

# veinGTFS

*Sergio*

31/10/2019

## Dados

```
library(data.table)
dt <- fread("C:/Users/Sergio/Downloads/70912_first1000.txt")

## Warning in fread("C:/Users/Sergio/Downloads/70912_first1000.txt"): Detected
## 13 column names but the data has 14 columns (i.e. invalid file). Added 1
## extra default column name for the first column which is guessed to be row
## names or an index. Use setnames() afterwards if this guess is not correct,
## or fix the file write command that created the file to create a valid file.

head(dt)

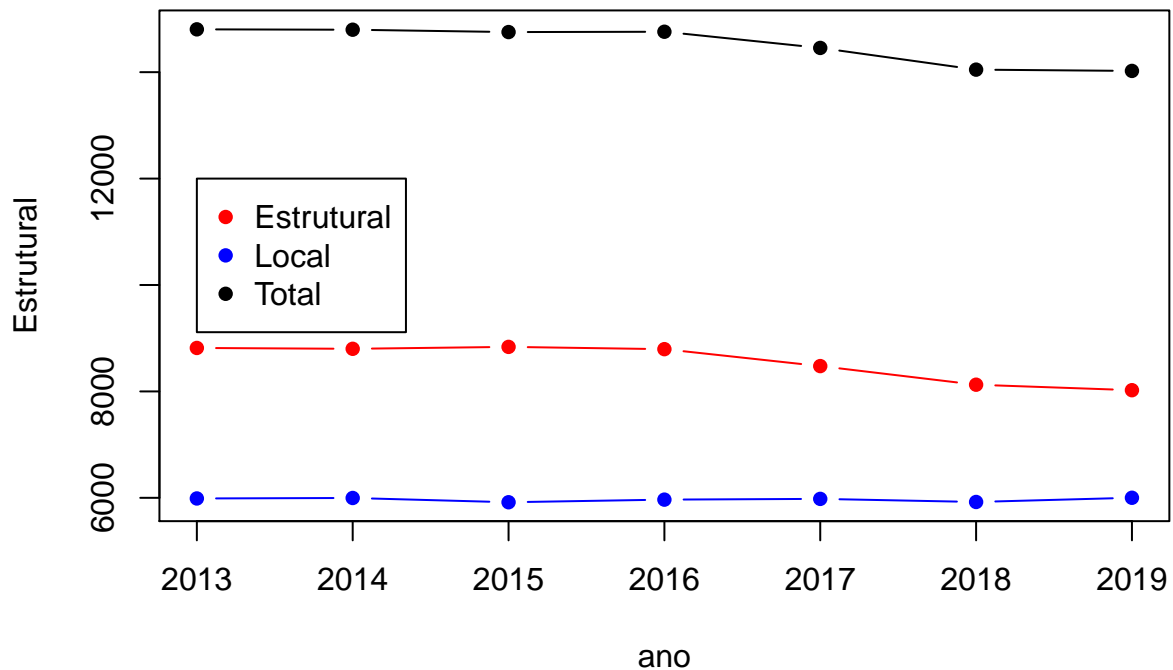
##      V1   trip_id route_type id shape_pt_lon shape_pt_lat departure_time
## 1:  1 573T-31-0          3  1   -46.50077    -23.61420         05:00:00
## 2:  2 573T-31-0          3  2   -46.50081    -23.61420         05:00:00
## 3:  3 573T-31-0          3  3   -46.50089    -23.61421         05:00:02
## 4:  4 573T-31-0          3  4   -46.50097    -23.61422         05:00:04
## 5:  5 573T-31-0          3  5   -46.50102    -23.61423         05:00:05
## 6:  6 573T-31-0          3  6   -46.50115    -23.61424         05:00:08
##      stop_id stop_sequence      dist   cumdist   speed   cumtime shape_id
## 1: 780006744             1  4.623534  4.623534  17.64989  0.9430498    70912
## 2:      NA             NA  8.423626 13.047160  17.64989  2.6611942    70912
## 3:      NA             NA  8.423626 21.470786  17.64989  4.3793385    70912
## 4:      NA             NA  4.826108 26.296894  17.64989  5.3637068    70912
## 5:      NA             NA 12.945892 39.242787  17.64989  8.0042457    70912
## 6:      NA             NA  5.750804 44.993591  17.64989  9.1772217    70912
```

Pelo jeito, seria possivel sim estimar as emissões pois temos a distancia em metros e a velocidade. Porém, precisamos identificar o tipo específico de ônibus para associar com o fator de emissão específico para aí estimar as emissões.

