Questoes.R

edevaldogaudencio

2020-12-13

# Metrado Profissional em Econômia  
# Alunos: Jeferson, Alfredo e Edevaldo  
# Disciplina: Estatística  
# Professor: Bruno De Oliveira Cruz  
# Monitor: hiago Mendes Rosa  
#### Partes dos códigos utilizados neste programa foram desenvolvidos a partir   
#### dos exemplos disponibilizados pelo monitor Thiago Mendes Rosa  
  
  
######Carregar pacotes necessários  
library(readr)  
library(survey)

## Loading required package: grid

## Loading required package: Matrix

## Loading required package: survival

##   
## Attaching package: 'survey'

## The following object is masked from 'package:graphics':  
##   
## dotchart

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(srvyr)

##   
## Attaching package: 'srvyr'

## The following object is masked from 'package:stats':  
##   
## filter

library(Hmisc)

## Loading required package: lattice

## Loading required package: Formula

## Loading required package: ggplot2

##   
## Attaching package: 'Hmisc'

## The following object is masked from 'package:srvyr':  
##   
## summarize

## The following objects are masked from 'package:dplyr':  
##   
## src, summarize

## The following object is masked from 'package:survey':  
##   
## deff

## The following objects are masked from 'package:base':  
##   
## format.pval, units

library(lubridate)

##   
## Attaching package: 'lubridate'

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

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ tibble 3.0.4 ✓ stringr 1.4.0  
## ✓ tidyr 1.1.2 ✓ forcats 0.5.0  
## ✓ purrr 0.3.4

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## x lubridate::as.difftime() masks base::as.difftime()  
## x lubridate::date() masks base::date()  
## x tidyr::expand() masks Matrix::expand()  
## x srvyr::filter() masks dplyr::filter(), stats::filter()  
## x lubridate::intersect() masks base::intersect()  
## x dplyr::lag() masks stats::lag()  
## x tidyr::pack() masks Matrix::pack()  
## x lubridate::setdiff() masks base::setdiff()  
## x Hmisc::src() masks dplyr::src()  
## x Hmisc::summarize() masks srvyr::summarize(), dplyr::summarize()  
## x lubridate::union() masks base::union()  
## x tidyr::unpack() masks Matrix::unpack()

library(convey)  
library(ineq)  
options(dplyr.width = Inf)  
  
######## Preparando o ambiente  
  
 ######Importaçõa de domicílios  
 pdad\_2018\_domicilios <- read\_delim("dados/PDAD\_2018\_dom2018\_31ras.csv",   
 ";", escape\_double = FALSE, locale = locale(decimal\_mark = ","),   
 trim\_ws = TRUE)

##   
## ── Column specification ────────────────────────────────────────────────────────  
## cols(  
## .default = col\_double(),  
## datavisita = col\_character()  
## )  
## ℹ Use `spec()` for the full column specifications.

######Importaçõa de moradores  
 pdad\_2018\_moradores <- read\_delim("dados/PDAD\_2018\_mor2018\_31ras.csv",   
 ";", escape\_double = FALSE, locale = locale(decimal\_mark = ","),   
 trim\_ws = TRUE)

##   
## ── Column specification ────────────────────────────────────────────────────────  
## cols(  
## .default = col\_double(),  
## E051 = col\_character()  
## )  
## ℹ Use `spec()` for the full column specifications.

###### Importando o dicionário de dados moradores  
 dic\_moradores <- readxl::read\_excel("dados/Dicionario\_de\_Variaveis\_PDAD\_2018.xlsx",  
 skip = 1,  
 sheet = 2)  
   
   
 ### Adicionar rótulos à base da pdad  
 # Criar um objeto com os rótulos  
 var.labels <- dic\_moradores$`Descrição da coluna` %>%  
 # Retirar as linhas ausentes  
 na.omit  
   
 # Nomear esses rótulos com o nome das variáveis do nosso banco de dados   
 names(var.labels) <- names(pdad\_2018\_moradores)  
   
 # Adicionar os rótulos ao nosso banco de dados  
 pdad\_2018\_moradores <- Hmisc::upData(pdad\_2018\_moradores, labels = var.labels)

## Input object size: 58496240 bytes; 104 variables 69654 observations  
## New object size: 30418728 bytes; 104 variables 69654 observations

# Verificar o resultado  
 #Hmisc::describe(pdad\_2018\_moradores)  
   
   
 ###### Importando o dicionário de dados domicilios  
 dic\_domicilios <- readxl::read\_excel("dados/Dicionario\_de\_Variaveis\_PDAD\_2018.xlsx",  
 skip = 1,  
 sheet = 1)  
 # Adicionar rótulos à base da pdad  
 # Criar um objeto com os rótulos  
 var.labelsDom <- dic\_domicilios$`Descrição da coluna` %>%  
 # Retirar as linhas ausentes  
 na.omit  
   
 # Nomear esses rótulos com o nome das variáveis do nosso banco de dados   
 names(var.labelsDom) <- names(pdad\_2018\_domicilios)  
   
 # Adicionar os rótulos ao nosso banco de dados  
 pdad\_2018\_domicilios <- Hmisc::upData(pdad\_2018\_domicilios, labels = var.labelsDom)

## Input object size: 22872224 bytes; 130 variables 21908 observations  
## New object size: 11730408 bytes; 130 variables 21908 observations

# Verificar o resultado  
 #Hmisc::describe(pdad\_2018\_domicilios)  
   
 # Criar variável para escolaridade  
 pdad\_2018\_moradores <- pdad\_2018\_moradores %>%  
 # Fazer a escolaridade de quem não estuda  
 dplyr::mutate(escolaridade\_nao\_estuda=case\_when(F02==4~"Sem escolaridade",  
 F11==1~"Sem escolaridade",  
 F11==2&F12==10~"Sem escolaridade",  
 F11==3&F12==10~"Sem escolaridade",  
 F11==2&F12 %in% c(1:7)~"Fundamental incompleto",  
 F11==3&F12 %in% c(1:8)~"Fundamental incompleto",  
 F11==5&F13==2~"Fundamental incompleto",  
 F11==2&F12==8~"Fundamental completo",  
 F11==3&F12==9~"Fundamental completo",  
 F11==5&F13==1~"Fundamental completo",  
 F11==4&F12==10~"Fundamental completo",  
 F11==4&F12 %in% c(1:2)~"Médio incompleto",  
 F11==6&F13==2~"Médio incompleto",  
 F11==4&F12 %in% c(3,4)~"Médio completo",  
 F11==6&F13==1~"Médio completo",  
 F11==7&F13==2~"Superior incompleto",  
 F11==7&F13==1~"Superior completo",  
 F11 %in% c(8:10)~"Superior completo",  
 TRUE~NA\_character\_),  
 # Fazer a escolaridade de quem estuda  
 escolaridadet=case\_when(F07 %in% c(1,2,3)~"Sem escolaridade",  
 F07 %in% c(4,7)~"Fundamental incompleto",  
 F07 %in% c(5,6,8)~"Médio incompleto",  
 F07==9~"Superior incompleto",  
 F07 %in% c(10:12)~"Superior completo",  
 TRUE~escolaridade\_nao\_estuda),  
 # Ajustar a escolaridade de quem já concluiu outro curso superior  
 escolaridadet=case\_when(F09==1&F10 %in% c(1:4)~"Superior completo",  
 TRUE~escolaridadet),  
 # Ajustar o fator ordenado  
 escolaridade=factor(ordered(case\_when(escolaridadet=="Sem escolaridade"~1,  
 escolaridadet=="Fundamental incompleto"~2,  
 escolaridadet=="Fundamental completo"~3,  
 escolaridadet=="Médio incompleto"~4,  
 escolaridadet=="Médio completo"~5,  
 escolaridadet=="Superior incompleto"~6,  
 escolaridadet=="Superior completo"~7),  
 levels=c(1:7),  
 labels=c("Sem escolaridade",  
 "Fundamental incompleto",  
 "Fundamental completo",  
 "Médio incompleto",  
 "Médio completo",  
 "Superior incompleto",  
 "Superior completo"))))  
 # Criando variável idoso  
 pdad\_2018\_moradores <- pdad\_2018\_moradores %>%  
 dplyr::mutate(MelhorIdade=case\_when(idade\_calculada>=60~"1",  
 idade\_calculada<60~"0",  
 TRUE~NA\_character\_),)  
   
   
   
 ###### Unificar arquivos de dados a partir do identificador único da fixa  
 # Fazer o join das bases  
 pdad <- pdad\_2018\_moradores %>%  
 # Entrar com a função para left join  
 dplyr::left\_join(  
 # Informar a base que iremos unir, filtrando para colunas repetidas  
 pdad\_2018\_domicilios %>%  
 # Filtrar as colunas repetidas  
 dplyr::select(-c(A01ra)),  
 by=c("A01nFicha"="A01nFicha")) %>%   
 # Mudar a variável pos-estrato para o tipo character  
 dplyr::mutate(POS\_ESTRATO=as.character(POS\_ESTRATO)) %>%   
 dplyr::mutate(  
 RA\_nome=factor(case\_when(  
 A01ra==1~"Plano Piloto",  
 A01ra==2~"Gama",  
 A01ra==3~"Taguatinga",  
 A01ra==4~"Brazlândia",  
 A01ra==5~"Sobradinho",  
 A01ra==6~"Planaltina",  
 A01ra==7~"Paranoá",  
 A01ra==8~"Núcleo Bandeirante",  
 A01ra==9~"Ceilândia",  
 A01ra==10~"Guará",  
 A01ra==11~"Cruzeiro",  
 A01ra==12~"Samambaia",  
 A01ra==13~"Santa Maria",  
 A01ra==14~"São Sebastião",  
 A01ra==15~"Recanto das Emas",  
 A01ra==16~"Lago Sul",  
 A01ra==17~"Riacho Fundo",  
 A01ra==18~"Lago Norte",  
 A01ra==19~"Candangolândia",  
 A01ra==20~"Águas Claras",  
 A01ra==21~"Riacho Fundo II",  
 A01ra==22~"Sudoeste/Octogonal",  
 A01ra==23~"Varjão",  
 A01ra==24~"Park Way",  
 A01ra==25~"Scia/Estrutural",  
 A01ra==26~"Sobradinho II",  
 A01ra==27~"Jardim Botânico",  
 A01ra==28~"Itapoã",  
 A01ra==29~"SIA",  
 A01ra==30~"Vicente Pires",  
 A01ra==31~"Fercal")),  
   
 idade\_faixas=cut(idade\_calculada,  
 breaks = c(-Inf,seq(4,84,by=5),Inf),  
 labels = c("0 a 4 anos","5 a 9 anos",  
 "10 a 14 anos","15 a 19 anos",  
 "20 a 24 anos","25 a 29 anos",  
 "30 a 34 anos","35 a 39 anos",  
 "40 a 44 anos","45 a 49 anos",  
 "50 a 54 anos","55 a 59 anos",  
 "60 a 64 anos","65 a 69 anos",  
 "70 a 74 anos","75 a 79 anos",  
 "80 a 84 anos","Mais de 85 anos"),  
 ordered\_result = T),  
 # Criar variável de sexo  
 sexo=factor(case\_when(E03==1~"Masculino",  
 E03==2~"Feminino")),  
   
 Regiao=factor(case\_when(E142 %in% c(11:17)~"Norte",  
 E142 %in% c(21:29)~"Nordeste",  
 E142 %in% c(31:35)~"Sudeste",  
 E142 %in% c(41:43)~"Sul",  
 E142 %in% c(50:52)~"Centro-oeste",  
 E13==1~"DF")),  
   
 renda\_trab=case\_when(G16 %in% c(77777,88888,99999)~NA\_real\_,  
 TRUE~as.numeric(G16)),  
   
 onibus\_trab=case\_when(G141==1~"Sim",  
 G141==2~"Não",  
 G141==88~"Não sabe"),  
   
   
 transp\_trab=case\_when(G141==1|G144==1~"Sim",  
 G141==2&G144==2~"Não",  
 G141==88&G144==88~"Não sabe"),  
   
 tempo\_trab=case\_when(G15==1~"Até 15 minutos",  
 G15==2~"Mais de 0:15 até 0:30",  
 G15==3~"Mais de 0:30 até 0:45",  
 G15==4~"Mais de 0:45 até 1:00",  
 G15==5~"Mais de 1:00 até 1:30",  
 G15==6~"Mais de 1:30 até 1:45",  
 G15==7~"Mais de 1:45 até 2:00",  
 G15==8~"Mais de 2:00",  
 G15==88~"Não sabe"),  
   
 tempo\_trab\_c=case\_when(G15==1~7.5,  
 G15==2~22.5,  
 G15==3~37.5,  
 G15==4~52.5,  
 G15==5~75,  
 G15==6~97.5,  
 G15==7~112.5,  
 G15==8~120),  
   
 automoveis=case\_when(C011==88888~NA\_real\_,  
 TRUE~as.numeric(C011)),  
   
 tempo\_afazeres=case\_when(G18 %in% c(88888,99999)~NA\_real\_,  
 TRUE~as.numeric(G18)),  
   
 idoso=case\_when(idade\_calculada>=60~1,  
 TRUE~0),  
   
 pessoa\_dorm=A01nPessoas/B12,  
   
 internet=factor(case\_when(C041==1|C042==1|C043==1|C044==1~"Sim",  
 C041==88&C042==88&C043==88&C044==88~NA\_character\_,  
 TRUE~"Não")),  
   
 set\_educacao=factor(case\_when(G06==16~"Sim",  
 G06 %in% c(1:15,17:21)~"Não")),  
   
 horas\_trab=case\_when(G17%in%c(88888,99999)~NA\_real\_,  
 TRUE~as.numeric(G17)),  
   
 tempo\_uso=horas\_trab+(tempo\_trab\_c/60\*10)+tempo\_afazeres,  
   
 freq\_escola=factor(case\_when(F02==1~"Pública",  
 F02==2~"Privada",  
 TRUE~"Não estuda")),  
   
 nivel\_escola=factor(case\_when(F07%in%c(1,2)~"Creche/Educação Infantil",  
 F07==4~"Ensino Fundamental",  
 F07%in%c(5,6)~"Ensino Médio",  
 F07 %in% c(3,7,8)~"AJA e EJA",  
 F07 %in% c(9:12)~"Ensino Superior")),  
   
 local\_estudo=factor(case\_when(  
 F04==1~"Plano Piloto",  
 F04==2~"Gama",  
 F04==3~"Taguatinga",  
 F04==4~"Brazlândia",  
 F04==5~"Sobradinho",  
 F04==6~"Planaltina",  
 F04==7~"Paranoá",  
 F04==8~"Núcleo Bandeirante",  
 F04==9~"Ceilândia",  
 F04==10~"Guará",  
 F04==11~"Cruzeiro",  
 F04==12~"Samambaia",  
 F04==13~"Santa Maria",  
 F04==14~"São Sebastião",  
 F04==15~"Recanto das Emas",  
 F04==16~"Lago Sul",  
 F04==17~"Riacho Fundo",  
 F04==18~"Lago Norte",  
 F04==19~"Candangolândia",  
 F04==20~"Águas Claras",  
 F04==21~"Riacho Fundo II",  
 F04==22~"Sudoeste/Octogonal",  
 F04==23~"Varjão",  
 F04==24~"Park Way",  
 F04==25~"Scia/Estrutural",  
 F04==26~"Sobradinho II",  
 F04==27~"Jardim Botânico",  
 F04==28~"Itapoã",  
 F04==29~"SIA",  
 F04==30~"Vicente Pires",  
 F04==31~"Fercal",  
 F04 %in% 32:45~"Fora do DF")),  
   
 desloc\_escola=factor(case\_when(F05==1~"Ônibus",  
 F05==2~"Transporte escolar público",  
 F05==3~"Transporte escolar privado",  
 F05==4~"Automóvel",  
 F05==5~"Utilitário",  
 F05==6~"Metrô",  
 F05==7~"Motocicleta",  
 F05==8~"Bicicleta",  
 F05==9~"A pé",  
 F05==10~"Outros")),  
   
 transp\_escola=factor(case\_when(F05==1|F05==2|F05==3|F05==6~"Sim",  
 TRUE~"Não")),  
   
 pos\_dom=case\_when(E02==1~"Responsável",  
 E02 %in% c(2,3)~"Cônjuge",  
 TRUE~"Outro"),  
   
 crianca\_estuda=case\_when(idade\_calculada<=12&F02%in%c(1,2)~1,  
 TRUE~0),  
   
 count=1) %>%   
   
 dplyr::group\_by(A01nFicha) %>%   
 dplyr::mutate(idoso=sum(idoso),  
 crianca\_estuda=sum(crianca\_estuda)) %>%   
 dplyr::ungroup()  
   
