appendix_covar_balancing.R

datalab

2023-06-16

```
#############################################################
############   Additional Balancing ##############
################      October 10, 2018     ##################
########  Rerun: December 16, 2022 ######

rm(list=ls())
library(coefplot)
library(stargazer)
library(ggplot2)
library(foreign)

#################################################
############ Multiplot Function #################
######   Source: http://www.cookbook-r.com/Graphs/Multiple_graphs_on_one_page_(ggplot2)/



multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {
  library(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

  numPlots = length(plots)

  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {
    # Make the panel
    # ncol: Number of columns of plots
    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                     ncol = cols, nrow = ceiling(numPlots/cols))
  }

  if (numPlots==1) {
    print(plots[[1]])

  } else {
    # Set up the page
    grid.newpage()
    pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))
```

```
    # Make each plot, in the correct location
    for (i in 1:numPlots) {
      # Get the i,j matrix positions of the regions that contain this subplot
      matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

      print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
                                      layout.pos.col = matchidx$col))
    }
  }
}
############################################################


##########################################
##### Create  Bandwidths ################
geo2001=read.csv("~/Dropbox/Personal Research 2017/replications/geocensus2001_nov16.csv")

#Distance to Mysore-Bombay Border
rd10.mb=geo2001[which(geo2001$NEAR_DIST_border1<10000),] #20 km
rd25.mb=geo2001[which(geo2001$NEAR_DIST_border1<25000),] #50 km
rd50.mb=geo2001[which(geo2001$NEAR_DIST_border1<50000),] #100 km
rd100.mb=geo2001[which(geo2001$NEAR_DIST_border1<100000),] #200 km

table(rd10.mb$border1)
```

```
##
##   0   1
## 505 470
```

```
#Distance to Hyderabad-Bombay Border
rd10.hb=geo2001[which(geo2001$NEAR_DIST_border2<10000),] #20 km
rd25.hb=geo2001[which(geo2001$NEAR_DIST_border2<25000),] #50 km
rd50.hb=geo2001[which(geo2001$NEAR_DIST_border2<50000),] #100 km
rd100.hb=geo2001[which(geo2001$NEAR_DIST_border2<100000),] #200 km
table(rd10.hb$border2)
```

```
##
##   0   1
## 404 424
```

```
############################################################
#### Plots of space for the balance and unbalance ####
##########     using long and lat    ##################

######borders######
mys=read.dbf("mysborderpts.dbf")
names(mys)
```

```
##   [1] "OBJECTID"   "VILLAGE_ID" "NAME"       "SUB_DISTRI" "DISTRICT"   "STATE_UT"
##   [7] "C_CODE01"   "LEVEL"      "TOT_NM_HH"  "TOT_POP"    "M_POP"      "F_POP"
##  [13] "TOT_L6"     "M_L6"       "F_L6"       "TOT_SC"     "M_SC"       "F_SC"
##  [19] "TOT_ST"     "M_ST"       "F_ST"       "TOT_LIT"    "M_LIT"      "F_LIT"
##  [25] "TOT_ILLT"   "M_ILLT"     "F_ILLT"     "TOT_W"      "M_W"        "F_W"
##  [31] "TOT_MNW"    "M_MNW"      "F_MNW"      "TOT_CULT"   "M_CULT"     "F_CULT"
##  [37] "TOT_AGLB"   "M_AGLB"     "F_AGLB"     "TOT_MFHH"   "M_MFHH"     "F_MFHH"
##  [43] "TOT_OTH_W"  "M_OTH_W"    "F_OTH_W"    "TOT_MRW"    "M_MRW"      "F_MRW"
```

```
##  [49] "T_MRG_CULT" "M_MRG_CULT" "F_MRG_CULT" "T_MRG_AGLB" "M_MRG_AGLB" "F_MRG_AGLB"
##  [55] "T_MRG_HH"   "M_MRG_HH"   "F_MRG_HH"   "T_MRG_OTH"  "M_MRG_OTH"  "F_MRG_OTH"
##  [61] "TOT_NNW"    "M_NNW"      "F_NNW"      "village__1" "name_2"     "sub_dist_1"
##  [67] "district_2" "state_ut_2" "c_code01_2" "level_2"    "tot_nm_hh_" "tot_pop_2"
##  [73] "m_pop_2"    "f_pop_2"    "tot_l6_2"   "m_l6_2"     "f_l6_2"     "tot_sc_2"
##  [79] "m_sc_2"     "f_sc_2"     "tot_st_2"   "m_st_2"     "f_st_2"     "tot_lit_2"
##  [85] "m_lit_2"    "f_lit_2"    "tot_illt_2" "m_illt_2"   "f_illt_2"   "tot_w_2"
##  [91] "m_w_2"      "f_w_2"      "tot_mnw_2"  "m_mnw_2"    "f_mnw_2"    "tot_cult_2"
##  [97] "m_cult_2"   "f_cult_2"   "tot_aglb_2" "m_aglb_2"   "f_aglb_2"   "tot_mfhh_2"
## [103] "m_mfhh_2"   "f_mfhh_2"   "tot_oth_w_" "m_oth_w_2"  "f_oth_w_2"  "tot_mrw_2"
## [109] "m_mrw_2"    "f_mrw_2"    "t_mrg_cu_1" "m_mrg_cu_1" "f_mrg_cu_1" "t_mrg_ag_1"
## [115] "m_mrg_ag_1" "f_mrg_ag_1" "t_mrg_hh_2" "m_mrg_hh_2" "f_mrg_hh_2" "t_mrg_oth_"
## [121] "m_mrg_oth_" "f_mrg_oth_" "tot_nnw_2"  "m_nnw_2"    "f_nnw_2"    "NEAR_FID"
## [127] "NEAR_DIST"  "ORIG_FID"   "Latitude"   "Longitude"
```

```
summary(mys)
```

```
##     OBJECTID        VILLAGE_ID                NAME          SUB_DISTRI            DISTRICT
##  Min.   :  1.00   8578   :  7   Waraha          :  7   Bhatkal   :12   Haveri          :152
##  1st Qu.: 66.75   28267  :  6   Tigani          :  6   Hangal    :26   Uttara Kannada:112
##  Median :132.50   28319  :  5   Guddadamadapur:  5   Hirekerur :78
##  Mean   :132.50   8291   :  5   Mudenur         :  5   Honavar   : 8
##  3rd Qu.:198.25   8843   :  5   Naroor          :  5   Ranibennur:48
##  Max.   :264.00   28400  :  4   Aigod           :  4   Siddapur  :51
##                   (Other):232   (Other)         :232   Sirsi     :41
##       STATE_UT               C_CODE01       LEVEL       TOT_NM_HH      TOT_POP
##  Karnataka:264   2911000600868500:  7   Forest :  2   327    :  7   1703   :  7
##                  2910000600759500:  6   Town   :  2   0      :  6   0      :  6
##                  2910000600760100:  5   Village:260   242    :  6   1006   :  6
##                  2911000600874000:  5                 13     :  5   1205   :  6
##                  2911000700883300:  5                 147    :  5   1988   :  5
##                  (Other)         :234                 209    :  5   2195   :  5
##                  NA's            :  2                 (Other):230   (Other):229
##      M_POP         F_POP         TOT_L6         M_L6          F_L6         TOT_SC
##  96     :  8   836    :  7   57     : 10   5      :  8   56     : 10   0      : 46
##  867    :  7   0      :  6   108    :  7   85     :  8   116    :  8   250    :  7
##  0      :  6   104    :  6   255    :  7   119    :  7   0      :  7   169    :  6
##  623    :  6   582    :  6   8      :  7   79     :  7   136    :  7   24     :  6
##  1028   :  5   1048   :  5   0      :  6   0      :  6   32     :  7   249    :  6
##  1147   :  5   499    :  5   150    :  6   15     :  6   93     :  7   32     :  6
##  (Other):227   (Other):229   (Other):221   (Other):222   (Other):218   (Other):187
##       M_SC          F_SC          TOT_ST         M_ST          