

Licence Professionnelle : Modélisation et Gestion de l'Environnement

Programmation R pour l'Environnement

Pr. Lamia ZIAD

École Supérieure de Technologie d'Essaouira

Table des matières

1	Introduction au Langage R pour l'Environnement	3
1.1	Pourquoi R en Sciences Environnementales?	3
1.2	Installation et Interface	3
1.3	Concepts de Base	4
2	Types de Données et Structures en Écologie	5
2.1	Types de Données de Base	5
2.2	Vecteurs - Collections de Données	6
2.3	Matrices - Données Multidimensionnelles	6
2.4	Data Frames - Tables de Données Environnementales	7
3	Importation et Manipulation de Données Environnementales	8
3.1	Importation de Données	8
3.2	Manipulation avec dplyr	9
3.3	Gestion des Données Manquantes	10
4	Statistiques Descriptives pour les Données Environnementales	11
4.1	Mesures de Tendances Centrales	11
4.2	Analyses par Groupes	12
4.3	Analyses de Corrélation Environnementale	14
5	Visualisation avec ggplot2 pour l'Environnement	15
5.1	Graphiques de Base	15
5.2	Graphiques pour l'Écologie des Communautés	16
6	Analyses Écologiques avec le Package vegan	17
6.1	Analyses de Diversité	17
6.2	Ordinations	18

7 Modélisation Environnementale avec R	20
7.1 Régression Linéaire et Multiple	20
7.2 Modèles de Présence/Absence	21
8 Analyses de Séries Temporelles Environnementales	22
8.1 Séries Temporelles de Base	22
8.2 Analyses de Tendance	23

Chapitre 1

Introduction au Langage R pour l'Environnement

Definition

R est un langage de programmation et un environnement logiciel spécialisé dans l'analyse statistique et la visualisation de données. Il est particulièrement adapté aux sciences environnementales.

1.1 Pourquoi R en Sciences Environnementales ?

- **Gratuit et Open Source** : Accessible à tous
- **Puissant pour les statistiques** : Analyses complexes
- **Visualisation avancée** : Graphiques de qualité publication
- **Communauté active** : Packages spécialisés en écologie
- **Reproductibilité** : Scripts réutilisables

1.2 Installation et Interface

Important

Téléchargez R depuis CRAN (cran.r-project.org) et RStudio depuis rstudio.com

Exemple

Listing 1.1 – Premier programme en R

```
# Installation des packages pour l'environnement
install.packages(c("vegan", "ggplot2", "dplyr"))

# Chargement des packages
library(vegan)
library(ggplot2)
library(dplyr)

# Message de bienvenue
print("Bienvenue en Modelisation Environnementale avec R!")
```

1.3 Concepts de Base

Application Ecologique

En écologie, nous travaillons souvent avec des données de biodiversité, des paramètres physico-chimiques, et des séries temporelles environnementales.

Chapitre 2

Types de Données et Structures en Écologie

2.1 Types de Données de Base

Important

R dispose de plusieurs types de données fondamentaux pour représenter les informations environnementales.

Exemple

Listing 2.1 – Types de données environnementales

```
# Donnees quantitatives (continues)
temperature <- 25.7
pH <- 7.2
profondeur <- 15.3

# Donnees qualitatives (categorielles)
type_habitat <- "foret"
saison <- factor(c("printemps", "ete", "automne", "hiver"))

# Donnees de presence/absence
presence_espece <- TRUE
```

2.2 Vecteurs - Collections de Données

Exemple

Listing 2.2 – Creation de vecteurs environnementaux

```
# Concentrations de nutriments dans l'eau (mg/L)
nitrates <- c(2.1, 3.4, 1.8, 4.2, 2.9)
phosphates <- c(0.3, 0.5, 0.2, 0.6, 0.4)

# Abondance d'espèces sur différents sites
abondance_oiseaux <- c(12, 25, 8, 30, 15)
noms_sites <- c("Site_A", "Site_B", "Site_C", "Site_D", "Site_E")

# Coordonnées géographiques
latitude <- c(31.5, 32.1, 30.8, 33.2, 31.9)
longitude <- c(-9.6, -8.7, -10.2, -7.9, -9.1)
```

2.3 Matrices - Données Multidimensionnelles

Application Ecologique

Les matrices sont idéales pour représenter des communautés d'espèces (sites en lignes, espèces en colonnes).