Não estou enxergando o campo que diz se é um ônibus diferente ou não. Parece que trip\_id representaria um ônibus.

A frota é

```
ano = 2013:2019
Estrutural = c(8817, 8801, 8837, 8794, 8476, 8126, 8023)
Local = c(5988, 5997, 5917, 5966, 5980, 5922, 6001)
Total = c(14805, 14798, 14754, 14760, 14456, 14048, 14024)
plot(x = ano, y = Estrutural, type = "b", pch = 16, col = "red",
     ylim = c(min(Local), max(Total)))
points(x = ano, y = Local, type = "b", pch = 16, col = "blue")
points(x = ano, y = Total, type = "b", pch = 16, col = "black")
legend(x = 2013, y = 12000, pch = 16,
       legend = c("Estrutural", "Local", "Total"),
       col = c("red", "blue", "black"))
```



Ano	GRUPO ESTRUTURAL	GRUPO LOCAL DE ARTICULAÇÃO REGIONAL	GRUPO LOCAL DE DISTRIBUIÇÃO
2019*	5 anos e 3 meses	5 anos e 7 meses	4 anos e 10 meses

[https://www.prefeitura.sp.gov.br/cidade/secretarias/transportes/institucional/sptrans/acesso\\_a\\_informacao/index.php?p=245223](https://www.prefeitura.sp.gov.br/cidade/secretarias/transportes/institucional/sptrans/acesso_a_informacao/index.php?p=245223)

Logo, aplicando uma curva de sobrevivencia, ou seja, descontando a frota chatarrizada fica

```
library(vein)
tot_cir <- age(x = rev(Total), type = "double_logistic", a = 0.16, b = 19.1, verbose = T)

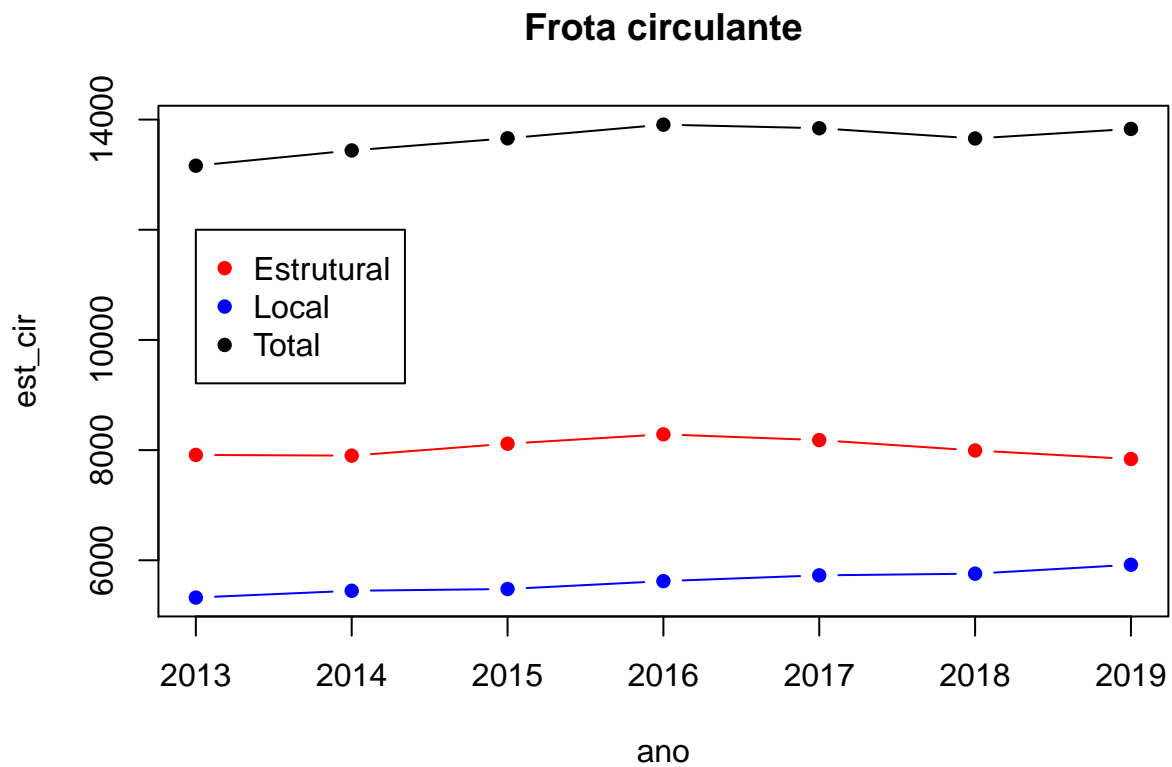
## Number of vehicles is 95.5 * 10^3 veh

est_cir <- age(x = rev(Estrutural), type = "double_logistic", a = 0.16, b = 19.1, verbose = T)

## Number of vehicles is 56.23 * 10^3 veh

loc_cir <- age(x = rev(Local), type = "double_logistic", a = 0.16, b = 19.1)
plot(x = ano, y = est_cir, type = "b", pch = 16, col = "red",
     ylim = c(min(loc_cir), max(tot_cir)), main = "Frota circulante")
points(x = 2019:2013, y = loc_cir, type = "b", pch = 16, col = "blue")
points(x = 2019:2013, y = tot_cir, type = "b", pch = 16, col = "black")
legend(x = 2013, y = 12000, pch = 16,
       legend = c("Estrutural", "Local", "Total"),
```

```
col = c("red", "blue", "black"))
```



