

Garimpo do Loreço

Darren Norris

20 dezembro, 2022

Sumário

1	Apresentação	3
1.1	Organização	4
1.1.1	Organização do código no tutorial	4
2	Pacotes necessários:	5
3	Área de estudo	5
4	Dados	5
4.1	Ponto de referência (EPSG: 4326)	5
4.2	Ponto de referência (EPSG: 31976)	6
4.2.1	Verificar com mapa de base (OpenStreetMap).	6
4.3	Dados: MapBiomas cobertura da terra	7
5	Cálculo de métricas	8
5.1	Métricas para a paisagem	9
5.2	Métricas para as classes	10
5.3	Métricas para as manchas	11
6	Quais métricas devo escolher?	12
6.1	Métricas por classe de mineração	12
7	Exportar as métricas	14
8	Preparando os resultados	14
9	Uma tabela versátil	17
9.1	1. Reorganização	17
9.2	2. Montar a tabela	18
9.3	3. Exportar a tabela	18

10 Uma figura elegante	18
10.1 Gráfico de barra	19
10.2 Gráfico de boxplot	20
11 Comparação entre anos	21
12 Conclusões e próximos passos	25

1 Apresentação

Mudanças na paisagem ao redor do Garimpo do Lourenço. Changes in the landscape surrounding the Lourenço gold mine.

Código de [R](#) e dados para calcular métricas de paisagem associadas com a exploração de recursos minerais.

O objetivo é calcular métricas de paisagem e descrever a composição e a configuração da paisagem no entorno do Garimpo do Lourenço.

As métricas de paisagem são a forma que os ecólogos de paisagem usam para descrever os padrões espaciais de paisagens para depois avaliar a influência destes padrões espaciais nos padrões e processos ecológicos.

Nesta exemplo (<https://rpubs.com/darren75/lourenco>) aprenderemos sobre como analisar a cobertura da terra com métricas de paisagem em R.

Este exemplo tem como base teórica o modelo “mancha-corredor-matriz” - uma representação da paisagem em manchas de habitat (fragmentos).

1.1 Organização

Os dados aqui apresentados (gráficos, mapas) representam conteúdo do domínio público, disponibilizados pelos institutos, órgãos e entidades federais, estaduais e privados ([IBGE](#), [MapBiomias](#), [Agência Nacional de Mineração](#)). O conteúdo está apresentado para divulgação ampla, respeitando as obrigações de transparência, assim para agilizar e facilitar ensino e o desenvolvimento técnico científico. O conteúdo não representar versões ou produtos finais e não devem ser apresentados/relatados/compartilhados/interpretados como conclusivos.

Os gráficos e mapas ficam na pasta [figures](#) (formato .png e .tif), dados geoespaciais “vector” na pasta [vector](#) (formato shapefile e GPKG) e “raster” na pasta [raster](#).

1.1.1 Organização do código no tutorial

O tutorial está organizado em etapas de processamento, com blocos de código em caixas cinzas:

```
codigo de R para executar
```

Para seguir os passos, os blocos de código precisam ser executados em sequência. Se você pular uma etapa, ou rodar fora de sequência o próximo bloco de código provavelmente não funcionará.

As linhas de código de R dentro de cada caixa também precisam ser executados em sequência. O símbolo `#` é usado para incluir comentários sobre os passos no código. Ou seja, linhas começando com `#` são ignorados por R, e não é código de executar.

```
# Passo 1
codigo de R passo 1 # texto e numeros tem cores diferentes
# Passo 2
codigo de R passo 2
# Passo 3
codigo de R passo 3
```

Além disso, os símbolos `#>` e/ou `[1]` no início de uma linha indica o resultado que você verá no console de R depois de rodar o código, como no próximo exemplo. Digite o código abaixo e veja o resultados (passos 1 a 4).

```
# Passo 1
1+1
```

```
[1] 2
```

```
# Passo 2
x <- 1+1
# Passo 3
x
```

```
[1] 2
```

```
# Passo 4
x + 1
```

```
[1] 3
```

2 Pacotes necessarios:

```
library(tidyverse)
library(readxl)
library(terra)
library(sf)
library(landscapemetrics)
library(mapview)
library(knitr)
library(kableExtra)
library(gridExtra)
```

3 Área de estudo

Para alcançar o objetivo de caracterizar a paisagem no entorno do Garimpo do Lourenço, precisamos estabelecer a extensão da área de estudo. Isso seria estabelecida com base nos objetivos e estudos anteriores. Sabemos que atividades associadas com a mineração pode aumentar a perda da floresta até 70 km além dos limites do processo de mineração: Sonter et. al. 2017. Mining drives extensive deforestation in the Brazilian Amazon <https://www.nature.com/articles/s41467-017-00557-w>

Para visualizar um exemplo com a Extração de bauxita na Flona Saracá-Taquera: <https://earthengine.google.com/timelapse/#v=-1.70085,-56.45017,8.939,latLng&t=2.70>

E aqui com o Garimpo do Lourenço: <https://earthengine.google.com/timelapse/#v=2.2994,-51.68423,11.382,latLng&t=0.03>

4 Dados

4.1 Ponto de referência (EPSG: 4326)

Aqui vamos incluir um raio de 20 km além do ponto de acesso para o Garimpo do Lourenço em 1985. Isso representa uma área quadrada de 40 x 40 km (1600 km²).

```
# Tabela de dados com coordenados de acesso em 1985.
acesso <- data.frame(nome = "garimpo do Lourenço",
  coord_x = -51.630871,
  coord_y = 2.318514)

# Converter para objeto espacial, com sistema de coordenados geográfica.
sf_acesso <- st_as_sf(acesso,
  coords = c("coord_x", "coord_y"),
  crs = 4326)
```

Visualizar para verificar.

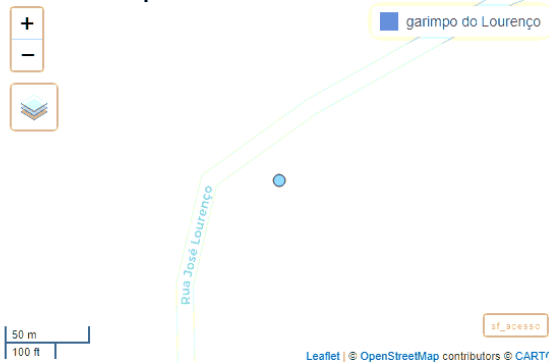
```
#
plot(sf_acesso) # teste basica
mapview(sf_acesso) #verificar com mapa de base (OpenStreetMap)
```

(A) basica

nome

o

(B) com mapa de base



4.2 Ponto de referência (EPSG: 31976)

As análises da paisagem com o modelo “mancha-corredor-matriz” depende de uma classificação categórica. Portanto, deve optar para uma sistema de coordenados projetados, com pixels de área igual e com unidade em metros. Temos um raio de 20 km, que é um area geografica onde o retângulo envolvente é menor que um fuso UTM. Assim sendo, vamos adotar a sistema de coordenados projetados de datum SIRGAS 2000, especificamente EPSG:31976 (SIRGAS 2000/UTM zone 22N).