   
 # Armazenar informação em um objeto  
 renda\_domiciliar <- pdad\_2018\_moradores %>%  
 # Vamos mudar para ausente os valores das variáveis G16,G19,G201 até G204  
 # com códigos 77777 ou 88888.  
 # Vamos também mudar para 0 quando os valores que não se aplicarem  
 # ou não forem observados rendimentos  
 dplyr::mutate\_at(vars(G16,G19,G201:G204), # Variáveis a serem alteradas  
 # Função a ser aplicada  
 list(M=~case\_when(. %in% c(77777,88888)~NA\_real\_,  
 . %in% c(66666,99999)~0,  
 TRUE~as.numeric(.)))) %>%  
 # Selecionar apenas as variáveis de interesse  
 dplyr::select(A01nFicha,E02,G16,G19,G201:G204,G16\_M:G204\_M) %>%  
 # Somar as variáveis modificadas para construir a renda individual  
 dplyr::mutate(renda\_individual=rowSums(.[,c("G16\_M","G19\_M",  
 "G201\_M","G202\_M",  
 "G203\_M","G204\_M")],na.rm = F)) %>%  
 # Desconsiderar os empregados domesticos moradores e seus parentes  
 dplyr::filter(!E02 %in% c(16,17,18)) %>%  
 # Agrupar por domicílio  
 dplyr::group\_by(A01nFicha) %>%  
 # Somar os valores por domicílios  
 dplyr::summarise(renda\_dom=sum(renda\_individual, na.rm = F),  
 # Construir o número de pessoas no domicílio, por esse critério de rendiment0  
 pessoas=n(),  
 # Calcular a renda domiciliar per capita  
 renda\_dom\_pc=renda\_dom/pessoas)

## `summarise()` ungrouping output (override with `.groups` argument)

# Juntar as bases  
 pdad <- pdad %>% dplyr::left\_join(renda\_domiciliar)

## Joining, by = "A01nFicha"

###### Criar o desenho inicial da pesquisa  
 #Defenir uma semente para reprodutibilidade  
 set.seed(8888)  
   
 # Declarar o desenho incial  
 sample.pdad <-   
 survey::svydesign(id = ~A01nFicha, # Identificador único da unidade amostrada  
 strata = ~A01setor, # Identificação do estrato  
 weights = ~PESO\_PRE, # Probabilidade da unidade ser sorteada  
 nest=TRUE, # Parâmetro de tratamento para dos IDs dos estratos  
 data=pdad # Declarar a base a ser utilizada  
 )  
 # Criar um objeto para pós estrato  
 post.pop <- pdad %>%  
 dplyr::group\_by(POS\_ESTRATO) %>% # Agrupar por pos-estrato  
 dplyr::summarise(Freq=max(POP\_AJUSTADA\_PROJ)) # Capturar o total da população

## `summarise()` ungrouping output (override with `.groups` argument)

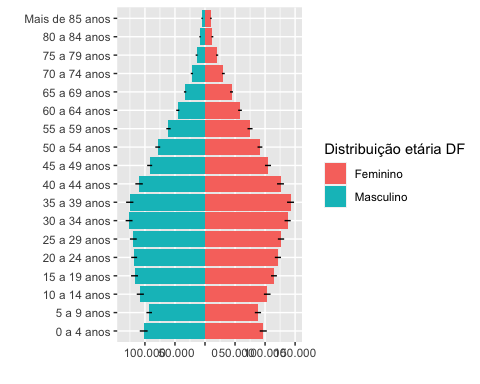
# Declarar o objeto de pós-estrato  
 # Estamos dizendo nesse passo qual é a população alvo para cada  
 # pós-estrato considerado  
 sample.pdad <- survey::postStratify(sample.pdad,~POS\_ESTRATO,post.pop)  
   
 # Criar objeto para calcular os erros por bootstrap (Rao and Wu’s(n − 1) bootstrap)  
 # J. N. K. Rao and C. F. J. Wu - Journal of the American Statistical Association  
 # Vol. 83, No. 401 (Mar., 1988), pp. 231-241  
 amostra <- survey::as.svrepdesign(sample.pdad, type = "subbootstrap")  
   
 # Ajustar estratos com apenas uma UPA (adjust=centered)  
 options( survey.lonely.psu = "adjust")  
   
 # Ajustar objeto de amostra, para uso com o pacote srvyr  
 amostra <- srvyr::as\_survey(amostra)  
   
   
 ###### Criar um objeto com as variáveis para as próximas questões  
 sm <- 954  
 vars\_relatorio <- amostra %>%  
 # Criar variável de sexo  
 srvyr::mutate(sexo=case\_when(E03==1~"Masculino",  
 E03==2~"Feminino"),  
 # Criar variável de faixas de idade  
 idade\_faixas=cut(idade\_calculada,  
 breaks = c(-Inf,seq(4,84,by=5),Inf),  
 labels = c("0 a 4 anos","5 a 9 anos",  
 "10 a 14 anos","15 a 19 anos",  
 "20 a 24 anos","25 a 29 anos",  
 "30 a 34 anos","35 a 39 anos",  
 "40 a 44 anos","45 a 49 anos",  
 "50 a 54 anos","55 a 59 anos",  
 "60 a 64 anos","65 a 69 anos",  
 "70 a 74 anos","75 a 79 anos",  
 "80 a 84 anos","Mais de 85 anos"),  
 ordered\_result = T),  
 # Criar variável para as RAs  
 RA=factor(A01ra,  
 levels=1:31,  
 labels=c('Plano Piloto',   
 'Gama',  
 'Taguatinga',  
 'Brazlândia',  
 'Sobradinho',  
 'Planaltina',  
 'Paranoá',  
 'Núcleo Bandeirante',  
 'Ceilândia',  
 'Guará',  
 'Cruzeiro',  
 'Samambaia',  
 'Santa Maria',  
 'São Sebastião',  
 'Recanto das Emas',  
 'Lago Sul',  
 'Riacho Fundo',  
 'Lago Norte',  
 'Candangolândia',  
 'Águas Claras',  
 'Riacho Fundo II',  
 'Sudoeste/Octogonal',  
 'Varjão',  
 'Park Way',  
 'SCIA-Estrutural',  
 'Sobradinho II',  
 'Jardim Botânico',  
 'Itapoã',  
 'SIA',  
 'Vicente Pires',  
 'Fercal'))) %>%  
 # Transformar em fator variáveis do tipo character  
 srvyr::mutate\_if(is.character,list(~factor(.))) %>%  
 # Selecionar as variáveis criadas e algumas variáveis auxiliares  
 srvyr::select(RA,E02,idade\_calculada,G05,sexo,idade\_faixas)  
  
   
########Questões   
   
##### 1.1 Apresente um perfil da RA X, Plano Piloto e do Distrito Federal, estimando as seguintes variáveis  
  
###### i) População total Plano Piloto e Samambaia  
 amostra %>%  
 # Filtrar Plano Piloto e Samambaia  
 #srvyr::filter(A01ra==1 | A01ra==12) %>%  
 # Ajustar nome das variáveis  
 srvyr::mutate(A01ra=factor(case\_when(A01ra==1~"Plano Piloto",  
 A01ra==12~"Samambaia",  
 TRUE~"Outras"))) %>%  
 # Agrupar por cidade  
 srvyr::group\_by(A01ra) %>%  
 # Calcular o total e o Percentual da população, com seu intervalo de confiança  
 srvyr::summarise("População Total"=survey\_total(vartype = "ci"),  
 # Calcular o percentual da população  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 3 x 7  
## A01ra `População Total` `População Total\_low` `População Total\_upp`  
## <fct> <dbl> <dbl> <dbl>  
## 1 Outras 2427635 2402660. 2452610.  
## 2 Plano Piloto 221326 214206. 228446.  
## 3 Samambaia 232893 225034. 240752.  
## pct pct\_low pct\_upp  
## <dbl> <dbl> <dbl>  
## 1 0.842 0.839 0.846   
## 2 0.0768 0.0745 0.0791  
## 3 0.0808 0.0781 0.0835

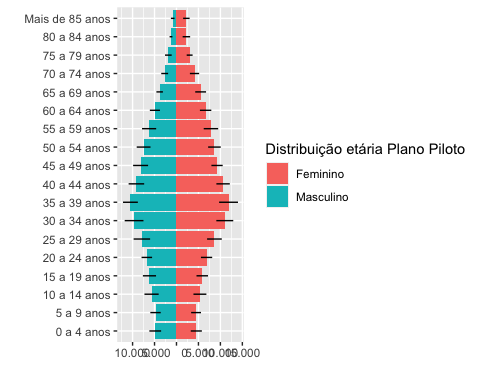
###### i) População total DF  
 amostra %>%  
 # Filtrar População Total  
 srvyr::filter(A01ra >= 1 ) %>%  
 # Criar contador  
 srvyr::mutate(count=1) %>%  
 # Calcular o total  
 srvyr::summarise("População Total"=survey\_total(count, vartype = "ci"))

## # A tibble: 1 x 3  
## `População Total` `População Total\_low` `População Total\_upp`  
## <dbl> <dbl> <dbl>  
## 1 2881854 2855396. 2908312.

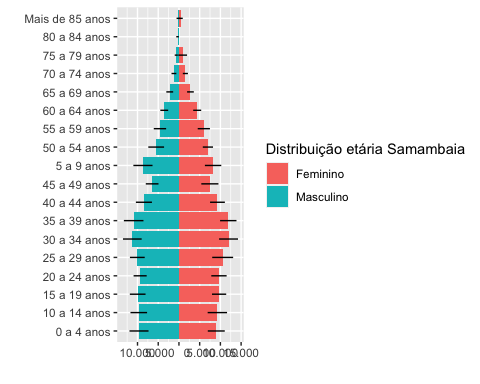
##### ii) Distribuição etária da população (faça uma pirâmide etária,   
##### separando homens e mulheres, com classes variando de 5 em 5 anos de 0-4; 5-9; 10-14...)  
  
### Distribuição etária DF  
 piramide <-  
 vars\_relatorio %>%  
 # Agrupar por faixas de idade e sexo  
 srvyr::group\_by(idade\_faixas,sexo) %>%  
 # Calcular os totais  
 srvyr::summarise(n=survey\_total(na.rm = T, vartype = "ci"))  
   
 # Fazer o gráfico com a pirâmide  
 piramide\_grafico <- piramide %>%  
 # Construir um plot com as idades no eixo x, as quantidades no eixo y,  
 # preenchimento com a variável sexo, e os intervalos de confiança  
 # inferiores e superiores  
 ggplot(aes(x=idade\_faixas,y=n, fill=sexo, ymin=n\_low,ymax=n\_upp))+  
 # Fazer o gráfico de barras para o sexo Feminino  
 geom\_bar(data = dplyr::filter(piramide, sexo == "Feminino"),  
 stat = "identity") +  
 # Fazer o gráfico de barras para o sexo Masculino  
 geom\_bar(data = dplyr::filter(piramide, sexo == "Masculino"),  
 stat = "identity",  
 position = "identity",  
 # Negativar os valores para espelhar no eixo  
 mapping = aes(y = -n))+  
 # Plotar os erros para o sexo Masculino, negativando os valores para espelhar o eixo  
 geom\_errorbar(data = dplyr::filter(piramide, sexo == "Masculino"),  
 mapping = aes(ymin = -n\_low,ymax=-n\_upp),  
 width=0,  
 color="black")+  
 # Plotar os erros para o sexo Feminino  
 geom\_errorbar(data = dplyr::filter(piramide, sexo == "Feminino"),  
 width=0,  
 color="black")+  
 # Inverter os eixos, fazendo com que o gráfico de colunas verticais fique  
 # horizontal  
 coord\_flip() +   
 # Ajustar as configurações de escala  
 scale\_y\_continuous(labels = function(x) format(abs(x),   
 big.mark = ".",  
 scientific = FALSE,  
 decimal.mark=",")) +  
 # Suprimir os nomes dos eixos  
 labs(x="",y="") +  
 # Nome do gráfico  
 scale\_fill\_discrete(name = "Distribuição etária DF")  
 # Plotar gráfico  
 piramide\_grafico



### Distribuição etária Plano Piloto  
 piramide <-  
 vars\_relatorio %>%  
 srvyr::filter(RA == "Plano Piloto" ) %>%  
 # Agrupar por faixas de idade e sexo  
 srvyr::group\_by(idade\_faixas,sexo) %>%  
 # Calcular os totais  
 srvyr::summarise(n=survey\_total(na.rm = T, vartype = "ci"))  
   
   
 # Fazer o gráfico com a pirâmide  
 piramide\_grafico <- piramide %>%  
 # Construir um plot com as idades no eixo x, as quantidades no eixo y,  
 # preenchimento com a variável sexo, e os intervalos de confiança  
 # inferiores e superiores  
 ggplot(aes(x=idade\_faixas,y=n, fill=sexo, ymin=n\_low,ymax=n\_upp))+  
 # Fazer o gráfico de barras para o sexo Feminino  
 geom\_bar(data = dplyr::filter(piramide, sexo == "Feminino"),  
 stat = "identity") +  
 # Fazer o gráfico de barras para o sexo Masculino  
 geom\_bar(data = dplyr::filter(piramide, sexo == "Masculino"),  
 stat = "identity",  
 position = "identity",  
 # Negativar os valores para espelhar no eixo  
 mapping = aes(y = -n))+  
 # Plotar os erros para o sexo Masculino, negativando os valores para espelhar o eixo  
 geom\_errorbar(data = dplyr::filter(piramide, sexo == "Masculino"),  
 mapping = aes(ymin = -n\_low,ymax=-n\_upp),  
 width=0,  
 color="black")+  
 # Plotar os erros para o sexo Feminino  
 geom\_errorbar(data = dplyr::filter(piramide, sexo == "Feminino"),  
 width=0,  
 color="black")+  
 # Inverter os eixos, fazendo com que o gráfico de colunas verticais fique  
 # horizontal  
 coord\_flip() +   
 # Ajustar as configurações de escala  
 scale\_y\_continuous(labels = function(x) format(abs(x),   
 big.mark = ".",  
 scientific = FALSE,  
 decimal.mark=",")) +  
 # Suprimir os nomes dos eixos  
 labs(x="",y="") +  
 # Nome do gráfico  
 scale\_fill\_discrete(name = "Distribuição etária Plano Piloto")  
 # Plotar gráfico  
 piramide\_grafico



### Distribuição etária Samambaia  
 piramide <-  
 vars\_relatorio %>%  
 srvyr::filter(RA == "Samambaia" ) %>%  
 # Agrupar por faixas de idade e sexo  
 srvyr::group\_by(idade\_faixas,sexo) %>%  
 # Calcular os totais  
 srvyr::summarise(n=survey\_total(na.rm = T, vartype = "ci"))  
   
