

Projet R avancé

Margaux Bailleul - Marie Guibert

2023-05-04

```
library(rvest)
library(tidyverse)
library(jsonlite)
library(sf)
library(stringr)
library(jpeg)
library(png)
```

N.B. : Au format PDF, certaines de nos sorties ne sont pas affichées entièrement. Afin de les avoir en entier, n'hésitez pas à exécuter le code Rmd.

Présentation de notre projet

Lors de cette étude, nous avons abordé un sujet qui nous tient particulièrement à coeur : les festivals de musique. En effet, nous avons choisi d'aborder ce sujet et de nous focaliser les festivals de musique amplifiées et électroniques. Ces événements ont une portée culturelle très importante et un impact économique dans le monde entier.

Données provenant de l'API

Premièrement, nous avons choisi une API nationale répertoriant tous les festivals de musique de France. Nous avons affiné notre étude en ne sélectionnant que les musiques amplifiées ou électroniques.

Nous allons donc procéder à l'étude de ce panorama de festivals !

Extraction des données via une API

```
url_festival <- "https://data.culture.gouv.fr/api/records/1.0/search/?dataset=panorama-des-festivals&q="
```

Nous utilisons le package **jsonlite** pour extraire les données de l'API.

```
contenu <- fromJSON(url_festival)
df_festivals <- contenu$records$fields
glimpse(df_festivals)
```

Tout d'abord, visualisons nos données de façon globale sans effectuer de modifications :

```
head(df_festivals,3)
```

```
##      dept_sk      coordonnees_insee mois_indicatif_en_chiffre_y_compris_double_mois
## 1         21 47.322944, 5.037888                                     1
## 2         31 43.596381, 1.431673                                     1
## 3         42 45.43012, 4.37914                                     2
##      date_de_fin_ancien nom_departement departement mois_habituel_de_debut
## 1      2019-02-06      Côte-d'Or          21          01 (janvier)
## 2      2019-02-02      Haute-Garonne      31          01 (janvier)
## 3      2019-02-15              Loire      42          02 (février)
##      code_postal      complement_domaine      domaine
## 1         21000              Danse contemporaine      Danse
## 2         31300              Chanson Musiques actuelles
## 3        42100 Jazz, blues et musiques improvisées Musiques actuelles
##      libelle_commune_pour_calcul_cp_insee date_debut_ancien code_insee
## 1              DIJON          2019-01-21      21231
## 2             TOULOUSE          2019-01-22      31555
## 3           ST ETIENNE          2019-02-01      42218
##      region commune_principale      site_web
## 1 Bourgogne-Franche-Comté      DIJON https://art-danse.org/
## 2              Occitanie      TOULOUSE www.detoursdechant.com
## 3 Auvergne-Rhône-Alpes      ST ETIENNE www.gagajazz.com
##      nom_de_la_manifestation ndeg_identification check_edition periodicite
## 1              Art Danse          BF001          2020      <NA>
## 2          DETOURS DE CHANT          KD088              0      Annuelle
## 3      LES JAZZERIES D'HIVER          AD316              0      Annuelle
##
##      autres_communes
## 1      <NA>
## 2 Launaguet (31) / Colomiers (31) / Bruguieres (31) / Bouloc (31) / Tournefeuille (31)
## 3      <NA>
##      date_de_creation ndeg_de_l_edition_2018 ndeg_de_l_edition_2019
## 1      <NA>              NA              NA
## 2      2002-01-01          17              18
## 3      2011-01-01          NA              9
##
## 1
## 2
## 3 9ème édition. Les Jazzeries d'hiver sont organisées par l'association Gaga Jazz à St Etienne dans c
##      soutenu_en_2017_par_le_centre_national_du_cinema
## 1      <NA>
## 2      <NA>
## 3      <NA>
##      soutenu_en_2017_par_le_ministere_de_la_culture
## 1      <NA>
## 2      <NA>
## 3      <NA>
##      soutien_2017_mcc_a_la_structure
## 1      <NA>
## 2      <NA>
## 3      <NA>
##      soutenu_en_2017_par_le_centre_national_des_varietes
## 1      <NA>
## 2      <NA>
```

```
## 3 <NA>
## soutenu_en_2018_par_le_ministere_de_la_culture
## 1 <NA>
## 2 <NA>
## 3 <NA>
## soutenu_en_2017_par_le_centre_national_du_livre enquete_drac_2017
## 1 <NA> <NA>
## 2 <NA> <NA>
## 3 <NA> <NA>
## mois_indicatif soutenu_en_2018_par_le_centre_national_du_cinema
## 1 <NA> <NA>
## 2 <NA> <NA>
## 3 <NA> <NA>
## soutenu_en_2018_par_le_centre_national_des_varietes
## 1 <NA>
## 2 <NA>
## 3 <NA>
## part_festival_sur_soutien_a_la_structure
## 1 <NA>
## 2 <NA>
## 3 <NA>
```

Nettoyage de la base de données

Premièrement, nous nous focalisons sur les données concernant les musiques amplifiées ou électroniques

```
df_festivals <- df_festivals |>
  filter(complement_domaine == "Musiques amplifiées ou électroniques")
dim(df_festivals)
```

```
## [1] 629 36
```

Nous avons maintenant une base de données avec 629 lignes et 36 colonnes.

