

# Topic: for Logistic Regression

 Tricks for  **sas** Users



# Welcome to Today's Lecture!

## Today's Lecture

- SAS is good at logistic regression
- But you can do a lot of fun things in R with it as well!
- SAS pain points:
  - Not obvious how to export \*.csv of ORs and CIs
  - No easy coefficient plots
- R to the rescue!



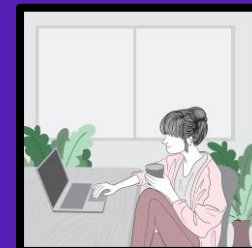
Try all of our career-building data science resources!



Join our group online data science mentoring program:  
<https://buff.ly/3Sw4Fsa>



Take our exclusive online data science courses aimed healthcare analysts:  
<https://monika-s-school-e295.thinkific.com>



Watch our tutorials on YouTube:  
<https://buff.ly/2W6ijuc>

Take our LinkedIn Learning courses: <https://buff.ly/2lhd4Rq>



# What is Regression like in R?

- R GUI vs. RStudio – I use R GUI
- In **SAS**, once you prepare the dataset, you run PROC LOGISTIC with options (or PROC GENMOD...)
  - Options set to give you ORs and 95% CIs, statistical tests, graphs
- In **R**, the whole thing is a fragmented affair! There are different ways of accomplishing your goal.
- I will demonstrate one approach I use a lot

## Today's Resources

Blog post about this demonstration:

<https://buff.ly/3QILvl6>

Download demonstration **dataset** from GitHub: <https://buff.ly/3E4VzgV>

It's the last one in the folder named *BRFSS\_j.rds*

Download the demonstration **code** from GitHub:

<https://buff.ly/3Laxp8P>



# Summary of My Approach

1. Prepare analytic dataset; specify candidate independent variables and dependent variable
2. Use *glm* to create an object in R (to create the logistic regression equation – on the log scale)
3. Using *devtools* and *broom* packages, run *tidy* on that object to create a tidy object in R (to make the results into a dataframe you can edit).
4. Using data editing commands, edit the new tidy dataframe with the logistic regression results so you can create the OR from the log OR for each slope, and calculate the confidence limits.
5. Export the results as a \*.csv, and now it is easy to use them in another program (e.g., for graphing).
6. **BONUS:** Using the original *glm* object, I used the command *coefplot* from the package *arm* to make a coefficient plot!



# Questions?

**Facilitator:** Monika M. Wahi, MPH, CPH

Email:  
[dethwench@gmail.com](mailto:dethwench@gmail.com)

Blog:  
<https://dethwench.com/blog/>

LinkedIn:  
<https://www.linkedin.com/in/dethwench/>



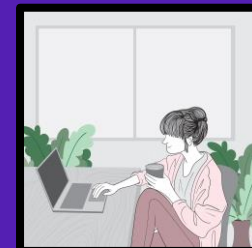
Try all of our career-building data science resources!



Join our group online data science mentoring program:  
<https://buff.ly/3Sw4Fsa>



Take our exclusive online data science courses aimed healthcare analysts:  
<https://monika-s-school-e295.thinkific.com>



Watch our tutorials on YouTube:  
<https://buff.ly/2W6ijuc>

Take our LinkedIn Learning courses: <https://buff.ly/2lhd4Rq>