Developer guide

Welcome to the FairChoices team! As a developer, you will be contributing to the collection, processing, and analysis of various types of data. In order to ensure accuracy, consistency, and efficiency, it is important that everyone adheres to a set of established data and coding guidelines. These guidelines are designed to provide a framework for how we collect, store, and analyze data, and they are essential for ensuring that we maintain high standards of quality and integrity in all of our work.

This document provides an overview of the key guidelines that you will need to follow, as well as some tips and best practices for working with data and code. We encourage you to read through this document carefully and to ask questions if there is anything that is unclear or if you need further guidance. FairChoices uses several tools for managing data and code and collaborating as a team. Keep reading to learn more these tools and how to get started.

Step by step instructions

1. Install the FairChoices-standard version of R according to your operating system (Windows or Mac).
2. Download RStudio.
3. If you have a pre-existing GitHub account, contact Jan-Magnus Økland (jan-magnus.okland@uib.no) to get added to the FairChoices repo.
4. If you don't have a pre-existing account, set up a GitHub account and then contact Jan-Magnus Økland (jan-magnus.okland@uib.no) to get added to the FairChoices repo.
   - Use your personal email address if you plan to keep using GitHub after your time at UiB.
5. For data access, contact Sarah Bolongaita (sarah.bolongaita@uib.no) to get added to the FairChoices data folder.

Optional steps for beginners

1. Take the Swirl R tutorial.
2. Take Applied Epi R tutorials.
3. Get familiar with our commonly used R packages.
   - dplyr
   - data.table
   - ggplot2
4. Take the Learn Git Branching tutorial
5. Take the following GitHub Skills courses (at a minimum - feel free to explore more courses!):
   - Introduction to GitHub
   - Review pull requests
   - Resolve merge conflicts

Best Practices

Working data

- Numbers: Period for decimal point; keep all digits
- Currencies: USD, LCU, INT
- Dates: YYYY-MM-DD
- Strings: First letter capital
- Country names: placeholder `function`
  - GBD has two Georgia's (Georgia, the country, and Georgia, the US state)
- Logicals: `TRUE` / `FALSE`
- Empty: `NA`
  - Namibia's ISO2 code is NA
- World bank income groups: LIC = Low income, LMC = Lower middle income, UMC = Upper middle income, HIC = High income, WLD = World

Naming conventions

1. Use clear, easily understandable, and uniquely identifiable names
2. Do not use abbreviations unless extremely common (i.e., HIV for human immunodeficiency virus) or absolutely necessary due to maximum length requirements
   - If you think an abbreviation is necessary, you should consider a more succinct name
3. Use singular form (ex. `location` as opposed to `locations`)
4. Use American English (ex. `center` as opposed to `centre`)
5. Do not use spaces or non-alphanumeric characters (e.g., ! # $ % & ' @ ^ ' ~ + , . ; =)
6. Follow the data type case rules
7. Avoid prefixes or suffixes denoting the data or entity type or are a short abbreviation

Data type case rules

1. **Data frames: lowercase, separate words with an underscore**

   Use all lowercase letters, separating words with an underscore ("_").

   Ex. `location`, `location_iso3`, `location_name`

2. **Columns: TitleCase**

   Capitalize the first letter of each word.

   Ex. `Location`, `LocationIso3`, `LocationName`

3. **Functions: camelCase**

Use lowercase letters for the first word; all following words begin with a capital letter.

Ex. `location`, `locationIso3`, `locationName`

R and RStudio

R is a programming language that is widely used for statistical computing, data analysis, and data visualization. It provides a powerful and flexible platform for exploring and manipulating data, and it has a large and active community of users who have developed a wide range of packages and tools for working with data.

RStudio is an integrated development environment for R that provides a user-friendly interface for writing, testing, and debugging R code. It includes a number of features that make it easy to work with R, such as syntax highlighting, auto-completion, and debugging tools. RStudio also has built-in support for version control systems like Git, which makes it easy to collaborate with other developers and to keep track of changes to your code over time.

As a member of the FairChoices team, you will use R and RStudio to explore and analyze data, creating visualizations, and building statistical models.

Git and GitHub

Git is a version control system that is widely used in software development to manage changes to source code. It allows developers to track changes to their code over time, collaborate with other developers, and roll back to previous versions of their code if necessary. With Git, developers can work on different parts of a codebase simultaneously without interfering with each other's work.

GitHub is a web-based platform that provides hosting for Git repositories. It allows developers to store their code in the cloud, making it easy to share code with other developers and collaborate on projects. GitHub also provides a number of tools and features that make it easy to manage code, such as pull requests, issues, and code reviews.

As a member of the FairChoices team, you will use Git and GitHub to track changes to code, collaborate with other team members, and manage projects.

Data storage

We are currently using **Dropbox** to manage the state of our analytical tool. Dropbox is a file server. Inputs are stored in .csv or .rds files. These files are read and written by the R source code.

Longer term, we would like to transition our data storage technology from a file server like Dropbox to a proper relational database (specifically PostgreSQL) but we're not there yet.