F_ST        TOT_LIT
##  0      : 46   0      : 48   0      : 84   0      : 87   0      : 89   870    :  7
##  125    :  9   143    : 10   7      :  9   2      : 14   5      : 13   0      :  6
##  70     :  8   85     :  8   1      :  8   7      : 10   3      :  9   773    :  6
##  107    :  7   17     :  6   5      :  8   49     :  7   63     :  9   1265   :  5
##  15     :  7   2      :  6   112    :  7   5      :  7   1      :  6   1343   :  5
##  77     :  7   208    :  6   96     :  6   74     :  6   104    :  5   632    :  5
##  (Other):180   (Other):180   (Other):142   (Other):133   (Other):133   (Other):230
##      M_LIT         F_LIT        TOT_ILLT       M_ILLT        F_ILLT         TOT_W
##  444    :  8   323    :  9   833    :  7   320    :  7   58     : 11   804    :  7
##  1015   :  7   483    :  8   0      :  6   365    :  7   513    :  7   0      :  6
##  547    :  7   23     :  7   1514   :  6   0      :  6   565    :  7   643    :  6
##  0      :  6   0      :  6   232    :  6   179    :  6   0      :  6   1180   :  5
##  14     :  5   329    :  6   432    :  6   30     :  6   253    :  6   1183   :  5
```

```
##  354    :  5   147    :  5   18     :  5   153    :  5   221    :  5   229    :  5
##  (Other):226   (Other):223   (Other):228   (Other):227   (Other):222   (Other):230
##       M_W           F_W         TOT_MNW         M_MNW         F_MNW        TOT_CULT
##  272    :  7   341    :  7   619    :  7   412    :  7   0      :  9   0      : 10
##  463    :  7   0      :  6   0      :  6   0      :  6   1      :  9   241    :  9
##  0      :  6   229    :  6   264    :  6   326    :  6   116    :  8   12     :  6
##  403    :  6   240    :  6   486    :  6   370    :  6   207    :  7   220    :  6
##  23     :  5   191    :  5   539    :  6   339    :  5   3      :  7   334    :  6
##  329    :  5   455    :  5   866    :  6   570    :  5   191    :  6   199    :  5
##  (Other):228   (Other):229   (Other):227   (Other):229   (Other):218   (Other):222
##      M_CULT        F_CULT        TOT_AGLB        M_AGLB        F_AGLB        TOT_MFHH
##  0      : 10   0      : 26   0      : 20   0      : 26   0      : 32   0      : 69
##  192    : 10   3      : 16   200    : 10   22     :  9   6      : 11   1      : 26
##  189    :  8   1      : 15   340    : 10   1      :  7   2      :  9   4      : 17
##  197    :  8   11     : 12   1      :  9   182    :  7   77     :  9   2      : 15
##  9      :  7   10     : 10   221    :  6   234    :  7   1      :  7   5      : 11
##  158    :  6   28     :  9   2      :  5   144    :  6   158    :  7   3      : 10
##  (Other):215   (Other):176   (Other):204   (Other):202   (Other):189   (Other):116
##      M_MFHH        F_MFHH       TOT_OTH_W       M_OTH_W       F_OTH_W       TOT_MRW
##  0      :72   0      :131   0      : 10   0      : 13   0      : 24   0      : 18
##  1      :33   1      : 47   35     : 10   16     : 11   1      : 24   34     : 10
##  5      :18   2      : 22   16     :  8   3      :  9   2      : 21   157    :  8
##  2      :17   3      : 14   1      :  7   14     :  8   3      : 19   15     :  7
##  4      :17   5      :  9   19     :  7   17     :  8   5      : 18   185    :  7
##  6      :17   6      :  6   4      :  7   45     :  8   7      : 16   181    :  6
##  (Other):90   (Other): 35   (Other):215   (Other):207   (Other):142   (Other):208
##      M_MRW         F_MRW        T_MRG_CULT    M_MRG_CULT   F_MRG_CULT    T_MRG_AGLB
##  0      : 34   0      : 24   0      : 55   0      :86   0      : 68   0      : 35
##  1      : 20   113    :  9   4      : 18   2      :33   2      : 15   11     : 11
##  14     : 13   134    :  9   17     : 17   1      :18   4      : 14   8      : 10
##  4      : 12   8      :  9   2      : 10   12     :12   6      : 13   9      : 10
##  51     : 11   124    :  6   8      :  9   3      :12   1      : 12   17     :  9
##  9      : 11   15     :  6   1      :  8   7      :11   11     : 12   2      :  8
##  (Other):163   (Other):201   (Other):147   (Other):92   (Other):130   (Other):181
##    M_MRG_AGLB    F_MRG_AGLB     T_MRG_HH      M_MRG_HH      F_MRG_HH     T_MRG_OTH
##  0      : 71   0      : 37   0      :151   0      :195   0      :161   0      :69
##  1      : 31   103    : 12   1      : 31   1      : 27   1      : 30   1      :28
##  4      : 14   3      :  9   2      : 20   3      : 15   2      : 20   2      :26
##  25     : 13   8      :  9   3      : 11   10     :  6   3      : 11   5      :19
##  3      :  9   2      :  8   15     :  6   7      :  6   8      :  7   3      :14
##  11     :  7   5      :  8   6      :  5   4      :  5   4      :  6   7      :11
##  (Other):119   (Other):181   (Other): 40   (Other): 10   (Other): 29   (Other):97
##    M_MRG_OTH     F_MRG_OTH       TOT_NNW         M_NNW         F_NNW       village__1
##  0      :95   0      :105   899    :  7   404    :  7   342    :  7   29058  :  7
##  1      :29   1      : 38   0      :  6   0      :  6   343    :  7   29808  :  7
##  2      :26   2      : 26   562    :  6   11     :  6   495    :  7   8066   :  6
##  3      :20   3      : 18   1015   :  5   178    :  6   0      :  6   8450   :  6
##  4      :15   11     : 11   1162   :  5   220    :  6   188    :  5   8833   :  6
##  5      :10   5      :  6   277    :  5   224    :  5   224    :  5   8235   :  5
##  (Other):69   (Other): 60   (Other):230   (Other):228   (Other):227   (Other):227
##                        name_2            sub_dist_1       district_2      state_ut_2
##  Jog Falls                 :  7   Harapanahalli: 10   Davanagere: 68   Karnataka:264
##  Ukkadagathri              :  7   Harihar      : 43   Shimoga   :196
##  Chandraguthi State Forest:  6   Honnali      : 15
```

```
