

Survey Data Analysis

Ehsan Karim

2020-09-01

Contents

1	About this Text	5
2	Model-based Approach	7
2.1	Example	7
2.2	Research question	7
2.3	Data	8
2.4	Checking assumptions	11
2.5	Analysis	13
2.6	Verdict	19
2.7	Exercises (Optional)	20
3	Design-based Approach	21
3.1	Sampling	21
3.2	Statistical inference	23
3.3	Complex surveys	24
3.4	Further readings	25
3.5	Exercise	26
4	Importing CCHS to R	27
4.1	Downloading CCHS data from UBC	27
4.2	Reading and Formatting the data	33
4.3	Processing data in R	45

5	Importing NHANES to R	49
5.1	NHANES Dataset	49
5.2	Accessing NHANES Data	49

Chapter 1

About this Text

This is the 3rd run of this rather newly developed course. The purpose of this course is to provide students with learning opportunities to understand fundamental epidemiological concepts through the application of methods using population and public health datasets. The purpose is also to introduce students to emerging epidemiological methodologies that are frequently being applied to population and public health-related research questions in prestigious epidemiology journal publications.

Chapter 2

Model-based Approach

Review of regression analysis and ANOVA from pre-requisites (+ some extra concepts). Below we see an example of a random data generating process that depends on specification of a probability model. We assume that the population data was generated from a **Normal distribution**, and we are merely dealing with a sample. All our inferences (point estimate or hypothesis testing) will depend on how closely the data fulfill such assumption. We call such approach as ‘**model-based**’ approach.

2.1 Example

Does plant weight increase with added nutrition?

The following problem was taken from Exercise set 2.5 (2.1) from Dobson and Barnett (2008):

“Genetically similar seeds are randomly assigned to be raised in either a nutritionally enriched environment (treatment group) or standard conditions (control group) using a completely randomized experimental design. After a predetermined time all plants are harvested, dried and weighed.”

2.2 Research question

We want to test whether there is any difference in yield (weight) between the two groups

- plants from nutritionally enriched environment (treatment group) and
- plants from standard conditions (control group)

2.2.1 Notations

1. Let k be the index of each plant, and $k = 1, \dots, 20$ for both groups.
2. Let j be the index for groups. Here, $j = 1$ for the treatment group (**Trt**), $j = 2$ for the control group (**Ctl**).
3. Let Y_{jk} denote the k th observation of weights in the j th group.

2.2.2 Assumptions

1. Assume that the Y_{jk} 's are independent random variables with $Y_{jk} \sim N(\mu_j, \sigma^2)$.
2. We also assume that the variances are homogenous, that is, σ_1^2 and σ_2^2 are not very different (and could be pooled to one single value of σ^2).

2.2.3 Hypothesis

The null hypothesis $H_0 : \mu_1 = \mu_2 = \mu$, that there is no difference, is to be compared with the alternative hypothesis $H_1 : \mu_1 \neq \mu_2$.

2.3 Data

2.3.1 Data table

```
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
length(ctl);length(trt)
```

```
## [1] 10
```

```
## [1] 10
```

```
group <- rep(c("Ctl","Trt"), each = length(ctl))
group
```

```
## [1] "Ctl" "Ctl" "Ctl" "Ctl" "Ctl" "Ctl" "Ctl" "Ctl" "Ctl" "Ctl" "Ctl" "Trt" "Trt"
## [13] "Trt" "Trt" "Trt" "Trt" "Trt" "Trt" "Trt" "Trt" "Trt"
```



```
mode(group)
```

```
## [1] "character"
```

```
weight <- c(ctl, trt)
weight
```

```
## [1] 4.17 5.58 5.18 6.11 4.50 4.61 5.17 4.53 5.33 5.14 4.81 4.17 4.41 3.59 5.87
## [16] 3.83 6.03 4.89 4.32 4.69
```

```
mode(weight)
```

```
## [1] "numeric"
```

```
Plant.Weight.Data <- data.frame(group=group, weight = c(ctl, trt))
mode(Plant.Weight.Data)
```

```
## [1] "list"
```

```
dim(Plant.Weight.Data)
```

```
## [1] 20 2
```

```
str(Plant.Weight.Data)
```

```
## 'data.frame': 20 obs. of 2 variables:
## $ group : chr "Ctl" "Ctl" "Ctl" "Ctl" ...
## $ weight: num 4.17 5.58 5.18 6.11 4.5 4.61 5.17 4.53 5.33 5.14 ...
```

The results, expressed in grams, for 20 plants in each group are shown in the following Table.

```
library(DT)
```

```
## Warning: package 'DT' was built under R version 4.0.2
```

```
datatable(Plant.Weight.Data)
```

Show 10 entries Search

	group	id	weight
1	Ctl		4.17
2	Ctl		5.58
3	Ctl		5.18
4	Ctl		6.11
5	Ctl		4.5
6	Ctl		4.61
7	Ctl		5.17
8	Ctl		4.53
9	Ctl		5.33
10	Ctl		5.14

Showing 1 to 10 of 20 entries Previous 1 2 Next

2.3.2 Visualization

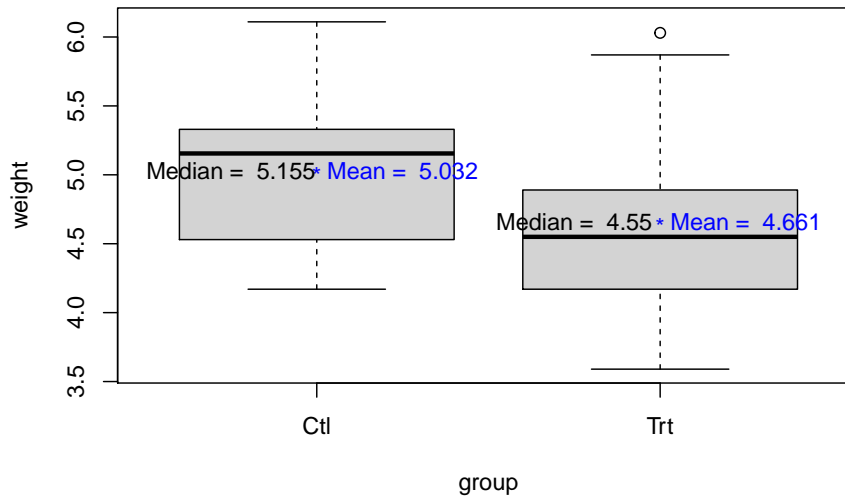
```
boxplot(weight ~ group, data=Plant.Weight.Data)
weight.means <- aggregate(weight ~ group, data=Plant.Weight.Data, FUN=mean)
weight.means
```

```
##   group weight
## 1   Ctl  5.032
## 2   Trt  4.661
```

```
weight.medians <- aggregate(weight ~ group, data=Plant.Weight.Data, FUN=median)
weight.medians
```

```
##   group weight
## 1   Ctl  5.155
## 2   Trt  4.550
```

```
points(1:2, weight.means$weight, pch = "*", col = "blue")
text(c(1:2)+0.25, weight.means$weight, labels =
      paste("Mean = ", weight.means$weight), col = "blue")
text(c(1:2)-0.25, weight.means$weight, labels =
      paste("Median = ", weight.medians$weight), col = "black")
```



Wait: so, plant weight reduces as we add nutrition? How confidently can we say that this added nutrition is harmful for the plants (e.g., so that the weight will be reduced)?

2.4 Checking assumptions

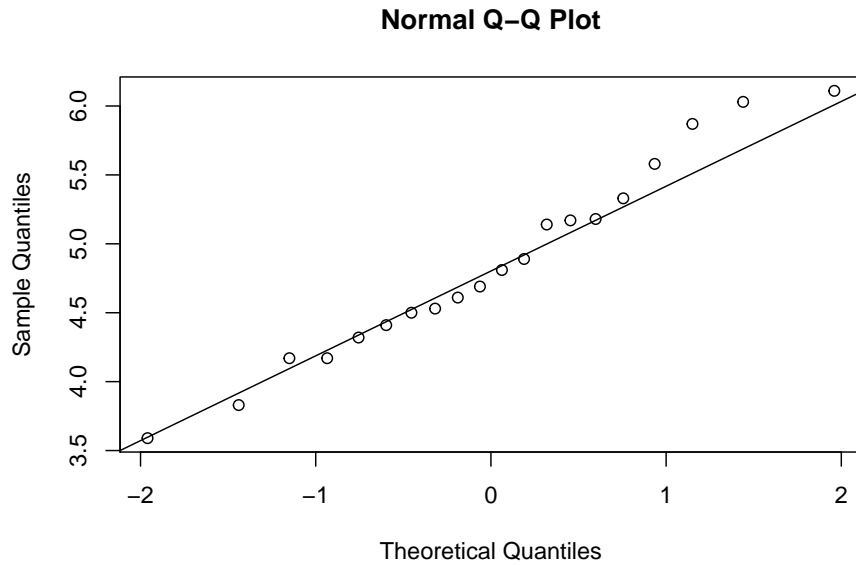
Test of normality of the outcomes (Shapiro-Wilk normality test):

```
shapiro.test(Plant.Weight.Data$weight)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Plant.Weight.Data$weight
## W = 0.97311, p-value = 0.8187
```

Therefore, we cannot reject the null hypothesis that samples come from a population which has a normal distribution. Also check a normal quantile-quantile plot:

```
qqnorm(Plant.Weight.Data$weight)
qqline(Plant.Weight.Data$weight)
```



Test of homogeneity of variances, that tests $H_0 : \sigma_1 = \sigma_2$ vs. $H_1 : \sigma_1 \neq \sigma_2$:

```
# SD from each groups
tapply(Plant.Weight.Data$weight,
      INDEX = Plant.Weight.Data$group, FUN = sd)
```

```
##          Ctl          Trt
## 0.5830914 0.7936757
```

```
bartlett.test(weight ~ group, data = Plant.Weight.Data) # Bartlett's test
```

```
##
## Bartlett test of homogeneity of variances
##
## data:  weight by group
## Bartlett's K-squared = 0.79805, df = 1, p-value = 0.3717
```

```
# leveneTest(weight ~ group, data = Plant.Weight.Data) # Levene's test
```

