**Lab 5: Gause Paramecium Growth Exercise - Instructions**

See PowerPoint file associated with Lab 5 for introduction

Open Gause Paramecium Growth Excel file

Click on Spreadsheet: Gause Data

Spreadsheet contains Day and Raw N columns (filled out) plus blank ln N column

**Intro (Power Point):**

The goal of this exercise is to create a model of density dependent growth for Paramecium data that we have from the textbook

Three questions we will address during this process:

What type of density dependent population growth model fits the Paramecium data?

What are the important parameters we need for this model?

How can we estimate those parameters?

**Step 1**: Make figure of Paramecium population growth

Make figure of Population Size (y-axis) vs. Time (days) (x-axis)

Highlight both columns (Day and Raw N)

Go to: Insert tab, Scatterplot

Clean up nonsense on figure

Click on chart – Go to Layout tab

Add y-axis title: Population Size

Add x-axis title: Time (days)

🡪 Our figure looks a lot like figure from textbook

🡪 Looks like scramble density dependence

**Step 2**: Estimating carrying capacity

Question: Given these data, what do you think (roughly) carrying capacity is?

We can estimate carrying capacity by taking the average population size of the later points

Looks like carrying capacity is hit by around day 10

Take average of population size from day 10-24 🡪 **K = 534**

Plug number into pre-made box

Calculate SD of carrying capacity as: stdev.s(*N from days 10-24*) 🡪 **SD = 34**

**Step 3**: Linearizing data

Question: Given these data, how can we estimate λmax?

Similar to before when we did not have density dependence

🡪 Get slope from plot of lnN vs. time

Creating column of lnN

Fill out data under column heading: lnN

Create formula: =ln(*cell to left*)

Copy down column

Plot ln(N) vs. time

Highlight Day column; press Ctrl; Highlight lnN column

Go to: Insert tab, Scatterplot

Clean up nonsense on figure

Add y-axis title [ln(Population Size)] and x-axis title [Time (days)]

Interpreting Graph of lnN vs. time

Doesn’t look very linear… not surprising given density dependence

We only need to figure out λmax, though…

Just need data at start of population growth, when it looks more linear

🡪 Let’s restrict the data to just the first 8 points

Reduce the data being plotted to first 8 points

Click on points on figure

Change the selection boxes for data in figure to just time 0 to 7

🡪 Data do not look all that linear just yet

Reduce the data being plotted to first 5 points

Click on points on figure

Change the selection boxes for data in figure to just time 0 to 4

🡪 Looks a lot more linear

**Step 4**: Estimating λmax

To get slope, we can add a trendline to data

Click on datapoints

Go to: Layout tab; select Trendline button; select More Trendline Options

🡪 Check boxes for “Display equation on chart” and “Display R-squared value on chart”

Slope = 0.8507

Plug slope into pre-made box

Calculate λmax in pre-made box as: =exp(*slope cell*)

🡪 **λmax = 2.3**

Know **SD of λmax is about 1.0**

R2 = 0.99 🡪 Really good fit!

***🡪 Excel Graphing***

**Step 6**: Graphing Ramas data in Excel

Go over to Excel

Click on Gause Data Ramas spreadsheet

Make a figure with Raw and Ramas data

Highlight Day, Raw N, and Ramas Columns

Go to: Insert tab, Scatterplot

Excel defaults to use left-hand column as x-axis

and other two columns as two separate y-trajectories

Remove lines from figure

Keep legend for now

Add y-axis title: Population Size

Add x-axis title: Time (days)

Rename the data series:

Go to: Design, Select Data

Click on Series 1, Edit 🡪 Rename to “Raw”, Click OK

Click on Series 2, Edit 🡪 Rename to “Ramas”, Click OK

Click OK

Show how to change size of markers (right click on trajectory, select Format Data Series)

Show how to add lines between markers (helpful for lab reports)

🡪 Our Ramas model hits carrying capacity sooner and does not fluctuate

🡪 Despite this model, fluctuations are common to have for Scramble competition

🡪 You will see in lab today how the relatively low λmax value keeps our model from fluctuating