Analise_Flood_Sentinel

Análise Exploratória e Explanatória em R para o conjunto "flood.csv"

Instalar pacotes

```
Installing packages into 'C:/Users/JonasLuisdaSilva/AppData/Local/R/win-library/4.5'
(as 'lib' is unspecified)
```

```
package 'tidyverse' successfully unpacked and MD5 sums checked
package 'GGally' successfully unpacked and MD5 sums checked
package 'corrplot' successfully unpacked and MD5 sums checked
package 'knitr' successfully unpacked and MD5 sums checked
package 'gridExtra' successfully unpacked and MD5 sums checked
package 'Metrics' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
    C:\Users\JonasLuisdaSilva\AppData\Local\Temp\Rtmp6dfTM1\downloaded_packages
```

Carregar as bibliotecas

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.2     v tibble    3.2.1
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.0.4
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to beco
```

```r
library(GGally)
```

```
Registered S3 method overwritten by 'GGally':
  method from
  +.gg   ggplot2
```

```r
library(corrplot)
```

```
corrplot 0.95 loaded
```

```r
library(knitr)
```

Carregamento dos Dados

```r
df <- read_csv("flood.csv", col_types = cols())
```

Verificar dimensões e colunas

```r
cat("Dimensões do dataset:\n")
```

```
Dimensões do dataset:
```

```r
dim(df)
```

```
[1] 50000    21
```

```r
cat("\nNomes das colunas:\n")
```

```

Nomes das colunas:
```

```r
names(df)
```

```
 [1] "MonsoonIntensity"               "TopographyDrainage"
 [3] "RiverManagement"                "Deforestation"
 [5] "Urbanization"                   "ClimateChange"
 [7] "DamsQuality"                    "Siltation"
 [9] "AgriculturalPractices"          "Encroachments"
[11] "IneffectiveDisasterPreparedness" "DrainageSystems"
[13] "CoastalVulnerability"           "Landslides"
[15] "Watersheds"                     "DeterioratingInfrastructure"
[17] "PopulationScore"                "WetlandLoss"
[19] "InadequatePlanning"             "PoliticalFactors"
[21] "FloodProbability"
```

Estrutura geral e tipos de cada coluna

```
cat("\nEstrutura (str) do dataset:\n")
```

```
Estrutura (str) do dataset:
```

```
str(df)
```

```
spc_tbl_ [50,000 x 21] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ MonsoonIntensity               : num [1:50000] 3 8 3 4 3 6 6 7 6 4 ...
 $ TopographyDrainage             : num [1:50000] 8 4 10 4 7 6 7 3 3 3 ...
 $ RiverManagement                : num [1:50000] 6 5 4 2 5 6 4 5 5 5 ...
 $ Deforestation                  : num [1:50000] 6 7 1 7 2 4 5 5 4 6 ...
 $ Urbanization                   : num [1:50000] 4 7 7 3 5 6 5 6 5 2 ...
 $ ClimateChange                  : num [1:50000] 4 9 5 4 8 4 5 6 11 3 ...
 $ DamsQuality                    : num [1:50000] 6 1 4 1 5 3 4 6 3 7 ...
 $ Siltation                      : num [1:50000] 2 5 7 4 2 1 8 7 2 7 ...
 $ AgriculturalPractices          : num [1:50000] 3 5 4 6 7 3 8 6 9 10 ...
 $ Encroachments                  : num [1:50000] 2 4 9 4 5 5 4 5 7 4 ...
 $ IneffectiveDisasterPreparedness: num [1:50000] 5 6 2 9 7 1 6 5 8 5 ...
 $ DrainageSystems                : num [1:50000] 10 9 7 4 7 10 8 4 2 7 ...
 $ CoastalVulnerability           : num [1:50000] 7 2 4 2 6 5 4 6 8 6 ...
 $ Landslides                     : num [1:50000] 4 6 4 6 5 9 5 9 7 5 ...
 $ Watersheds                     : num [1:50000] 2 2 8 6 3 5 4 7 5 6 ...
 $ DeterioratingInfrastructure    : num [1:50000] 3 1 6 8 3 5 7 10 4 7 ...
 $ PopulationScore                : num [1:50000] 4 1 1 8 4 7 7 6 9 5 ...
 $ WetlandLoss                    : num [1:50000] 3 9 8 6 4 3 5 5 6 7 ...
 $ InadequatePlanning             : num [1:50000] 2 1 3 6 3 3 4 4 5 4 ...
```