2.5 Analysis

2.5.1 Two-sample t-test

A two-sample (independent) t-test compares the weights of control and treatment group as follows (assuming equal variance; judging from the IQR from the boxplots or the above Bartlett test):

```
ttest<- t.test(weight ~ group, data = Plant.Weight.Data,
               paired = FALSE, var.equal = TRUE)
ttest

##
## Two Sample t-test
##
## data: weight by group
## t = 1.1913, df = 18, p-value = 0.249
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.2833003 1.0253003
## sample estimates:
## mean in group Ctl mean in group Trt
##           5.032           4.661
```

Here, we test $H_0 : \mu_1 = \mu_2 = \mu$ vs. $H_1 : \mu_1 \neq \mu_2$.

```
ttest$statistic
```

```
##           t
## 1.19126
```

2.5.2 Regression

A simple linear model exploring the relationship between the plant weight and the group status can be fitted as follows:

```
lm.group.including.intercept <- lm(weight ~ 1 + group, data = Plant.Weight.Data)
lm.group.including.intercept

##
## Call:
## lm(formula = weight ~ 1 + group, data = Plant.Weight.Data)
```

```
##
## Coefficients:
## (Intercept)      groupTrt
##          5.032        -0.371

lm.group <- lm(weight ~ group, data = Plant.Weight.Data)
lm.group

##
## Call:
## lm(formula = weight ~ group, data = Plant.Weight.Data)
##
## Coefficients:
## (Intercept)      groupTrt
##          5.032        -0.371

confint(lm.group)

##                2.5 %    97.5 %
## (Intercept)  4.56934  5.4946602
## groupTrt    -1.02530  0.2833003
```

2.5.2.1 Interpretation

Note that the variable `group` is dummy coded. R generally chooses the first category as the reference category.

```
levels(Plant.Weight.Data$group)
```

```
## NULL
```

1. In this case, the intercept 5.032 tells us the predicted mean value for the plant weights for the control group (reference category of the group variable).
2. On the other hand, the slope is interpreted as the expected difference in the mean of the plant weights for that treatment group as compared to the control group. On average, weight is 0.371 units (lb?) lower in plants who are in the treatment condition compared to those in the control condition.

2.5.2.2 Summary of the regression fit

The complete summary of the results is as follows:

```
summary(lm.group)
```

```
##
## Call:
## lm(formula = weight ~ group, data = Plant.Weight.Data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0710 -0.4938  0.0685  0.2462  1.3690
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.0320     0.2202  22.850 9.55e-15 ***
## groupTrt       -0.3710     0.3114  -1.191   0.249
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6964 on 18 degrees of freedom
## Multiple R-squared:  0.07308, Adjusted R-squared:  0.02158
## F-statistic: 1.419 on 1 and 18 DF,  p-value: 0.249
```

This is testing a different hypothesis (from the table): $H_0 : \alpha = 0$ vs. $H_1 : \alpha \neq 0$ (α being the intercept) and $H_0 : \beta = 0$ vs. $H_1 : \beta \neq 0$ (β being the slope). At the bottom of the `summary` output, the F-statistic tests $H_0 : \beta = 0$ vs. $H_1 : \beta \neq 0$. This is an overall, and could accomodate more slopes if the regression had more slopes. E.g., for 2 slopes, this would have tested $H_0 : \beta_1 = \beta_2 = 0$.

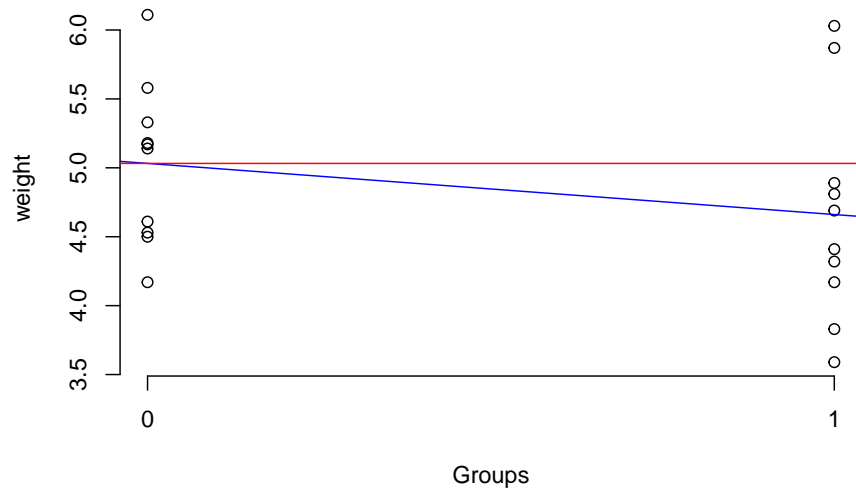
2.5.2.3 Regression plot

Let us visualize the scatter plot and the regression line:

```
Plant.Weight.Data$group.code <-
  ifelse(Plant.Weight.Data$group == "Trt", 1, 0)
Plant.Weight.Data$group.code

## [1] 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1

lm.code <- lm(weight ~ group.code, data = Plant.Weight.Data)
plot(weight ~ group.code, data = Plant.Weight.Data,
      axes = FALSE, xlab = "Groups")
axis(1, 0:1, levels(Plant.Weight.Data$group))
axis(2)
abline(lm.code, col = "blue") # regression line
abline(h=coef(lm.code)[1], col = "red") # intercept
```



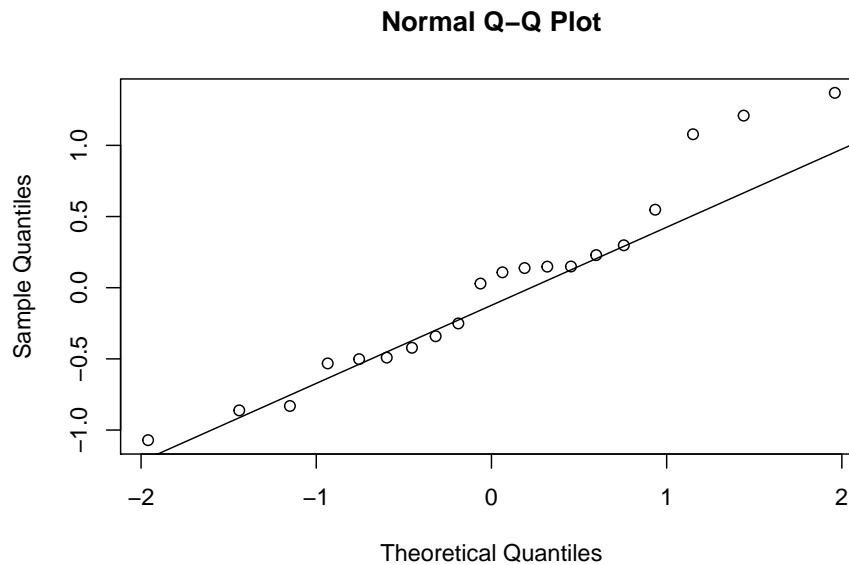
2.5.2.4 Assumption checking for the residuals

Checking normality of the residuals:

```
lm.residual <- residuals(lm.group)
shapiro.test(lm.residual)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  lm.residual
## W = 0.94744, p-value = 0.3299
```

```
qqnorm(lm.residual)
qqline(lm.residual)
```

2.5.2.5 Null model

A null model (with only intercept):

```
lm.null <- lm(weight ~ 1, data = Plant.Weight.Data) # Including just the intercept
summary(lm.null)
```

```
##
## Call:
## lm(formula = weight ~ 1, data = Plant.Weight.Data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2565 -0.4590 -0.0965  0.3710  1.2635
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.8465     0.1574   30.79  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.704 on 19 degrees of freedom
```

2.5.3 ANOVA

For testing for the significance of the group membership, we can compare the current model to the null model (is adding the variable `group` in the model useful?).

```
anova(lm.null, lm.group)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ 1
## Model 2: weight ~ group
##   Res.Df    RSS Df Sum of Sq    F Pr(>F)
## 1      19  9.4175
## 2      18  8.7292   1   0.68821 1.4191  0.249
```

Or, we could directly test $H_0 : \mu_1 = \mu_2 = \mu$ vs. $H_1 : \mu_1 \neq \mu_2$ under the homogeneity of variances assumption:

```
anova(lm.group)
```

```
## Analysis of Variance Table
##
## Response: weight
##           Df Sum Sq Mean Sq F value Pr(>F)
## group      1  0.6882  0.68820   1.4191  0.249
## Residuals 18  8.7292  0.48496
```

```
# Alternate ways to do the same
# car::Anova(lm.group, type="II")
aov.fit <- aov(lm.group)
summary(aov.fit)
```

```
##           Df Sum Sq Mean Sq F value Pr(>F)
## group      1   0.688   0.6882   1.419  0.249
## Residuals 18   8.729   0.4850
```

```
# Multiple pairwise-comparison:
# (compare with t-test; same p-value?)
TukeyHSD(aov.fit)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
```

```
##
## Fit: aov(formula = lm.group)
##
## $group
##      diff      lwr      upr      p adj
## Trt-Ctl -0.371 -1.0253 0.2833003 0.2490232
```

Checking normality of the residuals (not run; same as above):

```
# aov.residual <- residuals(aov.fit)
# shapiro.test(aov.residual)
# qqnorm(aov.residual)
# qqline(aov.residual)
```

ANOVA is basically a generalization of the two-sample t-test (verify that the calculated $F = t^2$):

```
ttest$statistic^2
```

```
##      t
## 1.419101
```

An alternative non-parametric version of this independent 2-sample test is as follows (a Kruskal-Wallis rank sum test):

```
# Assuming groups come from similar shaped populations:
kruskal.test(weight ~ group, data = Plant.Weight.Data)
```

```
##
## Kruskal-Wallis rank sum test
##
## data: weight by group
## Kruskal-Wallis chi-squared = 1.7513, df = 1, p-value = 0.1857
```

2.6 Verdict

2.6.1 Informal conclusion

With added nutrition, plant weights generally decrease (judging from the point estimate), but such trend could be due to sampling fluctuation (e.g., as the 95% confidence interval includes the null value of 0) and we can not confidently (not at least with 95% confidence) say that adding nutrition will cause plant weights to go down.

2.6.2 A word of caution

Note that, we are inherently trying to infer ‘causality’ out of a statistical analysis, even though our hypothesis is not about ‘cause’ explicitly. Unfortunately, correlation does not imply causation, and we need to know more about the subject-area and study-design before we make such inference or interpretation.

