

# Lab05

*Vanderlin Amorim Júnior - Curso Verão 2016 - Curso R  
25 de fevereiro de 2016*

## Aula 05 - Laboratório

### Recuperação de imagem

A bibliotecas utilizadas

```
library(magrittr)
library(ggplot2)
library(tidyr)

## 
## Attaching package: 'tidyr'

## The following object is masked from 'package:magrittr':
## 
##     extract

library(dplyr)

## 
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

library(jpeg)
library(tree)
```

### Preparação do Banco de dados

Para construir nossos bancos de dados, carregue as duas imagens abaixo (clique para download e salve na pasta do seu projeto):

```
## Imagem: purple_wave.jpg
```

```



```

```
set.seed(1)
```

parte1 b= 0

```

img_df_parte1 <- img_df %>%
  sample_frac(3/5) %>% # separando 3/5 do banco
  mutate(b_backup = b, # backup do azul original
         b = 0, # retirando o azul da imagem
         cor = rgb(r, g, b)) # cor da imagem sem o azul

```

total de pixel agora:

```
dim(img_df_parte1)
```

```
## [1] 42336      8
```

filtra os que não estão no img\_df\_parte1

```
img_df_parte2 <- img_df %>% filter(!id %in% img_df_parte1$id)
```

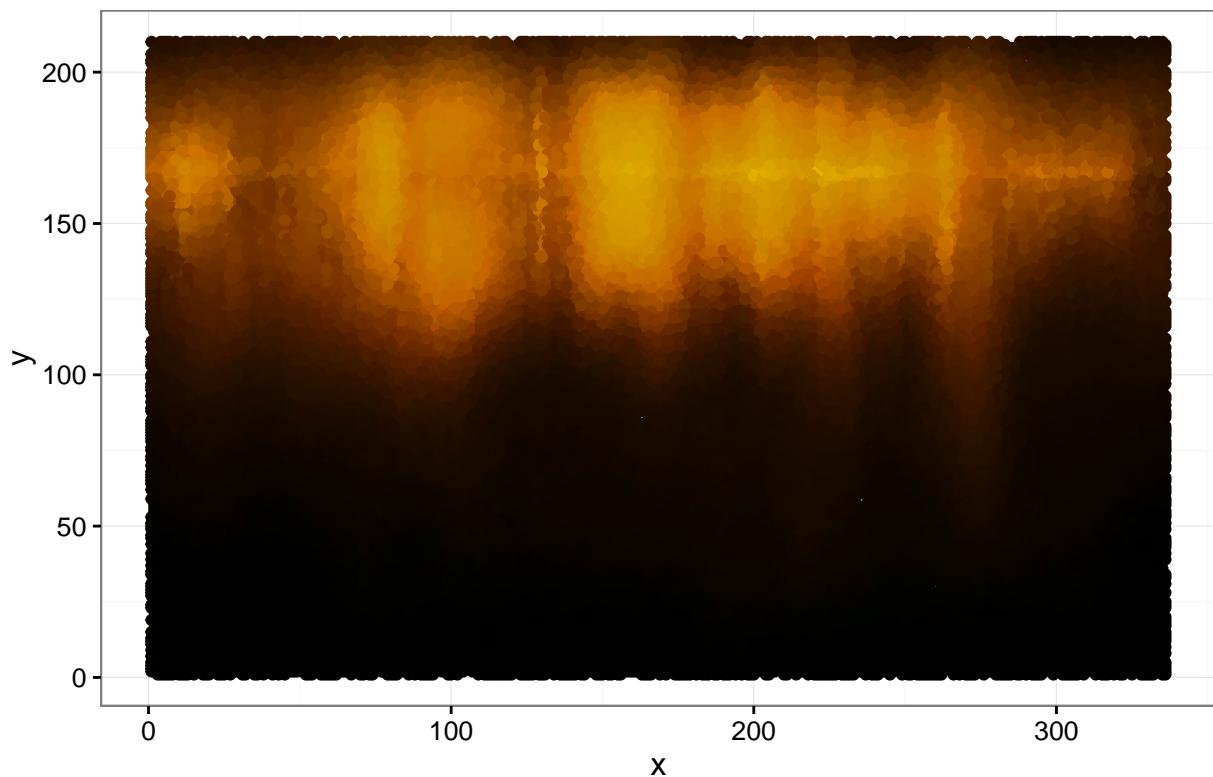
Veja como fica a imagem original sem o azul

```

ggplot(data = img_df_parte1, aes(x = x, y = y)) +
  geom_point(colour = img_df_parte1$cor) +
  labs(x = "x", y = "y", title = "Imagen sem B (azul)") +
  coord_fixed(ratio = 1) +
  theme_bw()