   
 # Fazer o gráfico com a pirâmide  
 piramide\_grafico <- piramide %>%  
 # Construir um plot com as idades no eixo x, as quantidades no eixo y,  
 # preenchimento com a variável sexo, e os intervalos de confiança  
 # inferiores e superiores  
 ggplot(aes(x=idade\_faixas,y=n, fill=sexo, ymin=n\_low,ymax=n\_upp))+  
 # Fazer o gráfico de barras para o sexo Feminino  
 geom\_bar(data = dplyr::filter(piramide, sexo == "Feminino"),  
 stat = "identity") +  
 # Fazer o gráfico de barras para o sexo Masculino  
 geom\_bar(data = dplyr::filter(piramide, sexo == "Masculino"),  
 stat = "identity",  
 position = "identity",  
 # Negativar os valores para espelhar no eixo  
 mapping = aes(y = -n))+  
 # Plotar os erros para o sexo Masculino, negativando os valores para espelhar o eixo  
 geom\_errorbar(data = dplyr::filter(piramide, sexo == "Masculino"),  
 mapping = aes(ymin = -n\_low,ymax=-n\_upp),  
 width=0,  
 color="black")+  
 # Plotar os erros para o sexo Feminino  
 geom\_errorbar(data = dplyr::filter(piramide, sexo == "Feminino"),  
 width=0,  
 color="black")+  
 # Inverter os eixos, fazendo com que o gráfico de colunas verticais fique  
 # horizontal  
 coord\_flip() +   
 # Ajustar as configurações de escala  
 scale\_y\_continuous(labels = function(x) format(abs(x),   
 big.mark = ".",  
 scientific = FALSE,  
 decimal.mark=",")) +  
 # Suprimir os nomes dos eixos  
 labs(x="",y="") +  
 # Nome do gráfico  
 scale\_fill\_discrete(name = "Distribuição etária Samambaia")  
 # Plotar gráfico  
 piramide\_grafico



###### iii) Número de domicílios   
 #Plano Piloto e Samambaia  
 amostra %>%  
 # Filtrar Proprietários das casas  
 srvyr::filter(E02==1) %>%  
 # Ajustar nome das variáveis  
 srvyr::mutate(A01ra=factor(case\_when(A01ra==1~"Plano Piloto",  
 A01ra==12~"Samambaia",  
 TRUE~"Outras"))) %>%  
 # Agrupar por cidade  
 srvyr::group\_by(A01ra) %>%  
 # Calcular o total e o Percentual da população, com seu intervalo de confiança  
 srvyr::summarise("População Total"=survey\_total(vartype = "ci"),  
 # Calcular o percentual da população  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 3 x 7  
## A01ra `População Total` `População Total\_low` `População Total\_upp`  
## <fct> <dbl> <dbl> <dbl>  
## 1 Outras 729529. 726663. 732395.  
## 2 Plano Piloto 85176. 84052. 86300.  
## 3 Samambaia 68804. 67760. 69848.  
## pct pct\_low pct\_upp  
## <dbl> <dbl> <dbl>  
## 1 0.826 0.824 0.827   
## 2 0.0964 0.0952 0.0976  
## 3 0.0779 0.0767 0.0791

#Distrito Federal   
 amostra %>%  
 # Filtrar População DF pelo proprietário da casa  
 srvyr::filter(E02==1) %>%  
 # Criar contador  
 srvyr::mutate(count=1) %>%  
 # Calcular o total  
 srvyr::summarise("População Total"=survey\_total(count, vartype = "ci"))

## # A tibble: 1 x 3  
## `População Total` `População Total\_low` `População Total\_upp`  
## <dbl> <dbl> <dbl>  
## 1 883509. 880580. 886439.

###### iv) Naturalidade dos residentes (Região de Nascimento use as grandes regiões   
##### Norte, Nordeste, Centro-Oeste sem DF, Sudeste e Sul, crie uma categoria especial nascido DF.  
 amostra %>%  
 # Ajustar nome das variáveis  
 srvyr::mutate(E142=factor(case\_when(E142 %in% c(11:17)~"Norte",  
 E142 %in% c(21:29)~"Nordeste",  
 E142 %in% c(31:35)~"Sudeste",  
 E142 %in% c(41:43)~"Sul",  
 E142 %in% c(50:52)~"Centro-oeste",  
 E142==88~"Z\_Não Sabe",  
 E142==0~"Z\_Outro País",  
 E13==1~"DF"))) %>%  
 # Agrupar por região  
 srvyr::group\_by(E142) %>%  
 # Calcular o total e o Percentual, com seu intervalo de confiança  
 srvyr::summarise("Naturalidade"=survey\_total(vartype = "ci"),  
 # Calcular o percentual   
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 8 x 7  
## E142 Naturalidade Naturalidade\_low Naturalidade\_upp pct pct\_low  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Centro-oeste 168008. 160107. 175910. 0.0583 0.0556   
## 2 DF 1594817. 1570209. 1619424. 0.553 0.548   
## 3 Nordeste 644841. 632090. 657592. 0.224 0.219   
## 4 Norte 59755. 54160. 65349. 0.0207 0.0188   
## 5 Sudeste 337804. 326358. 349251. 0.117 0.113   
## 6 Sul 42106. 38377. 45835. 0.0146 0.0133   
## 7 Z\_Não Sabe 25220. 22193. 28247. 0.00875 0.00769  
## 8 Z\_Outro País 9303. 7466. 11140. 0.00323 0.00259  
## pct\_upp  
## <dbl>  
## 1 0.0610   
## 2 0.559   
## 3 0.228   
## 4 0.0227   
## 5 0.121   
## 6 0.0159   
## 7 0.00981  
## 8 0.00387

##### 1.2 Calcule (quando possível) a média, mediana, variância, desvio padrão,   
 # para as seguintes variáveis com os seguintes filtros Distrito Federal   
 # como um todo, Plano Piloto e RA X:  
   
###### i) Renda domiciliar per capita (calcule também Quantis Q1, Q3 e o   
 # percentil 99, ou seja, o valor do 1% mais rico daquela RA)   
   
 #Calculos para o DF  
 amostra %>%   
 srvyr::filter(E02==1) %>%   
 srvyr::summarise("Renda domiciliar per capita DF"=survey\_mean(renda\_dom\_pc,na.rm=T),  
 "Variância DF"=survey\_var(renda\_dom\_pc,na.rm=T),  
 "Mediana DF"=survey\_median(renda\_dom\_pc,na.rm=T),  
 "Desvio Padrão DF"=survey\_sd(renda\_dom\_pc,na.rm=T),  
 "Renda Q1 (<25%) DF"=survey\_quantile(renda\_dom\_pc, 0.25, na.rm=TRUE),  
 "Renda Q3 (<75%) DF"=survey\_quantile(renda\_dom\_pc, 0.75, na.rm=TRUE),  
 "Renda Q99 (<99%) DF"=survey\_quantile(renda\_dom\_pc, 0.99, na.rm=TRUE))

## # A tibble: 1 x 13  
## `Renda domiciliar per capita DF` `Renda domiciliar per capita DF\_se`  
## <dbl> <dbl>  
## 1 2476. 39.4  
## `Variância DF` `Variância DF\_se` `Mediana DF` `Mediana DF\_se`  
## <dbl> <dbl> <dbl> <dbl>  
## 1 18333169. 2273409. 1075 15.9  
## `Desvio Padrão DF` `Renda Q1 (<25%) DF\_q25` `Renda Q1 (<25%) DF\_q25\_se`  
## <dbl> <dbl> <dbl>  
## 1 4282. 500 4.98  
## `Renda Q3 (<75%) DF\_q75` `Renda Q3 (<75%) DF\_q75\_se` `Renda Q99 (<99%) DF\_q99`  
## <dbl> <dbl> <dbl>  
## 1 2900 61.8 18667.  
## `Renda Q99 (<99%) DF\_q99\_se`  
## <dbl>  
## 1 746.

# Calculos para o Plano Piloto  
 amostra %>%   
 srvyr::filter(E02==1&A01ra==1) %>%   
 srvyr::summarise("Renda domiciliar per capita Plano"=survey\_mean(renda\_dom\_pc,na.rm=T),  
 "Variância Plano"=survey\_var(renda\_dom\_pc,na.rm=T),  
 "Mediana Plano"=survey\_median(renda\_dom\_pc,na.rm=T),  
 "Desvio Padrão Plano"=survey\_sd(renda\_dom\_pc,na.rm=T),  
 "Renda Q1 (<25%) Plano"=survey\_quantile(renda\_dom\_pc, 0.25, na.rm=TRUE),  
 "Renda Q3 (<75%) Plano"=survey\_quantile(renda\_dom\_pc, 0.75, na.rm=TRUE),  
 "Renda Q99 (<99%) Plano"=survey\_quantile(renda\_dom\_pc, 0.99, na.rm=TRUE))

## # A tibble: 1 x 13  
## `Renda domiciliar per capita Plano` `Renda domiciliar per capita Plano\_se`  
## <dbl> <dbl>  
## 1 6745. 279.  
## `Variância Plano` `Variância Plano\_se` `Mediana Plano` `Mediana Plano\_se`  
## <dbl> <dbl> <dbl> <dbl>  
## 1 79829337. 20655972. 4754. 205.  
## `Desvio Padrão Plano` `Renda Q1 (<25%) Plano\_q25`  
## <dbl> <dbl>  
## 1 8935. 2501.  
## `Renda Q1 (<25%) Plano\_q25\_se` `Renda Q3 (<75%) Plano\_q75`  
## <dbl> <dbl>  
## 1 99.5 8653.  
## `Renda Q3 (<75%) Plano\_q75\_se` `Renda Q99 (<99%) Plano\_q99`  
## <dbl> <dbl>  
## 1 311. 30000  
## `Renda Q99 (<99%) Plano\_q99\_se`  
## <dbl>  
## 1 3824.

# Calculos para Samambaia  
 amostra %>%   
 srvyr::filter(E02==1&A01ra==12) %>%   
 srvyr::summarise("Renda domiciliar per capita Samambaia"=survey\_mean(renda\_dom\_pc,na.rm=T),  
 "Variância Samambaia"=survey\_var(renda\_dom\_pc,na.rm=T),  
 "Mediana Samambaia"=survey\_median(renda\_dom\_pc,na.rm=T),  
 "Desvio Padrão Samambaia"=survey\_sd(renda\_dom\_pc,na.rm=T),  
 "Renda Q1 (<25%) Samambaia"=survey\_quantile(renda\_dom\_pc, 0.25, na.rm=TRUE),  
 "Renda Q3 (<75%) Samambaia"=survey\_quantile(renda\_dom\_pc, 0.75, na.rm=TRUE),  
 "Renda Q99 (<99%) Samambaia"=survey\_quantile(renda\_dom\_pc, 0.99, na.rm=TRUE))

## # A tibble: 1 x 13  
## `Renda domiciliar per capita Samambaia`  
## <dbl>  
## 1 975.  
## `Renda domiciliar per capita Samambaia\_se` `Variância Samambaia`  
## <dbl> <dbl>  
## 1 82.0 1938712.  
## `Variância Samambaia\_se` `Mediana Samambaia` `Mediana Samambaia\_se`  
## <dbl> <dbl> <dbl>  
## 1 733261. 623. 55.3  
## `Desvio Padrão Samambaia` `Renda Q1 (<25%) Samambaia\_q25`  
## <dbl> <dbl>  
## 1 1392. 284.  
## `Renda Q1 (<25%) Samambaia\_q25\_se` `Renda Q3 (<75%) Samambaia\_q75`  
## <dbl> <dbl>  
## 1 49.9 1090  
## `Renda Q3 (<75%) Samambaia\_q75\_se` `Renda Q99 (<99%) Samambaia\_q99`  
## <dbl> <dbl>  
## 1 54.2 6719.  
## `Renda Q99 (<99%) Samambaia\_q99\_se`  
## <dbl>  
## 1 2496.

###### ii) Renda do trabalho principal (calcule também Quantis Q1, Q3 e o   
 # percentil 99, ou seja, o valor do 1% mais rico daquela RA).Agora os   
 # cálculos apenas para a Renda Primária NÃO PER CAPITA  
   
 # Calculos para o DF  
 amostra %>%   
 mutate(renda\_prim=case\_when(G16 == 77777 ~ NA\_real\_,  
 G16 == 88888 ~ NA\_real\_,  
 G16 == 99999 ~ NA\_real\_,  
 TRUE ~as.numeric(G16))) %>%  
 srvyr::summarise("Renda Média Trabalho Principal DF"=survey\_mean(renda\_prim,na.rm=T),  
 "Variância DF"=survey\_var(renda\_prim,na.rm=T),  
 "Mediana DF"=survey\_median(renda\_prim,na.rm=T),  
 "Desvio Padrão DF"=survey\_sd(renda\_prim,na.rm=T),  
 "Renda Q1 (<25%) DF"=survey\_quantile(renda\_prim, 0.25, na.rm=TRUE),  
 "Renda Q3 (<75%) DF"=survey\_quantile(renda\_prim, 0.75, na.rm=TRUE),  
 "Renda Q99 (<99%) DF"=survey\_quantile(renda\_prim, 0.99, na.rm=TRUE))

## # A tibble: 1 x 13  
## `Renda Média Trabalho Principal DF` `Renda Média Trabalho Principal DF\_se`  
## <dbl> <dbl>  
## 1 3446. 35.5  
## `Variância DF` `Variância DF\_se` `Mediana DF` `Mediana DF\_se`  
## <dbl> <dbl> <dbl> <dbl>  
## 1 19424581. 743986. 1800 49.8  
## `Desvio Padrão DF` `Renda Q1 (<25%) DF\_q25` `Renda Q1 (<25%) DF\_q25\_se`  
## <dbl> <dbl> <dbl>  
## 1 4407. 1020 19.9  
## `Renda Q3 (<75%) DF\_q75` `Renda Q3 (<75%) DF\_q75\_se` `Renda Q99 (<99%) DF\_q99`  
## <dbl> <dbl> <dbl>  
## 1 4000 124. 20000  
## `Renda Q99 (<99%) DF\_q99\_se`  
## <dbl>  
## 1 498.

# Calculos para o Plano Piloto  
 amostra %>% filter(A01ra==1) %>%   
 mutate(renda\_prim=case\_when(G16 == 77777 ~ NA\_real\_,  
 G16 == 88888 ~ NA\_real\_,  
 G16 == 99999 ~ NA\_real\_,  
 TRUE ~as.numeric(G16))) %>%  
 srvyr::summarise("Renda Média Trabalho Principal Plano"=survey\_mean(renda\_prim,na.rm=T),  
 "Variância Plano"=survey\_var(renda\_prim,na.rm=T),  
 "Mediana Plano"=survey\_median(renda\_prim,na.rm=T),  
 "Desvio Padrão Plano"=survey\_sd(renda\_prim,na.rm=T),  
 "Renda Q1 (<25%) Plano"=survey\_quantile(renda\_prim, 0.25, na.rm=TRUE),  
 "Renda Q3 (<75%) Plano"=survey\_quantile(renda\_prim, 0.75, na.rm=TRUE),  
 "Renda Q99 (<99%) Plano"=survey\_quantile(renda\_prim, 0.99, na.rm=TRUE))

## # A tibble: 1 x 13  
## `Renda Média Trabalho Principal Plano`  
## <dbl>  
## 1 7308.  
## `Renda Média Trabalho Principal Plano\_se` `Variância Plano`  
## <dbl> <dbl>  
## 1 186. 41509609.  
## `Variância Plano\_se` `Mediana Plano` `Mediana Plano\_se` `Desvio Padrão Plano`  
## <dbl> <dbl> <dbl> <dbl>  
## 1 3235948. 5202. 249. 6443.  
## `Renda Q1 (<25%) Plano\_q25` `Renda Q1 (<25%) Plano\_q25\_se`  
## <dbl> <dbl>  
## 1 2800 199.  
## `Renda Q3 (<75%) Plano\_q75` `Renda Q3 (<75%) Plano\_q75\_se`  
## <dbl> <dbl>  
## 1 10000 249.  
## `Renda Q99 (<99%) Plano\_q99` `Renda Q99 (<99%) Plano\_q99\_se`  
## <dbl> <dbl>  
## 1 30000 1766.

