Survey design-informed inference with British Social Attitudes Survey data using R

UK Data Service

April 2024

This exercise is part of the [‘Introduction to the British Social Attitudes Survey (BSA)’](https://trainingmodules.ukdataservice.ac.uk/attitudes/#/) online module. In this exercise, we will practice statistical inference with data from the [British Social Attitudes Survey (BSA) 2017](https://beta.ukdataservice.ac.uk/datacatalogue/studies/study?id=8450) using weights and survey design variables.

Please note that at the time of writing this document only some of the BSA editions include survey design variables. For more information about inference from social surveys, including cases where weights and/or survey design variables are not available, please consult [our guidelines](https://ukdataservice.ac.uk/learning-hub/survey-data/).

Answers to the questions asked throughout the exercise can be found at the end of the page.

### Getting started

Data can be downloaded from the [UK Data Service website](https://beta.ukdataservice.ac.uk/datacatalogue/studies/study?id=8450) following [registration](https://ukdataservice.ac.uk/help/registration/registration-login-faqs/). Download the compressed folder, unzip and save it somewhere accessible on your computer.

The examples below assume that the dataset has been saved in a new folder named *UKDS* on your Desktop (Windows computers). The path would typically be C:\Users\YOUR\_USER\_NAME\Desktop\UKDS. Feel free to change it to the location that best suits your needs

The code below will need to be adjusted in order to match the location of the data on your computer.

We begin by loading the R packages needed for the exercise and set the working directory.

library(dplyr) ### Data manipulation functions  
library(haven) ### Functions for importing data from commercial packages  
library(Hmisc) ### Extra statistical functions  
library(survey) ### Survey design functions  
  
### Setting up the working directory  
### Change the setwd() command to match the location of the data on your computer   
### if required   
  
setwd("C:\Users\Your\_Username\_here\")  
  
getwd()  
  
# Opening the BSA dataset in SPSS format  
bsa17<-read\_spss("data/UKDA-8450-spss/spss/spss25/bsa2017\_for\_ukda.sav")

[1] C:\Users\Your\_Username\_here\

### 1. Identifying the survey design and variables

We first need to find out about the survey design that was used in the BSA 2017, and the design variables available in the dataset. Such information can usually be found in the documentation that comes together with the data under the mrdoc/pdf folder or in the data catalogue pages for the data on the [UK Data Service website](https://beta.ukdataservice.ac.uk/datacatalogue/studies/study?id=8450#!/documentation).

**Question 1**

What is the design that was used in this survey (i.e. how many sampling stages were there, and what were the units sampled). What were the primary sampling units; the strata (if relevant)?

Now that we are a bit more familiar with the way the survey was designed, we need to try and identify the design variables we can include when producing estimates. The information can usually be found in the data documentation or the data dictionary available in the BSA documentation.

**Question 2**

What survey design variables are available? Are there any that are missing – if so which ones? What is the name of the weights variables?

### 2. Specifying the survey design

We need to tell R about the survey design. In practice this often means specifying the units selected at the initial sampling stage ie the *Primary Sampling Units*, as well as the strata. This is achieved with the svydesign() command. In effect this command creates a copy of the dataset with the survey design information attached, that can then subsequently be used for further estimation.

bsa17.s<-svydesign(ids=~Spoint, strata=~StratID, weights=~WtFactor,data=bsa17)  
class(bsa17.s)

[1] "survey.design2" "survey.design"