##  Mudidoddikoppa (Mudi)    :  6   Sagar        : 45
##  Sydur                    :  6   Shikarpur    : 42
##  Eduru                    :  5   Sorab        :109
##  (Other)                  :227
##             c_code01_2     level_2      tot_nm_hh_    tot_pop_2      m_pop_2       f_pop_2
##  2914000101047000:  7   Town   : 10   0      : 16   0      : 16   0      : 16   0      : 16
##  2915000141501000:  7   Village:254   2838   :  7   12570  :  7   1252   :  7   1080   :  7
##  2915000101142200:  6                 441    :  7   2332   :  7   6364   :  7   6206   :  7
##  2915000201166400:  6                 1      :  6   2125   :  6   1075   :  6   1050   :  6
##  2915000201177400:  6                 137    :  6   2954   :  6   1508   :  6   1446   :  6
##  2915000101159600:  5                 157    :  6   167    :  5   82     :  6   410    :  5
##  (Other)         :227                 (Other):216   (Other):217   (Other):216   (Other):217
##     tot_l6_2       m_l6_2        f_l6_2       tot_sc_2       m_sc_2        f_sc_2
##  0      : 22   0      : 27   0      : 22   0      : 60   0      : 62   0      : 60
##  149    :  8   152    :  7   62     : 11   10     :  8   5      :  9   148    :  9
##  1412   :  7   37     :  7   23     :  8   2283   :  7   11     :  7   5      :  9
##  288    :  7   705    :  7   103    :  7   328    :  7   1145   :  7   1138   :  7
##  231    :  6   8      :  7   136    :  7   98     :  7   141    :  7   140    :  6
##  356    :  6   87     :  7   38     :  7   281    :  6   180    :  7   154    :  6
##  (Other):208   (Other):202   (Other):202   (Other):169   (Other):165   (Other):167
##     tot_st_2       m_st_2        f_st_2      tot_lit_2      m_lit_2       f_lit_2
##  0      : 95   0      : 95   0      : 97   0      : 16   0      : 18   0      : 16
##  254    :  7   139    :  7   115    :  7   1245   :  7   5106   :  7   448    :  9
##  301    :  7   156    :  7   145    :  7   9295   :  7   797    :  7   165    :  7
##  39     :  7   15     :  6   21     :  7   1206   :  6   1073   :  6   296    :  7
##  528    :  6   270    :  6   258    :  6   142    :  6   69     :  6   4189   :  7
##  550    :  5   82     :  6   3      :  6   1883   :  6   705    :  6   501    :  6
##  (Other):137   (Other):137   (Other):134   (Other):216   (Other):214   (Other):212
##    tot_illt_2     m_illt_2      f_illt_2       tot_w_2        m_w_2         f_w_2
##  0      : 16   0      : 16   0      : 16   0      : 16   0      : 16   0      : 20
##  1087   :  7   2      :  8   549    :  8   1237   :  7   3286   :  7   45     :  8
##  3275   :  7   1258   :  7   2017   :  7   4029   :  7   784    :  7   107    :  7
##  1071   :  6   455    :  7   632    :  7   564    :  7   277    :  6   453    :  7
##  159    :  6   100    :  6   19     :  6   107    :  6   624    :  6   743    :  7
##  919    :  6   293    :  6   211    :  6   1500   :  6   929    :  6   154    :  6
##  (Other):216   (Other):214   (Other):214   (Other):215   (Other):216   (Other):209
##    tot_mnw_2      m_mnw_2       f_mnw_2      tot_cult_2     m_cult_2      f_cult_2
##  0      : 19   0      : 19   0      : 20   0      : 21   0      : 21   0      : 47
##  1123   :  7   222    :  7   1      :  9   189    :  7   60     :  8   1      : 18
##  203    :  7   2854   :  7   44     :  9   609    :  7   157    :  7   2      : 13
##  3339   :  7   730    :  7   156    :  7   320    :  6   398    :  7   3      : 11
##  1091   :  6   584    :  6   393    :  7   594    :  6   555    :  6   92     :  9
##  626    :  6   852    :  6   485    :  7   647    :  6   561    :  6   211    :  7
##  (Other):212   (Other):212   (Other):205   (Other):211   (Other):209   (Other):159
##    tot_aglb_2     m_aglb_2      f_aglb_2     tot_mfhh_2     m_mfhh_2      f_mfhh_2
##  0      : 36   0      : 42   0      : 46   0      :83   0      :84   0      :153
##  2      :  9   47     : 12   3      : 15   1      :22   1      :33   2      : 39
##  67     :  9   4      : 10   1      : 12   2      :20   3      :26   1      : 24
##  360    :  7   201    :  8   2      : 10   3      :20   13     :19   3      : 15
##  16     :  6   10     :  7   18     :  8   12     :18   9      :18   5      : 12
##  292    :  6   164    :  6   21     :  8   5      :17   2      :16   8      :  5
##  (Other):191   (Other):179   (Other):165   (Other):84   (Other):68   (Other): 16
##    tot_oth_w_     m_oth_w_2     f_oth_w_2     tot_mrw_2      m_mrw_2       f_mrw_2
##  0      : 42   0      : 44   0      : 57   0      : 29   0      : 43   0      : 33
```

```
##  15      : 13   23      : 12   3       : 26   114     : 10   2       : 10   1       : 16
##  1       :  9   12      : 11   7       : 17   86      : 10   8       : 10   107     : 10
##  17      :  8   1       :  8   2       : 15   1       :  7   4       :  8   46      :  9
##  138     :  7   118     :  8   6       : 14   135     :  7   54      :  8   60      :  9
##  3065    :  7   11      :  7   5       : 13   2       :  7   7       :  8   258     :  7
##  (Other):178   (Other):174   (Other):122   (Other):194   (Other):177   (Other):180
##    t_mrg_cu_1    m_mrg_cu_1   f_mrg_cu_1    t_mrg_ag_1    m_mrg_ag_1    f_mrg_ag_1
##  0       : 51   0       :86   0       : 70   0       : 36   0       : 59   0       : 42
##  1       : 32   1       :35   1       : 35   2       : 10   3       : 15   3       : 10
##  2       : 15   2       :18   2       : 22   3       :  8   1       : 13   1       :  9
##  4       : 15   3       :17   4       : 12   236     :  7   7       : 13   106     :  9
##  5       : 11   40      :11   6       : 11   7       :  7   2       : 12   54      :  9
##  15      :  7   5       :11   3       :  9   85      :  7   5       : 11   7       :  9
##  (Other):133   (Other):86   (Other):105   (Other):189   (Other):141   (Other):176
##    t_mrg_hh_2    m_mrg_hh_2    f_mrg_hh_2    t_mrg_oth_    m_mrg_oth_    f_mrg_oth_
##  0       :150   0       :193   0       :169   0       : 93   0       :107   0       :117
##  1       : 44   1       : 28   1       : 34   1       : 20   1       : 25   1       : 36
##  2       : 14   3       : 14   2       : 23   3       : 14   6       : 18   5       : 16
##  6       : 12   2       : 10   4       : 12   7       : 12   2       : 11   3       : 14
##  5       : 11   9       :  5   3       :  7   13      :  9   7       : 11   2       :  9
##  4       :  6   4       :  4   30      :  5   6       :  8   8       : 11   7       :  8
##  (Other): 27   (Other): 10   (Other): 14   (Other):108   (Other): 81   (Other): 64
##    tot_nnw_2      m_nnw_2       f_nnw_2       NEAR_FID        NEAR_DIST    ORIG_FID
##  0       : 18   0       : 18   0       : 18   Min.   :  80.0   Min.   :0   Min.   :  0.00
##  1095    :  7   451     : 11   5463    :  7   1st Qu.: 319.0   1st Qu.:0   1st Qu.: 65.75
##  8541    :  7   190     :  9   627     :  7   Median : 529.0   Median :0   Median :131.50
##  1413    :  6   3078    :  7   962     :  7   Mean   : 693.4   Mean   :0   Mean   :131.50
##  1454    :  6   468     :  7   224     :  6   3rd Qu.:1003.0   3rd Qu.:0   3rd Qu.:197.25
##  350     :  6   341     :  6   875     :  6   Max.   :2330.0   Max.   :0   Max.   :263.00
##  (Other):214   (Other):206   (Other):213
##     Latitude       Longitude
##  Min.   :13.99   Min.   :74.64
##  1st Qu.:14.33   1st Qu.:74.95
##  Median :14.47   Median :75.20
##  Mean   :14.44   Mean   :75.23
##  3rd Qu.:14.56   3rd Qu.:75.56
##  Max.   :14.72   Max.   :75.82
##
```

