Tarea clase 4 en R mapas

Sergio Carrero

2022-03-18

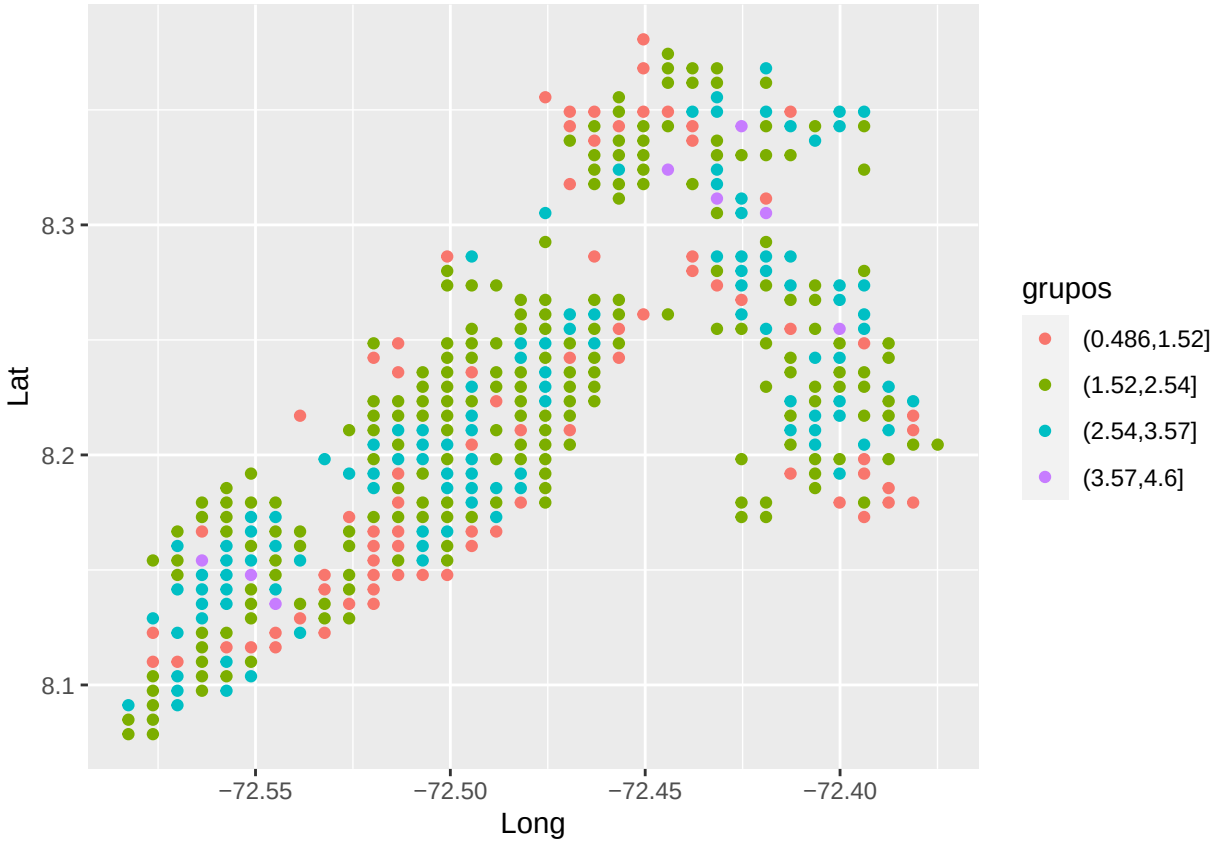
Estadisticas descriptivas

```
library(readxl)
XPABLO_2_ <- read_excel("~/Computacion estadistica/Clase 4/XPABLO (2).xlsx")
View(XPABLO_2_)
#df = data.frame(calcio = rnorm(n = , mean = , sd = )) # Dataframe -> Marco de datos, de forma tabular
```

```
MO =  XPABLO_2_$MO
grupos <- cut(x = XPABLO_2_$MO, breaks = 4) # cut -> Función para crear categorías
unique(grupos)
```

```
## [1] (1.52,2.54]  (2.54,3.57]  (0.486,1.52] (3.57,4.6]
## Levels: (0.486,1.52] (1.52,2.54] (2.54,3.57] (3.57,4.6]
```

```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = grupos)) + #Aesthetic "estetica = aes", #Mapeo de 2 pu
  geom_point() #Geom point genera los puntos
```

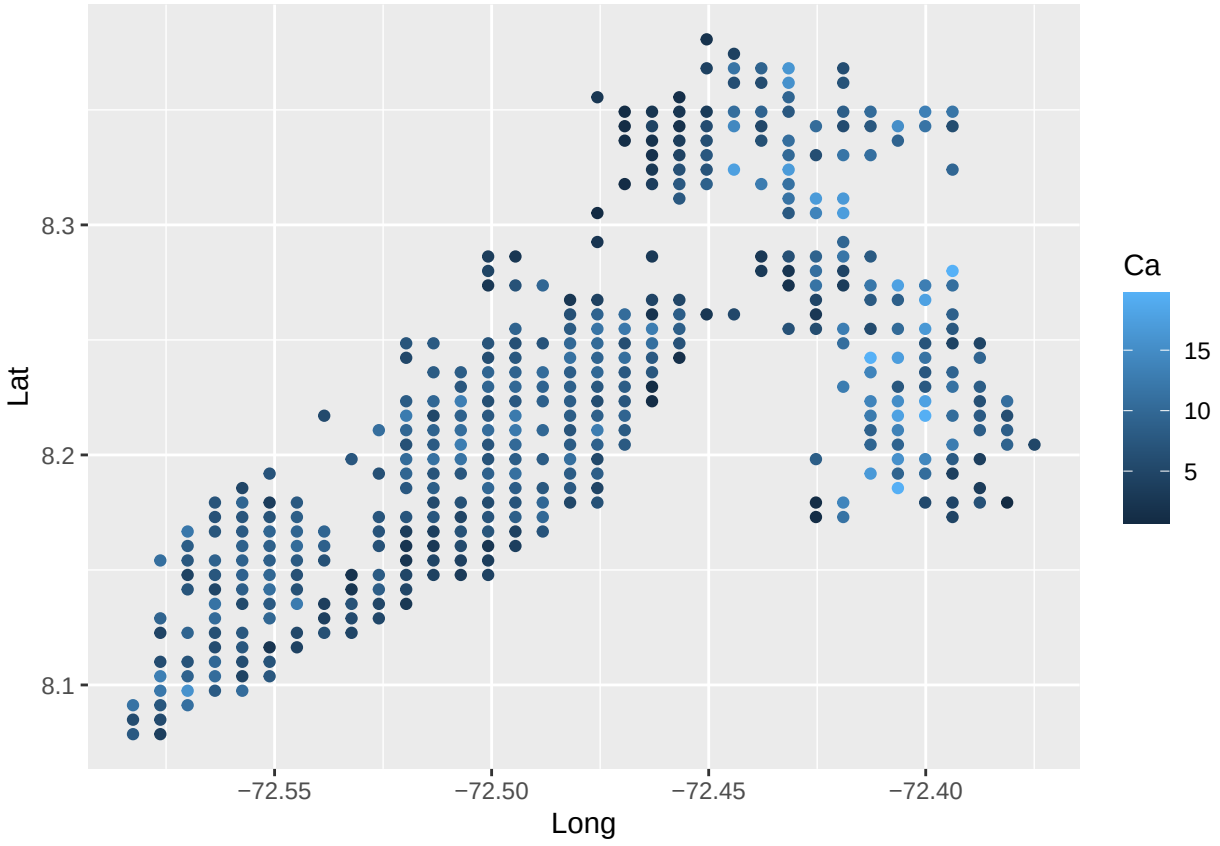



```
# si se requiere dividir por categorias hay funciones como cut2, fill
```

```
ca =  XPABLO_2_$Ca
grupos <- cut(x = XPABLO_2_$Ca, breaks = 4) # cut -> Función para crear categorías, con el 4 se crean 4
unique(grupos)
```

```
## [1] (5.49,10.2]  (0.722,5.49] (10.2,15]    (15,19.7]
## Levels: (0.722,5.49] (5.49,10.2] (10.2,15] (15,19.7]
```

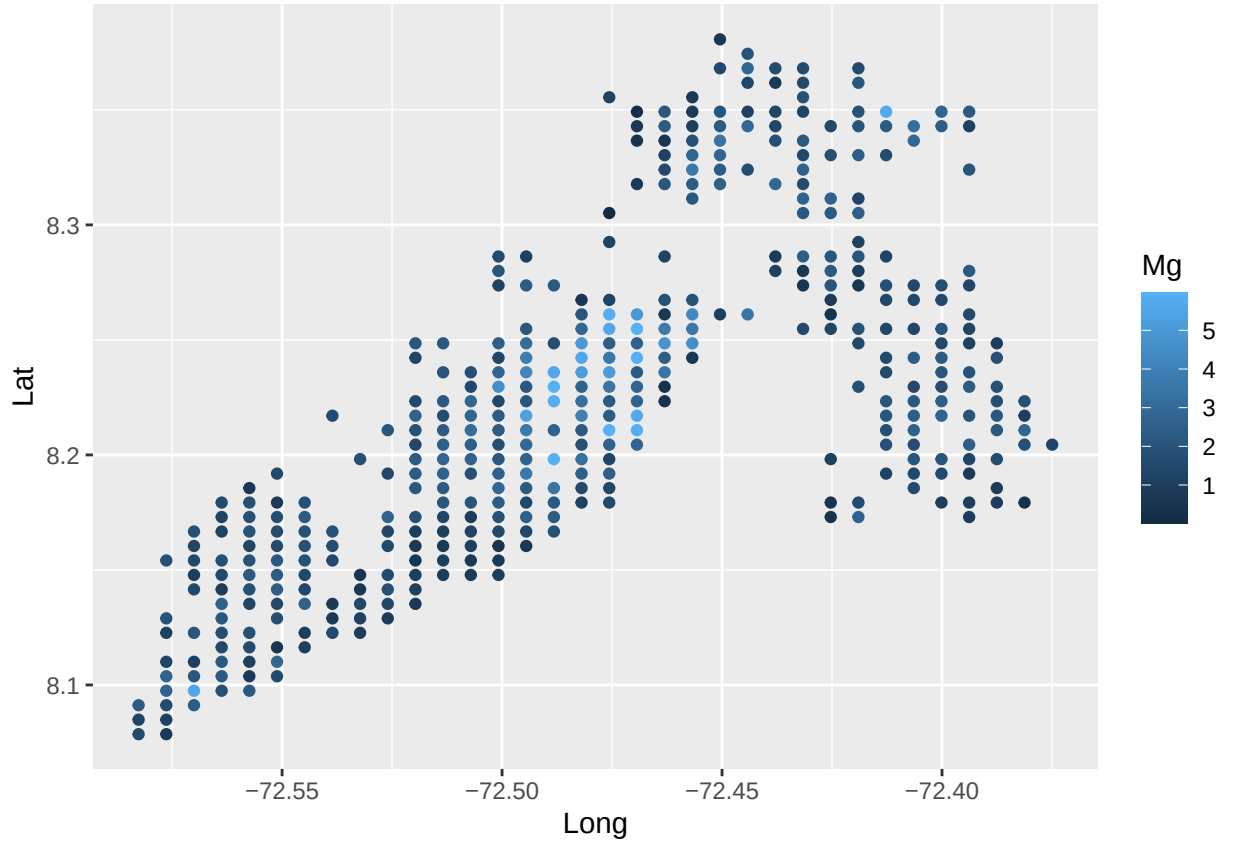
```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = Ca)) +#Aesthetic, puede usarse fill=color
  geom_point() #Geom point genera los puntos
```




```
Mg =  XPABLO_2_$Mg
grupos <- cut(x = XPABLO_2_$Mg, breaks = 4) # cut -> Función para crear categorías, con el 4 se crean 4
unique(grupos)
```

```
## [1] (1.51,3]      (0.0205,1.51] (4.48,5.97]   (3,4.48]
## Levels: (0.0205,1.51] (1.51,3] (3,4.48] (4.48,5.97]
```

