

Section 3

- Comparing sequences to reference strategies with event state coding
- Assessing strategy consistency with auto-RQA and cross-RQA

Section 4

- Recurrence quantification statistics

Section 5

- Creating recurrence plots and computing recurrence statistics in R

1. Section 1
   - Why use RQA in decision making research?
   - Technical foundation of discrete recurrence quantification analysis
   - Questions and answers

Why use RQA in decision making research?

Because decision making is a complex, dynamic system.

Why use RQA in decision making research?

Because decision making is a complex, dynamic system.

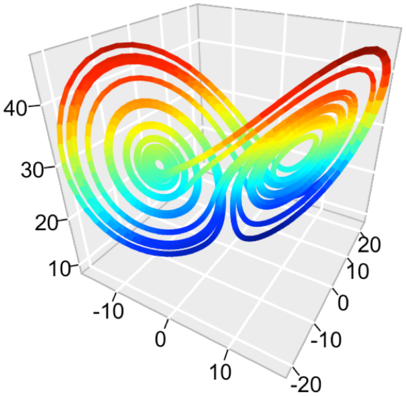
Consider that choice sequences are samples of this dynamic system over time.

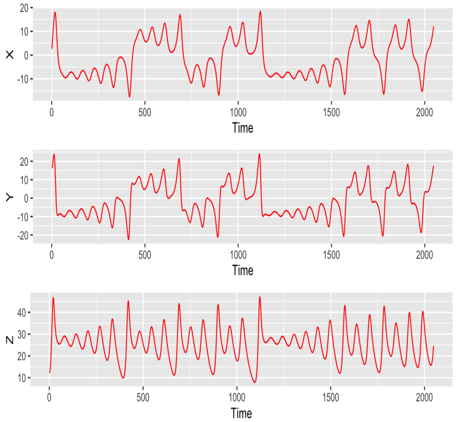
These choice sequences hold a wealth of information about decision making strategies are playing out or being leveraged.

Instead of aggregating over time, losing some of that information, we can use recurrence analysis to **capture, quantify and characterize decision strategies** through the dynamics of decision making.

Technical Foundations

Recurrence analysis was developed in physics to aid the study of complex dynamic systems.

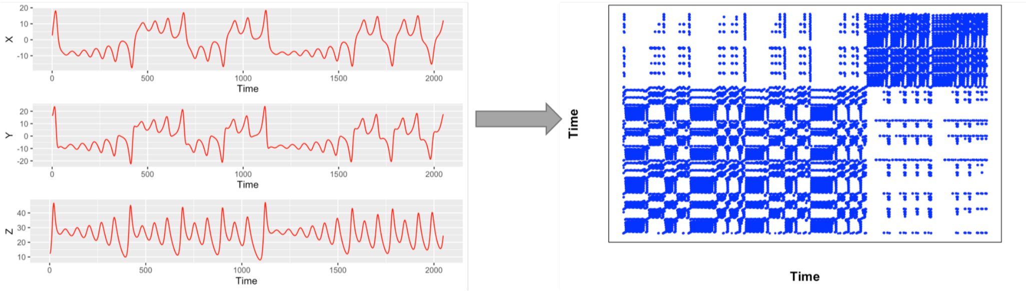




Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences 20*(2) 130–141.

Technical Foundations

Eckmann et al. (1987) demonstrated that visualizing patterns of recurrence, or repetitions in state, in the phase space might provide tractable insights into the dynamics and behavior of complex systems.



Eckmann, J.-P., Kamphorst, S. O. & Ruelle, D. (1987). Recurrence plots of dynamical systems. *Europhysics Letters, 5*, 973-977.

Technical Foundations

RQA has been adopted to behavioral sciences in a few places, with some traction in the cognitive development literature, multi-modal and motor behavior data, communication literature, and recently decision making.

- Quantifying patterns/coordination of limb movements + vocalizations in infant development (Abney et al., 2014)
- Coupling and coordination of speaker and listener in interpersonal interactions (Richardson & Dale, 2005; Richardson & Johnston, 2005)
- Examination of bimanual rhythmic coordination (Shockley & Turvey, 2005)
- Characterizing adaptive behaviors in dynamic decision making strategies (McCormick et al., 2020)
- Comparing communication patterns in all-human compared to human-synthetic agent teams (Bibyk, et al., 2021)

Technical Foundations

RQA is an approach to computing a low-dimensional phase-space embedding of a potentially high-dimensional dynamic system.

The state of a dynamic system is called recurrent if it is sufficiently close (defined by $\delta()$) to a state from an earlier point in time.

Formal definition: $R_{i,j}^{m,\epsilon} = \Theta(\epsilon - \|\mathrm{x}_i - \mathrm{x}_j\|) * \delta(\dot{\mathrm{x}}_i * (\mathrm{x}_i - \mathrm{x}_j))$

Technical Foundations

RQA is an approach to computing a low-dimensional phase-space embedding of a potentially high-dimensional dynamic system.

The state of a dynamic system is called recurrent if it is sufficiently close (defined by $\delta()$) to a state from an earlier point in time.

Formal definition: $R_{i,j}^{m,\epsilon} = \Theta(\epsilon - \|\mathrm{x}_i - \mathrm{x}_j\|) * \delta(\dot{\mathrm{x}}_i * (\mathrm{x}_i - \mathrm{x}_j))$

- Definition of “close” is particularly important for assessing continuous measures.
- In continuous dimensions, may not return to exactly the same values, but may be very close.
- e.g., gaze position may repeat the same fixation but slightly different $< x, y >$ coordinates of the eye movement
- Researchers must choose a distance metric appropriate for the nature and dimensionality of data

Marwan & Webber (2014, p. 7, equation 1.5)

Discrete Recurrence Quantification Analysis

In the case of discrete variables, “sufficiently close” means revisiting a categorical state that occurred at a prior point in time.

$$\mathbf{R}_{i,j} = \begin{cases} 1, & \mathit{state}_i = \mathit{state}_j \\ 0, & \mathit{state}_i \neq \mathit{state}_j \end{cases}$$

RQA for Choice Sequences

Applying this to decision making or choice sequences, we use the following definition of current choice behaviors:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & \textit{choice}_i = \textit{choice}_j \\ 0, & \textit{choice}_i \neq \textit{choice}_j \end{cases}$$

Auto-RQA versus Cross-RQA

RQA can be applied to study the patterns within and between pairs of times series

- Caveat that the times scales need to be comparable if you are comparing two different time series

Auto-recurrence is the analysis of recurrent patterns within a single time series

- Does some pattern of behavior repeat over within that single measurement or single system?
- e.g., Does gaze revisit any ROIs in a visual search task? Are there repeated sequences of choices?

Auto-RQA versus Cross-RQA

RQA can be applied to study the patterns within and between pairs of times series

- Caveat: the times scales need to be comparable if you are comparing two different time series

Auto-recurrence is the analysis of recurrent patterns within a single time series

- Does some pattern of behavior repeat over time within that single measurement or single system?

Cross-recurrence is the analysis of recurrent patterns between two different time series

- Are there points in time where two systems are in the same state?
- e.g., Are two individuals looking at the same ROIs? Is one person repeating what another person is saying? Is a model making the same patterns of behavior as a person?

Auto-RQA versus Cross-RQA

RQA can be applied to study the patterns within and between pairs of times series

Auto-recurrence is the analysis of recurrent patterns within a single time series

Cross-recurrence is the analysis of recurrent patterns between two different time series

The bulk of this tutorial will be on foundations for computing and interpreting auto-RQA for characterizing individual’s behavior.

We will demonstrate ways to use cross-RQA in decision making research.

The R code will also have content for both auto-RQA and cross-RQA.

Discrete RQA for Decision Making

For the study of decision making, we emphasize the study of the sequence of choices or decisions made.

Note that unlike many complex, continuous dynamic systems under study, our question of interest is <u>not</u> *“is there recurrence in this system?”*

Discrete RQA for Decision Making

For the study of decision making, we emphasize the study of the sequence of choices or decisions made.

Note that unlike many complex, continuous dynamic systems under study, our question of interest is not *“is there recurrence in this system?”*

Rather, in a task with n choice options and with at least $n + 1$ trials or measurements, we will observe recurrence.

So the questions of interest are:

- *what are the recurrence patterns in the choice data?*
- *what do they reveal about decision making strategies?*

Exercises

Check your understanding

Complete Exercises 1.01 to 1.03

Question and answer

Questions?

RQA Interpretation

2 Section 2

- Recurrence plot basics
- Questions and answers
- Higher-level visual features/patterns
- Questions and answers

Recurrence basics

Using the definition of discrete recurrence:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & choice_i = choice_j \\ 0, & choice_i \neq choice_j \end{cases}$$

Recurrence basics

Using the definition of discrete recurrence:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & choice_i = choice_j \\ 0, & choice_i \neq choice_j \end{cases}$$

We can analyze an example sequence, where decision makers chose between Button A and Button B for 7 trials:

A, B, A, B, B, B, B

Recurrence basics

Example: A, B, A, B, B, B, B

Recurrence Quantification Analysis steps:

1. Define possible event states
2. Calculate recurrence values for the sequence
3. Create recurrence plot (and interpret)
4. Compute recurrence statistics (and interpret)

Recurrence basics

Example: A, B, A, B, B, B, B

Recurrence Quantification Analysis steps:

1. Define possible event states
2. Calculate recurrence values for the sequence
3. Create recurrence plot (and interpret)
4. Compute recurrence statistics (and interpret)

Define possible event states:

- Simple example: { Button A chosen, Button B chosen }

Recurrence basics

Example: A, B, A, B, B, B, B

Recurrence Quantification Analysis steps:

1. Define possible event states
2. Calculate recurrence values for the sequence
3. Create recurrence plot (and interpret)
4. Compute recurrence statistics (and interpret)

Define possible event states:

- Simple example: { Button A chosen, Button B chosen }
- The set of possible event states must have mutually exclusive and exhaustive events.

