

Minicurso de Shiny

Como Criar Aplicativos Web Utilizando o R

Marcus Nunes

Universidade Federal do Rio Grande do Norte

3 e 4 de Fevereiro de 2021

XIV Semana da Estatística da UFES

Apresentação

- Professor da Universidade Federal do Rio Grande do Norte
- <https://marcusnunes.me/>
- <https://introbigdata.org/>
- <http://shiny.estatistica.ccet.ufrn.br/>

Introdução

Introdução

- **shiny** é um pacote do R com um *framework* para criação de aplicativos web
- Ele permite que pessoas com pouca experiência em programação web consigam criar sites dinâmicos utilizando seus conhecimentos em R
- Algumas aplicações feitas com o **shiny** podem ser vistas em <http://shiny.estatistica.ccet.ufrn.br>
- O conteúdo desse minicurso está disponível no endereço https://github.com/mnunes/shiny_ufes/

Alguns sites com exemplos de aplicações no **shiny** são os seguintes:

- <https://shiny.rstudio.com/gallery/>
- <http://shiny.estatistica.ccet.ufrn.br>
- <https://r.tquant.eu/>

Instalação do shiny

Instalação do shiny

- Como todo pacote do R, o **shiny** pode ser instalado a partir do prompt através do comando

```
> install.packages("shiny")
```

- Ao rodar

```
> library(shiny)
```

o pacote estará carregado e pronto para uso

- Veja as instruções disponíveis em https://github.com/mnunes/shiny_ufes/ para a listagem completa de pacotes necessários para o minicurso

Anatomia de um shiny App

Anatomia de um shiny App

- O paradigma de programação em **shiny** é ligeiramente diferente daquele que estamos acostumados no R
- Sempre que algum *input* é alterado, todo o app é rodado novamente
- Imagine uma planilha do Excel: sempre que a célula A1 é alterada, todas as células que dependem de A1 são alteradas também

Anatomia de um shiny App

- Todo shiny app é composto de até três partes:
 - **ui**: é onde a interface com o usuário (user interface) é definida
 - **server**: os comandos do R que são a alma do app estão nesse arquivo, ou seja, é aqui que os gráficos são construídos, que dados são filtrados etc.
 - **global**: serve para organizar o carregamento de pacotes, conjuntos de dados e tudo o que necessitar ser acessado de maneira global pelo app
- Enquanto os objetos **server** e **ui** são obrigatórios, o objeto **global** é opcional

Anatomia de um shiny App

- Programas simples podem ser rodados em apenas um arquivo
- Vamos ver como isso funciona na prática
- Abra o arquivo `exemplos/01-histograma/hist-simples.R` para ver uma aplicação

Anatomia de um shiny App

- Outra maneira de criar aplicativos é separar os objetos `server`, `ui` e `global` em arquivos diferentes
- Abra os arquivos `server.R`, `ui.R` e `global.R` presentes na pasta `exemplos/01-histograma/`
- É o mesmo aplicativo visto anteriormente, organizado de outra maneira

Tipos de Layout

Tipos de Layout

- O `shiny` já vem com diversos layouts pré-configurados para que possamos criar nossas ferramentas
- Basta escolher um deles e começar a produzir o nosso app

Tipos de Layout

- `sidebarLayout`
- `splitLayout`
- `verticalLayout`
- `flowLayout`

Tipos de Inputs

Tipos de Inputs

- Já vimos informalmente algumas maneiras de interagir com os apps criados
- Essas maneiras não são as únicas de criarmos formas de interação com nossos programas
- Abra os arquivos da pasta **03-inputs** para que exploremos elas

Tipos de Inputs

- `checkboxInput`: cria uma caixa de seleção com apenas uma opção
- `checkboxGroupInput`: cria uma caixa de seleção com mais de uma opção
- `dateInput`: abre um calendário para a seleção de datas

Tipos de Inputs

- `textInput`: cria uma caixa de texto
- `numericInput`: cria uma caixa que recebe apenas números
- `passwordInput`: cria uma caixa de texto para receber senhas

Tipos de Inputs

- `selectInput`: cria uma caixa com uma lista de seleção
- `actionButton`: cria um botão de ação