2.7 Exercises (Optional)

1. What is the difference between a regression analysis with a dummy coded predictor variable vs. an ANOVA?
2. Was multiple pairwise-comparison (**TukeyHSD**) necessary in the above example?
3. Which R package includes the **leveneTest** function? (hint: use **help.search()** function.)
4. Is ‘multicollinearity’ an issue in the above example?
5. In the current example, can we interpret the slope as follows: **the change in Y for a 1-unit change in X** where, Y being the outcome and X being the predictor? Why, or why not?

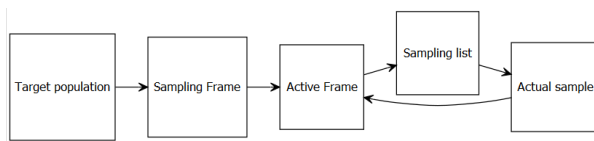
Chapter 3

Design-based Approach

Before discussing design-based approach, let us review some of concepts related to **sampling**.

3.1 Sampling

3.1.1 Steps of generalization



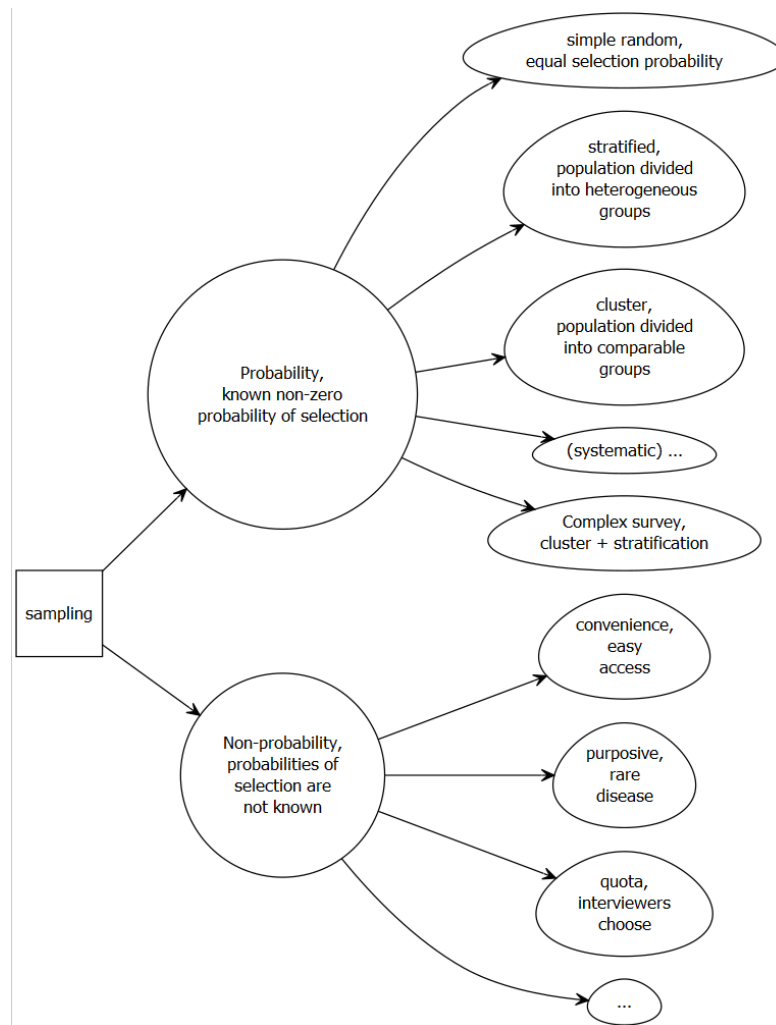
Example: Let us consider CCHS.

- Target population
 - Canadian population 12 years of age and over
- Sampling Frame
 - Canadian population 12 years of age and over excluding about 3% population (e.g., aboriginal settlements, canadian Forces, institutionalized, foster care, 2 selected Quebec health regions)
- Active Frame
 - People that are still reachable e.g., not dead or have not moved
- Sampling list
 - Prepared from a specific sampling technique

- Actual sample
 - people that have responded

Note that, results from ‘actual sample’ can be generalized to the ‘active frame’. Hence, an inference from a sample is not really generalizable to the target population (strictly speaking).

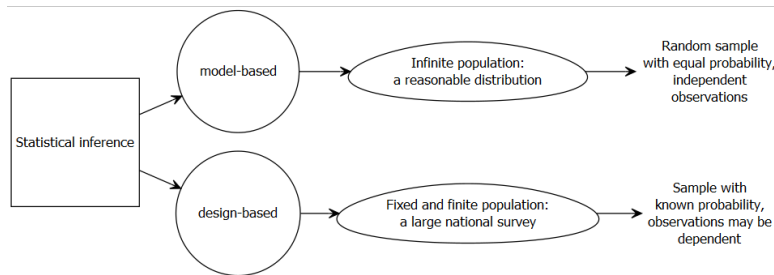
3.1.2 Types of sampling techniques



3.2 Statistical inference

3.2.1 Model-based

Most of the statistical techniques we have seen in our pre-requisite courses (SPPH 400, 500) generally assumed that we are dealing with a sample that was obtained from an infinite population. We usually assume that a random process can approximate such data generation process, and the data was collected by a simple random sampling or SRS (everyone has equal opportunity to be selected in the sample). All our conclusions are based on such assumptions. If we are wrong in specifying correct distribution to approximate the data generating process, our subsequent inferences may not be valid anymore.



3.2.2 Design-based

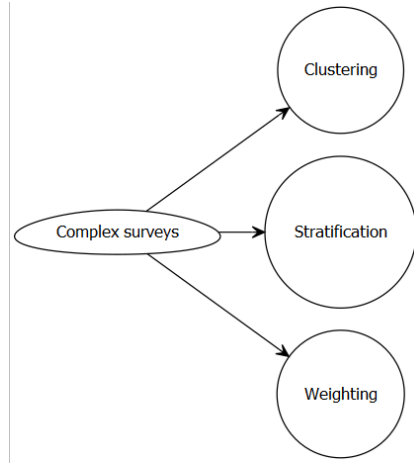
Generally, when wide-scale surveys are designed, simple random sampling or SRS may not be feasible for various practical considerations. Maybe researchers and policy-makers want that a special but small sub-group subjects should be included in our sample (e.g., people suffering from a rare disease), but it is possible that by a SRS scheme, none of the subject from that small subgroup will be included. For convenience of sampling, and for controlling variance, researchers may have to make decisions regarding how the survey needs to collect sample. Researchers may resort to cluster or stratified sampling; or a mix of both (trade-off between cost and precision). Unfortunately, in these cases, equal probability of being selected in the sample is not there any more. Lumley (2011) discussed the following properties for making design-based inference:

- properties needed to get valid estimates
 - non-zero probability ($P_i > 0$ for subject i) of being selected in the sample
 - every subject has a known probability (P_i) of being selected
- properties needed to achieve accuracy of those estimates
 - Every pair of subjects must have a non-zero probability ($P_{ij} > 0$ for subjects i and j) of being selected in the sample and

- that probability (P_{ij}) must be known as well.

3.3 Complex surveys

3.3.1 Design features



3.3.1.1 Stratification

Considering sub-groups that are sufficiently different from each other with respect to characteristics. Usual examples:

- different geographical location
- income
- gender

For each stratum (single unit), sampling is done separately. As we can select sample size from each stratum, we are able to control for variability of the estimates (SE) from each strata as well.

3.3.1.2 Clustering

Clustering is done for convenience of data collection, generally. In a nationwide survey, researchers may choose to collect more samples from selected geographic locations. This is generally the case for cost considerations. In doing so, the surveyers don't have to travel too far, as they could essentially get many neighboring subjects at a much lower cost. An obvious consequence could be that

the neighboring subjects may be more correlated with each other compared to subjects who are selected by randomness. This may cause the observations not being independent anymore.

3.3.1.3 Weighting

Assume that, in a SRS, a subject is selected in a sample with a probability of $p_i = 0.04$. This mean, that person is representing $(1/p_i) = (1/0.04) = 25$ subjects in the population. We call this the **sampling weight** ($w_i = 25$). There are other type of weight:

- **precision weight**
- **frequency weight**

but we are not really interested about those in this course in general.

In a complex survey, where we have stratification and clustering, this weight is not as straight-forward because, then, it is coming from an unequal probability sampling. As a consequence, not all subjects in the population will have the same probability (p_i) of being included in the sample, and the sampling weights (w_i) will vary as well (but the probability or weight is known for each subjects).

3.3.2 Design effect

Compared to a SRS, all of the design features of a complex survey, such as, stratification, cluster sampling, and weighting generally influence the SEs of the estimates. Survey researchers use a ratio called design effect, to account for the difference in SEs between a complex survey versus a SRS:

$$DE^2 = \frac{SE_{Complex.Survey}^2}{SE_{SRS}^2}.$$

3.4 Further readings

Available via UBC library:

- Chapter 2 of Heeringa et al. (2017)
- Chapter 1 of Lumley (2011)
- Section 6.3 of Bilder and Loughin (2014)
- Chapter 12 of Vittinghoff et al. (2011)

3.5 Exercise

- Skim through the first chapter (from the further readings list). Should be easier to read most of it after this lecture.
- If any terminology remains unfamiliar, please discuss on Canvas.

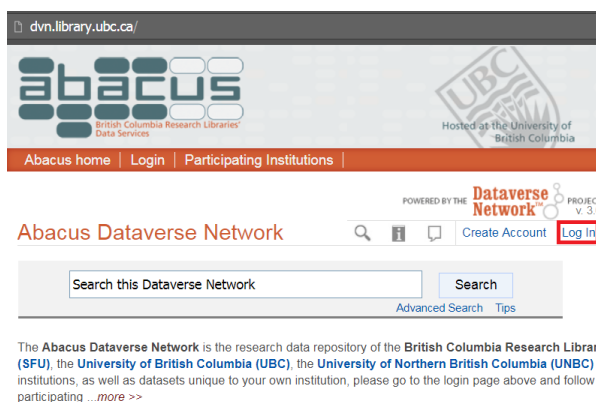
Chapter 4

Importing CCHS to R

This is a short instruction document of how to get CCHS dataset from the UBC library site to your RStudio environment. Once we bring the dataset into RStudio, the next step is to think about creating analytic dataset.