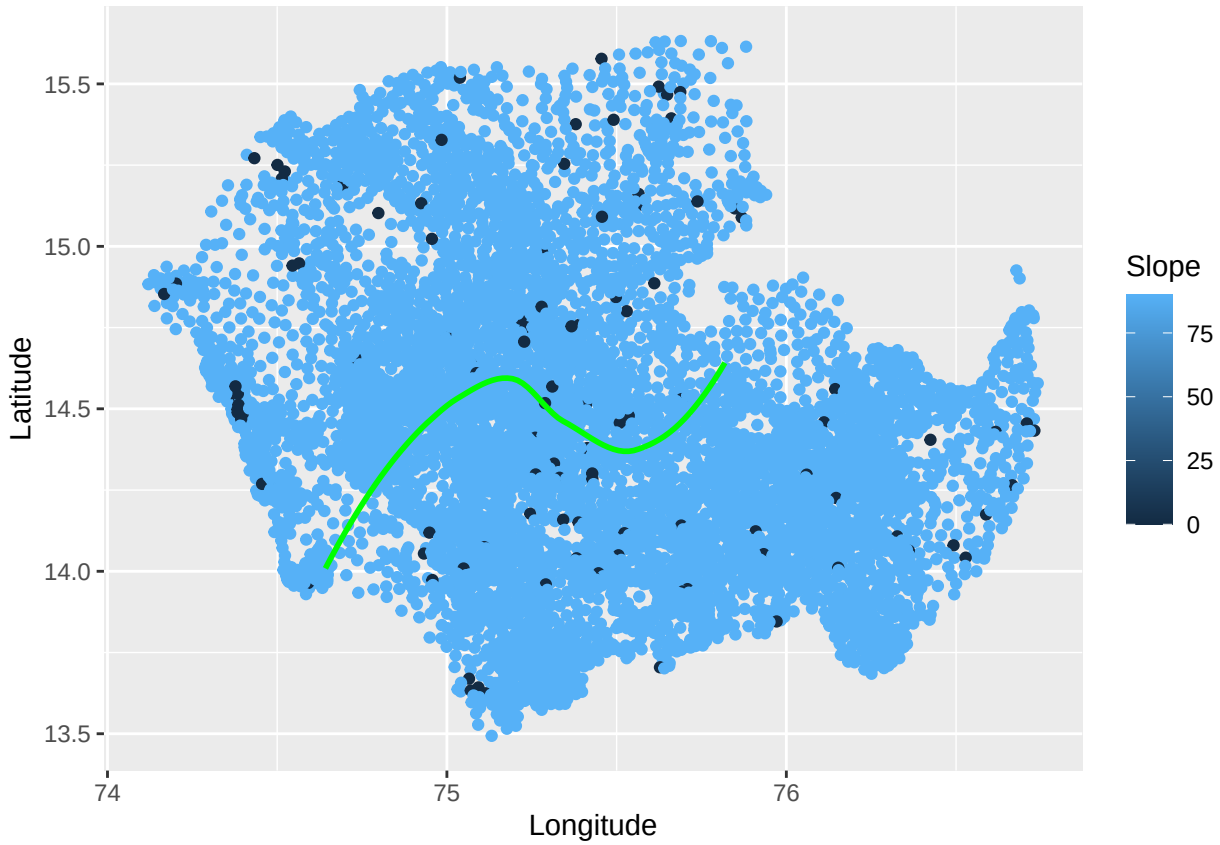
```
hyd=read.dbf("hyd_borderpts.dbf")
names(hyd)
```

```
##   [1] "OBJECTID"   "VILLAGE_ID" "NAME"       "SUB_DISTRI" "DISTRICT"   "STATE_UT"
##   [7] "C_CODE01"   "LEVEL"      "TOT_NM_HH"  "TOT_POP"    "M_POP"      "F_POP"
##  [13] "TOT_L6"     "M_L6"       "F_L6"       "TOT_SC"     "M_SC"       "F_SC"
##  [19] "TOT_ST"     "M_ST"       "F_ST"       "TOT_LIT"    "M_LIT"      "F_LIT"
##  [25] "TOT_ILLT"   "M_ILLT"     "F_ILLT"     "TOT_W"      "M_W"        "F_W"
##  [31] "TOT_MNW"    "M_MNW"      "F_MNW"      "TOT_CULT"   "M_CULT"     "F_CULT"
##  [37] "TOT_AGLB"   "M_AGLB"     "F_AGLB"     "TOT_MFHH"   "M_MFHH"     "F_MFHH"
##  [43] "TOT_OTH_W"  "M_OTH_W"    "F_OTH_W"    "TOT_MRW"    "M_MRW"      "F_MRW"
##  [49] "T_MRG_CULT" "M_MRG_CULT" "F_MRG_CULT" "T_MRG_AGLB" "M_MRG_AGLB" "F_MRG_AGLB"
##  [55] "T_MRG_HH"   "M_MRG_HH"   "F_MRG_HH"   "T_MRG_OTH"  "M_MRG_OTH"  "F_MRG_OTH"
##  [61] "TOT_NNW"    "M_NNW"      "F_NNW"      "village__1" "name_2"     "sub_dist_1"
##  [67] "district_2" "state_ut_2" "c_code01_2" "level_2"    "tot_nm_hh_" "tot_pop_2"
##  [73] "m_pop_2"    "f_pop_2"    "tot_l6_2"   "m_l6_2"     "f_l6_2"     "tot_sc_2"
```

```
##  [79] "m_sc_2"     "f_sc_2"     "tot_st_2"   "m_st_2"     "f_st_2"     "tot_lit_2"
##  [85] "m_lit_2"    "f_lit_2"    "tot_illt_2" "m_illt_2"   "f_illt_2"   "tot_w_2"
##  [91] "m_w_2"      "f_w_2"      "tot_mnw_2"  "m_mnw_2"    "f_mnw_2"    "tot_cult_2"
##  [97] "m_cult_2"   "f_cult_2"   "tot_aglb_2" "m_aglb_2"   "f_aglb_2"   "tot_mfhh_2"
## [103] "m_mfhh_2"   "f_mfhh_2"   "tot_oth_w_" "m_oth_w_2"  "f_oth_w_2"  "tot_mrw_2"
## [109] "m_mrw_2"    "f_mrw_2"    "t_mrg_cu_1" "m_mrg_cu_1" "f_mrg_cu_1" "t_mrg_ag_1"
## [115] "m_mrg_ag_1" "f_mrg_ag_1" "t_mrg_hh_2" "m_mrg_hh_2" "f_mrg_hh_2" "t_mrg_oth_"
## [121] "m_mrg_oth_" "f_mrg_oth_" "tot_nnw_2"  "m_nnw_2"    "f_nnw_2"    "ORIG_FID"
## [127] "Latitude"   "Longitude"
```