# Calculos para Samambaia  
 amostra %>% filter(A01ra==12) %>%   
 mutate(renda\_prim=case\_when(G16 == 77777 ~ NA\_real\_,  
 G16 == 88888 ~ NA\_real\_,  
 G16 == 99999 ~ NA\_real\_,  
 TRUE ~as.numeric(G16))) %>%  
 srvyr::summarise("Renda Média Trabalho Principal Samambaia"=survey\_mean(renda\_prim,na.rm=T),  
 "Variância Samambaia"=survey\_var(renda\_prim,na.rm=T),  
 "Mediana Samambaia"=survey\_median(renda\_prim,na.rm=T),  
 "Desvio Padrão Samambaia"=survey\_sd(renda\_prim,na.rm=T),  
 "Renda Q1 (<25%) Samambaia"=survey\_quantile(renda\_prim, 0.25, na.rm=TRUE),  
 "Renda Q3 (<75%) Samambaia"=survey\_quantile(renda\_prim, 0.75, na.rm=TRUE),  
 "Renda Q99 (<99%) Samambaia"=survey\_quantile(renda\_prim, 0.99, na.rm=TRUE))

## # A tibble: 1 x 13  
## `Renda Média Trabalho Principal Samambaia`  
## <dbl>  
## 1 1958.  
## `Renda Média Trabalho Principal Samambaia\_se` `Variância Samambaia`  
## <dbl> <dbl>  
## 1 132. 4685428.  
## `Variância Samambaia\_se` `Mediana Samambaia` `Mediana Samambaia\_se`  
## <dbl> <dbl> <dbl>  
## 1 1292284. 1200 44.8  
## `Desvio Padrão Samambaia` `Renda Q1 (<25%) Samambaia\_q25`  
## <dbl> <dbl>  
## 1 2165. 955.  
## `Renda Q1 (<25%) Samambaia\_q25\_se` `Renda Q3 (<75%) Samambaia\_q75`  
## <dbl> <dbl>  
## 1 11.4 2000  
## `Renda Q3 (<75%) Samambaia\_q75\_se` `Renda Q99 (<99%) Samambaia\_q99`  
## <dbl> <dbl>  
## 1 124. 11476.  
## `Renda Q99 (<99%) Samambaia\_q99\_se`  
## <dbl>  
## 1 2857.

###### iii) Escolaridade das pessoas de 25 anos ou mais.   
 # (note que é variável qualitativa ordenada)   
   
 # Calculos para o DF   
 amostra %>%   
 srvyr::filter(is.na(escolaridade)==F) %>%   
 srvyr::filter(idade\_calculada >= 25) %>%  
 srvyr::group\_by(escolaridade) %>%   
 srvyr::summarise(n=survey\_total(),  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 7 x 6  
## escolaridade n n\_se pct pct\_low pct\_upp  
## <ord> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Sem escolaridade 52031. 1991. 0.0291 0.0269 0.0313  
## 2 Fundamental incompleto 319525. 4063. 0.179 0.174 0.183   
## 3 Fundamental completo 72340. 1803. 0.0405 0.0384 0.0425  
## 4 Médio incompleto 92657. 1944. 0.0518 0.0496 0.0541  
## 5 Médio completo 524374. 5175. 0.293 0.288 0.298   
## 6 Superior incompleto 119985. 3241. 0.0671 0.0633 0.0709  
## 7 Superior completo 606932. 5063. 0.339 0.334 0.345

# Calculos para o Plano   
 amostra %>%   
 srvyr::filter(is.na(escolaridade)==F) %>%   
 srvyr::filter(idade\_calculada >= 25 & A01ra == 1) %>%  
 srvyr::group\_by(escolaridade) %>%   
 srvyr::summarise(n=survey\_total(),  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 7 x 6  
## escolaridade n n\_se pct pct\_low pct\_upp  
## <ord> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Sem escolaridade 1703. 372. 0.0103 0.00573 0.0148  
## 2 Fundamental incompleto 4620. 547. 0.0279 0.0213 0.0345  
## 3 Fundamental completo 1859. 365. 0.0112 0.00689 0.0156  
## 4 Médio incompleto 1885. 443. 0.0114 0.00605 0.0167  
## 5 Médio completo 20387. 1071. 0.123 0.111 0.135   
## 6 Superior incompleto 9491. 954. 0.0573 0.0456 0.0690  
## 7 Superior completo 125675. 2606. 0.759 0.738 0.780

# Calculos para a Samambaia   
 amostra %>%   
 srvyr::filter(is.na(escolaridade)==F) %>%   
 srvyr::filter(idade\_calculada >= 25 & A01ra == 12) %>%  
 srvyr::group\_by(escolaridade) %>%   
 srvyr::summarise(n=survey\_total(),  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 7 x 6  
## escolaridade n n\_se pct pct\_low pct\_upp  
## <ord> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Sem escolaridade 8597. 1005. 0.0638 0.0485 0.0791  
## 2 Fundamental incompleto 28361. 1876. 0.211 0.185 0.236   
## 3 Fundamental completo 6225. 738. 0.0462 0.0350 0.0574  
## 4 Médio incompleto 11269. 1170. 0.0836 0.0666 0.101   
## 5 Médio completo 48164. 2142. 0.357 0.328 0.387   
## 6 Superior incompleto 8604. 1131. 0.0639 0.0474 0.0803  
## 7 Superior completo 23510. 1406. 0.174 0.154 0.195

###### iv) Modo de transporte para o trabalho (apenas uma variável qualitativa)  
   
 # Como existem pessoas com mais de uma opção de transporte para o   
 # trabalho, optamos por agrupar as diferentes formas numa matrix   
 # de possibilidades.  
 amostra %>% # Cálculos para o DF  
 mutate(Onibus=factor(case\_when(G141 == 1 ~ "onibus",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Automovel=factor(case\_when(G142 == 1 ~ "automovel",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Utilitario=factor(case\_when(G143 == 1 ~ "utilitario",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Metro=factor(case\_when(G144 == 1 ~ "metro",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Moto=factor(case\_when(G145 == 1 ~ "moto",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Bike=factor(case\_when(G146 == 1 ~ "bike",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Andando=factor(case\_when(G147 == 1 ~ "andando",  
 TRUE ~ NA\_character\_))) %>%  
 group\_by(Onibus, Automovel,Utilitario, Metro, Moto, Bike, Andando) %>%  
 summarise("Quantidade DF"=survey\_total(na.rm=TRUE)) %>% print(n=100)

## # A tibble: 58 x 9  
## # Groups: Onibus, Automovel, Utilitario, Metro, Moto, Bike [36]  
## Onibus Automovel Utilitario Metro Moto Bike Andando `Quantidade DF`  
## <fct> <fct> <fct> <fct> <fct> <fct> <fct> <dbl>  
## 1 onibus automovel utilitario <NA> <NA> <NA> <NA> 619.   
## 2 onibus automovel <NA> metro moto <NA> andando 55.2  
## 3 onibus automovel <NA> metro moto <NA> <NA> 72.7  
## 4 onibus automovel <NA> metro <NA> bike <NA> 45.4  
## 5 onibus automovel <NA> metro <NA> <NA> andando 93.5  
## 6 onibus automovel <NA> metro <NA> <NA> <NA> 2132.   
## 7 onibus automovel <NA> <NA> moto bike andando 10.5  
## 8 onibus automovel <NA> <NA> moto <NA> <NA> 331.   
## 9 onibus automovel <NA> <NA> <NA> bike andando 30.1  
## 10 onibus automovel <NA> <NA> <NA> bike <NA> 599.   
## 11 onibus automovel <NA> <NA> <NA> <NA> andando 2087.   
## 12 onibus automovel <NA> <NA> <NA> <NA> <NA> 56341.   
## 13 onibus <NA> utilitario metro <NA> <NA> andando 78.5  
## 14 onibus <NA> utilitario metro <NA> <NA> <NA> 22.1  
## 15 onibus <NA> utilitario <NA> <NA> <NA> <NA> 797.   
## 16 onibus <NA> <NA> metro moto <NA> <NA> 109.   
## 17 onibus <NA> <NA> metro <NA> bike andando 84.4  
## 18 onibus <NA> <NA> metro <NA> bike <NA> 219.   
## 19 onibus <NA> <NA> metro <NA> <NA> andando 539.   
## 20 onibus <NA> <NA> metro <NA> <NA> <NA> 13601.   
## 21 onibus <NA> <NA> <NA> moto bike andando 29.9  
## 22 onibus <NA> <NA> <NA> moto bike <NA> 41.0  
## 23 onibus <NA> <NA> <NA> moto <NA> andando 60.3  
## 24 onibus <NA> <NA> <NA> moto <NA> <NA> 1354.   
## 25 onibus <NA> <NA> <NA> <NA> bike andando 318.   
## 26 onibus <NA> <NA> <NA> <NA> bike <NA> 2223.   
## 27 onibus <NA> <NA> <NA> <NA> <NA> andando 13131.   
## 28 onibus <NA> <NA> <NA> <NA> <NA> <NA> 387998.   
## 29 <NA> automovel utilitario metro <NA> <NA> <NA> 62.4  
## 30 <NA> automovel utilitario <NA> <NA> <NA> andando 80.1  
## 31 <NA> automovel utilitario <NA> <NA> <NA> <NA> 1273.   
## 32 <NA> automovel <NA> metro <NA> bike andando 29.3  
## 33 <NA> automovel <NA> metro <NA> bike <NA> 153.   
## 34 <NA> automovel <NA> metro <NA> <NA> andando 141.   
## 35 <NA> automovel <NA> metro <NA> <NA> <NA> 5665.   
## 36 <NA> automovel <NA> <NA> moto bike <NA> 64.3  
## 37 <NA> automovel <NA> <NA> moto <NA> andando 159.   
## 38 <NA> automovel <NA> <NA> moto <NA> <NA> 4211.   
## 39 <NA> automovel <NA> <NA> <NA> bike andando 387.   
## 40 <NA> automovel <NA> <NA> <NA> bike <NA> 2718.   
## 41 <NA> automovel <NA> <NA> <NA> <NA> andando 9155.   
## 42 <NA> automovel <NA> <NA> <NA> <NA> <NA> 507746.   
## 43 <NA> <NA> utilitario metro <NA> <NA> <NA> 224.   
## 44 <NA> <NA> utilitario <NA> moto bike <NA> 17.3  
## 45 <NA> <NA> utilitario <NA> <NA> bike <NA> 20.6  
## 46 <NA> <NA> utilitario <NA> <NA> <NA> andando 214.   
## 47 <NA> <NA> utilitario <NA> <NA> <NA> <NA> 4844.   
## 48 <NA> <NA> <NA> metro moto <NA> <NA> 177.   
## 49 <NA> <NA> <NA> metro <NA> bike <NA> 267.   
## 50 <NA> <NA> <NA> metro <NA> <NA> andando 160.   
## 51 <NA> <NA> <NA> metro <NA> <NA> <NA> 22173.   
## 52 <NA> <NA> <NA> <NA> moto bike <NA> 218.   
## 53 <NA> <NA> <NA> <NA> moto <NA> andando 226.   
## 54 <NA> <NA> <NA> <NA> moto <NA> <NA> 28463.   
## 55 <NA> <NA> <NA> <NA> <NA> bike andando 1903.   
## 56 <NA> <NA> <NA> <NA> <NA> bike <NA> 17422.   
## 57 <NA> <NA> <NA> <NA> <NA> <NA> andando 154606.   
## 58 <NA> <NA> <NA> <NA> <NA> <NA> <NA> 1636053.   
## `Quantidade DF\_se`  
## <dbl>  
## 1 221.   
## 2 61.1   
## 3 72.9   
## 4 47.2   
## 5 64.6   
## 6 405.   
## 7 9.78  
## 8 136.   
## 9 25.8   
## 10 239.   
## 11 353.   
## 12 2056.   
## 13 74.9   
## 14 17.9   
## 15 230.   
## 16 63.9   
## 17 94.2   
## 18 106.   
## 19 224.   
## 20 833.   
## 21 26.9   
## 22 37.7   
## 23 59.4   
## 24 336.   
## 25 102.   
## 26 341.   
## 27 935.   
## 28 4991.   
## 29 64.0   
## 30 83.3   
## 31 203.   
## 32 30.9   
## 33 139.   
## 34 96.6   
## 35 804.   
## 36 80.1   
## 37 73.6   
## 38 537.   
## 39 138.   
## 40 379.   
## 41 725.   
## 42 5191.   
## 43 137.   
## 44 17.3   
## 45 22.7   
## 46 110.   
## 47 620.   
## 48 138.   
## 49 107.   
## 50 113.   
## 51 1519.   
## 52 115.   
## 53 96.6   
## 54 1343.   
## 55 288.   
## 56 992.   
## 57 2950.   
## 58 10181.

amostra %>% # Cálculos para o Plano Piloto  
 filter(A01ra == 1) %>%   
 mutate(Onibus=factor(case\_when(G141 == 1 ~ "onibus",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Automovel=factor(case\_when(G142 == 1 ~ "automovel",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Utilitario=factor(case\_when(G143 == 1 ~ "utilitario",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Metro=factor(case\_when(G144 == 1 ~ "metro",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Moto=factor(case\_when(G145 == 1 ~ "moto",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Bike=factor(case\_when(G146 == 1 ~ "bike",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Andando=factor(case\_when(G147 == 1 ~ "andando",  
 TRUE ~ NA\_character\_))) %>%  
 group\_by(Onibus, Automovel,Utilitario, Metro, Moto, Bike, Andando) %>%  
 summarise("Qtd. Plano"=survey\_total(na.rm=TRUE)) %>% print(n=100)

## # A tibble: 26 x 9  
## # Groups: Onibus, Automovel, Utilitario, Metro, Moto, Bike [19]  
## Onibus Automovel Utilitario Metro Moto Bike Andando `Qtd. Plano`  
## <fct> <fct> <fct> <fct> <fct> <fct> <fct> <dbl>  
## 1 onibus automovel utilitario <NA> <NA> <NA> <NA> 64.0  
## 2 onibus automovel <NA> metro <NA> <NA> <NA> 368.   
## 3 onibus automovel <NA> <NA> <NA> bike <NA> 184.   
## 4 onibus automovel <NA> <NA> <NA> <NA> andando 433.   
## 5 onibus automovel <NA> <NA> <NA> <NA> <NA> 3675.   
## 6 onibus <NA> utilitario <NA> <NA> <NA> <NA> 263.   
## 7 onibus <NA> <NA> metro <NA> <NA> andando 44.5  
## 8 onibus <NA> <NA> metro <NA> <NA> <NA> 295.   
## 9 onibus <NA> <NA> <NA> <NA> bike <NA> 82.2  
## 10 onibus <NA> <NA> <NA> <NA> <NA> andando 263.   
## 11 onibus <NA> <NA> <NA> <NA> <NA> <NA> 12824.   
## 12 <NA> automovel utilitario <NA> <NA> <NA> <NA> 213.   
## 13 <NA> automovel <NA> metro <NA> <NA> <NA> 69.1  
## 14 <NA> automovel <NA> <NA> moto <NA> <NA> 610.   
## 15 <NA> automovel <NA> <NA> <NA> bike andando 42.0  
## 16 <NA> automovel <NA> <NA> <NA> bike <NA> 636.   
## 17 <NA> automovel <NA> <NA> <NA> <NA> andando 1323.   
## 18 <NA> automovel <NA> <NA> <NA> <NA> <NA> 74660.   
## 19 <NA> <NA> utilitario metro <NA> <NA> <NA> 59.3  
## 20 <NA> <NA> utilitario <NA> <NA> <NA> <NA> 631.   
## 21 <NA> <NA> <NA> metro <NA> <NA> <NA> 368.   
## 22 <NA> <NA> <NA> <NA> moto <NA> <NA> 813.   
## 23 <NA> <NA> <NA> <NA> <NA> bike andando 235.   
## 24 <NA> <NA> <NA> <NA> <NA> bike <NA> 1986.   
## 25 <NA> <NA> <NA> <NA> <NA> <NA> andando 11543.   
## 26 <NA> <NA> <NA> <NA> <NA> <NA> <NA> 109643.   
## `Qtd. Plano\_se`  
## <dbl>  
## 1 66.5  
## 2 168.   
## 3 119.   
## 4 226.   
## 5 509.   
## 6 146.   
## 7 50.5  
## 8 137.   
## 9 84.2  
## 10 127.   
## 11 1160.   
## 12 111.   
## 13 45.2  
## 14 235.   
## 15 47.3  
## 16 220.   
## 17 330.   
## 18 2479.   
## 19 65.0  
## 20 258.   
## 21 138.   
## 22 250.   
## 23 136.   
## 24 434.   
## 25 1151.   
## 26 3126.