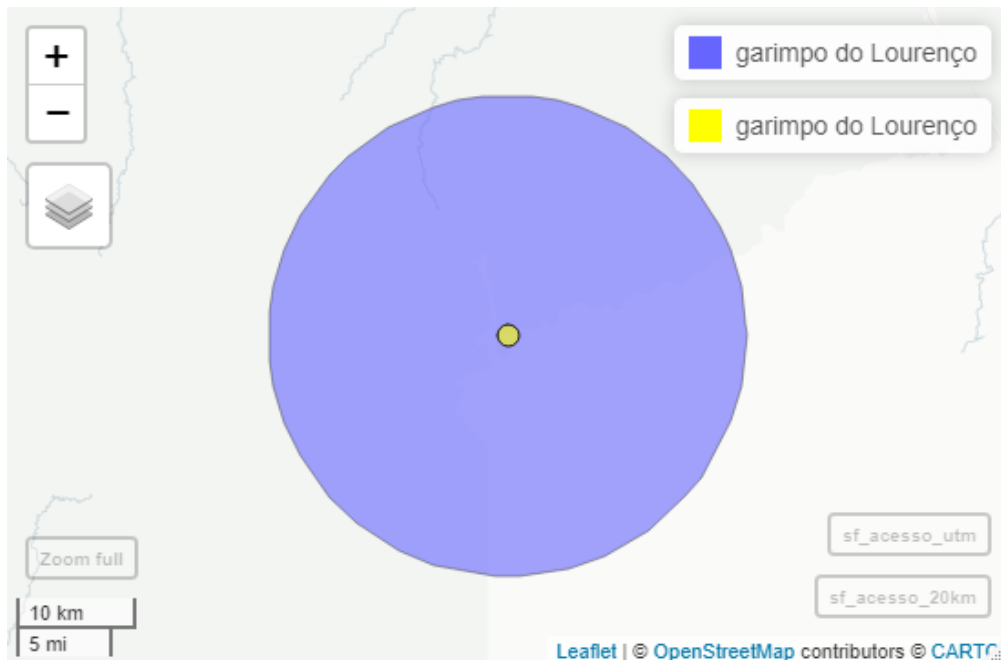
Precisamos então reprojetar o objeto original (em coordenados geográficas) para a sistema de coordenados projetados. Em seguida, vamos produzir um polígono com raio de 20 km no entorno do ponto.

```
# Reprojetar o ponto.
sf_acesso_utm <- st_transform(sf_acesso, crs = 31976)
# Polígono com raio de 500 metros no entorno do ponto.
sf_acesso_500m <- st_buffer(sf_acesso_utm, dist=500) %>%
  mutate(raio_km = 0.5)
# Polígono com raio de 1 km no entorno do ponto.
sf_acesso_1km <- st_buffer(sf_acesso_utm, dist=1000) %>%
  mutate(raio_km = 1)
# Polígono com raio de 2 km no entorno do ponto.
sf_acesso_2km <- st_buffer(sf_acesso_utm, dist=2000) %>%
  mutate(raio_km = 2)
# Polígono com raio de 4 km no entorno do ponto.
sf_acesso_4km <- st_buffer(sf_acesso_utm, dist=4000) %>%
  mutate(raio_km = 4)
# Polígono com raio de 20 km no entorno do ponto.
sf_acesso_20km <- st_buffer(sf_acesso_utm, dist=20000)

acesso_buffers <- bind_rows(sf_acesso_500m, sf_acesso_1km,
                           sf_acesso_2km, sf_acesso_4km)
```

4.2.1 Verificar com mapa de base (OpenStreetMap).

```
#
mapview(sf_acesso_20km) +
  mapview(sf_acesso_utm, color = "black", col.regions = "yellow")
```



4.3 Dados: MapBiomias cobertura da terra

Agora vamos olhar cobertura e uso da terra no espaço que preciso (área de estudo). Para isso, vamos utilizar um arquivo de raster do projeto [MapBiomias](#) com cobertura de terra ao redor do Garimpo do Lourenço em 1985. Este arquivo no formato raster, tem apenas valores inteiros, em que cada célula/pixel representa uma área considerada homogênea, como uso do solo ou tipo de vegetação. Arquivo “.tif” disponível aqui: [utm_cover_AP_lorenco_1985.tif](#)

Não vamos construir mapas, portanto as cores nas visualizações não corresponde ao mundo real (por exemplo, verde não é floresta). Para visualizar em QGIS preciso baixar um arquivo com a legenda e cores para Coleção 6 (<https://mapbiomas.org/codigos-de-legenda>) e segue tutoriais: <https://www.youtube.com/watch?v=WtyotodHK8E>.

Este vez, a entrada de dados espaciais seria através da importação de um raster (arquivo de .tif). Lembre-se, para facilitar, os arquivos devem ficar no mesmo diretório do seu código (verifique com `getwd()`). Como nós já sabemos a sistema de coordenadas desejadas, o geoprocessamento da raster foi concluído antes de começar com as análises da paisagem.

```
r1985 <- rast("utm_cover_AP_lorenco_1985.tif")
r1985

#class      : SpatRaster
#dimensions : 1341, 1341, 1 (nrow, ncol, nlyr)
#resolution : 29.87713, 29.87713 (x, y)
#extent      : 409829.5, 449894.7, 236241.1, 276306.3 (xmin, xmax, ymin, ymax)
#coord. ref. : SIRGAS 2000 / UTM zone 22N (EPSG:31976)
#source      : utm_cover_AP_lorenco_1985.tif
#name        : classification_1985
#min value   : 3
#max value   : 33
```

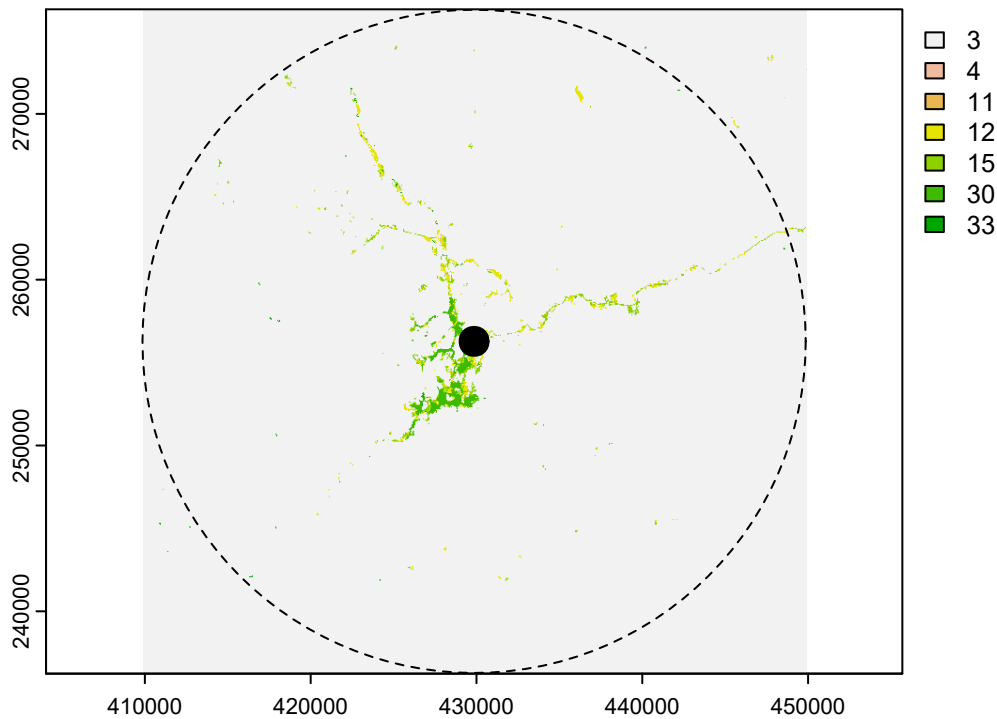
Ou use a função `file.choose()`, que faz a busca para arquivos.

```
r1985 <- rast(file.choose())
r1985
```

```
#>          used (Mb) gc trigger (Mb) max used (Mb)
#> Ncells 3195819 170.7   5690792 304.0  5399570 288.4
#> Vcells 4278423  32.7   10146329 77.5   8385453  64.0
```