Recurrence basics

Example: A, B, A, B, B, B, B

Recurrence Quantification Analysis steps:

1. Define possible event states
2. Calculate recurrence values for the sequence
3. Create recurrence plot (and interpret)
4. Compute recurrence statistics (and interpret)

Define possible event states:

- Simple example: { Button A chosen, Button B chosen }
- The set of possible event states must have mutually exclusive and exhaustive events.
- Mutually exclusive, could not be { Button A chosen, Button A or B chosen }

Recurrence basics

Example: A, B, A, B, B, B, B

Recurrence Quantification Analysis steps:

1. Define possible event states
2. Calculate recurrence values for the sequence
3. Create recurrence plot (and interpret)
4. Compute recurrence statistics (and interpret)

Define possible event states:

- Simple example: { Button A chosen, Button B chosen }
- The set of possible event states must have mutually exclusive and exhaustive events.
- Mutually exclusive, could not be { Button A chosen, Button A or B chosen }
- Exhaustive, could not be { Button A chosen } alone

Recurrence basics

Example: A, B, A, B, B, B, B

Recurrence Quantification Analysis steps:

1. Define possible event states
2. Calculate recurrence values for the sequence
3. Create recurrence plot (and interpret)
4. Compute recurrence statistics (and interpret)

Define possible event states:

- Alternate example:
  { First trial,
  Chose Button A after choosing Button A on previous trial,
  Chose A after choosing B,
  Chose B after choosing A,
  Chose B after choosing B }

Recurrence basics

Example: A, B, A, B, B, B, B

Recurrence Quantification Analysis steps:

1. Define possible event states
2. Calculate recurrence values for the sequence
3. Create recurrence plot (and interpret)
4. Compute recurrence statistics (and interpret)

Possible event states:

- { Button A chosen, Button B chosen }

Recurrence basics

Example: A, B, A, B, B, B, B

Recurrence Quantification Analysis steps:

1. Define possible event states
2. Calculate recurrence values for the sequence
3. Create recurrence plot (and interpret)
4. Compute recurrence statistics (and interpret)

Calculate recurrence values for the sequence:

Recurrence basics

Example: A, B, A, B, B, B, B

Recurrence Quantification Analysis steps:

1. Define possible event states
2. Calculate recurrence values for the sequence
3. Create recurrence plot (and interpret)
4. Compute recurrence statistics (and interpret)

Calculate recurrence values for the sequence:

Using the definition of discrete recurrence:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & state_i = state_j \\ 0, & state_i \neq state_j \end{cases}$$

Recurrence basics

i	j	state$_i$	state$_j$	?	=	$r_{i,j}$
1	1	A	A	same	=	1

Example: A, B, A, B, B, B, B

Discrete recurrence:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & state_i = state_j \\ 0, & state_i \neq state_j \end{cases}$$

Recurrence basics

Example: A, B, A, B, B, B, B

Discrete recurrence:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & state_i = state_j \\ 0, & state_i \neq state_j \end{cases}$$

i	j	state$_i$	state$_j$	?	=	$r_{i,j}$
1	1	A	A	same	=	1
1	2	A	B	different	=	0
1	3	A	A	same	=	1

Recurrence basics

Example: A, B, A, B, B, B, B

Discrete recurrence:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & state_i = state_j \\ 0, & state_i \neq state_j \end{cases}$$

i	j	state$_i$	state$_j$	?	=	$r_{i,j}$
1	1	A	A	same	=	1
1	2	A	B	different	=	0
1	3	A	A	same	=	1
...	...	...	...	...		...
2	1	B	A	different	=	0
2	2	B	B	same	=	1
2	3	B	A	different	=	0
...	...	...	...	...		...
7	6	B	B	same	=	1
7	7	B	B	same	=	1

Recurrence basics

Example: A, B, A, B, B, B, B

Discrete recurrence:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & choice_i = choice_j \\ 0, & choice_i \neq choice_j \end{cases}$$

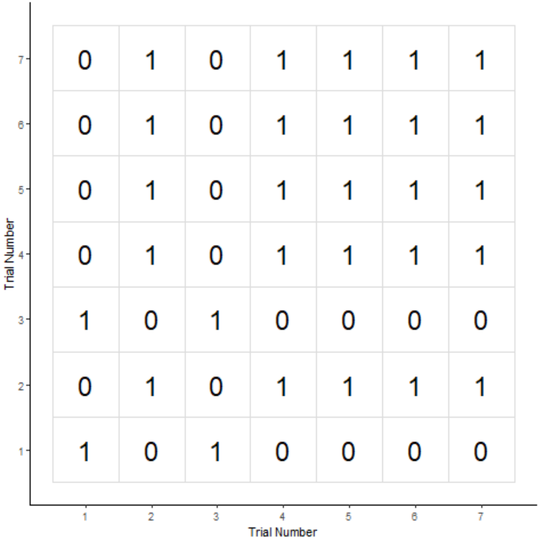
i	j	choice$_i$	choice$_j$	?	=	$r_{i,j}$
1	1	A	A	recurrent	=	1
1	2	A	B	not recurrent	=	0
1	3	A	A	recurrent	=	1
...	...	...	...	...		...
2	1	B	A	not recurrent	=	0
2	2	B	B	recurrent	=	1
2	3	B	A	not recurrent	=	0
...	...	...	...	...		...
7	6	B	B	recurrent	=	1
7	7	B	B	recurrent	=	1

How are recurrence plots constructed?

Example: A, B, A, B, B, B, B

Discrete recurrence:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & choice_i = choice_j \\ 0, & choice_i \neq choice_j \end{cases}$$

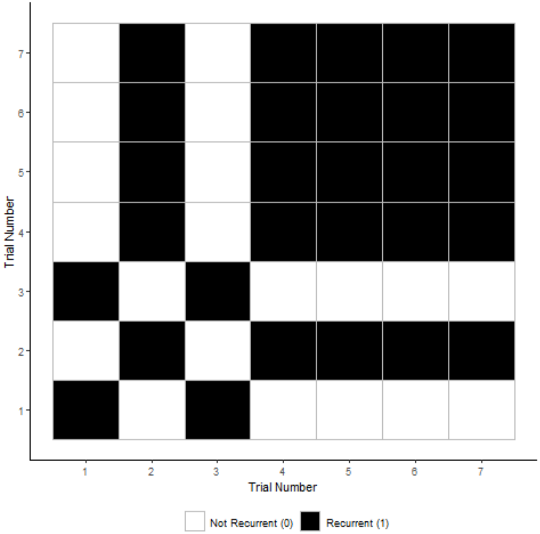


How are recurrence plots constructed?

Example: A, B, A, B, B, B, B

Discrete recurrence:

$$\mathbf{R}_{i,j} = \begin{cases} 1, & \mathit{choice}_i = \mathit{choice}_j \\ 0, & \mathit{choice}_i \neq \mathit{choice}_j \end{cases}$$

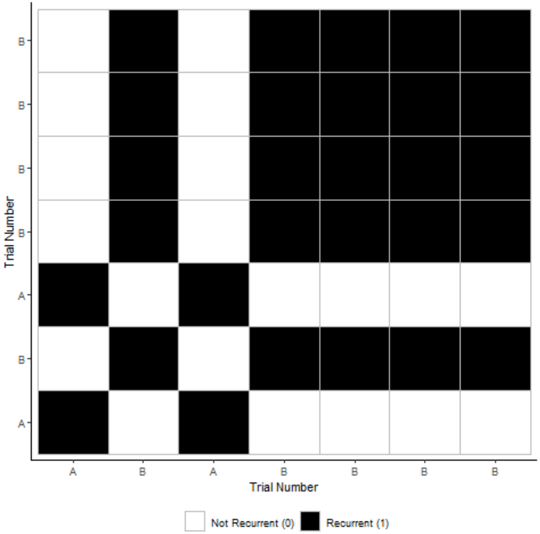


How are recurrence plots constructed?

Example: A, B, A, B, B, B, B

Discrete recurrence:

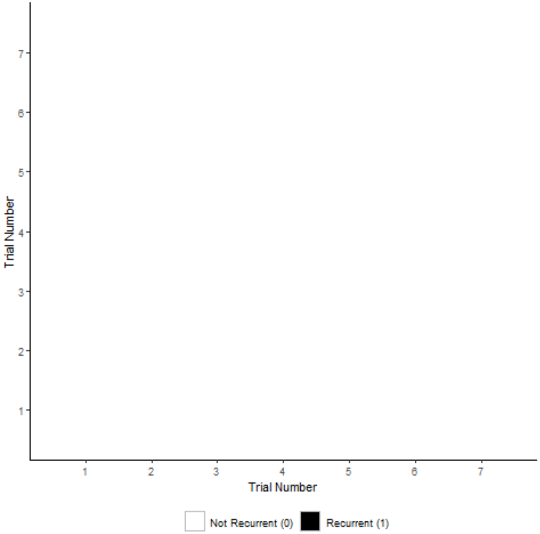
$$\mathbf{R}_{i,j} = \begin{cases} 1, & \mathit{choice}_i = \mathit{choice}_j \\ 0, & \mathit{choice}_i \neq \mathit{choice}_j \end{cases}$$



How are recurrence plots constructed?

To build our intuition, we can also build recurrence plots directly from a sequence

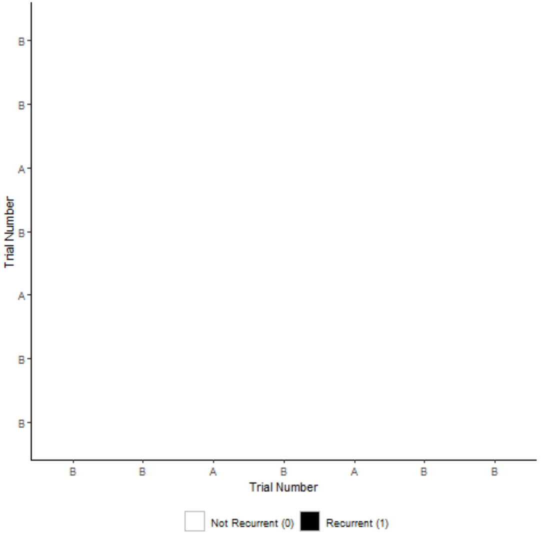
Second example sequence:
B, B, A, B, A, B, B



How are recurrence plots constructed?

To build our intuition, we can also build recurrence plots directly from a sequence

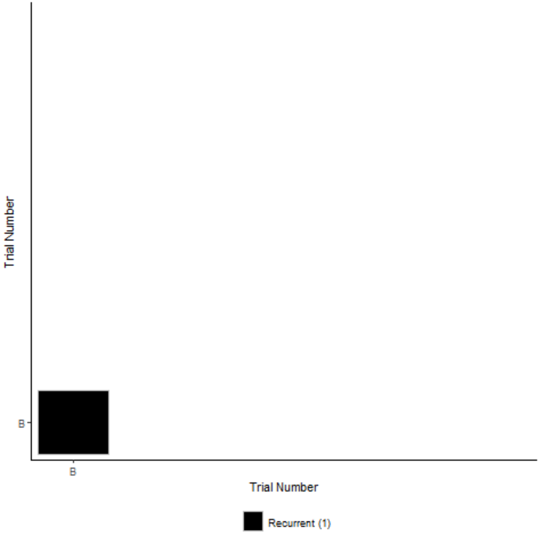
Second example sequence:
B, B, A, B, A, B, B



How are recurrence plots constructed?