4.1 Downloading CCHS data from UBC

- **Step 1:** Go to dvn.library.ubc.ca/, and press ‘log-in’



- **Step 2:** Select ‘UBC’ from the dropdown menu

- **Step 3:** Enter your CWL or UBC library authentication information

EZproxy Login

- **Step 4:** Once you log-in, search the term 'cchs' in the search-box

- **Step 5:** For illustrative purposes, let us work with the Cycle 3.1 of the CCHS dataset from the list of results

29

- **Step 6:** CCHS Cycle 3.1 information

CANADIAN COMMUNITY HEALTH SURVEY, CYCLE 3.1, 2005 [2006]

Version: 5 – Released: Thu Oct 26 08:54:15 PDT 2017

- **Step 7:** Choose the 'Data and Analysis' tab

Abacus Dataverse Network

CANADIAN COMMUNITY HEALTH SURVEY, CYCLE 3.1, 2005 [2006]

hdl:11272/M29BS

Version: 5 – Released: Thu Oct 26 08:54:15 PDT 2017

Cataloging Information
DATA & ANALYSIS
Comments (0)
Versions

Use the check boxes next to the file name to download multiple files. Data files will be downloaded next to the category name. You will be prompted to save a single archive file. Study files that have re

Due to the large number of files associated with this study, only 25 files are loaded at a time.

☐ Select all files
Download Selected Files

☐ **Command Files: SAS**

<input type="checkbox"/> HS_fmt sas SAS Syntax - 32 KB - 9 downloads MD5 Checksum: 448714c68726f5732ca2d494012f93f9	Download
<input type="checkbox"/> HS_i sas SAS Syntax - 77 KB - 11 downloads MD5 Checksum: 1eb9879737027d90d760cc9495f9bfc5	Download
<input type="checkbox"/> HS_lbe sas SAS Syntax - 108 KB - 9 downloads MD5 Checksum: 00ca0b94219d5abde2b3cfd8cf53dab2	Download
<input type="checkbox"/> HS_pfe sas SAS Syntax - 157 KB - 12 downloads MD5 Checksum: 85b4bbe8c8b8fe933bcf6065f7fa0fa9	Download

- **Step 8:** Download the entire data (about 159 MB) as a zip file

Abacus Dataverse Network

CANADIAN COMMUNITY HEALTH SURVEY, CYCLE 3.1, 2005 [2006]

hdl:11272/M29BS

Version: 5 – Released: Thu Oct 26 08:54:15 PDT 2017

Cataloging Information
DATA & ANALYSIS
Comments (0)
Versions

Use the check boxes next to the file name to download multiple files. Data files will be downloaded next to the category name. You will be prompted to save a single archive file. Study files that have res

Due to the large number of files associated with this study, only 25 files are loaded at a time.

☐ Select all files
Download Selected Files

☐ **Command Files: SAS**

<input type="checkbox"/> readfile sas SAS Syntax - 3 KB - 9 downloads MD5 Checksum: 36ea851aa1cdd346f6e0eef1ed97328	Download
--	----------

☐ **Command Files: SPSS**

<input type="checkbox"/> HS_i sps SPSS Syntax - 31 KB - 9 downloads MD5 Checksum: 55597a88beaf954f18dccc2545302ba1f	Download
--	----------

☐ **Data: CD**

<input checked="" type="checkbox"/> cchs_cycle3-1CD zip Zip Archive - 159 MB - 25 downloads MD5 Checksum: d444be61719b34d3daaf3436ef8cd7ce	Download
---	----------

☐ **Documentation**

<input type="checkbox"/> CCHS_2000-2014-Errata.pdf Adobe PDF - 1 MB - 1 download MD5 Checksum: 272bebee53704d53b2ac5d24268ba579	Download
--	----------

- **Step 9:** Accept the ‘terms of use’

SPECIAL PERMISSIONS

Resource LICENSED for Abacus institutions.

CONDITIONS

Licence agreement for Data from Statistics Canada, including Public Use Microdata Files,
Postal Code products and the CIHI Discharge Abstracts Database.

DEFINITION

1. "Microdata file" means a non-identifiable data set containing characteristics pertaining to surveyed units as described in section 2.

DESCRIPTION OF PRODUCT

2. The Microdata file referred to in this Agreement relates to Public Use Microdata Files (PUMF) in the DLI collection.

By clicking the "I agree" checkbox below, I confirm that I have read and understood each and every term set forth in the terms and conditions for the use of data and other materials found above, and I agree to be bound by all of such terms and conditions.

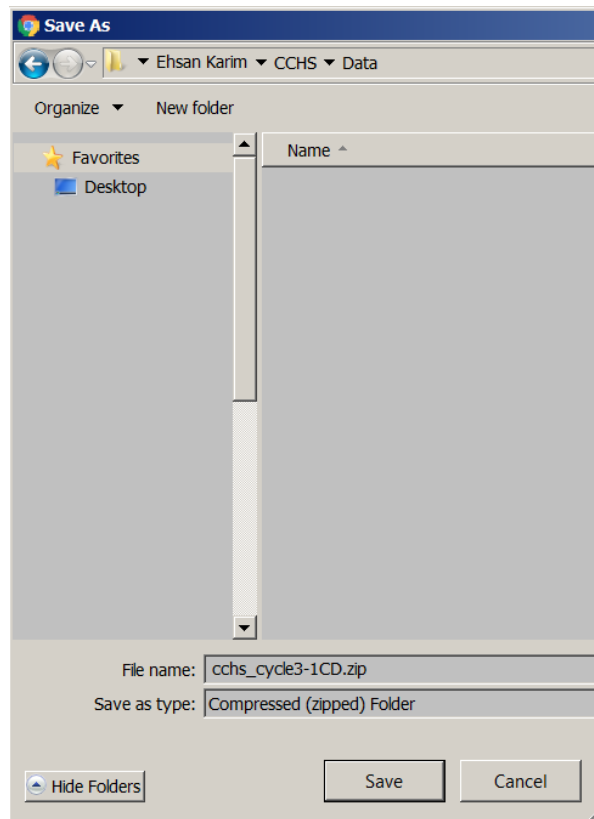
If I do not understand or agree to all of the terms and conditions, I must not use or download any data or other materials.

☒ I agree and accept these terms of use.

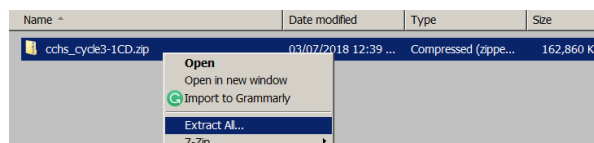
Continue

Cancel

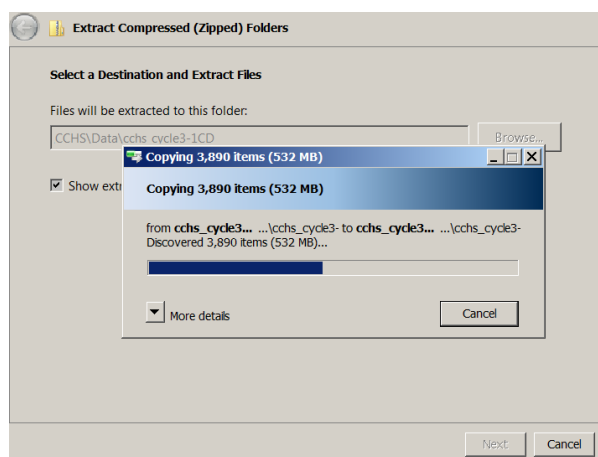
- **Step 10:** Select a directory to download the zip file



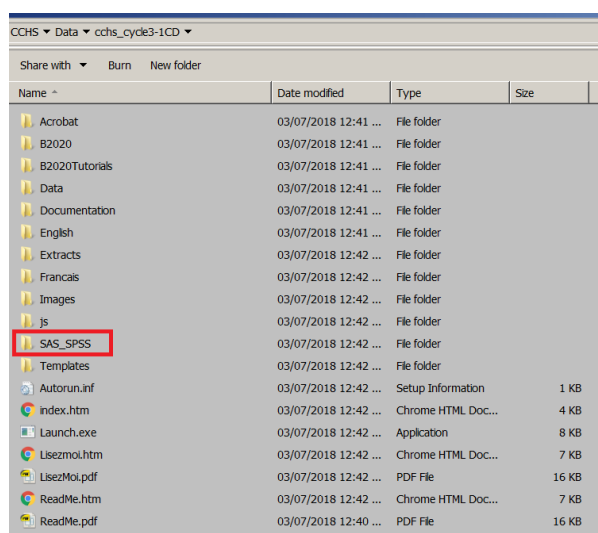
- **Step 11:** Extract the zip file



- **Step 12:** Be patient with the extraction



- **Step 13:** Once extraction is complete, take a look at the folders inside. You will see that there is a folder named 'SAS_SPSS'



4.2 Reading and Formatting the data

4.2.1 Option 1: Processing data using SAS

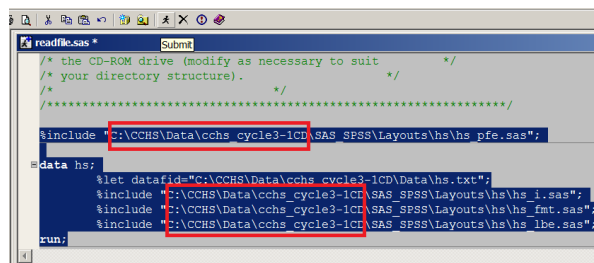
SAS is a commercial software. You may be able to get access to educational version. In case you don't have access to it, later we outline how to use free packages to read these datasets.