Agora que o arquivo foi importado, podemos visualizá-lo.

```
# Visualizar para verificar
# Gradiente de cores padrão não corresponde
# ao mundo real (por exemplo verde não é floresta)
plot(r1985, type="classes")
plot(sf_acesso_20km, add = TRUE, lty = "dashed", color = "black")
plot(sf_acesso_utm, add = TRUE, cex = 2, pch = 19, color = "black")
```



5 Cálculo de métricas

Vamos olhar alguns exemplos de métricas para cada nível da análise:

- landscape (métricas para a paisagem como um todo).
- class (métricas por classe ou tipo de habiat).
- patch (para a mancha ou fragmento).

Primeiro, precisamos verificar se o raster está no formato correto.

```
check_landscape(r1985)
```

```
#>   layer      crs units  class n_classes OK
#> 1      1 projected    m integer         7  v
```

```
# layer crs      units  class n_classes OK
# 1 projected    m integer         7  v
```

Tudo certo (veja a coluna do “OK”)!

5.1 Métricas para a paisagem

Vamos começar avaliando a área total da paisagem (área) de estudo.

```
area.total <- lsm_l_ta(r1985)
area.total #160264 Hectares
```

```
#> # A tibble: 1 x 6
#>   layer level      class    id metric  value
#>   <int> <chr>      <int> <int> <chr>   <dbl>
#> 1      1 landscape    NA    NA ta     160264.
```

Agora vamos ver a distância total de borda (te= “total edge”).

```
#>
#>      used (Mb) gc trigger (Mb) max used (Mb)
#> Ncells 3357889 179.4   5690792 304.0 5399570 288.4
#> Vcells 4579217  35.0   14786712 112.9 14422195 110.1
```

```
te <- lsm_l_te(r1985)
te # 547140 metros
```

```
#> # A tibble: 1 x 6
#>   layer level      class    id metric  value
#>   <int> <chr>    <int> <int> <chr>   <dbl>
#> 1     1  landscape    NA    NA te    547140.
```

Total de borda mede a configuração da paisagem porque uma paisagem altamente fragmentada terá muitas bordas. No entanto, a borda total é uma medida absoluta, dificultando comparações entre paisagens com áreas totais diferentes. Mas pode ser aplicado para comparar a configuração na mesma paisagem em anos diferentes.

Agora vamos ver a densidade de Borda (“Edge Density”). Densidade de Borda mede a configuração da paisagem porque uma paisagem altamente fragmentada terá valores mais altas. “Densidade” é uma medida adequado para comparacoes de paisagens com áreas totais diferentes.

```
ed <- lsm_l_ed(r1985)
ed #3.41 metros por hectare
```

```
#> # A tibble: 1 x 6
#>   layer level      class    id metric value
#>   <int> <chr>    <int> <int> <chr>   <dbl>
#> 1     1  landscape    NA    NA ed     3.41
```

5.2 Métricas para as classes

Area de cada classe em hectares.

```
lsm_c_ca(r1985)
```

```
#> # A tibble: 7 x 6
#>   layer level class    id metric  value
#>   <int> <chr> <int> <int> <chr>   <dbl>
#> 1     1  class     3    NA ca    158582.
#> 2     1  class     4    NA ca      1.70
#> 3     1  class    11    NA ca      1.79
#> 4     1  class    12    NA ca      548.
#> 5     1  class    15    NA ca      563.
#> 6     1  class    30    NA ca      526.
#> 7     1  class    33    NA ca      41.4
```

Como tem varios classes é difícil de interpretar os resultados porque os numeros (3, 4, 11....) não tem uma referencia do mundo real. Para entender os resultados, precisamos acrescentar nomes para os valores. Ou seja incluir uma coluna de legenda com os nomes para cada classe. Para isso precisamos outro arquivo com os nomes.

Baixar a arquivo de legenda: [mapbiomas_6_legend.xlsx](#).

Agora carregar o arquivo com o código a seguir.

```
class_nomes <- read_excel(file.choose())
```

Agora rodar de novo, com os resultados juntos com a legenda de cada classe. Nos resultados acima, os valores na coluna “class” são as mesmas que tem na coluna “aid” no objeto “class_nomes”, onde também tem os nomes. Assim, podemos repetir, mas agora incluindo os nomes para cada valor de class, com base na ligação (join) entre as colunas.

```
# Área de cada classe em hectares, incluindo os nomes para cada classe
lsm_c_ca(r1985) %>%
  left_join(class_nomes, by = c("class" = "aid"))

# Numero de fragmentos (manchas)
lsm_c_np(r1985) %>%
  left_join(class_nomes, by = c("class" = "aid"))

# Maior numero de manchas em classes de cobertura classificadas como
# pasto (pasture) e formação campestre (grassland).

# layer level class      id metric value class_description group_description
#      1 class      3      NA np      28 Forest Formation  Natural forest
#      1 class      4      NA np       2 Savanna Formation  Natural forest
#      1 class     11      NA np       7 Wetlands           Natural non fore
#      1 class     12      NA np     246 Grassland           Natural non fore.
#      1 class     15      NA np     262 Pasture             Farming
#      1 class     30      NA np      35 Mining              Non vegetated
#      1 class     33      NA np     50 River,Lake and Ocean Water
```

5.3 Métricas para as manchas

Vamos calcular o tamanho de cada mancha agora.

```
mancha_area <- lsm_p_area(r1985) # 630 manchas
mancha_area
```

Agora queremos saber o tamanho da maior mancha em cada class, e portanto o tamanho da maior mancha de mineração.

```
mancha_area %>%
  group_by(class) %>%
  summarise(max_ha = max(value))
# 30.8 hectares (class 15 = mineração)
```

6 Quais métricas devo escolher?

A decisão deve ser tomada com base em uma combinação de fatores. Incluindo tais fatores como: base teórica, considerações estatísticas, relevância para o objetivo/hipótese e a escala e heterogeneidade na paisagem de estudo.

Queremos caracterizar áreas de mineração na paisagem, e aqui vamos olhar somente uma paisagem, em um momento do tempo. Então as métricas para a paisagem como todo não tem relevância.