```
 $ PoliticalFactors               : num [1:50000] 6 3 6 10 4 2 8 5 7 8 ...
 $ FloodProbability               : num [1:50000] 0.45 0.475 0.515 0.52 0.475 0.47 0.57 0.585
 - attr(*, "spec")=
  .. cols(
  ..   MonsoonIntensity = col_double(),
  ..   TopographyDrainage = col_double(),
  ..   RiverManagement = col_double(),
  ..   Deforestation = col_double(),
  ..   Urbanization = col_double(),
  ..   ClimateChange = col_double(),
  ..   DamsQuality = col_double(),
  ..   Siltation = col_double(),
  ..   AgriculturalPractices = col_double(),
  ..   Encroachments = col_double(),
  ..   IneffectiveDisasterPreparedness = col_double(),
  ..   DrainageSystems = col_double(),
  ..   CoastalVulnerability = col_double(),
  ..   Landslides = col_double(),
  ..   Watersheds = col_double(),
  ..   DeterioratingInfrastructure = col_double(),
  ..   PopulationScore = col_double(),
  ..   WetlandLoss = col_double(),
  ..   InadequatePlanning = col_double(),
  ..   PoliticalFactors = col_double(),
  ..   FloodProbability = col_double()
  .. )
 - attr(*, "problems")=<externalptr>
```

Mostrar as primeiras linhas

```
cat("\nPrimeiras 6 linhas do dataset:\n")
```

```
Primeiras 6 linhas do dataset:
```

```
print(head(df, 6))
```

```
# A tibble: 6 x 21
  MonsoonIntensity TopographyDrainage RiverManagement Deforestation Urbanization
             <dbl>              <dbl>           <dbl>         <dbl>        <dbl>
1                3                  8               6             6            4
```

```
2                8                  4               5              7              7
3                3                 10               4              1              7
4                4                  4               2              7              3
5                3                  7               5              2              5
6                6                  6               6              4              6
# i 16 more variables: ClimateChange <dbl>, DamsQuality <dbl>, Siltation <dbl>,
#   AgriculturalPractices <dbl>, Encroachments <dbl>,
#   IneffectiveDisasterPreparedness <dbl>, DrainageSystems <dbl>,
#   CoastalVulnerability <dbl>, Landslides <dbl>, Watersheds <dbl>,
#   DeterioratingInfrastructure <dbl>, PopulationScore <dbl>,
#   WetlandLoss <dbl>, InadequatePlanning <dbl>, PoliticalFactors <dbl>,
#   FloodProbability <dbl>
```

Análise Exploratória

Estatísticas de resumo (média, mediana, quartis, etc.)

```
cat("\nResumo Estatístico das variáveis numéricas:\n\n")
```

```
Resumo Estatístico das variáveis numéricas:
```

```
df %>%
  select_if(is.numeric) %>%
  summary() %>%
  print()
```

```
 MonsoonIntensity TopographyDrainage RiverManagement  Deforestation
 Min.   : 0.000   Min.   : 0.000     Min.   : 0.000   Min.   : 0.000
 1st Qu.: 3.000   1st Qu.: 3.000     1st Qu.: 3.000   1st Qu.: 3.000
 Median : 5.000   Median : 5.000     Median : 5.000   Median : 5.000
 Mean   : 4.991   Mean   : 4.984     Mean   : 5.016   Mean   : 5.008
 3rd Qu.: 6.000   3rd Qu.: 6.000     3rd Qu.: 6.000   3rd Qu.: 6.000
 Max.   :16.000   Max.   :18.000     Max.   :16.000   Max.   :17.000
  Urbanization    ClimateChange     DamsQuality       Siltation
 Min.   : 0.000   Min.   : 0.000   Min.   : 0.000   Min.   : 0.000
 1st Qu.: 3.000   1st Qu.: 3.000   1st Qu.: 3.000   1st Qu.: 3.000
 Median : 5.000   Median : 5.000   Median : 5.000   Median : 5.000
 Mean   : 4.989   Mean   : 4.988   Mean   : 5.015   Mean   : 4.989
 3rd Qu.: 6.000   3rd Qu.: 6.000   3rd Qu.: 6.000   3rd Qu.: 6.000
 Max.   :17.000   Max.   :17.000   Max.   :16.000   Max.   :16.000
```