summary(bsa17.s) ### Warning: very long output

Stratified 1 - level Cluster Sampling design (with replacement)  
With (372) clusters.  
svydesign(ids = ~Spoint, strata = ~StratID, weights = ~WtFactor,   
 data = bsa17)  
Probabilities:  
 Min. 1st Qu. Median Mean 3rd Qu. Max.   
 0.2645 0.8288 1.0983 1.2386 1.6236 3.3318   
Stratum Sizes:   
 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117  
obs 18 22 30 18 16 21 22 37 10 22 19 35 23 19 19 21 25  
design.PSU 2 2 3 2 2 2 2 3 2 3 2 3 2 2 2 2 2  
actual.PSU 2 2 3 2 2 2 2 3 2 3 2 3 2 2 2 2 2  
 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134  
obs 12 12 32 40 25 21 23 26 23 18 34 23 20 29 39 19 30  
design.PSU 2 2 3 3 3 2 2 2 3 2 2 2 2 3 3 2 3  
actual.PSU 2 2 3 3 3 2 2 2 3 2 2 2 2 3 3 2 3  
 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151  
obs 20 10 21 12 26 16 20 17 21 24 30 30 18 29 24 19 28  
design.PSU 2 2 2 2 3 2 2 2 2 3 2 3 2 3 2 3 2  
actual.PSU 2 2 2 2 3 2 2 2 2 3 2 3 2 3 2 3 2  
 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168  
obs 18 8 23 33 14 23 17 39 13 22 16 19 21 18 26 13 14  
design.PSU 2 2 2 3 2 2 2 3 2 2 2 2 2 2 3 2 2  
actual.PSU 2 2 2 3 2 2 2 3 2 2 2 2 2 2 3 2 2  
 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185  
obs 22 20 8 22 31 22 24 19 38 20 29 24 29 21 23 32 36  
design.PSU 2 2 2 2 2 2 2 2 3 2 2 2 3 2 2 3 3  
actual.PSU 2 2 2 2 2 2 2 2 3 2 2 2 3 2 2 3 3  
 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202  
obs 24 22 43 38 38 47 34 15 22 35 17 20 20 21 21 43 35  
design.PSU 3 2 3 3 3 3 3 2 2 3 2 2 2 2 3 3 3  
actual.PSU 3 2 3 3 3 3 3 2 2 3 2 2 2 2 3 3 3  
 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219  
obs 28 25 19 18 28 15 21 30 24 33 24 22 30 24 44 18 26  
design.PSU 3 3 2 2 2 2 2 2 2 3 2 2 3 2 3 2 2  
actual.PSU 3 3 2 2 2 2 2 2 2 3 2 2 3 2 3 2 2  
 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236  
obs 22 28 20 27 34 33 41 24 23 26 17 23 36 20 45 32 27  
design.PSU 2 2 2 3 2 3 3 2 2 2 2 2 3 2 3 3 3  
actual.PSU 2 2 2 3 2 3 3 2 2 2 2 2 3 2 3 3 3  
 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253  
obs 33 25 39 31 29 33 20 43 22 24 26 29 37 22 27 25 43  
design.PSU 3 3 3 3 2 2 2 3 2 2 2 2 3 2 2 2 3  
actual.PSU 3 3 3 3 2 2 2 3 2 2 2 2 3 2 2 2 3  
 254 255 256 257 258 259  
obs 7 32 26 25 28 35  
design.PSU 2 3 2 2 2 3  
actual.PSU 2 3 2 2 2 3  
Data variables:  
 [1] "Sserial" "Spoint" "StratID"   
 [4] "WtFactor" "OldWt" "GOR\_ID"   
 [7] "ABCVer" "Country" "househlde"   
 [10] "hhtypee" "Rsex" "RAgeE"   