## Composicao veicular

Tem varios tipos de onibus, articulado, bi-articulado, standard, mini

## Composicao tecnologica

A frota de onibus de Sao Paulo tem onibus eletricos e a diesel. seria bom saber o trajeto dos onibus eletricos pois eles emitem so material particulado dos freis, pneus e abrasao da rua

Por tanto, acredito qu estos problemas poderiam ser resolvidos aplicando uma distribuicao estatistica da ultima figura. Podemos pegar a frota cirlante total como regra, que e mais o menos uma reta, entao, a probabilidade de pegar onibus de diferente idade e quase a mesma (falando a grosso modo). Asumindo que o veiculo mais velho em circulacao e efetivamente o 2013, os veiculos vao desde 2019 ate 2013, e eles tem a mesma probabilidade de estarem circulacao.

```
length(2019:2013)
```

```
## [1] 7
```

## Fatores de emissao"de onibus

Podemos escolher? "UB", "SUB", "COACH", "ARTIC"

O FE mais novo e 2017, mas colocando um FE mais novo, VEIN projeta repetindo o ultimo FE mais novo por default

```
ub_co <- ef_cetesb(p = "CO", veh = "UB", year = 2019)[1:7]
ub_co_2019 <- ub_co[1]
ub_co_2018 <- ub_co[2] # ...
```

## Emissões

Para simplificar e assumindo um caso bem simples, que todo é a mesma coisa. Como os dados já tem a distância percorrida e a velocidade, calcular a emissão só seria direto, sem emiss e tal.