Estamos olhando uma classe (mineração), portanto vamos incluir as métricas para classes. Além disso, as métricas de paisagem em nível de classe são mais eficazes na definição de processos ecológicos (Tischendorf, L. Can landscape indices predict ecological processes consistently?. Landscape Ecology 16, 235–254 (2001). <https://doi.org/10.1023/A:1011112719782>).

```
# métricas de composição para a paisagem por classes
list_lsm(level = "class", type = "area and edge metric")

# métricas de configuração para a paisagem por classes
list_lsm(level = "class", type = "aggregation metric")
```

6.1 Métricas por classe de mineração

Aqui vamos calcular todas as métricas por classe (função `calculate_lsm()`).

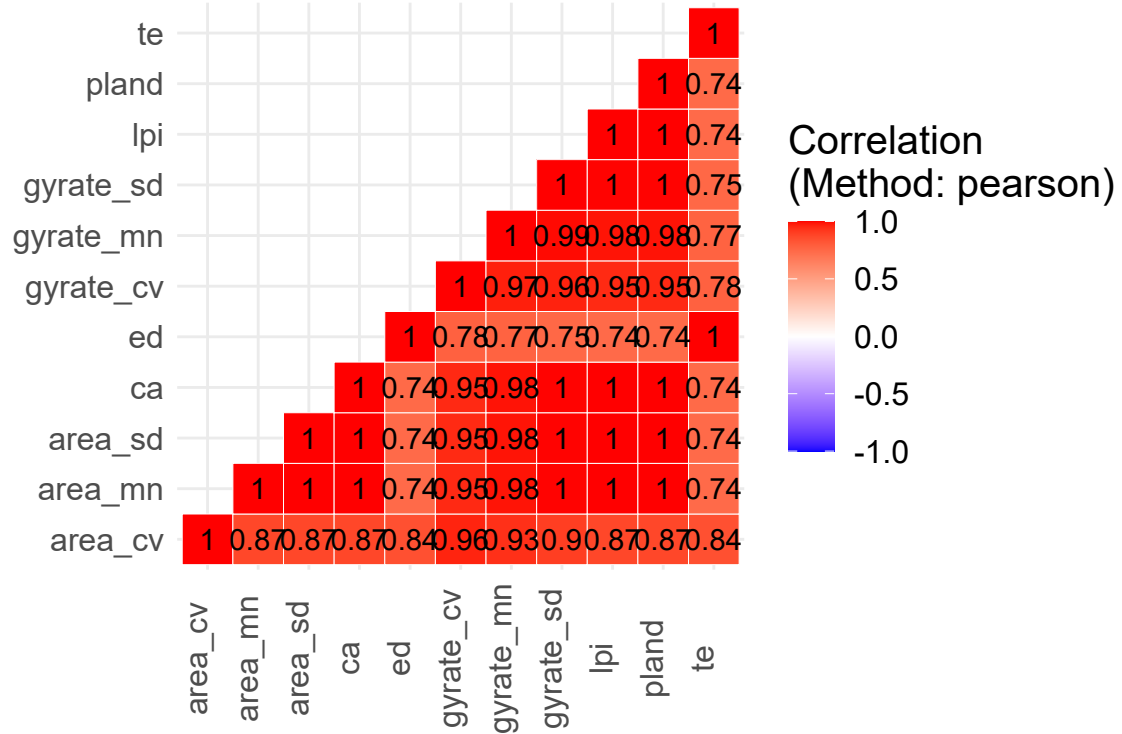
```
# métricas de composição para a paisagem por classes
metrics_comp <- calculate_lsm(r1985, level = "class", type = "area and edge metric")

# métricas de configuração para a paisagem por classes
metrics_config <- calculate_lsm(r1985, level = "class", type = "aggregation metric")
```

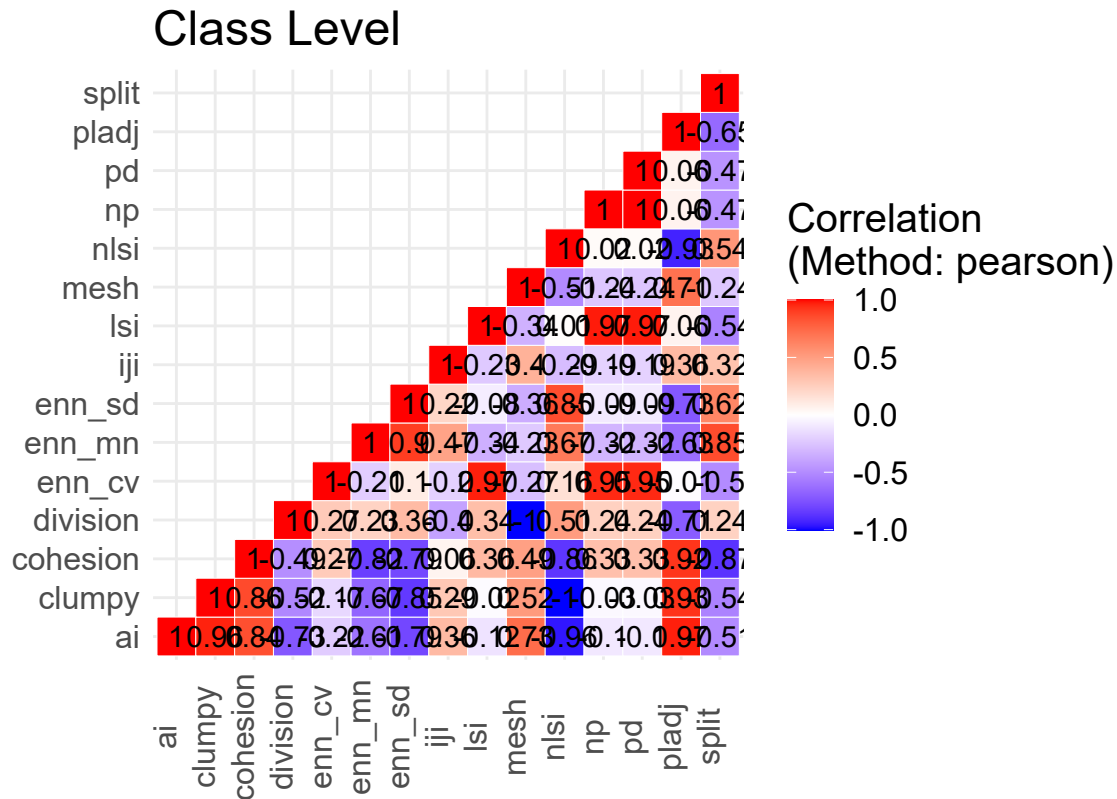
E aqui, calcular correlações entre todas as métricas por classe (função `show_correlation()`).

```
show_correlation(data = metrics_comp, method = "pearson", labels = TRUE)
```

Class Level



```
show_correlation(data = metrics_config, method = "pearson", labels = TRUE)
```



Temos muitos valores e muitas métricas. Este se chama um “tiro no escuro”, algo cujo resultado se desconhece ou é imprevisível. Isso não é recomendado. Para fazer uma escolha melhor (mais robusta), seguindo princípios básicos da ciência, precisamos ler os estudos anteriores (artigos) para obter as métricas mais relevantes para nosso objetivo e a hipótese a ser testada. Com base em os estudos anteriores e os objetivos vamos incluir 8 métricas nos resultados.