Afin de faciliter notre étude, nous avons choisi de supprimer certaines colonnes de la base de données. De plus, certaines informations sont redondantes, nous avons donc choisi de les omettre aussi. Par exemple, le domaine correspond aux musiques actuelles et plus spécialement aux musiques amplifiées ou électroniques (complement_domaine). Ces deux colonnes n'étaient donc pas pertinentes pour la suite de notre analyse.

```
df_festivals <-df_festivals |>
  select(coordonnees_insee,date_de_fin_ancien,nom_departement,departement,periodicite,mois_habituel_de_c
head(df_festivals,3)
```

```
##      coordonnees_insee date_de_fin_ancien nom_departement departement
## 1 48.400500, -4.502791      2019-02-10      Finistère          29
## 2 43.571628, 3.832218      2019-03-09      Hérault           34
## 3 46.866613, -1.022161      2019-03-09      Vendée            85
##      periodicite mois_habituel_de_debut code_postal
## 1 Annuelle      02 (février)      29200
## 2 Annuelle      03 (mars)        34430
## 3 Annuelle      03 (mars)        85500
##      libelle_commune_pour_calcul_cp_insee date_debut_ancien      region
```

```
## 1          BREST          2019-02-05          Bretagne
## 2          ST JEAN DE VEDAS 2019-03-08          Occitanie
## 3          LES HERBIERS    2019-03-08 Pays de la Loire
##   nom_de_la_manifestation      site_web
## 1   ASTROPOLIS L'HIVER          <NA>
## 2   THIS IS ENGLAND tafproduction.blogspot.fr/
## 3   FREE SONS DIVERS    www.freesonsdivers.com
```

Deuxièmement, nous allons transformer les variables caractères en facteurs pour effectuer des traitements de données et des graphiques plus facilement.

```
# str(df_festivals) # permet de connaître le type de chaque variable du dataframe
df_festivals[c("nom_departement",
               "periodicite",
               "code_postal",
               "libelle_commune_pour_calcul_cp_insee",
               "region")] <- lapply(df_festivals[c("nom_departement",
               "periodicite",
               "code_postal",
               "libelle_commune_pour_calcul_cp_insee",
               "region")], as.factor)
# str(df_festivals)
```

Dans cette étude, nous allons nous concentrer sur les festivals annuels. Nous choisissons donc de ne pas prendre en compte les autres modalités de la variable **periodicite**.

```
table(df_festivals$periodicite)
```

```
##
##          Annuelle          Biennale Biennale années impaires
##          553              1              1
##   Biennale années paires
##          2
```

```
df_festivals <- df_festivals |>
  filter(periodicite == "Annuelle") |> # filtrage pour n'avoir que les festivals annuels
  select(-periodicite) # suppression de la colonne periodicite
                        # car les informations sont redondantes à présent

# Vérification :
# table(df_festivals$periodicite)
```

Nous allons extraire les coordonnées GPS de la variable **coordonnees_insee** afin de la scinder en deux colonnes : latitude et longitude. Cette étape nous permettra de réaliser plus facilement notre carte par la suite.

```
# On extrait d'abord les coordonnées et on crée un dataframe contenant la latitude et la longitude
coord_df <- data.frame(matrix(unlist(sapply(df_festivals$coordonnees_insee, function(x) {
  unlist(strsplit(paste(x, collapse = ", "), ", "))
})), ncol = 2, byrow = TRUE))

coord_df <- coord_df |>
```

```

rename("longitude" = X1, "latitude" = X2)

# Vérification de la bonne forme du dataframe
# coord_df

# On concatène les deux dataframes
df_festivals <- bind_cols(df_festivals, coord_df)
# df_festivals

# On supprime la colonne coordonnees_insee car on ne la réutilisera pas
df_festivals <- df_festivals |>
  select(-coordonnees_insee)

# Vérification
head(df_festivals, 3)

##   date_de_fin_ancien nom_departement departement mois_habituel_de_debut
## 1      2019-02-10      Finistère          29          02 (février)
## 2      2019-03-09      Hérault           34          03 (mars)
## 3      2019-03-09      Vendée           85          03 (mars)
##   code_postal libelle_commune_pour_calcul_cp_insee date_debut_ancien
## 1      29200                      BREST          2019-02-05
## 2      34430                      ST JEAN DE VEDAS          2019-03-08
## 3      85500                      LES HERBIERS          2019-03-08
##           region nom_de_la_manifestation          site_web
## 1      Bretagne      ASTROPOLIS L'HIVER          <NA>
## 2      Occitanie      THIS IS ENGLAND tafproduction.blogspot.fr/
## 3 Pays de la Loire      FREE SONS DIVERS      www.freesonsdivers.com
##           longitude          latitude
## 1 48.4004997828 -4.5027907853
## 2 43.5716282319  3.83221847952
## 3 46.8666125813 -1.02216086186

```

Récupération de la base de données finale

Afin d'avoir une base de données accessible à tous, nous avons décidé d'exporter notre base de données après traitement en format csv. Cette étape n'est pas indispensable mais elle permet de créer une base de données "propre".

```
write.table(df_festivals, "donnees_festivals.csv", sep=";")
```

Quelques graphiques

Tout d'abord, nous allons pouvoir visualiser les régions avec le plus d'évènements.

```

# Calculer les fréquences de chaque région
freq <- table(df_festivals$region)

# Créer un dataframe avec les fréquences de chaque région
df_regions <- data.frame(table(df_festivals$region)) |>
  rename(Region = Var1, Frequence = Freq) |>