Assumindo que todos os ônibus são 2019

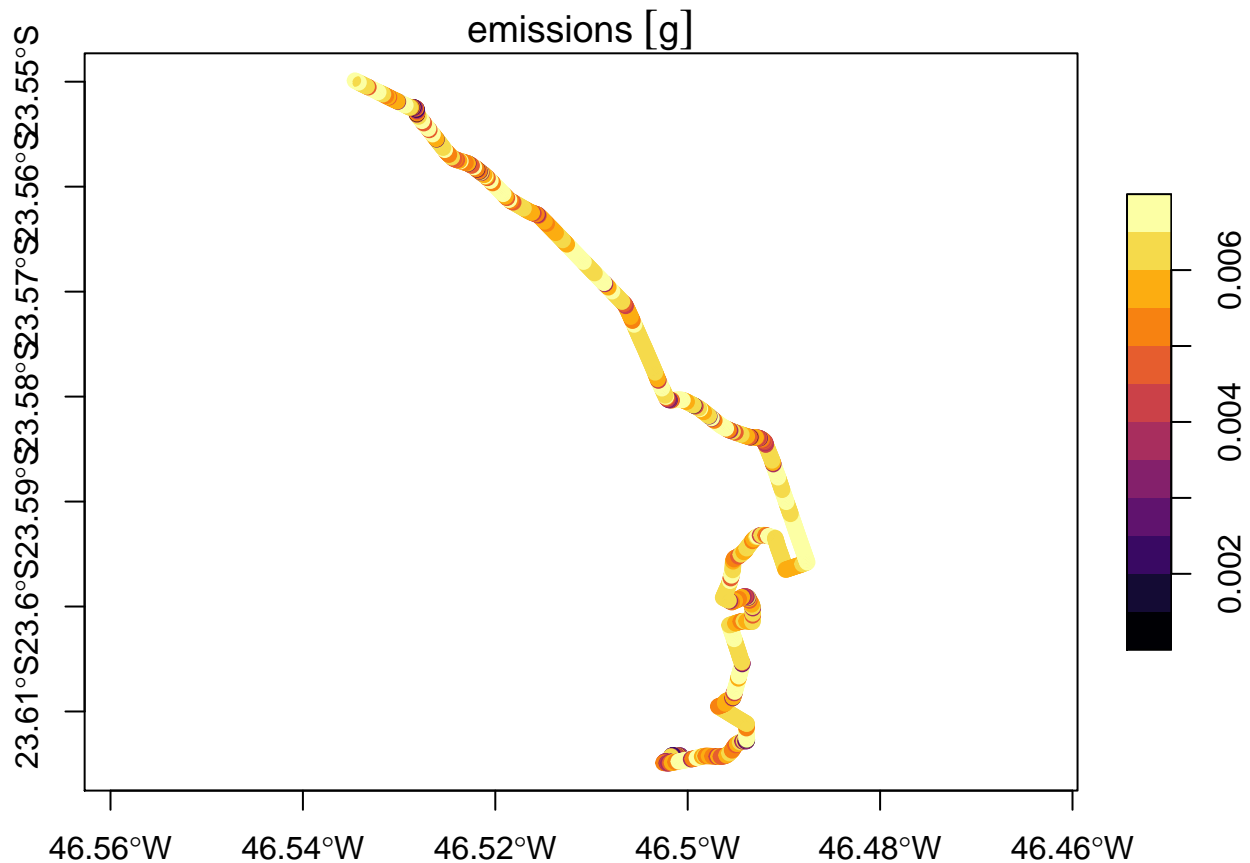
```
library(units)
```

```
## udunits system database from C:/Users/Sergio/Documents/R/win-library/3.6/units/share/udunits
```

```
library(sf)
```

```
## Linking to GEOS 3.6.1, GDAL 2.2.3, PROJ 4.9.3
```

```
library(cptcity)
dt$veh <- 1
dt$dist_m <- set_units(dt$dist, "m")
dt$1km <- set_units(dt$dist_m, "km")
dt$emissions <- dt$veh * ub_co_2019 * dt$1km
dft <- st_as_sf(dt, coords = c("shape_pt_lon", "shape_pt_lat"), crs = 4326)