7 Exportar as métricas

O próximo passo é comunicar os resultados obtidos. Para isso precisamos resumir e apresentar as métricas selecionadas em tabelas e figuras. Agora já fizemos os cálculos, as tabelas e figuras podem ser feitas no R ([figuras](#)), tanto quanto em aplicativos diferentes (por exemplo tabelas através [“tabelas dinamicas”] no [Microsoft Excel](#) ou [LibreOffice calc](#)). Mas por isso, primeiramente precisamos exportar os resultados (veja mais exemplos aqui: [Introdução ao R import-export](#)).

O arquivo vai sair no mesmo diretório do seu código (verifique com `getwd()`).

```
bind_rows(metrics_comp, metrics_config) -> metrics_1985
write.csv2(metrics_1985, "metrics_lourenco_1985.csv", row.names=FALSE)
```

8 Preparando os resultados

A entrada de dados seria com as métricas da paisagem calculados anteriormente.

Vocês devem baixar o arquivo de Excel [metricas_lourenco_1985.xlsx](#). Lembre-se, para facilitar, os dados deve ficar no mesmo diretório do seu código (verifique com `getwd()`).

No caso de um arquivo de Excel simples, a importação poderia ser feita através menu de “Import Dataset” na janela/panel “Environment” de Rstudio. Ou com linhas de código:

```
metricas_1985 <- read_excel("metricas_lourenco_1985.xlsx")
metricas_1985
```

```
#   layer level class id   metric    value
#   <dbl> <chr> <dbl> <chr> <chr>    <dbl>
#     1 class     3 NA   area_cv  529.
#     1 class     4 NA   area_cv  22.3
#     1 class    11 NA   area_cv  71.2
```

Ou use o função `file.choose()`, que faz a busca para arquivos.

```
metricas_1985 <- read_excel(file.choose())
metricas_1985
```

```
#> # A tibble: 182 x 6
#>   layer level class id   metric    value
#>   <dbl> <chr> <dbl> <chr> <chr>    <dbl>
#> 1     1 class     3 NA   area_cv  529.
#> 2     1 class     4 NA   area_cv  22.3
#> 3     1 class    11 NA   area_cv  71.2
#> 4     1 class    12 NA   area_cv  169.
#> 5     1 class    15 NA   area_cv  175.
#> 6     1 class    30 NA   area_cv  276.
#> 7     1 class    33 NA   area_cv   74.2
#> 8     1 class     3 NA   area_mn 5664.
#> 9     1 class     4 NA   area_mn   0.848
#> 10    1 class    11 NA   area_mn   0.255
#> # ... with 172 more rows
```

Os dados são padronizados (“tidy”), mas ainda não parece adequados para apresentação em tabelas ou figuras. Temos muitos valores e muitas métricas (listadas na coluna “metric”). Com base em os estudos anteriores e os objetivos vamos incluir 8 métricas (4 de composição e 4 de configuração).

Métricas de composição:

- mean patch area (`lsm_c_area_mn`) Área médio das manchas por classe.
- SD patch area (`lsm_c_area_sd`) Desvio padrão das áreas dos manchas por classe.
- class area percentage of landscape (`lsm_c_pland`) Porcentagem de área na paisagem por classe.
- largest patch index (`lsm_c_lpi`) Índice de maior mancha (proporção da paisagem).

Métricas de configuração:

- aggregation index (`lsm_c_ai`) Índice de agregação.

- patch cohesion index (lsm_c_cohesion) Índice de coesão das manchas.
- number of patches (lsm_c_np) Número de manchas.
- patch density (lsm_c_pd) Densidade de manchas.

Escolheremos (atraves um filtro) as métricas que queremos para obter uma tabela de dados. Mantendo os dados originais, assim sendo para acrescentar mais métricas nos resultados, preciso somente acrescentar mais no código.

```
# Arquivo com os nomes das classes
class_in <- "data/mapbiomas_6_legend.xlsx"
class_nomes <- read_excel(class_in)

# Especificar métricas desejados
met_comp <- c("pland", "lpi", "area_mn", "area_sd")
met_conf <- c("ai", "cohesion", "np", "pd")
met_todos <- c(met_comp, met_conf)

# Escolher métricas desejados do conjunto completo
metricas_1985 %>%
  filter(metric %in% met_todos) %>%
  left_join(class_nomes, by = c("class" = "aid")) -> metricas_nomes
```


9 Uma tabela versatil

Mas, ainda não tem uma coluna com os nomes das métricas. Portanto, solução simples é de exportar no formato de .csv e finalizar/editar no Excel / calc.

Outra opção que pode facilitar, particularmente quando pode há mudanças e revisões, é produzir a tabela no R. Aqui vamos repetir no R os passos que vocês conhecem com as ferramentas de Excel (arraste e solte, copiar-colar, filtro, tabela dinâmica).

Podemos organizar os dados que nos temos (objecto “metricas_nomes”) e apresentar em uma tabela em 3 passos: 1. Reorganização, 2. Montar a tabela e 3. Exportar a tabela em uma formato versatil, compatível com documentos (e.g. Word) e planilhas (e.g. Excel).

9.1 1. Reorganização

Escolhendo as colunas desejadas (select), reorganizando para as métricas ficam nas colunas (pivot_wider) e colocando as colunas novas na sequência desejada (select).

```
metricas_nomes %>%  
# Escolher métricas desejados do conjunto completo de métricas.  
select(c(type_class, classe_descricao, hexadecimal_code,  
metric, value)) %>%  
# reorganizando  
pivot_wider(names_from = metric, values_from = value) -> metricas_tab
```

9.2 2. Montar a tabela

Agora vamos produzir uma tabela simples e exportar em um formato versátil (html) para finalização no Word / Excel.

```
# Nomes para as colunas
col_nomes <- c("Tipo", "Descrição", "Área total", "Índice maior mancha",
"Número de manchas", "Área manchas (médio)", "Área manchas (DP)",
"Índice de agregação", "Índice de coesão", "Densidade de manchas")

# Valores para pontos decimais de cada coluna.
meu_digits <- c(0, 0, 0, 1, 0, 1, 1, 1, 1, 2)

metricas_tab %>%
# Colocar as colunas na sequência desejada.
select(type_class, classe_descricao, pland, lpi, np, area_mn, area_sd,
ai, cohesion, pd) %>%
# Especificar nomes para as colunas.
kable(col.names = col_nomes,
digits = meu_digits) %>%
kableExtra::kable_styling() #visualizar para verificar
```

Tipo	Descrição	Área total	Índice maior mancha	Número de manchas	Área manchas (médio)	Área manchas (DP)	Índice de agregação	Índice de coesão	Densidade de manchas
natural	Formação Florestal	99	98.9	28	5683.6	29940.5	99.8	100.0	0.02
natural	Formação Savânica	0	0.0	2	0.8	0.2	65.5	67.6	0.00
natural	Campo Alagado e Área Pantanosa	0	0.0	7	0.3	0.2	48.4	45.7	0.00
natural	Formação Campestre	0	0.0	246	2.2	3.8	68.1	86.1	0.15
antropic	Pastagem	0	0.0	262	2.1	3.8	63.2	86.4	0.16
antropic	Mineração	0	0.1	35	15.0	41.5	86.9	96.7	0.02
natural	Rio, Lago e Oceano	0	0.0	50	0.8	0.6	49.5	70.3	0.03

Parece um bom começo. Vamos exportar. Depois, pode finalizar no documento você está escrevendo (inserir -> objeto, e depois segue os passos) ou em uma planilha.