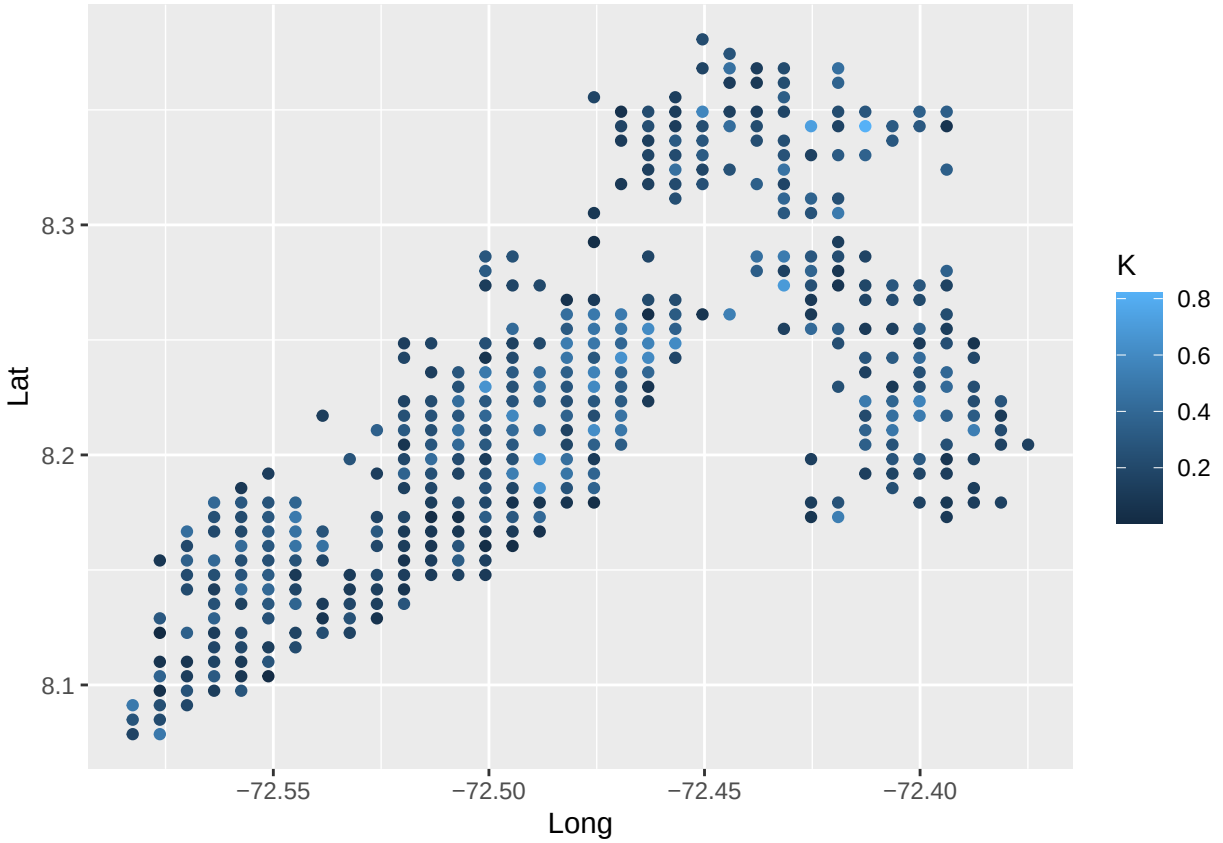
```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = Mg)) +#Aesthetic, puede usarse fill=color
  geom_point() #Geom point genera los puntos
```




```
K =  XPABLO_2_$K
grupos <- cut(x = XPABLO_2_$K, breaks = 4) # cut -> Función para crear categorías, con el 4 se crean 4 
unique(grupos)
```

```
## [1] (0.00442,0.209] (0.413,0.616]   (0.209,0.413]   (0.616,0.821]
## Levels: (0.00442,0.209] (0.209,0.413] (0.413,0.616] (0.616,0.821]
```

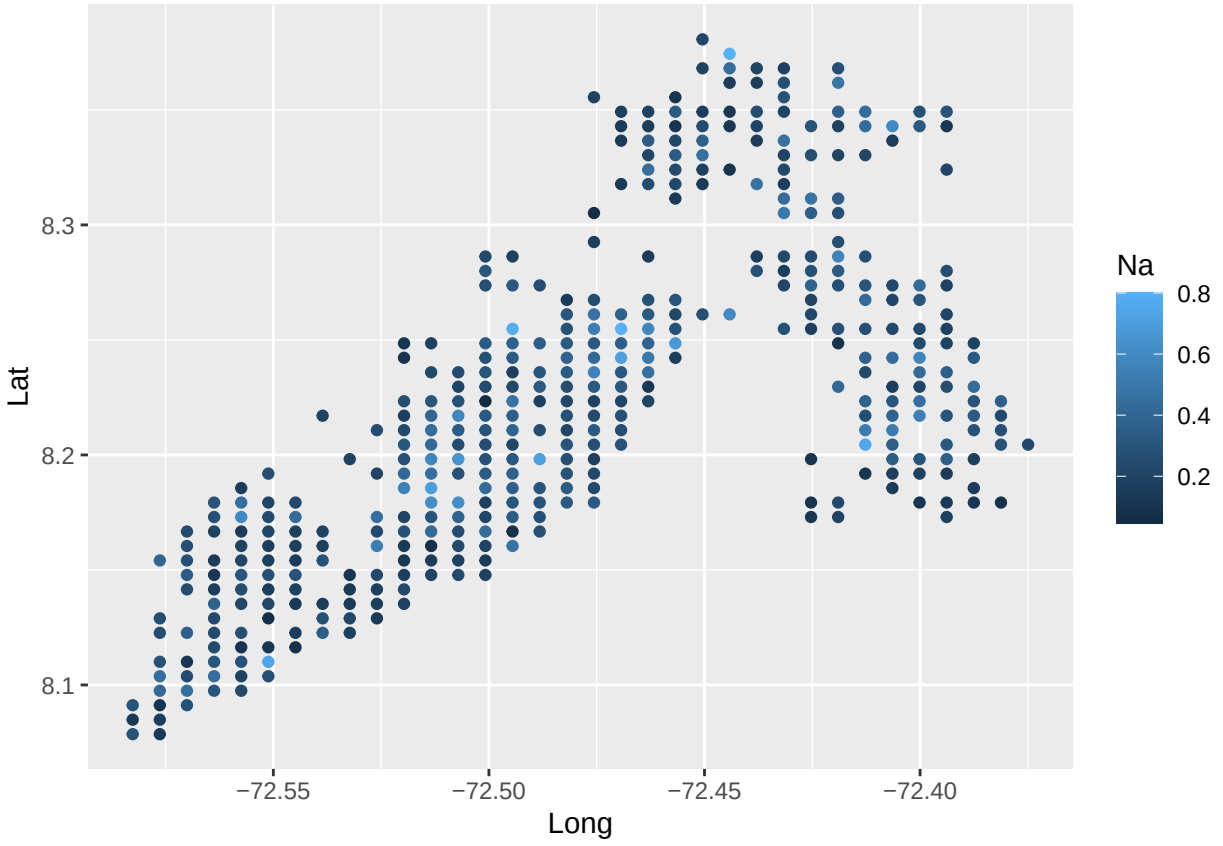
```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = K)) +#Aesthetic, puede usarse fill=color
  geom_point() #Geom point genera los puntos
```




```
Na =  XPABLO_2_$Na
grupos <- cut(x = XPABLO_2_$Na, breaks = 4) # cut -> Función para crear categorías, con el 4 se crean 4
unique(grupos)
```

```
## [1] (0.236,0.424]  (0.0474,0.236] (0.424,0.612]  (0.612,0.801]
## Levels: (0.0474,0.236] (0.236,0.424] (0.424,0.612] (0.612,0.801]
```

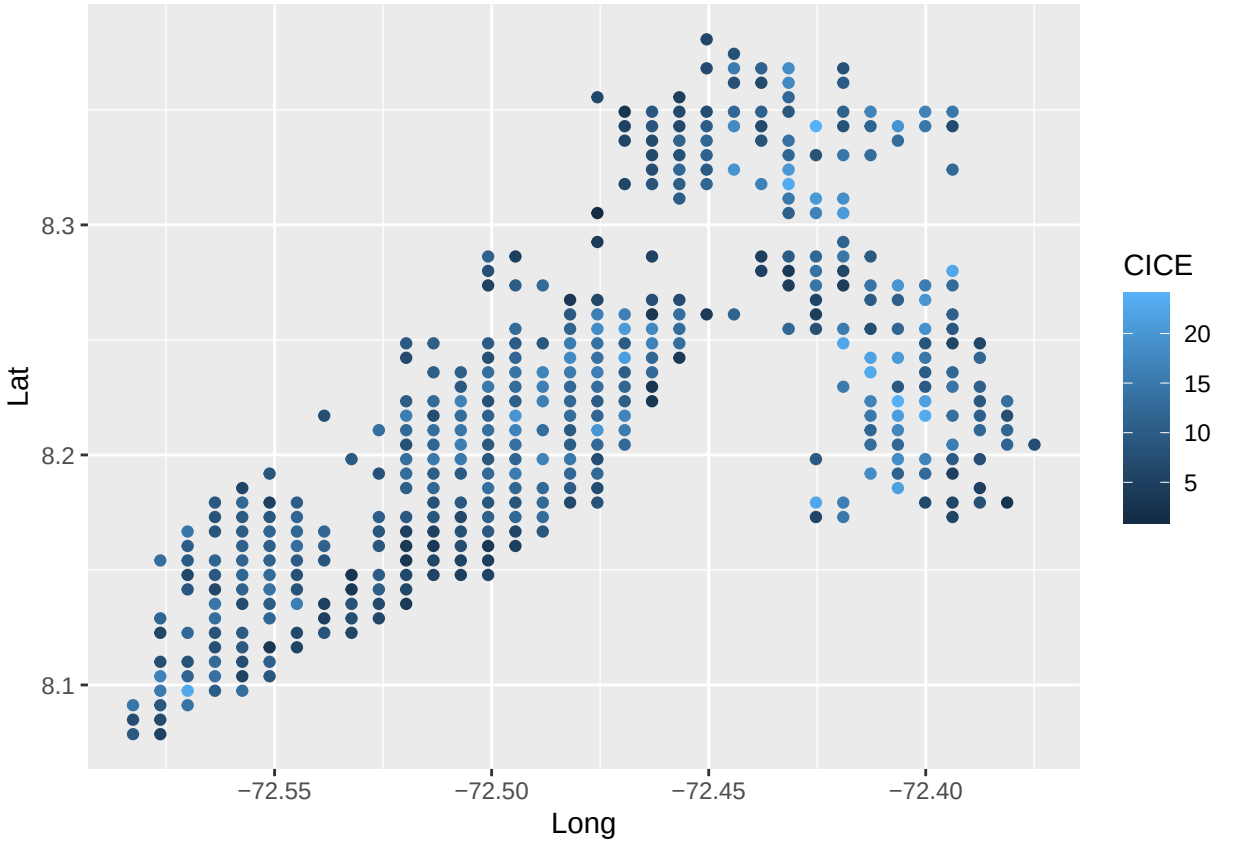
```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = Na)) +#Aesthetic, puede usarse fill=color
  geom_point() #Geom point genera los puntos
```




```
CICE =  XPABLO_2_$CICE
grupos <- cut(x = XPABLO_2_$CICE, breaks = 4) # cut -> Función para crear categorías, con el 4 se crean
unique(grupos)
```

```
## [1] (6.72,12.5]  (0.912,6.72] (12.5,18.3]  (18.3,24.1]
## Levels: (0.912,6.72] (6.72,12.5] (12.5,18.3] (18.3,24.1]
```