amostra %>% # Cálculos para Samambaia  
 filter(A01ra == 12) %>%   
 mutate(Onibus=factor(case\_when(G141 == 1 ~ "onibus",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Automovel=factor(case\_when(G142 == 1 ~ "automovel",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Utilitario=factor(case\_when(G143 == 1 ~ "utilitario",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Metro=factor(case\_when(G144 == 1 ~ "metro",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Moto=factor(case\_when(G145 == 1 ~ "moto",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Bike=factor(case\_when(G146 == 1 ~ "bike",  
 TRUE ~ NA\_character\_))) %>%  
 mutate(Andando=factor(case\_when(G147 == 1 ~ "andando",  
 TRUE ~ NA\_character\_))) %>%  
 group\_by(Onibus, Automovel,Utilitario, Metro, Moto, Bike, Andando) %>%  
 summarise("Qtd. Samambaia"=survey\_total(na.rm=TRUE)) %>% print(n=100)

## # A tibble: 22 x 9  
## # Groups: Onibus, Automovel, Utilitario, Metro, Moto, Bike [17]  
## Onibus Automovel Utilitario Metro Moto Bike Andando `Qtd. Samambaia`  
## <fct> <fct> <fct> <fct> <fct> <fct> <fct> <dbl>  
## 1 onibus automovel <NA> metro <NA> <NA> <NA> 187.   
## 2 onibus automovel <NA> <NA> moto <NA> <NA> 86.6  
## 3 onibus automovel <NA> <NA> <NA> <NA> andando 190.   
## 4 onibus automovel <NA> <NA> <NA> <NA> <NA> 4321.   
## 5 onibus <NA> utilitario <NA> <NA> <NA> <NA> 105.   
## 6 onibus <NA> <NA> metro <NA> <NA> <NA> 2044.   
## 7 onibus <NA> <NA> <NA> moto <NA> <NA> 183.   
## 8 onibus <NA> <NA> <NA> <NA> bike <NA> 117.   
## 9 onibus <NA> <NA> <NA> <NA> <NA> andando 1505.   
## 10 onibus <NA> <NA> <NA> <NA> <NA> <NA> 38295.   
## 11 <NA> automovel <NA> metro <NA> <NA> <NA> 337.   
## 12 <NA> automovel <NA> <NA> moto <NA> <NA> 308.   
## 13 <NA> automovel <NA> <NA> <NA> <NA> andando 435.   
## 14 <NA> automovel <NA> <NA> <NA> <NA> <NA> 24413.   
## 15 <NA> <NA> utilitario <NA> <NA> <NA> <NA> 105.   
## 16 <NA> <NA> <NA> metro moto <NA> <NA> 99.5  
## 17 <NA> <NA> <NA> metro <NA> <NA> <NA> 3705.   
## 18 <NA> <NA> <NA> <NA> moto <NA> <NA> 3719.   
## 19 <NA> <NA> <NA> <NA> <NA> bike andando 83.6  
## 20 <NA> <NA> <NA> <NA> <NA> bike <NA> 1305.   
## 21 <NA> <NA> <NA> <NA> <NA> <NA> andando 10982.   
## 22 <NA> <NA> <NA> <NA> <NA> <NA> <NA> 140367.   
## `Qtd. Samambaia\_se`  
## <dbl>  
## 1 154.   
## 2 76.3  
## 3 148.   
## 4 697.   
## 5 117.   
## 6 509.   
## 7 116.   
## 8 114.   
## 9 342.   
## 10 1994.   
## 11 159.   
## 12 169.   
## 13 236.   
## 14 1256.   
## 15 99.8  
## 16 112.   
## 17 693.   
## 18 585.   
## 19 90.1  
## 20 381.   
## 21 1084.   
## 22 3896.

###### v) Tempo gasto de deslocamento ao trabalho (veja que a variável está em classes,   
 # podemos colocar o ponto médio do intervalo para o cálculo de medidas de posição).   
   
 # Calculos para o DF  
 amostra %>%   
 #filter(A01ra == 12) %>%   
 mutate(deslocamento=case\_when(G15 == 1 ~ 7L,  
 G15 == 2 ~ 22L,  
 G15 == 3 ~ 37L,  
 G15 == 4 ~ 52L,  
 G15 == 5 ~ 75L,  
 G15 == 6 ~ 97L,  
 G15 == 7 ~ 112L,  
 G15 == 8 ~ 120L,  
 TRUE ~ NA\_integer\_)) %>%  
 summarise("Tempo Médio Trabalho DF"=survey\_mean(deslocamento,na.rm=TRUE),  
 "Tempo Mediana Trabalho DF"=survey\_median(deslocamento,na.rm=TRUE),  
 "Tempo Variância Trabalho DF"=survey\_var(deslocamento,na.rm=TRUE),  
 "Tempo Desvio Padrão Trabalho DF"=survey\_sd(deslocamento,na.rm=TRUE),  
 "Tempo Q1 (<25%) Trabalho DF"=survey\_quantile(deslocamento, 0.25, na.rm=TRUE),  
 "Tempo Q3 (<75%) Trabalho DF"=survey\_quantile(deslocamento, 0.75, na.rm=TRUE),  
 "Tempo Q99 (<99%) Trabalho DF"=survey\_quantile(deslocamento, 0.99, na.rm=TRUE))

## # A tibble: 1 x 13  
## `Tempo Médio Trabalho DF` `Tempo Médio Trabalho DF\_se`  
## <dbl> <dbl>  
## 1 31.1 0.183  
## `Tempo Mediana Trabalho DF` `Tempo Mediana Trabalho DF\_se`  
## <dbl> <dbl>  
## 1 22 3.73  
## `Tempo Variância Trabalho DF` `Tempo Variância Trabalho DF\_se`  
## <dbl> <dbl>  
## 1 555. 8.04  
## `Tempo Desvio Padrão Trabalho DF` `Tempo Q1 (<25%) Trabalho DF\_q25`  
## <dbl> <dbl>  
## 1 23.6 7  
## `Tempo Q1 (<25%) Trabalho DF\_q25\_se` `Tempo Q3 (<75%) Trabalho DF\_q75`  
## <dbl> <dbl>  
## 1 0 37  
## `Tempo Q3 (<75%) Trabalho DF\_q75\_se` `Tempo Q99 (<99%) Trabalho DF\_q99`  
## <dbl> <dbl>  
## 1 3.73 112  
## `Tempo Q99 (<99%) Trabalho DF\_q99\_se`  
## <dbl>  
## 1 3.73

# Calculos para o Plano Piloto  
 amostra %>%   
 filter(A01ra == 1) %>%   
 mutate(deslocamento=case\_when(G15 == 1 ~ 7L,  
 G15 == 2 ~ 22L,  
 G15 == 3 ~ 37L,  
 G15 == 4 ~ 52L,  
 G15 == 5 ~ 75L,  
 G15 == 6 ~ 97L,  
 G15 == 7 ~ 112L,  
 G15 == 8 ~ 120L,  
 TRUE ~ NA\_integer\_)) %>%  
 summarise("Tempo Médio Trabalho Plano"=survey\_mean(deslocamento,na.rm=TRUE),  
 "Tempo Mediana Trabalho Plano"=survey\_median(deslocamento,na.rm=TRUE),  
 "Tempo Variância Trabalho Plano"=survey\_var(deslocamento,na.rm=TRUE),  
 "Tempo Desvio Padrão Trabalho Plano"=survey\_sd(deslocamento,na.rm=TRUE),  
 "Tempo Q1 (<25%) Trabalho Plano"=survey\_quantile(deslocamento, 0.25, na.rm=TRUE),  
 "Tempo Q3 (<75%) Trabalho Plano"=survey\_quantile(deslocamento, 0.75, na.rm=TRUE),  
 "Tempo Q99 (<99%) Trabalho Plano"=survey\_quantile(deslocamento, 0.99, na.rm=TRUE))

## # A tibble: 1 x 13  
## `Tempo Médio Trabalho Plano` `Tempo Médio Trabalho Plano\_se`  
## <dbl> <dbl>  
## 1 17.8 0.409  
## `Tempo Mediana Trabalho Plano` `Tempo Mediana Trabalho Plano\_se`  
## <dbl> <dbl>  
## 1 22 3.73  
## `Tempo Variância Trabalho Plano` `Tempo Variância Trabalho Plano\_se`  
## <dbl> <dbl>  
## 1 204. 21.3  
## `Tempo Desvio Padrão Trabalho Plano` `Tempo Q1 (<25%) Trabalho Plano\_q25`  
## <dbl> <dbl>  
## 1 14.3 7  
## `Tempo Q1 (<25%) Trabalho Plano\_q25\_se` `Tempo Q3 (<75%) Trabalho Plano\_q75`  
## <dbl> <dbl>  
## 1 0 22  
## `Tempo Q3 (<75%) Trabalho Plano\_q75\_se` `Tempo Q99 (<99%) Trabalho Plano\_q99`  
## <dbl> <dbl>  
## 1 3.73 75  
## `Tempo Q99 (<99%) Trabalho Plano\_q99\_se`  
## <dbl>  
## 1 5.72

# Calculos para Samambaia  
 amostra %>%   
 filter(A01ra == 12) %>%   
 mutate(deslocamento=case\_when(G15 == 1 ~ 7L,  
 G15 == 2 ~ 22L,  
 G15 == 3 ~ 37L,  
 G15 == 4 ~ 52L,  
 G15 == 5 ~ 75L,  
 G15 == 6 ~ 97L,  
 G15 == 7 ~ 112L,  
 G15 == 8 ~ 120L,  
 TRUE ~ NA\_integer\_)) %>%  
 summarise("Tempo Médio Trabalho Samambaia"=survey\_mean(deslocamento,na.rm=TRUE),  
 "Tempo Mediana Trabalho Samambaia"=survey\_median(deslocamento,na.rm=TRUE),  
 "Tempo Variância Trabalho Samambaia"=survey\_var(deslocamento,na.rm=TRUE),  
 "Tempo Desvio Padrão Trabalho Samambaia"=survey\_sd(deslocamento,na.rm=TRUE),  
 "Tempo Q1 (<25%) Trabalho Samambaia"=survey\_quantile(deslocamento, 0.25, na.rm=TRUE),  
 "Tempo Q3 (<75%) Trabalho Samambaia"=survey\_quantile(deslocamento, 0.75, na.rm=TRUE),  
 "Tempo Q99 (<99%) Trabalho Samambaia"=survey\_quantile(deslocamento, 0.99, na.rm=TRUE))

## # A tibble: 1 x 13  
## `Tempo Médio Trabalho Samambaia` `Tempo Médio Trabalho Samambaia\_se`  
## <dbl> <dbl>  
## 1 34.7 0.851  
## `Tempo Mediana Trabalho Samambaia` `Tempo Mediana Trabalho Samambaia\_se`  
## <dbl> <dbl>  
## 1 37 3.73  
## `Tempo Variância Trabalho Samambaia` `Tempo Variância Trabalho Samambaia\_se`  
## <dbl> <dbl>  
## 1 629. 37.1  
## `Tempo Desvio Padrão Trabalho Samambaia`  
## <dbl>  
## 1 25.1  
## `Tempo Q1 (<25%) Trabalho Samambaia\_q25`  
## <dbl>  
## 1 22  
## `Tempo Q1 (<25%) Trabalho Samambaia\_q25\_se`  
## <dbl>  
## 1 3.73  
## `Tempo Q3 (<75%) Trabalho Samambaia\_q75`  
## <dbl>  
## 1 52  
## `Tempo Q3 (<75%) Trabalho Samambaia\_q75\_se`  
## <dbl>  
## 1 3.73  
## `Tempo Q99 (<99%) Trabalho Samambaia\_q99`  
## <dbl>  
## 1 120  
## `Tempo Q99 (<99%) Trabalho Samambaia\_q99\_se`  
## <dbl>  
## 1 1.99

###### vi) Número de automóveis no domicilio   
 # Cálculos para DF  
 amostra %>%   
 filter(E02 == 1) %>%   
 mutate(carro=case\_when(C011 == 88888 ~NA\_real\_,  
 TRUE ~ as.numeric(C011))) %>%  
 summarise("Total da Carros DF"=survey\_total(carro, na.rm=TRUE),  
 "Média de Carros DF"=survey\_mean(carro, na.rm=TRUE),  
 "Desvio de Carros DF"=survey\_sd(carro, na.rm=TRUE),  
 "Carros Q1 (<25%) DF"=survey\_quantile(carro, 0.25, na.rm=TRUE),  
 "Carros Q3 (<75%) DF"=survey\_quantile(carro, 0.75, na.rm=TRUE),  
 "Carros Q99 (<99%) DF"=survey\_quantile(carro, 0.99, na.rm=TRUE))

## # A tibble: 1 x 11  
## `Total da Carros DF` `Total da Carros DF\_se` `Média de Carros DF`  
## <dbl> <dbl> <dbl>  
## 1 846540. 6230. 0.959  
## `Média de Carros DF\_se` `Desvio de Carros DF` `Carros Q1 (<25%) DF\_q25`  
## <dbl> <dbl> <dbl>  
## 1 0.00705 0.869 0  
## `Carros Q1 (<25%) DF\_q25\_se` `Carros Q3 (<75%) DF\_q75`  
## <dbl> <dbl>  
## 1 0 1  
## `Carros Q3 (<75%) DF\_q75\_se` `Carros Q99 (<99%) DF\_q99`  
## <dbl> <dbl>  
## 1 0.249 4  
## `Carros Q99 (<99%) DF\_q99\_se`  
## <dbl>  
## 1 0.249

amostra %>% # Cálculos para o Plano Piloto  
 filter(E02 == 1 & A01ra == 1) %>%   
 mutate(carro=case\_when(C011 == 88888 ~NA\_real\_,  
 TRUE ~ as.numeric(C011))) %>%  
 summarise("Total da Carros Plano"=survey\_total(carro, na.rm=TRUE),  
 "Média de Carros Plano"=survey\_mean(carro, na.rm=TRUE),  
 "Desvio de Carros Plano"=survey\_sd(carro, na.rm=TRUE),  
 "Carros Q1 (<25%) Plano"=survey\_quantile(carro, 0.25, na.rm=TRUE),  
 "Carros Q3 (<75%) Plano"=survey\_quantile(carro, 0.75, na.rm=TRUE),  
 "Carros Q99 (<99%) Plano"=survey\_quantile(carro, 0.99, na.rm=TRUE))

## # A tibble: 1 x 11  
## `Total da Carros Plano` `Total da Carros Plano\_se` `Média de Carros Plano`  
## <dbl> <dbl> <dbl>  
## 1 108122. 1932. 1.27  
## `Média de Carros Plano\_se` `Desvio de Carros Plano`  
## <dbl> <dbl>  
## 1 0.0215 0.819  
## `Carros Q1 (<25%) Plano\_q25` `Carros Q1 (<25%) Plano\_q25\_se`  
## <dbl> <dbl>  
## 1 1 0.249  
## `Carros Q3 (<75%) Plano\_q75` `Carros Q3 (<75%) Plano\_q75\_se`  
## <dbl> <dbl>  
## 1 2 0.249  
## `Carros Q99 (<99%) Plano\_q99` `Carros Q99 (<99%) Plano\_q99\_se`  
## <dbl> <dbl>  
## 1 4 0.249

amostra %>% # Cálculos para Samambaia  
 filter(E02 == 1 & A01ra == 12) %>%   
 mutate(carro=case\_when(C011 == 88888 ~NA\_real\_,  
 TRUE ~ as.numeric(C011))) %>%  
 summarise("Total da Carros Samambaia"=survey\_total(carro, na.rm=TRUE),  
 "Média de Carros Samambaia"=survey\_mean(carro, na.rm=TRUE),  
 "Desvio de Carros Samambaia"=survey\_sd(carro, na.rm=TRUE),  
 "Carros Q1 (<25%) Samambaia"=survey\_quantile(carro, 0.25, na.rm=TRUE),  
 "Carros Q3 (<75%) Samambaia"=survey\_quantile(carro, 0.75, na.rm=TRUE),  
 "Carros Q99 (<99%) Samambaia"=survey\_quantile(carro, 0.99, na.rm=TRUE))