9.3 3. Exportar a tabela

```
metricas_tab %>%
# Colocar as colunas na sequência desejada.
select(type_class, classe_descricao, pland, lpi, np, area_mn, area_sd,
ai, cohesion, pd) %>%
# Especificar nomes para as colunas.
kable(col.names = col_nomes,
digits = meu_digits) %>%
kableExtra::kable_styling() %>%
kableExtra::save_kable("tabela_metricas_1985.html")
```

E agora pode finalizar a tabela “tabela_metricas_1985.html” no documento você está escrevendo (inserir -> objeto, e depois segue passos) ou em uma [planilha](#).

10 Uma figura elegante

Uma imagem vale mais que mil palavras. Portanto, gráficos/figuras/imagens são uma das mais importantes formas de comunicar a ciência.

Como exemplo ilustrativo, aqui vamos produzir gráficos comparando métricas de composição e configuração da paisagem ao redor do Garimpo do Lourenço.

É uma boa ideia gastar bastante tempo para tornar figuras científicas as mais informativas e atraentes possíveis. Escusado será dizer que a precisão empírica é primordial. E por isso, o que fica excluído/omitido é tão importante quanto o que foi incluído. Para ajudar, você deve se perguntar o seguinte ao criar uma figura: eu apresentaria essa figura em uma apresentação para um grupo de colegas? Eu o apresentaria a um público de não especialistas? Eu gostaria que essa figura aparecesse em um artigo de notícias sobre meu trabalho? É claro que todos esses locais exigem diferentes graus de precisão, complexidade e estética, mas uma boa figura deve servir para educar simultaneamente públicos muito diferentes.

Tabelas versus gráficos — A primeira pergunta que você deve se fazer é se você pode transformar aquela tabela (chata e feia) em algum tipo de gráfico. Você realmente precisa dessa tabela no texto principal? Você não pode simplesmente traduzir as entradas das células em um gráfico de barras/colunas/xy? Se você pode, você deve. Quando uma tabela não pode ser facilmente traduzida em uma figura, na maioria das vezes a provavelmente pertence às Informações Suplementares/Anexos/Apêndices.

10.1 Gráfico de barra

Primeiramente, vamos produzir uma gráfico de barra comparando a proporção que cada classe representa na paisagem.

```
# Inclundo cores conforme legenda da Mapbiomas Coleção 6
# Legenda nomes ordem alfabetica
classe_cores <- c("Campo Alagado e Área Pantanosa" = "#45C2A5",
"Formação Campestre" = "#B8AF4F",
"Formação Florestal" = "#006400",
"Formação Savânica" = "#00ff00",
"Mineração" = "#af2a2a",
"Pastagem" = "#FFD966",
"Rio, Lago e Oceano" = "#0000FF")
```

E agora o grafico.....

```
# Grafico de barra basica
metricas_tab %>%
mutate(class_prop = pland) %>%
ggplot(aes(x = classe_descricao, y = class_prop)) +
geom_col()

# Agora com ajustes
# Agrupando por tipo (natural e antropico)
# Com cores conforme legenda da Mapbiomas Coleção 6
# Corrigindo texto dos eixos.
# Mudar posição da legenda para o texto com nomes longas encaixar.
metricas_tab %>%
mutate(class_prop = pland) %>%
ggplot(aes(x = type_class, y = class_prop,
fill = classe_descricao)) +
scale_fill_manual("classe", values = classe_cores) +
geom_col(position = position_dodge2(width = 1)) +
coord_flip() +
labs(title = "MapBiomas cobertura da terra",
```

```

subtitle = "Entorno do Garimpo do Lorengo 1985",
y = "Proporção da paisagem (%)",
x = "") +
theme(legend.position="bottom") +
guides(fill = guide_legend(nrow = 4))

```

Uma imagem vale mais que mil palavras:

Mas existe uma separação grande na faixa de valores e ainda é difícil de ver todas as classes. Temos uma distribuição com valores muito mais altos comparada com os outros. extremos. Uma solução seria uma transformação (por exemplo “log”), assim os valores ficarem mais próximos.

10.2 Gráfico de boxplot

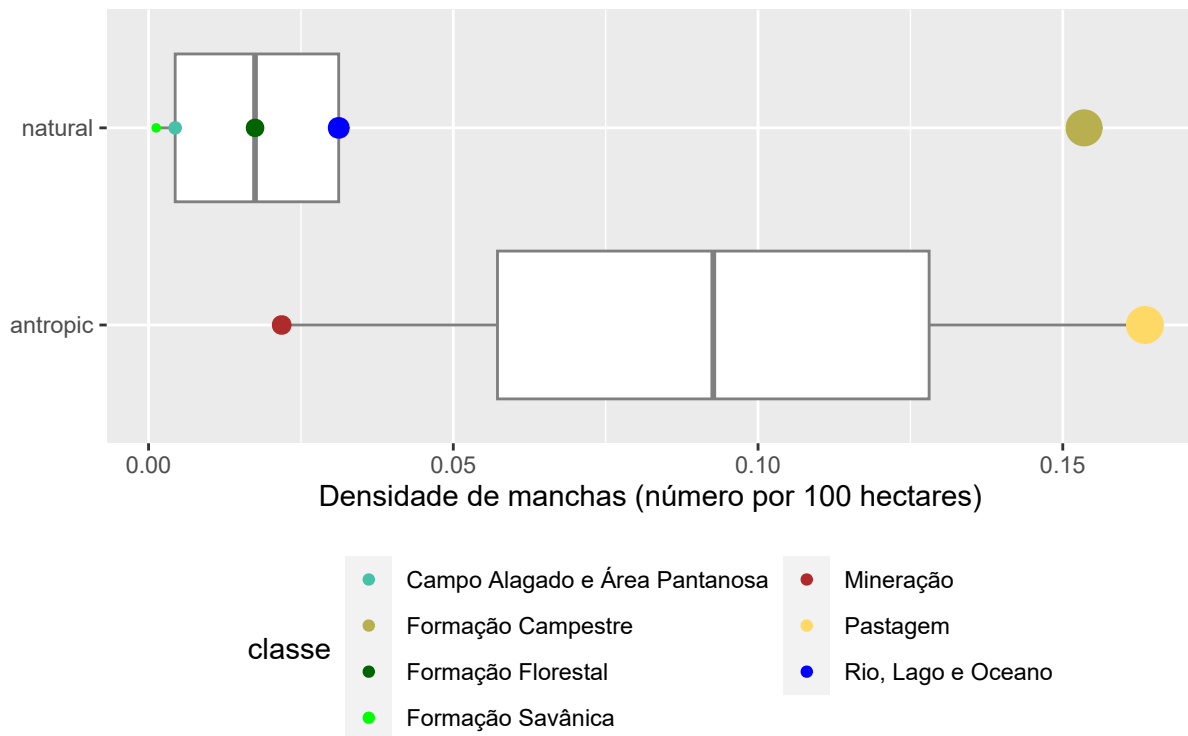
Agora com uma métrica de configuração:

```

# Métrica de configuração: Densidade de manchas (coluna "pd").
# Agrupando por tipo (natural e antrópico)
# Incluindo boxplot indicando tendência central (mediano)
# Com cores conforme legenda da Mapbiomas Coleção 6
# Tamanho dos pontos proporcional o numero de manchas
# Corrigindo texto dos eixos.
# Mudar posição da legenda para o texto com nomes longas encaixar.
metricas_tab %>%
ggplot(aes(x = type_class, y = pd)) +
geom_boxplot(colour = "grey50") +
geom_point(aes(size = np, colour = classe_descricao)) +
scale_color_manual("classe", values = classe_cores) +
scale_size(guide = "none") +
coord_flip() +
labs(title = "MapBiomas cobertura da terra",
subtitle = "Entorno do Garimpo do Lorengo 1985",
y = "Densidade de manchas (número por 100 hectares)",
x = "") +
theme(legend.position="bottom") +
guides(col = guide_legend(nrow = 4))