Exemple

Listing 2.3 – Matrice communauté-espèces

```
# Creation d'une matrice d'abondance d'espèces
especes_abondance <- matrix(
  c(10, 5, 0, 25, 15,
    8, 12, 3, 30, 10,
    15, 8, 2, 20, 18,
    5, 20, 1, 35, 8),
  nrow = 4, ncol = 5,
  byrow = TRUE,
  dimnames = list(
    c("Station_1", "Station_2", "Station_3", "Station_4"),
    c("Espece_A", "Espece_B", "Espece_C", "Espece_D", "Espece_E")
  )
)

print(especes_abondance)
```

2.4 Data Frames - Tables de Données Environnementales**Exemple**

Listing 2.4 – Data frame de données de terrain

```
# Creation d'un data frame pour des données d'échantillonnage
donnees_terrain <- data.frame(
  site = c("Foret_Nord", "Foret_Sud", "Prairie", "Zone_humide"),
  superficie_ha = c(50, 35, 20, 15),
  altitude = c(450, 320, 280, 190),
  temperature_moyenne = c(18.5, 20.2, 22.1, 19.8),
  precipitation_annuelle = c(800, 750, 600, 900),
  richesse_specifique = c(45, 38, 28, 52)
)

# Affichage des données
print(donnees_terrain)
summary(donnees_terrain)
```

Chapitre 3

Importation et Manipulation de Données Environnementales

3.1 Importation de Données

Important

R peut lire de nombreux formats de données environnementales : CSV, Excel, shapefiles, données climatiques, etc.

Exemple

Listing 3.1 – Importation de données environnementales

```
# Importation depuis un fichier CSV
library(readr)
donnees_eau <- read_csv("qualite_eau.csv")

# Importation de données Excel
library(readxl)
donnees_sol <- read_excel("analyses_sol.xlsx")

# Importation de shapefiles (données spatiales)
library(sf)
limites_parc <- st_read("limites_parc_national.shp")
```

3.2 Manipulation avec dplyr

Application Ecologique

Le package dplyr permet de filtrer, sélectionner et transformer facilement des données écologiques.

Exemple

Listing 3.2 – Manipulation de données écologiques avec dplyr

```
library(dplyr)

# Données d'inventaire forestier
inventaire_forestier <- data.frame(
  parcelle = rep(1:10, each = 5),
  essence = sample(c("Chene", "Hêtre", "Pin", "Sapin"), 50, replace =
    TRUE),
  diametre = runif(50, 10, 80),
  hauteur = runif(50, 5, 35),
  statut = sample(c("Vivant", "Mort", "Malade"), 50, replace = TRUE)
)

# Opérations de base
# Filtrer les arbres vivants de plus de 30 cm
arbres_vivants <- inventaire_forestier %>%
  filter(statut == "Vivant", diametre > 30)

# Sélectionner des colonnes spécifiques
donnees_essentielles <- inventaire_forestier %>%
  select(parcelle, essence, diametre)

# Grouper par essence et calculer les statistiques
statistiques_essences <- inventaire_forestier %>%
  group_by(essence) %>%
  summarise(
    nb_arbres = n(),
    diametre_moyen = mean(diametre),
    hauteur_moyenne = mean(hauteur),
    diametre_max = max(diametre)
)
```

3.3 Gestion des Données Manquantes

Exemple

Listing 3.3 – Traitement des données manquantes en écologie

```
# Donnees avec valeurs manquantes (frequent en ecologie)
donnees_meteo <- data.frame(
  date = seq(as.Date("2023-01-01"), as.Date("2023-01-10"), by = "day"),
  temperature = c(15.2, 16.8, NA, 14.5, 17.2, NA, 16.1, 15.8, 14.9,
    16.5),
  precipitation = c(0, 2.5, 1.8, 0, NA, 5.2, 0.8, 0, 3.1, 0.2),
  humidite = c(65, 72, 68, NA, 70, 75, 78, 65, 80, 72)
)

# Identifier les donnees manquantes
sum(is.na(donnees_meteo))

# Supprimer les lignes avec donnees manquantes
donnees_completes <- na.omit(donnees_meteo)

# Imputer les valeurs manquantes (moyenne)
donnees_meteo$temperature[is.na(donnees_meteo$temperature)] <-
  mean(donnees_meteo$temperature, na.rm = TRUE)
```

Chapitre 4

Statistiques Descriptives pour les Données Environnementales

4.1 Mesures de Tendances Centrales

Important

Les statistiques descriptives aident à résumer et comprendre les patterns dans les données environnementales.

