```
/* ACCESSING, EXPLORING AND PREPARING OUR DATA */

/* creating a library to store our SAS tables */
libname group14 base "/home/u49191131/STAT395 Project";

/* importing the excel file dataset into SAS */
proc import datafile="/home/u49191131/STAT395 Project/Health Insurance Costs.xlsx"
            dbms=xlsx out=group14.health_insurance_costs_original;
run;

proc contents data=group14.health_insurance_costs_original;
run;    /* to see the contents/descriptor portion of the data,
            noting that the variables bmi & charges are stored as character variables */

/* converting the character variables (bmi & charges) to numeric variables using the input funtion,
    and thereafter cleaning the dataset */
data group14.health_insurance_costs_numeric;
    set group14.health_insurance_costs_original (rename=(bmi=bmi_character charges=charges_character));
        bmi=input(bmi_character, 8.);
        charges=input(charges_character, 13.);
    drop sex bmi_character charges_character;
    format charges dollar16.2;
run;

/* creating new character columns/variables for bmi & age with their different categories/levels,
    also categorizing the variable charges as being either greater than or less than $10,000 */
data group14.clean;
    set group14.health_insurance_costs_numeric;

    length bmi_category $ 14;
    if bmi<18.5 then bmi_category="underweight";
    if bmi>=18.5 and bmi<25 then bmi_category="normal weight";
    if bmi>=25.0 and bmi<30 then bmi_category="overweight";
    if bmi>=30.0 then bmi_category="obese";

    length age_category $ 12;
    if age>=18 and age<=34 then age_category="younger";
    if age>=35 and age<=50 then age_category="middle-aged";
    if age>=51 and age<=64 then age_category="older";

    if charges>10000 then charges_category="high";
    else charges_category="low";

run;

/* identifying and removing any duplicate observations */
proc sort data=group14.clean out=group14.clean2 noduprecs dupout=work.dups;
    by _all_;
run;    /* found one duplicate observation which was removed
            and the total population size is now 1337 */

/* in order to determine what sample size we need to select from the population,
    we first draw a pilot sample of size 50, then work out the variance of it,
    and we want the error in estimation to be less than $1,500 */
proc sort data=group14.clean2;
    by charges;
run;

proc surveyselect data=group14.clean2
                  out=work.initial_samplesize
                  seed=14
                  method=srs
                  sampsize=50
                  stats;
run;

proc univariate data=work.initial_samplesize;
    var charges;
run;    /* variance for the pilot sample = 158415515 */

/* we used SRS to sample and calculated n=225,
    so we need to sample at least 225 individuals from the 1337 individuals, using SRS,
    to get an error of no more than $1,500 */
```

```
/* RUNNING THE VARIOUS SAMPLING SURVEY PROCEDURES IN ORDER TO ESTIMATE THE AVERAGE CHARGES */

/* Selecting a sample of size 225 using Simple Random Sampling without replacement ...
    sorting the data by our variable of interest first */
proc sort data=group14.clean2;
    by charges;
run;

proc surveyselect data=group14.clean2
                  sampsize=225
                  out=group14.costs_srs_sample
                  method=srs
                  seed=14
                  stats;
run;

/* Estimating the average charges billed by the health insurance for the entire population */
title1 'SRS';
proc surveymeans data=group14.costs_srs_sample
                 total=1337
                 mean clm var cv;
                 var charges;
                 weight SamplingWeight;
run;
title;

/* Selecting a sample of size 225 using Simple Random Sampling with replacement ...
    sorting the data by our variable of interest first */
proc sort data=group14.clean2;
    by charges;
run;

proc surveyselect data=group14.clean2
                  sampsize=225
                  out=group14.costs_urs_sample
                  method=urs
                  seed=14
                  stats;
run;

/* Estimating the average charges billed by the health insurance for the entire population */
title2 'URS';
proc surveymeans data=group14.costs_urs_sample
                 total=1337
                 mean clm var cv;
                 var charges;
                 weight SamplingWeight;
run;
title;

/* Selecting a sample of size 225 using Stratified Simple Random Sampling without replacement ...