```

MapBiomas cobertura da terra Entorno do Garimpo do Loreço 1985



11 Comparação entre anos

Calcular as métricas para 3 anos.

```
# métricas desejados
what_metricas <- c("lsm_c_pland", "lsm_c_lpi", "lsm_c_area_mn", "lsm_c_area_sd",
                  "lsm_c_ai", "lsm_c_cohesion", "lsm_c_np", "lsm_c_pd")

# rodar
metricas_anos <- sample_lsm(landscape = mapbiomas_85a20,
                           y = acesso_buffers,
                           plot_id = data.frame(acesso_buffers)[, 'raio_km'],
                           what = what_metricas,
                           edge_depth = 1)

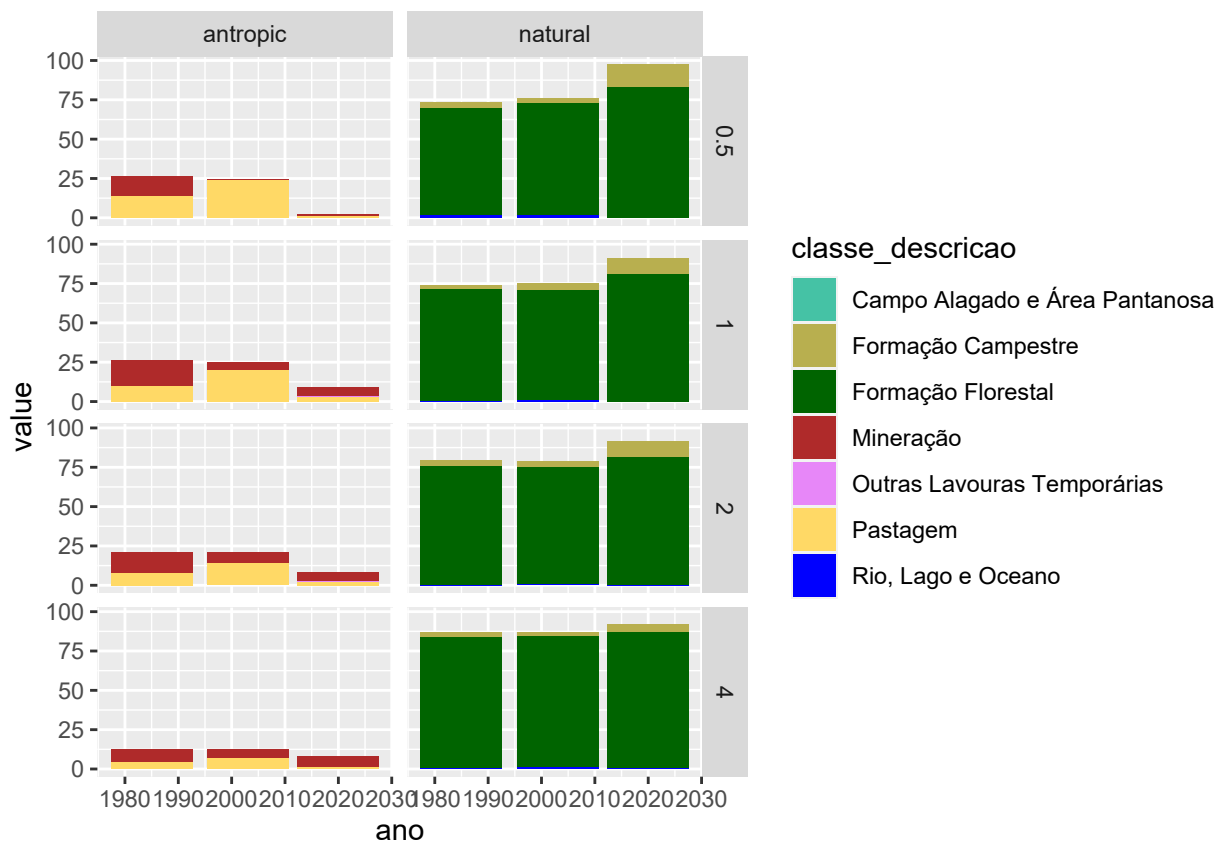
# Organizar dados
# Dados referentes os buffers
resultados_anos <- acesso_buffers %>%
  left_join(metricas_anos %>%
    dplyr::mutate(value = round(value,2),
                 ano = case_when(layer==1 ~1985,
                                layer==2~2003,
                                layer==3~2020)) %>%
    dplyr::select(ano, plot_id, class, metric, value),
    by=c("raio_km"="plot_id"))
```

```
# Dados referentes os classes
resultados_anos <- resultados_anos %>%
left_join(class_nomes, by = c("class" = "aid"))
```

Agora grafico de barra com varios anos

```
# It's recommended to use a named vector
# Legenda nomes ordem alfabetica
classe_cores <- c("Campo Alagado e Área Pantanosa" = "#45C2A5",
"Formação Campestre" = "#B8AF4F",
"Formação Florestal" = "#006400",
"Formação Savânica" = "#00ff00",
"Mineração" = "#af2a2a",
"Pastagem" = "#FFD966",
"Rio, Lago e Oceano" = "#0000FF",
"Outras Lavouras Temporárias" = "#e787f8")
```

```
resultados_anos %>%
  mutate(rcor = paste("#", hexadecimal_code, sep="")) %>%
  filter(metric=="pland") %>%
  ggplot(aes(x=ano, y=value)) +
  geom_col(position="stack", aes(fill=classe_descricao)) +
  scale_fill_manual(values = classe_cores) +
  facet_grid(raio_km~type_class)
```