plot(dft["emissions"], axes = T, pal = cpt(colorRampPalette = T))
```



```

g_001 <- make_grid(dft, 0.001)

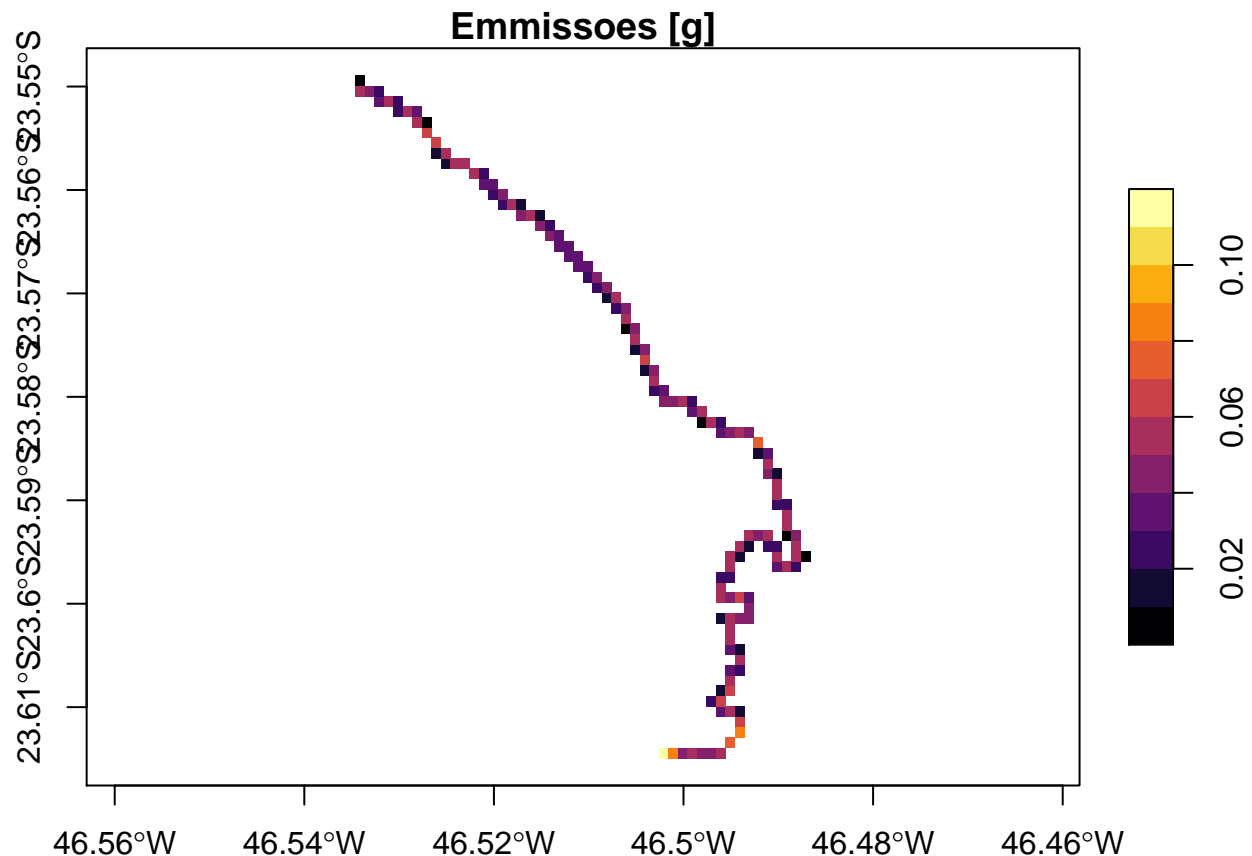
## Number of lon points: 48
## Number of lat points: 66

gx <- emis_grid(spobj = dft[, "emissions"], g = g_001, type = "points")

## Sum of point emissions 5.8
## Sum of gridded emissions 5.8

gx <- gx[gx$emissions > 0, ]
plot(gx["emissions"], axes = T, pal = cpt(colorRampPalette = T), lty = 0,
     main = "Emmissoes [g]")