To build our intuition, we can also build recurrence plots directly from a sequence

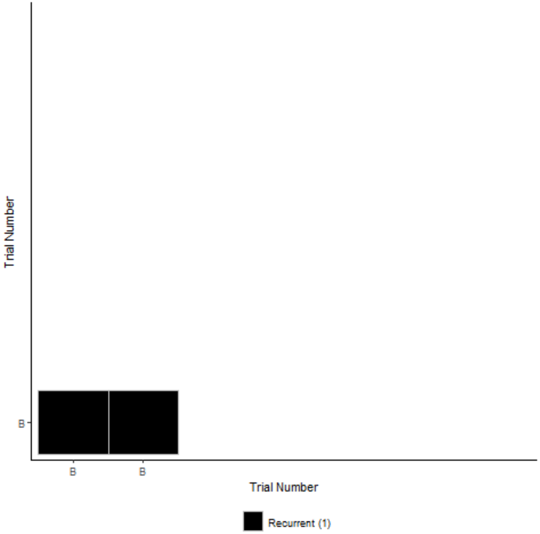
Second example sequence:
B, B, A, B, A, B, B



How are recurrence plots constructed?

To build our intuition, we can also build recurrence plots directly from a sequence

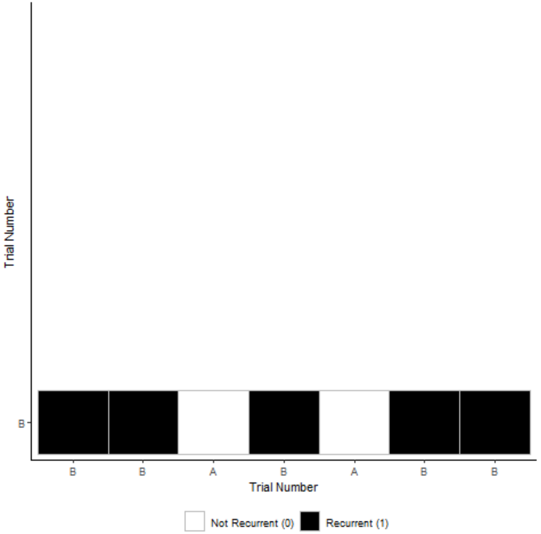
Second example sequence:
B, B, A, B, A, B, B



How are recurrence plots constructed?

To build our intuition, we can also build recurrence plots directly from a sequence

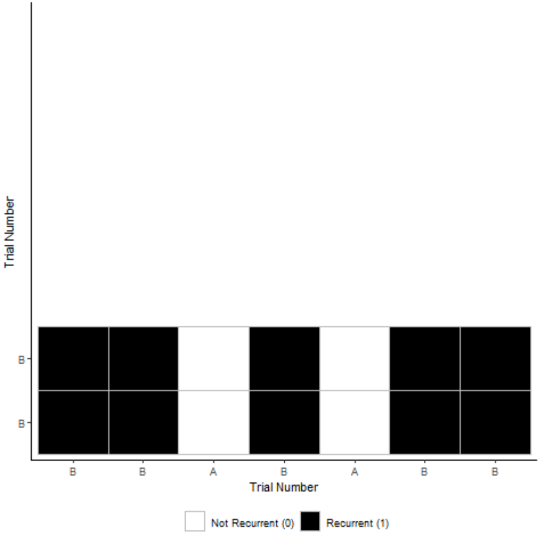
Second example sequence:
B, B, A, B, A, B, B



How are recurrence plots constructed?

To build our intuition, we can also build recurrence plots directly from a sequence

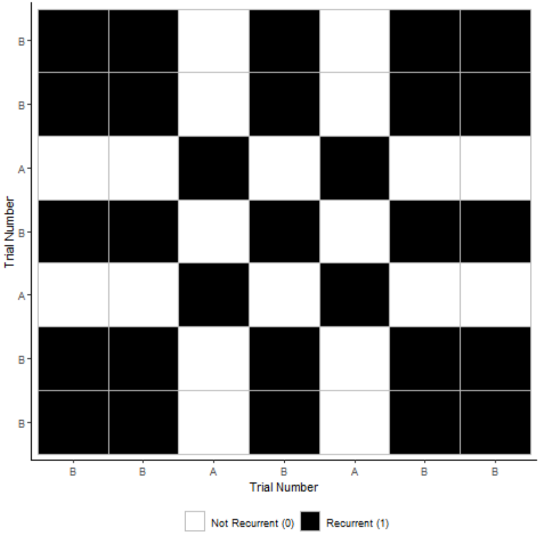
Second example sequence:
B, B, A, B, A, B, B



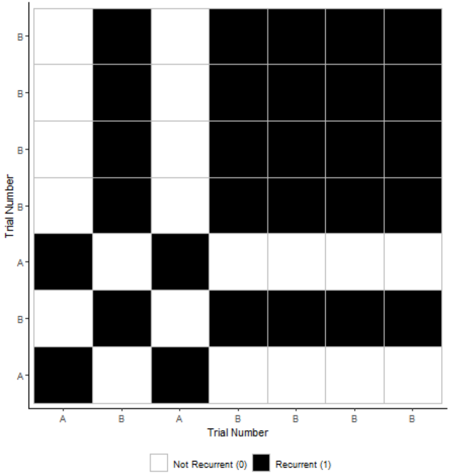
How are recurrence plots constructed?

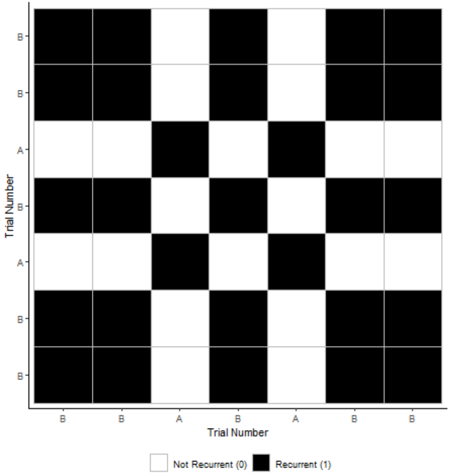
To build our intuition, we can also build recurrence plots directly from a sequence

Second example sequence:
B, B, A, B, A, B, B



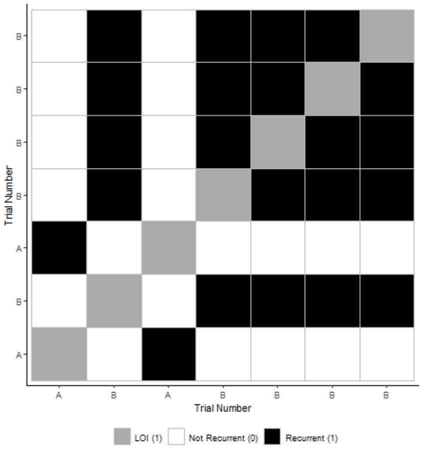
How are recurrence plots constructed?

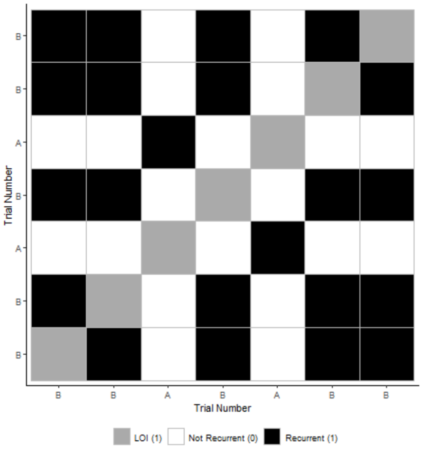




How are recurrence plots constructed?

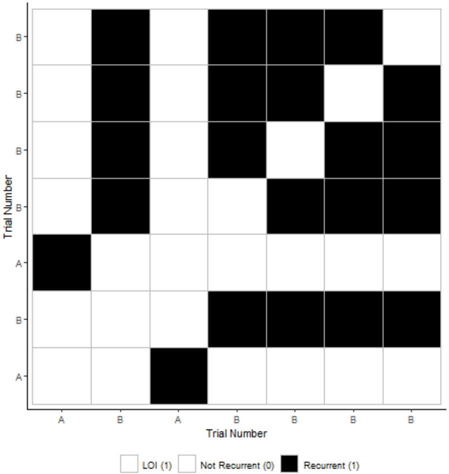
In auto-recurrence, comparing an element against itself is always recurrent, creating the Line of Incidence/Identity (LOI).

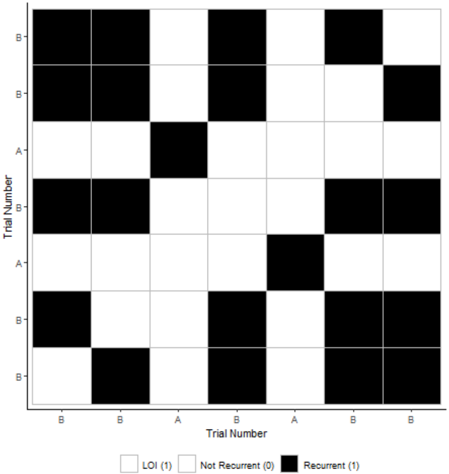




How are recurrence plots constructed?

Because this is redundant information, the LOI is often removed from auto-recurrence plots, and is considered implied.





Exercises

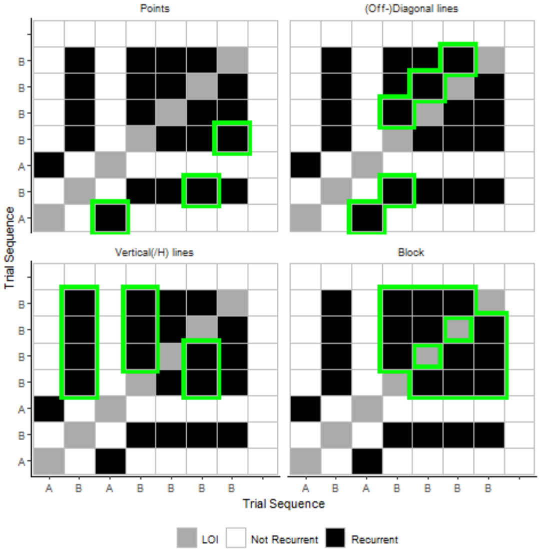
Check your understanding of how recurrence plots are constructed.