Agora com “pland” e densidade de manchas juntos.

```

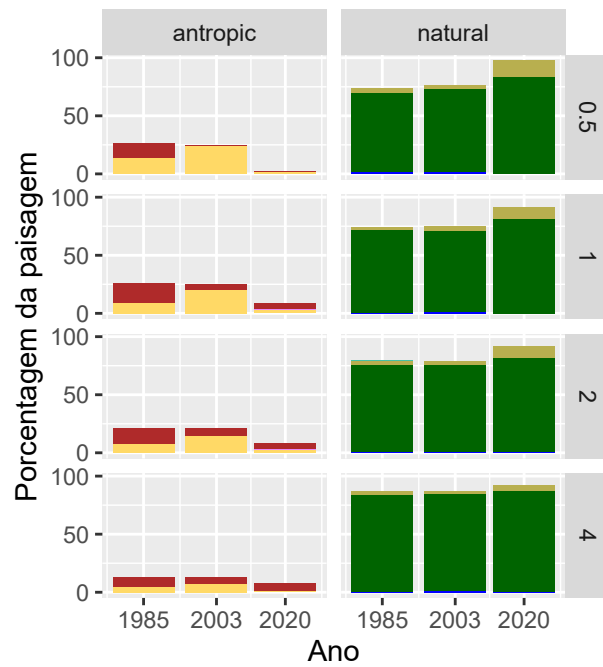
resultados_anos %>%
  mutate(rcor = paste("#", hexadecimal_code, sep="")) %>%
  filter(metric=="pland") %>%
  ggplot(aes(x=ano, y=value)) +
  geom_col(position="stack", aes(fill=classe_descricao)) +
  scale_fill_manual("classe", values = classe_cores) +
  scale_y_continuous(breaks=c(0,50,100)) +
  scale_x_continuous(breaks=c(1985,2003, 2020)) +
  facet_grid(raio_km~type_class) +
  labs(title = "MapBiomias cobertura da terra",
  subtitle = "Entorno do Garimpo do Lorenço 1985-2020",
  y = "Porcentagem da paisagem",
  x = "Ano") +
  theme(legend.position="bottom",
        legend.title = element_text(size = 3),
        legend.text = element_text(size = 3),
        legend.key.size = unit(0.1, "lines")) +
  guides(fill = guide_legend(nrow = 4)) -> fig_pland

# Densidade de manchas
resultados_anos %>%
  mutate(rcor = paste("#", hexadecimal_code, sep="")) %>%
  filter(metric=="pd") %>%
  ggplot(aes(x = factor(ano), y = value)) +
  geom_boxplot(colour = "grey50") +
  geom_point(aes(colour = classe_descricao)) +
  scale_color_manual("classe", values = classe_cores) +
  scale_size(guide = "none") +
  facet_grid(raio_km~type_class) +
  labs(title = "MapBiomias cobertura da terra",
  subtitle = "Entorno do Garimpo do Lorenço 1985-2020",
  y = "Densidade de manchas (número por 100 hectares)",
  x = "Ano") +
  theme(legend.position="bottom",
        legend.title = element_text(size = 3),
        legend.text = element_text(size = 3),
        legend.key.size = unit(0.1, "lines")) +
  guides(fill = guide_legend(nrow = 4)) -> fig_pd

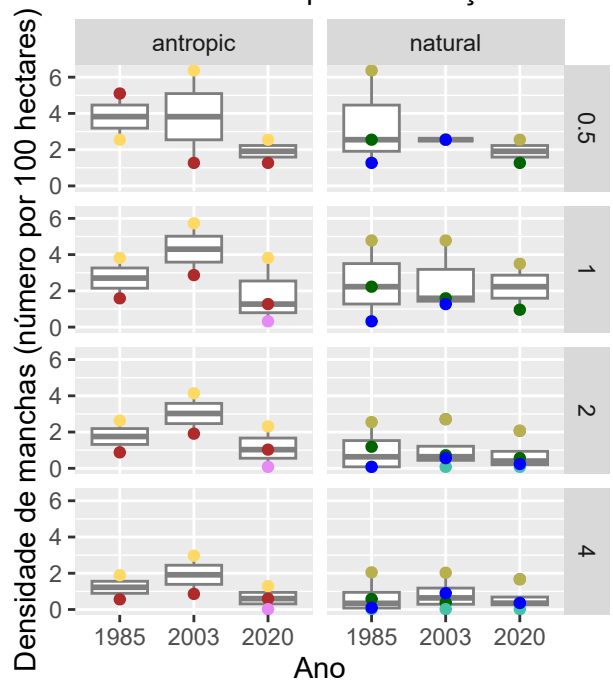
grid.arrange(fig_pland, fig_pd, nrow=1)

```

MapBiomas cobertura da terra
Entorno do Garimpo do Lorenço 1985-2021



MapBiomas cobertura da terra
Entorno do Garimpo do Lorenço 1985-2022



12 Conclusões e próximos passos

Os resultados apresentados na figura anterior não seguem os resultados esperados que a cobertura de classes antrópicas ia aumentar ao longo do tempo. Para entender melhor os padrões, precisamos:

- Verificar padrões nas outras métricas calculadas
- Verificar padrões com mais anos
- Verificar padrões usando polígonos/pontos dos processos de mineração (dados no SIGMINE <https://www.gov.br/anm/pt-br> e https://app.anm.gov.br/dadosabertos/SIGMINE/PROCESSOS_MINERARIOS/)

Além disso, seria relevante buscar complementar os dados de MapBiomas com uma classificação supervisionada usando imagens de Sentinel-2 (exemplo com QGIS Semi-Automatic Classification plugin aqui: <https://fromgistors.blogspot.com/2016/09/basic-tutorial-2.html>). Assim para aumentar a precisão dos resultados.

Uma forma alternativa para visualização das mudanças entre anos seria um diagrama “Sankey”/“Alluvial”. Como exemplo, veja figura 3 no artigo “Rapid land use conversion in the Cerrado has affected water transparency in a hotspot of ecotourism, Bonito, Brazil” <https://doi.org/10.1177/19400829221127087>.

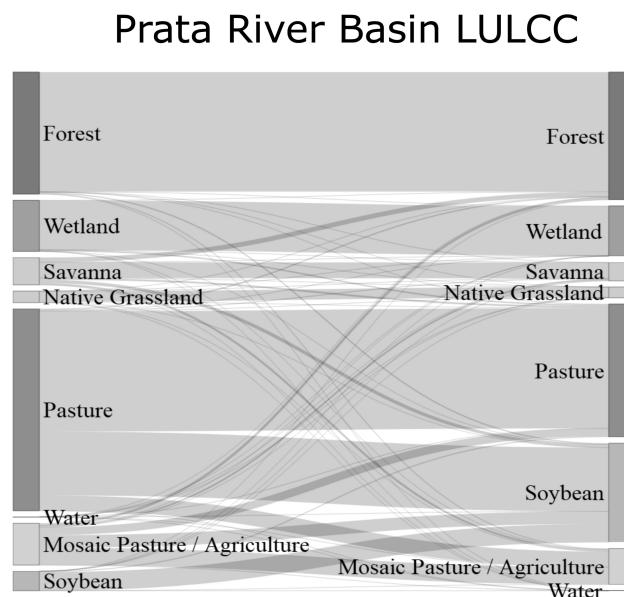


Figura 12.1: Diagrama Sankey mostrando a mudança de uso da terra na bacia do rio Prata entre 2010 (lado esquerdo) e 2020 (lado direito). Fonte: Figura 3. Chiaravalloti et. al. 2022. Tropical Conservation Science. doi:10.1177/19400829221127087

No R pode fazer com o pacote “networkD3” segue tutoriais:

- <https://www.displayr.com/sankey-diagrams-r/>
- <https://epirhandbook.com/en/diagrams-and-charts.html#alluvialsankey-diagrams>
- <https://rpubs.com/droach/CP526-codethrough>