```

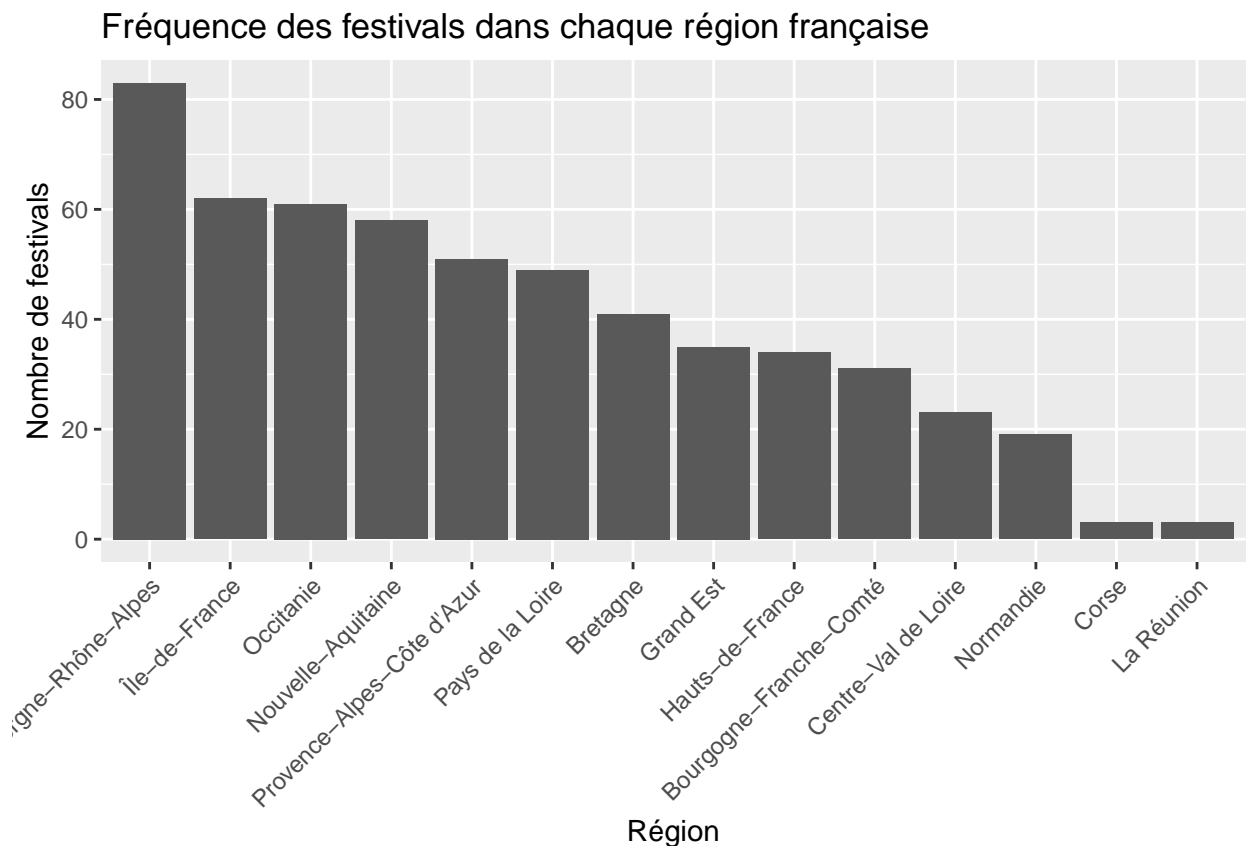
```
mutate(Pourcentage = round(Frequence / sum(Frequence) * 100, 1)) |>
  arrange(-Frequence) # on trie selon le nombre de festivals dans la région
head(df_regions,3)
```

```
##           Region Frequence Pourcentage
## 1 Auvergne-Rhône-Alpes      83        15.0
## 2 Île-de-France            62        11.2
## 3 Occitanie                61        11.0
```

Ce tableau nous présente les trois régions organisant le plus de festivals de musique électroniques ou amplifiées. En première place, nous retrouvons la région Auvergne-Rhône-Alpes avec 83 festivals mis en place chaque année. Ensuite, l'Île de France en seconde place avec 62 festivals et l'Occitanie avec 61 festivals.

Ce diagramme en barres nous permet d'avoir une information plus exhaustive au du nombre de festivals annuels dans chaque région de France.

```
ggplot(df_regions, aes(x = reorder(Region,-Frequence), y = Frequence)) +
  geom_bar(stat = "identity") +
  labs(title = "Fréquence des festivals dans chaque région française", x = "Région", y = "Nombre de festivals") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  guides(fill=FALSE)
```



Nous allons maintenant représenter le nombre de festivals par département.

```
# Création de la base de données comportant tous les départements
# de France ainsi que leurs caractéristiques
dpt <- read_sf("dpt")
```

```
# Création d'un tableau de fréquence des festivals par département
# freq <- table(df_festivals$departement)
```

```
# Création d'un dataframe avec le nombre de festivals par département
```

```
df_depart <- data.frame(table(df_festivals$departement)) |>
```

```
  rename(ID_GEOFLA = Var1, Frequence = Freq)|> # renommage des colonnes
```

```
  mutate(Pourcentage = round(Frequence / sum(Frequence) * 100, 1)) # création d'une variable en pourcentage
```

```
head(df_depart,3)
```

```
##   ID_GEOFLA Frequence Pourcentage
## 1         01          6          1.1
## 2         02          2          0.4
## 3         03          3          0.5
```

```
# Création du dataframe final pour réaliser la carte
```

```
france_map_count <- merge(dpt, df_depart, by.x = "CODE_DEPT", by.y = "ID_GEOFLA", all.x = TRUE)
head(france_map_count,3)
```

```
## Simple feature collection with 3 features and 13 fields
```

```
## Geometry type: MULTIPOLYGON
```

```
## Dimension: XY
```

```
## Bounding box: xmin: 644570 ymin: 6504682 xmax: 943396 ymax: 6997000
```

```
## Projected CRS: RGF93 v1 / Lambert-93
```

```
##   CODE_DEPT ID_GEOFLA NOM_DEPT CODE_CHF      NOM_CHF X_CHF_LIEU Y_CHF_LIEU
## 1         01          1      AIN      053 BOURG-EN-BRESSE    8717    65696
## 2         02          2     AISNE      408          LAON     7451    69406
## 3         03          3    ALLIER      190      MOULINS     7254    66072
```

```
##   X_CENTROID Y_CENTROID CODE_REG  NOM_REGION Frequence Pourcentage
## 1       8814      65582       82 RHONE-ALPES          6          1.1
## 2       7404      69401       22  PICARDIE          2          0.4
## 3       7144      65882       83  AUVERGNE          3          0.5
```

```
##                               geometry
```