```
 AgriculturalPractices Encroachments    IneffectiveDisasterPreparedness
 Min.   : 0.000        Min.   : 0.000   Min.   : 0.000
 1st Qu.: 3.000        1st Qu.: 3.000   1st Qu.: 3.000
 Median : 5.000        Median : 5.000   Median : 5.000
 Mean   : 5.006        Mean   : 5.006   Mean   : 5.005
 3rd Qu.: 6.000        3rd Qu.: 6.000   3rd Qu.: 6.000
 Max.   :16.000        Max.   :18.000   Max.   :16.000
 DrainageSystems  CoastalVulnerability   Landslides       Watersheds
 Min.   : 0.000   Min.   : 0           Min.   : 0.000   Min.   : 0.00
 1st Qu.: 3.000   1st Qu.: 3           1st Qu.: 3.000   1st Qu.: 3.00
 Median : 5.000   Median : 5           Median : 5.000   Median : 5.00
 Mean   : 5.006   Mean   : 5           Mean   : 4.984   Mean   : 4.98
 3rd Qu.: 6.000   3rd Qu.: 6           3rd Qu.: 6.000   3rd Qu.: 6.00
 Max.   :17.000   Max.   :17           Max.   :16.000   Max.   :16.00
 DeterioratingInfrastructure PopulationScore   WetlandLoss
 Min.   : 0.000              Min.   : 0.000   Min.   : 0.000
 1st Qu.: 3.000              1st Qu.: 3.000   1st Qu.: 3.000
 Median : 5.000              Median : 5.000   Median : 5.000
 Mean   : 4.988              Mean   : 4.985   Mean   : 5.005
 3rd Qu.: 6.000              3rd Qu.: 6.000   3rd Qu.: 6.000
 Max.   :17.000              Max.   :19.000   Max.   :22.000
 InadequatePlanning PoliticalFactors FloodProbability
 Min.   : 0.000     Min.   : 0.000   Min.   :0.2850
 1st Qu.: 3.000     1st Qu.: 3.000   1st Qu.:0.4650
 Median : 5.000     Median : 5.000   Median :0.5000
 Mean   : 4.994     Mean   : 4.991   Mean   :0.4997
 3rd Qu.: 6.000     3rd Qu.: 6.000   3rd Qu.:0.5350
 Max.   :16.000     Max.   :16.000   Max.   :0.7250
```

Tabela de contagem de valores ausentes (NA) por coluna

```
cat("\nContagem de valores ausentes por coluna:\n\n")
```

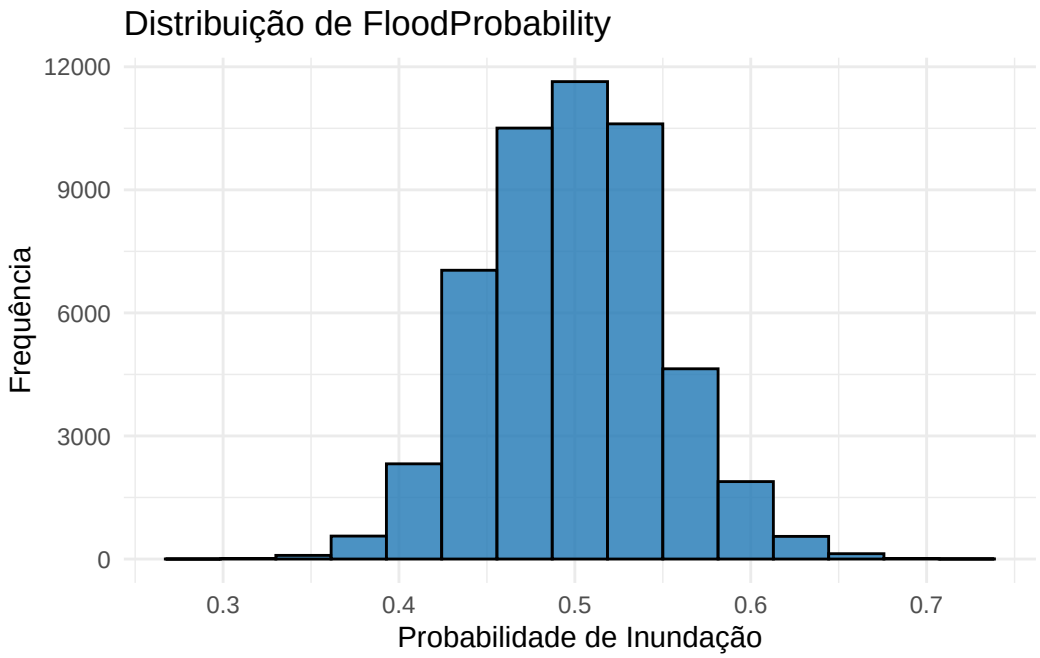
```
Contagem de valores ausentes por coluna:
```

