



R Shiny Introduction



Rendez-vous SFDS Méthodes et Logiciel
Jeudi 20 avril 2017

Anne GAYET
agayet@aid.fr



CRM Services

- 30 BDD marketing B2B, B2C
- 200M de clients
- 5 milliards de transactions
- Référentiels client unique
- Data Quality
- Data Visualisation



Data Science

- Connaissance client
- Performance et optimisation
- Textmining, NLP
- Prévisions
- Recommandation
- Géomarketing
- Webmining



datakili®

- Plate-forme SaaS
- Dédicée aux parcours clients multi-canaux
- Outils big data
- Pour analystes et marketeurs



Conseil et pilotage de projets

- Marketing & Fidélisation
- Accompagnement du changement
- Gestion de projet



AID Academy

- Formations Outils et Méthodes
- Séminaires

VOUS

0) Je programme en R chaque semaine

→ *presque tous !*

1) J'ai déjà créé une mini app Shiny avec un tutoriel

→ *presque tous !*

2) J'ai déjà créé et déployé une app Shiny "personnelle"

→ *2 personnes (hors intervenants)*





Shiny est un (ensemble de) package(s) pour construire des applications interactives web.



Développé par Rstudio, première version 11/2012 (sur le CRAN)



Entièrement compatible avec R, donc customisable à souhait.

Toute la “force” de R et ses packages pour la dataviz peut être exploitée



L'objectif est de disséminer des résultats ou des “schémas d'analyse”,
une application Shiny a donc besoin d'être déployée auprès des utilisateurs.



Côté données :
données pré-calculées,
calculs à la volée,
calculs à la demande ?
à vous de voir ...



Coté navigateur:
Chrome, Firefox ou Safari,
pas Explorer

Shiny génère automatiquement du code html associé à un fichier CSS et javascript directement interprété par le navigateur web.

Le **CSS** est un langage permettant de mettre en forme une page web. Associé à un fichier html (celui de l'application Shiny générée), il décrit la mise en forme.



La personnalisation d'une app peut aussi passer par l'intégration de javascript, de CSS, d'HTML et d'images.

L'apparence d'une application utilise un « masque » (framework Bootstrap). Il est donc possible de la personnaliser (fichier stylesheet.css) Cela en fait des applications « responsives ».

PETIT CHOIX D'APPS SHINY EN LIGNE

<https://craigwang.shinyapps.io/rPackageStats/>

https://gedemced.shinyapps.io/DP_CATALONIA/

<https://cpelat.shinyapps.io/mass/>

<https://daattali.com/shiny/timevis-demo/>

https://daroczig.shinyapps.io/rinfinance_Berlinger-Daroczi-demo/ ? jouer avec le graph

<socialdash.adoptitude.com/> à essayer avec @GAYET_Anne ou @trump ☺

www.davesteps.com/geoExploreR/

traipop.parisgeo.cnrs.fr/ clustering de courbes d'évolution

<https://asheshwor.shinyapps.io/migrationviz/>

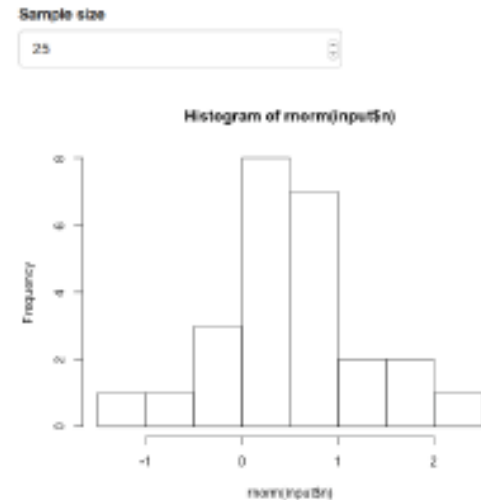
<https://sarthakdasadia.shinyapps.io/GlobalTerrorism/>

<http://shiny.rstudio.com/gallery/bus-dashboard.html>

<https://davesteps.shinyapps.io/machLearn/>

<https://dreamrs-fan.shinyapps.io/pimpMyShiny/>

```
> library(shiny)
> ui <- fluidPage(
  numericInput(inputId = "n",
    "Sample size", value = 25),
  plotOutput(outputId = "hist"))
> server <- function(input, output){
  output$hist <- renderPlot({
    hist(rnorm(input$n))
  })
}
> shinyApp(ui = ui, server = server)
```



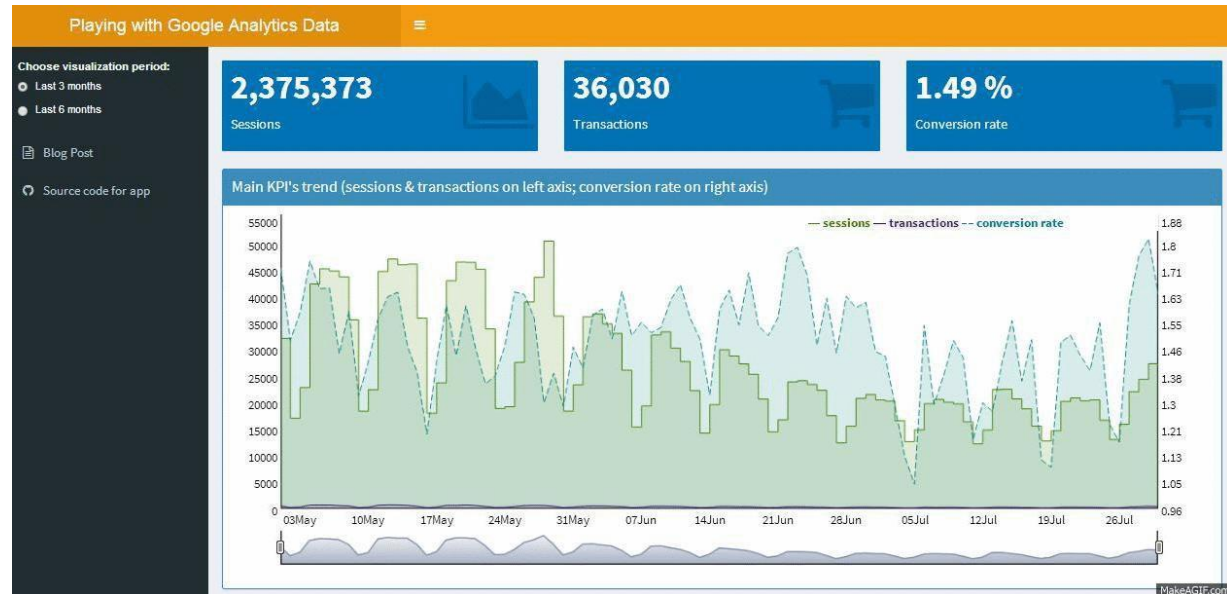
ui (user interface): définit les zones d'input et d'output

