Deploying Shiny apps

- The Shiny package itself is designed to run Shiny applications locally.
- To share Shiny applications with other R users, you can send them your application source as a GitHub gist, R package, or zip file.

Sharing Apps to Run Locally

- Once youve written your Shiny app, you can distribute it for others to run on their own computersthey can download and run Shiny apps with a single `R` command. All that this requires that they have `R` and Shiny installed on their computers.
- If you want your Shiny app to be accessible over the web, so that users only need a web browser, see Deploying Shiny Apps over the Web.

Deploying Shiny

Gist

- One easy way is to put your code on gist.github.com, a code pasteboard service from **GitHub**.
- Both `server.R` and `ui.R` must be included in the same gist, and you must use their proper filenames.
- See *http://gist.github.com/3239667* for an example.

- Your recipient must have R and the Shiny package installed, and then running the app is as easy as entering the following command:

```
shiny::runGist('3239667')
```

- In place of '**3239667**' you will use your gists ID; or, you can use the entire URL of the gist (e.g. '*https://gist.github.com/3239667*').

Deploying Shiny

Advantages of using Gist

- Source code is easily visible by recipient (if desired)
- Easy to run (for `R` users)
- Easy to post and update

Cons

- Code is published to a third-party server

GitHub repository

- If your project is stored in a git repository on GitHub, then others can download and run your app directly. An example repository is at **http://github.com/rstudio**
- The following command will download and run the application:

```
shiny::runGitHub('shiny_example', 'rstudio')
```

In this example, the GitHub account is 'rstudio' and the repository is 'shiny example'; you will need to replace them with your account and repository name.

Github: Advantages

- Source code is easily visible by recipient (if desired)
- Easy to run (for R users)
- Very easy to update if you already use GitHub for your project
- Git-savvy users can clone and fork your repository

Disadvantages

- Developer must know how to use git and GitHub.
- Code is hosted by a third-party server.

Deploying Shiny

Making it into a Package

- If your Shiny app is useful to a broader audience, it might be worth the effort to turn it into an `R` package. Put your Shiny application directory under the packages inst directory, then create and export a function that contains something like this:

```
shiny::runApp(system.file('appdir',
 package='packagename'))
```

where `appdir` is the name of your apps subdirectory in inst, and **packagename** is the name of your package.

Deploying Shiny

Making it into a Package:

Advantages

- Publishable on CRAN
- Easy to run (for `R` users)

Disadvantages

- More work to set up
- Source code is visible by recipient (if not desired)

Deployment over the Web

- You can also deploy Shiny applications over the web, so that users need only a web browser and your applications URL.
- For this, youll need a Linux server and our Shiny Server software.
- Shiny Server is free and open source, though in the future RStudio will offer a commercially licensed edition with additional features for larger organizations.
- RStudio also working on a subscription-based hosting service for Shiny.

Deploying Shiny apps : Shiny Server

Shiny Server

- Shiny Server is if you want to use your own server instead of hosting it on Rstudio's server (i.e. **glimmer**).
- This is really important for those who can't let their code or data out of their organization, or want more computational/storage resources than glimmer can offer, or need their apps to access their internal network.