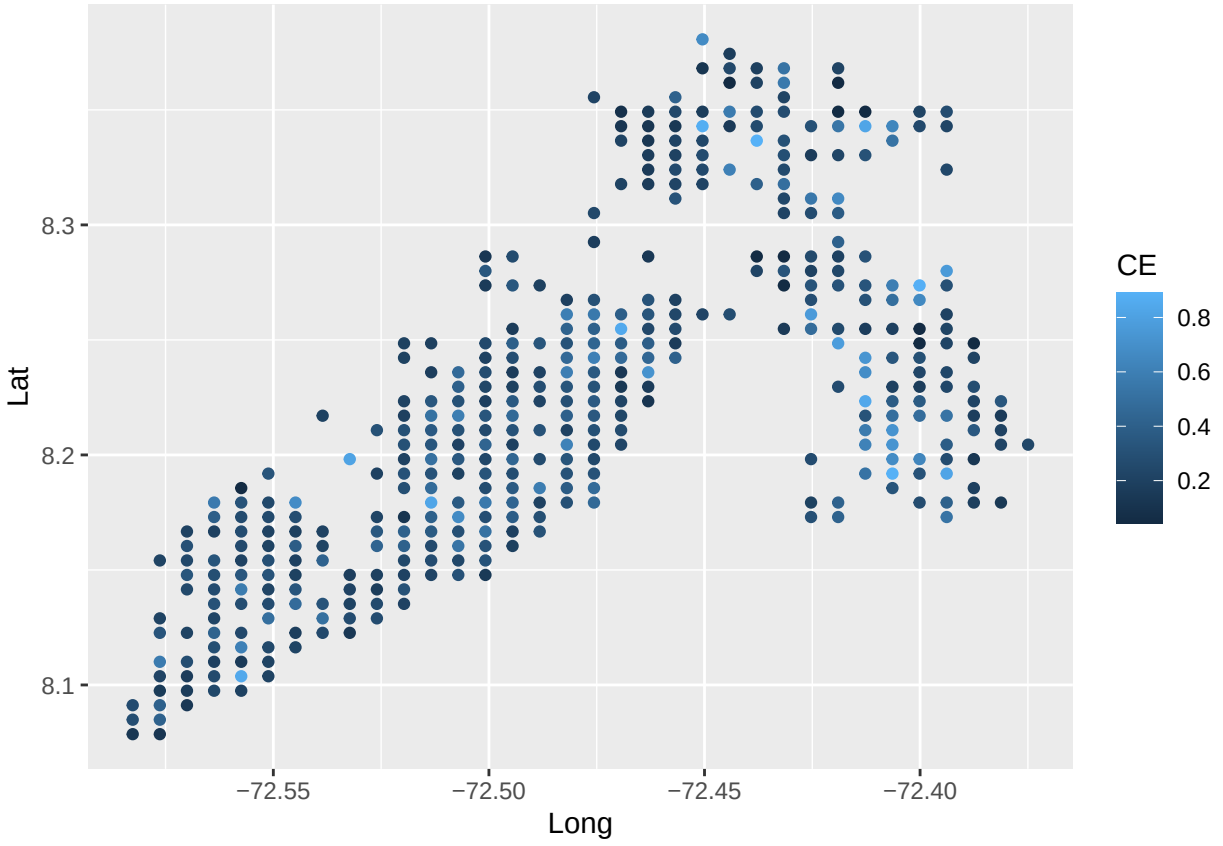
```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = CICE)) +#Aesthetic, puede usarse fill=color
  geom_point() #Geom point genera los puntos
```




```
CE =  XPABLO_2_$CE
grupos <- cut(x = XPABLO_2_$CE, breaks = 4) # cut -> Función para crear categorías, con el 4 se crean 4
unique(grupos)
```

```
## [1] (0.0446,0.257] (0.257,0.468]  (0.679,0.891]  (0.468,0.679]
## Levels: (0.0446,0.257] (0.257,0.468] (0.468,0.679] (0.679,0.891]
```

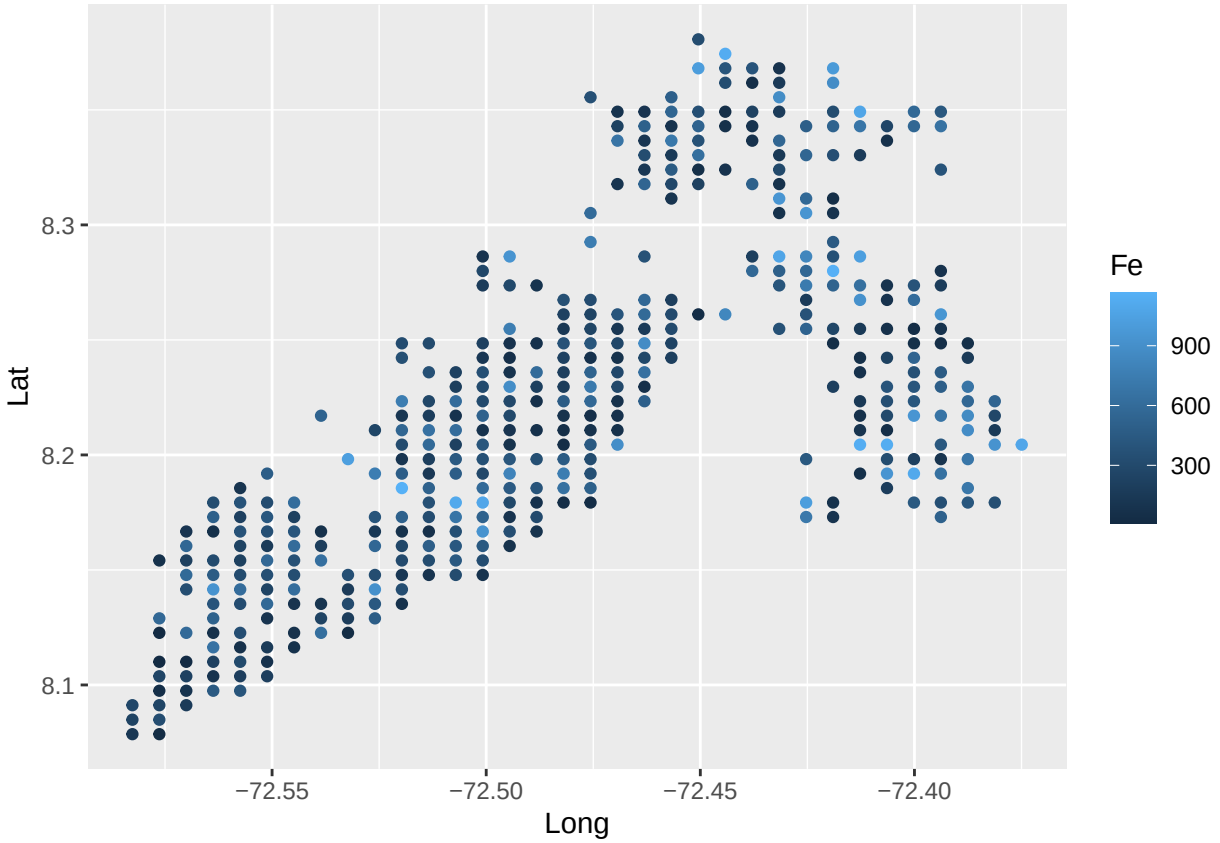
```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = CE)) +#Aesthetic, puede usarse fill=color
  geom_point() #Geom point genera los puntos
```




```
Fe =  XPABLO_2_$Fe
grupos <- cut(x = XPABLO_2_$Ca, breaks = 4) # cut -> Función para crear categorías, con el 4 se crean 4
unique(grupos)
```

```
## [1] (5.49,10.2]  (0.722,5.49] (10.2,15]    (15,19.7]
## Levels: (0.722,5.49] (5.49,10.2] (10.2,15] (15,19.7]
```

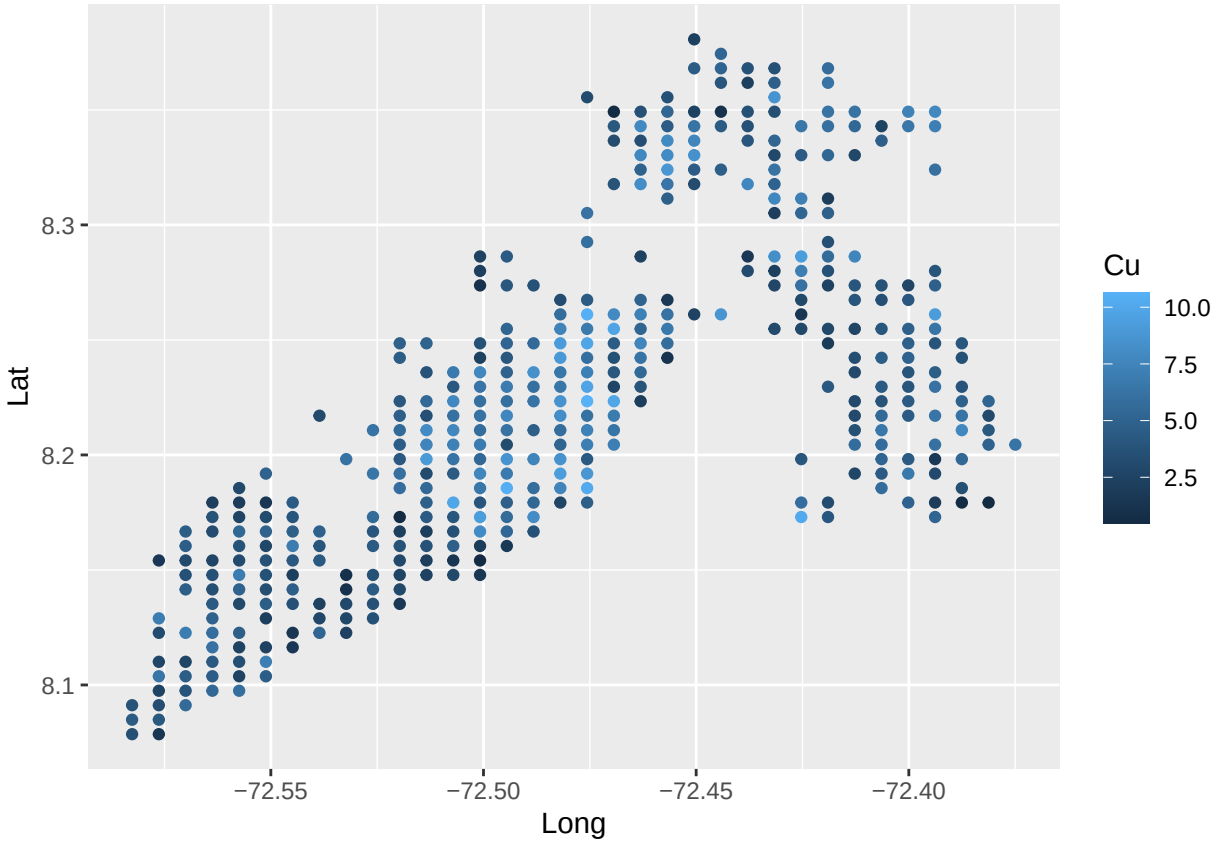
```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = Fe)) +#Aesthetic, puede usarse fill=color
  geom_point() #Geom point genera los puntos
```




```
Cu =  XPABLO_2_$Cu
grupos <- cut(x = XPABLO_2_$Cu, breaks = 4) # cut -> Función para crear categorías, con el 4 se crean 4
unique(grupos)
```

```
## [1] (3.03,5.56] (0.49,3.03] (5.56,8.1]  (8.1,10.6]
## Levels: (0.49,3.03] (3.03,5.56] (5.56,8.1] (8.1,10.6]
```

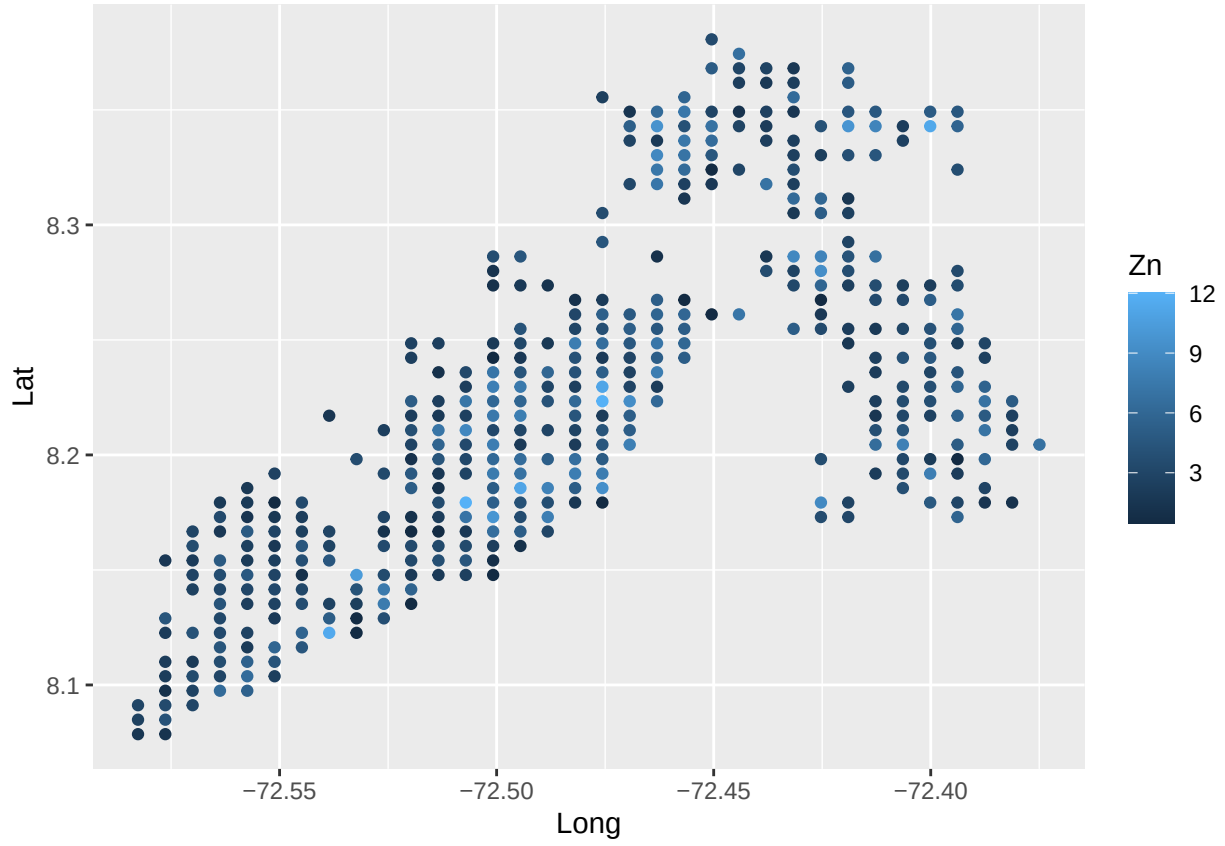
```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = Cu)) +#Aesthetic, puede usarse fill=color
  geom_point() #Geom point genera los puntos
```




```
Zn =  XPABLO_2_$Zn
grupos <- cut(x = XPABLO_2_$Zn, breaks = 4) # cut -> Función para crear categorías, con el 4 se crean 4
unique(grupos)
```

```
## [1] (0.488,3.38] (3.38,6.25]  (6.25,9.13]  (9.13,12]
## Levels: (0.488,3.38] (3.38,6.25] (6.25,9.13] (9.13,12]
```