 [13] "RAgeCat" "RAgeCat2" "RAgecat3"   
 [16] "RAgecat4" "RAgecat5" "RSexAge"   
 [19] "RSexAge2" "MarStat" "Married"   
 [22] "legmarste" "ChildHh" "nch415e"   
 [25] "nch318e" "hhch04e" "hhch511e"   
 [28] "hhch1215e" "hhch1617e" "rch04e"   
 [31] "rch511e" "rch1215e" "rch1617e"   
 [34] "ownche" "reconacte" "RLastJob"   
 [37] "seconacte" "Readpap" "WhPaper"   
 [40] "paptype" "TVNews" "WebNews"   
 [43] "WNwSite1" "WNwSite2" "SMNews"   
 [46] "Internet" "IntPers" "MedResI"   
 [49] "SupParty" "ClosePty" "PartyIDN"   
 [52] "Partyid1" "PartyId2" "PartyID3"   
 [55] "PtyAlleg" "Idstrng" "Politics"   
 [58] "Coalitin" "ConLabDf" "VoteSyst"   
 [61] "ScotPar2" "ECPolicy2" "GovTrust"   
 [64] "Monarchy" "MiEcono" "MiCultur"   
 [67] "Spend1" "Spend2" "SocSpnd1"   
 [70] "SocSpnd2" "SocSpnd3" "SocSpnd4"   
 [73] "SocSpnd5" "SocSpnd6" "Dole"   
 [76] "TaxSpend" "IncomGap" "SRInc"   
 [79] "CMArran" "RBGaran2" "SepInvol"   
 [82] "SepServ" "WkMent" "WkPhys"   
 [85] "HProbRsp" "PhsRetn" "PhsRecov"   
 [88] "MntRetn" "MntRecov" "HCWork21"   
 [91] "HCWork22" "HCWork23" "HCWork24"   
 [94] "HCWork25" "HCWork26" "HCWork27"   
 [97] "HCWork28" "HCWork29" "NatFrEst"   
[100] "FalseBn2" "RepFrau3" "RepWho1"   
[103] "RepWho2" "RepWho3" "RepWho4"   
[106] "RepWho5" "RepWho6" "RepWho7"   
[109] "RepWho8" "RepWho9" "RepWho10"   
[112] "WhyNRep1" "WhyNRep2" "WhyNRep3"   
[115] "WhyNRep4" "WhyNRep5" "WhyNRep6"   
[118] "WhyNRep7" "WhyNRep8" "WhyNRep9"   
[121] "BFPnsh1" "BFPnsh2" "BFPnsh3"   
[124] "BFPnsh4" "BFPnsh5" "BFPnsh6"   
[127] "BFPnsh7" "BFPnsh8" "BFPnsh9"   
[130] "BFPnsh10" "BFPnsh11" "AwrPB"   
[133] "AdminPn2" "LosofBen" "AwrCRec"   
[136] "GovDoBF" "ImpHDoc" "ImpHPar"   
[139] "ImpHBeha" "ImpHFam" "ImpHEd"   
[142] "ImpHJob" "ImpHNeig" "ImpHArea"   
[145] "ImpHSafe" "RespoHl2" "HomsBult"   
[148] "YSBEmpl" "YSBTrans" "YSBGreen"   
[151] "YSBSch" "YSBAfRnt" "YSBAfOwn"   
[154] "YSBDesig" "YSBShops" "YSBMedic"   
[157] "YSBLibry" "YSBLeis" "YSBFinan"   
[160] "YSBOther" "YSBDeps" "YSBNone"   
[163] "HousGSD" "Buldres" "EdSpnd1c"   
[166] "EdSpnd2c" "VocVAcad" "ATTD151"   
[169] "ATTD152" "ATTD153" "ATTD154"   
[172] "ATTD155" "ATTD156" "ATTD157"   
[175] "ATTD158" "ATTD81" "ATTD82"   
[178] "ATTD83" "ATTD84" "ATTD85"   
[181] "ATTD86" "ATTD87" "ATTD88"   
[184] "GCSEFur" "GCSEWrk" "ALevFur"   
[187] "ALevWrk" "HEdOpp" "ChLikUn2"   
[190] "HEFee" "FeesUni" "FeesSub"   
[193] "Himp" "PREVFR" "TRFPB6U"   
[196] "TRFPB9U" "TrfPb10u" "TrfConc1"   
[199] "DRIVE" "carnume" "CycDang"   