Exemple

Listing 4.1 – Statistiques descriptives de base

```
# Donnees de qualite d'eau
parametres_eau <- data.frame(
  site = paste0("Site_", 1:12),
  pH = runif(12, 6.5, 8.5),
  conductivite = runif(12, 200, 800),
  oxygene_dissous = runif(12, 5, 12),
  turbidite = runif(12, 1, 50)
)

# Statistiques descriptives de base
summary(parametres_eau)

# Mesures de tendance centrale
moyennes <- colMeans(parametres_eau[, -1])
mediane_pH <- median(parametres_eau$pH)
ecart_type_oxygene <- sd(parametres_eau$oxygene_dissous)
```

4.2 Analyses par Groupes

Application Ecologique

En écologie, il est fréquent de comparer des paramètres entre différents habitats ou types d'écosystèmes.

Exemple

Listing 4.2 – Comparaison entre habitats

```
# Donnees de biodiversite par type d'habitat
biodiversite <- data.frame(
  habitat = rep(c("Foret", "Prairie", "Zone_humide"), each = 8),
  richesse_especes = c(
    rnorm(8, 45, 5),      # Foret
    rnorm(8, 25, 4),      # Prairie
    rnorm(8, 35, 6)       # Zone humide
  ),
  abondance_totale = c(
    rnorm(8, 120, 15),
    rnorm(8, 80, 10),
    rnorm(8, 150, 20)
  )
)

# Statistiques par habitat
library(dplyr)
statistiques_habitat <- biodiversite %>%
  group_by(habitat) %>%
  summarise(
    richesse_moyenne = mean(richesse_especes),
    richesse_sd = sd(richesse_especes),
    abondance_moyenne = mean(abondance_totale),
    abondance_sd = sd(abondance_totale)
  )

print(statistiques_habitat)
```

4.3 Analyses de Corrélation Environnementale

Exemple

Listing 4.3 – Analyses de correlation en ecologie

```
# Matrice de correlation entre parametres environnementaux
matrice_correlation <- cor(parametres_eau[, -1])
print(matrice_correlation)

# Test de correlation specifique
cor_test <- cor.test(parametres_eau$pH, parametres_eau$oxygene_dissous)
print(cor_test)
```

Chapitre 5

Visualisation avec ggplot2 pour l'Environnement

5.1 Graphiques de Base

Important

ggplot2 est le package de référence pour créer des visualisations professionnelles en R.

Exemple

Listing 5.1 – Graphiques environnementaux de base

```
library(ggplot2)

# Donnees de monitoring de population
donnees_population <- data.frame(
  annee = 2010:2023,
  espece_A = c(120, 115, 125, 130, 128, 135, 140, 138, 145, 150, 148,
             155, 160, 158),
  espece_B = c(80, 85, 82, 78, 75, 72, 70, 68, 65, 62, 60, 58, 55,
             52)
)

# Graphique lineaire simple
ggplot(donnees_population, aes(x = annee)) +
  geom_line(aes(y = espece_A, color = "Espece_A")) +
  geom_line(aes(y = espece_B, color = "Espece_B")) +
  labs(title = "Evolution des Populations",
       x = "Annee", y = "Nombre d'individus") +
  theme_minimal()
```

5.2 Graphiques pour l'Écologie des Communautés

Application Ecologique

Les diagrammes en barres et boxplots sont essentiels pour comparer la biodiversité entre habitats.

Exemple

Listing 5.2 – Comparaison visuelle entre habitats

```
# Boxplot pour comparer les distributions
ggplot(biodiversite, aes(x = habitat, y = richesse_especes, fill =
  habitat)) +
  geom_boxplot() +
  labs(title = "Distribution de la Richesse Spécifique",
       x = "Habitat", y = "Nombre d'Espèces") +
  theme_minimal()
```

Chapitre 6

Analyses Écologiques avec le Package vegan

6.1 Analyses de Diversité

Important

Le package vegan est spécialisé dans les analyses écologiques numériques.

