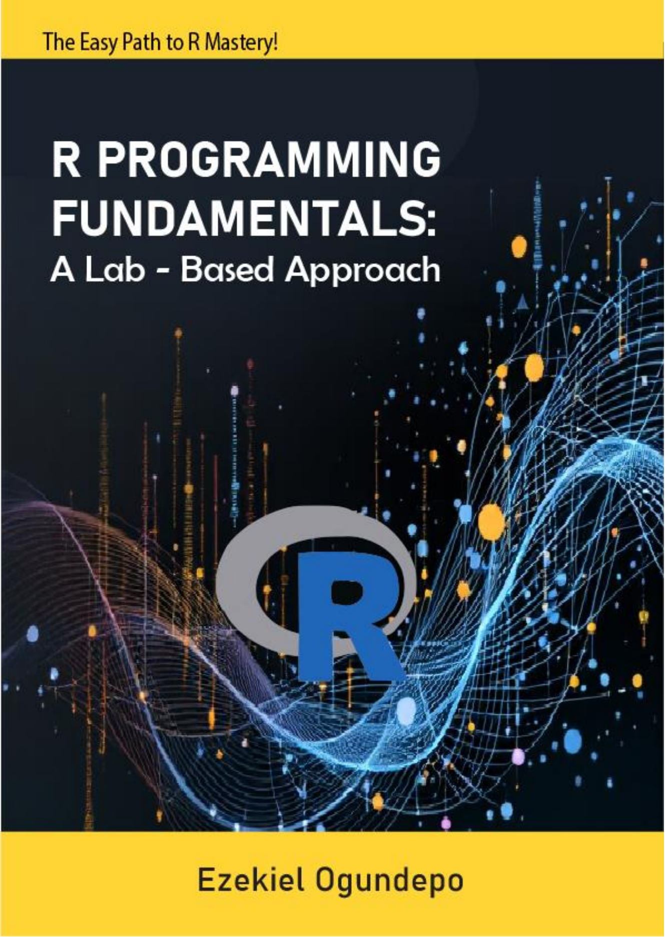
REPRODUCIBLE WORKFLOWS With

RStudio Project, renv, and Version Control

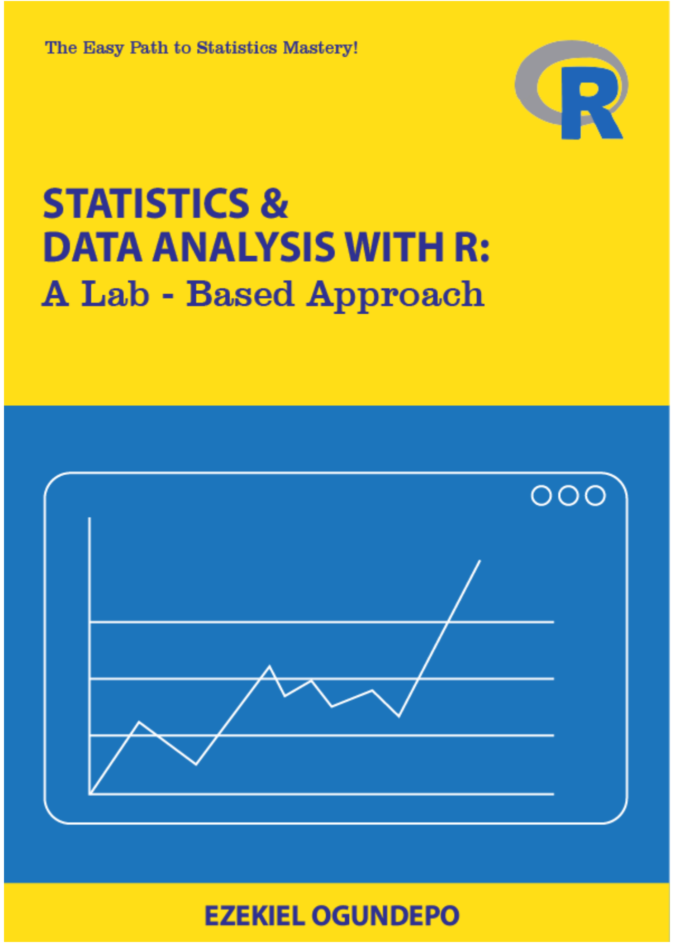
GBG Analytics



The Easy Path to R Mastery!