[202] "Bikeown2" "BikeRid" "TRAVEL1"   
[205] "TRAVEL2" "TRAVEL3" "TRAVEL4a"   
[208] "TRAVEL6" "airtrvle" "CCTrans1"   
[211] "CCTrans2" "CCTrans3" "CCTrans4"   
[214] "CCTrans5" "CCTrans6" "CCTrans7"   
[217] "CCTrans8" "CCTrans9" "CCALowE"   
[220] "CCACar" "CCAPLANE" "CCBELIEV"   
[223] "EUBrld" "EUExInf2" "EUExUne2"   
[226] "EUExIm2" "EUExEco2" "EUImpSov"   
[229] "LeavEUI" "EUconte" "EUcontu"   
[232] "EUconth" "EULtop1" "EULtop2"   
[235] "EULtop3" "NHSSat" "WhySat1"   
[238] "WhySat2" "WhySat3" "WhySat4"   
[241] "WhySat5" "WhySat6" "WhySat7"   
[244] "WhySat8" "WhySat9" "WhySat10"   
[247] "WhyDis1" "WhyDis2" "WhyDis3"   
[250] "WhyDis4" "WhyDis5" "WhyDis6"   
[253] "WhyDis7" "WhyDis8" "WhyDis9"   
[256] "WhyDis10" "GPSat" "DentSat"   
[259] "InpatSat" "OutpaSat" "AESat"   
[262] "CareSat3" "NHSFProb" "NHS5yrs"   
[265] "NHSNx5Yr" "NHSAcc" "NHSImp"   
[268] "AEtravel" "CareNee2" "PaySocia"   
[271] "CarePa2" "SocFutur" "Tranneed"   
[274] "Prejtran" "PMS" "HomoSex"   
[277] "SSRel" "RSuperv" "rocsect2e"   
[280] "REmpWork" "REmpWrk2" "SNumEmp"   
[283] "WkJbTim" "ESrJbTim" "SSrJbTim"   
[286] "WkJbHrsI" "ExPrtFul" "EJbHrCaI"   
[289] "SJbHrCaI" "RPartFul" "S2PartFl"   
[292] "Remplyee" "UnionSA" "TUSAEver"   
[295] "NPWork10" "RES2010" "RES2000"   
[298] "SLastJb2" "S2Employ" "S2Superv"   
[301] "S2ES2010" "S2ES2000" "rjbtype"   
[304] "REconSum" "REconPos" "RNSEGGrp"   
[307] "RNSocCl" "RNSSECG" "RClass"   
[310] "RClassGp" "RSIC07GpE" "seconsum"   
[313] "S2NSEGGp" "S2NSSECG" "S2NSocCl"   
[316] "S2Class" "S2ClassG" "WAGMIN"   
[319] "RESPPAY" "TRCURJM" "TRCURJN"   
[322] "TRMRSJM" "TRMRSJN" "TRDIFJM"   
[325] "TRDIFJN" "PHOURS" "REGHOUR"   
[328] "WRKCON" "JBMRESP" "JBMWH1"   
[331] "JBMWH2" "JBMWH3" "JBMWH4"   
[334] "JBMWH5" "JBMWH6" "JBMWH7"   
[337] "JBMWH8" "FLEXHRS" "MgCWld"   
[340] "MgMWld" "ChgAsJb1" "ChgAsJb2"   
[343] "ChgAsJb3" "ChgJbTim" "RetExp"   
[346] "RetExpb" "DVRetAge" "PenKnow2"   
[349] "RPenSrc1" "RPenSrc2" "RPenSrc3"   
[352] "whrbrne" "NatIdGB" "NatId"   
[355] "tenure2e" "RentPrf1" "HAWhat"   
[358] "HAgdbd" "HANotFM" "LikeHA"   
[361] "HAYwhy" "HANwhy" "HsDepnd"   
[364] "ResPres" "ReligSum" "RlFamSum"   
[367] "ChAttend" "bestnatu2" "raceori4"   
[370] "DisNew2" "DisAct" "DisActDV"   
[373] "Knowdis1" "Knowdis2" "Knowdis3"   
[376] "Knowdis4" "Knowdis5" "Knowdis6"   
[379] "Knowdis7" "DisPrj" "Dis100"   
[382] "tea3" "HEdQual" "HEdQual2"   
[385] "HEdQual3" "EUIdent" "BritID2"   
[388] "Voted" "Vote" "EURefV2"   
[391] "EUVOTWHO" "EURefb" "AnyBN3"   