```



## Esqueci incorporar a velocidade! (rs)

Para transformar os fatores da cetesb em funcao de velocidade, temos que saber que euro e cada onibus, eso se sabe pelo ano, e dando uma olhada na pagina <https://www.transportpolicy.net/standard/brazil-heavy-duty-emissions/> mas como desde 2012 todos sao Euro V, fica facil

```

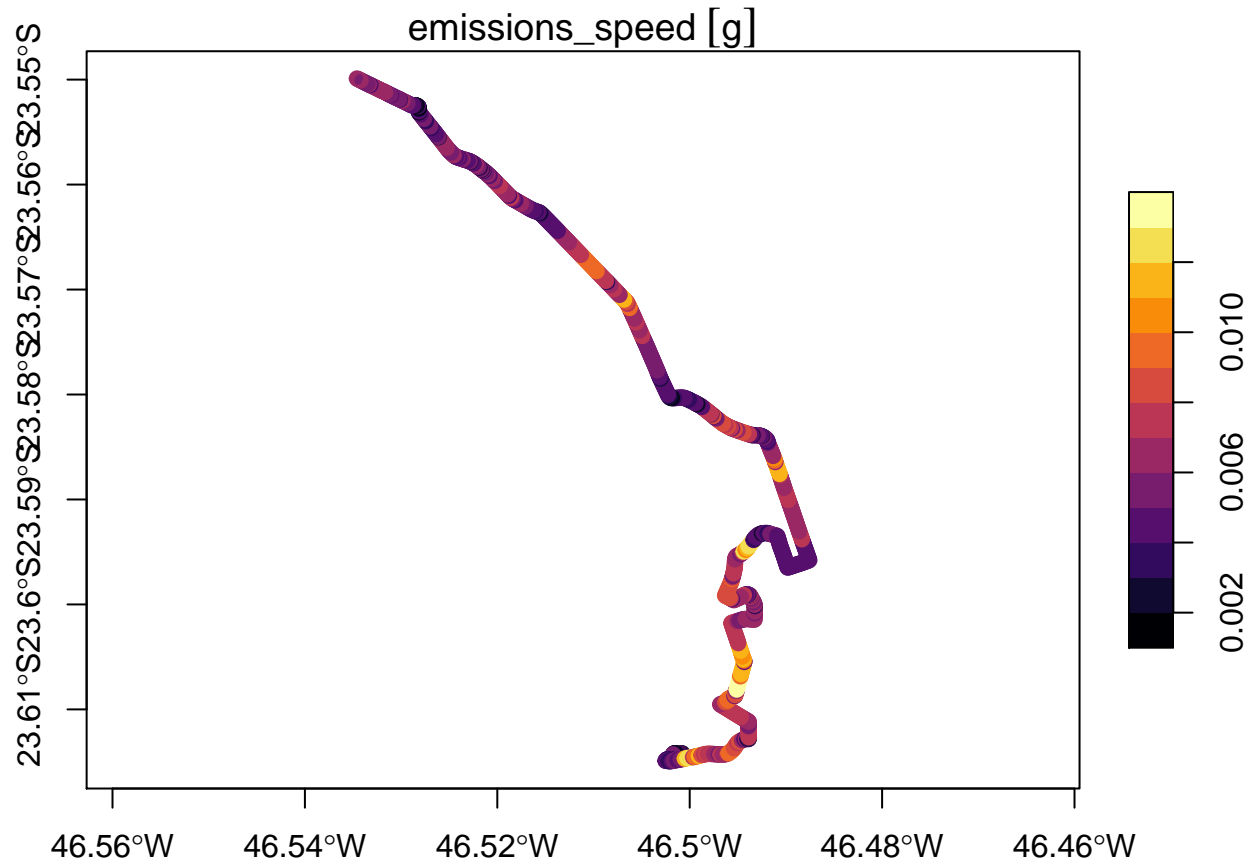
ef <-
ef_ub_co <- ef_hdv_scaled(dfcol = ub_co, v = "Ubus", t = "Std",
g = ">15 & <=18", eu = rep("V", 7), gr = 0, l = 0.5, p = "CO")
ef_ub_co # 7 funcoes uma por ano, de 2019 a 2013

```

```
## This EmissionFactorsList has 7 functions
```

```
dt$emissions_speed <- dt$veh * ub_co_2019 * dt$lkm * ef_ub_co[[1]](dt$speed)

dft <- st_as_sf(dt, coords = c("shape_pt_lon", "shape_pt_lat"), crs = 4326)
plot(dft["emissions_speed"], axes = T, pal = cpt(colorRampPalette = T))
```



