🚜

Preparar o Ambiente R

This wiki is built in Notion. Here are all the tips you need to contribute.

- The Basics
  - Importando as libraries
  - Load libraries
  - List of libraries
  - Organizing Libraries
- Advanced Techniques

The Basics

Importando as libraries

Verificação das bibliotecas para carga no ambiente do R.

> 💡 **Notion Tip:**
> **install.packages("lib_name")**

Load libraries

Carregar as bibliotecas

> 💡 **Notion Tip:**
> **library(lib_name)**

List of libraries

Listagem das bibliotecas utilizadas.

```
# Todas as libraries utilizadas library(lib_name).
library(Amelia)
library(AUC)
library(car)
library(caret)        # Handy ML functions
library(caretEnsemble)
library(caTools)
library(cdparcoord)
library(CGPfunctions)
library(CHAID)
library(class)
library(cluster)    # clustering algorithms
library(conquer)
library(correlationfunnel)
library(corrr)
library(corrplot)
library(cowplot)
library(cusumcharter)
library(cvms)
library(DataExplorer)
library(data.table) # Fast I/O
library(dendextend) # for comparing two dendrograms
library(dplyr)      # Data munging
library(DT)
library(e1071)      # Misc stat fns
library(epiDisplay)
library(excelR)
library(factoextra) # clustering visualization
library(fmsb)       # for radar plot
library(funModeling)
library(GGally)
library(ggcorrplot)
library(ggfun)
library(ggimage)
library(ggmosaic)
library(ggnewscale)
library(ggplot2)
library(ggplotify)
library(ggpubr)
library(ggthemes)
library(gmodels)
library(gplots)
library(gridExtra)
library(gridGraphics)
library(groupdata2)
library(gtools)
library(Hmisc)
library(imbalance)
library(InformationValue)
library(ipred)
```

```r
library(janitor)
require(kableExtra)
library(lattice)
library(LTRCtrees)    # or survival tree
library(lime)
library(lubridate)    # Makes dates easy
library(magick)
library(magrittr)     # pipe operators
library(MASS)
library(Metrics)
library(mice)
library(MLDataR)
library(mosaic)
library(numbers)
library(party)
library(partykit)
library(plotly)       # Interactive charts
library(plyr)
library(pROC)
library(pscl)
library(purrr)
library(randomForest) # rf
library(RColorBrewer)
library(Rcpp)
library(readr)
library(recipes)
library(repr)
library(reshape2)
library(rlang)
library(Rmisc)
library(rms)
library(ROCR)         # ML evaluation
library(rpart)        # Decision Trees
library(rpart.plot)   # Pretty tree plots
library(rpivotTable)  # for displaying the data
library(rsample)
library(rsvg)
library(scales)
library(shiny)
library(survminer)
library(survRM2)
library(survival)
library(stats19)
library(states)
library(stringr)
library(tidyr)      # Data munging
library(tidyverse)  # data manipulation
library(xgboost)
library(xpectr)
library(yardstick)
library(yulab.utils)
library(workflows)

theme_set(theme_bw()) # set theme for ggplot2
```

```
if(!require('epr')) {
  install.packages('epr')
  library('epr')
}


# Sintaxe utilizada para instalação das libraries.
install.packages("Amelia", type = "source",  dependencies=TRUE)
install.packages("car", type = "source",  dependencies=TRUE)
install.packages("caret", type = "source",  dependencies=TRUE)
install.packages("caretEnsemble", type = "source",  dependencies=TRUE)
install.packages("caTools", type = "source",  dependencies=TRUE)
install.packages("cdparcoord", type = "source",  dependencies=TRUE)
install.packages("CGPfunctions", type = "source",  dependencies=TRUE)
install.packages("CHAID", type = "source",  dependencies=TRUE)
install.packages("class", type = "source",  dependencies=TRUE)
install.packages("cluster", type = "source",  dependencies=TRUE)
install.packages("ConfusionTableR", type = "source",  dependencies=TRUE)
install.packages("conquer", type = "source",  dependencies=TRUE)
install.packages("correlationfunnel", type = "source",  dependencies=TRUE)
install.packages("corrr", type = "source",  dependencies=TRUE)
install.packages("corrplot", type = "source",  dependencies=TRUE)
install.packages("cowplot", type = "source",  dependencies=TRUE)
install.packages("cvms", type = "source",  dependencies=TRUE)
install.packages("cusumcharter", type = "source",  dependencies=TRUE)
install.packages("DataExplorer", type = "source",  dependencies=TRUE)
install.packages("data.table", type = "source",  dependencies=TRUE)
install.packages("dendextend", type = "source",  dependencies=TRUE)
install.packages("dplyr", type = "source",  dependencies=TRUE)
install.packages("DT", type = "source",  dependencies=TRUE)
install.packages("e1071", type = "source",  dependencies=TRUE)
install.packages("epiDisplay", type = "source",  dependencies=TRUE)
install.packages("excelR", type = "source",  dependencies=TRUE)
install.packages("factoextra", type = "source",  dependencies=TRUE)
install.packages("fmsb", type = "source",  dependencies=TRUE)
install.packages("funModeling", type = "source",  dependencies=TRUE)
install.packages("GGally", type = "source",  dependencies=TRUE)
install.packages("ggcorrplot", type = "source",  dependencies=TRUE)