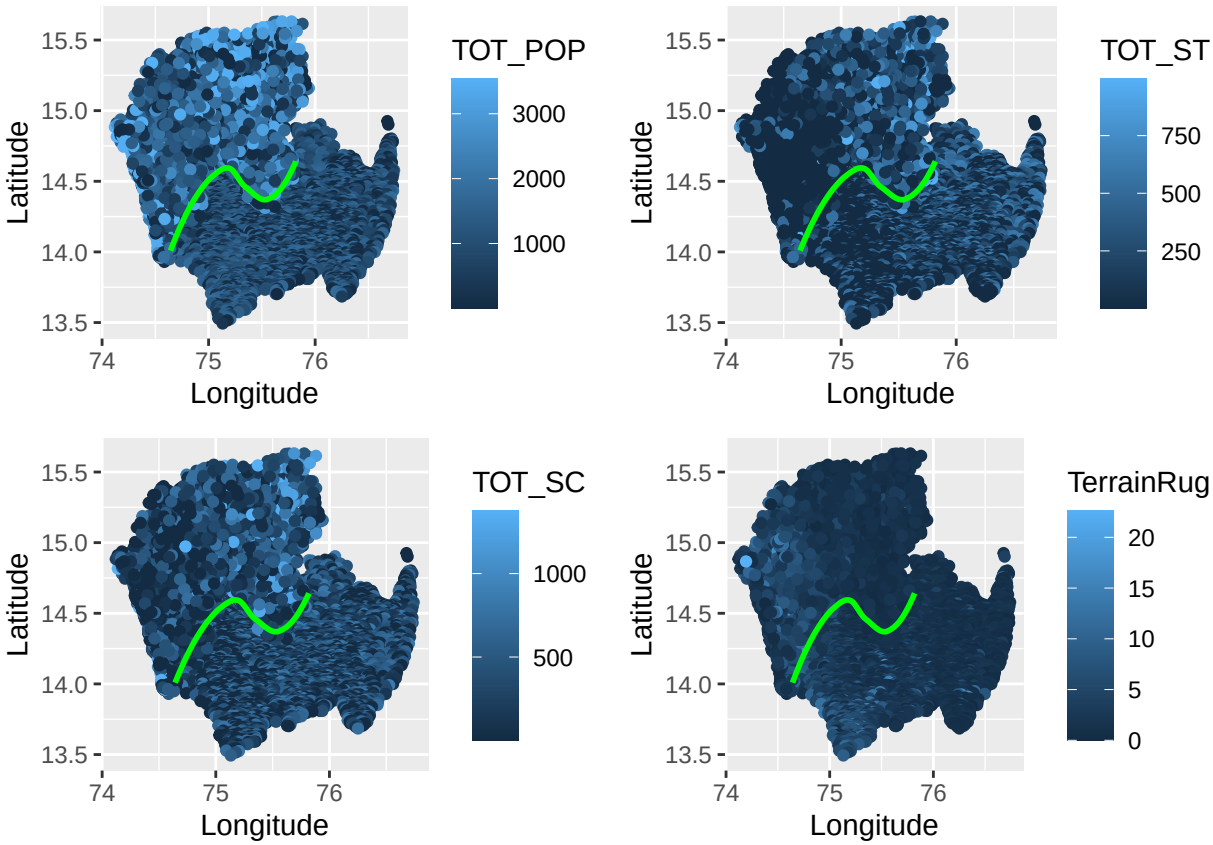
```
#bw 100 km (basically the whole territory)
#mys
g1=ggplot(rd100.mb, aes(x=Longitude, y=Latitude, colour=TOT_POP))+
  geom_point()+
  geom_smooth(data=mys, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)

g2=ggplot(rd100.mb, aes(x=Longitude, y=Latitude, colour=TOT_SC))+
  geom_point()+
  geom_smooth(data=mys, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)

g3=ggplot(rd100.mb, aes(x=Longitude, y=Latitude, colour=TOT_ST))+
  geom_point()+
  geom_smooth(data=mys, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)

g4=ggplot(rd100.mb, aes(x=Longitude, y=Latitude, colour=TerrainRug))+
  geom_point()+
  geom_smooth(data=mys, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)

ggplot(rd100.mb, aes(x=Longitude, y=Latitude, colour=Slope))+
  geom_point()+
  geom_smooth(data=mys, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



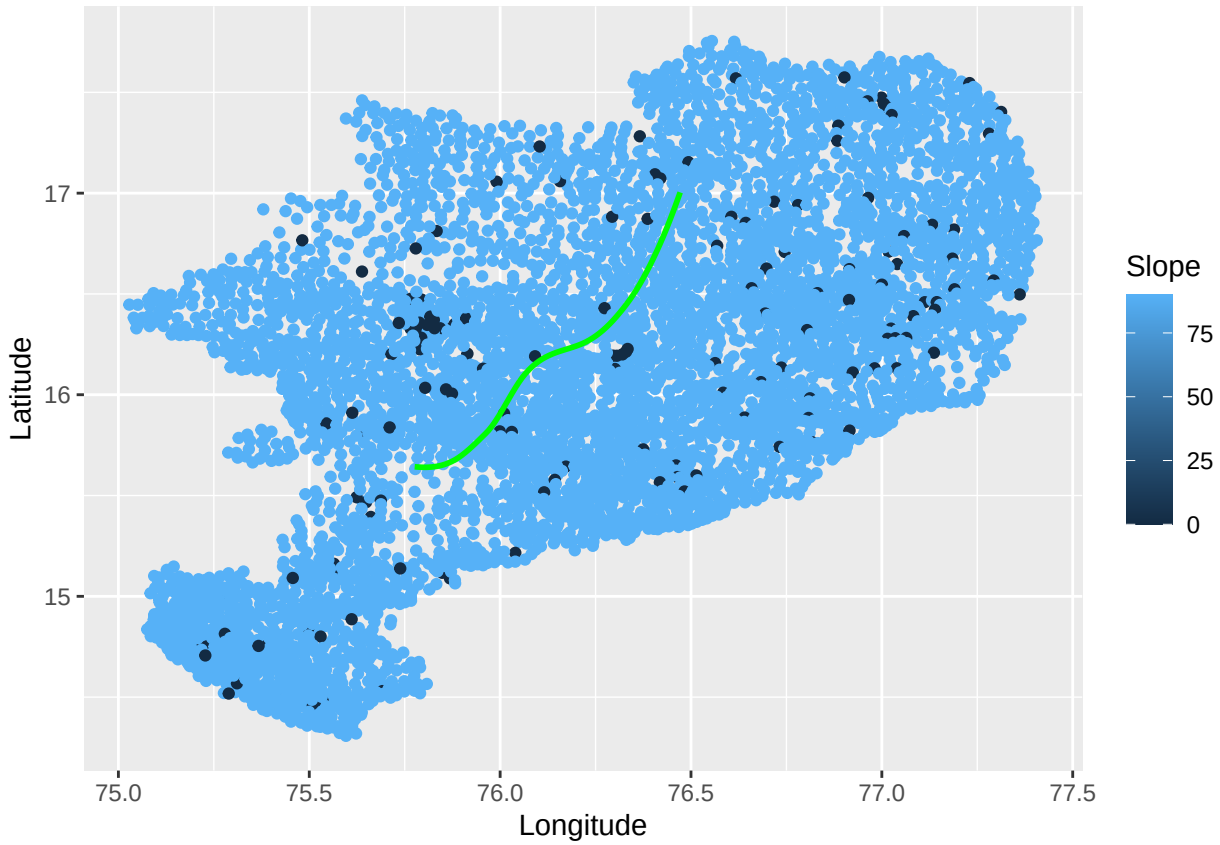