Tipos de Outputs

Tipos de Outputs

- Após entrarmos com os dados nos programas, é importante que utilizemos eles para algo
- A partir de agora veremos como transformar os inputs (entradas) em outputs (saídas)

Tipos de Outputs

- `renderPlot`: exibe o resultado de um gráfico criado pelo server
- `renderPrint`: mostra saídas do console do R
- `renderText`: cria um output no formato de texto
- `renderTable`: exibe resultados de data frames ou matrizes

- `renderPlotly`: cria um gráfico interativo utilizando o pacote `plotly`

O pacote `plotly` é capaz de criar um gráfico interativo a partir de um gráfico padrão do R, sem que seja necessário programar passos extras

Uma Aplicação Completa

Uma Aplicação Completa

- O endereço `http://shiny.estadistica.ccet.ufrn.br/regressao-linear-interativa/` possui uma aplicação educacional do `shiny`
- Vá ao endereço `https://github.com/mnunes/regressao/` e baixe os arquivos desse aplicativo
- Nós iremos dar uma olhada em como ele funciona

Nosso Primeiro Projeto

Nosso Primeiro Projeto

- Não há como iniciar um projeto sem sabermos onde queremos chegar
- Por isso, é importante definirmos qual o objetivo
- Eu tenho uma proposta: criar um dashboard para análise de dados a respeito de países

Nosso Primeiro Projeto

- **Dashboard** é uma página que exibe informações importantes sobre algum assunto de interesse
- Estas informações vão desde informações simples, como um gráfico de linha com o total de vendas anuais de uma empresa, até informações complexas, como o mapa de calor das vendas de acordo com a sua localização geográfica
- Em um dashboard são exibidos tabelas, gráficos e mecanismos de controle e personalização das informações
- Assim, em vez de criarmos milhares de relatórios personalizados, deixamos que o usuário decida que informações ele deseja

Nosso Primeiro Projeto

- Os dados que vamos utilizar estão no arquivo `BancoMundial.csv`
- Carregue-o na memória do `R` e verifique se ele possui 844 linhas e 9 colunas

Nosso Primeiro Projeto

- As colunas desse conjunto de dados são
 - Pais: o país ao qual os dados se referem (em inglês)
 - Regiao: a região do mundo na qual esse país está localizado (em inglês)
 - Ano: ano da informação
 - Populacao: tamanho da população do país
 - ExpectativaVida: expectativa de vida ao nascer
 - PIB: Produto Interno Bruto em US\$
 - PIB_Capita: Produto Interno Bruto por habitante em US\$
 - Fertilidade: número de filhos por mulher
 - Pobreza: percentual de habitantes abaixo da linha de pobreza

Nosso Primeiro Projeto

- A partir de agora iremos construir nosso primeiro projeto juntos, de maneira colaborativa
- Vamos criar um dashboard para análise de dados sócio-econômicos de países

Alterando o Layout

Alterando o Layout

- É possível adicionar abas aos apps criados
- Assim, adicionamos mais conteúdo e ele fica mais organizado
- As funções `tabsetPanel` e `tabPanel` fazem isso de maneira trivial

Publicando o Trabalho

Publicando o Trabalho

- O site <https://www.shinyapps.io/> permite que qualquer pessoa publique seus aplicativos na internet gratuitamente
- Entretanto, há limites de números de apps e acessos para o plano de hospedagem gratuito desse site
- Outra maneira de ter seu app na internet é conversando comigo e enviando os arquivos para mim
- Eu administro o site <http://shiny.estatistica.ccet.ufrn.br> e nele é possível colocar qualquer aplicativo feito em **shiny**, sem preocupações com limites

Considerações Finais

Considerações Finais

- Esse curso foi uma introdução ao que o **shiny** é capaz de fazer
- O pacote é uma excelente maneira de divulgar estatística e criar produtos para empresas
- Crie seus próprios aplicativos e construa o seu portfólio

Minicurso de Shiny

Como Criar Aplicativos Web Utilizando o R

Marcus Nunes

Universidade Federal do Rio Grande do Norte

3 e 4 de Fevereiro de 2021

XIV Semana da Estatística da UFES