```
1
2   cls
3   clear all
4
5   *************
6   **#CLEANING**
7   *************
8   cap use "...", replace
9   drop hba1c_date fgt fgt_date smk_date weight w_date index_hes yob cause uts_cprd index_cprd hes_end pracid year agein_grp bmi_grp dob
10  rename smk_status smk
11
12  sum bmi
13  egen float bmig = cut(bmi), at(0 18.5 25 30 35 40 100) icodes label
14  tabstat bmi, statistics(N min max) by(bmig)
15  tab bmig, nolab
16  label define bmig 0 `"<18.5"', modify
17  label define bmig 1 `"≥18.5 to <25"', modify
18  label define bmig 2 `"≥25 to <30"', modify
19  label define bmig 3 `"≥30 to <35"', modify
20  label define bmig 4 `"≥35 to <40"', modify
21  label define bmig 5 `"≥40"', modify
22  tabstat bmi, statistics(min max) by(bmig)
23
24  *All_cancer*
25  gen all = 1 if allcancer_site != ""
26  replace all = 0 if allcancer_site == ""
27  distinct patid
28  mdesc
29  order index allcancer_site allcancer_end dodeath, after(gender)
30  gen dif = allcancer_end - dodeath
31  sum dif
32  mdesc dif
33  drop dif
34  gen t_all = (allcancer_end - index)/365.24
35  drop allcancer_site allcancer_end
36
37  *Four commonest_cancer and BMI_related*
38  foreach cancer in bmirel lung breast prostate colorectal {
39      gen t_`cancer' = (`cancer'_end - index)/365.24, after(`cancer')
40      drop `cancer'_end
41  }
42  drop liver_end liver pancreas_end pancreas gallbladder_end gallbladder endometrial_end endometrial
43  compress
44  order lung t_lung breast t_breast prostate t_prostate colorectal t_colorectal bmirel t_bmirel all t_all, last
45  cap save "...", replace
46
47  ********************
48  **#*CHARACTERISTICS**
49  ********************
50  cap use "...", replace
51
52  distplot agein, over(gender) xlabel(35(5)105, grid) ylabel(#10, format(%5.1f) angle(h))
53  graph close _all
54  sum agein, d
55  egen float ageg = cut(agein), at(0 30 40 50 60 70 80 110) icodes label
56  tabstat agein, statistics(N min max) by(ageg)
57  tab ageg, nolab
58  label define ageg 1 `"<40"', modify
59  label define ageg 2 `"≥40 to <50"', modify
60  label define ageg 3 `"≥50 to <60"', modify
61  label define ageg 4 `"≥60 to <70"', modify
62  label define ageg 5 `"≥70 to <80"', modify
63  label define ageg 6 `"≥80"', modify
64  tabstat agein, statistics(N min max) by(ageg)
65
66  label variable agein  `"Age at diabetes diagnosis (years)"'
67  label variable ageg   `"Age at diabetes diagnosis group (years)"'
```

```
label variable bmi    `"Body mass index (kg/m2)"'
label variable bmig   `"Body mass index category (kg/m2)"'

****Table1, including age at diagnosis****
preserve
drop if gender == 2
baselinetable                                /*
*/  agein(cts tab("p50 (p25, p75)"))        /*
*/  bmi(cts tab("p50 (p25, p75)"))          /*
*/  bmig(cat)                               /*
*/  , by(ageg, total missing) reportmissing countformat(%10.0fc) notable /*
*/  exportexcel("...", replace)
restore

preserve
drop if gender == 1
baselinetable                                /*
*/  agein(cts tab("p50 (p25, p75)"))        /*
*/  bmi(cts tab("p50 (p25, p75)"))          /*
*/  bmig(cat)                               /*
*/  , by(ageg, total missing) reportmissing countformat(%10.0fc) notable /*
*/  exportexcel("...", replace)
restore

****TableS1 [ICD-10 codes for BMI-related cancers]****

****TableS2, rates****
stset t_all, f(all == 1)
sum t_all, d
strate, per(1000)
tab all gender, m col
strate gender, per(1000)

foreach var of varlist lung breast prostate colorectal bmirel all {
    forvalues j = 1/2 {
        qui stset t_`var', f(`var' == 1)
        qui strate ageg if gender == `j', per(1000) output("...", replace)
    }
}

clear
tempfile ratescan
save `ratescan', emptyok replace
foreach nm in lung breast prostate colorectal bmirel all {
    forvalues j = 1/2 {
        qui use "...", clear
        qui gen cancer = "`nm'"
        qui gen sex = `j'
        qui append using `ratescan'
        save `ratescan', replace
    }
}
use `ratescan', replace
replace cancer = proper(cancer)
replace cancer = "All cancers" if cancer == "All"
renames sex _D _Y _Rate _Lower _Upper cancer \ Sex Events PYRs1000 Rate1000 LB95 UB95 Type
order Sex Type, first
tostring Sex, replace
replace Sex = "Men" if Sex == "1"
replace Sex = "Women" if Sex == "2"
gsort -Sex Type ageg
foreach var of varlist PYRs1000-UB95 {
    tostring `var', format(%5.1f) force replace
}
tostring Events, replace
drop PYRs1000
gen Rate = Rate1000 + " (" + LB95 + ", " + UB95 + ")"
```

```
drop Rate1000-UB95
reshape wide Events Rate, i(Sex Type) j(ageg)
replace Type = "zBmirel" if Type == "Bmirel"    /*last in the table*/
gsort -Sex Type
compress
export excel using "...", firstrow(variables) replace

foreach nm in lung breast prostate colorectal bmirel all {
    forvalues j = 1/2 {
        qui erase "..."