    sorting the data by our stratification variable first */
proc sort data=group14.clean2;
    by smoker;
run;

/* Get the size of each stratum i.e. how many elements fall into each category of the stratification variable */
proc freq data=group14.clean2;
    table smoker / out=work.str_sizes (rename=(count=_total_));
run;

proc surveyselect data=group14.clean2
                  sampsize=(179 46)
                  out=group14.costs_strata_sample
                  method=srs
                  seed=14
                  stats;
                  strata smoker;
run;

/* Estimating the average charges billed by the health insurance for the entire population */
title3 'Stratified';
proc surveymeans data=group14.costs_strata_sample
                 total=work.str_sizes /* this is the table that stored the stratum population sizes */
```

```
                mean clm var cv;
                var charges;
                weight SamplingWeight;
                strata smoker;
                *by smoker; /* the estimates of the means for each stratum were very different */
run;
title;

/* Selecting a sample using Cluster Simple Random Sampling without replacement ...
    sorting the data by our cluster variable first */
proc sort data=group14.clean2;
    by region;
run;

proc surveyselect data=group14.clean2
                  sampsize=2 /* selected 2 clusters */
                  out=group14.costs_cluster_sample
                  method=srs
                  seed=14
                  stats;
                  cluster region;
run;

/* Estimating the average charges billed by the health insurance for the entire population */
title4 'Cluster';
proc surveymeans data=group14.costs_cluster_sample
                 total=4 /* number of categories in the cluster variable */
                 mean clm var cv;
                 var charges;
                 weight SamplingWeight;
                 cluster region;
run;
title;


/* RUNNING THE VARIOUS SAMPLING SURVEY PROCEDURES IN ORDER TO ESTIMATE THE PROPORTION OF
    INDIVIDUALS WHOSE CHARGES ARE GREATER THAN $10,000 */

/* recoding the variable charges_category as either 1's or 0's,
    in order to estimate the proportion */
data group14.clean3;
    set group14.clean2;
    if charges_category="high" then charges_recoded=1;
    else charges_recoded=0;
run;

/* Selecting a sample of size 225 using Simple Random Sampling without replacement ...
    sorting the data by our variable of interest first */
proc sort data=group14.clean3;
    by charges_recoded;
run;

proc surveyselect data=group14.clean3
                  sampsize=225
                  out=group14.prop_srs_sample
                  method=srs
                  seed=14
                  stats;
run;

/* Estimating the proportion of individuals whose medical costs exceed $10,000 */
title5 'SRS';
proc surveymeans data=group14.prop_srs_sample
                 total=1337
                 mean clm var cv;
                 var charges_recoded;
                 weight SamplingWeight;
run;
title;

/* Selecting a sample of size 225 using Simple Random Sampling with replacement ...
    sorting the data by our variable of interest first */
proc sort data=group14.clean3;
    by charges_recoded;
run;
```

```
proc surveyselect data=group14.clean3
                  sampsize=225
                  out=group14.prop_urs_sample
                  method=urs
                  seed=14
                  stats;
run;

/* Estimating the proportion of individuals whose medical costs exceed $10,000 */
title6 'URS';
proc surveymeans data=group14.prop_urs_sample
                 total=1337
                 mean clm var cv;
                 var charges_recoded;
                 weight SamplingWeight;
run;
title;

/* Selecting a sample of size 225 using Stratified Simple Random Sampling without replacement ...
    sorting the data by our stratification variable first */
proc sort data=group14.clean3;
    by smoker;
run;

proc freq data=group14.clean3;
    table smoker / out=work.strata_sizes (rename=(count=_total_));
run;

proc surveyselect data=group14.clean3
                  sampsize=(179 46)
                  out=group14.prop_strata_sample
                  method=srs
                  seed=14
                  stats;
                  strata smoker;
run;

/* Estimating the proportion of individuals whose medical costs exceed $10,000 */
title7 'Stratified';
proc surveymeans data=group14.prop_strata_sample
                 total=work.strata_sizes
                 mean clm var cv;
                 var charges_recoded;
                 weight SamplingWeight;
                 strata smoker;
                 *by smoker;
run;
title;

/* Selecting a sample using Cluster Simple Random Sampling without replacement ...
    sorting the data by our cluster variable first */
proc sort data=group14.clean3;
    by region;
run;

proc surveyselect data=group14.clean3
                  sampsize=2
                  out=group14.prop_cluster_sample
                  method=srs
                  seed=14
                  stats;
                  cluster region;
run;

/* Estimating the proportion of individuals whose medical costs exceed $10,000 */
title8 'Cluster';
proc surveymeans data=group14.prop_cluster_sample
                 total=4
                 mean clm var cv;
                 var charges_recoded;
                 weight SamplingWeight;
                 cluster region;
run;
title;
```