```
multiplot(g1, g2, g3, g4, cols=2)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



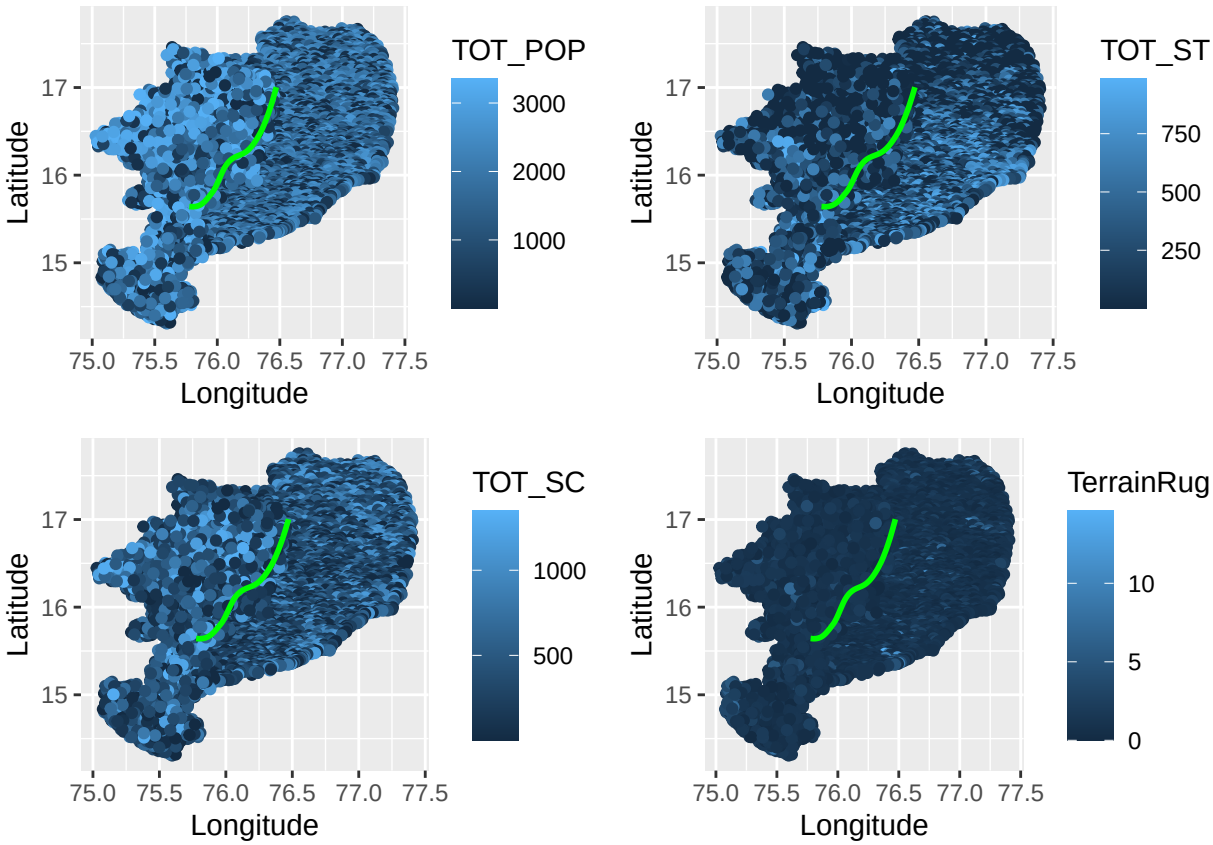

```
#hyd
g1=ggplot(rd100.hb, aes(x=Longitude, y=Latitude, colour=TOT_POP))+
  geom_point()+
  geom_smooth(data=hyd, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)

g2=ggplot(rd100.hb, aes(x=Longitude, y=Latitude, colour=TOT_SC))+
  geom_point()+
  geom_smooth(data=hyd, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)

g3=ggplot(rd100.hb, aes(x=Longitude, y=Latitude, colour=TOT_ST))+
  geom_point()+
  geom_smooth(data=hyd, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)

g4=ggplot(rd100.hb, aes(x=Longitude, y=Latitude, colour=TerrainRug))+
  geom_point()+
  geom_smooth(data=hyd, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)

ggplot(rd100.hb, aes(x=Longitude, y=Latitude, colour=Slope))+
  geom_point()+
  geom_smooth(data=hyd, aes(x=Longitude, y=Latitude), col="green", size=1, se=FALSE)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

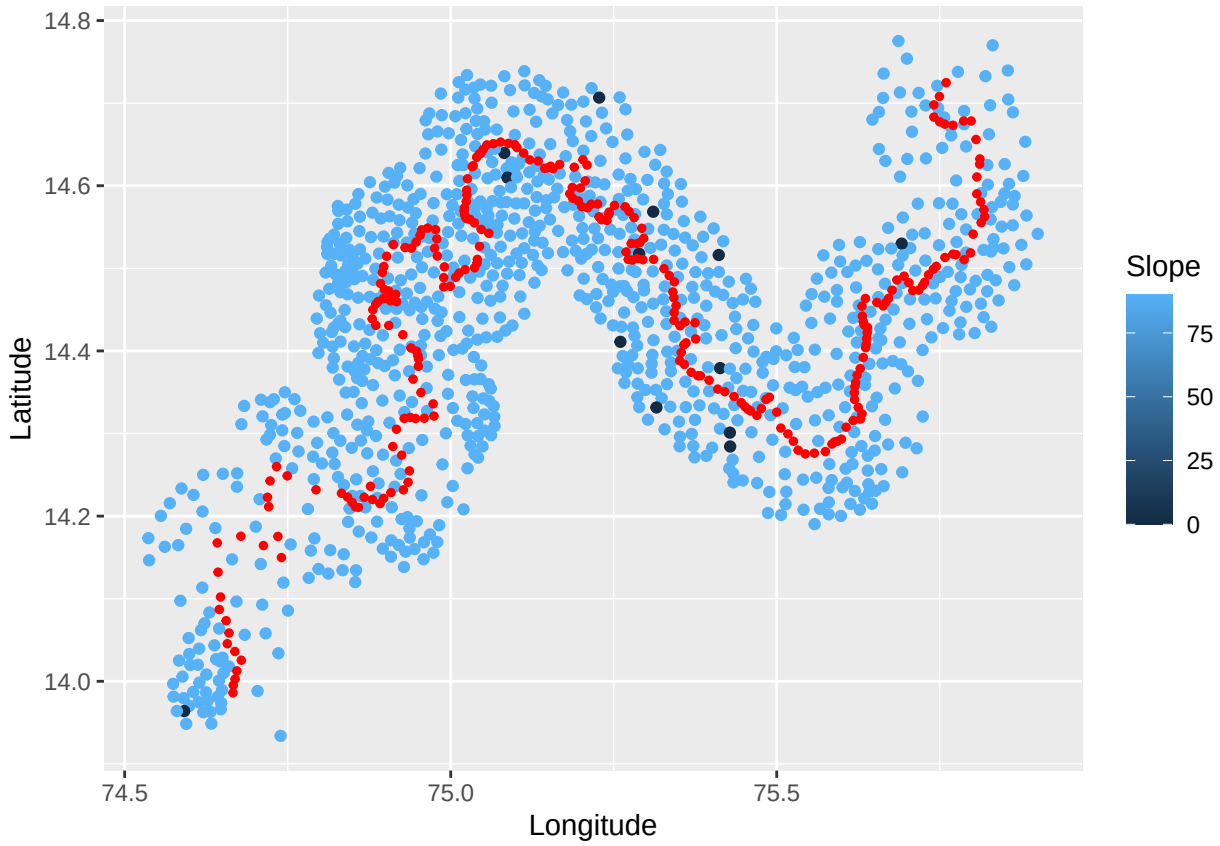