## # A tibble: 1 x 11  
## `Total da Carros Samambaia` `Total da Carros Samambaia\_se`  
## <dbl> <dbl>  
## 1 51261. 1687.  
## `Média de Carros Samambaia` `Média de Carros Samambaia\_se`  
## <dbl> <dbl>  
## 1 0.745 0.0232  
## `Desvio de Carros Samambaia` `Carros Q1 (<25%) Samambaia\_q25`  
## <dbl> <dbl>  
## 1 0.741 0  
## `Carros Q1 (<25%) Samambaia\_q25\_se` `Carros Q3 (<75%) Samambaia\_q75`  
## <dbl> <dbl>  
## 1 0 1  
## `Carros Q3 (<75%) Samambaia\_q75\_se` `Carros Q99 (<99%) Samambaia\_q99`  
## <dbl> <dbl>  
## 1 0.249 3  
## `Carros Q99 (<99%) Samambaia\_q99\_se`  
## <dbl>  
## 1 0.249

###### vii) Número de pessoas no domicilio   
 amostra %>% filter(E02==1) %>% #Cálculo para o Distrito Federal  
 summarise("Média p/ Domicílio DF"=survey\_mean(A01nPessoas,na.rm=TRUE),  
 "Mediana p/ Domicílio DF"=survey\_median(A01nPessoas,na.rm=TRUE),  
 "Variancia p/ Domicílio DF"=survey\_mean(A01nPessoas,na.rm=TRUE),  
 "Desvio Padrão p/ Domicílio DF"=survey\_sd(A01nPessoas,na.rm=TRUE),  
 "Q1 (<25%) p/ Domicílio DF"=survey\_quantile(A01nPessoas,0.25,na.rm=TRUE),  
 "Q3 (<75%) p/ Domicílio DF"=survey\_quantile(A01nPessoas,0.75,na.rm=TRUE),  
 "Q3 (<75%) p/ Domicílio DF"=survey\_quantile(A01nPessoas,0.99,na.rm=TRUE))

## # A tibble: 1 x 13  
## `Média p/ Domicílio DF` `Média p/ Domicílio DF\_se` `Mediana p/ Domicílio DF`  
## <dbl> <dbl> <dbl>  
## 1 3.16 0.0126 3  
## `Mediana p/ Domicílio DF\_se` `Variancia p/ Domicílio DF`  
## <dbl> <dbl>  
## 1 0.249 3.16  
## `Variancia p/ Domicílio DF\_se` `Desvio Padrão p/ Domicílio DF`  
## <dbl> <dbl>  
## 1 0.0126 1.45  
## `Q1 (<25%) p/ Domicílio DF\_q25` `Q1 (<25%) p/ Domicílio DF\_q25\_se`  
## <dbl> <dbl>  
## 1 2 0.249  
## `Q3 (<75%) p/ Domicílio DF\_q75` `Q3 (<75%) p/ Domicílio DF\_q75\_se`  
## <dbl> <dbl>  
## 1 4 0.249  
## `Q3 (<75%) p/ Domicílio DF\_q99` `Q3 (<75%) p/ Domicílio DF\_q99\_se`  
## <dbl> <dbl>  
## 1 7 0.249

amostra %>% filter(E02==1 & A01ra==1) %>% # Cálculo para o Plano Piloto  
 summarise("Média p/ Domicílio Plano"=survey\_mean(A01nPessoas,na.rm=TRUE),  
 "Mediana p/ Domicílio Plano"=survey\_median(A01nPessoas,na.rm=TRUE),  
 "Variancia p/ Domicílio Plano"=survey\_mean(A01nPessoas,na.rm=TRUE),  
 "Desvio Padrão p/ Domicílio Plano"=survey\_sd(A01nPessoas,na.rm=TRUE),  
 "Q1 (<25%) p/ Domicílio Plano"=survey\_quantile(A01nPessoas,0.25,na.rm=TRUE),  
 "Q3 (<75%) p/ Domicílio Plano"=survey\_quantile(A01nPessoas,0.75,na.rm=TRUE),  
 "Q3 (<75%) p/ Domicílio Plano"=survey\_quantile(A01nPessoas,0.99,na.rm=TRUE))

## # A tibble: 1 x 13  
## `Média p/ Domicílio Plano` `Média p/ Domicílio Plano\_se`  
## <dbl> <dbl>  
## 1 2.60 0.0384  
## `Mediana p/ Domicílio Plano` `Mediana p/ Domicílio Plano\_se`  
## <dbl> <dbl>  
## 1 2 0.249  
## `Variancia p/ Domicílio Plano` `Variancia p/ Domicílio Plano\_se`  
## <dbl> <dbl>  
## 1 2.60 0.0384  
## `Desvio Padrão p/ Domicílio Plano` `Q1 (<25%) p/ Domicílio Plano\_q25`  
## <dbl> <dbl>  
## 1 1.23 2  
## `Q1 (<25%) p/ Domicílio Plano\_q25\_se` `Q3 (<75%) p/ Domicílio Plano\_q75`  
## <dbl> <dbl>  
## 1 0.249 3  
## `Q3 (<75%) p/ Domicílio Plano\_q75\_se` `Q3 (<75%) p/ Domicílio Plano\_q99`  
## <dbl> <dbl>  
## 1 0.249 6  
## `Q3 (<75%) p/ Domicílio Plano\_q99\_se`  
## <dbl>  
## 1 0.249

amostra %>% filter(E02==1 & A01ra==12) %>% # Cálculo para Samambaia  
 summarise("Média p/ Domicílio Samambaia"=survey\_mean(A01nPessoas,na.rm=TRUE),  
 "Mediana p/ Domicílio Samambaia"=survey\_median(A01nPessoas,na.rm=TRUE),  
 "Variancia p/ Domicílio Samambaia"=survey\_mean(A01nPessoas,na.rm=TRUE),  
 "Desvio Padrão p/ Domicílio Samambaia"=survey\_sd(A01nPessoas,na.rm=TRUE),  
 "Q1 (<25%) p/ Domicílio Samambaia"=survey\_quantile(A01nPessoas,0.25,na.rm=TRUE),  
 "Q3 (<75%) p/ Domicílio Samambaia"=survey\_quantile(A01nPessoas,0.75,na.rm=TRUE),  
 "Q3 (<75%) p/ Domicílio Samambaia"=survey\_quantile(A01nPessoas,0.99,na.rm=TRUE))

## # A tibble: 1 x 13  
## `Média p/ Domicílio Samambaia` `Média p/ Domicílio Samambaia\_se`  
## <dbl> <dbl>  
## 1 3.26 0.0492  
## `Mediana p/ Domicílio Samambaia` `Mediana p/ Domicílio Samambaia\_se`  
## <dbl> <dbl>  
## 1 3 0.249  
## `Variancia p/ Domicílio Samambaia` `Variancia p/ Domicílio Samambaia\_se`  
## <dbl> <dbl>  
## 1 3.26 0.0492  
## `Desvio Padrão p/ Domicílio Samambaia` `Q1 (<25%) p/ Domicílio Samambaia\_q25`  
## <dbl> <dbl>  
## 1 1.49 2  
## `Q1 (<25%) p/ Domicílio Samambaia\_q25\_se`  
## <dbl>  
## 1 0.249  
## `Q3 (<75%) p/ Domicílio Samambaia\_q75`  
## <dbl>  
## 1 4  
## `Q3 (<75%) p/ Domicílio Samambaia\_q75\_se`  
## <dbl>  
## 1 0.249  
## `Q3 (<75%) p/ Domicílio Samambaia\_q99`  
## <dbl>  
## 1 7  
## `Q3 (<75%) p/ Domicílio Samambaia\_q99\_se`  
## <dbl>  
## 1 0.249

###### viii)Número de dormitórios no domicílio   
 amostra %>% filter(E02==1) %>% #Cálculo para o Distrito Federal  
 summarise("Média p/ Cômodo DF"=survey\_mean(B12,na.rm=TRUE),  
 "Mediana p/ Cômodo DF"=survey\_median(B12,na.rm=TRUE),  
 "Variancia p/ Cômodo DF"=survey\_mean(B12,na.rm=TRUE),  
 "Desvio Padrão p/ Cômodo DF"=survey\_sd(B12,na.rm=TRUE),  
 "Q1 (<25%) p/ Cômodo DF"=survey\_quantile(B12,0.25,na.rm=TRUE),  
 "Q3 (<75%) p/ Cômodo DF"=survey\_quantile(B12,0.75,na.rm=TRUE),  
 "Q3 (<75%) p/ Cômodo DF"=survey\_quantile(B12,0.99,na.rm=TRUE))

## # A tibble: 1 x 13  
## `Média p/ Cômodo DF` `Média p/ Cômodo DF\_se` `Mediana p/ Cômodo DF`  
## <dbl> <dbl> <dbl>  
## 1 2.49 0.00810 2  
## `Mediana p/ Cômodo DF\_se` `Variancia p/ Cômodo DF` `Variancia p/ Cômodo DF\_se`  
## <dbl> <dbl> <dbl>  
## 1 0.249 2.49 0.00810  
## `Desvio Padrão p/ Cômodo DF` `Q1 (<25%) p/ Cômodo DF\_q25`  
## <dbl> <dbl>  
## 1 1.13 2  
## `Q1 (<25%) p/ Cômodo DF\_q25\_se` `Q3 (<75%) p/ Cômodo DF\_q75`  
## <dbl> <dbl>  
## 1 0.249 3  
## `Q3 (<75%) p/ Cômodo DF\_q75\_se` `Q3 (<75%) p/ Cômodo DF\_q99`  
## <dbl> <dbl>  
## 1 0.249 6  
## `Q3 (<75%) p/ Cômodo DF\_q99\_se`  
## <dbl>  
## 1 0.249

amostra %>% filter(E02==1 & A01ra==1) %>% # Cálculo para o Plano Piloto  
 summarise("Média p/ Cômodo Plano"=survey\_mean(B12,na.rm=TRUE),  
 "Mediana p/ Cômodo Plano"=survey\_median(B12,na.rm=TRUE),  
 "Variancia p/ Cômodo Plano"=survey\_mean(B12,na.rm=TRUE),  
 "Desvio Padrão p/ Cômodo Plano"=survey\_sd(B12,na.rm=TRUE),  
 "Q1 (<25%) p/ Cômodo Plano"=survey\_quantile(B12,0.25,na.rm=TRUE),  
 "Q3 (<75%) p/ Cômodo Plano"=survey\_quantile(B12,0.75,na.rm=TRUE),  
 "Q3 (<75%) p/ Cômodo Plano"=survey\_quantile(B12,0.99,na.rm=TRUE))

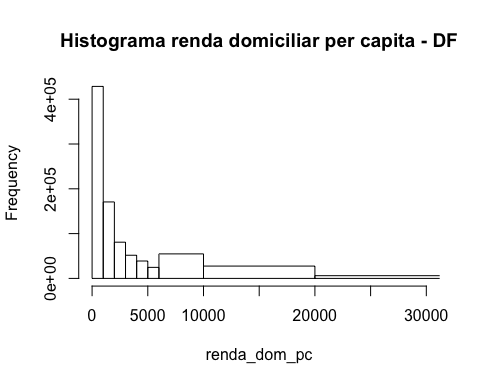
## # A tibble: 1 x 13  
## `Média p/ Cômodo Plano` `Média p/ Cômodo Plano\_se` `Mediana p/ Cômodo Plano`  
## <dbl> <dbl> <dbl>  
## 1 2.72 0.0371 3  
## `Mediana p/ Cômodo Plano\_se` `Variancia p/ Cômodo Plano`  
## <dbl> <dbl>  
## 1 0.249 2.72  
## `Variancia p/ Cômodo Plano\_se` `Desvio Padrão p/ Cômodo Plano`  
## <dbl> <dbl>  
## 1 0.0371 1.14  
## `Q1 (<25%) p/ Cômodo Plano\_q25` `Q1 (<25%) p/ Cômodo Plano\_q25\_se`  
## <dbl> <dbl>  
## 1 2 0.249  
## `Q3 (<75%) p/ Cômodo Plano\_q75` `Q3 (<75%) p/ Cômodo Plano\_q75\_se`  
## <dbl> <dbl>  
## 1 3 0.249  
## `Q3 (<75%) p/ Cômodo Plano\_q99` `Q3 (<75%) p/ Cômodo Plano\_q99\_se`  
## <dbl> <dbl>  
## 1 6 0.249

amostra %>% filter(E02==1 & A01ra==12) %>% # Cálculo para Samambaia  
 summarise("Média p/ Cômodo Samambaia"=survey\_mean(B12,na.rm=TRUE),  
 "Mediana p/ Cômodo Samambaia"=survey\_median(B12,na.rm=TRUE),  
 "Variancia p/ Cômodo Samambaia"=survey\_mean(B12,na.rm=TRUE),  
 "Desvio Padrão p/ Cômodo Samambaia"=survey\_sd(B12,na.rm=TRUE),  
 "Q1 (<25%) p/ Cômodo Samambaia"=survey\_quantile(B12,0.25,na.rm=TRUE),  
 "Q3 (<75%) p/ Cômodo Samambaia"=survey\_quantile(B12,0.75,na.rm=TRUE),  
 "Q3 (<75%) p/ Cômodo Samambaia"=survey\_quantile(B12,0.99,na.rm=TRUE))

## # A tibble: 1 x 13  
## `Média p/ Cômodo Samambaia` `Média p/ Cômodo Samambaia\_se`  
## <dbl> <dbl>  
## 1 2.26 0.0353  
## `Mediana p/ Cômodo Samambaia` `Mediana p/ Cômodo Samambaia\_se`  
## <dbl> <dbl>  
## 1 2 0.249  
## `Variancia p/ Cômodo Samambaia` `Variancia p/ Cômodo Samambaia\_se`  
## <dbl> <dbl>  
## 1 2.26 0.0353  
## `Desvio Padrão p/ Cômodo Samambaia` `Q1 (<25%) p/ Cômodo Samambaia\_q25`  
## <dbl> <dbl>  
## 1 0.857 2  
## `Q1 (<25%) p/ Cômodo Samambaia\_q25\_se` `Q3 (<75%) p/ Cômodo Samambaia\_q75`  
## <dbl> <dbl>  
## 1 0.249 3  
## `Q3 (<75%) p/ Cômodo Samambaia\_q75\_se` `Q3 (<75%) p/ Cômodo Samambaia\_q99`  
## <dbl> <dbl>  
## 1 0.249 5  
## `Q3 (<75%) p/ Cômodo Samambaia\_q99\_se`  
## <dbl>  
## 1 0.249

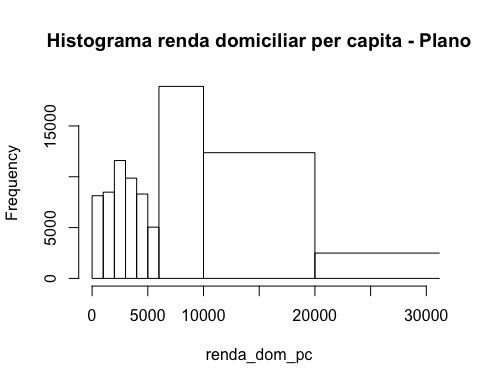
##### 1.3. Faça um histograma (com barras e alisado) para as variáveis renda domiciliar   
 # per capita e número de automóveis no domicílio para a RA X´ com o Plano Piloto e o Distrito Federal  
   
 # Histograma renda domiciliar per capita   
 #Cálculo para o Distrito Federal  
 survey::svyhist(~renda\_dom\_pc, probability = FALSE, xlim=c(0, 30000),   
 main = "Histograma renda domiciliar per capita - DF",   
 breaks=c(0,1000,2000,3000,4000,5000,6000,10000,20000,500000),   
 subset(amostra,E02==1))

## Warning in plot.histogram(h, ..., freq = freq, xlab = xlab, main = main): the  
## AREAS in the plot are wrong -- rather use 'freq = FALSE'



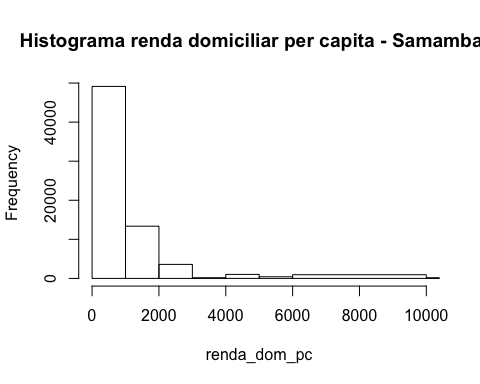
#Cálculo para o Plano  
 survey::svyhist(~renda\_dom\_pc, probability = FALSE, xlim=c(0, 30000),   
 main = "Histograma renda domiciliar per capita - Plano",   
 breaks=c(0,1000,2000,3000,4000,5000,6000,10000,20000,500000),   
 subset(amostra,E02==1 & A01ra==1))

## Warning in plot.histogram(h, ..., freq = freq, xlab = xlab, main = main): the  
## AREAS in the plot are wrong -- rather use 'freq = FALSE'



#Cálculo para o Samambaia  
 survey::svyhist(~renda\_dom\_pc, probability = FALSE, xlim=c(0, 10000),   
 main = "Histograma renda domiciliar per capita - Samambaia",   
 breaks=c(0,1000,2000,3000,4000,5000,6000,10000,20000),   
 subset(amostra,E02==1 & A01ra==12))

## Warning in plot.histogram(h, ..., freq = freq, xlab = xlab, main = main): the  
## AREAS in the plot are wrong -- rather use 'freq = FALSE'



##### Parte 3 – Tópicos Especiais   
##### 3.1. Calcule o Gini da renda domiciliar per capita para o DF, RA X´   
 # e o Plano Piloto. Qual a sua conclusão?  
   