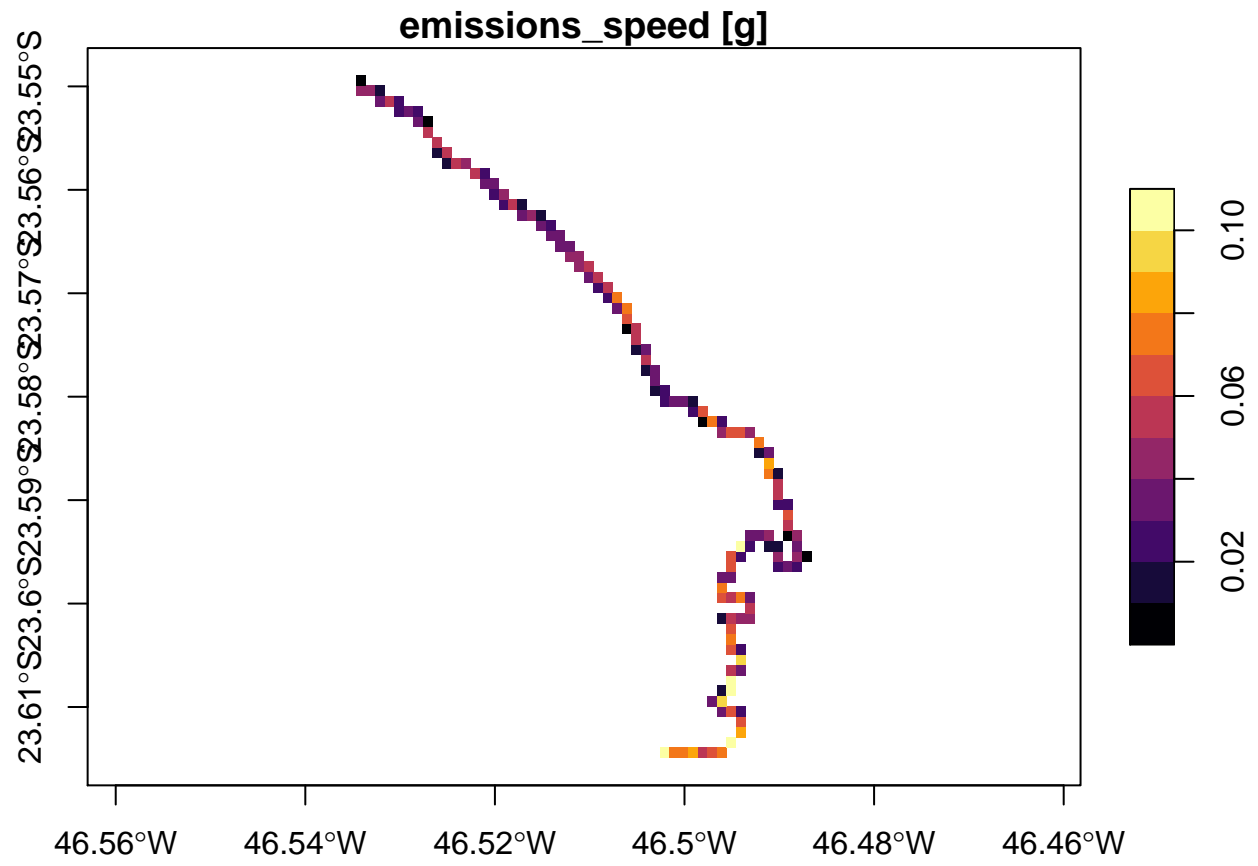
```
g_001 <- make_grid(dft, 0.001)

## Number of lon points: 48
## Number of lat points: 66

gx <- emis_grid(spobj = dft[, "emissions_speed"], g = g_001, type = "points")

## Sum of point emissions 6.35
## Sum of gridded emissions 6.35

gx <- gx[gx$emissions_speed > 0, ]
plot(gx["emissions_speed"], axes = T, pal = cpt(colorRampPalette = T), lty = 0,
     main = "emissions_speed [g]")
```



A soma total das emissões com velocidade e 6.35 [g] e sem velocidade 5.8 [g]