- **Step 1:** Inside that ‘SAS_SPSS’ folder, find the file *hs_pfe.sas*. It is a long file, but we are going to work on part of it. First thing we want to do it to change all the directory names to where you have unzipped the downloaded file (for example, here the zip file was extracted to C:/CCHS/Data/cchs_cycle3-1CD/). We only need the first part of the code (as shown below; only related to data ‘hs’). Delete the rest of the codes for now. The resulting code should like like this:

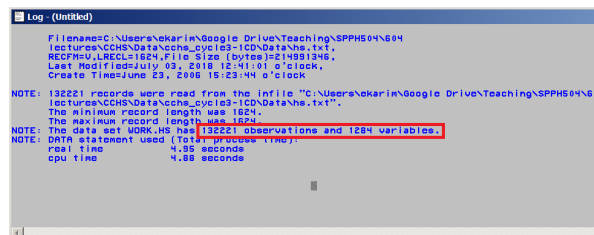
```
%include "C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hs_pfe.sas";

data hs;
    %let datafid="C:\CCHS\Data\cchs_cycle3-1CD\Data\hs.txt";
    %include "C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hs_i.sas";
    %include "C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hs_fmt.sas";
    %include "C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hs_lbe.sas";
run;
```

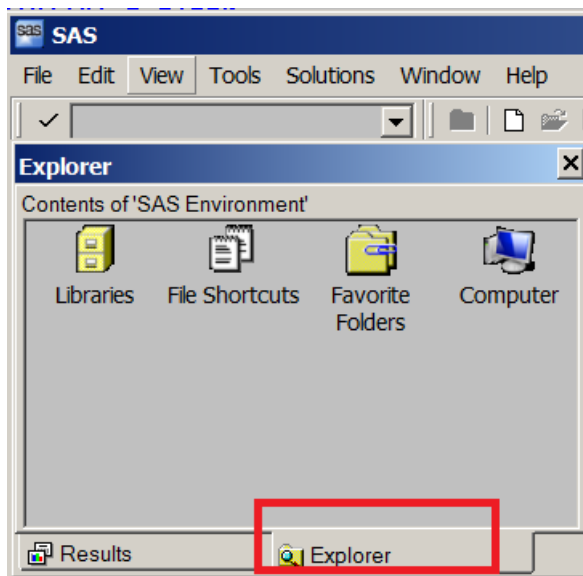
Once the modifications are done, submit the codes in SAS. Note that, the name of the data is ‘hs’.



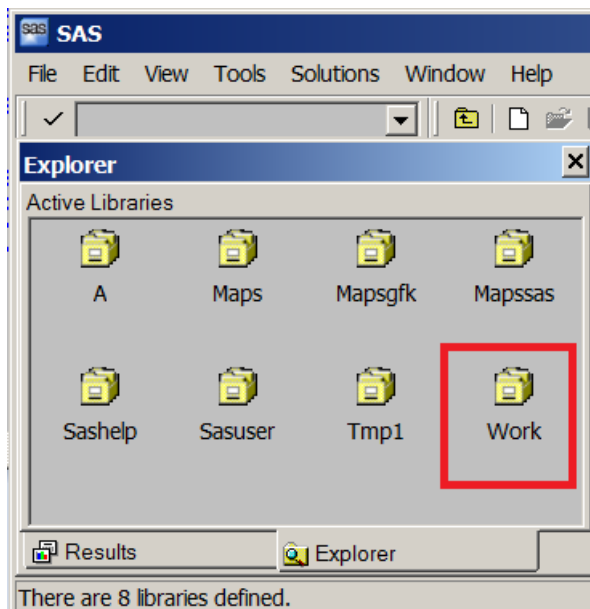
- **Step 2:** Once you submit the code, you can check the log window in SAS to see how the code submission went. It should tell you how many observations and variables were read.



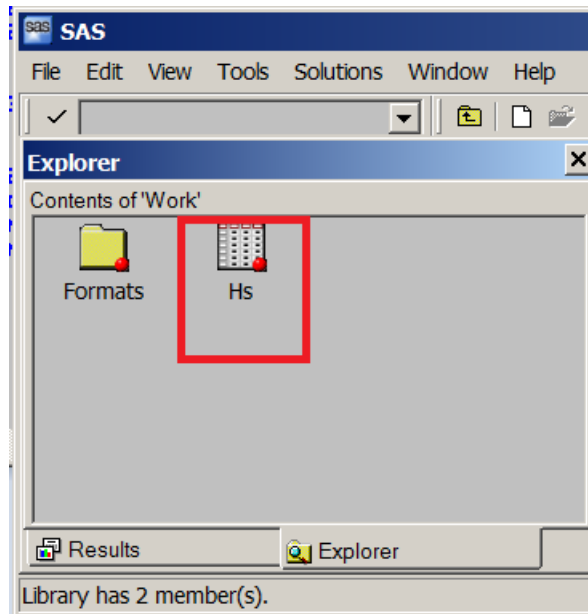
- **Step 3:** If you one to view the dataset, you can go to ‘Explorer’ window within SAS.



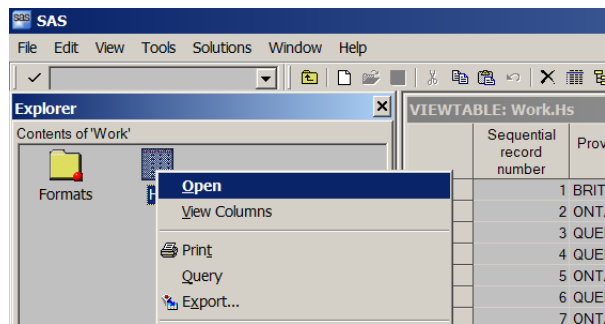
- **Step 4:** Generally, if you haven't specified where to load the files, SAS will by default save the data into a library called 'Work'



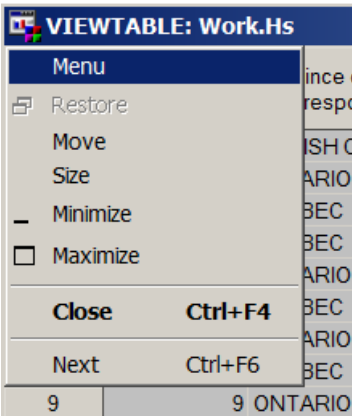
- **Step 5:** Open that folder, and you will be able to find the dataset 'Hs'.



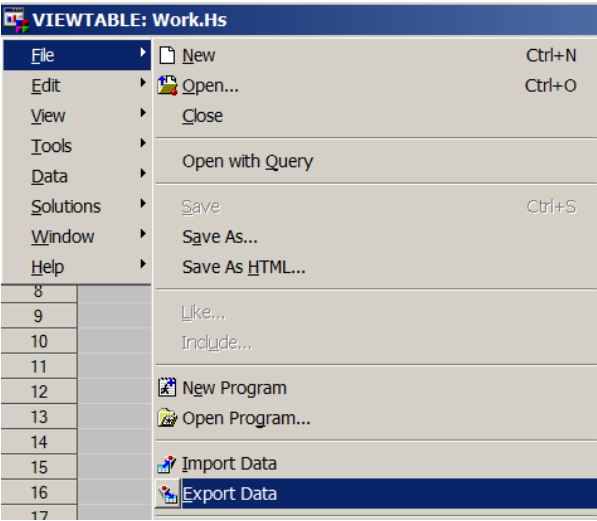
- **Step 6:** Right click on the data, and click 'open' to view the datafile.



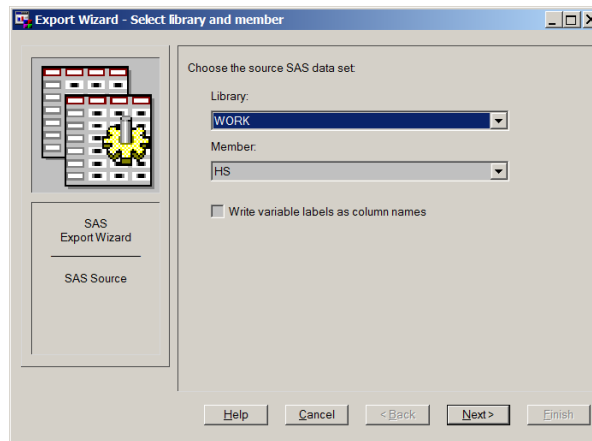
- **Step 7:** To export the data into a CSV format data (so that we can read this data into other software packages), click 'Menu'.



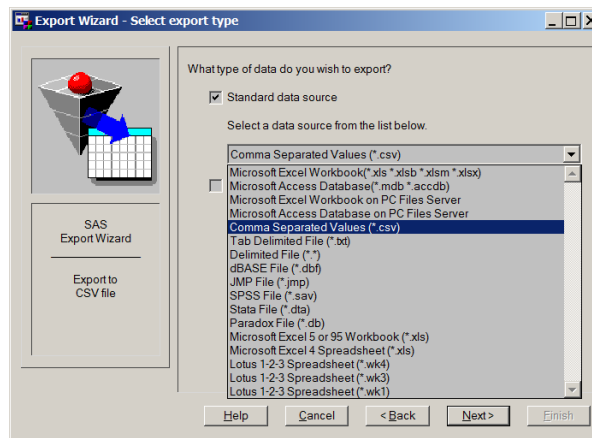
- **Step 8:** then press ‘Export Data’.



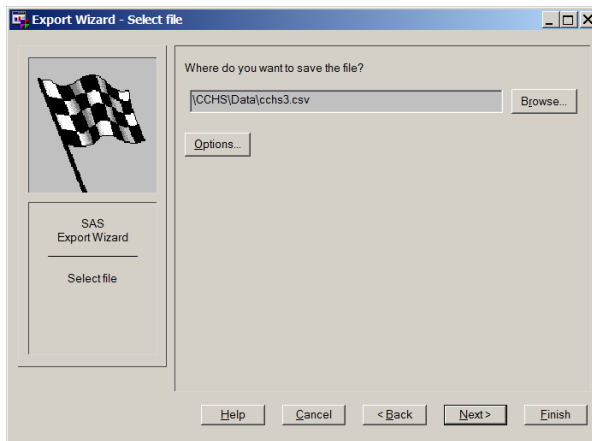
- **Step 9:** choose the library and the data.



- **Step 10:** choose the format in which you may want to save the existing data.



- **Step 11:** also specify where you want to save the csv file and the name of that file (e.g., cchs3.csv).