 # Preparar a base  
 amostra\_gini<- convey::convey\_prep(amostra)  
  
   
 # Cálculo Gini para o DF   
 convey::svygini(~renda\_dom\_pc,  
 subset(amostra\_gini,E02==1),  
 na.rm=T)

## gini SE  
## renda\_dom\_pc 0.61502 0.004

# Cálculo Gini para o Plano  
 convey::svygini(~renda\_dom\_pc,  
 subset(amostra\_gini,E02==1 & A01ra==1),  
 na.rm=T)

## gini SE  
## renda\_dom\_pc 0.47725 0.0141

# Cálculo Gini para a Sammabaia  
 convey::svygini(~renda\_dom\_pc,  
 subset(amostra\_gini,E02==1 & A01ra==12),  
 na.rm=T)

## gini SE  
## renda\_dom\_pc 0.56621 0.032

##### 3.2 – Vamos supor que o grupo seja convidado a opinar sobre a discussão   
 # da gestão da pandemia de Covid-19:   
   
##### i) Para subsidiar a discussão, primeiro calcule o número de crianças e   
 # adolescentes (0 a 18 anos) que mora com pessoas mais de 60 anos. Calcule   
 # o intervalo de confiança desta estimativa. Faça isso para RA X´ e para o   
 # DF.  
   
 # Cálculo para o DF   
 amostra %>%  
 # Filtrar População que mora com idoso e que tenho 18 anos ou menos  
 srvyr::filter(idoso >=1 & idade\_calculada <=18 ) %>%  
 # Criar contador  
 srvyr::mutate(count=1) %>%  
 # Calcular o total  
 srvyr::summarise("População Total"=survey\_total(count, vartype = "ci"))

## # A tibble: 1 x 3  
## `População Total` `População Total\_low` `População Total\_upp`  
## <dbl> <dbl> <dbl>  
## 1 114461. 107639. 121284.

# Cálculo para o Plano  
 amostra %>%  
 # Filtrar População que mora com idoso e que tenho 18 anos ou menos  
 srvyr::filter(A01ra == 1 & idoso >=1 & idade\_calculada <=18 ) %>%  
 # Criar contador  
 srvyr::mutate(count=1) %>%  
 # Calcular o total  
 srvyr::summarise("População Total"=survey\_total(count, vartype = "ci"))

## # A tibble: 1 x 3  
## `População Total` `População Total\_low` `População Total\_upp`  
## <dbl> <dbl> <dbl>  
## 1 5454. 4203. 6705.

# Cálculo para o Samambaia  
 amostra %>%  
 # Filtrar População que mora com idoso e que tenho 18 anos ou menos  
 srvyr::filter(A01ra == 12 & idoso >=1 & idade\_calculada <=18 ) %>%  
 # Criar contador  
 srvyr::mutate(count=1) %>%  
 # Calcular o total  
 srvyr::summarise("População Total"=survey\_total(count, vartype = "ci"))

## # A tibble: 1 x 3  
## `População Total` `População Total\_low` `População Total\_upp`  
## <dbl> <dbl> <dbl>  
## 1 9369. 7053. 11686.

# Crie um filtro para alunos de escolas particulares e públicas  
 # Cálculo para alunos de escolas particulares e públicas - DF   
 amostra %>%  
 # Filtrar População que mora com idoso e que tenho 18 anos ou menos e filtrar quem estuda  
 srvyr::filter(A01ra >= 1 & idoso >=1 & idade\_calculada <=18  
 & F02 <=2 ) %>%  
   
 # Ajustar nome das variáveis  
 srvyr::mutate(F02=factor(case\_when(F02 %in% 1 ~"Pública",  
 F02 %in% 2 ~"Privada",  
 TRUE ~ NA\_character\_))) %>%  
 # Agrupar por tipo de escola  
 srvyr::group\_by(F02) %>%  
 # Calcular o total e o Percentual, com seu intervalo de confiança  
 srvyr::summarise("Escolas"=survey\_total(vartype = "ci"),  
 # Calcular o percentual   
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 2 x 7  
## F02 Escolas Escolas\_low Escolas\_upp pct pct\_low pct\_upp  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Privada 28877. 26232. 31522. 0.311 0.287 0.334  
## 2 Pública 64074. 59283. 68866. 0.689 0.666 0.713

# Cálculo para alunos de escolas particulares e públicas - Plano   
 amostra %>%  
 # Filtrar População que mora com idoso e que tenho 18 anos ou menos e filtrar quem estuda  
 srvyr::filter(A01ra == 1 & idoso >=1 & idade\_calculada <=18   
 & F02 <=2 ) %>%  
   
 # Ajustar nome das variáveis  
 srvyr::mutate(F02=factor(case\_when(F02 %in% 1 ~"Pública",  
 F02 %in% 2 ~"Privada",  
 TRUE ~ NA\_character\_))) %>%  
 # Agrupar por tipo de escola  
 srvyr::group\_by(F02) %>%  
 # Calcular o total e o Percentual, com seu intervalo de confiança  
 srvyr::summarise("Escolas"=survey\_total(vartype = "ci"),  
 # Calcular o percentual   
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 2 x 7  
## F02 Escolas Escolas\_low Escolas\_upp pct pct\_low pct\_upp  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Privada 2516. 1563. 3469. 0.547 0.418 0.675  
## 2 Pública 2084. 1361. 2808. 0.453 0.325 0.582

# Cálculo para alunos de escolas particulares e públicas - Samambaia   
 amostra %>%  
 # Filtrar População que mora com idoso e que tenho 18 anos ou menos e filtrar quem estuda  
 srvyr::filter(A01ra == 12 & idoso >=1 & idade\_calculada <=18   
 & F02 <=2 ) %>%  
   
 # Ajustar nome das variáveis  
 srvyr::mutate(F02=factor(case\_when(F02 %in% 1 ~"Pública",  
 F02 %in% 2 ~"Privada",  
 TRUE ~ NA\_character\_))) %>%  
 # Agrupar por tipo de escola  
 srvyr::group\_by(F02) %>%  
 # Calcular o total e o Percentual, com seu intervalo de confiança  
 srvyr::summarise("Escolas"=survey\_total(vartype = "ci"),  
 # Calcular o percentual   
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 2 x 7  
## F02 Escolas Escolas\_low Escolas\_upp pct pct\_low pct\_upp  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Privada 1467. 771. 2163. 0.194 0.0916 0.296  
## 2 Pública 6104. 3978. 8231. 0.806 0.704 0.908

##### ii) Calcule o número médio de pessoas por dormitório desses   
 # domicílios com crianças para a RA X´ e o DF. Crie um filtro para alunos  
 # de escolas particulares e públicas. (crianças de 0 a 12 anos)   
   
 #Cálculo para o Distrito Federal  
 amostra %>%   
 srvyr::filter(crianca\_estuda >=1 & E02==1) %>%  
 summarise("Média Pessoas p/ Dormitório DF"=survey\_mean(  
 pessoa\_dorm, vartype = "ci", na.rm=TRUE))

## # A tibble: 1 x 3  
## `Média Pessoas p/ Dormitório DF` `Média Pessoas p/ Dormitório DF\_low`  
## <dbl> <dbl>  
## 1 1.91 1.88  
## `Média Pessoas p/ Dormitório DF\_upp`  
## <dbl>  
## 1 1.94

#Cálculo para o Plano Piloto  
 amostra %>%   
 srvyr::filter(A01ra == 1 & crianca\_estuda >=1 & E02==1) %>%  
 summarise("Média Pessoas p/ Dormitório Plano"=survey\_mean(  
 pessoa\_dorm, vartype = "ci",na.rm=TRUE))

## # A tibble: 1 x 3  
## `Média Pessoas p/ Dormitório Plano` `Média Pessoas p/ Dormitório Plano\_low`  
## <dbl> <dbl>  
## 1 1.47 1.38  
## `Média Pessoas p/ Dormitório Plano\_upp`  
## <dbl>  
## 1 1.57

#Cálculo para Samambaia  
 amostra %>%   
 srvyr::filter(A01ra == 12 & crianca\_estuda >=1 & E02==1) %>%  
 summarise("Média Pessoas p/ Dormitório Samambaia"=survey\_mean(  
 pessoa\_dorm, vartype = "ci",na.rm=TRUE))

## # A tibble: 1 x 3  
## `Média Pessoas p/ Dormitório Samambaia`  
## <dbl>  
## 1 1.98  
## `Média Pessoas p/ Dormitório Samambaia\_low`  
## <dbl>  
## 1 1.87  
## `Média Pessoas p/ Dormitório Samambaia\_upp`  
## <dbl>  
## 1 2.09

# Crie um filtro para alunos de escolas particulares e públicas.  
 #Cálculo para o Distrito Federal  
 amostra %>%   
 srvyr::filter(crianca\_estuda >=1 & E02==1 & F02 <=2 ) %>%  
 srvyr::mutate(F02=factor(case\_when(F02 %in% 1 ~"Pública",  
 F02 %in% 2 ~"Privada",  
 TRUE ~ NA\_character\_))) %>%  
 # Agrupar por tipo de escola  
 srvyr::group\_by(F02) %>%  
 # Calcular o total e o Percentual, com seu intervalo de confiança  
 srvyr::summarise("Média de Pessoas/Domitório DF"=survey\_mean(  
 pessoa\_dorm, vartype = "ci",na.rm=TRUE))

## # A tibble: 2 x 4  
## F02 `Média de Pessoas/Domitório DF` `Média de Pessoas/Domitório DF\_low`  
## <fct> <dbl> <dbl>  
## 1 Privada 1.80 1.71  
## 2 Pública 2.03 1.84  
## `Média de Pessoas/Domitório DF\_upp`  
## <dbl>  
## 1 1.89  
## 2 2.23

#Cálculo para o Plano Piloto  
 amostra %>%   
 srvyr::filter(A01ra == 1 & crianca\_estuda >=1 & E02==1 & F02 <=2 ) %>%  
 srvyr::mutate(F02=factor(case\_when(F02 %in% 1 ~"Pública",  
 F02 %in% 2 ~"Privada",  
 TRUE ~ NA\_character\_))) %>%  
 # Agrupar por tipo de escola  
 srvyr::group\_by(F02) %>%  
 # Calcular o total e o Percentual, com seu intervalo de confiança  
 srvyr::summarise("Média de Pessoas/Domitório Plano"=survey\_mean(  
 pessoa\_dorm, vartype = "ci",na.rm=TRUE))

## Warning in svrVar(repmeans, scale, rscales, mse = design$mse, coef = rval): 1  
## replicates gave NA results and were discarded.

## # A tibble: 2 x 4  
## F02 `Média de Pessoas/Domitório Plano`  
## <fct> <dbl>  
## 1 Privada 1.40  
## 2 Pública 1.57  
## `Média de Pessoas/Domitório Plano\_low` `Média de Pessoas/Domitório Plano\_upp`  
## <dbl> <dbl>  
## 1 1.03 1.76  
## 2 1.19 1.94

#Cálculo para o Samambaia  
 amostra %>%   
 srvyr::filter(A01ra == 12 & crianca\_estuda >=1 & E02==1 & F02 <=2 ) %>%  
 srvyr::mutate(F02=factor(case\_when(F02 %in% 1 ~"Pública",  
 F02 %in% 2 ~"Privada",  
 TRUE ~ NA\_character\_))) %>%  
 # Agrupar por tipo de escola  
 srvyr::group\_by(F02) %>%  
 # Calcular o total e o Percentual, com seu intervalo de confiança  
 srvyr::summarise("Média de Pessoas/Domitório Plano"=survey\_mean(  
 pessoa\_dorm, vartype = "ci",na.rm=TRUE))

## Warning in svrVar(repmeans, scale, rscales, mse = design$mse, coef = rval): 2  
## replicates gave NA results and were discarded.

## # A tibble: 2 x 4  
## F02 `Média de Pessoas/Domitório Plano`  
## <fct> <dbl>  
## 1 Privada 1.75  
## 2 Pública 1.87  
## `Média de Pessoas/Domitório Plano\_low` `Média de Pessoas/Domitório Plano\_upp`  
## <dbl> <dbl>  
## 1 1.15 2.35  
## 2 1.47 2.28

##### iii) Calcule o número de crianças, adolescentes e jovens (0 a 29 anos)   
 # por nível escolar. Calcule também quantos destes utilizam o transporte   
 # coletivo para ir à escola (ônibus, metrô, transporte escolar público ou   
 # transporte escolar privado). Calcule o intervalo de confiança desta   
 # estimativa. Faça também um filtro para unidades públicas e particulares.  
   
 # Calcule o número de crianças, adolescentes e jovens (0 a 29 anos)   
 # por nível escolar  
   
 #Cálculo para o Distrito Federal   
 amostra %>%   
 srvyr::filter(F02 <=2 & idade\_calculada <= 29 & is.na(nivel\_escola)==F) %>%  
 srvyr::group\_by(nivel\_escola) %>%   
 srvyr::summarise(n=survey\_total())

## # A tibble: 5 x 3  
## nivel\_escola n n\_se  
## <fct> <dbl> <dbl>  
## 1 AJA e EJA 13766. 1047.  
## 2 Creche/Educação Infantil 122260. 3075.  
## 3 Ensino Fundamental 352839. 6064.  
## 4 Ensino Médio 112256. 2857.  
## 5 Ensino Superior 119591. 2754.

#Cálculo para o Plano  
 amostra %>%   
 srvyr::filter(A01ra == 1 & F02 <=2 & idade\_calculada <= 29 & is.na(nivel\_escola)==F) %>%  
 srvyr::group\_by(nivel\_escola) %>%   
 srvyr::summarise(n=survey\_total())

## # A tibble: 5 x 3  
## nivel\_escola n n\_se  
## <fct> <dbl> <dbl>  
## 1 AJA e EJA 397. 159.  
## 2 Creche/Educação Infantil 7432. 792.  
## 3 Ensino Fundamental 17442. 1501.  
## 4 Ensino Médio 5443. 629.  
## 5 Ensino Superior 12806. 1101.

#Cálculo para o Samambaia  
 amostra %>%   
 srvyr::filter(A01ra == 12 & F02 <=2 & idade\_calculada <= 29 & is.na(nivel\_escola)==F) %>%  
 srvyr::group\_by(nivel\_escola) %>%   
 srvyr::summarise(n=survey\_total())

## # A tibble: 5 x 3  
## nivel\_escola n n\_se  
## <fct> <dbl> <dbl>  
## 1 AJA e EJA 1473. 367.  
## 2 Creche/Educação Infantil 11742. 1029.  
## 3 Ensino Fundamental 33453. 2430.  
## 4 Ensino Médio 9855. 869.  
## 5 Ensino Superior 7241. 688.