Exemple

Listing 6.1 – Analyses de diversité avec vegan

```
library(vegan)

# Données de communauté (sites x espèces)
communaute <- matrix(
  c(10, 5, 0, 8, 12,
    15, 8, 2, 6, 10,
    8, 12, 3, 4, 15,
    5, 20, 1, 2, 8,
    12, 6, 0, 10, 18),
  nrow = 5, ncol = 5,
  dimnames = list(
    paste0("Site_", 1:5),
    paste0("Espece_", LETTERS[1:5]))
)
)

# Indices de diversité alpha
diversite_alpha <- diversity(communaute, index = "shannon")
richesse_alpha <- specnumber(communaute)

# Tableau des résultats
resultats_diversite <- data.frame(
  Site = rownames(communaute),
  Richesse = richesse_alpha,
  Shannon = diversite_alpha
)

print(resultats_diversite)
```

6.2 Ordinations

Application Ecologique

Les analyses d'ordination permettent de visualiser les similarités entre sites basées sur leur composition en espèces.

Exemple

Listing 6.2 – Analyses d’ordination en ecologie

```
# Analyse des correspondances (CA)
ordination_ca <- cca(communaute)
summary(ordination_ca)
plot(ordination_ca)

# Analyse des composantes principales (PCA)
donnees_env <- data.frame(
  temperature = c(18, 22, 20, 19, 21),
  precipitation = c(800, 600, 750, 850, 700)
)

pca_env <- rda(donnees_env, scale = TRUE)
summary(pca_env)
```

Chapitre 7

Modélisation Environnementale avec R

7.1 Régression Linéaire et Multiple

Important

Les modèles de régression permettent de quantifier les relations entre variables environnementales.

Exemple

Listing 7.1 – Modèles de régression en écologie

```
# Relation entre température et diversité
modele_lineaire <- lm(richesse_spécifique ~ température_moyenne,
                      data = données_terrain)
summary(modele_lineaire)

# Régression multiple
modele_multiple <- lm(richesse_spécifique ~ température_moyenne +
                       précipitation_annuelle + altitude,
                       data = données_terrain)
summary(modele_multiple)

# Diagnostic du modèle
par(mfrow = c(2, 2))
plot(modele_multiple)
par(mfrow = c(1, 1))
```

7.2 Modèles de Présence/Absence

Application Ecologique

Les modèles logistiques sont adaptés pour prédire la présence ou l'absence d'espèces.

Exemple

Listing 7.2 – Modele logistique pour presence/absence

```
# Donnees de presence/absence
donnees_presence <- data.frame(
  site = paste0("Site_", 1:50),
  temperature = runif(50, 15, 25),
  humidite = runif(50, 40, 90),
  presence = rbinom(50, 1, 0.6)
)

# Modele logistique
modele_logistique <- glm(presence ~ temperature + humidite,
                           data = donnees_presence,
                           family = binomial(link = "logit"))
summary(modele_logistique)
```

Chapitre 8

Analyses de Séries Temporelles Environnementales

8.1 Séries Temporelles de Base

Important

Les séries temporelles sont essentielles pour étudier les changements environnementaux.

Exemple

Listing 8.1 – Manipulation de series temporelles environnementales

```
# Creation d'une serie temporelle de temperature
dates <- seq(as.Date("2010-01-01"), as.Date("2020-12-31"), by = "
month")
temperature <- 15 + 10 * sin(2 * pi * as.numeric(format(dates, "%m"))
/12) +
rnorm(length(dates), 0, 2) +
0.05 * (1:length(dates))

serie_temporelle <- ts(temperature, start = c(2010, 1), frequency =
12)

# Analyse de base
plot(serie_temporelle, main = "Evolution de la Temperature", ylab = "
Degres C")

# Decomposition
decomposition <- decompose(serie_temporelle)
plot(decomposition)
```

8.2 Analyses de Tendance

Application Ecologique

La détection de tendances est cruciale pour évaluer les impacts des changements climatiques.

Exemple

Listing 8.2 – Analyses de tendance environnementale

```
# Test de Mann-Kendall pour tendance
library(trend)
test_mk <- mk.test(temperature)
print(test_mk)

# Regression pour estimation de pente
modele_tendance <- lm(temperature ~ time(serie_temporelle))
summary(modele_tendance)
```