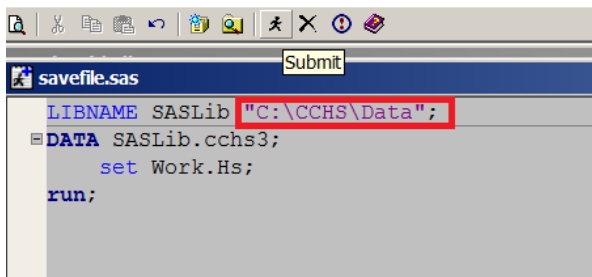
- **Step 12:** go to that directory to see the file cchs3.csv

Name	Date modified	Type	Size
cchs_cycle3-1CD	03/07/2018 12:42 ...	File folder	
cchs3.csv	03/07/2018 1:06 PM	Microsoft Excel Co...	2,103,015 ...
readfile.sas	03/07/2018 1:18 PM	SAS System Program	2 KB
readfile.sps	03/07/2018 1:33 PM	SPS File	2 KB

- **Step 13:** If you want to save the file in SAS format, you can do so by writing the following sas code into the 'Editor' window. Here we are saving the data Hs within the Work library in to a data called cchs3 within the SASLib library. Note that, the directory name has to be where you want to save the output file.

```
LIBNAME SASLib "C:\CCHS\Data";
DATA SASLib.cchs3;
    set Work.Hs;
run;
```

Submit these codes into SAS:



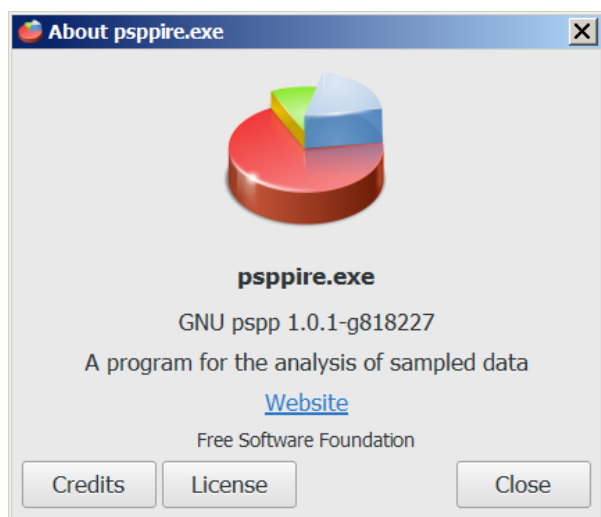
- **Step 13:** go to that directory to see the file cchs3.sas7dbat

Name	Date modified	Type	Size
cchs_cycle3-1CD	03/07/2018 12:42 ...	File folder	
cchs3.csv	03/07/2018 1:06 PM	Microsoft Excel Co...	2,103,015 ...
cchs3.sas7bdat	03/07/2018 2:11 PM	SAS Data Set	1,333,332 ...
readfile.sas	03/07/2018 1:18 PM	SAS System Program	2 KB
readfile.sps	03/07/2018 1:33 PM	SPS File	2 KB

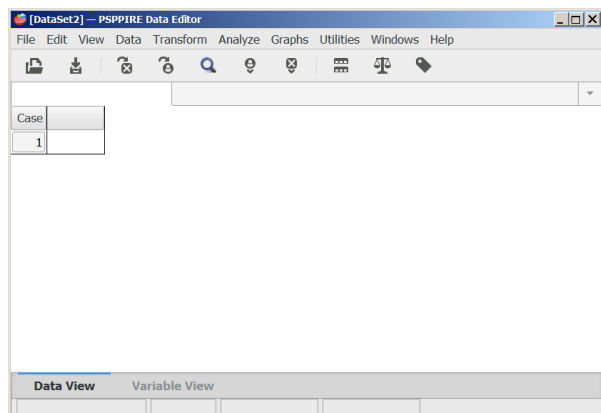
4.2.2 Option 2: Processing data using PSPP (Free)

PSPP is a free package; alternative to commercial software SPSS. We can use the same SPSS codes to read the datafile into PSPP, and save.

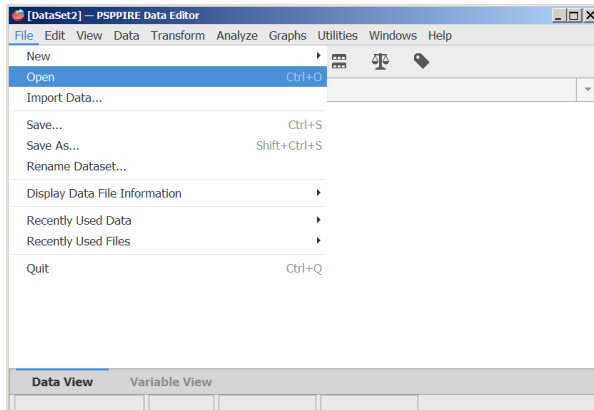
- **Step 1:** Get the free PSPP software from the website: www.gnu.org/software/pspp/



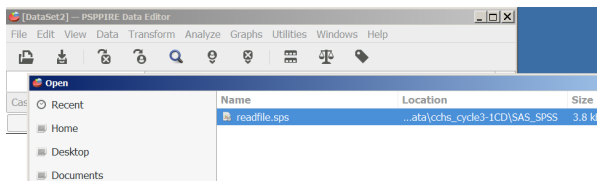
- **Step 2:** Open PSPP



- **Step 3:** Go to ‘file’ menu and click ‘open’

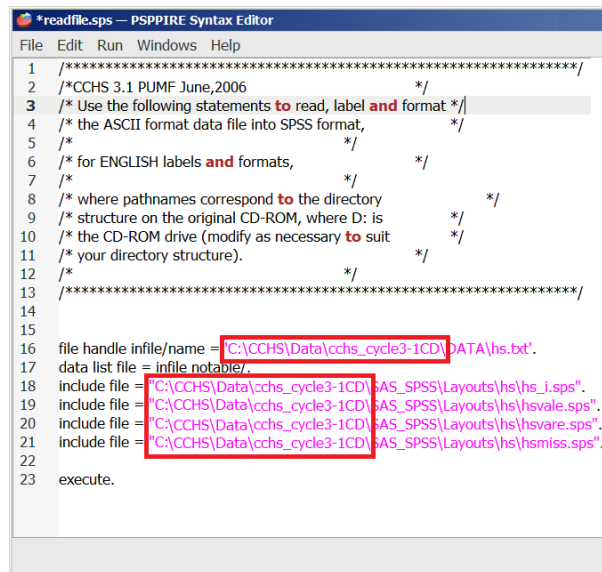


- **Step 4:** Specify the *readfile.sps* file from the ‘SAS_SPSS’ folder.



- **Step 5:** Similar to before, change the directories as appropriate. Get rid of the extra lines of codes. Resulting codes are as follows:

```
file handle infile/name = 'C:\CCHS\Data\cchs_cycle3-1CD\DATA\hs.txt'.
data list file = infile notable/.
include file = "C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hs_i.sps".
include file = "C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hsvale.sps".
include file = "C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hsvare.sps".
include file = "C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hsmisss.sps".
execute.
```

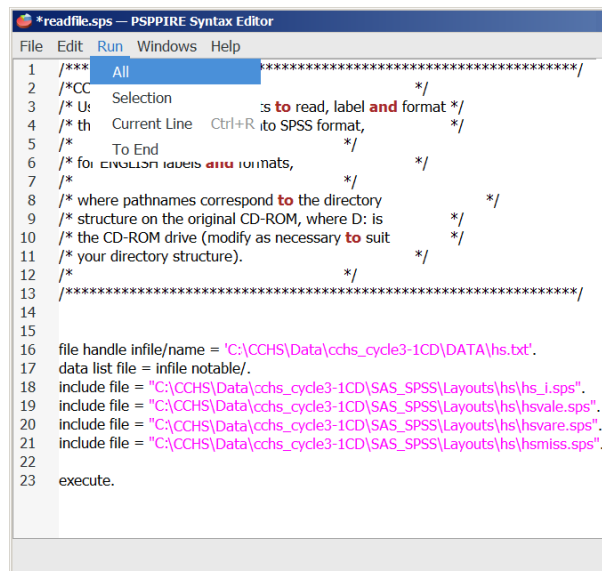


```

1  /*****
2  /*CCHS 3.1 PUMF June,2006
3  /* Use the following statements to read, label and format */
4  /* the ASCII format data file into SPSS format,
5  /*
6  /* for ENGLISH labels and formats,
7  /*
8  /* where pathnames correspond to the directory
9  /* structure on the original CD-ROM, where D: is
10 /* the CD-ROM drive (modify as necessary to suit
11 /* your directory structure).
12 /*
13 /*****/
14
15
16 file handle infile/name = 'C:\CCHS\Data\cchs_cycle3-1CD\DATA\hs.txt'.
17 data list file = infile notable/.
18 include file = 'C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hs_i.sps'.
19 include file = 'C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hsvale.sps'.
20 include file = 'C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hsware.sps'.
21 include file = 'C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hsmiss.sps'.
22
23 execute.

```

- **Step 6:** Run the codes.

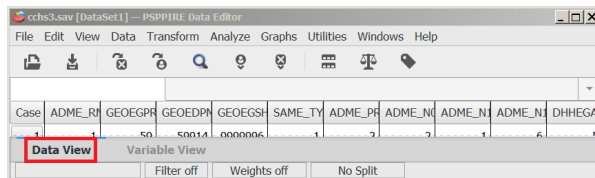


```

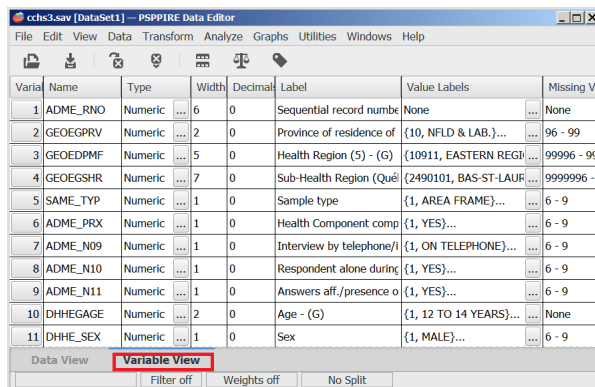
1  /*****
2  /*CC
3  /* U: Selection
4  /* th Current Line
5  /* To End
6  /* for ENGLISH labels and formats,
7  /*
8  /* where pathnames correspond to the directory
9  /* structure on the original CD-ROM, where D: is
10 /* the CD-ROM drive (modify as necessary to suit
11 /* your directory structure).
12 /*
13 /*****/
14
15
16 file handle infile/name = 'C:\CCHS\Data\cchs_cycle3-1CD\DATA\hs.txt'.
17 data list file = infile notable/.
18 include file = 'C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hs_i.sps'.
19 include file = 'C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hsvale.sps'.
20 include file = 'C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hsware.sps'.
21 include file = 'C:\CCHS\Data\cchs_cycle3-1CD\SAS_SPSS\Layouts\hs\hsmiss.sps'.
22
23 execute.