#Calcule também quantos destes utilizam o transporte   
 #coletivo para ir à escola (ônibus, metrô, transporte escolar público ou   
 #transporte escolar privado).  
   
 #Cálculo para o DF   
 amostra %>%   
 srvyr::filter( F02 <=2 & idade\_calculada <= 29 & is.na(nivel\_escola  
 )==F) %>%  
 srvyr::filter(F05==1 | F05==2 | F05==3 | F05==6) %>%  
 srvyr::group\_by(desloc\_escola) %>%   
 srvyr::summarise(n=survey\_total(vartype = "ci"))

## # A tibble: 4 x 4  
## desloc\_escola n n\_low n\_upp  
## <fct> <dbl> <dbl> <dbl>  
## 1 Metrô 9333. 7371. 11295.  
## 2 Ônibus 162369. 155341. 169397.  
## 3 Transporte escolar privado 52344. 48246. 56443.  
## 4 Transporte escolar público 21635. 19668. 23603.

#Cálculo para o Plano   
 amostra %>%   
 srvyr::filter(A01ra == 1 & F02 <=2 & idade\_calculada <= 29 &   
 is.na(nivel\_escola)==F) %>%  
 srvyr::filter(F05==1 | F05==2 | F05==3 | F05==6) %>%  
 srvyr::group\_by(desloc\_escola) %>%   
 srvyr::summarise(n=survey\_total(vartype = "ci"))

## # A tibble: 4 x 4  
## desloc\_escola n n\_low n\_upp  
## <fct> <dbl> <dbl> <dbl>  
## 1 Metrô 210. -56.0 477.  
## 2 Ônibus 7530. 6019. 9040.  
## 3 Transporte escolar privado 2606. 1685. 3526.  
## 4 Transporte escolar público 679. 276. 1083.

#Cálculo para o Samambaia   
 amostra %>%   
 srvyr::filter(A01ra == 12 & F02 <=2 & idade\_calculada <= 29 &   
 is.na(nivel\_escola)==F) %>%  
 srvyr::filter(F05==1 | F05==2 | F05==3 | F05==6) %>%  
 srvyr::group\_by(desloc\_escola) %>%   
 srvyr::summarise(n=survey\_total(vartype = "ci"))

## # A tibble: 4 x 4  
## desloc\_escola n n\_low n\_upp  
## <fct> <dbl> <dbl> <dbl>  
## 1 Metrô 1518. 622. 2414.  
## 2 Ônibus 18761. 16583. 20939.  
## 3 Transporte escolar privado 4011. 2826. 5197.  
## 4 Transporte escolar público 218. -75.0 511.

#Faça também um filtro para unidades públicas e particulares  
 #Cálculo para o DF   
 amostra %>%   
 srvyr::filter( F02 <=2 & idade\_calculada <= 29 & is.na(nivel\_escola  
 )==F) %>%  
 srvyr::filter(F05==1 | F05==2 | F05==3 | F05==6) %>%  
 srvyr::mutate(F02=factor(case\_when(F02 %in% 1 ~"Pública",  
 F02 %in% 2 ~"Privada",  
 TRUE ~ NA\_character\_))) %>%  
 srvyr::group\_by(F02) %>%   
 srvyr::summarise(n=survey\_total(vartype = "ci"))

## # A tibble: 2 x 4  
## F02 n n\_low n\_upp  
## <fct> <dbl> <dbl> <dbl>  
## 1 Privada 81689. 77028. 86350.  
## 2 Pública 163992. 157061. 170923.

#Cálculo para o Plano  
 amostra %>%   
 srvyr::filter(A01ra == 1 &F02 <=2 & idade\_calculada <= 29 &   
 is.na(nivel\_escola)==F) %>%  
 srvyr::filter(F05==1 | F05==2 | F05==3 | F05==6) %>%  
 srvyr::mutate(F02=factor(case\_when(F02 %in% 1 ~"Pública",  
 F02 %in% 2 ~"Privada",  
 TRUE ~ NA\_character\_))) %>%  
 srvyr::group\_by(F02) %>%   
 srvyr::summarise(n=survey\_total(vartype = "ci"))

## # A tibble: 2 x 4  
## F02 n n\_low n\_upp  
## <fct> <dbl> <dbl> <dbl>  
## 1 Privada 4374. 2984. 5764.  
## 2 Pública 6651. 5261. 8040.

#Cálculo para a Samambaia  
 amostra %>%   
 srvyr::filter(A01ra == 12 &F02 <=2 & idade\_calculada <= 29 &   
 is.na(nivel\_escola)==F) %>%  
 srvyr::filter(F05==1 | F05==2 | F05==3 | F05==6) %>%  
 srvyr::mutate(F02=factor(case\_when(F02 %in% 1 ~"Pública",  
 F02 %in% 2 ~"Privada",  
 TRUE ~ NA\_character\_))) %>%  
 srvyr::group\_by(F02) %>%   
 srvyr::summarise(n=survey\_total(vartype = "ci"))

## # A tibble: 2 x 4  
## F02 n n\_low n\_upp  
## <fct> <dbl> <dbl> <dbl>  
## 1 Privada 6195. 4532. 7859.  
## 2 Pública 18314. 15846. 20782.

##### iv) Estime o percentual de domicílios conectados à internet para o   
 # Distrito Federal. Faça também um filtro estudantes de escolas públicas   
 # e particulares, independentemente da idade. Faça também um teste de   
 # hipótese a 95% de confiança para comparar se as duas proporções são  
 # equivalentes ou não.  
   
 #Cálculo para o Distrito Federal   
 #Domicílios conectados à internet  
 amostra %>%  
 # Filtrar População DF pelo proprietário da casa  
 srvyr::filter(E02==1 & is.na(internet)==F) %>%  
 # Criar contador  
 srvyr::mutate(count=1) %>%  
 # Agrupar por internet  
 srvyr::group\_by(internet) %>%   
 # Calcular o total  
 srvyr::summarise("População Total"=survey\_total(vartype = "ci"),  
 # Calcular o percentual  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 2 x 7  
## internet `População Total` `População Total\_low` `População Total\_upp` pct  
## <fct> <dbl> <dbl> <dbl> <dbl>  
## 1 Não 55181. 51707. 58656. 0.0626  
## 2 Sim 826729. 822177. 831281. 0.937   
## pct\_low pct\_upp  
## <dbl> <dbl>  
## 1 0.0586 0.0665  
## 2 0.933 0.941

# Faça também um filtro estudantes de escolas públicas   
 # e particulares, independentemente da idade.  
 amostra %>%   
 srvyr::filter(F02 <=2) %>%  
 srvyr::group\_by(freq\_escola) %>%   
 srvyr::summarise(n=survey\_total(vartype = "ci"),  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 2 x 7  
## freq\_escola n n\_low n\_upp pct pct\_low pct\_upp  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Privada 295123. 285724. 304521. 0.378 0.366 0.389  
## 2 Pública 486145. 470285. 502004. 0.622 0.611 0.634

# Faça também um teste de   
 # hipótese a 95% de confiança para comparar se as duas proporções são  
 # equivalentes ou não.  
 svyttest(F02~internet,  
 subset(amostra,F02 <=2 & E02==1),na.rm=T)

##   
## Design-based t-test  
##   
## data: F02 ~ internet  
## t = 2.546, df = 48, p-value = 0.01416  
## alternative hypothesis: true difference in mean is not equal to 0  
## 95 percent confidence interval:  
## 0.05734266 0.44091855  
## sample estimates:  
## difference in mean   
## 0.2491306

##### v) Pela variável G06 (Qual atividade da empresa que lhe paga o trabalho   
 # principal?), caracterize os trabalhadores do setor educação (não   
 # necessariamente todos estão em sala de aula, mas é uma boa proxy do   
 # perfil dos trabalhadores do setor). Calcule o número de pessoas nessas   
 # atividades, o tempo médio de deslocamento ao trabalho e modo de   
 # transporte, além da distribuição etária. Calcule também quantos destes   
 # trabalhadores moram com pessoas com mais de 60 anos.  
   
 #Caracterize os trabalhadores do setor educação - número de pessoas   
 #nessas atividades:  
 amostra %>%  
 # Filtrar População Total  
 srvyr::filter(set\_educacao=="Sim") %>%  
 # Criar contador  
 srvyr::mutate(count=1) %>%  
 # Calcular o total  
 srvyr::summarise("Trabalhadores da Educação"=survey\_total  
 (count, vartype = "ci"))

## # A tibble: 1 x 3  
## `Trabalhadores da Educação` `Trabalhadores da Educação\_low`  
## <dbl> <dbl>  
## 1 90274. 85271.  
## `Trabalhadores da Educação\_upp`  
## <dbl>  
## 1 95277.

#Tempo médio de deslocamento ao trabalho e modo de transporte  
 amostra %>%   
 filter(set\_educacao=="Sim") %>%   
 srvyr::summarise("Tempor médio para o trabalho"=survey\_mean  
 (tempo\_trab\_c,na.rm=T))

## # A tibble: 1 x 2  
## `Tempor médio para o trabalho` `Tempor médio para o trabalho\_se`  
## <dbl> <dbl>  
## 1 25.8 0.542

# Modo de transporte  
 amostra %>%   
 filter(set\_educacao=="Sim" & is.na(transp\_trab)==F) %>%   
 srvyr::mutate(transp\_trab=factor(case\_when(transp\_trab == "Sim" ~"Transporte Coletivo",  
 transp\_trab == "Não" ~"Transporte Privado",  
 transp\_trab == "Não sabe" ~"Não sabe",  
 TRUE ~ NA\_character\_))) %>%  
 srvyr::group\_by(transp\_trab) %>%   
 srvyr::summarise(n=survey\_total(vartype = "ci", na.rm=TRUE),  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 3 x 7  
## transp\_trab n n\_low n\_upp pct pct\_low pct\_upp  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Não sabe 566. 126. 1006. 0.00627 0.00143 0.0111  
## 2 Transporte Coletivo 26415. 23717. 29113. 0.293 0.265 0.320   
## 3 Transporte Privado 63293. 58764. 67821. 0.701 0.674 0.728

# Distribuição etária  
 amostra %>%  
 # Filtrar População Total  
 srvyr::filter(set\_educacao=="Sim") %>%  
 # Criar contador  
 #srvyr::mutate(count=1) %>%  
 srvyr::group\_by(idade\_faixas) %>%   
  
 # Calcular o total  
 srvyr::summarise("Trabalhadores da Educação"=survey\_total  
 (count, vartype = "ci"),  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 14 x 7  
## idade\_faixas `Trabalhadores da Educação` `Trabalhadores da Educação\_low`  
## <ord> <dbl> <dbl>  
## 1 15 a 19 anos 2058. 1353.   
## 2 20 a 24 anos 6689. 5172.   
## 3 25 a 29 anos 11647. 10084.   
## 4 30 a 34 anos 14839. 13063.   
## 5 35 a 39 anos 14062. 12156.   
## 6 40 a 44 anos 13917. 12066.   
## 7 45 a 49 anos 10450. 8779.   
## 8 50 a 54 anos 9038. 7437.   
## 9 55 a 59 anos 4056. 3290.   
## 10 60 a 64 anos 1541. 987.   
## 11 65 a 69 anos 1505. 961.   
## 12 70 a 74 anos 238. 75.9  
## 13 75 a 79 anos 191. 14.2  
## 14 Mais de 85 anos 44.2 -45.0  
## `Trabalhadores da Educação\_upp` pct pct\_low pct\_upp  
## <dbl> <dbl> <dbl> <dbl>  
## 1 2762. 0.0228 0.0152 0.0303   
## 2 8205. 0.0741 0.0575 0.0906   
## 3 13211. 0.129 0.113 0.145   
## 4 16616. 0.164 0.147 0.182   
## 5 15968. 0.156 0.137 0.174   
## 6 15768. 0.154 0.135 0.173   
## 7 12122. 0.116 0.0983 0.133   
## 8 10640. 0.100 0.0845 0.116   
## 9 4821. 0.0449 0.0368 0.0531   
## 10 2095. 0.0171 0.0110 0.0231   
## 11 2049. 0.0167 0.0106 0.0227   
## 12 400. 0.00264 0.000788 0.00448  
## 13 367. 0.00211 0.0000648 0.00416  
## 14 133. 0.000490 -0.000484 0.00146

# Quantos destes trabalhadores moram com pessoas com mais de 60 anos?  
 amostra %>%  
 # Filtrar População Total  
 srvyr::filter(set\_educacao=="Sim" & idoso >=1 ) %>%  
 # Criar contador  
 srvyr::mutate(count=1) %>%  
 # Calcular o total  
 srvyr::summarise("Trabalhadores Edu. que moram com idoso"=survey\_total  
 (count, vartype = "ci", na.rm=TRUE))

## # A tibble: 1 x 3  
## `Trabalhadores Edu. que moram com idoso`  
## <dbl>  
## 1 22985.  
## `Trabalhadores Edu. que moram com idoso\_low`  
## <dbl>  
## 1 20420.  
## `Trabalhadores Edu. que moram com idoso\_upp`  
## <dbl>  
## 1 25550.

##### vi) Estime uma variável de uso do tempo: some o total de horas   
 # trabalhadas, o tempo declarado com deslocamento e o tempo declarado com   
 # afazeres domésticos para o responsável pelo domicílio e pelo cônjuge.   
 # Faça um filtro para domicílios com e sem crianças (menores de 12 anos)   
 # que frequentam escola e por sexo  
 # some o total de horas trabalhadas, o tempo declarado com deslocamento   
 # e o tempo declarado com afazeres domésticos para o responsável pelo   
 # domicílio e pelo cônjuge  
 amostra %>%  
 # Filtrar População Total  
 srvyr::filter(pos\_dom != "Outro") %>%  
 # Agrupar por responsável ou cônjunge  
 srvyr::group\_by(pos\_dom) %>%   
 # Calcular o total  
 srvyr::summarise("Uso do Tempo"=survey\_mean  
 (horas\_trab+tempo\_afazeres+tempo\_trab\_c,   
 vartype = "ci", na.rm=TRUE))

## # A tibble: 2 x 4  
## pos\_dom `Uso do Tempo` `Uso do Tempo\_low` `Uso do Tempo\_upp`  
## <chr> <dbl> <dbl> <dbl>  
## 1 Cônjuge 86.4 85.4 87.3  
## 2 Responsável 84.3 83.7 85.0

# Faça um filtro para domicílios com e sem crianças (menores de 12   
 # anos) que frequentam escola  
   
 #Distrito Federal   
 amostra %>%  
 # Filtrar População DF pelo proprietário da casa  
 srvyr::filter(E02==1) %>%  
 # Criar categorias  
 srvyr::mutate(crianca\_estuda=factor  
 (case\_when(crianca\_estuda == 0 ~"Sem crianças",  
 crianca\_estuda >= 1 ~"Com crianças",  
 TRUE ~ NA\_character\_))) %>%  
 # Agrupar por crianças  
 srvyr::group\_by(crianca\_estuda) %>%   
 # Calcular o total  
 srvyr::summarise("Domicílios"=survey\_total  
 (vartype = "ci", na.rm=TRUE),  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 2 x 7  
## crianca\_estuda Domicílios Domicílios\_low Domicílios\_upp pct pct\_low pct\_upp  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Com crianças 263753. 256653. 270854. 0.299 0.291 0.306  
## 2 Sem crianças 619756. 613524. 625988. 0.701 0.694 0.709

# Domicílios por sexo do responsável  
 #Distrito Federal   
 amostra %>%  
 # Filtrar População DF pelo proprietário da casa  
 srvyr::filter(E02==1) %>%  
 # Agrupar por crianças  
 srvyr::group\_by(sexo) %>%   
 # Calcular o total  
 srvyr::summarise("Domicílios"=survey\_total  
 (vartype = "ci", na.rm=TRUE),  
 pct=survey\_mean(vartype = "ci"))

## # A tibble: 2 x 7  
## sexo Domicílios Domicílios\_low Domicílios\_upp pct pct\_low pct\_upp  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Feminino 403047. 395613. 410481. 0.456 0.448 0.465  
## 2 Masculino 480462. 472403. 488522. 0.544 0.535 0.552