```
na_summary <- df %>%
  summarize_all(~ sum(is.na(.))) %>%
  gather(key = "variavel", value = "n_missing")
kable(na_summary, col.names = c("Variável", "N de missing"))
```

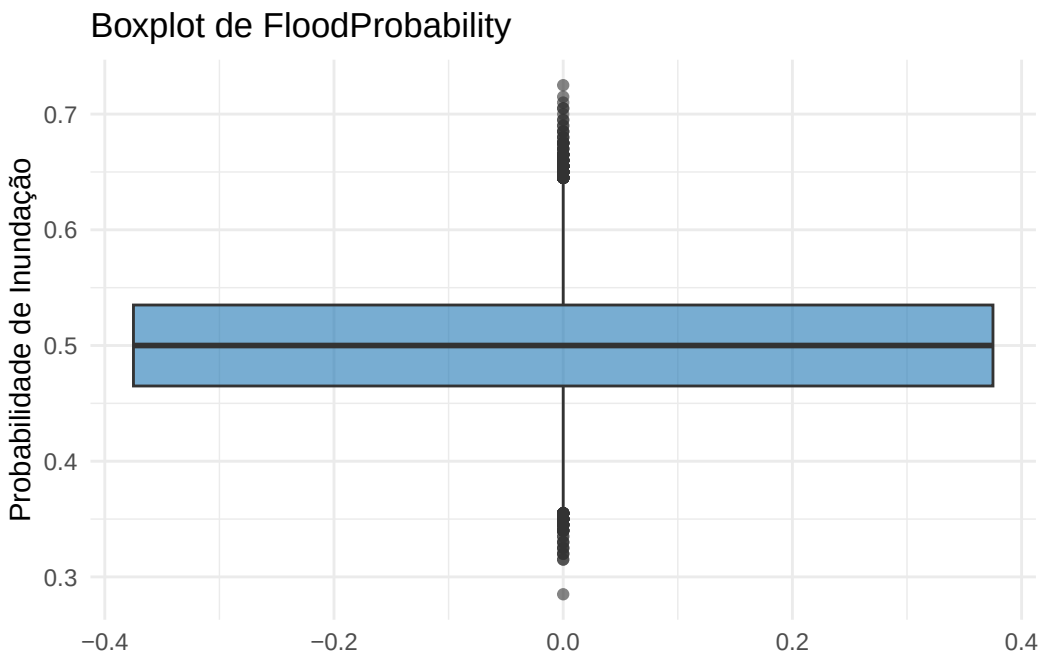
Variável	N de missing
MonsoonIntensity	0
TopographyDrainage	0
RiverManagement	0
Deforestation	0
Urbanization	0
ClimateChange	0
DamsQuality	0
Siltation	0
AgriculturalPractices	0
Encroachments	0
IneffectiveDisasterPreparedness	0
DrainageSystems	0
CoastalVulnerability	0
Landslides	0
Watersheds	0
DeterioratingInfrastructure	0
PopulationScore	0
WetlandLoss	0
InadequatePlanning	0
PoliticalFactors	0
FloodProbability	0

Distribuição da Variável-Alvo (FloodProbability)

```
ggplot(df, aes(x = FloodProbability)) +
  geom_histogram(bins = 15, fill = "#1f77b4", color = "black", alpha = 0.8) +
  labs(
    title = "Distribuição de FloodProbability",
    x = "Probabilidade de Inundação",
    y = "Frequência"
  ) +
  theme_minimal()
```




```
ggplot(df, aes(y = FloodProbability)) +
  geom_boxplot(fill = "#1f77b4", alpha = 0.6) +
  labs(
    title = "Boxplot de FloodProbability",
    y = "Probabilidade de Inundação"
  ) +
  theme_minimal()
```



Distribuição das demais variáveis preditoras