[394] "MainInc5" "HHIncD" "HHIncQ"   
[397] "REarnD" "REarnQ" "SelfComp"   
[400] "knwbdri" "knwexec" "knwclea"   
[403] "knwhair" "knwhr" "knwlaw"   
[406] "knwmech" "knwnurs" "knwpol"   
[409] "knwtchr" "incdiffs" "incdsml"   
[412] "govldif" "socblaz" "whoprvhc"   
[415] "whoprvca" "actgrp" "actpol"   
[418] "actchar" "govnosa2" "hhldjob"   
[421] "hhmsick" "hdown" "hadvice"   
[424] "hsococc" "hlpmny" "hlpjob"   
[427] "hlpadmin" "hlplive" "hlpill"   
[430] "lckcomp" "isolate" "leftout"   
[433] "peopadvt" "peoptrst" "trstcrts"   
[436] "trstprc" "helpeldy" "helpslf1"   
[439] "helpfrnd" "fampress" "reltdemd"   
[442] "ffrangr" "eatout" "newfrnd"   
[445] "pplcont" "pplftf" "parcont"   
[448] "sibcon2" "chdcon2" "othcont"   
[451] "frndcont" "contint" "ltsgnhth"   
[454] "depres" "diffpile" "acgoals"   
[457] "lifesat2" "makeem" "langgs"   
[460] "helpslf2" "payback" "domconv"   
[463] "sitwhr" "hmecont" "religcon"   
[466] "spseedu" "ben3000" "ben3000d"   
[469] "falcatch" "uniaff" "unicar"   
[472] "bothearn" "sexrole" "womworka"   
[475] "womworkb" "parlvmf2" "gendwrk"   
[478] "gendmath" "gendcomp" "sxbstrm"   
[481] "sxbintm" "sxbstrw" "sxbintw"   
[484] "sxblaw" "sxbprov" "sxboffb"   
[487] "sxbnoone" "sxboth" "sxbcc"   
[490] "carwalk2" "carbus2" "carbike2"   
[493] "shrtjrn" "plnallow" "plnterm"   
[496] "plnenvt" "plnuppri" "cartaxhi"   
[499] "carallow" "carreduc" "carnod2"   
[502] "carenvdc" "resclose" "res20mph"   
[505] "resbumps" "ddnodrv" "ddnklmt"   
[508] "specamsl" "specammo" "specamtm"   
[511] "speedlim" "speavesc" "mobdsafe"   
[514] "mobddang" "mobdban" "mobdlaw"   
[517] "eutrdmv" "consvfa" "labrfa"   
[520] "libdmfa" "ukipfa" "rthdswa2"   
[523] "rthdsaw2" "rthdsca2" "rthdssa2"   
[526] "rthdsprd" "eqrdisab" "nhsoutp2"   
[529] "nhsinp2" "bodimr" "bodimop"   
[532] "girlwapp" "tprwrong2" "eulunem"   
[535] "eulimm" "eulecon" "eulwork"   
[538] "eullowi" "eulmlow" "eulnhs"   
[541] "jbernmny" "jbenjoy" "topupchn"   
[544] "topupnch" "topuplpa" "worknow"   
[547] "losejob" "jbgdcurr" "robots"   
[550] "robown" "voteduty" "welfhelp"   
[553] "morewelf" "unempjob" "sochelp"   
[556] "dolefidl" "welffeet" "damlives"   
[559] "proudwlf" "redistrb" "BigBusnn"   
[562] "wealth" "richlaw" "indust4"   
[565] "tradvals" "stifsent" "deathapp"   
[568] "obey" "wronglaw" "censor"   
[571] "leftrigh" "libauth" "welfare2"   
[574] "libauth2" "leftrig2" "welfgrp"   
[577] "eq\_inc\_deciles" "eq\_inc\_quintiles" "eq\_bhcinc2\_deciles"   
[580] "eq\_bhcinc2\_quintiles"