R PROGRAMMING FUNDAMENTALS:
A Lab - Based Approach

Ezekiel Ogundepo



The Easy Path to Statistics Mastery!

STATISTICS & DATA ANALYSIS WITH R:
A Lab - Based Approach

EZEKIEL OGUNDEPO

https://bit.ly/preorder-gbgbook

Learning Objectives

The goal of this session is to help you develop essential skills in creating a reproducible workflow using version control.

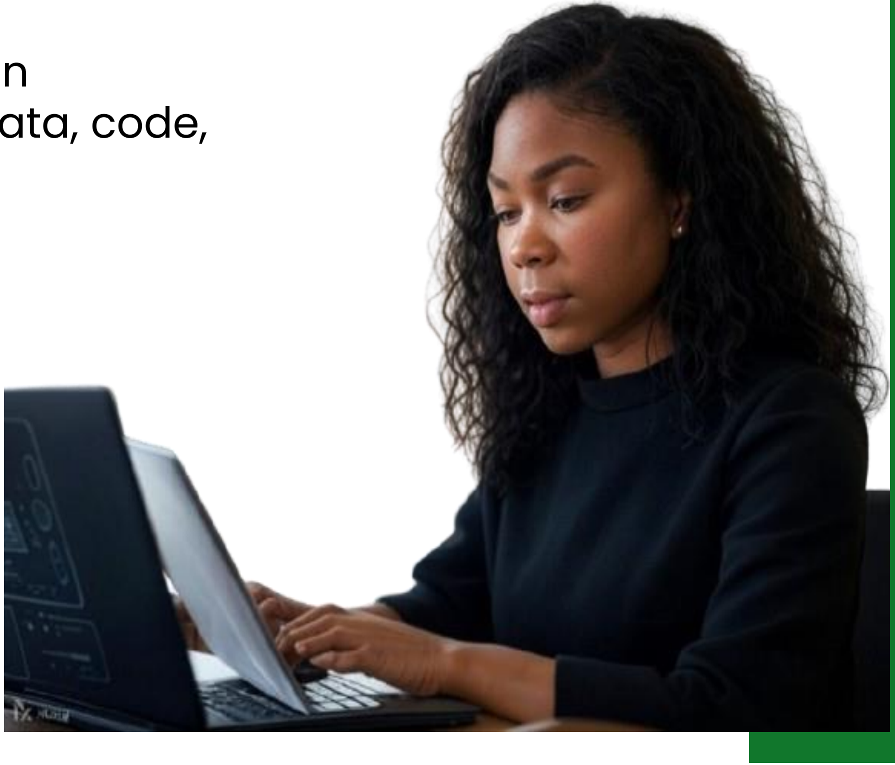
- Explain the concept of reproducibility.
- Organise your work using RStudio Projects
- Work with git as a version control system.
- Collaborate effectively using GitHub.
- Create isolated and reproducible environments with renv.

What is Reproducibility?

Reproducibility is the ability to obtain consistent results using the same data, code, and computational environment.