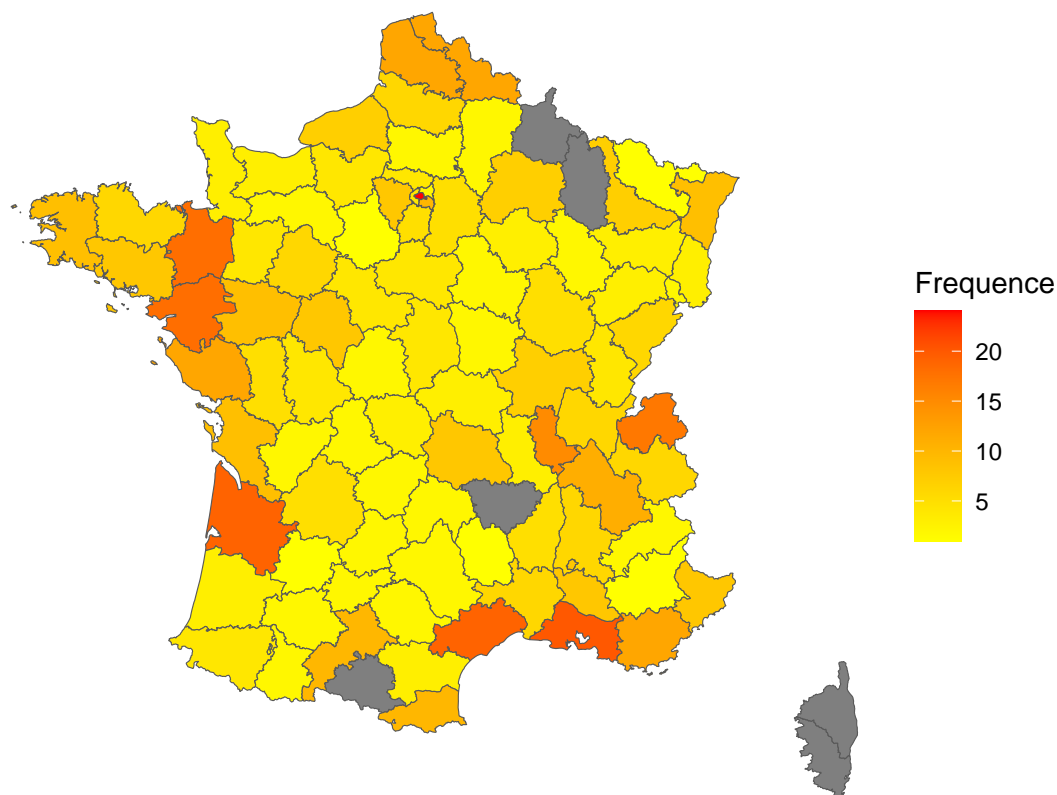
```
## 1 MULTIPOLYGON (((919195 6541...
```

```
## 2 MULTIPOLYGON (((735603 6861...
```

```
## 3 MULTIPOLYGON (((753769 6537...
```

```
# Carte
```

```
ggplot(france_map_count) + geom_sf(aes(fill=Frequence)) +
  scale_fill_continuous(low="yellow", high="red")+theme_void()
```



Cette carte est bien en adéquation avec les données de région que nous avons vu auparavant. Le région Auvergne-Rhône Alpes présente de nombreux festivals, tout comme l'Ile de France. Par ailleurs, nous pouvons aussi remarquer qu'en Ile-et-Vilaine de nombreux festivals ont aussi lieu !

N.B. : Cette carte montre qu'il y a quelques départements sans correspondance, nous avons décidé de ne pas en tenir compte.

Nous allons maintenant nous intéresser au nombre de festivals par mois :

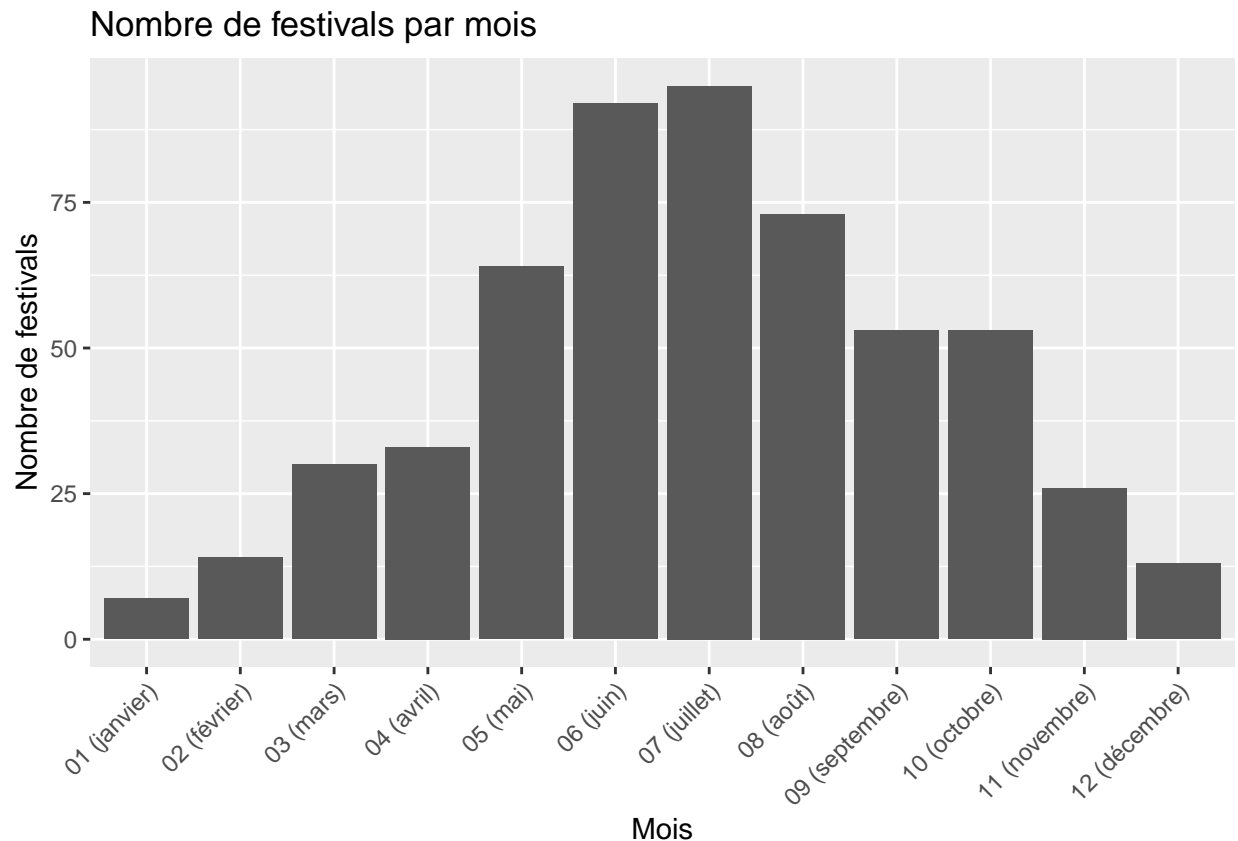
```
# Création d'un tableau avec le nombre de festivals par mois
freq <- table(df_festivals$mois_habituel_de_debut)

df_mois <- data.frame(table(df_festivals$mois_habituel_de_debut)) |>
  rename(Mois = Var1, Frequence = Freq) |>
  mutate(Pourcentage = round(Frequence / sum(Frequence) * 100, 1)) #Création d'une variable en pourcent
df_mois
```