```

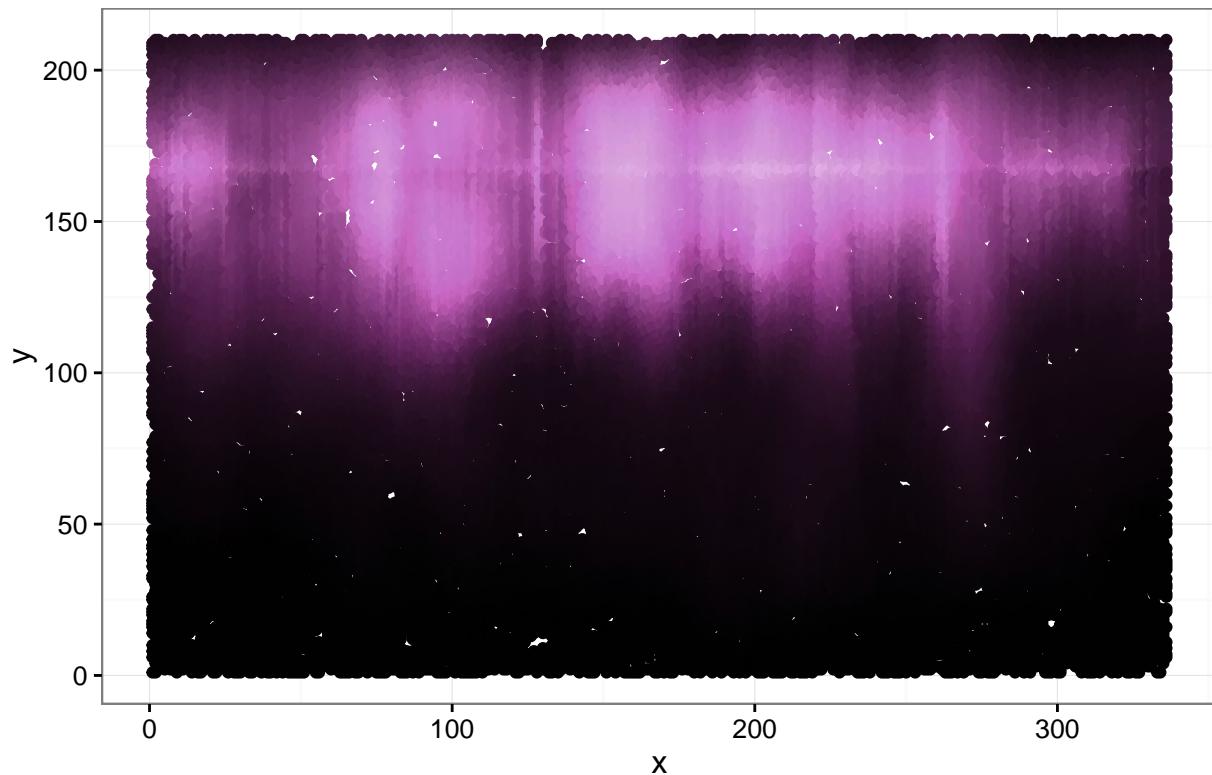
Imagen sem B (azul)



Veja como fica a imagem como é o azul original isolado.

```
ggplot(data = img_df_parte2, aes(x = x, y = y)) +  
  geom_point(colour = img_df_parte2$cor) +  
  labs(x = "x", y = "y", title = "Imagen sem B (azul)") +  
  coord_fixed(ratio = 1) +  
  theme_bw()
```

Imagen sem B (azul)



Para construir nossos bancos de dados, carregue as duas imagens abaixo (clique para download e salve na pasta do seu projeto):

```
## Imagem: xadrez_colorido.jpg
```

Armazendo a imagem

```
img <- readJPEG("xadrez_colorido.jpg")
img_dim <- dim(img)

img_df <- data.frame(
  x = rep(1:img_dim[2], each = img_dim[1]),
  y = rep(img_dim[1]:1, img_dim[2]),
  r = as.vector(img[,1]),
  g = as.vector(img[,2]),
  b = as.vector(img[,3]))
) %>%
  mutate(cor = rgb(r, g, b),
        id = 1:n())
```

```



```

```



```

mais codigo:

```



```

```

## [1] 11059      8

```

```



```

## Exercício 5: Outra Imagem

Repita os exercícios de 1 a 4, mas agora para a imagem xadrez\_colorido.jpg. Quanto ao desempenho para recuperar o azul de imagens, teve uma técnica melhor? Resposta: foi pela técnica da árvores de decisão

pelo modelo da regressão linear

```

modelo_lm = lm(b ~ r + g + x + y, data = img_df_parte2)

```

pelo modelo da árvores de decisão

```

modelo_tree = tree(b ~ r + g + x + y, data = img_df_parte2)