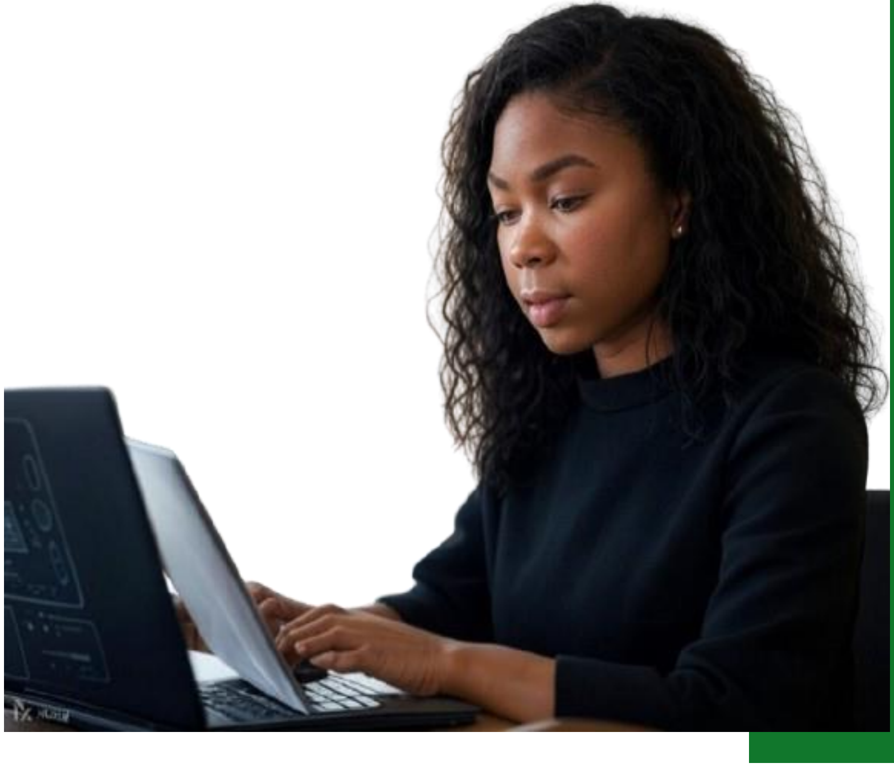
Key Components

- Data
- Code
- Environment



Why is it Important?

- Enhances transparency and trust.
- Enables independent verification of results.
- Facilitates collaboration and iterative improvements.