##	Mois	Frequence	Pourcentage
## 1	01 (janvier)	7	1.3
## 2	02 (février)	14	2.5
## 3	03 (mars)	30	5.4
## 4	04 (avril)	33	6.0
## 5	05 (mai)	64	11.6
## 6	06 (juin)	92	16.6
## 7	07 (juillet)	95	17.2
## 8	08 (août)	73	13.2
## 9	09 (septembre)	53	9.6


```
## 10 10 (octobre)      53      9.6
## 11 11 (novembre)    26      4.7
## 12 12 (décembre)    13      2.4
```

```
ggplot(df_mois, aes(x = Mois, y = Frequence)) +
  geom_bar(stat = "identity") +
  labs(title = "Nombre de festivals par mois", x = "Mois", y = "Nombre de festivals") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  guides(fill=FALSE)
```



Nous observons que plus les beaux jours arrivent, plus le nombre de festivals augmente. Cette conclusion est bien cohérente avec nos données.

Données provenant d'un site web

Dans un second temps, nous avons choisi d'étudier deux sites web concernant les festivals les plus reconnus en France et un autre au sujet du festival le plus connu de France : l'ElectroBeach Festival.

Extraction des données via un site web

```
site_festivals <- "https://martinbeatz.com/festivals-electro-france/"
```

```
festival_html <- read_html(site_festivals)
```

Grâce à cette page web, nous allons pouvoir établir les festivals les plus en vogue. Nous avons choisi d'utiliser **xpath** pour se déplacer librement dans l'arbre.

```
listes_festivals <- festival_html |>
  html_nodes(xpath = "//h2") |>
  html_text()
# listes_festivals

cat("Les festivals les plus recommandés sont :", paste("\n", listes_festivals))
```

```
## Les festivals les plus recommandés sont :
## Electrobeach Festival
## Les Plages Electroniques
## Elektric Park
## Cocorico Electro
## Tomorrowland Winter
## Delta Festival
## Hope Festival - Toulouse
## Reperkusound
## Cercle Festival
## Stereoparc
## Touquet Music Beach
## Summer Festival
## Panoramas Festival
## Marvellous Festival
## I Love Techno
## Dream Nation
## Nuits Sonores
## Pharaonic Festival
```

Nous allons maintenant récupérer les différents lieux des festivals :

```
liste_lieux <- festival_html |>
  html_nodes(xpath = "//*[@id='post-2934']/div/div/div/p[position()=4 or position()=9 or position()=14 or position()=15]") |>
  html_text()
liste_lieux <- gsub("Lieu : ", "", liste_lieux)
liste_lieux
```

```
## [1] "Port- Barcarès 66420 - Pyrénées Orientales"
## [2] "Plage du Palais des Festivals - Cannes"
## [3] "Châtou - Ile des Impressionnistes"
## [4] "Château de la Ferté Saint-Aubin - Loiret"
## [5] "Alpes d'Huez - Isère"
## [6] "Marseille"
## [7] "Hippodrome de Toulouse"
## [8] "Villeurbanne - Rhône-Alpes"
## [9] "Paris-Le Bourget, Musée de l'Air et de l'Espace"
## [10] "Rochefort - Charente-Maritime"
## [11] "Le Touquet Paris Plage"
## [12] "Marseille"
```

Puis nous allons récupérer les différentes dates des festivals :

```
## [1] "14, 15 et 16 Juillet 2023" "4, 5 et 6 Août 2023"
## [3] "1 et 2 Septembre 2023"    "13, 14 et 15 Juillet 2023"
## [5] "18 au 24 Mars 2023"       "23 au 27 Août 2023"
## [7] "2021"                     "8, 8 et 9 Avril 2023"
## [9] "11 & 12 Septembre 2021"   "21 et 22 Juillet 2023"
## [11] "25 et 26 Août 2023"       "10 et 11 Septembre 2021"
## [13] "22 au 24 Septembre 2023"  "27 et 28 Mai 2023"
## [15] "7, 8 et 9 Avril 2023"     "17, 18, 19 Septembre 2021"
## [17] "19 au 25 Juillet 2021"     "26 Mars 2022"
```

```
for(i in seq_along(listes_festivals)){
  cat("Le festival", listes_festivals[[i]], "aura lieu les", liste_dates[[i]], "à", liste_lieux[[i]],
}
```

Zoom sur le Electrobeach Music Festival

11

```
# url du site web
site_electro_beach <- "https://fr.wikipedia.org/wiki/Electrobeach_Music_Festival"
```

```
electro_beach_html <- read_html(site_electro_beach)
```

L'histoire de l'Electro Beach Festival

Nous allons maintenant chercher à en savoir un peu plus sur ce festival, en commençant par son histoire :

```
electro_beach_html |>
  html_nodes(xpath = "/html/body/div[2]/div/div[3]/main/div[3]/div[3]/div[1]/p[3]") |>
  html_text()
```

```
## [1] "Le festival est créé en 2009 par Alain Ferrand.\nIl attire en 2012 40 000 festivaliers. Le fest."
```

Lorsque ce festival a été créé, il était d'une plus petite envergure et d'une durée plus courte qu'actuellement. C'est ce que nous précise ce paragraphe :

```
electro_beach_html |>
  html_nodes(xpath = "/html/body/div[2]/div/div[3]/main/div[3]/div[3]/div[1]/p[4]") |>
  html_text()
```

```
## [1] "En 2013, l'événement dure deux jours, et dispose d'une seule grande scène d'une surface de 300 m"
```

Le festival a eu lieu pour la dernière fois en 2019, avant la pandémie du Covid-19. Sur cette page web, nous pouvons voir un article dédié à ce sujet.

```
electro_beach_html |>
  html_nodes(xpath = "/html/body/div[2]/div/div[3]/main/div[3]/div[3]/div[1]/p[8]") |>
  html_text()
```

```
## [1] "Le 8 avril 2020, le maire du Barcarès, et propriétaire du festival, Alain Ferrand, annonce l'an"
```