```

```



```

```

##  Mean    : 72.62   Mean    : 64.73   Mean    :0.411706   Mean    :0.250854
##  3rd Qu.:109.00   3rd Qu.: 97.00   3rd Qu.:0.945098   3rd Qu.:0.505882
##  Max.   :144.00   Max.   :128.00   Max.   :1.000000   Max.   :1.000000
##      b          cor           id          b_backup
##  Min.   :0   Length:11059   Min.   : 1   Min.   :0.000000
##  1st Qu.:0   Class :character  1st Qu.: 4602  1st Qu.:0.003922
##  Median :0   Mode  :character  Median : 9224  Median :0.027451
##  Mean   :0                   Mean   : 9232  Mean   :0.173376
##  3rd Qu.:0                   3rd Qu.:13896 3rd Qu.:0.145098
##  Max.   :0                   Max.   :18432   Max.   :1.000000
##      predito_lm      predito_tree
##  Min.   :0.04973  Min.   :0.03826
##  1st Qu.:0.13034  1st Qu.:0.04634
##  Median :0.15759  Median :0.04877
##  Mean   :0.17327  Mean   :0.17412
##  3rd Qu.:0.21155  3rd Qu.:0.04877
##  Max.   :0.31476  Max.   :0.90381

```

```
?summary
```

```
## starting httpd help server ...
```

```
## done
```

```
mean((mdf_predito_x$b_backup-mdf_predito_x$predito_lm)^2)
```

```
## [1] 0.08951189
```

```
mean((mdf_predito_x$b_backup-mdf_predito_x$predito_tree)^2)
```

```
## [1] 0.007416139
```

```

mdf_predito_cores <- mdf_predito_x %>%
  mutate(predito_lm = ifelse(predito_lm < 0, 0, predito_lm),
         predito_tree = ifelse(predito_tree < 0, 0, predito_tree)) %>%
  mutate(cor_lm = rgb(r, g, predito_lm),
         cor_tree = rgb(r, g, predito_tree))

```

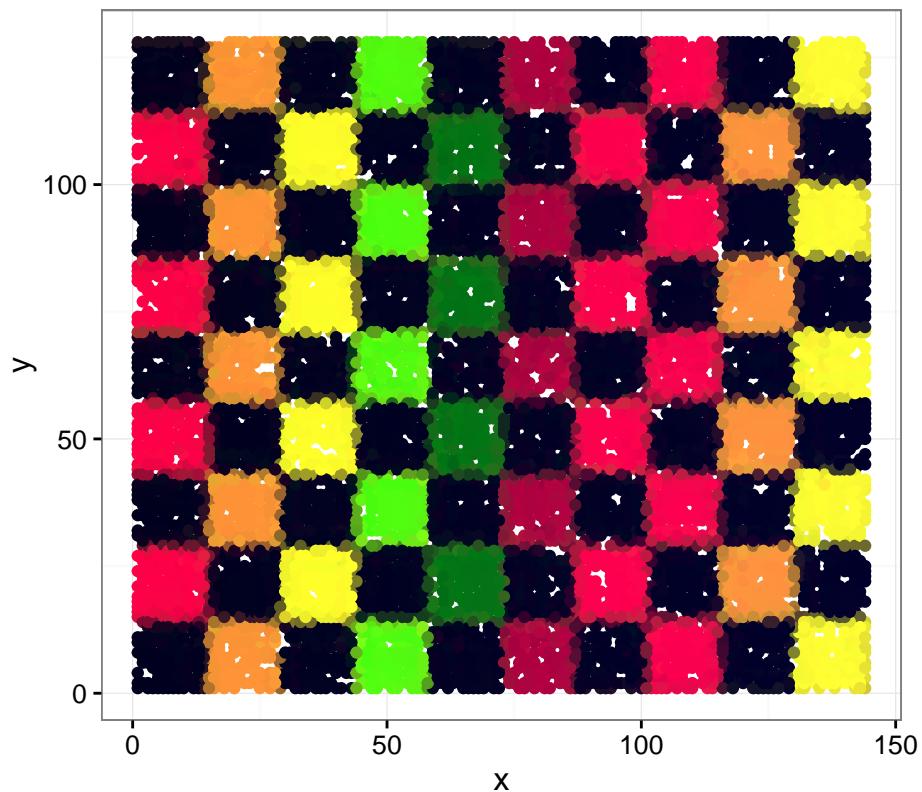
## Tentativa de recuperação do azul pela regressão linear

```

ggplot(mdf_predito_cores, aes(x = x, y = y)) +
  geom_point(colour = mdf_predito_cores$cor_lm) +
  labs(x = "x", y = "y", title = "Imagen sem B (azul)") +
  coord_fixed(ratio = 1) +
  theme_bw()

```

Imagen sem B (azul)



Tentativa de recuperação do azul pela árvores de decisão

```
ggplot(mdf_predito_cores, aes(x = x, y = y)) +  
  geom_point(colour = mdf_predito_cores$cor_tree) +  
  labs(x = "x", y = "y", title = "Imagen sem B (azul)") +  
  coord_fixed(ratio = 1) +  
  theme_bw()
```

Imagen sem B (azul)