```
multiplot(g1, g2, g3, g4, cols=2)
```

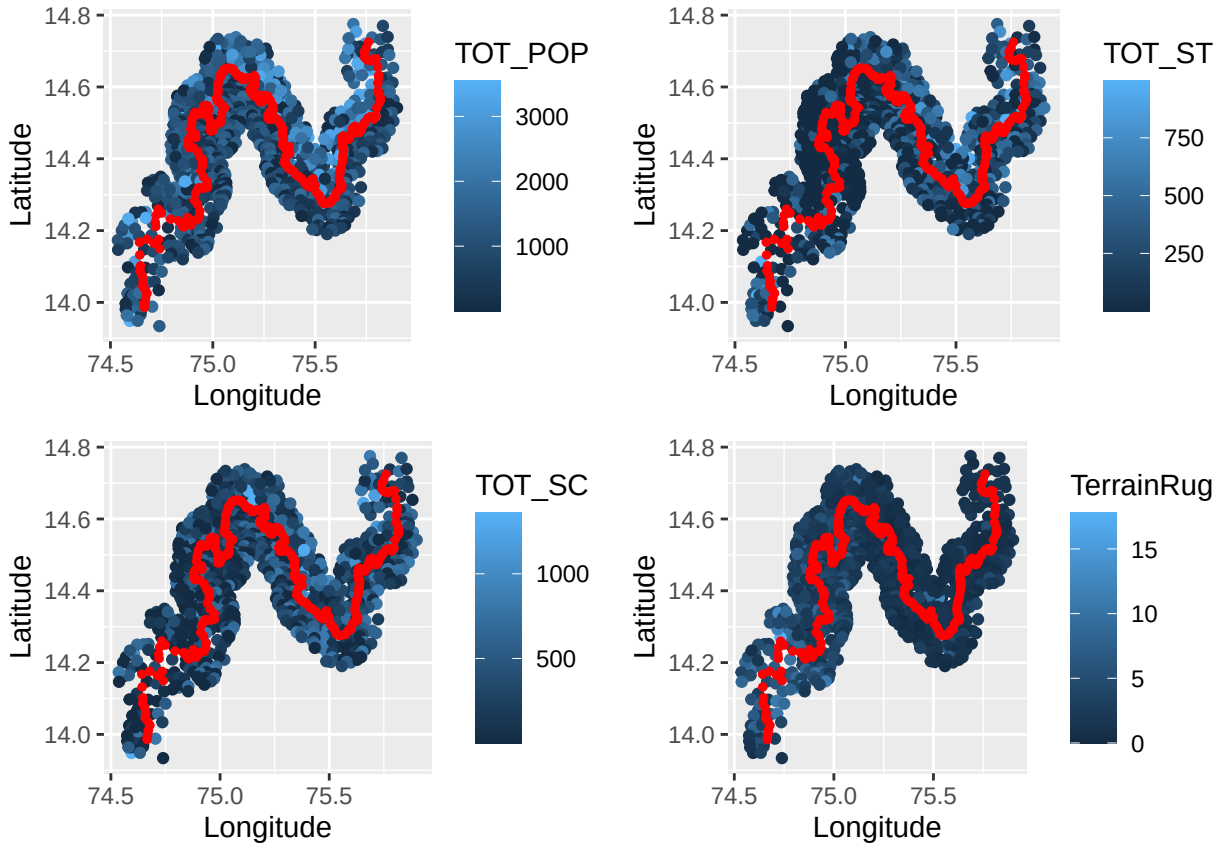
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



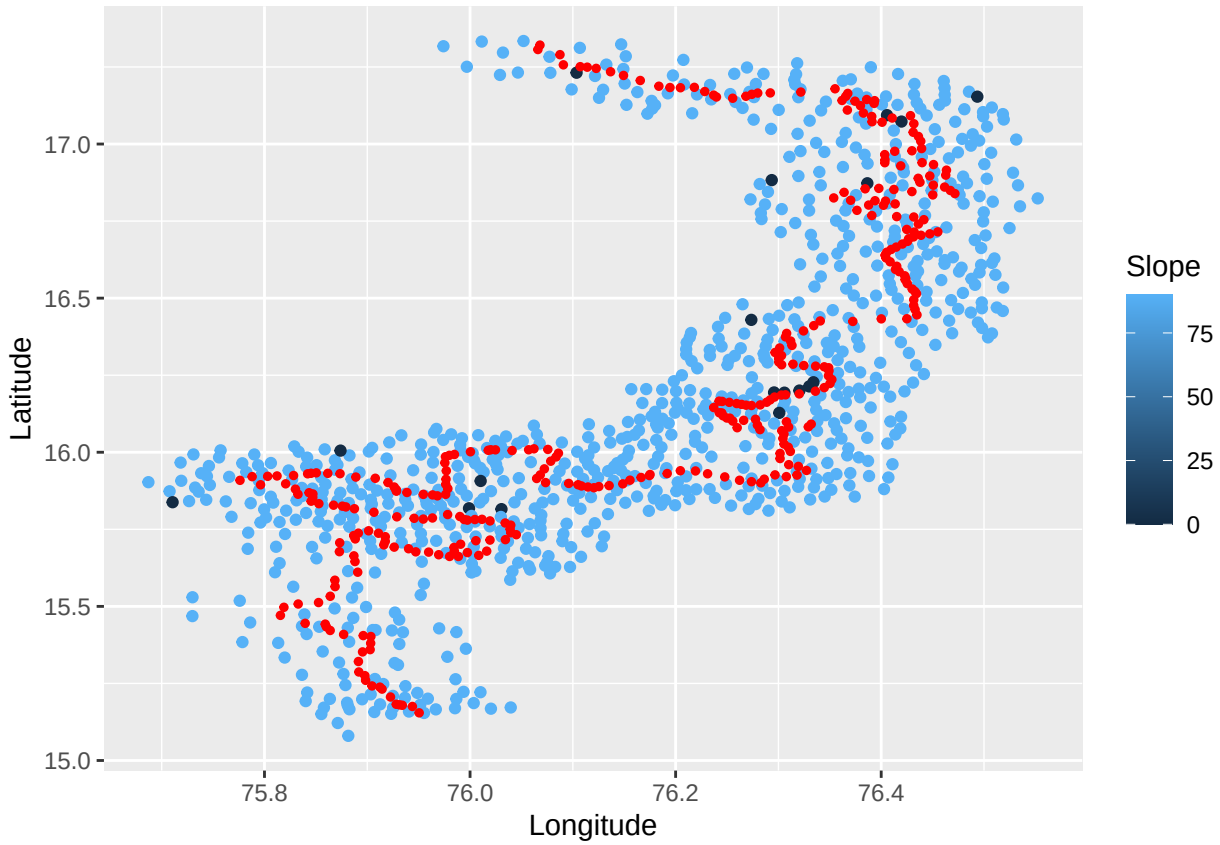