### 3. Mean age and its 95% confidence interval

We can now produce a first set of estimates using this information and compare them with those we would have got without accounting for the survey design. We will compute the average (ie mean) age of respondents in the sample. We will need to use svymean()

svymean(~RAgeE,bsa17.s)

mean SE  
RAgeE 48.313 0.4236

By default svymean() computes the standard error of the mean. We need to  
embed it within confint() in order to get a confidence interval.

confint(svymean(~RAgeE,bsa17.s)) ### Just the confidence interval...

2.5 % 97.5 %  
RAgeE 47.48289 49.1433

round(  
 c(  
 svymean(~RAgeE,bsa17.s),  
 confint(svymean(~RAgeE,bsa17.s))  
 ),  
 1) ### ... Or both, rounded

RAgeE   
 48.3 47.5 49.1

*What difference would it make to the estimates and 95% CI to compute respectively, an unweighted mean, as well as a weighted mean without accounting for the survey design?*

There are different ways of computing ‘naive estimates’ in R. Below we demonstrate how to do it ´by hand’ for greater transparency.

Base R provides a function for computing the variance of a variable: var(). Since we know that:

* The standard deviation of the mean is the square root of its variance
* The standard error of a sample mean is its standard deviation divided by the square root of the sample size
* A 95% confidence interval is the sample mean respectively minus and plus 1.96 times its standard error. It is then relatively straightforward to compute unweighted and ‘casually weighted’ confidences intervals for the mean.

### Unweighted means and CI  
u.m<- mean(bsa17$RAgeE)  
u.se<-sqrt(var(bsa17$RAgeE))/sqrt(length(bsa17$RAgeE))  
u.ci<-c(u.m - 1.96\*u.se,u.m + 1.96\*u.se)  
round(c(u.m,u.ci),1)

[1] 52.2 51.6 52.8

### Weighted means and CI without survey design  
w.m<- wtd.mean(bsa17$RAgeE,bsa17$WtFactor)  
w.se<-sqrt(wtd.var(bsa17$RAgeE,bsa17$WtFactor))/sqrt(length(bsa17$RAgeE))  
w.ci<-c(w.m - 1.96\*w.se,w.m + 1.96\*w.se)  
round(c(w.m,w.ci),1)

[1] 48.3 47.7 48.9

**Question 3** What are the consequences of not accounting for the sample design; not using weights and accounting for the sample design when: - inferring the mean value of the population age? - inferring the uncertainty of our estimate of the population age?

### 4. Computing a proportion and its 95% confidence interval

We can now similarly estimate the distribution of a categorical variable in the population by computing proportions (or percentages), for instance, the proportion of people who declare themselves interested in politics. This is the Politics variable. It has five categories that we are going to recode into ‘Significantly’ (interested) and ‘Not’ (significantly), for simplicity. The BSA regards ‘don’t know’ and ‘refusal’ responses as valid but in this case there is only one ‘don’t know’ and no ‘refusal’ responses, so we can safely ignore these categories.

attr(bsa17$Politics,"label") ### Phrasing of the question

[1] "How much interest do you have in politics?"

table(as\_factor(bsa17$Politics)) ### Sample distribution

skip, version off route Item not applicable ... a great deal,   
 0 0 739   
 quite a lot, some, not very much,   
 982 1179 708   
 or, none at all? Don`t know Refusal   
 379 1 0

bsa17$Politics.s<-ifelse(bsa17$Politics==1 | bsa17$Politics==2,  
 "Significantly",NA)  
bsa17$Politics.s<-ifelse(bsa17$Politics>=3 & bsa17$Politics<=5,  
 "Not Interested",bsa17$Politics.s)  
bsa17$Politics.s<-as.factor(bsa17$Politics.s)  
rbind(table(bsa17$Politics.s),  
 round(100\*prop.table(table(bsa17$Politics.s)),1)  
)

Not Interested Significantly  
[1,] 2266.0 1721.0  
[2,] 56.8 43.2

Changes in a data frame are not automatically transferred into svydesign objects used for inferences. We therefore need to recreate it each time we create or recode a variable.

rbind(round(wtd.table(bsa17$Politics.s,bsa17$WtFactor)$sum.of.weights,1),  
 round(100\*prop.table(wtd.table(bsa17$Politics.s,bsa17$WtFactor)$sum.of.weights),1)  
)

[,1] [,2]  
[1,] 2270.6 1715.2  
[2,] 57.0 43.0

bsa17.s<-svydesign(ids=~Spoint, strata=~StratID, weights=~WtFactor,data=bsa17)  
  
rbind(round(svytable(~Politics.s,bsa17.s),1),  
 round(100\*prop.table(svytable(~Politics.s,bsa17.s)),1)  
)

Not Interested Significantly  
[1,] 2270.6 1715.2  
[2,] 57.0 43.0

As with the mean of age earlier, we can see that the weighted and unweighted point estimates of the proportion of respondents significantly interested in politics differ, even if slightly, and that weighted point estimates do not differ irrespective of the survey design being accounted for.