install.packages("ggmosaic", type = "source",  dependencies=TRUE)
install.packages("ggplot2", type = "source",  dependencies=TRUE)
install.packages("ggpubr", type = "source",  dependencies=TRUE)
install.packages("ggthemes", type = "source",  dependencies=TRUE)
install.packages("gmodels", type = "source",  dependencies=TRUE)
install.packages("gplots", type = "source",  dependencies=TRUE)
install.packages("gridExtra", type = "source",  dependencies=TRUE)
install.packages("gtools", type = "source",  dependencies=TRUE)
install.packages("Hmisc", type = "source",  dependencies=TRUE)
install.packages("imbalance", type = "source",  dependencies=TRUE)
install.packages("InformationValue", type = "source",  dependencies=TRUE)
install.packages("ipred", type = "source", dependencies = TRUE)
install.packages("janitor", type = "source",  dependencies=TRUE)
install.packages("jsonlite", type = "source", dependencies=TRUE)
install.packages("kableExtra", type = "source",  dependencies=TRUE)
install.packages("lattice", type = "source",  dependencies=TRUE)
install.packages("LTRCtrees", type = "source",  dependencies=TRUE)
install.packages("lime", type = "source",  dependencies=TRUE)
install.packages("lubridate", type = "source",  dependencies=TRUE)
install.packages("magrittr", type = "source",  dependencies=TRUE)
install.packages("MASS", type = "source",  dependencies=TRUE)
install.packages("Metrics", type = "source",  dependencies=TRUE)
install.packages("mice", type = "source",  dependencies=TRUE)
install.packages("MLDataR", type = "source",  dependencies=TRUE)
install.packages("mosaic", type = "source",  dependencies=TRUE)
install.packages("party", type = "source",  dependencies=TRUE)
install.packages("partykit", type = "source",  dependencies=TRUE)
install.packages("plotly", type = "source",  dependencies=TRUE)
install.packages("plyr", type = "source",  dependencies=TRUE)
install.packages("pROC", type = "source",  dependencies=TRUE)
install.packages("pscl", type = "source",  dependencies=TRUE)
install.packages("purrr", type = "source",  dependencies=TRUE)
install.packages("randomForest", type = "source",  dependencies=TRUE)
install.packages("RColorBrewer", type = "source",  dependencies=TRUE)
install.packages("Rcpp", type = "source",  dependencies=TRUE)
install.packages("readr", type = "source",  dependencies=TRUE)
install.packages("recipes",type = "source",  dependencies=TRUE)
install.packages("repr", type = "source",  dependencies=TRUE)
install.packages("reshape2", type = "source",  dependencies=TRUE)
install.packages("rlang", type = "source",  dependencies=TRUE)
install.packages("Rmisc", type = "source",  dependencies=TRUE)
install.packages("rms", type = "source",  dependencies=TRUE)
install.packages("ROCR", type = "source",  dependencies=TRUE)
install.packages("rpart.plot", type = "source",  dependencies=TRUE)
install.packages("rpivotTable", type = "source",  dependencies=TRUE)
install.packages("rpart", type = "source",  dependencies=TRUE)
install.packages("rpart.plot", type = "source",  dependencies=TRUE)
install.packages("rsample", type = "source",  dependencies=TRUE)
install.packages("scales", type = "source",  dependencies=TRUE)
install.packages("shiny", type = "source",  dependencies=TRUE)
install.packages("survminer", type = "source",  dependencies=TRUE)
install.packages("survRM2", type = "source",  dependencies=TRUE)
install.packages("survival", type = "source",  dependencies=TRUE)
install.packages("stats19", type = "source",  dependencies=TRUE)
install.packages("states", type = "source",  dependencies=TRUE)
install.packages("stringr", type = "source",  dependencies=TRUE)
install.packages("tidyr", type = "source", dependencies = TRUE)
install.packages("tidyverse", type = "source", dependencies = TRUE)
install.packages("xgboost", type = "source",  dependencies=TRUE)
install.packages("yardstick", type = "source",  dependencies=TRUE)
install.packages("workflows", type = "source",  dependencies=TRUE)
```

Organizing Libraries

A fim de otimizar a preparação do ambiente, algumas formas de efetuar o load em formato de pacotes:

```
# Definindo uma função de aplicação.
  mypack <- function(package){
    new.package <- package[!(package %in% installed.packages()[, "Package"])]
    if (length(new.package))
      install.packages(new.package, dependencies = TRUE)
    sapply(package, require, character.only = TRUE)
  }

# Criando um vetor chamado [packages].
packages <- c("lib_name1", "lib_name2", ...)
packages
mypack(packages)

# Aplicando as libraries dentro de [packages]
ipak <- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
}

# Verificando as libraries aplicadas.
ipak(packages)
```

Instead of using folders, Notion lets you nest pages inside pages. Type `/page` and press `enter` to create a sub-page inside a page. Like this:

Copy of Example sub-page

Advanced Techniques

Check out this Notion Editor 101 guide for more advanced tips and how-to's.