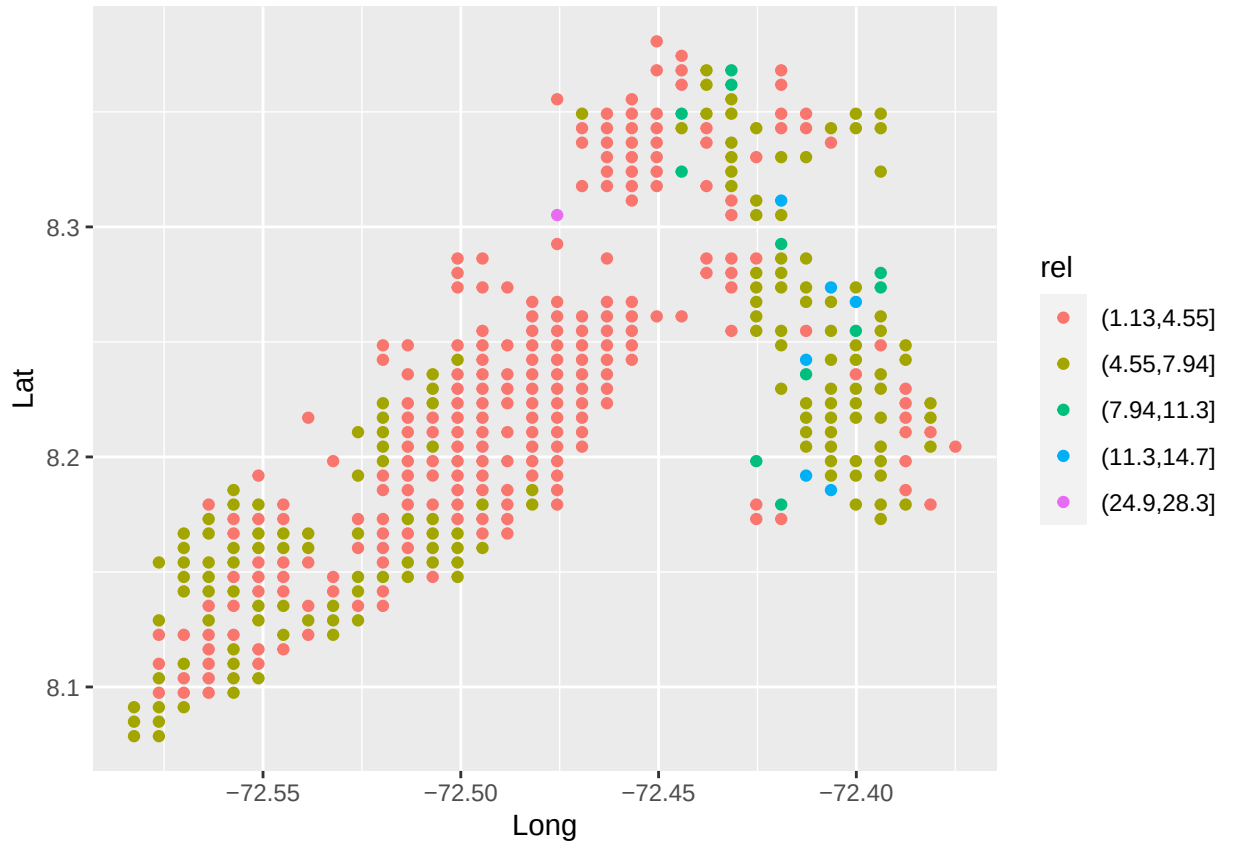
```
ggplot(XPABLO_2_, aes(x = Long, y = Lat, color = Zn)) +#Aesthetic, puede usarse fill=color
  geom_point() #Geom point genera los puntos
```




```
Ca_Mg = XPABLO_2_$Ca/XPABLO_2_$Mg
rel = cut(Ca_Mg, breaks = 8)
unique(rel)
```

```
## [1] (4.55,7.94] (1.13,4.55] (7.94,11.3] (11.3,14.7] (24.9,28.3]
## 8 Levels: (1.13,4.55] (4.55,7.94] (7.94,11.3] (11.3,14.7] ... (24.9,28.3]
```

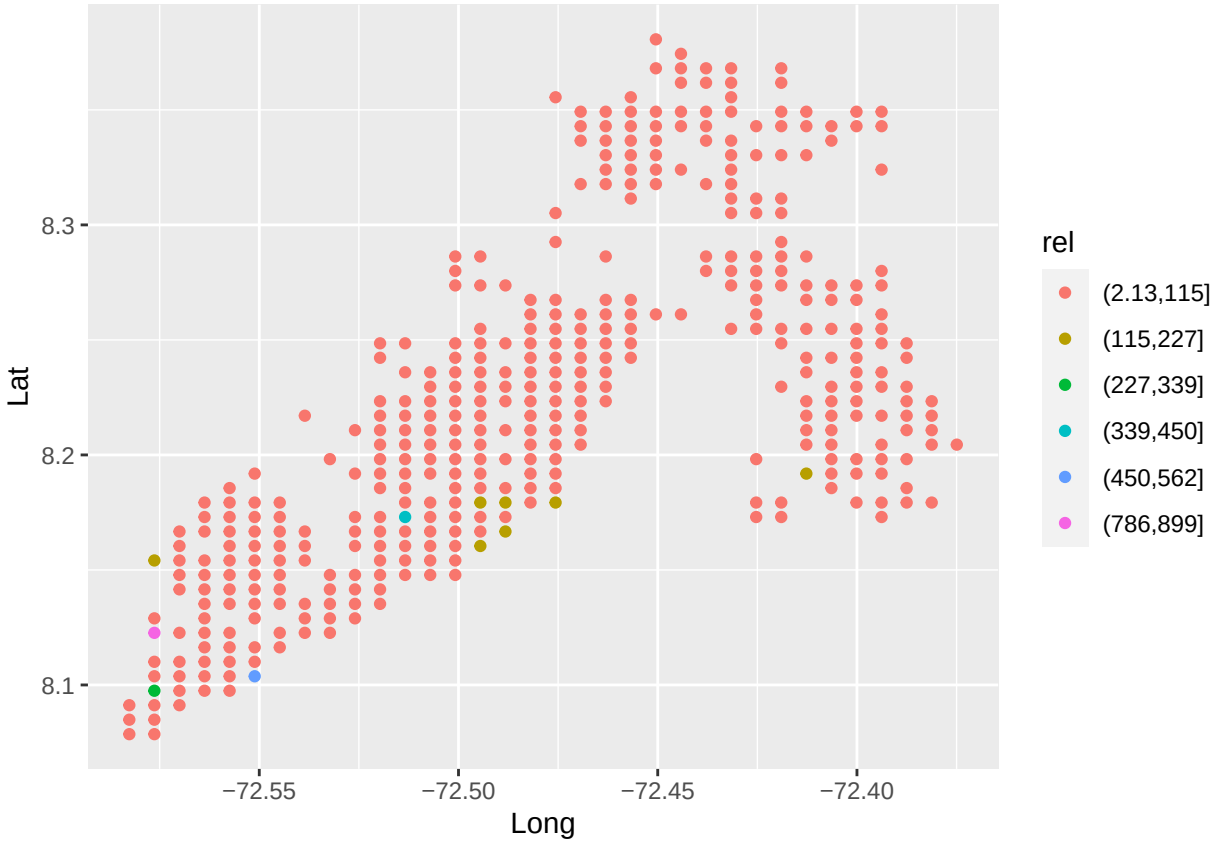
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
Ca_K = XPABLO_2_$Ca/XPABLO_2_$K
rel = cut(Ca_K, breaks = 8)
unique(rel)
```

```
## [1] (2.13,115] (227,339]  (450,562]  (786,899]  (115,227]  (339,450]
## 8 Levels: (2.13,115] (115,227] (227,339] (339,450] (450,562] ... (786,899]
```

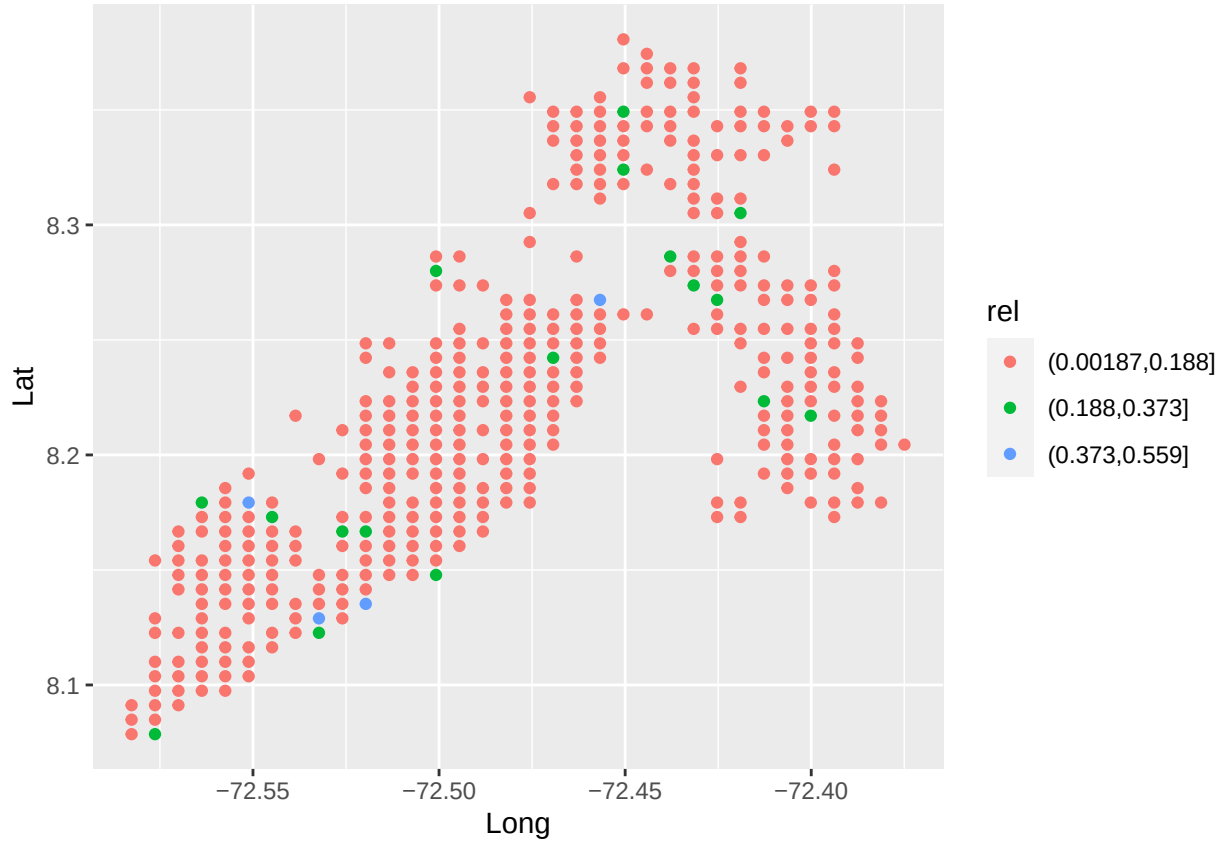
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
Ca_Zn = XPABLO_2_$K/XPABLO_2_$Zn
rel = cut(Ca_Zn, breaks = 3)
unique(rel)
```

```
## [1] (0.00187,0.188] (0.188,0.373]   (0.373,0.559]
## Levels: (0.00187,0.188] (0.188,0.373] (0.373,0.559]
```