Let us now examine the confidence intervals of these proportions. Traditional statistical software usually compute these without telling us about the underlying computations going on. By contrast, doing this in R requires more coding, but in the process we gain a better understanding of what is actually estimated.

Confidence intervals for proportion of categorical variables are usually computed as a sequence of binomial/dichotomic estimations – ie one for each category. In R this needs to be specified explicitly via the svyciprop() and I() functions. The former actually computes the proportion and its confidence interval (by default 95%), whereas the latter allows us to define the category we are focusing on (in case of non dichotomic variable).

svyciprop(~I(Politics.s=="Significantly"),bsa17.s)

2.5% 97.5%  
I(Politics.s == "Significantly") 0.430 0.411 0.450

round(100\*  
 c(prop.table(svytable(~Politics.s,bsa17.s))[2],  
attr(svyciprop(~I(Politics.s=="Significantly"),bsa17.s),"ci")),1  
)

Significantly 2.5% 97.5%   
 43.0 41.1 45.0

**Question 4**

What is the proportion of respondents aged 17-34 in the sample, as well as its 95% confidence interval? You can use RAgecat5

### 5. Domain (ie subpopulation) estimates

Computing estimates for specific groups of a sample (for example the average age of people who reported being interested in politics) is not much more difficult than doing it for the sample as a whole. However doing it as part of an inferential analysis requires some caution. Calculating weighted estimates for a subpopulation, amounts to computing second order estimates ie an estimate for a group whose size needs to be estimated first. Therefore, attempting this while leaving out of the rest of the sample might yield incorrect results. This is why using survey design informed functions is particularly recommended in such cases.

The survey package functionsvyby() makes such domain estimation relatively straightforward. For instance, if we would like to compute the mean age of BSA respondents by Government Office Regions, we need to specify:

* The outcome variable whose estimate we want to compute: ie RAgeE
* The grouping variable(s) GOR\_ID
* The estimate function we are going to use here: svymean, the same as we used before
* And the type of type of variance estimation we would like to see displayed ie standard errors or confidence interval

bsa17$gor.f<-as\_factor(bsa17$GOR\_ID)  
bsa17.s<-svydesign(ids=~Spoint, strata=~StratID, weights=~WtFactor,data=bsa17)  
  
round(svyby(~RAgeE,by=~gor.f,svymean,design=bsa17.s,vartype = "ci")[-1],1)

RAgeE ci\_l ci\_u  
A North East 46.1 43.6 48.6  
B North West 49.6 47.3 52.0  
D Yorkshire and The Humber 48.0 45.2 50.8  
E East Midlands 48.6 45.9 51.3  
F West Midlands 48.1 45.0 51.2  
G East of England 49.0 46.0 52.0  
H London 45.0 43.0 46.9  
J South East 48.0 45.1 50.8  
K South West 53.4 51.5 55.2  
L Wales 49.1 45.1 53.1  
M Scotland 47.3 44.7 50.0

*Note:* we used [-1] from the object created by svyby() in order to remove a column with alphanumeric values (the region names), so that we could round the results without getting an error.

Our inference seem to suggest that the population in London is among the youngest in the country, and that those in the South West are among the oldest – their respective 95% confidence intervals do not overlap. We should not feel so confident about differences between London and the South East for example, as the CIs partially overlap.

