Fall 2023

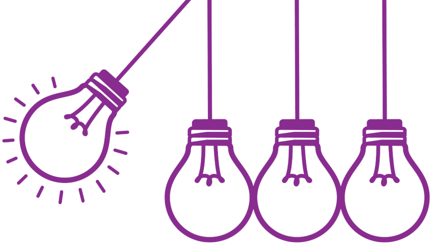
DIME Analytics

# REPRODUCIBLE RESEARCH FUNDAMENTALS

THE WORLD BANK
IBRD • IDA | WORLD BANK GROUP

i2i DIME
TRANSFORM DEVELOPMENT

# Cleaning Secondary Data



Reproducible Research Fundamentals

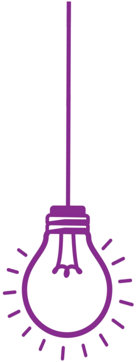
September 26, 2022

Development Impact Evaluation (DIME)
The World Bank

- During the training, find all materials in our shared OneDrive: here

WORLD BANK GROUP

i2i DIME TRANSFORM DEVELOPMENT



# Tailored Strategies for Data Types

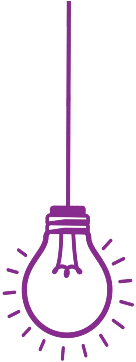
# Comparing Primary and Secondary Data

**Primary Data**

- Custom-made
- Current insights*
- Time-consuming
- Can be expensive

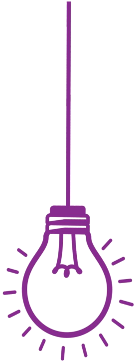
**Secondary Data**

- Economical
- Broad database
- Potential misalignment
- Quality concerns
- Immediate



# Data

- **Colombia's Connectivity - Decleaned**
  - File: `colombia_connectivity_decleaned.csv`
  - Source: Ookla and Humanitarian Data Exchange
  - Description: This file is a modified version of the cleaned Colombia connectivity dataset. It has been ”decleaned” to reintroduce common sources of error, providing a realistic set for this cleaning exercise.



# Exercise

# Exercise

Apply the tasks you've learned in the last few sessions to colombia_connecivity_decleaned.csv.

1. **Data cleaning:** Clean the dataset using a script.
   1.1 Check for data collection metadata variables not needed for analysis and drop them (id_test_data)
   1.2 Make sure there is one or more identifying variables in the data
   1.3 Make sure each variable has a correct data type
   1.4 Handle missing values appropriately using packages like `haven` or function `is.na()`
   1.5 See if there are any special characters in the data and remove them. You can use `stri_trans_general()`
   1.6 Check that all variables have a label in the working language of your team (assume it's English for this exercise)

# Exercise (continued)

2. **Documenting metadata:** Create or export a codebook or data dictionary of your cleaned dataset
3. **Documenting data cleaning and consistency:**
    - Document all the data cleaning tasks and the changes you apply to the dataset
    - Review all the variables and check that they have consistent values. Document your checks and any anomalies you consider important for next stages.

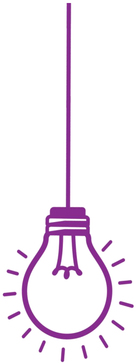
# Importance in Social Research

**Benefits of Integration:**

- Richer analysis
- Enhanced reliability
- Cost-effective

**Tailoring the Cleaning Process:**

- Ensures data reliability
- Prevents misinformation
- Facilitates valid conclusions



# Hints for data cleaning

# Unique ID

Commands for testing that a variable is uniquely and fully identifying

In R:

- `distinct()`
- `is.na()`
- `unique()`
- `length()`
- `dim()`

# Identifying and Handling Missing Values in R

- Understanding the nature and pattern of missing data
- Using ‘naniar‘ package for visualizing missing data
- Employing methods like mean imputation, regression imputation, etc. for handling missing data

# Variable Labeling and Encoding in R

- Here function `glimpse()` can be useful, to understand the data type of each variable within the dataframe. You should run it for every dataset you encounter.
- Leveraging ‘labelled‘ package for labelling variables
- Utilizing ‘forcats‘ package for working with categorical variables

# (Not) Renaming variables

- Do not change the names of variables.
- Renaming variables will make it harder to find.

# Working with Date and Time in R

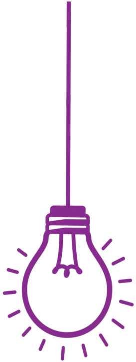
- Understanding ‘Date‘ and ‘POSIXct‘ classes in R
- Using ‘lubridate‘ package for easier date-time manipulation

# Saving files

- During the data cleaning process, you might have saved multiple intermediate files, for example if you cleaned long modules separately to make your code more readable
- After cleaning your data and merging it back together, you'll want to save a final cleaned data set, containing all variables you will use in the analysis
- This new data set will probably be quite heavy. Use `compress` to save your variables in the most economic format

# Naming files

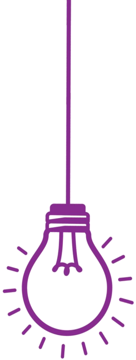
- Make sure all output files, datasets and others are clearly and uniquely labeled, i.e.: “desc_stats_tmt_only.xls” “input_plan_adm_data.dta”
- It’s often desirable to have the names of your data sets and do-files linked, so it is easy to understand which do-files is creating which data set, such as “merge.do” and “merged.dta” or “cleaning.do” and “clean.dta”
- Do not use _v1, _v2 etc. for any final files. This leads to bugs in do-files that depend on these files when a new versions is added.
- It’s ok to use _v1, _v2 etc. for old versions of files if you really need to keep an archive



# Hints for metadata documentation

# Documenting metadata

- Variable labels must be short and self-explanatory, as they will be used in tables and graphs
- However, there is much more information that is useful for someone opening the data for the first time
- This information should be stored in a data dictionary/codebook, including
  - The definition of each variable or corresponding survey question
  - The number of missing observations in each variable
  - Summary statistics
  - Any field notes or corrections made to each variable
- You can use ‘set_variable_labels‘ from ‘labelled‘.



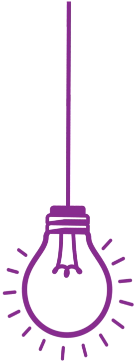
# Hints for data cleaning documentation and data consistency

# Documenting data cleaning

- Describe in order the data cleaning tasks you're doing. Use the working language of your team
- Even if you don't edit the dataset after a task (for example, there might not be duplicated entries in your data), it's a good practice to document the task and note that no changes were implemented

# Check variables consistency

- Check that values are consistent across variables
- For example, if an individual is male, then he cannot be pregnant
- This kind of inconsistency should usually be corrected during the high-frequency checks, but often times there’s no time when the enumerators are in the field to identify and correct all of them
- So if you find any issues, create flag variables that identify observations with inconsistent values



# Utilizing R Packages for Data Cleaning

## Useful R Commands and Packages

- ‘summary‘, ‘table‘, ‘count‘ for basic data summary
- ‘assert_that‘ for validation checks
- ‘skimr‘ for easy and fast data summarization
- ‘tidyverse‘ for a collection of R packages designed for data science that are really useful when cleaning data