Complete Exercises 2.01 to 2.04

Basic recurrence plot features

Recurrence plots have four basic features:

- Points
- Diagonal Lines
- Vertical/Horizontal Lines
- Blocks

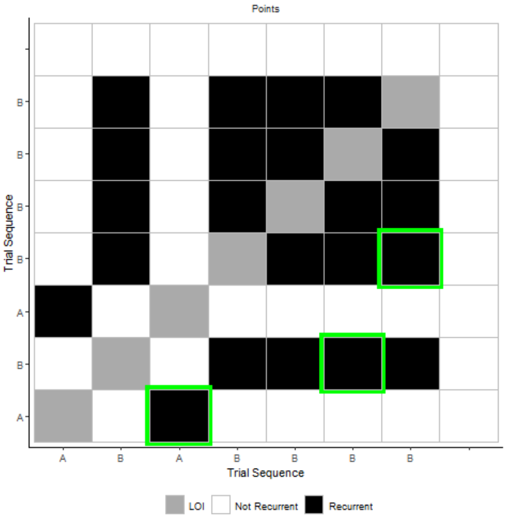


Basic feature: points

Recurrence points indicate a repeated event state.

Adjacent points combine to create lines and blocks.

Isolated points indicate rare or short-lived event states.

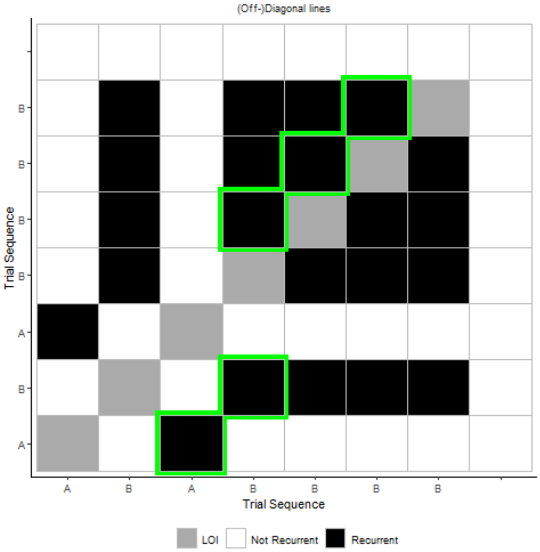


Basic feature: diagonal lines

Diagonal lines indicate a repeated sequence of event states.

Could be repeated event states:
AAAAAA

OR repeated sequences with differing event states:
ABCABCABCABAC

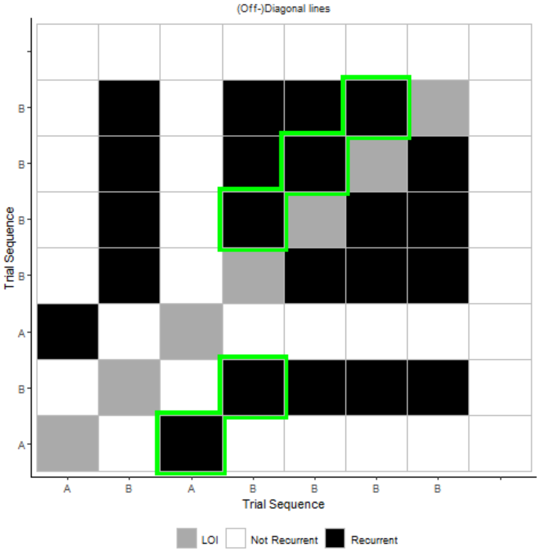


Basic feature: diagonal lines

Diagonal lines indicate a repeated sequence of event states.

Parallel to LOI indicates forward-evolving sequence.

Perpendicular to LOI indicates a backward evolving sequence.

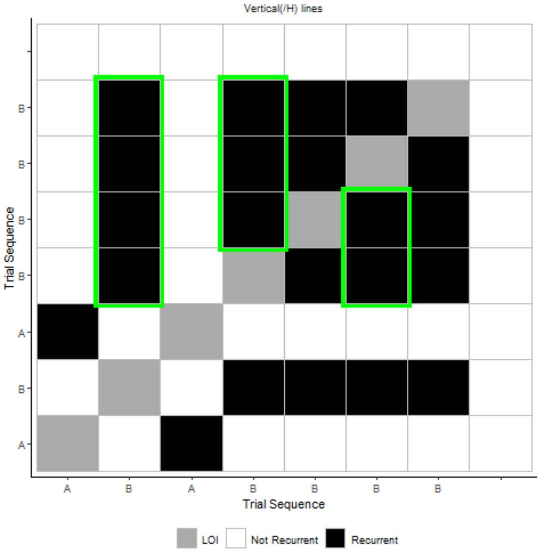


Basic feature: vertical/horizontal lines

Vertical/horizontal lines occur when the same event state repeats consecutively (AAAAA)

In auto-RQA, the plot is symmetric, so vertical and horizontal lines are equivalent.

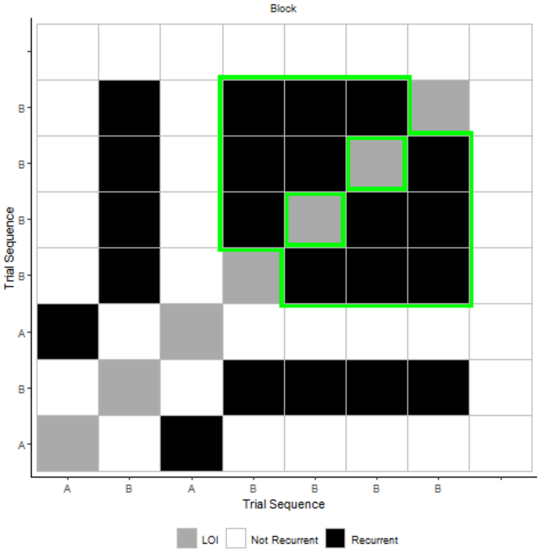
Indicates an event state does not change, or changes very slowly.



Basic feature: blocks

Blocks are formed only by repeated event states.

Easy visual feature to identify consecutive repeats in a sequence.



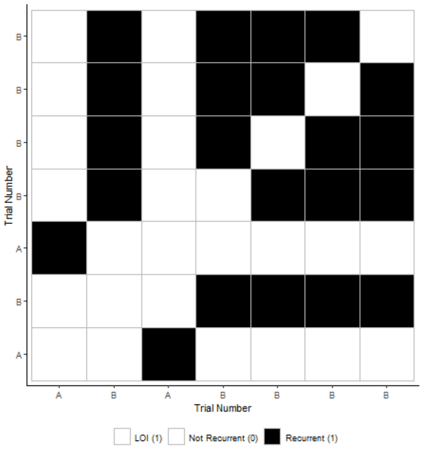
Exercises

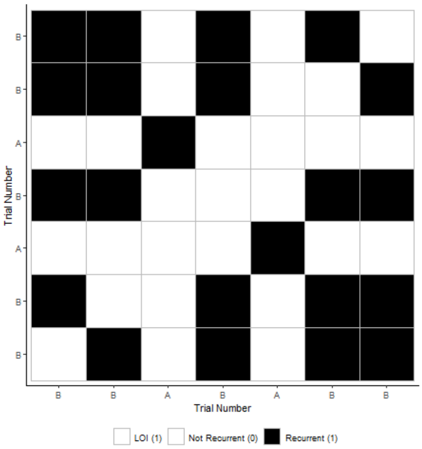
Check your understanding of the basic features of recurrence plots.

Complete Exercises 2.05 to 2.08

Recurrence plot basics: color encoding

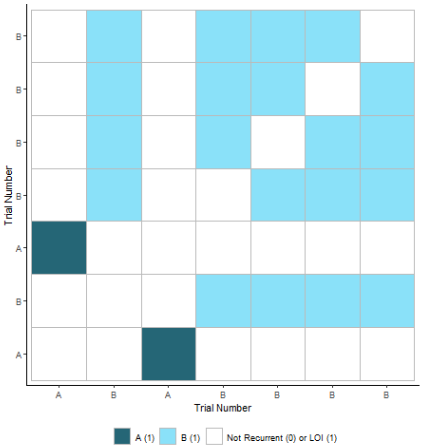
One additional helpful approach for interpreting recurrence plots is to add color by event state, instead of just using the color black.

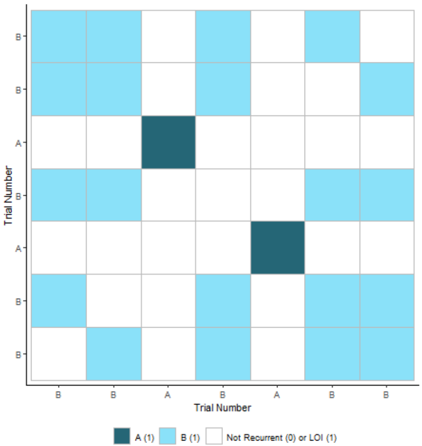




Recurrence plot basics: color encoding

One additional helpful approach for interpreting recurrence plots is to add color by event state, instead of just using the color black.





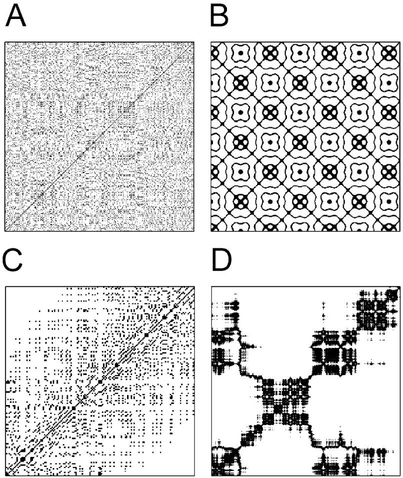
Question and answer session

Questions?

Higher-level visual features/patterns

Types of higher-level structures:

- A Homogeneous: uniform noise
- B Periodic: regular patterns
- C Drift: non-recurrent corners
- D White areas/white bands: sharp changes in state



From Marwan et al., 2007

Connecting features to decision strategies

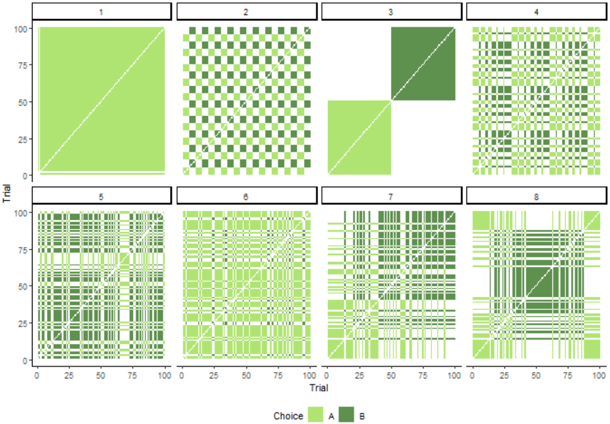
Let's think about several decision strategies in a task with 100 trials of a choice between Button A and Button B.