```

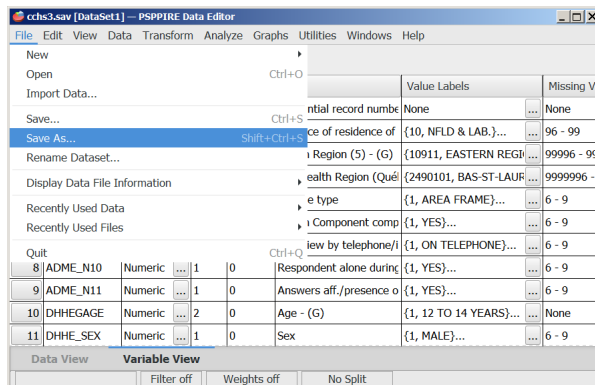
- **Step 7:** This is a large data, and will take some time to load the data into the PSPP data editor. Be patient. Once loading is complete, it will show the 'data view'.



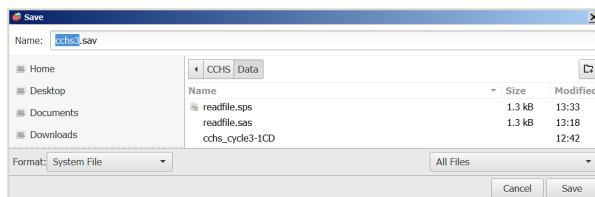
- **Step 7:** You can also check the ‘variable view’.



- **Step 8:** Save the data by clicking ‘File’ and then ‘save as ...’



- **Step 9:** Specify the name of the datafile and the location / folder to save the data file.



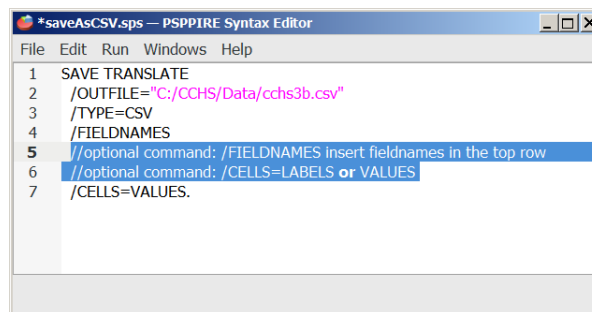
- **Step 10:** See the SAV file saved in the directory.

Name ^	Date modified	Type	Size
cchs_cycle3-1CD	03/07/2018 12:42 ...	File folder	
cchs3.csv	03/07/2018 1:06 PM	Microsoft Excel Co...	2,103,015 ...
cchs3.sav	03/07/2018 1:35 PM	SAV File	252,073 KB
readfile.sas	03/07/2018 1:18 PM	SAS System Program	2 KB
readfile.sps	03/07/2018 1:33 PM	SPS File	2 KB

- **Step 11:** To save CSV format data, use the following syntax.

```
SAVE TRANSLATE
/OUTFILE="C:/CCHS/Data/cchs3b.csv"
/TYPE=CSV
/FIELDNAMES
/CELLS=VALUES.
```

Note that, for categorical data, you can either save values or labels. For our purpose, we prefer values, and hence saved with values here.



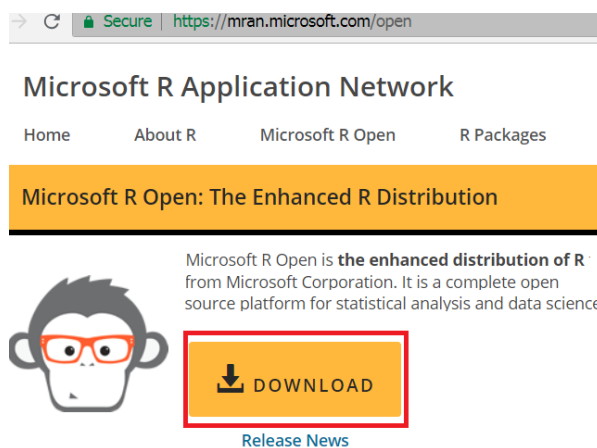
- **Step 12:** See the CSV file saved in the directory extracted from PSPP.

Name ^	Date modified
cchs_cycle3-1CD	03/07/2018 12:42 ...
cchs3.csv	03/07/2018 1:06 PM
cchs3.sas7bdat	03/07/2018 2:11 PM
cchs3.sav	03/07/2018 4:37 PM
cchs3b.csv	03/07/2018 5:05 PM
processdata.R	03/07/2018 5:05 PM
readfile.sas	03/07/2018 1:18 PM
readfile.sps	03/07/2018 1:33 PM
saveAsCSV.sps	03/07/2018 5:07 PM
savefile.sas	03/07/2018 2:13 PM

4.3 Processing data in R

4.3.1 Download software

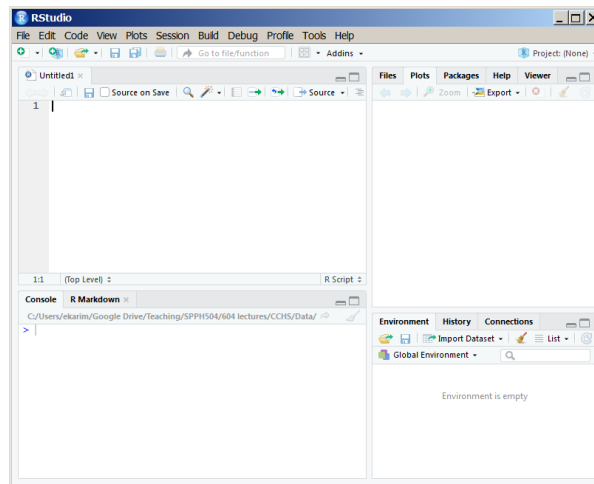
- **Step 1:** Download either 'R' from CRAN www.r-project.org or 'R open' from Microsoft mran.microsoft.com/open



- **Step 2:** Download RStudio from www.rstudio.com/



- **Step 3:** Open RStudio



4.3.2 Import, export and load data into R

- **Step 1:** Set working directory

```
setwd("C:/CCHS/Data/") # or something appropriate
```

- **Step 2:** Read the dataset created from PSPP with cell values. We can also do a small check to see if the cell values are visible. For example, we choose a variable 'CCCE_05A', and tabulate it.

```
Hs <- read.csv("cchs3b.csv", header = TRUE)
table(Hs$CCCE_05A)
```

```

Console R Markdown x
C:/CCHS/Data/
> table(Hs$CCCE_05A)
  1      2      3      4      6      7      8      9
5098 14141  2096  2236 103781  4609   41   219
>

```

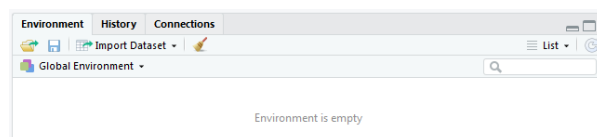
- **Step 3:** Save the RData file from R into a folder SurveyData:

```
save(Hs, file = "SurveyData/cchs3.RData")
```

- **Step 4:** See the RData file saved in the directory extracted from R.

Name	Date modified
cchs_cycle3-1CD	03/07/2018 12:42 ..
processdata.R	03/07/2018 5:25 PM
cchs3.RData	03/07/2018 5:23 PM
.Rhistory	03/07/2018 5:31 PM
readfile.sas	03/07/2018 1:18 PM
savefile.sas	03/07/2018 2:13 PM
CCHS3.1 - Shortcut	03/07/2018 5:24 PM
readfile.sps	03/07/2018 1:33 PM
saveAsCSV.sps	03/07/2018 5:07 PM

- **Step 5:** Close R / RStudio and restart it. Environment window within RStudio should be empty.



- **Step 6:** Load the saved RData into R. Environment window within RStudio should have 'Hs' dataset.

```
load("SurveyData/cchs3.RData")
```

Environment	History	Connections
<div> <div>Import Dataset</div> <div>Global Environment</div> </div>		
Data		
Hs	132221 obs. of 1284 variables	

Chapter 5

Importing NHANES to R

This is a short instruction document of how to get NHANES dataset from the US CDC site to your RStudio environment. Once we bring the dataset into RStudio, the next step is to think about creating analytic dataset.

5.1 NHANES Dataset

National Center for Health Statistics (NCHS) conducts National Health and Nutrition Examination Survey (NHANES) (CDC,NCHS (2018)). These surveys are designed to evaluate the health and nutritional status of U.S. adults and children. These surveys are being administered in two-year cycles or intervals starting from 1999-2000. Prior to 1999, a number of surveys were conducted (e.g., NHANES III), but in our discussion, we will mostly restrict our discussions to ‘continuous NHANES’ (e.g., NHANES 1999-2000 to NHANES 2017-2018).

Witin the CDC website, continuous NHANES data are available in 5 categories:

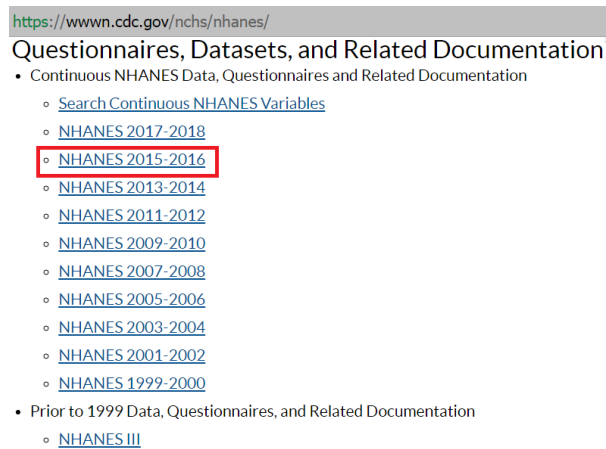
- Demographics
- Dietary
- Examination
- Laboratory
- Questionnaire

5.2 Accessing NHANES Data

In the following example, we will see how to download ‘Demographics’ data, and check associated variable in that data.

5.2.1 Accessing NHANES Data Directly from the CDC website

NHANES 1999-2000 and onward survey datasets are publicly available at www.cdc.gov/nchs/nhanes/.



- **Step 1:** Say, for example, we are interested about NHANES 2015-2016 surveys. Clicking the associated link in the above Figure gets us to the page for the cirresponding cycle (see below).