La dernière édition a accueilli de nombreux festivaliers :

```
electro_beach_html |>
  html_nodes(xpath = "/html/body/div[2]/div/div[3]/main/div[3]/div[3]/div[1]/ul[1]/li[9]") |>
  html_text()
```

```
## [1] "2019 : 165 000 festivaliers[37]."
```

Cependant, nous pouvons observer que certaines éditions n'ont pas eu lieu à cause de la crise sanitaire.

```
electro_beach_html |>
  html_nodes(xpath = "/html/body/div[2]/div/div[3]/main/div[3]/div[3]/div[1]/ul[1]/li[10]") |>
  html_text()
```

```
## [1] "2020 et 2021 ont été annulées en raison de la Pandémie de Covid-19[21]."
```

Nous pouvons visualiser l'évolution du nombre de festivaliers à l'Electro Beach Club.

```
freq <- electro_beach_html |>
  html_nodes("ul:first-of-type li") |>
  html_text() |>
  str_extract("\\d+\\s\\d+") |>
  na.omit() # suppression des données manquantes
frequentation <- freq[1:9]
annees <- 2011:2019

df <- data.frame(annees,frequentation) |>
  rename(Annee = annees, Frequentation = frequentation)
df
```

```
##   Année Frequentation
## 1  2011         30 000
## 2  2012         40 000
## 3  2013         60 000
## 4  2014         80 000
## 5  2015        140 000
## 6  2016        176 000
## 7  2017        192 000
## 8  2018        165 000
## 9  2019        165 000
```

Nous pouvons constater que le nombre de festivaliers le plus important est en 2017. De plus, nous observons une forte évolution entre sa création et les dernières éditions !

Les différents logos de l'Electro Beach Music Festival

N.B. : Les logos présentent une qualité médiocre due aux différents traitements réalisés. De plus, leur dimension est assez petite sur wikipédia, leur agrandissement a donc des conséquences sur leur visualisation.

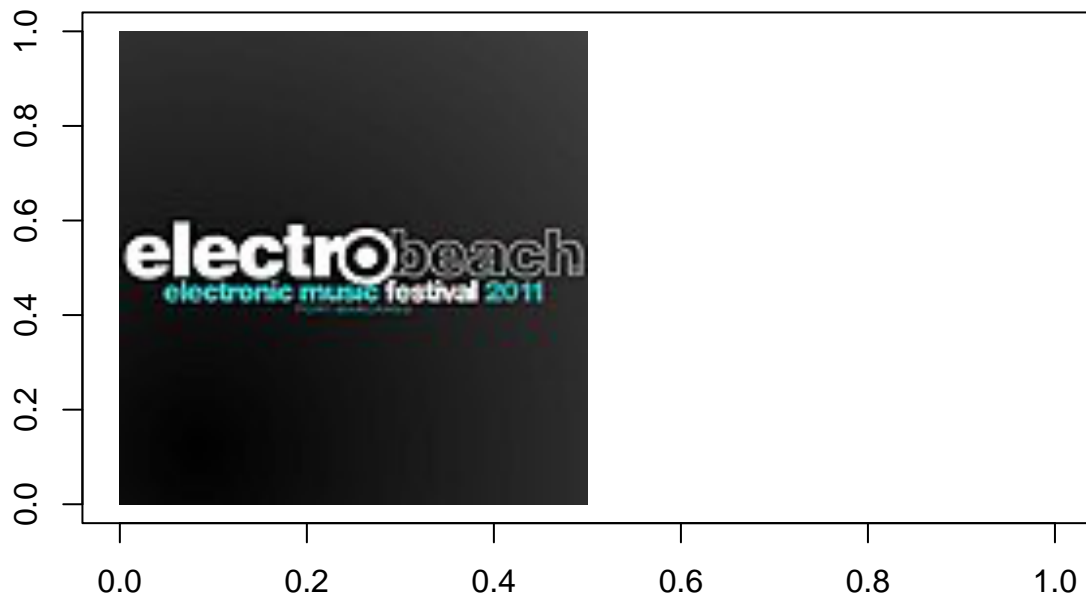
Nous avons choisi de représenter les différents logos du festivals, ici respectivement les années 2011, 2012, 2013, 2014 et 2016 (qui est encore le logo actuel du festival).

```
logo1 <- electro_beach_html |>
  html_nodes(xpath = "//*[@id='mw-content-text']/div[1]/ul[2]/li[1]/div/div[1]/div/a/img") |>
  html_attr("src") # récupération du lien de l'image

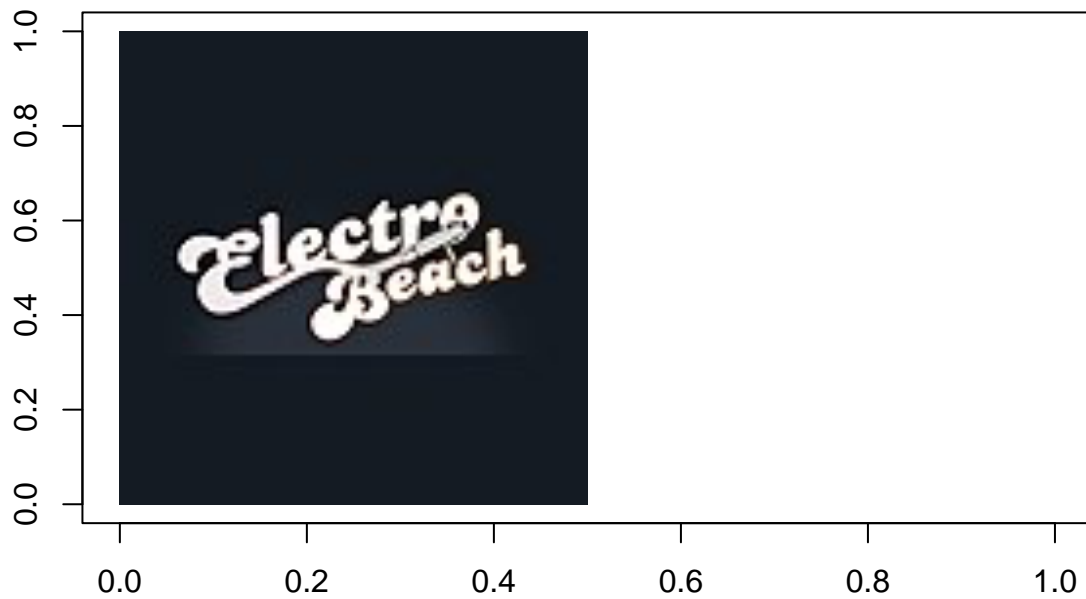
# download.file(paste0("http:", logo1), destfile = "logo_electrobeach.jpg") # enregistrement du logo sur disque

img <- readJPEG("logo_electrobeach.jpg") # lecture de l'image

plot(0, 0, type = "n", xlim = c(0, 1), ylim = c(0, 1), xlab = "", ylab = "") # création d'un graphique
rasterImage(img, 0, 0, 0.5, 1) # position de l'image sur le graphique
```



```
logo2 <- electro_beach_html |>
  html_nodes(xpath = "//*[@id='mw-content-text']/div[1]/ul[2]/li[2]/div/div[1]/div/a/img") |>
  html_attr("src")
# download.file(paste0("http:", logo2), destfile = "logo_electrobeach2.jpg")
img <- readJPEG("logo_electrobeach2.jpg")
plot(0, 0, type = "n", xlim = c(0, 1), ylim = c(0, 1), xlab = "", ylab = "")
rasterImage(img, 0, 0, 0.5, 1)
```