REPRODUCIBLE WORKFLOWS With

RStudio Project

GBG Analytics

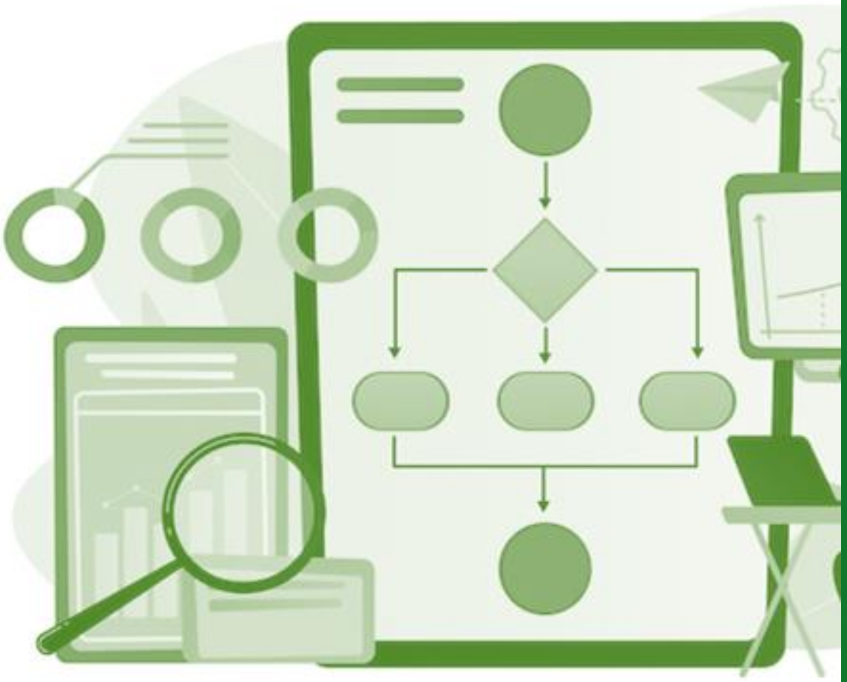
Benefits of Using RStudio Projects

Isolated Environment

Relative File Paths

Version Control Integration

Organized Structure



Lab session

REPRODUCIBLE WORKFLOWS With

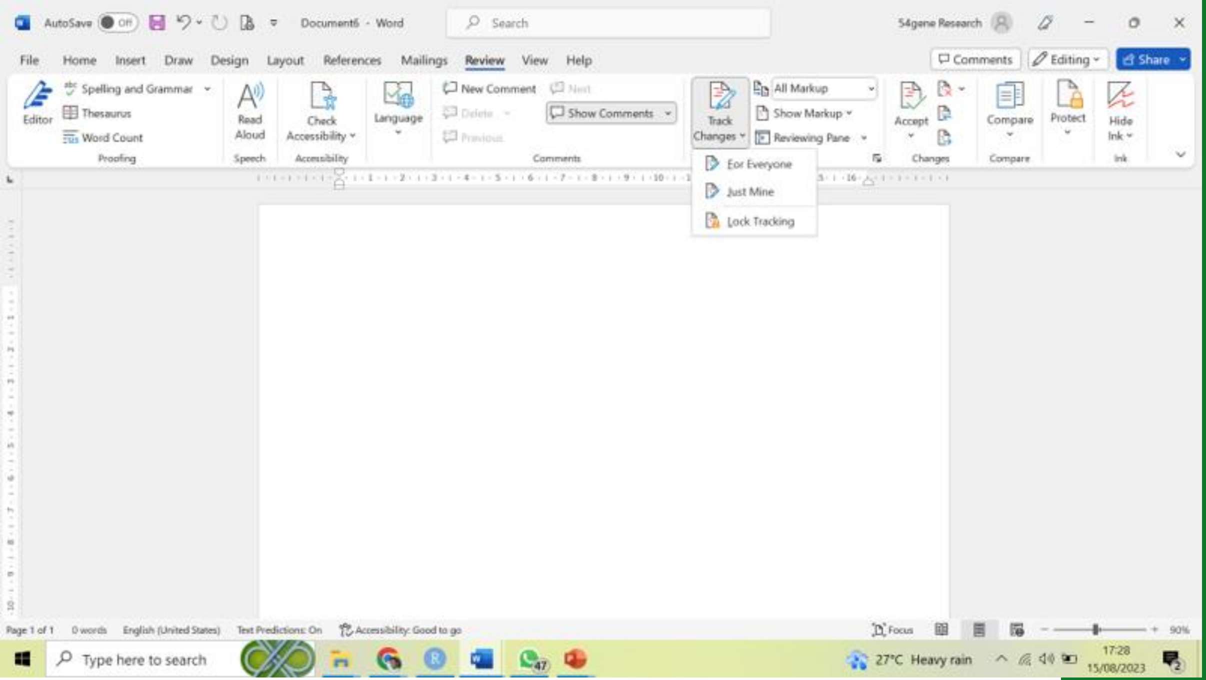
Using Version Control

GBG Analytics

The Role of Version Control

Benefits:

- Tracking Changes
- Collaboration
- Safety



git as a version control system

Git is a distributed version control system that helps you track changes in files, collaborate with others, and manage your project's history efficiently

Other version control

- Mercurial
- Subversion (SVN)
- Perforce (Helix Core)

Git Installation

https://git-scm.com/downloads

git --version

GitHub & GitLab - What are they?

GitHub/GitLab is a website that allows you to store your code on the cloud. It is not just a cloud storage but a full version control system powered by git.

Why Github/Gitlab are Popular Choices for Reproducible Research

Open source

Collaboration features

Integration with other tools

Large user community

Setting up a Github/Gitlab Repository

GitHub

1. Create a Github account.
2. Create a new repository.
3. Clone the repository to your computer.
4. Add files to the repository.
5. Commit changes and push them to the remote repository.

GitLab

1. Create a Gitlab account.
2. Create a new project.
3. Clone the project to your computer.
4. Add files to the project.
5. Commit changes and push them to the remote project.

First-Time git Setup

```
git config –-global user.name “your_github_username”
```

```
git config –-global user.email “email_address”
```

Collaborating with GitHub

Linking Your Local Repository

1. **Create a repository on GitHub:**
Do not initialize with a README if you already have one locally.

1. **Add a remote:**

```
git remote add origin https://github.com/username/my_project.git
```

2. **Push your code:**

```
git push -u origin main
```

Lab session

LABORATORY
EDUCATION

REPRODUCIBLE WORKFLOWS With

renv package

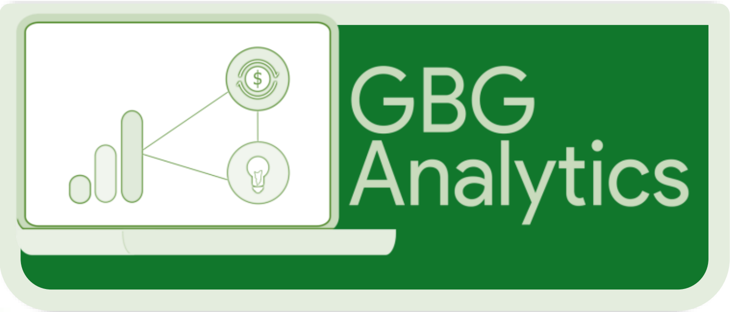
GBG Analytics

Setting Up Reproducible R Environment with renv

The **renv** package is a dependency management tool that creates a project-specific library and ensures that your computing environment is reproducible across different machines. Using renv helps avoid the "it works on my machine" problem by locking in package versions for reproducibility.

Lab session

LABORATORY
EDUCATION



GBG
Analytics