We can follow a similar approach with proportions: we just need to specify the category of the variable we are interested in as an outcome, for instance respondents who are significantly interested in politics, and replace svymean by svyciprop.

round(  
 100\*  
 svyby(~I(Politics.s=="Significantly"),  
 by=~gor.f,  
 svyciprop,  
 design=bsa17.s,  
 vartype = "ci")[-1],  
 1)

I(Politics.s == "Significantly") ci\_l ci\_u  
A North East 33.4 26.6 40.9  
B North West 42.1 36.3 48.2  
D Yorkshire and The Humber 35.6 29.1 42.6  
E East Midlands 36.9 32.9 41.1  
F West Midlands 36.3 31.5 41.5  
G East of England 47.2 41.4 53.1  
H London 54.2 47.2 61.1  
J South East 44.6 38.7 50.8  
K South West 46.5 39.4 53.8  
L Wales 38.6 27.7 50.7  
M Scotland 42.7 36.0 49.8

**Question 5**

What is the 95% confidence interval for the proportion of people interested in politics in the South West? Is the proportion likely to be different in London? In what way? What is the region of the UK for which the precision of the estimates is likely to be the smallest?

When using svyby(), we can define domains or subpopulations with several variables, not just one. For example, we could have looked at gender differences in political affiliations by regions. However, as the size of subgroups decrease, so does the precision of the estimates as their confidence interval widens, to a point where their substantive interest is not meaningful anymore.

**Question 6**

Using interest in politics as before, and three category age RAgecat5 (which you may want to recode as a factor in order to improve display clarity):

* Produce a table of results showing the proportion of respondents significantly interested in Politics by age group
* Assess whether the age difference in interest for politics is similar for each gender?
* Based on the data, is it fair to say that men aged under 35 tend to be more likely to declare themselves interested in politics than women aged 55 and above?

### Answers

**Question 1** The 2017 BSA is a three stage stratified random survey, with postcode sectors, adresses and individuals as the units selected at each stage. Primary sampling units were furthermore stratified according to geographies (sub regions), population density, and proportion of owner-occupiers. Sampling rate was proportional to the size of postcode sectors (ie number of addresses)

**Question 2** From the Data Dictionary it appears that the primary sampling units (sub regions) are identified bySpoint and the strata byStratID. The weights variable isWtFactor. Addresses are not provided but could be approximated with a household identifier.

**Question 3** Not using weights would make us overestimate the mean age in the population (of those aged 16+) by about 4 years. This is likely to be due to the fact that older respondents are more likely to take part to surveys. Using survey design variables does not alter the value of the estimated population mean. However, not accounting for them would lead us to overestimate the precision/underestimate the uncertainty of our estimate with a narrower confidence interval – by about plus and minus 2 months .

**Question 4** The proportion of 17-25 year old in the sample is 28.5 and its 95%confidence interval 26.5, 30.6

**Question 5** The 95% confidence interval for the proportion of people interested in politics in the South West is 39.4, 53.8. By contrast, it is likely to be 47.2, 61.1 in London. The region with the lowest precision of estimates (ie the widest confidence interval) is Wales, with a 23 percentage point difference between the upper and lower bounds of the confidence interval.

**Question 6**

bsa17$RAgecat5.f<-as\_factor(bsa17$RAgecat5)  
bsa17$Rsex.f<-as\_factor(bsa17$Rsex)  
  
  
bsa17.s<-svydesign(ids=~Spoint, strata=~StratID, weights=~WtFactor,data=bsa17)  
  
round(  
 100\*  
 svyby(~I(Politics.s=="Significantly"),  
 by=~RAgecat5.f+Rsex.f,  
 svyciprop,  
 design=bsa17.s,  
 vartype = "ci")[c(-8,-4),c(-2,-1)],  
 1)

I(Politics.s == "Significantly") ci\_l ci\_u  
17-34.Male 42.9 37.7 48.2  
35-54.Male 50.8 46.6 54.9  
55+.Male 57.8 53.9 61.6  
17-34.Female 26.3 22.0 31.1  
35-54.Female 34.1 30.6 37.8  
55+.Female 43.0 39.6 46.5

Older respondents both male and female tend to be more involved in politics than younger ones.

The confidence interval for the proportion of men under 35 and women above 55 interested in politics overlaps; it is unlikely that they differ in the population.