NHANES 2015-2016

Contents in Detail

- [Survey Questionnaires](#)
- [Examination and Laboratory Procedure Manuals](#)
- [Brochures and Consent Documents](#)

Using the Data

- [NHANES 2015-2016 Overview](#)
- [Technical Notes for Data Release](#)
- [Survey Methods and Analytic Guidelines](#)
- [Response Rates and Population Totals](#)
- [NHANES Web Tutorial](#)

Data, Documentation, Codebooks, SAS Code

- [Demographics](#)
- [Dietary](#)
- [Examination](#)
- [Laboratory](#)
- [Questionnaire](#)
- [Limited Access](#)

- **Step 2:** There are various types of data available for this survey. Let's explore the demographic information from this cycle. These data are mostly available in the form of SAS 'XPT' format (see below).

NHANES 2015-2016 Demographics Data

Data File Name	Doc File	Data File	Date Published
Demographic Variables and Sample Weights	DEMO_1 Doc	DEMO_1 Data (XPT - 3.6 MB)	September, 2017

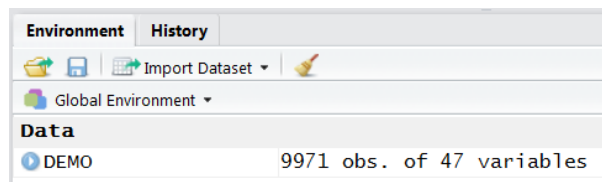
- **Step 3:** We can download the XPT data in the local PC folder and read the data into R as follows:

```
# install.packages("SASxport")
require(SASxport)
library(foreign)
DEMO <- read.xport("SurveyData\\DEMO_1.XPT")
```

```
##
## Attaching package: 'foreign'
```

```
## The following objects are masked from 'package:SASxport':
##
##      lookup.xport, read.xport
```

- **Step 4:** Once data is imported in RStudio, we will see the DEMO object listed under data window (see below):



- **Step 5:** We can also check the variable names in this DEMO dataset as follows:

```
names(DEMO)
```

```
## [1] "SEQN"      "SDDSRVYR" "RIDSTATR" "RIAGENDR" "RIDAGEYR" "RIDAGEMN"
## [7] "RIDRETH1" "RIDRETH3" "RIDEXMON" "RIDEXAGM" "DMQMILIZ" "DMQADFC"
## [13] "DMDBORN4" "DMDCITZN" "DMDYRSUS" "DMDEDUC3" "DMDEDUC2" "DMDMARTL"
## [19] "RIDEXPRG" "SIALANG" "SIAPROXY" "SIAINTRP" "FIALANG" "FIAPROXY"
## [25] "FIAINTRP" "MIALANG" "MIAPROXY" "MIAINTRP" "AIALANG" "DMDHHSIZ"
## [31] "DMDFMSIZ" "DMDHHSZA" "DMDHHSZB" "DMDHHSZE" "DMDHRGND" "DMDHRAGE"
## [37] "DMDHRBR4" "DMDHREDU" "DMDHRMAR" "DMDHSEDU" "WTINT2YR" "WTMEC2YR"
## [43] "SDMVPSU"  "SDMVSTRA" "INDHHIN2" "INDFMIN2" "INDFMPIR"
```

- **Step 6:** We can open the data in RStudio in the dataview window (by clicking the DEMO data from the data window). The next Figure shows only a few columns and rows from this large dataset. Note that there are some values marked as “NA”, which represents missing values.

DMDHSEDU HHS ref person's spouse's education level	WTINT2YR Full sample 2 year interview weight	WTMEC2YR Full sample 2 year MEC exam weight	SDMVPSU Masked variance pseudo-PSU	SDMVSTRA Masked variance pseudo-stratum
NA	9964.725	9860.625	1	120
5	44749.890	46173.307	2	124
NA	9891.944	10963.314	1	119
5	37043.087	39353.307	2	128
4	22744.355	23557.163	1	125
4	18526.180	18249.326	2	122
NA	20395.535	20068.663	2	126
NA	24788.723	25399.385	2	126
4	10998.012	11273.998	2	129
NA	34513.078	35673.964	1	121
NA	10988.317	11184.295	2	131
4	60125.441	63059.813	2	131
5	96194.928	97001.988	1	125
2	14862.011	14802.214	1	128

- **Step 7:** There is a column name associated with each column, e.g., DMDHSEDU in the first column in the above Figure. To understand what the column names mean in this Figure, we need to take a look at the codebook. To access codebook, click the 'DEMO|Doc' link (in step 2). This will show the data documentation and associated codebook (see the next Figure).

TABLE OF CONTENTS	
•	Component Description
•	Eligible Sample
•	Interview Setting and Mode of Administration
•	Quality Assurance & Quality Control
•	Data Processing and Editing
•	Analytic Notes
•	References
•	Codebook
•	SEQN - Respondent sequence number
•	SDDSRVYR - Data release cycle
•	RIDSTATR - Interview/Examination status
•	RIAGENDR - Gender
•	RIDAGEYR - Age in years at screening
•	DMDHRMAR - HH ref person's marital status
•	DMDHSEDU - HH ref person's spouse's education level
•	WTINT2YR - Full sample 2 year interview weight
•	WTMEC2YR - Full sample 2 year MEC exam weight

- **Step 8:** We can see a link for the column or variable DMDHSEDU in the table of content (in the above Figure). Clicking that link will provide us further information about what this variable means (see the next Figure).

DMDHSEDU - HH ref person's spouse's education level				
Variable Name:	DMDHSEDU			
SAS Label:	HH ref person's spouse's education level			
English Text:	HH reference person's spouse's education level			
Target:	Both males and females 0 YEARS - 150 YEARS			
Code or Value	Value Description	Count	Cumulative	Skip to Item
1	Less Than 9th Grade	619	619	
2	9-11th Grade (Includes 12th grade with no diploma)	511	1130	
3	High School Grad/GED or Equivalent	980	2110	
4	Some College or AA degree	1462	3572	
5	College Graduate or above	1629	5201	
7	Refused	2	5203	
9	Don't Know	23	5226	
.	Missing	4745	9971	

- **Step 9:** We can assess if the numbers reported under count and cumulative (from the above Figure) matches with what we get from the DEMO data we just imported (particularly, for the DMDHSEDU variable):

```
table(DEMO$DMDHSEDU)
```

```
##
##      1      2      3      4      5      7      9
## 619  511  980 1462 1629      2     23
```

```
cumsum(table(DEMO$DMDHSEDU))
```

```
##      1      2      3      4      5      7      9
## 619 1130 2110 3572 5201 5203 5226
```

```
length(is.na(DEMO$DMDHSEDU))
```

```
## [1] 9971
```

5.2.2 Accessing NHANES Data Using R Packages

5.2.2.1 nhanesA

`nhanesA` provides a convenient way to download and analyze NHANES survey data.

```
#install.packages("nhanesA")
library(nhanesA)
```

- **Step 1:** Within the CDC website, NHANES data are available in 5 categories
 - Demographics (DEMO)
 - Dietary (DIET)
 - Examination (EXAM)
 - Laboratory (LAB)
 - Questionnaire (Q)

To get a list of available variables within a datafile, we run the following command (e.g., we check variable names within DEMO data):

```
library(nhanesA)
```

```
## Warning: package 'nhanesA' was built under R version 4.0.2
```

```
nhanesTables(data_group='DEMO', year=2015)
```

```
##   FileName                                Description
## 1   DEMO_I Demographic Variables and Sample Weights
```

- **Step 2:** We can obtain the summaries of the downloaded data as follows (see below):

```
demo <- nhanes('DEMO_I')
```

```
## Processing SAS dataset DEMO_I    ..
```

```
names(demo)
```

```
## [1] "SEQN"      "SDDSRVYR" "RIDSTATR" "RIAGENDR" "RIDAGEYR" "RIDAGEMN"
## [7] "RIDRETH1" "RIDRETH3" "RIDEXMON" "RIDEXAGM" "DMQMILIZ" "DMQADFC"
## [13] "DMDBORN4" "DMDCITZN" "DMDYRSUS" "DMDEDUC3" "DMDEDUC2" "DMDMARTL"
## [19] "RIDEXPRG" "SIALANG" "SIAPROXY" "SIAINTRP" "FIALANG" "FIAPROXY"
## [25] "FIAINTRP" "MIALANG" "MIAPROXY" "MIAINTRP" "AIALANGA" "DMDHHSIZ"
## [31] "DMDFMSIZ" "DMDHHSZA" "DMDHHSZB" "DMDHHSZE" "DMDHRGND" "DMDHRAGE"
## [37] "DMDHRBR4" "DMDHREDU" "DMDHRMAR" "DMDHSEDU" "WTINT2YR" "WTMEC2YR"
## [43] "SDMVPSU"  "SDMVSTRA" "INDHHIN2" "INDFMIN2" "INDFMPPIR"
```

```
table(demo$DMDHSEDU)
```

```
##
##      1      2      3      4      5      7      9
## 619  511  980 1462 1629      2     23
```

```
cumsum(table(demo$DMDHSEDU))
```

```
##      1      2      3      4      5      7      9  
## 619 1130 2110 3572 5201 5203 5226
```

```
length(is.na(demo$DMDHSEDU))
```

```
## [1] 9971
```

5.2.2.2 RNHANES

RNHANES (Susmann (2016)) is another packages for downloading the NHANES data easily. Try yourself.

Bibliography

- Bilder, C. R. and Loughin, T. M. (2014). *Analysis of categorical data with R*. CRC Press.
- CDC,NCHS (2018). National health and nutrition examination survey data. <https://wwwn.cdc.gov/nchs/nhanes/>. [Online; accessed 11-April-2018].
- Dobson, A. and Barnett, A. (2008). *An introduction to generalized linear models, third edition*. Chapman and Hall/CRC.
- Heeringa, S. G., West, B. T., and Berglund, P. A. (2017). *Applied survey data analysis*. Chapman and Hall/CRC.
- Lumley, T. (2011). *Complex surveys: a guide to analysis using R*, volume 565. John Wiley & Sons.
- Susmann, H. (2016). *RNHANES: Facilitates Analysis of CDC NHANES Data*. R package version 1.1.0.
- Vittinghoff, E., Glidden, D. V., Shiboski, S. C., and McCulloch, C. E. (2011). *Regression methods in biostatistics: linear, logistic, survival, and repeated measures models*. Springer Science & Business Media.