1. Explore both A and B, then stick with A
2. Alternate between A and B every 5 trials.
3. Choose A for first 50 trials, then switch to B for remaining trials.
4. Repeat an arbitrary pattern of choices.
5. Pick randomly between A and B with equal probability.
6. Pick randomly between A and B with an 75% chance of picking A.
7. Start preferring A, but have an increasing probability of picking B.
8. Pick A more often initially, then switch to preferring B, then switch back to A.

Connecting features to decision strategies

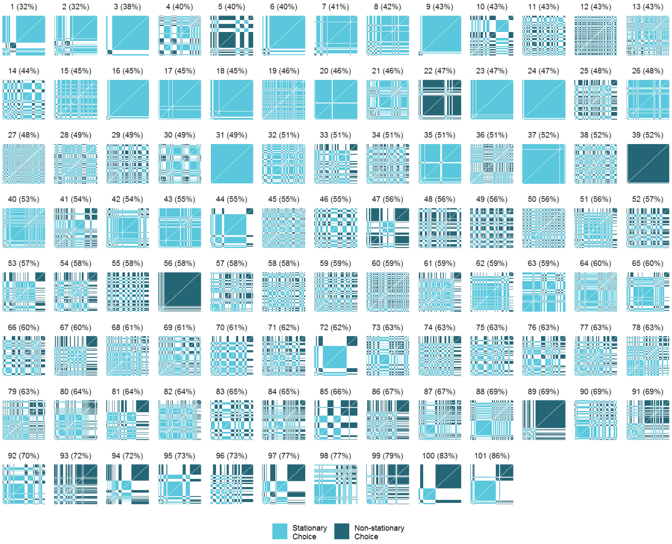
Here are the corresponding recurrence plots

1. A,B,A,A,A,A...
2. Alternate every 5 trials
3. A for first 50, B for remaining trials
4. Arbitrary pattern
5. Equal probability
6. 75% chance of A
7. Increasing probability of B
8. A initially, switch to B, then back to A



Connecting features to decision strategies

Data from actual human decision makers (McCormick, Cheyette & Gonzalez, in review)



Exercises

Check your understanding of connecting recurrence plots to decision strategies.

Complete Exercises 2.09 to 2.11

Question and answer session

Questions?

RQA Interpretation

3 Section 3
- Assessing Strategy Consistency with Event States
- Questions and answers
- Auto-RQA and Cross-RQA for assessing strategy consistency
- Questions and answers

Assessing Strategy Consistency with Event States

We may want to ask the question: how well does my choice sequence reflect a strategy that we think people use to complete the task under study?

There are two ways we can use event states to examine consistency of empirical data with a strategy of interest.

1. Encode each decision with its associated state of the strategy and compute RQA on that data coding.
   - This addresses the patterns of recurrence of different strategy states.
2. Encode each decision with binary adherence to the strategy value and compute RQA.
   - This addresses the overall level of consistency or compliance with a strategy.

Coding to Assess Strategy Dynamics

We refer to the *strategy dynamics* as the patterns of different strategy states captures by the choice sequence.

Example Strategy States:

- Maximizing Gamble Decisions: Choices might reflect $\{\text{Maximal Expected Value}, \text{Non-Maximal Expected Value}\}$
- Risky Decision Making: Choices might reflect $\{\text{High-Risk}, \text{Low-Risk}\}$
- Win-Stay, Lose-Shift: Choices might reflect $\{\text{Win-Stay}, \text{Win-Shift}, \text{Lose-Stay}, \text{Lose-Shift}\}$

Encoding strategy dynamics will likely require information about the task, such as expected trial outcomes/values or actual reward/value earned by the participant. This is in addition to the sequence of empirically observed choices.

Coding to Assess Strategy Dynamics

To encode data according to strategy states:

1. identify the strategy states of interest $m = 1, \ldots, M$
2. use the following state encoding scheme:

$$\mathbf{S}_i = \begin{cases} m_1, & choice_i \text{ is in strategy state } m_1 \\ m_2, & choice_i \text{ is in strategy state } m_2 \\ \vdots & \\ m_M, & choice_i \text{ is in strategy state } m_M \end{cases}$$

Coding to Assess Strategy Dynamics

To encode data according to strategy states:

1. identify the strategy states of interest $m = 1, \ldots, M$
2. use the following state encoding scheme:

$$\mathbf{S}_i = \begin{cases} m_1, & \textit{choice}_i \text{ is in strategy state } m_1 \\ m_2, & \textit{choice}_i \text{ is in strategy state } m_2 \\ \vdots & \\ m_M, & \textit{choice}_i \text{ is in strategy state } m_M \end{cases}$$

Choice	A	B	B	B	A	B
Outcome	-50	+60	-40	-40	+20	...
$\mathbf{S}_i$	Lose-Shift	Win-Stay	Lose-Stay	Lose-Shift	Win-Shift	...

Assessing Strategy Dynamics

Example: Participants in McCormick, Cheyette, & Gonzalez (in review) had to select between two gambles on each trial, and could lose (win 0) or win 500 points on each trial.

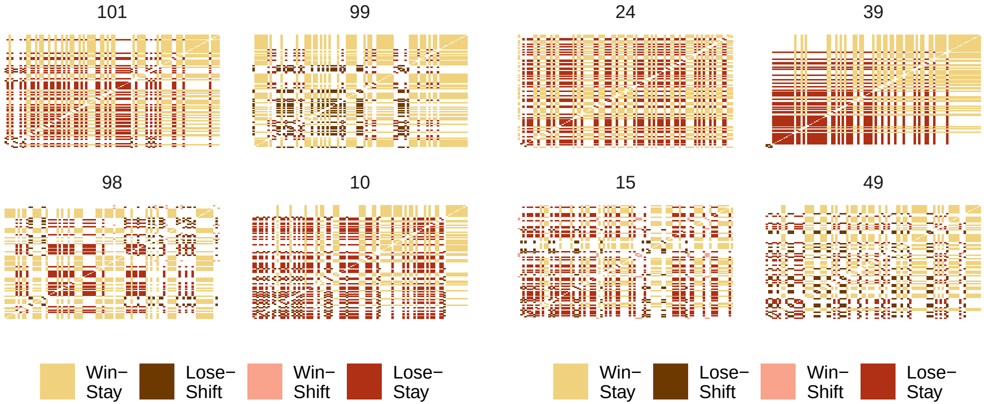
We code the choices according to Win-Stay, Lose-Shift (WSLS) strategy using the following 4 states:

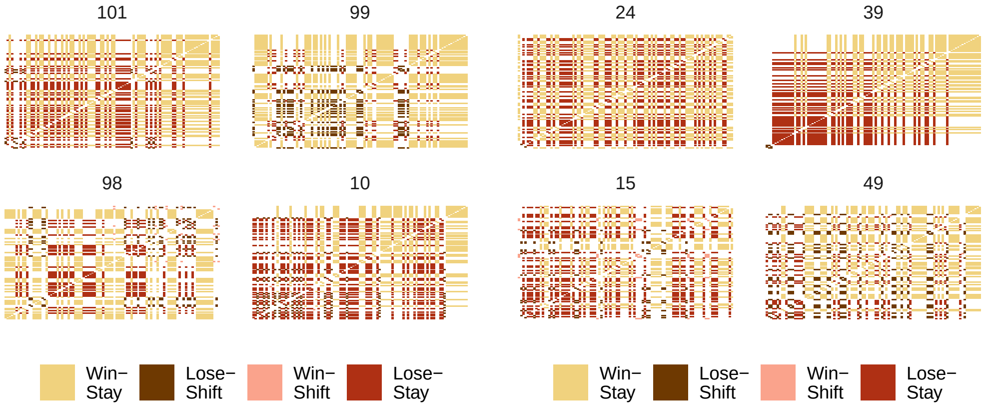
$$\textbf{Choice}_i^{WSLS-4} = \begin{cases} 1, & \textit{choice}_i \in \text{Win-Stay} \\ 2, & \textit{choice}_i \in \text{Lose-Shift} \\ 3, & \textit{choice}_i \in \text{Win-Shift} \\ 4, & \textit{choice}_i \in \text{Lose-Stay} \end{cases}$$

Then we compute RQA $\mathbf{R}_{i,j}^{WSLS-4}$ over $\textbf{Choice}_i^{WSLS-4}$ sequences using the discrete definition of recurrence.

Assessing Strategy Dynamics

Example: Participants in McCormick, Cheyette, & Gonzalez (in review) had to select between two gambles on each trial, and could lose (earn 0) or win (earn 500 points) on each trial.





Observations:

- All show frequent Win-Stay behavior and Win-Shift is infrequent
- Participant 99 shows recurrence for Lose-Shift and Win-Stay states
- There is a high recurrence of Lose-Stay behavior in many participants

Coding to Assess Adherence to a Strategy

We can also ask the question: what are the dynamics of compliance with or consistency with a strategy of interest?

To address this, we use the following binary state encoding scheme:

$$\mathbf{s}_i = \begin{cases} 1, & \textit{choice}_i \text{ is consistent with } \textit{strategy} \\ 0, & \textit{choice}_i \text{ is not consistent with } \textit{strategy} \end{cases}$$

Coding to Assess Adherence to a Strategy

We can also ask the question: what are the dynamics of compliance with or consistency with a strategy of interest?

To address this, we use the following binary state encoding scheme:

$$\mathbf{S}_i = \begin{cases} 1, & \textit{choice}_i \text{ is consistent with } \textit{strategy} \\ 0, & \textit{choice}_i \text{ is not consistent with } \textit{strategy} \end{cases}$$

Choice	A	B	B	B	A	B
Outcome	-50	+60	-40	-40	+20	...
Strategy State	Lose-Shift	Win-Stay	Lose-Stay	Lose-Shift	Win-Shift	...
$\mathbf{S}_i$	1	1	0	1	0	...

Assessing Strategy Adherence

Example: Consider again the participants in McCormick, Cheyette, & Gonzalez (in review).