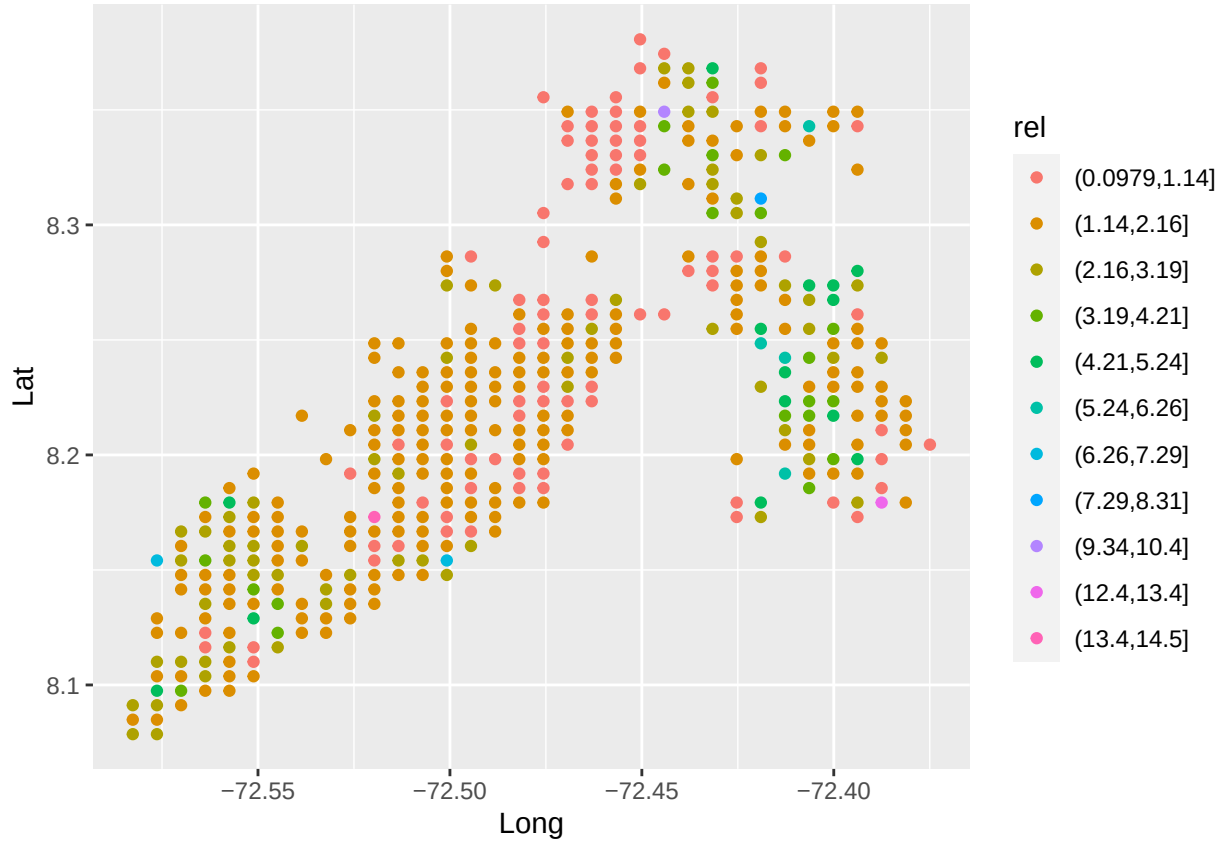
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
Ca_Cu = XPABLO_2_$Ca/XPABLO_2_$Cu
rel = cut(Ca_Cu, breaks = 14)
unique(rel)
```

```
##  [1] (2.16,3.19]   (1.14,2.16]   (4.21,5.24]   (3.19,4.21]   (0.0979,1.14]
##  [6] (6.26,7.29]   (13.4,14.5]   (12.4,13.4]   (5.24,6.26]   (7.29,8.31]
## [11] (9.34,10.4]
## 14 Levels: (0.0979,1.14] (1.14,2.16] (2.16,3.19] (3.19,4.21] ... (13.4,14.5]
```

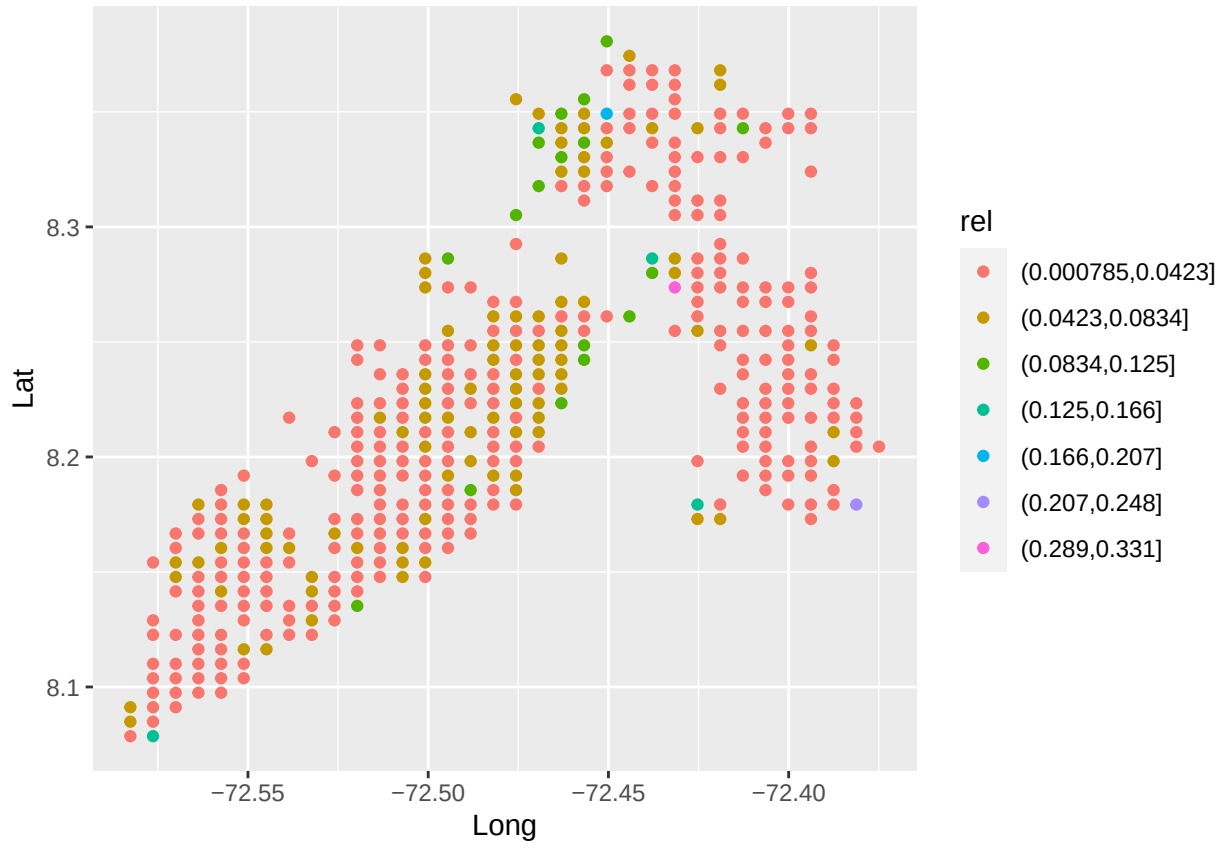
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
K_Ca = XPABLO_2_$K/XPABLO_2_$Ca
rel = cut(K_Ca, breaks = 8)
unique(rel)
```

```
## [1] (0.000785,0.0423] (0.125,0.166]     (0.0423,0.0834]   (0.0834,0.125]
## [5] (0.207,0.248]     (0.289,0.331]     (0.166,0.207]
## 8 Levels: (0.000785,0.0423] (0.0423,0.0834] (0.0834,0.125] ... (0.289,0.331]
```

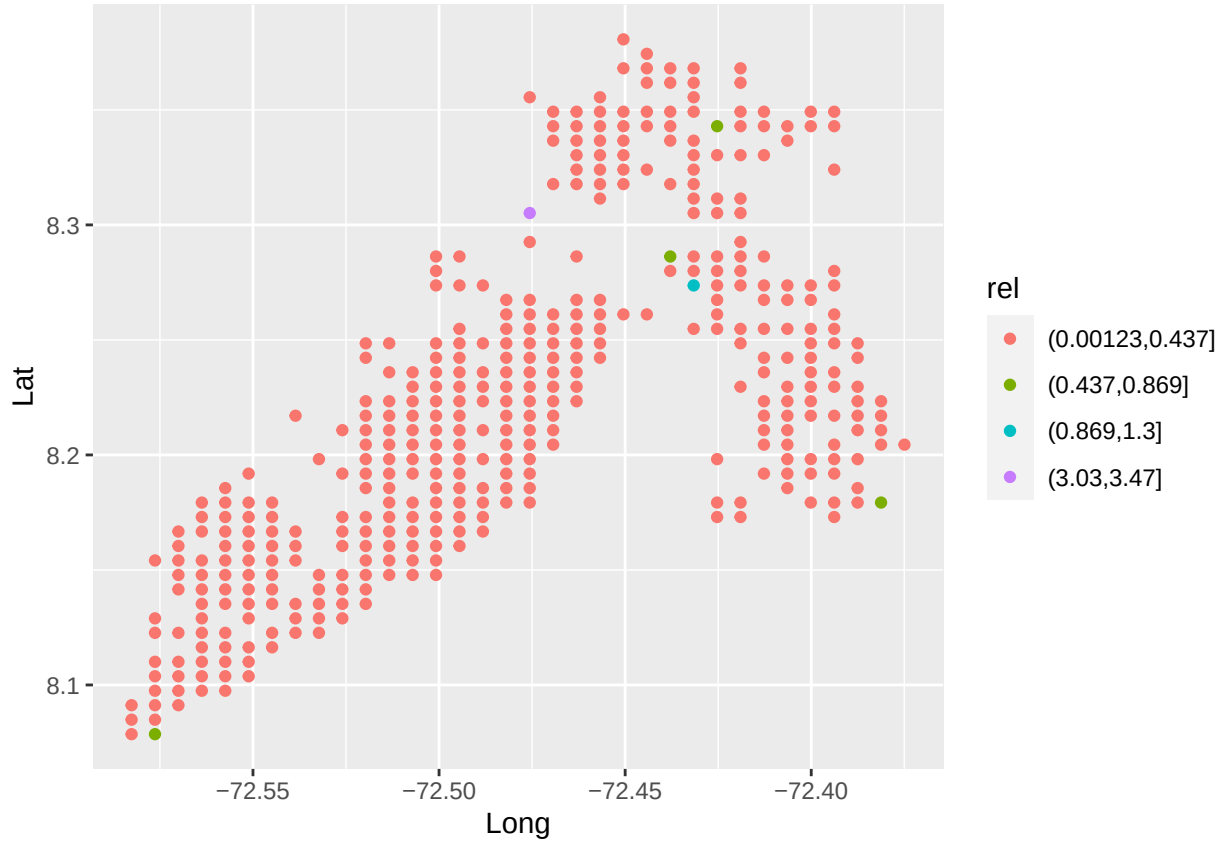
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
K_Mg = XPABLO_2_$K/XPABLO_2_$Mg
rel = cut(K_Mg, breaks = 8)
unique(rel)
```