```
#baseline bw=10km
#mys
g1=ggplot(rd10.mb, aes(x=Longitude, y=Latitude, colour=TOT_POP))+
  geom_point()+
  geom_point(data=mys, aes(x=Longitude, y=Latitude), col="red", size=1)

g2=ggplot(rd10.mb, aes(x=Longitude, y=Latitude, colour=TOT_SC))+
  geom_point()+
  geom_point(data=mys, aes(x=Longitude, y=Latitude), col="red", size=1)

g3=ggplot(rd10.mb, aes(x=Longitude, y=Latitude, colour=TOT_ST))+
  geom_point()+
  geom_point(data=mys, aes(x=Longitude, y=Latitude), col="red", size=1)

g4=ggplot(rd10.mb, aes(x=Longitude, y=Latitude, colour=TerrainRug))+
  geom_point()+
  geom_point(data=mys, aes(x=Longitude, y=Latitude), col="red", size=1)

ggplot(rd10.mb, aes(x=Longitude, y=Latitude, colour=Slope))+
  geom_point()+
  geom_point(data=mys, aes(x=Longitude, y=Latitude), col="red", size=1)
```



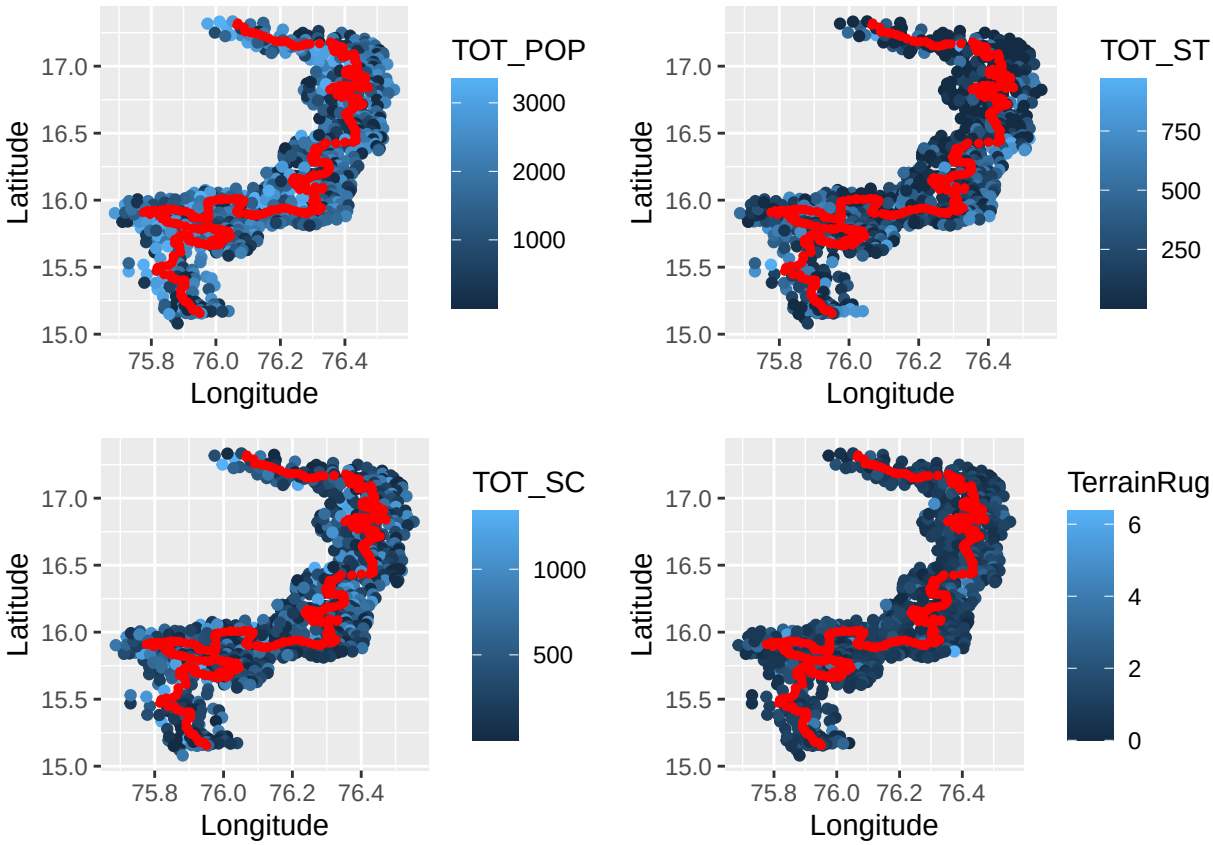

```
multiplot(g1, g2, g3, g4, cols=2)
```




```
#hyd
g1=ggplot(rd10.hb, aes(x=Longitude, y=Latitude, colour=TOT_POP))+
  geom_point()+
  geom_point(data=hyd, aes(x=Longitude, y=Latitude), col="red", size=1)

g2=ggplot(rd10.hb, aes(x=Longitude, y=Latitude, colour=TOT_SC))+
  geom_point()+
  geom_point(data=hyd, aes(x=Longitude, y=Latitude), col="red", size=1)

g3=ggplot(rd10.hb, aes(x=Longitude, y=Latitude, colour=TOT_ST))+
  geom_point()+
  geom_point(data=hyd, aes(x=Longitude, y=Latitude), col="red", size=1)

g4=ggplot(rd10.hb, aes(x=Longitude, y=Latitude, colour=TerrainRug))+
  geom_point()+
  geom_point(data=hyd, aes(x=Longitude, y=Latitude), col="red", size=1)

ggplot(rd10.hb, aes(x=Longitude, y=Latitude, colour=Slope))+
  geom_point()+
  geom_point(data=hyd, aes(x=Longitude, y=Latitude), col="red", size=1)
```




```
multiplot(g1, g2, g3, g4, cols=2)
```


TOT_POP
3000
2000
1000
TOT_ST
750
500
250
TOT_SC
1000
500
TerrainRug
6
4
2
0
Latitude
Longitude
17.0
16.5
16.0
15.5
15.0
75.8
76.0
76.2
76.4