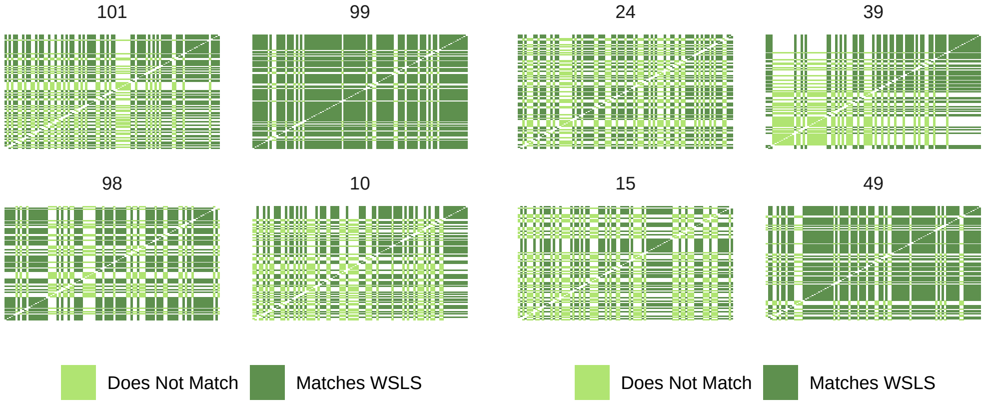
We define recurrence on whether each response was consistent with WSLS strategy:

$$\mathbf{Choice}_i^{WSLS} = \begin{cases} 1, & choice_i \in \text{Win-Stay or Lose-Shift} \\ 0, & choice_i \in \text{Win-Shift or Lose-Stay} \end{cases}$$

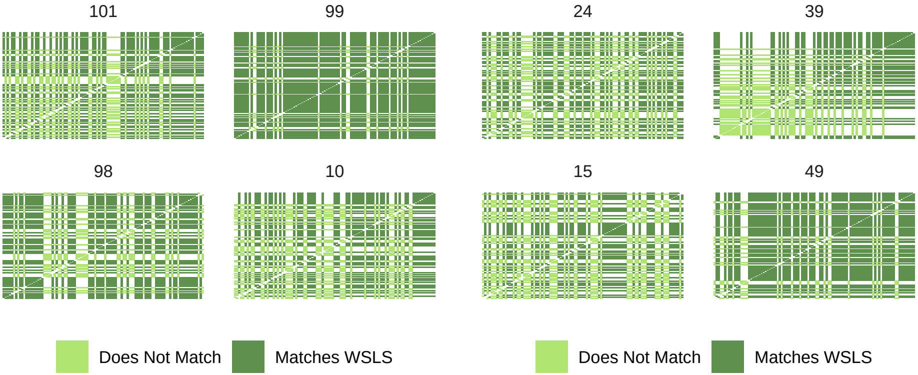
Then we compute RQA $\mathbf{R}_{i,j}^{WSLS}$ over $\mathbf{Choice}_i^{WSLS}$ sequences using the same discrete definition of recurrence.

Assessing Strategy Adherence

Example: Consider again the participants in McCormick, Cheyette, & Gonzalez (in review).



Assessing Strategy Adherence

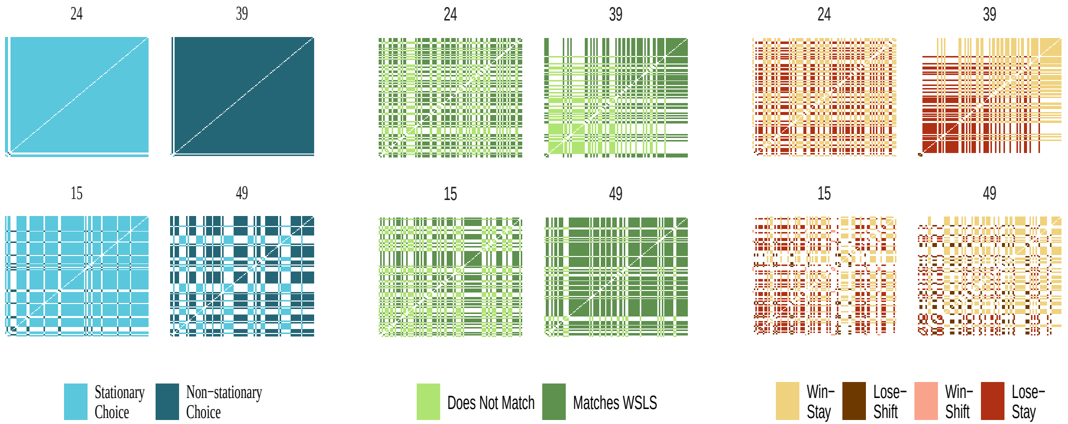


Observations:

- Participants with predominantly dark green RPs show heavy reliance on WSLS-consistent behavior (99, 98, 49)
- Participant 39 shows a pattern of early non-WSLS behavior that shifts to more compliance over the experiment

Assessing Strategy Consistency with Event States

We can begin to put various state encoding RPs together to get more insights into decision making strategies employed by people.



Exercises

Check your understanding.

Complete Exercises 3.01 - 3.04.

Question and answer

Questions?

Assessing Strategy Consistency: ARQA vs CRQA

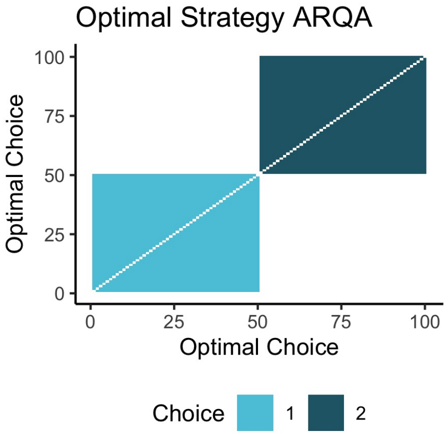
While Auto-RQA summarizes patterns within a single choice sequence, Cross-RQA directly compares two choice sequences.

To assess strategies with Cross-RQA (CRQA):

1. Generate or simulate the choice sequence representative of the strategy of interest.
2. Compute CRQA between the strategy sequence and the human sequence(s) of interest.
3. Plot.

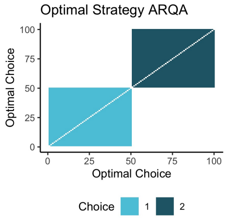
Assessing Strategy Consistency: ARQA vs CRQA

Example: The optimal strategy for these participants in McCormick, Cheyette, & Gonzalez (in review) was to make Choice A for the first 50 trials and Choice B for the second 50 trials.



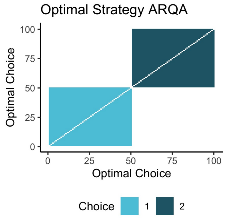
Assessing Strategy Consistency: ARQA vs CRQA

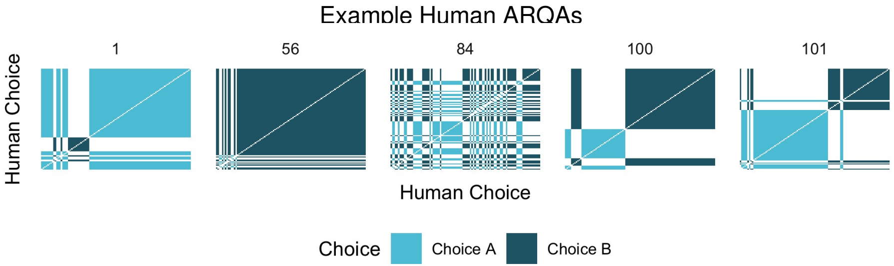
Example: The optimal strategy for these participants in McCormick, Cheyette, & Gonzalez (in review) was to make Choice A for the first 50 trials and Choice B for the second 50 trials.



Assessing Strategy Consistency: ARQA vs CRQA

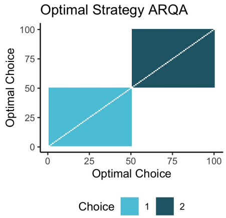
Example: The optimal strategy for these participants in McCormick, Cheyette, & Gonzalez (in review) was to make Choice A for the first 50 trials and Choice B for the second 50 trials.

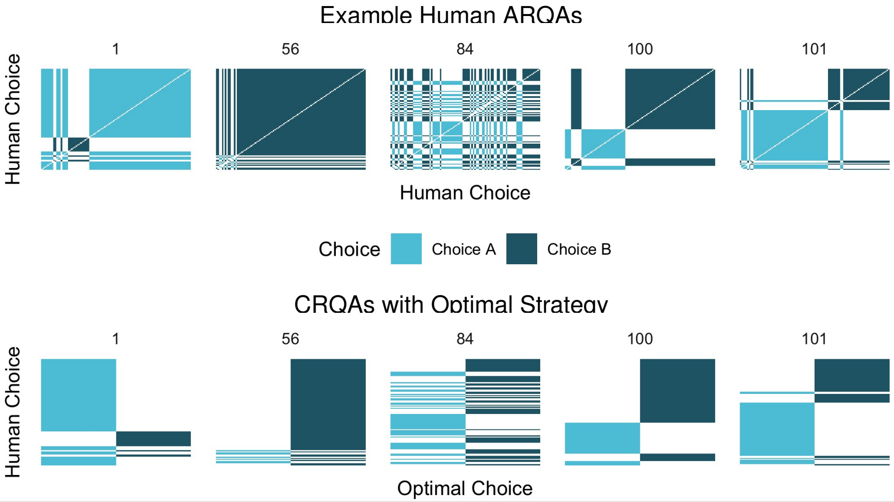




Assessing Strategy Consistency: ARQA vs CRQA

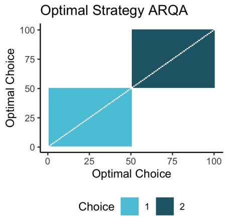
Example: The optimal strategy for these participants in McCormick, Cheyette, & Gonzalez (in review) was to make Choice A for the first 50 trials and Choice B for the second 50 trials.

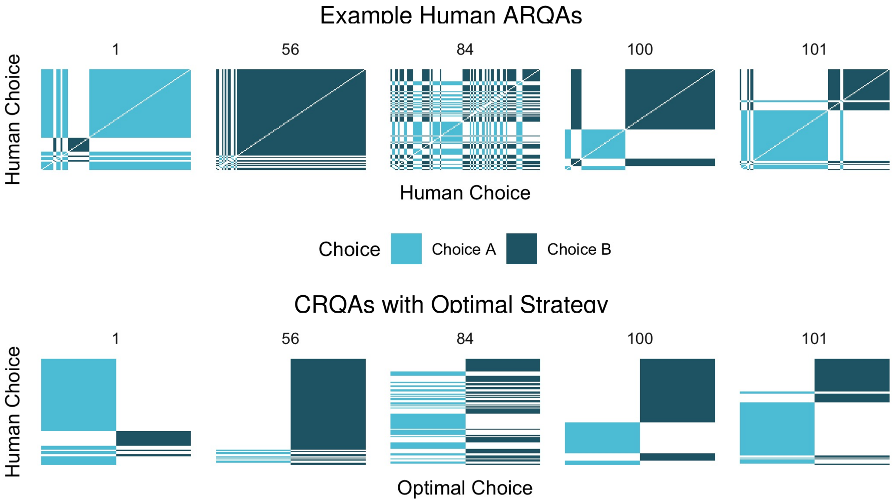




Assessing Strategy Consistency: ARQA vs CRQA

Note that the closer the human data is to the strategy of interest, the human-strategy CRQA recurrence plot will converge on the ARQA recurrent plot of the strategy.