```
## [1] (0.00123,0.437] (0.437,0.869]   (0.869,1.3]     (3.03,3.47]
## 8 Levels: (0.00123,0.437] (0.437,0.869] (0.869,1.3] (1.3,1.73] ... (3.03,3.47]
```

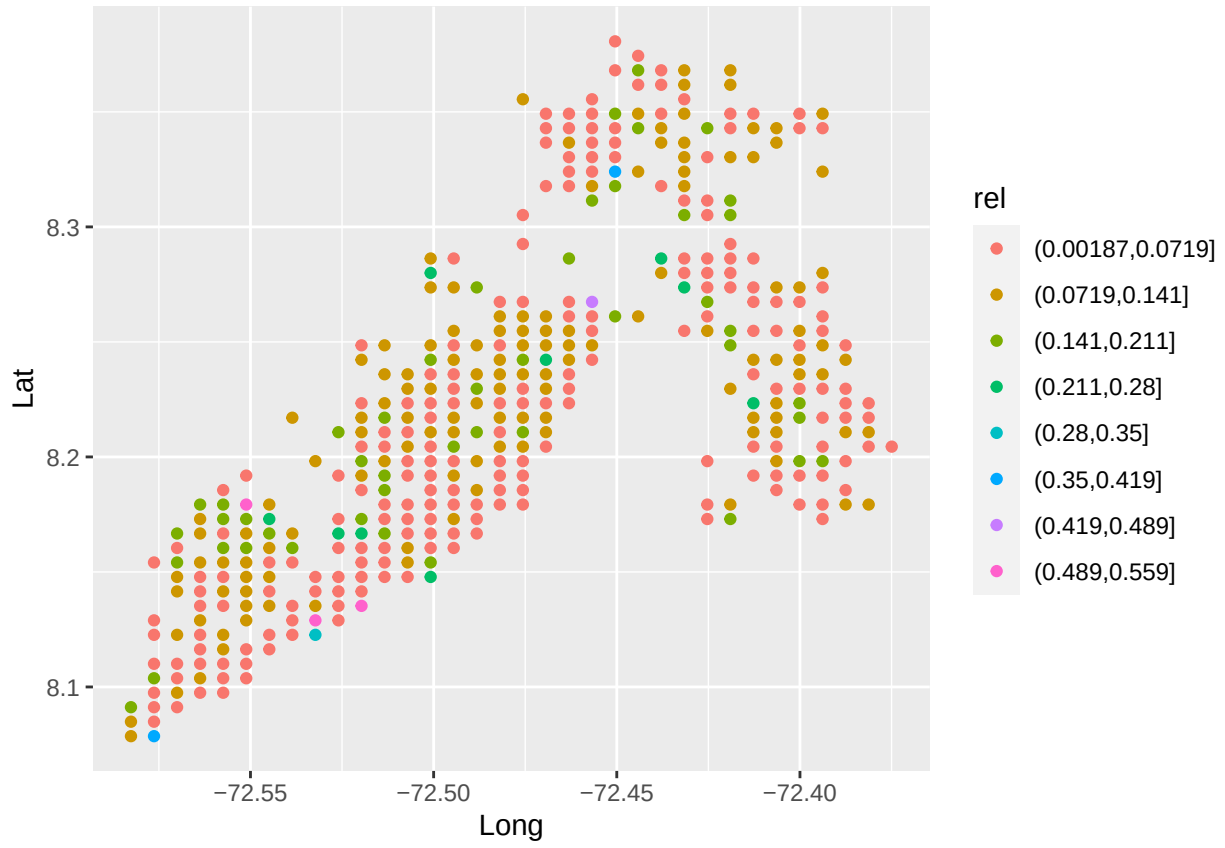
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
K_Zn = XPABLO_2_$K/XPABLO_2_$Zn
rel = cut(K_Zn, breaks = 8)
unique(rel)
```

```
## [1] (0.0719,0.141]   (0.35,0.419]     (0.00187,0.0719] (0.141,0.211]
## [5] (0.28,0.35]      (0.489,0.559]    (0.211,0.28]     (0.419,0.489]
## 8 Levels: (0.00187,0.0719] (0.0719,0.141] (0.141,0.211] ... (0.489,0.559]
```

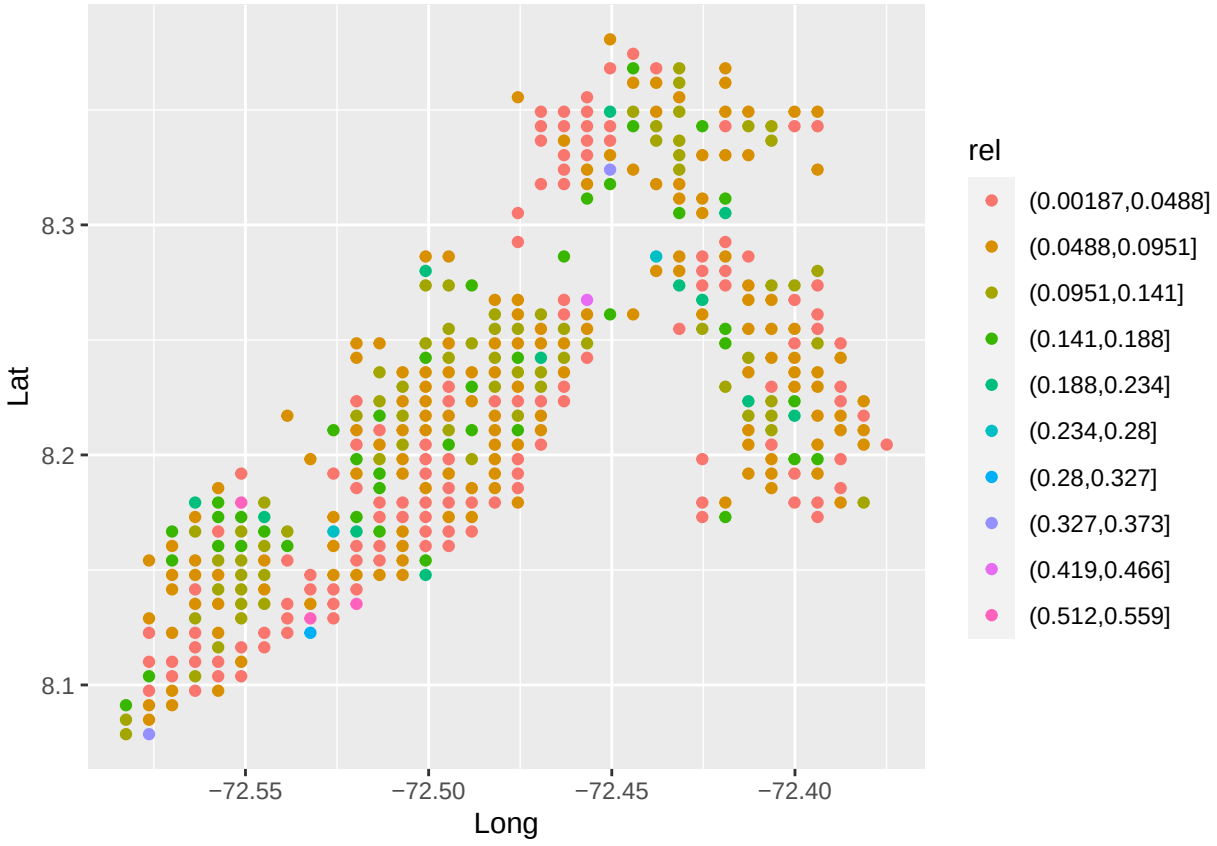
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
K_Cu = XPABLO_2_$K/XPABLO_2_$Cu
rel = cut(K_Zn, breaks = 12)
unique(rel)
```

```
##  [1] (0.0951,0.141]   (0.327,0.373]    (0.0488,0.0951]  (0.141,0.188]
##  [5] (0.00187,0.0488] (0.28,0.327]     (0.512,0.559]    (0.188,0.234]
##  [9] (0.234,0.28]     (0.419,0.466]
## 12 Levels: (0.00187,0.0488] (0.0488,0.0951] (0.0951,0.141] ... (0.512,0.559]
```

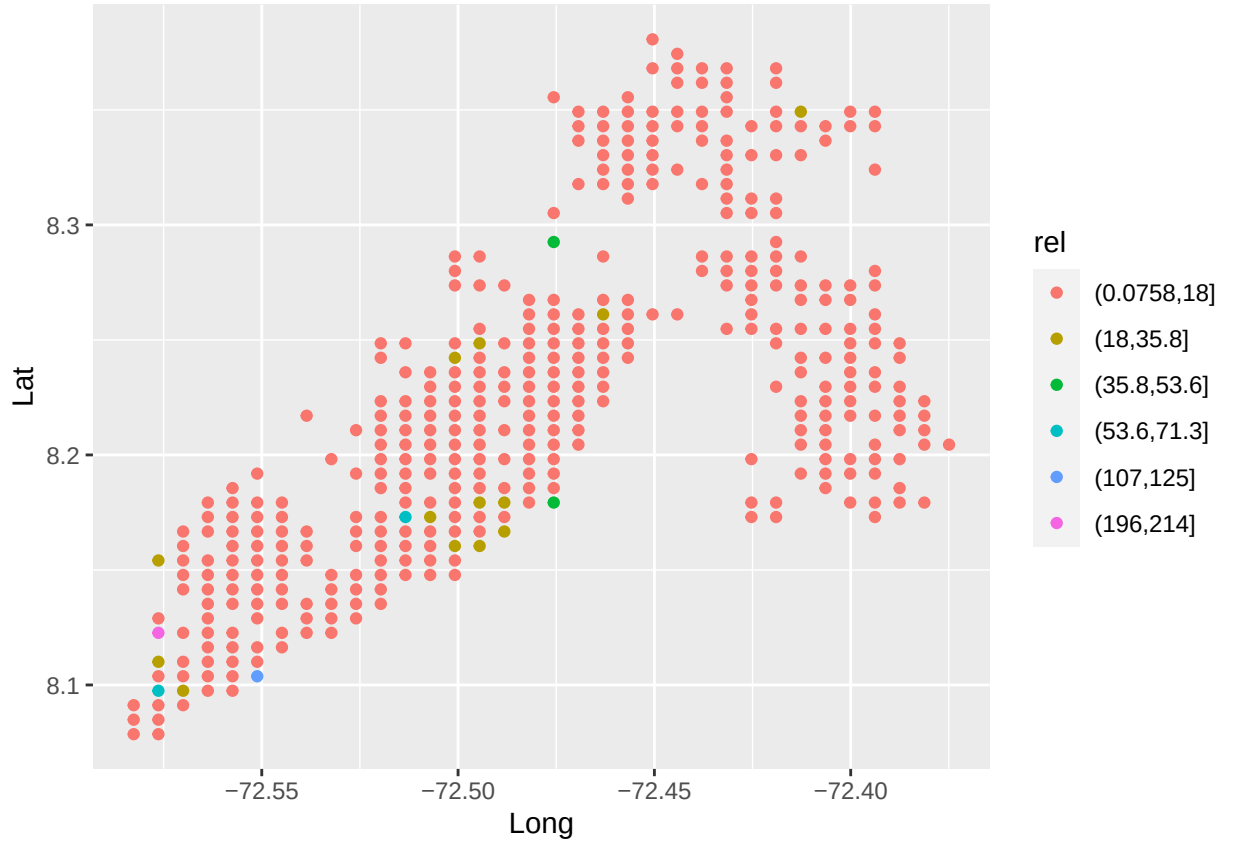
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
Mg_K = XPABLO_2_$Mg/XPABLO_2_$K
rel = cut(Mg_K, breaks = 12)
unique(rel)
```

```
## [1] (0.0758,18] (53.6,71.3] (18,35.8]   (107,125]   (196,214]   (35.8,53.6]
## 12 Levels: (0.0758,18] (18,35.8] (35.8,53.6] (53.6,71.3] ... (196,214]
```