```
library(gridExtra)
```

```
Attaching package: 'gridExtra'
```

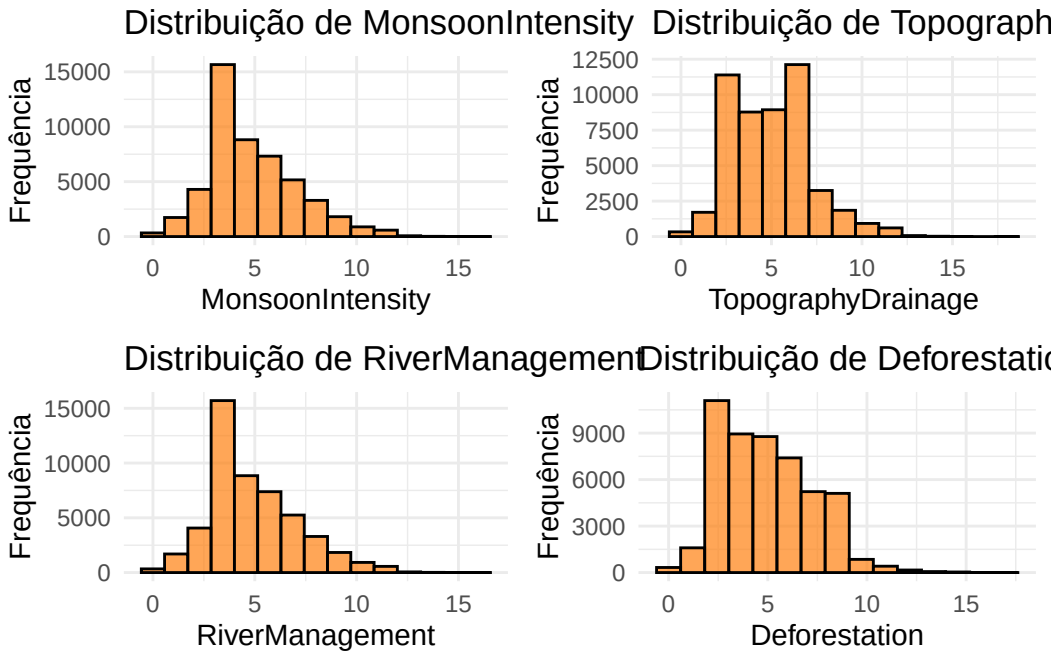
```
The following object is masked from 'package:dplyr':

    combine
```

```
variaveis_preditoras <- df %>% select(-FloodProbability)


vars_to_plot <- names(variaveis_preditoras)[1:4]

plot_list <- list()
for (v in vars_to_plot) {
  p <- ggplot(df, aes_string(x = v)) +
       geom_histogram(bins = 15, fill = "#ff7f0e", color = "black", alpha = 0.7) +
       labs(
         title = paste("Distribuição de", v),
```

```
        x = v,
        y = "Frequência"
      ) +
      theme_minimal()
  plot_list[[v]] <- p
}
```

```
Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
i Please use tidy evaluation idioms with `aes()`.
i See also `vignette("ggplot2-in-packages")` for more information.
```

```
do.call(grid.arrange, c(plot_list, ncol = 2))
```



Matriz de Correlação

```
num_df <- df %>% select_if(is.numeric)
corr_mat <- cor(num_df, use = "pairwise.complete.obs")
```

Exibir matriz de correlação numérica

```r
cat("\nMatriz de Correlação (primeiras 6 linhas):\n")
```

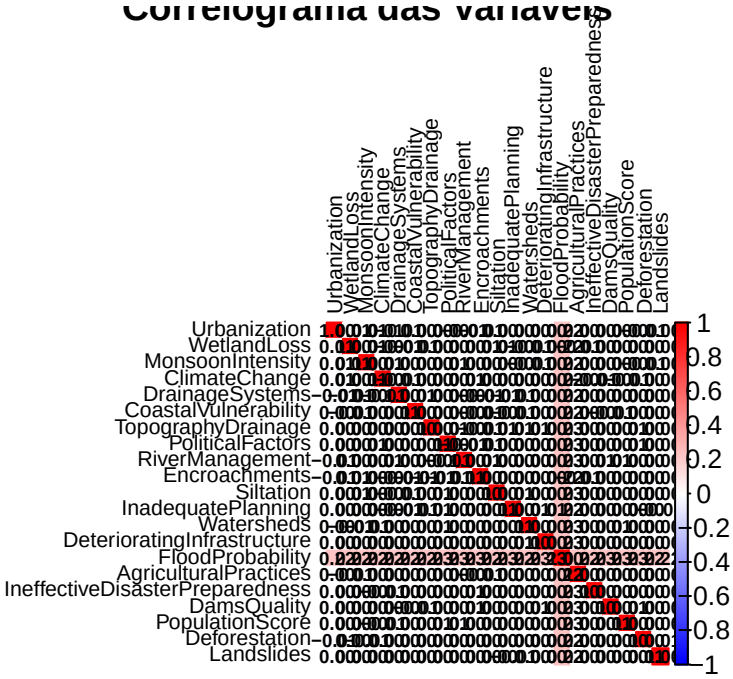
```
Matriz de Correlação (primeiras 6 linhas):
```