Assessing Strategy Consistency: ARQA vs CRQA

Some additional considerations

- Statistics for empirical choice sequences highly similar to a strategy do not reach their maximum values.
  - They approach the RQA statistics of the strategy's ARQA.

Assessing Strategy Consistency: ARQA vs CRQA

Some additional considerations

- Statistics for empirical choice sequences highly similar to a strategy do not reach their maximum values.
  - They approach the RQA statistics of the strategy's ARQA.
- If you have a strategy that is more stochastic and so might exhibit different choice sequences for any given simulation,
  - Generate several samples
  - Compute CRQA for an empirical choice sequence against each simulated strategy sample
  - Examine the distributions of CRQA statistics
  - Looking for centrality of distributions to match the strategy ARQA statistics with low variance.

Exercises

Check your understanding.

Complete Exercise 3.05.

Question and answer

Questions?

RQA Interpretation

4 Section 4
- Recurrence Quantification Statistics
- Questions and answers

Recurrence quantification statistics

There are several well-defined, popularly used statistics for summarizing the variable $R_{i,j}$.

- Statistics are less subjective than visually inspecting recurrent plots.
- Summarize (aggregate) alternate pieces of information about the patterns of decisions (going beyond mean/rate, etc.)
- Statistics emphasize vertical and diagonal structures
- Variety of statistics are included in the *crqa* R package, and more in the literature (see Webber & Marwan, 2015)

Recurrence quantification statistics

There are several well-defined, popularly used statistics for summarizing the variable $R_{i,j}$.

- Statistics are less subjective than visually inspecting recurrent plots.
- Summarize (aggregate) alternate pieces of information about the patterns of decisions (going beyond mean/rate, etc.)
- Statistics emphasize vertical and diagonal structures
- Variety of statistics are included in the *crqa* R package, and more in the literature (see Webber & Marwan, 2015)

We can use them to

- quantify the recurrence of choices for a single person, model or strategy
- quantitatively describe a set of participants or models

Recurrence quantification statistics

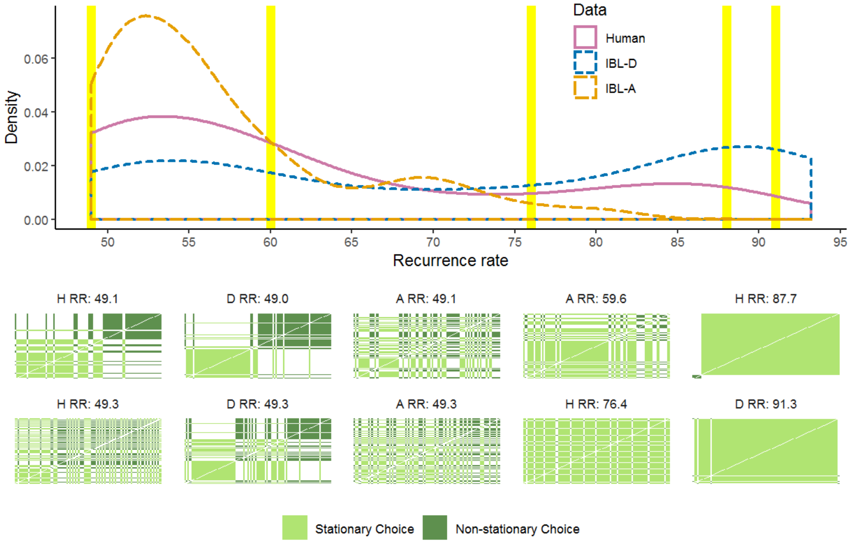
Abbr.	Statistic	Equation	Interpretation
RR	Recurrence Rate	$RR(\varepsilon, N) = \frac{1}{N^2 - N} \sum_{i \neq j = 1}^{N} R_{i,j}^{m,\varepsilon}$	Overall proportion of recurrent points in the RP
Statistics for Diagonal Line Structures			
$H_D(l)$	Frequency histogram of diagonal lines	$H_D(l) = \sum_{i,j=1}^{N} (1 - R_{i-1,j-1})(1 - R_{i+1,j+1}) \prod_{k=0}^{l-1} R_{i+k,j+k}$	
DET	Percent determinism	$DET = \frac{\sum_{l=d_{min}}^{N} l H_D(l)}{\sum_{i,j=1}^{N} R_{i,j}}$	Proportion of recurrent points falling into diagonal line structures
$D_{\max}$	Maximum diagonal length	$D_{\max} = \mathrm{argmax}_l H_D(l)$	Longest diagonal line length in the RP, indicating longest repeating sequence
$\langle D \rangle$	Average diagonal length	$\langle D \rangle = \frac{\sum_{l=d_{min}}^{N} l H_D(l)}{\sum_{l=d_{min}}^{N} H_D(l)}$	Average length of diagonal lines, indicating average length of repeating sequences
ENT	Shannon entropy of the frequency distribution of the diagonal line lengths	$ENT = - \sum_{l=d_{min}}^{N} p(l) \ln p(l)$ $p(l) = \frac{H_D(l)}{\sum_{l=d_{min}}^{N} H_D(l)}$	Complexity of the RP with respect to the diagonal line structures

Recurrence quantification statistics

Abbr.	Statistic	Equation	Interpretation
Statistics for Vertical Line Structures			
$H_V(l)$	Frequency histogram of vertical lines	$H_V(l) = \sum_{i,j=1}^{N}(1 - R_{i,j-1})(1 - R_{i,j+l})\prod_{k=0}^{l-1} R_{i,j+k}$	
LAM	Laminarity	$LAM = \frac{\sum_{l=v_{min}}^{N} lHv(l)}{\sum_{i,j=1}^{N} R_{i,j}}$	Proportion of recurrent points falling into vertical line structures, which indicate stating in the same state or repeating a choice
$V_{\max}$	Maximal length of vertical lines	$V_{\max} = \mathrm{argmax}_l H_V(l)$	Maximum vertical line length, indicting maximum number of sequential trials repeating the same choice
TT	Trapping time	$TT = \frac{\sum_{l=v_{min}}^{N} vH_V(l)}{\sum_{l=v_{min}}^{N} H_V(l)}$	Average length of the vertical line structures, indicting average number of sequential trials repeating the same choice

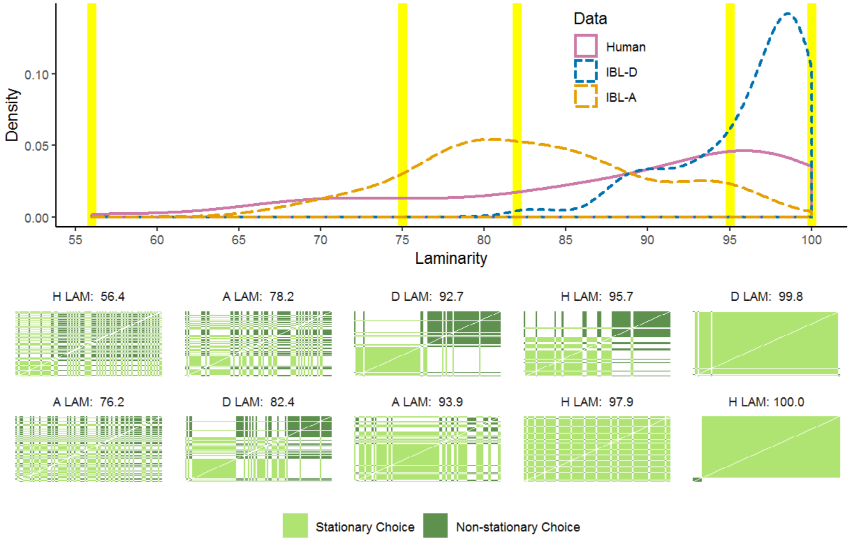
See Marwan & Webber (2015) for more technical details and additional statistics.

Statistics Examples: Recurrence Rates



Reproduced from McCormick, Blaha, Gonzalez (2020b)

Statistics Examples: Laminarity



Reproduced from McCormick, Blaha, Gonzalez (2020b)

Exercises

Check your understanding

Complete exercises 4.01 - 4.03

Question and answer session

Questions?

RQA Interpretation

5. Section 5
   - Creating recurrence plots and computing recurrence statistics in R
   - *arqahelper.R* functions and the *tidyverse* for multiple recurrence plots
   - Questions and answers

Recurrence plots and statistics in R

Please visit the documents *RQA_DemonstrationCode_2021-07-26.Rmd* and *RQA_DemonstrationCode_2021-07-26.html* to work through a series of demonstrations on functions available for computing Recurrence Plots and Recurrence Quantification Analysis Statistics.

We make available a function *arqahelper.R* that provides functions to leverage the *tidyverse* packages for RQA.

We rely heavily on the package *crqa* developed by Coco & Dale (2014) for the basic *crqa* function, the core of the RQA computations.

Here we summarize some of the material contained in *RQA_DemonstrationCode_2021-07-26.Rmd*.

Quick overview of RPs and RQA with *crqa*

The basic process for performing recurrence quantification analysis is:

1. Create a data set for your analysis with at least the following columns: Data Identifier (participant ID, strategy code, etc.), Trial Number or Order, Choice or State
2. Use the *crqa* or helper *arqa.map* or *crqa.map* functions to compute the desired ARQA or CRQA analysis. This outputs both the recurrence matrix and recurrence statistics
3. To plot with *tidyverse* tools, organize this output into a long-format tibble; we provide helper functions for this step
4. Plot the RPs
5. Evaluate or plot the associated RQA statistics.

arqahelper.R functions and the *tidyverse* for RPs

We provide mapper functions that use some pre-determined values for the *crqa* function that have proven useful for discrete choice sequence data:

```
source("arqahelper.r")
```

Of these helper functions, you may want to modify arqa.map() to modify the parameters sent to crqa() . (Future change planned: specify crqa() parameters in the call to calc.arqa.stats() .)

```
arqa.map
```

```
## function (seq)
## {
##     crqa(ts1 = seq, ts2 = seq, delay = 1, embed = 1, rescale = 0,
##         radius = 1e-04, normalize = 0, mindiagline = 2, minvertline = 2,
##         tw = 1)
## }
```