server: définit l'ensemble des traitements

Je ne vais pas plus loin, ensuite cela dépend

shiny tout court

ou shinydashboard



- ❖ Package plus ou moins « dérivé » de shiny. Il présente des fonctionnalités similaires mais un design de base de l'appli un peu différent, plus proche des « tableaux de bords ».
- ❖ Interface basée sur le [framework Admin LTE 2](#)
- ❖ Toutes les fonctionnalités disponibles dans ce framework n'ont pas été développées dans le package R → limite du package
- ❖ Moins customisable que les applications développées avec le package R Shiny, par exemple :
 - Pas de barre de navigation (navbar)
 - Sidebar est toujours à gauche (pas possible de le changer)

shiny server

- serveur à installer chez vous
- Vos données restent chez vous.
- Version gratuite
- Version payante: gère des utilisations concomitantes et la sécurisation des connexions
Shiny Server Pro
10 k\$ / an pour 20 utilisateurs max en même temps, dégressif
- Problème éventuel: la gestion du serveur, DSI +/- compatissante et compétente

shinyapps.io

- hébergé sur un serveur mis à disposition par Rstudio (cloud)
- Vos données sortent.
- Version gratuite
- Version payante: offre Pro 3.3 k\$ / an pour 5000h d'utilisation / mois
- Techno: solution basée sur Docker

Rconnect

- New : shiny server en version "clic-bouton"
- Vos données sortent
- Permet de faire du développement collaboratif, d'automatiser des mises à jour
- 15k\$ / an pour 20 users nommés
25k\$ pour 100 users
75k\$ pour 1000 users

www.rstudio.com/pricing3/#Comparison

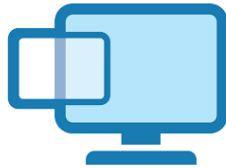
COMPARATIF DES 2 SOLUTIONS PRO DE RSTUDIO

	Shiny Server Pro	Shinyapps.io
Tarif	10 k\$ / an pour 20 utilisateurs Pas de limite en temps d'utilisation	Pas de limite en nombre d'utilisateur 1,1 k\$ / an pour 1000 heures d'utilisation / mois (offre Standard) 3,3 k\$ / an pour 5000 heures d'utilisation / mois (offre PRO)
Hébergement	Serveur internes - gestion par les équipes IT - données déposées sur le serveur sécurisées	Cloud Public - géré par RStudio - données à l'extérieur
Performances	Dépendant du serveur utilisé (CPU / RAM / Disque, ...)	Nombre non limité de machines Chaque machine limité à 500 Mo de mémoire (évolution possible)
Persistance des données	oui	non : - à chaque nouvelle version d'application déployée, tout est "écrasé" - les données écrites par chaque instance sont perdus à la déconnexion de l'utilisateur
Authentification	Géré par l'IT via LDAP	Sécurisé par un système d'identifiant / mot de passe via des comptes : - google - github - shinyapps.io
Gestion des packages	Centralisé et géré par les équipes internes	Propre à chaque application et automatisé

On « envoie » l'application, avec ou sans données (avec seulement si « petites »).

On contrôle ainsi l'ensemble des composants de l'application (version de R, de Rstudio, version des packages).

L'application tournera en local, donc avec la puissance du PC de l'utilisateur.



🚧 dans une VM

isolation parfaite
(possible de changer de système
d'exploitation)



🚧 avec un container Docker ou Rocker

open source
conseillé aussi pour des chaînes R sans Shiny

<http://r-addict.com/eRka10/#>

- Il y a une tonne de raisons pour qu'une app shiny transportée d'une machine à une autre ne marche plus.
NB: La plupart des raisons sont valables pour R tout court.
- C'est d'autant plus vrai si l'application a un but analytique et combine de nombreux packages.
- Avantages :
 - ◆ Contrôler les dépendances
 - ◆ Joindre des données dans un format R
 - ◆ Gérer « proprement » les mises à jour et les tests
 - ◆ Plus facile que d'utiliser Docker

Il ne faut pas se voiler la face: il faut aimer développer !

L'univers de concurrence est assez flou:



etc ..



(complémentaires)

le site shiny : <http://shiny.rstudio.com/>

les galeries shiny :

<http://shiny.rstudio.com/gallery/>

<https://www.rstudio.com/products/shiny/shiny-user-showcase/>

<http://www.showmeshiny.com/> (attention, beaucoup ne marchent plus ...)

les tutoriels et pense-bête :

<http://shiny.rstudio.com/tutorial/>

<http://deanattali.com/blog/building-shiny-apps-tutorial/>

<http://shiny.rstudio.com/images/shiny-cheatsheet.pdf>