```r
print(round(corr_mat[1:6, 1:6], 2))
```

```
                   MonsoonIntensity TopographyDrainage RiverManagement
MonsoonIntensity               1.00                  0            0.00
TopographyDrainage             0.00                  1            0.00
RiverManagement                0.00                  0            1.00
Deforestation                 -0.01                  0            0.00
Urbanization                   0.01                  0           -0.01
ClimateChange                  0.01                  0            0.01
                   Deforestation Urbanization ClimateChange
MonsoonIntensity           -0.01         0.01          0.01
TopographyDrainage          0.00         0.00          0.00
RiverManagement             0.00        -0.01          0.01
Deforestation               1.00        -0.01          0.00
Urbanization               -0.01         1.00          0.01
ClimateChange               0.00         0.01          1.00
```

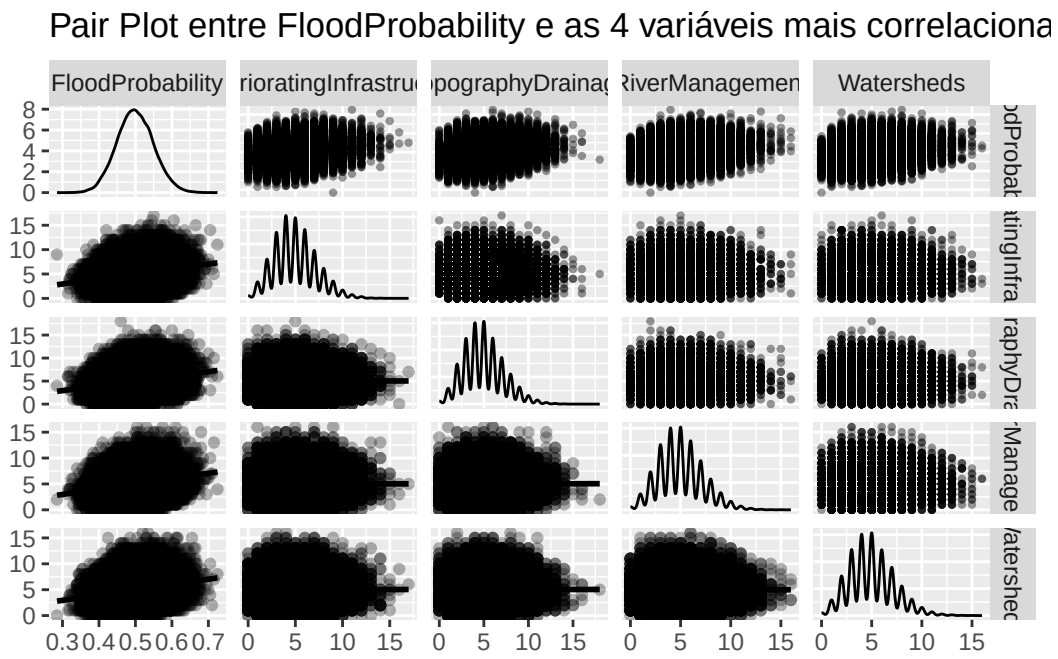
Correlograma completo

```r
corrplot(
  corr_mat,
  method = "color",
  order = "hclust",
  tl.cex = 0.7,
  addCoef.col = "black",
  number.cex = 0.6,
  tl.col = "black",
  col = colorRampPalette(c("blue", "white", "red"))(200),
  title = "Correlograma das Variáveis"
)
```



Pair Plot (GGally) para subconjunto de variáveis

```
corr_flood <- corr_mat[, "FloodProbability"] %>%
  abs() %>%
  sort(decreasing = TRUE)
top_pred <- names(corr_flood)[2:5]
pair_df <- df %>% select(all_of(c("FloodProbability", top_pred)))

ggpairs(
  pair_df,
  upper = list(continuous = wrap("points", alpha = 0.4, size = 0.8)),
  lower = list(continuous = wrap("smooth", se = FALSE, alpha = 0.3)),
  diag = list(continuous = wrap("densityDiag")),
  title = "Pair Plot entre FloodProbability e as 4 variáveis mais correlacionadas"
)
```



Análise Explanatória

separar variaveis de teste:

```
set.seed(123)
train_indices <- sample(seq_len(nrow(df)), size = 0.7 * nrow(df))
df_train <- df[train_indices, ]
df_test  <- df[-train_indices, ]
```

Modelo de Regressão Linear Múltipla

Selecionamos as 5 variáveis de maior correlação (absoluta) com FloodProbability

```
top5_vars <- names(corr_flood)[2:6]

cat("\nVariáveis selecionadas para o modelo:", paste(top5_vars, collapse = ", "), "\n")
```

```
Variáveis selecionadas para o modelo: DeterioratingInfrastructure, TopographyDrainage, RiverM
```

```r
formula_modelo <- as.formula(
  paste("FloodProbability ~", paste(top5_vars, collapse = " + "))
)
```