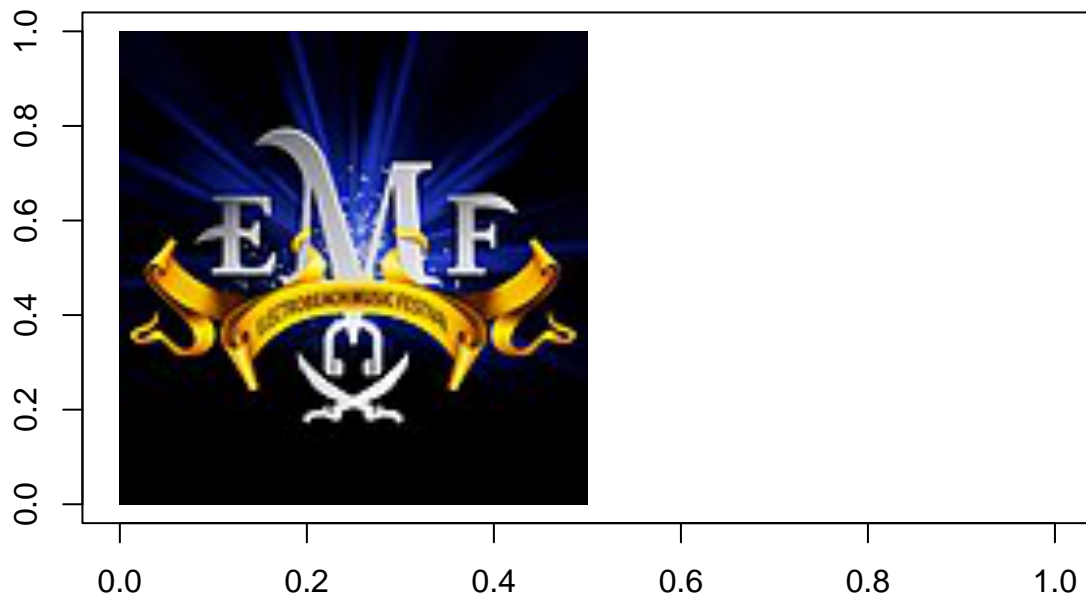


```
logo3 <- electro_beach_html |>
  html_nodes(xpath = "//*[@id='mw-content-text']/div[1]/ul[2]/li[3]/div/div[1]/div/a/img") |>
  html_attr("src")

# download.file(paste0("http:", logo3), destfile = "logo_electrobeach3.jpg")

img <- readJPEG("logo_electrobeach3.jpg")

plot(0, 0, type = "n", xlim = c(0, 1), ylim = c(0, 1), xlab = "", ylab = "")
rasterImage(img, 0, 0, 0.5, 1)
```



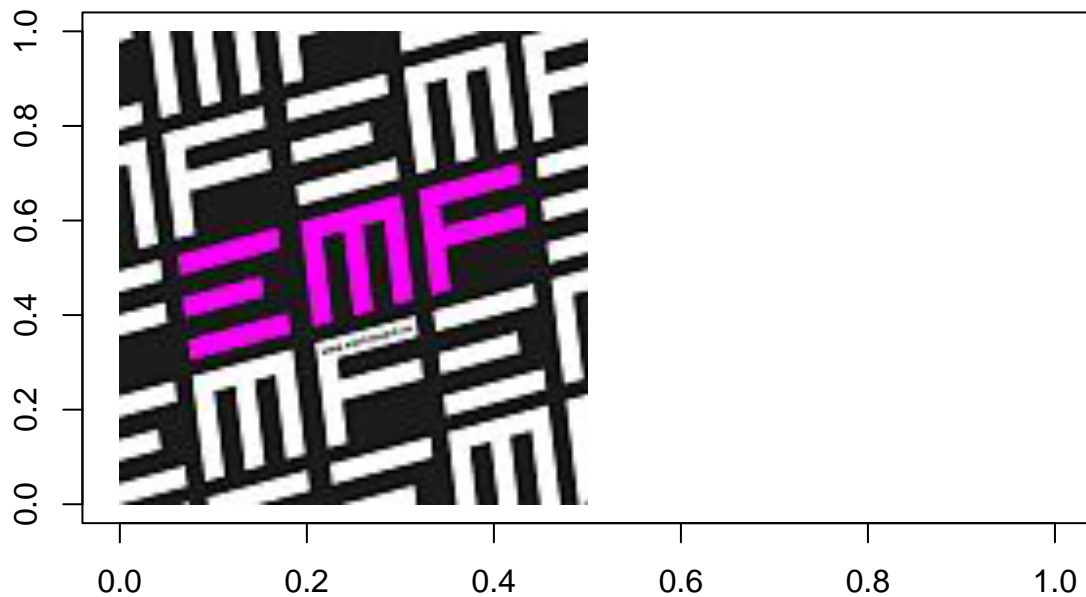
```
logo4 <- electro_beach_html |>
  html_nodes(xpath = "//*[ @id='mw-content-text']/div[1]/ul[2]/li[4]/div/div[1]/div/a/img") |>
  html_attr("src")
logo4
```

```
## [1] "//upload.wikimedia.org/wikipedia/commons/thumb/5/5f/Logo_Electrobeach_2014.jpg/120px-Logo_Electr
```

```
# download.file(paste0("http:", logo4), destfile = "logo_electrobeach4.jpg")

img <- readJPEG("logo_electrobeach4.jpg")

plot(0, 0, type = "n", xlim = c(0, 1), ylim = c(0, 1), xlab = "", ylab = "")
rasterImage(img, 0, 0, 0.5, 1)
```