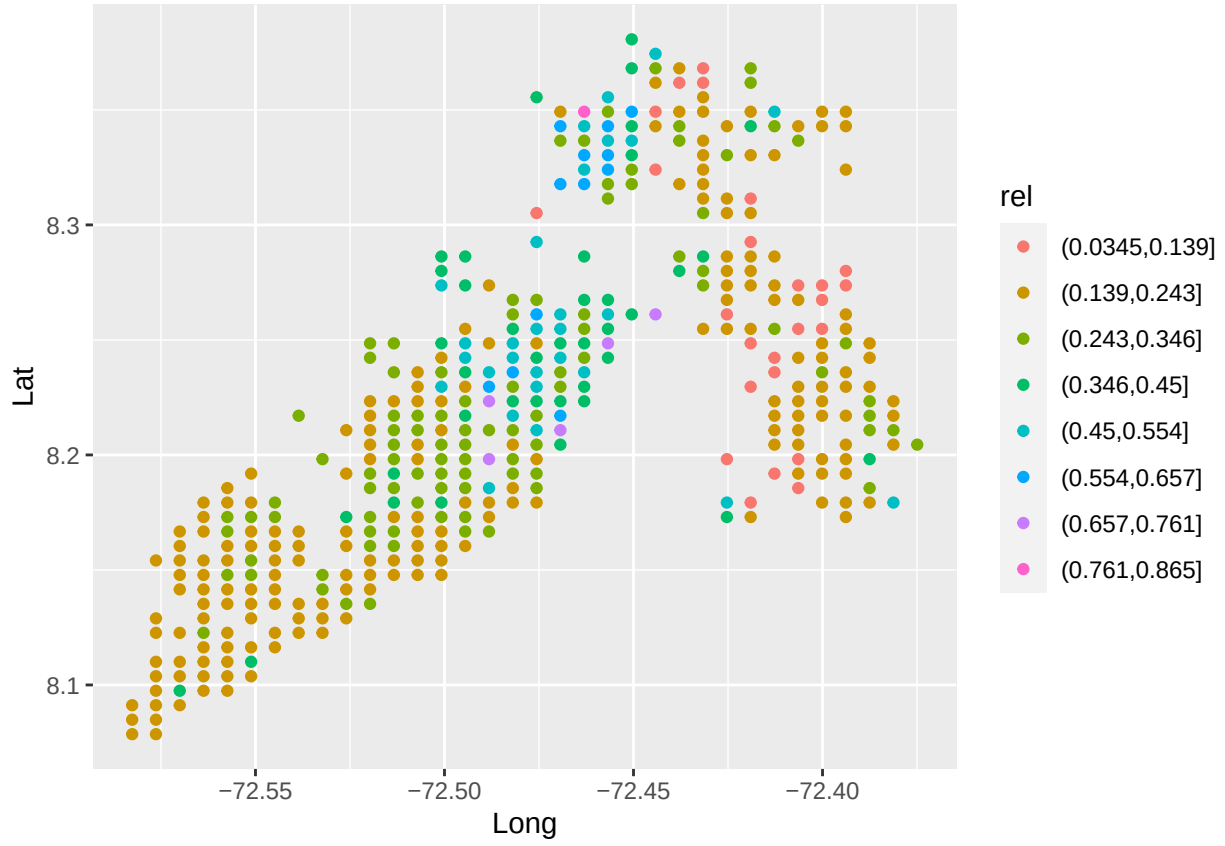
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
Mg_Ca = XPABLO_2_$Mg/XPABLO_2_$Ca
rel = cut(Mg_Ca, breaks = 8)
unique(rel)
```

```
## [1] (0.139,0.243]  (0.346,0.45]   (0.243,0.346]  (0.45,0.554]   (0.0345,0.139]
## [6] (0.657,0.761]  (0.554,0.657]  (0.761,0.865]
## 8 Levels: (0.0345,0.139] (0.139,0.243] (0.243,0.346] ... (0.761,0.865]
```

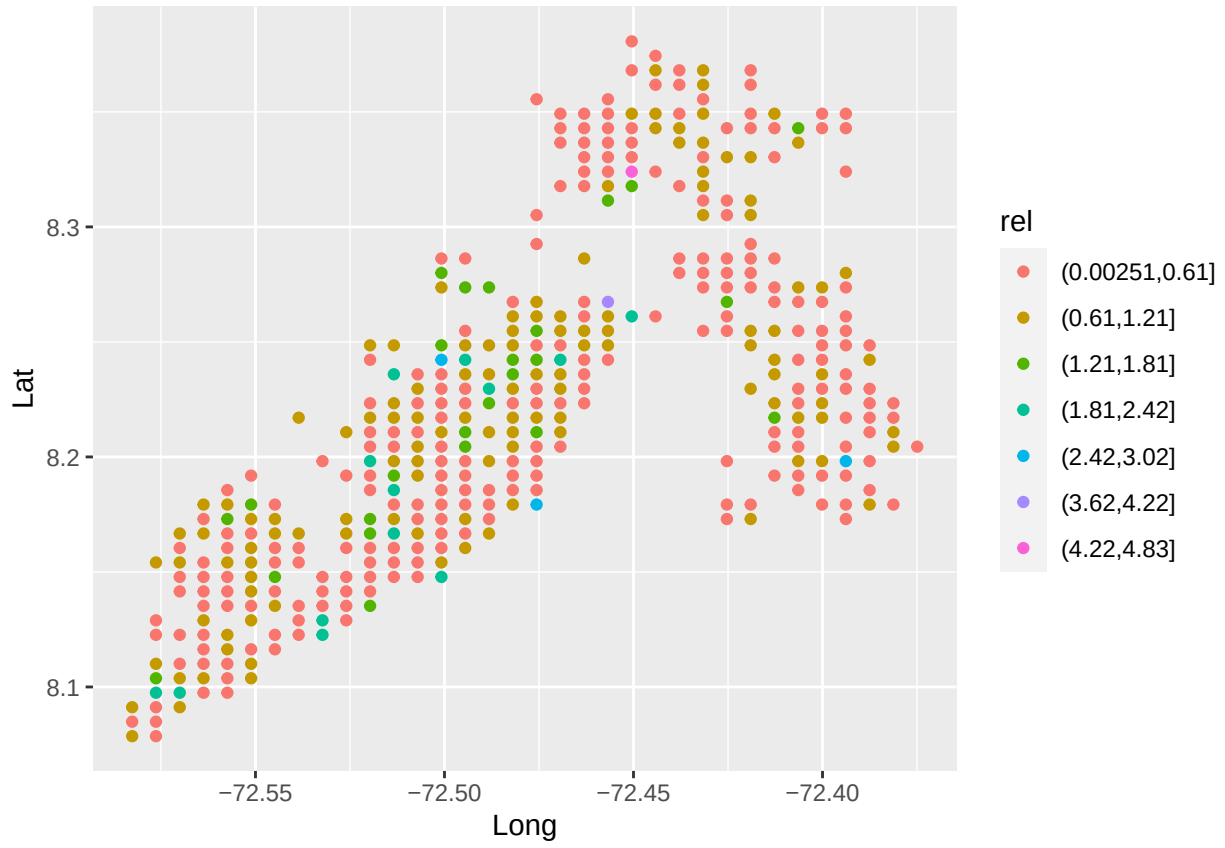
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
Mg_Zn = XPABLO_2_$Mg/XPABLO_2_$Zn
rel = cut(Mg_Zn, breaks = 8)
unique(rel)
```

```
## [1] (0.61,1.21]    (0.00251,0.61] (1.81,2.42]    (1.21,1.81]    (2.42,3.02]
## [6] (3.62,4.22]    (4.22,4.83]
## 8 Levels: (0.00251,0.61] (0.61,1.21] (1.21,1.81] (1.81,2.42] ... (4.22,4.83]
```

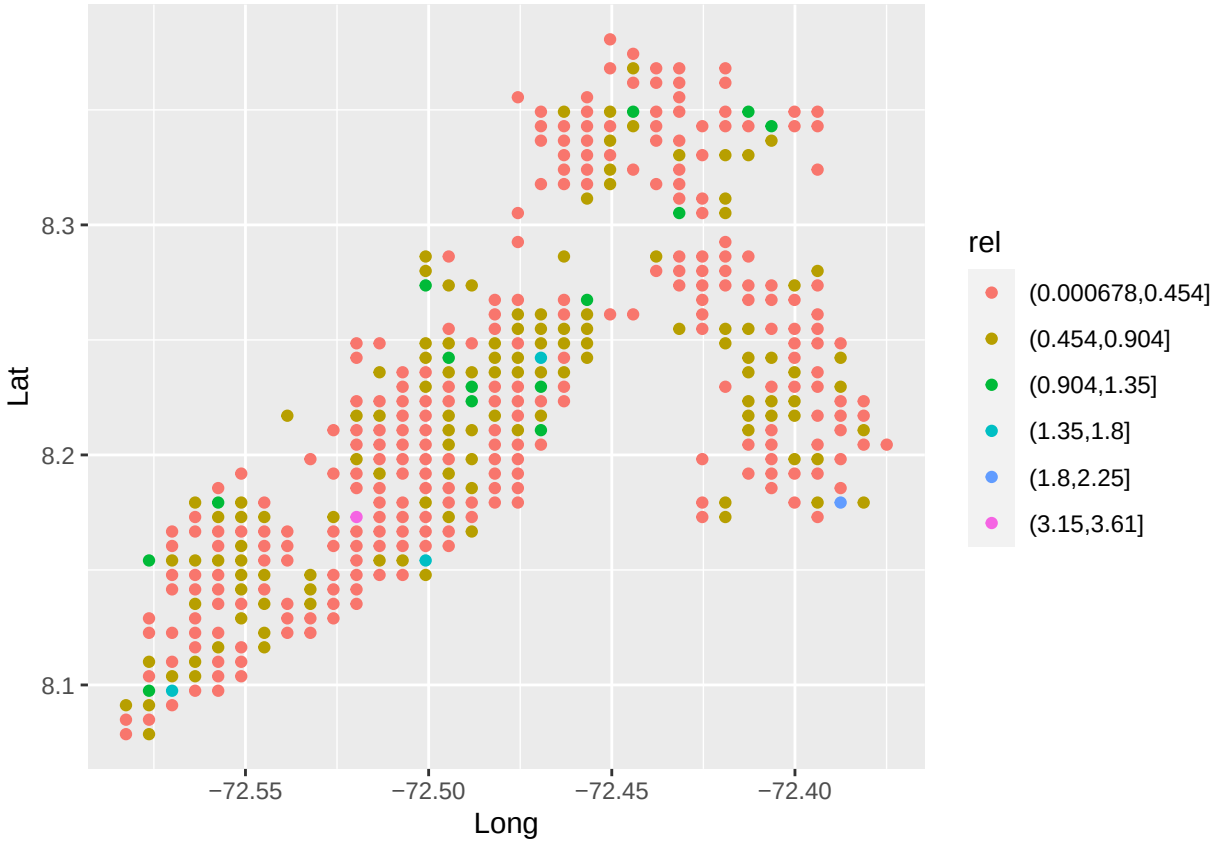
```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```




```
Mg_Cu = XPABLO_2_$Mg/XPABLO_2_$Cu
rel = cut(Mg_Cu, breaks = 8)
unique(rel)
```

```
## [1] (0.000678,0.454] (0.454,0.904]    (0.904,1.35]     (1.35,1.8]
## [5] (3.15,3.61]      (1.8,2.25]
## 8 Levels: (0.000678,0.454] (0.454,0.904] (0.904,1.35] ... (3.15,3.61]
```

```
ggplot(XPABLO_2_, aes(Long, Lat, color = rel))+
  geom_point()
```



Mapas usando los quartiles

```
# la funcion quantile le da del conjunto de datos ual es la proabilidad de asignarle a cada rango dentr
K_Ca = XPABLO_2_$K/XPABLO_2_$Ca
nuevos_grupos <- quantile(x = K_Ca, probs = c(0.25, 0.5, 0.75))
nuevos_grupos
```