```r
modelo_lm <- lm(formula_modelo, data = df_train)
cat("\nResumo do modelo de Regressão Linear:\n")
```

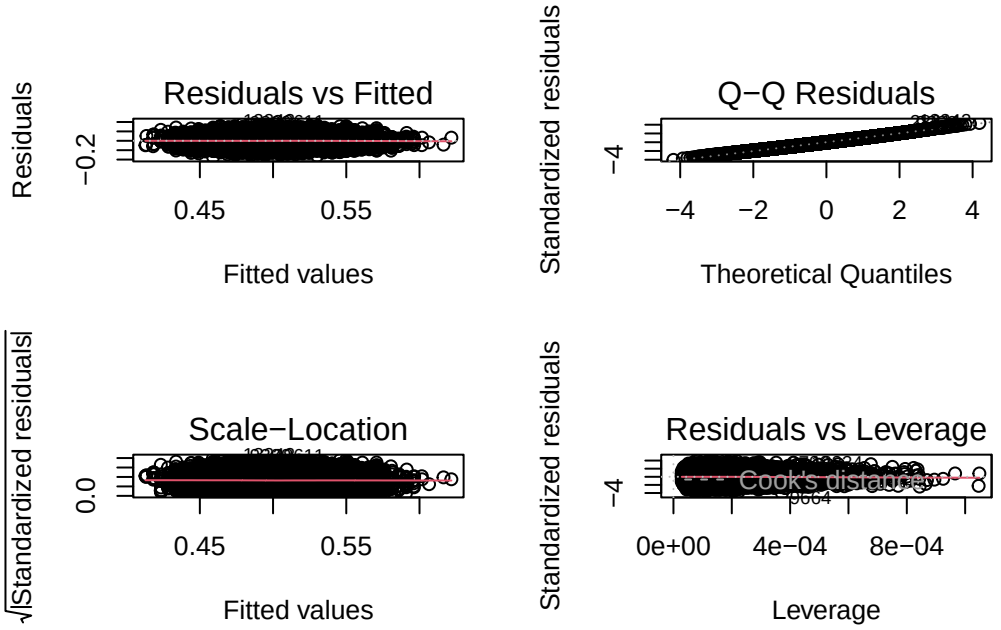
```
Resumo do modelo de Regressão Linear:
```

```r
print(summary(modelo_lm))
```

```
Call:
lm(formula = formula_modelo, data = df_train)

Residuals:
      Min        1Q    Median        3Q       Max
-0.174075 -0.029604  0.000009  0.029810  0.189549

Coefficients:
                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                 0.3726657  0.0011726  317.81   <2e-16 ***
DeterioratingInfrastructure 0.0050384  0.0001036   48.62   <2e-16 ***
TopographyDrainage          0.0051573  0.0001027   50.22   <2e-16 ***
RiverManagement             0.0050900  0.0001034   49.23   <2e-16 ***
Watersheds                  0.0049815  0.0001033   48.23   <2e-16 ***
DamsQuality                 0.0051272  0.0001029   49.83   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.04319 on 34994 degrees of freedom
Multiple R-squared:  0.258, Adjusted R-squared:  0.2579
F-statistic:  2434 on 5 and 34994 DF,  p-value: < 2.2e-16
```

Avaliar suposições: resíduos vs valores ajustados

```r
par(mfrow = c(2, 2))
plot(modelo_lm)
```




```
par(mfrow = c(1, 1))
```

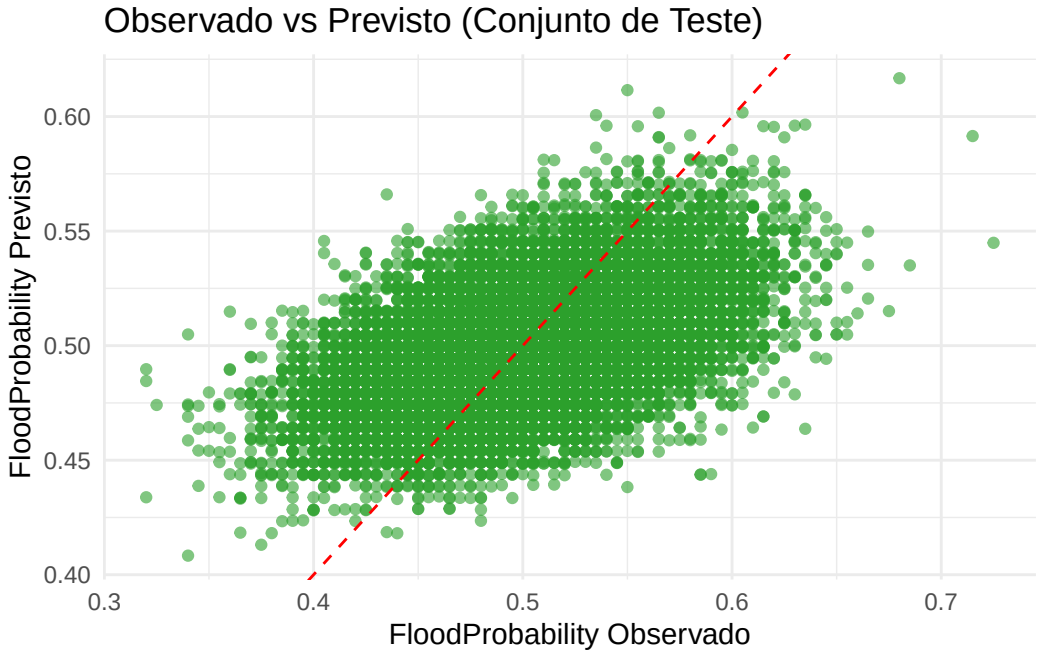
Previsão e Métricas no Conjunto de Teste