We also have the version for cross-recurrence analysis (not that the LOI is not removed in this case).

```
crqa.map
```

```
## function (seq1, seq2)
## {
##     crqa(ts1 = seq1, ts2 = seq2, delay = 1, embed = 1, rescale = 0,
##         radius = 1e-04, normalize = 0, mindiagline = 2, minvertline = 2)
## }
```

arqahelper.R functions and the *tidyverse* for RPs

We provide functions to split by any unique identifier variables (like participant ID) and compute the RQA for each unique value:

```
calc.arqa.stats
```

```
## function (thedf, seqkey.var, seqorder.var, seqevents.var)
## {
##     results <- thedf %>% group_by(!!as.name(seqkey.var)) %>%
##         arrange(!!as.name(seqorder.var)) %>% summarise(event.sequence = list(!!sym(seqevents.va
r))) %>%
##         ungroup() %>% mutate(crqa.object = purrr::map(event.sequence,
##         arqa.map), crqa.RR = unlist(purrr::map(crqa.object, function(x) {
##         x$RR
##     })), crqa.DET = unlist(purrr::map(crqa.object, function(x) {
##         x$DET
##     })), crqa.NRLINE = unlist(purrr::map(crqa.object, function(x) {
##         x$NRLINE
```

```
calc.crqa.stats
```

```
## function (thedf, seqkey.var, seqorder1.var, seqevents1.var, seqorder2.var = NA,
##     seqevents2.var)
## {
##     interim.df <- thedf %>% group_by(!!as.name(seqkey.var)) %>%
##         arrange(!!as.name(seqorder1.var)) %>% summarise(event.sequence1 = list(!!sym(seqevents1.v
ar))) %>%
##         ungroup()
##     if (is.character(seqevents2.var) & length(seqevents2.var) ==
##         1) {
##         if (is.na(seqorder2.var[1])) {
##             seq2.df <- thedf %>% group_by(!!as.name(seqkey.var)) %>%
##                 arrange(!!as.name(seqorder1.var)) %>% summarise(event.sequence2 = list(!!sym(seqe
vents2.var))) %>%
##                 ungroup()
##         }
##         else {
```

arqahelper.R functions and the *tidyverse* for RPs

We provide a function to organize the RQA results into a *tidyverse*-friendly format:

The function tidy.recurrence.matrices() is called on the object resulting from calc.arqa.stats() or calc.crqa.stats().
tidy.recurrence.matrices() creates ggplot-friendly recurrence plot data.

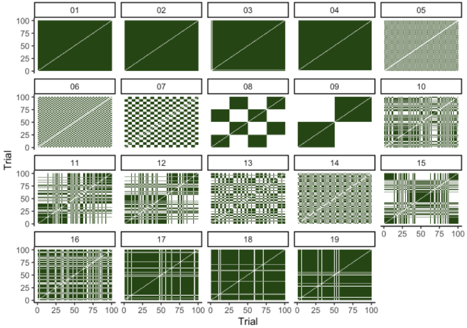
```
tidy.recurrence.matrices
```

```
## function (rqa.results, seqkey.var, rqa.stats = FALSE, key.info.df = NA,
##     key.info.var = NA)
## {
##     base.tidydf <- rqa.results %>% pull(crqa.RP.long) %>% bind_rows()
##     if (rqa.stats & is.na(key.info.var)) {
##         results.tidydf <- base.tidydf %>% left_join(rqa.results %>%
##             select(!!as.name(seqkey.var), crqa.RR:crqa.TT), by = c(seqkey = seqkey.var))
##     }
##     else if (rqa.stats & !is.na(key.info.var)) {
##         results.tidydf <- base.tidydf %>% left_join(rqa.results %>%
##             select(!!as.name(seqkey.var), crqa.RR:crqa.TT), by = c(seqkey = seqkey.var)) %>%
##             left_join(key.info.df, by = c(seqkey = key.info.var))
##     }
##     else if (rqa.stats == FALSE & is.na(key.info.var)) {
##         results.tidydf <- base.tidydf
##     }
##     else if (rqa.stats == FALSE & !is.na(key.info.var)) {
##         results.tidydf <- base.tidydf %>% left_join(key.info.df,
##             by = c(seqkey = key.info.var))
##     }
##     else {
```

arqahelper.R functions and the *tidyverse* for RPs

We can put these together to plot “quilts” or various desired plot formats or layouts:

```
arqa.ggplot.df %>%
  ggplot() + geom_tile(aes(x = x, y = y, fill = as.factor(recurrent))) +
  facet_wrap( ~ strategy, ncol = 5) +
  labs(#title = "Recurrence for choices by trial",
       x = "Trial", y = "Trial") +
  theme_classic() +
  theme(legend.position="none") +
  scale_fill_manual(name = "", values=c("1" = "#2b5013"))
```



arqahelper.R functions and the *tidyverse* for RPs

And we further leverage the easy tibble structure, where each RQA statistic is in its own column, to create kables or plots of the recurrence statistics:

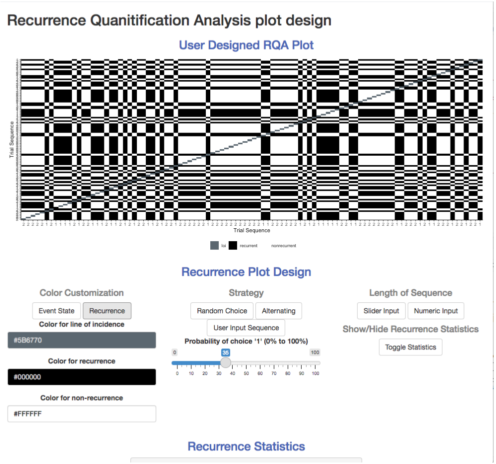
```
arqa.results %>%
  filter(strategy %in% c("01","02","03","04","05","13","16","17")) %>% # filter() used for a simplified example
  left_join(
    pidinfo.df,
    by = c("strategy" = "strategy")
  ) %>%
  select(strategy, choice2.rate, crqa.RR:crqa.TT) %>%
  knitr::kable(caption = "ARQA statistics for example choice patterns",
               digits = c(1,2,1,0,0,1,2,3,1,1),
               col.names = c("Strategy", "Choice 2 Rate", "RR",
                             "DET", "NRLINE", "maxL", "L", "ENTR",
                             "rENTR","LAM", "TT")) %>%
  kable_styling(bootstrap_options = c("striped", "hover"),full_width = F)
```

ARQA statistics for example choice patterns

Strategy	Choice 2 Rate	RR	DET	NRLINE	maxL	L	ENTR	rENTR	LAM	TT
01	0.00	99.0	100	196	99	50.50	4.585	1.0	100.0	50.5
02	1.00	99.0	100	196	99	50.50	4.585	1.0	100.0	50.5
03	0.01	97.0	98	192	97	49.50	4.564	1.0	99.0	49.8
04	0.99	97.0	100	194	98	50.00	4.575	1.0	100.0	50.0
05	0.50	49.0	100	98	98	50.00	3.892	1.0	0.0	NaN

R Shiny App

Also available in the zip is a Shiny app for interactively exploring recurrence plots:



Exercises

Try it yourself with the *RQA_DemonstrationCode_2021-07-26.Rmd* file!

Question and answer session

Questions?

6 Overall Question & Answer

Overall question and answer session

Questions for discussion?

References

- Abney, D. H., Warlaumont, A. S., Haussman, A., Ross, J. M. & Wallot, S. (2014). Using nonlinear methods to quantify changes in infant limb movements and vocalizations. *Frontiers in Psychology, 5*, 771.
- Anscombe, F. J. (1973). Graphs in Statistical Analysis. *The American Statistician, 27(1)*, 17-21. https://doi.org/10.1080/00031305.1973.10478966
- Bibyk, S. A., Blaha, L. M. & Myers, C. W. (2021). How packaging of information in conversation is impacted by communication medium and restrictions. *Frontiers in Psychology, 12*, 1096.
- Coco, M. I. & Dale, R. (2014). Cross-recurrence quantification analysis of categorical and continuous time series: An R package. *Frontiers in Psychology, 5*, 510.
- Dutt, V. & Gonzalez, C. (2012). The Role of Inertia in Modeling Decisions from Experience with Instance-Based Learning. *Frontiers in Psychology, 3*. https://doi.org/10.3389/fpsyg.2012.00177
- Eckmann, J.-P., Kamphorst, S. O. & Ruelle, D. (1987). Recurrence plots of dynamical systems. *EurophysicsLetters, 5*, 973-977.
- Gonzalez, C., Lerch, J. F., & Lebiere, C. (2003). Instance-based learning in dynamic decision making. *Cognitive Science, 27(4)*, 591–635. https://doi.org/10.1207/s15516709cog2704_2

References

- Konstantinidis, E., Harman, J. L., & Gonzalez, C. (under review). *Memory patterns for choice adaptation in dynamic environments.*
- Lejarraga, T., Dutt, V. & Gonzalez, C. (2012). Instance-based Learning: A General Model of Repeated Binary Choice. *Journal of Behavioral Decision Making, 25(2)*, 143–153. https://doi.org/10.1002/bdm.722
- Marwan, N., Carmenromano, M., Thiel, M., & Kurths, J. (2007). Recurrence plots for the analysis of complex systems. Physics Reports, 438(5–6), 237–329. https://doi.org/10.1016/j.physrep.2006.11.001
- Marwan, N. & Kurths, J. (2002). Nonlinear analysis of bivariate data with cross recurrence plots. *Physics Letters A 302.5*, 299-307.
- McCormick, E. N., Blaha, L. M. & Gonzalez, C. (2020a). Exploring Dynamic Decision Making Strategies with Recurrence Quantification Analysis. In *Proceedings of the 42nd Annual Cognitive Cognitive Science Society Meeting*, July 2020.
- McCormick, E. N., Blaha, L. M., & Gonzalez, C. (2020b). Analyzing variability in instance-based learning model predictions using recurrence quantification analysis. In *Proceedings of the Virutal MathPsych/ICCM 2021, mathpsych.org*, July 2020.

References

- Richardson, D. C. & Dale, R. (2005). Looking to understand: The coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. *Cognitive Science, 29*(6), 1045-1060.
- Richardson, M. J. & Johnston, L. (2005). Person recognition from dynamic events: The kinematic specification of individual identity in walking style. *Journal of Nonverbal Behavior, 29*(1), 25-44.
- Shockley, K. & Turvey, M. T. (2005). Encoding and retrieval during bimanual rhythmic coordination. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 31*(5), 980.
- Wallot, S. & Leonardi, G. (2018). Analyzing multivariate dynamics usingcross-recurrence quantification analysis (CRQA), diagonal-cross-recurrence profiles (DCRP), and multidimensional recurrence quantification analysis (MDRQA)—a tutorial in R. *Frontiers in Psychology, 9*, 2232.
- Webber, C. L. & Marwan, N. (2015). *Recurrence quantification analysis. Theory and Best Practices.* New York: Springer.
- Webber Jr, C. L. & Zbilut, J. P. (2005). Recurrence quantification analysis of nonlinear dynamical systems. *Tutorials in contemporary nonlinear methods for the behavioral sciences*, 26-94.