```
##        25%        50%        75%
## 0.02203894 0.03148457 0.04546647
```

```
q_1 = nuevos_grupos[1];q_1
```

```
##        25%
## 0.02203894
```

```
q_2 = nuevos_grupos[2];q_2
```

```
##        50%
## 0.03148457
```

```
q_3 = nuevos_grupos[3];q_3
```

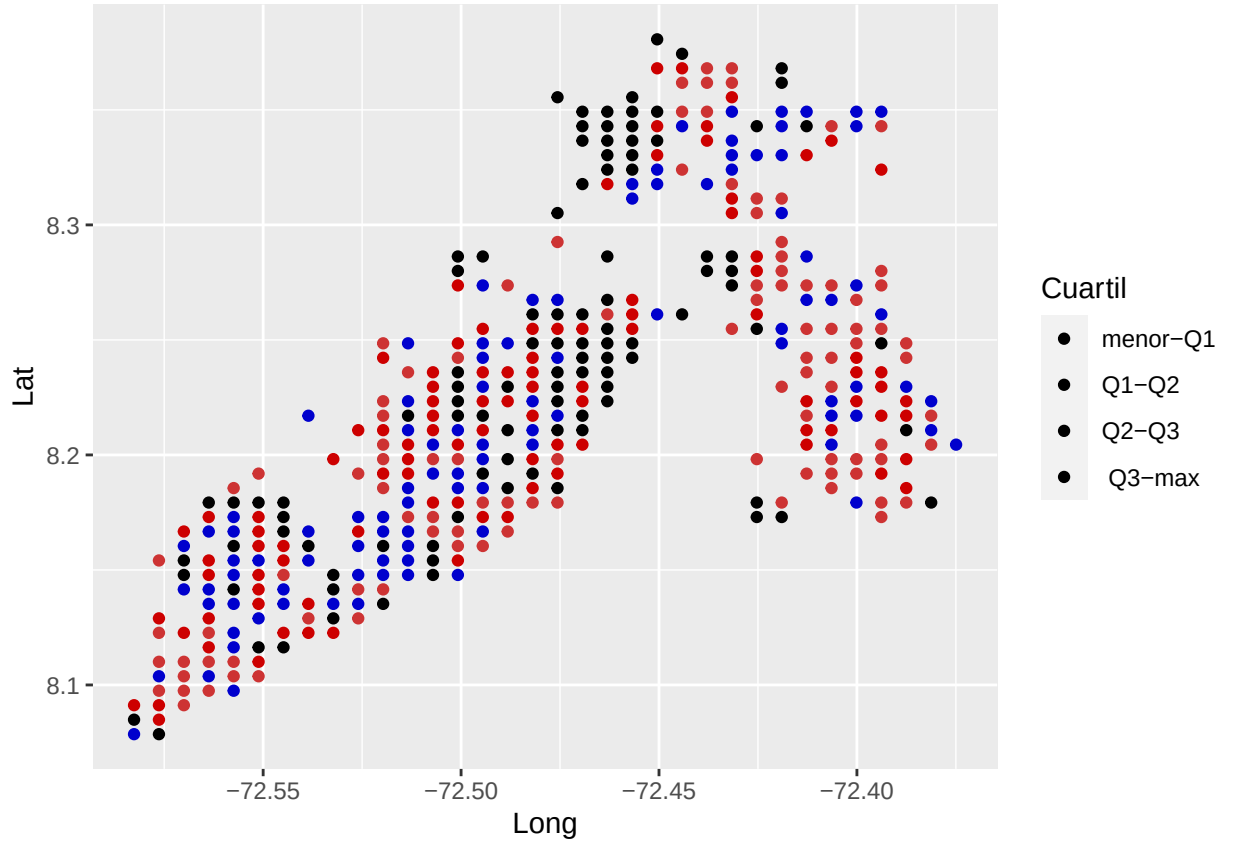
```
##        75%
## 0.04546647
```

```r
data1 <- subset(K_Ca, K_Ca < q_1);data1
```

```
##   [1] 0.017770783 0.003631882 0.016899106 0.016672485 0.014244369 0.017829779
##   [7] 0.001866777 0.011364413 0.011759057 0.017456972 0.018758963 0.001113752
##  [13] 0.020051633 0.019308300 0.011607138 0.016786280 0.014469950 0.016641156
##  [19] 0.008126308 0.010006420 0.006820446 0.013054225 0.013310712 0.007865238
##  [25] 0.002893776 0.009259782 0.021563955 0.008578242 0.007230433 0.014098125
##  [31] 0.004585193 0.018884785 0.020176754 0.019447441 0.018961197 0.018919623
##  [37] 0.012393473 0.018652539 0.020939762 0.008483663 0.018623551 0.016933475
##  [43] 0.018558460 0.019209688 0.018346348 0.021209367 0.016857029 0.018011475
##  [49] 0.021734712 0.010123586 0.011344915 0.019762726 0.017377756 0.020121604
##  [55] 0.017022637 0.018410169 0.020647505 0.019319348 0.013830512 0.016808097
##  [61] 0.010858379 0.012090250 0.014476432 0.018579109 0.020468857 0.018779099
##  [67] 0.016364071 0.013354781 0.021975858 0.018850442 0.014164048 0.020929304
##  [73] 0.017366913 0.018011114 0.020818265 0.011874328 0.020987011 0.021841715
##  [79] 0.021989167 0.015994396 0.016362357 0.015278017 0.017173177 0.018130008
##  [85] 0.019748126 0.013541530 0.013414413 0.019172781 0.021886637 0.015045369
##  [91] 0.016363858 0.014954292 0.020821714 0.011759917 0.013618601 0.013935915
##  [97] 0.019987578 0.017372749 0.014763464 0.014087817 0.013602341
```

```r
data2 <- subset(K_Ca, K_Ca < q_2 & K_Ca > q_1)#;data2
data3 <- subset(K_Ca, K_Ca < q_3 & K_Ca > q_2)#;data3
data4 <- subset(K_Ca, K_Ca > q_3)#;data4
```

```r
colors <- ifelse(K_Ca %in% data1, 'brown3',
                 ifelse(K_Ca %in% data2, 'blue3',
                        ifelse(K_Ca %in% data3, 'red3','black')))
ggplot(XPABLO_2_, aes(Long, Lat, fill = colors ))+
  geom_point(color = colors)+
  scale_fill_discrete(name = 'Cuartil', labels = c('menor-Q1', 'Q1-Q2', 'Q2-Q3', ' Q3-max'))
```




```
K_Zn = XPABLO_2_$K/XPABLO_2_$Zn
nuevos_grupos <- quantile(x = K_Zn, probs = c(0.25, 0.5, 0.75))
nuevos_grupos
```

```
##        25%        50%        75%
## 0.04125529 0.06982484 0.11083651
```

```
q_1 = nuevos_grupos[1];q_1
```

```
##        25%
## 0.04125529
```

```
q_2 = nuevos_grupos[2];q_2
```

```
##        50%
## 0.06982484
```

```
q_3 = nuevos_grupos[3];q_3
```

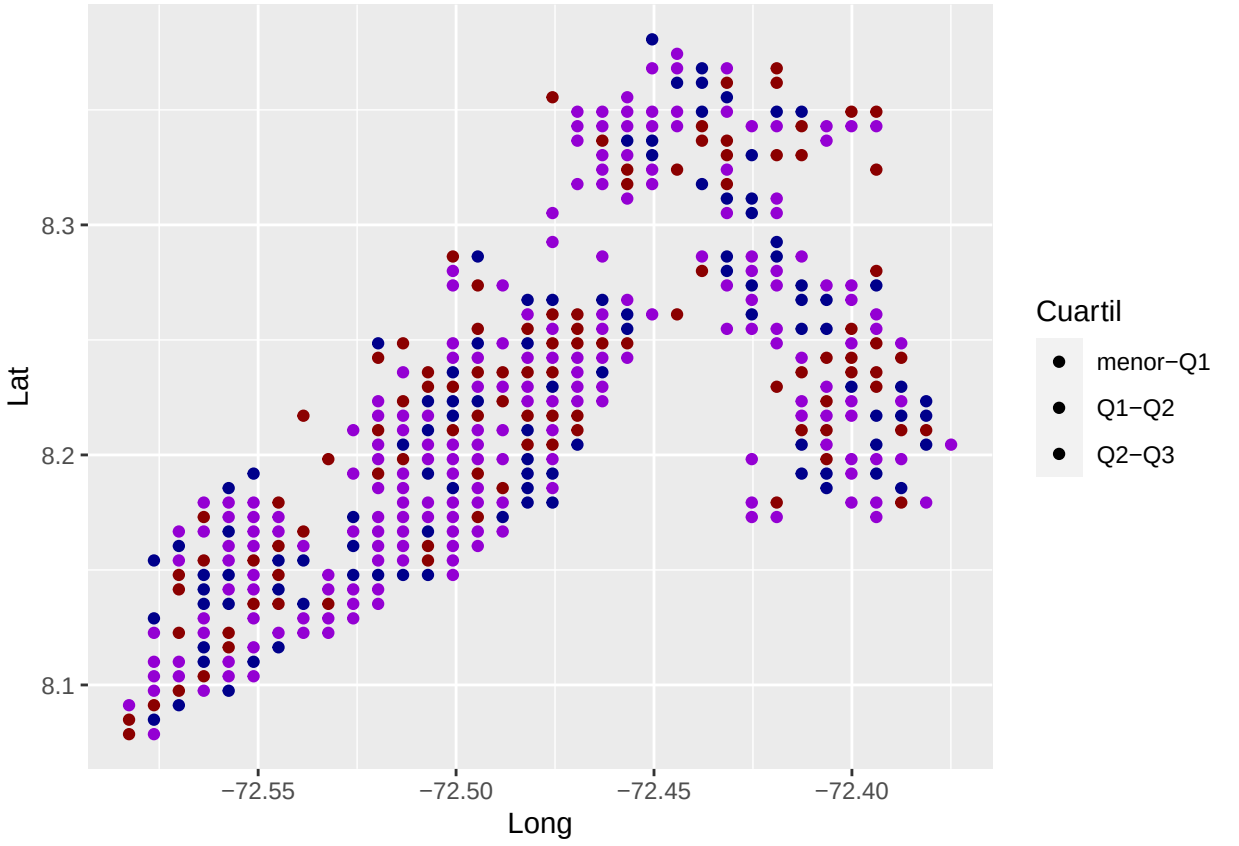
```
##       75%
## 0.1108365
```

```
data1 <- subset(K_Zn, K_Zn < q_1);data1
```

```
##   [1] 0.037336262 0.020910945 0.037134616 0.012561006 0.005814360 0.031475486
##   [7] 0.041249851 0.020557693 0.023798248 0.002427145 0.038231454 0.029146296
##  [13] 0.020990226 0.014729267 0.015501959 0.019338210 0.026605270 0.018770839
##  [19] 0.012822843 0.011208656 0.020185677 0.033580463 0.036241239 0.025269447
##  [25] 0.010681958 0.024669739 0.014455969 0.021196250 0.018528194 0.009908507
##  [31] 0.006204092 0.034090747 0.021854999 0.017946531 0.037736815 0.017817419
##  [37] 0.037936061 0.014883794 0.018880874 0.014949760 0.032507499 0.032913957
##  [43] 0.028243727 0.017366351 0.039703685 0.040824880 0.033776954 0.022583659
##  [49] 0.020448681 0.040942086 0.023059538 0.038706260 0.027876577 0.026729307
##  [55] 0.038701684 0.033567157 0.021987041 0.033972836 0.033589346 0.039393115
##  [61] 0.024726167 0.034845736 0.017506623 0.033014139 0.034712188 0.035967285
##  [67] 0.030378426 0.036823004 0.029907028 0.021998800 0.028522697 0.022691367
##  [73] 0.006259560 0.032153143 0.040653739 0.031054523 0.040394578 0.021903082
##  [79] 0.034884814 0.026237749 0.007254283 0.025365059 0.032756205 0.020079197
##  [85] 0.019332088 0.020514911 0.031407064 0.040053517 0.033294988 0.024970657
##  [91] 0.032476832 0.034789794 0.017307014 0.026762218 0.012242863 0.038055835
##  [97] 0.034575605 0.017098813 0.024332370 0.034183061 0.040293682
```

```
data2 <- subset(K_Zn, K_Zn < q_2 & K_Zn > q_1)#;data2
data3 <- subset(K_Zn, K_Zn < q_3 & K_Zn > q_2)#;data3
data4 <- subset(K_Zn, K_Zn > q_3)#;data4
```

```
colors <- ifelse(K_Ca %in% data1, 'brown3',
                 ifelse(K_Zn %in% data2, 'blue4',
                        ifelse(K_Zn %in% data3, 'red4','darkviolet')))
ggplot(XPABLO_2_, aes(Long, Lat, fill = colors ))+
  geom_point(color = colors)+
  scale_fill_discrete(name = 'Cuartil', labels = c('menor-Q1', 'Q1-Q2', 'Q2-Q3', ' Q3-max'))
```



Tarea

#* Buscar que es driss(se usa en diagnostico de suelo de propiedades, fisicas, quimicas y biologicas en funcion de la relación Ca/Mg, K/) # 1 https://www.researchgate.net/figure/Plant-soil-nutrient-constraints-diagnosis-using-DRIS-indices_tbl4_233104806 # 2 https://www.researchgate.net/publication/328723237_El_Sistema_Integrado_de_Diagnostico_y_Recomendacion_DRIS_para_conocer_el_balance_nutricional_del_trigo # 3 https://bookdown.org/mateotabares7/drisa/DRIS.nb.html # 4 https://revistas.unal.edu.co/index.php/agrocol/article/download/20933/21836 # 5 https://revistas.uptc.edu.co/index.php/ciencia_agricultura/article/view/12933/11020 # 6 http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352008000500004 # 7 https://www.redalyc.org/pdf/573/57324446005.pdf # 8 http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0187-57792012000200139 # 9 http://ve.scielo.org/scielo.php?script=sci_arttext&pid=S1316-33612010000200005 # 10 https://www.intechopen.com/chapters/44395