```
pred_test <- predict(modelo_lm, newdata = df_test)

library(Metrics)  # para RMSE e MAE
rmse_val  <- rmse(df_test$FloodProbability, pred_test)
mae_val   <- mae(df_test$FloodProbability, pred_test)
mape_val  <- mape(df_test$FloodProbability, pred_test) * 100

cat(sprintf("\nMétricas no conjunto de teste:\n  RMSE = %.4f\n  MAE  = %.4f\n  MAPE = %.2f%%\
            rmse_val, mae_val, mape_val))
```

```
Métricas no conjunto de teste:
  RMSE = 0.0428
  MAE  = 0.0341
  MAPE = 6.92%
```

Gráfico de Previsão vs Observado

```r
df_pred_obs <- tibble(
  Observado  = df_test$FloodProbability,
  Previsto   = pred_test
)

ggplot(df_pred_obs, aes(x = Observado, y = Previsto)) +
  geom_point(alpha = 0.6, color = "#2ca02c") +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed", color = "red") +
  labs(
    title = "Observado vs Previsto (Conjunto de Teste)",
    x = "FloodProbability Observado",
    y = "FloodProbability Previsto"
  ) +
  theme_minimal()
```



Interpretação dos Resultados

```r
cat("\nCoeficientes do modelo:\n")
```

```
Coeficientes do modelo:
```

```r
coef_df <- broom::tidy(modelo_lm)
kable(coef_df, digits = 4, col.names = c("Termo", "Estimativa", "Std. Erro", "t valor", "Pr(>
```

Termo	Estimativa	Std. Erro	t valor	Pr(>|t|)
(Intercept)	0.3727	0.0012	317.8149	0
DeterioratingInfrastructure	0.0050	0.0001	48.6168	0
TopographyDrainage	0.0052	0.0001	50.2181	0
RiverManagement	0.0051	0.0001	49.2267	0
Watersheds	0.0050	0.0001	48.2252	0
DamsQuality	0.0051	0.0001	49.8323	0

```r
cat("\nInterpretação resumida:\n")
```

```
Interpretação resumida:
```

```r
cat("
- O modelo de regressão explica aproximadamente", round(summary(modelo_lm)$r.squared, 3),
    "dos desvios em FloodProbability (R² ajustado =",
    round(summary(modelo_lm)$adj.r.squared, 3), ").\n
- Variáveis com p-valor < 0.05 indicam influência estatisticamente significativa sobre Flood
- Os coeficientes positivos (p.ex., se 'DeterioratingInfrastructure' tiver coeff > 0) sugere
  para cada unidade adicional nessa variável, a probabilidade de inundação tende a aumentar,
- Da mesma forma, coeficientes negativos indicam relação inversa.\n
- As métricas de erro no conjunto de teste (RMSE =", round(rmse_val, 4),
    ", MAE =", round(mae_val, 4), ", MAPE =", round(mape_val, 2), "%) fornecem feedback sobr
- Gráfico Observado vs Previsto: se a maioria dos pontos estiver próxima à linha pontilhada
")
```

```
- O modelo de regressão explica aproximadamente 0.258 dos desvios em FloodProbability (R² aj

- Variáveis com p-valor < 0.05 indicam influência estatisticamente significativa sobre Flood
- Os coeficientes positivos (p.ex., se 'DeterioratingInfrastructure' tiver coeff > 0) sugere
  para cada unidade adicional nessa variável, a probabilidade de inundação tende a aumentar,
- Da mesma forma, coeficientes negativos indicam relação inversa.

- As métricas de erro no conjunto de teste (RMSE = 0.0428 , MAE = 0.0341 , MAPE = 6.92 %) fo

- Gráfico Observado vs Previsto: se a maioria dos pontos estiver próxima à linha pontilhada
```