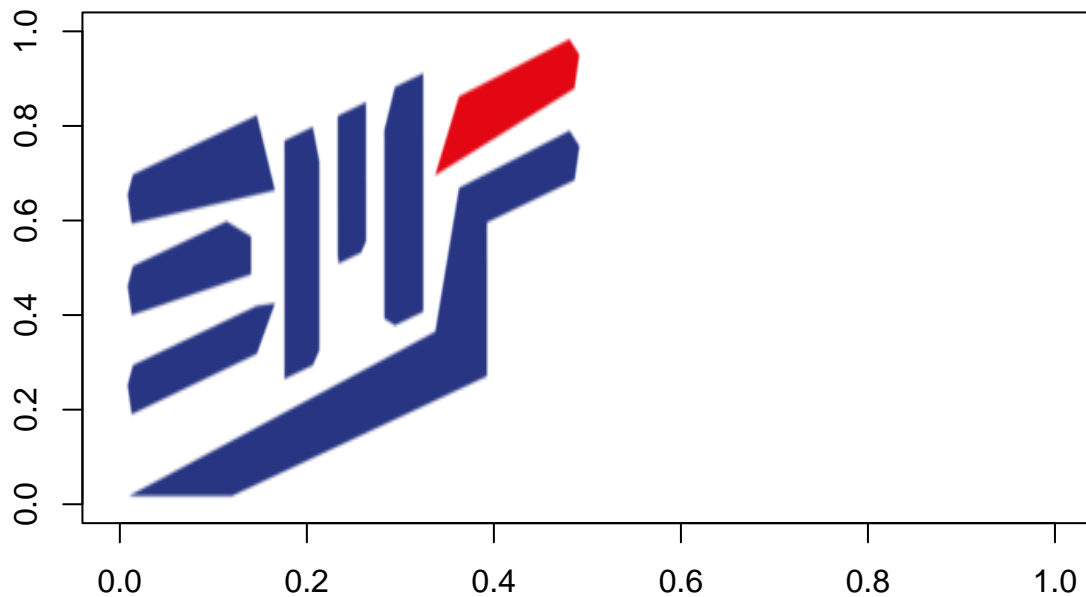
Le logo de l'année 2015 n'est pas disponible sur le site de wikipédia. Cependant, nous avons trouvé pertinent le fait de montrer le logo actuel du festival, en place depuis 2016.

```
electrobeach_site <- "https://www.electrobeach.com" # site web où se trouve le logo
electro_beach_site_html <- read_html(electrobeach_site)
logo2016 <- electro_beach_site_html |>
  html_nodes(xpath = "//*[@id='header_main']/div/div/span/a/img") |>
  html_attr("src")

# download.file(paste0(logo2016), destfile = "logo_electrobeach2016.png")

img <- readPNG("logo_electrobeach2016.png")

plot(0, 0, type = "n", xlim = c(0, 1), ylim = c(0, 1), xlab = "", ylab = "")
rasterImage(img, 0, 0, 0.5, 1)
```



Mise en relation entre le JSON et le site des festivals

Nous pouvons maintenant mettre en lien le fichier JSON et le web-scraping réalisé sur les sites web. En effet, nous allons chercher les festivals en commun entre ces deux sources. Pour y parvenir, nous devons mettre les champs dans le même format.

```
# Mise en minuscules des champs
listes_festivals_low <- str_to_lower(listes_festivals)
listes_festivals_low
```

```
## [1] "electrobeach festival"    "les plages electroniques"
## [3] "elektric park"           "cocorico electro"
## [5] "tomorrowland winter"     "delta festival"
## [7] "hope festival - toulouse" "reperkusound"
## [9] "cercle festival"         "stereoparc"
## [11] "touquet music beach"     "summer festival"
## [13] "panoramas festival"      "marvellous festival"
## [15] "i love techno"           "dream nation"
## [17] "nuits sonores"           "pharaonic festival"
```

```
# Mise en minuscules des champs
df_festivals_low <- str_to_lower(df_festivals$nom_de_la_manifestation)
head(df_festivals_low)
```

```
## [1] "astropolis l'hiver"      "this is england"
## [3] "free sons divers"       "snowboxx"
## [5] "radiomeuh circus festival" "festival off kultur"

# Intersections des deux sources :
festivals_communs <- intersect(df_festivals_low, listes_festivals_low)
cat("Les festivals qui sont présents dans le JSON et dans la liste des festivals conseillés par le site i

## Les festivals qui sont présents dans le JSON et dans la liste des festivals conseillés par le site i
## delta festival
## reperkusound
## nuits sonores
## touquet music beach
## dream nation
```

Conclusion

Finalement, ce projet nous a permis de traiter des données de natures différentes. Ce projet a été d'autant plus agréable que nous avons choisi nos propres données, les festivals de musique électroniques et amplifiées. Les données étant variées, nous avons pu les illustrer à travers des graphiques, une carte, la récupération d'images... Le projet se déclinant sur plusieurs échelles nous a permis de ne pas rendre le projet monotone et d'explorer plusieurs approches. En espérant vous avoir donné envie d'aller danser sur des musiques électroniques (comme nous l'avons vu, beaucoup de festivals se trouvent en Ile-et-Vilaine et la saison des beaux jours arrive).