    }
}


***************
**REGRESSIONS**
***************

**#**All and 4 most-common, by sex****
cap use "...", replace
rename agein ageatt

preserve
keep if gender == 1
tempfile db1_Men
save `db1_Men', replace
restore

preserve
keep if gender == 2
tempfile db1_Women
save `db1_Women', replace
restore

foreach nm in Men Women {

    foreach type in lung breast prostate colorectal all {

    di "------------------------------------------------"
    di "SEX -- `nm' || TYPE -- `type' -- $S_TIME  $S_DATE"
    di "------------------------------------------------"

    if ("`nm'" == "Men" & "`type'" == "breast") | ("`nm'" == "Women" & "`type'" == "prostate") {
        di "Few events"
    }

    else {

        qui use `db1_`nm'', clear

        stset t_`type', f(`type' == 1) id(patid)
        qui stmt, time1(df(4)) time2(df(4) start(ageatt)) timeint(t1:t2 4:4)

        preserve
        forvalues j = 0(0.1)20.1 {                      /*time since diagnosis/duration, i.e. timeinc*/
            qui clear
            qui set obs 1201
            qui generate time1 = `j'
            qui range time2 0 120 1201                  /*attained age*/
            qui tempfile temppred`n'
            qui save `temppred`n''
            local datalist `datalist' `temppred`n''
            local n = `n' + 1
        }
        qui clear
        qui set obs 0
        qui append using `datalist'
```

```
        qui tempfile timedata
        qui save `timedata'
        restore

        qui merge 1:1 _n using `timedata', nogen

        preserve
        qui predict rate, hazard time1var(time1) time2var(time2) ci
        qui gen agedd_p = time2 - time1
        qui renames time1 time2 \ diabdur_p ageatt_p
        qui order agedd_p, before(diabdur_p)
        qui keep agedd_p-ageatt_p rate*
        qui cap drop if time1 == .
        qui gen gender = "`nm'"
        qui gen type = "`type'"
        qui tempfile rates_`nm'_`type'
        qui save `rates_`nm'_`type''
        restore

        qui drop time1 time2
        qui egen float time2 = seq() in 1/6, from(5) to(10) block(1)
        qui replace time2 = time2*10
        qui gen time1_5 = 5 if time2 !=.
        qui gen time1_20 = 20 if time2 !=.
        qui predictnl lnhr  = ln(predict(hazard time1var(time1_20) time2var(time2))) - ln(predict(hazard time1var(time1_5) time2var(time2))), ci(lnhr_lci lnhr_uci)
        qui keep time2 time1_5 time1_20 lnhr*
        qui cap drop if time2 == .
        qui gen gender = "`nm'"
        qui gen type = "`type'"
        qui tempfile ratesratio_`nm'_`type'
        qui save `ratesratio_`nm'_`type''
        }
    }
}

clear
foreach nm in Men Women {
    foreach type in lung breast prostate colorectal all {
    cap append using `rates_`nm'_`type''
    }
}
foreach var of varlist rate* {
    replace `var' = `var'*1000
}
foreach var of varlist agedd_p diabdur_p ageatt_p {
    tostring `var', replace force format(%5.1f)
}
foreach var of varlist agedd_p diabdur_p ageatt_p {
    destring `var', replace
}
cap drop if rate == .
duplicates drop
cap save "...", replace

clear
foreach nm in Men Women {
    foreach type in lung breast prostate colorectal all {
    cap append using `ratesratio_`nm'_`type''
    }
}
cap save "...", replace


**#**BMI-related, by sex and BMI****
cap use "...", replace
rename agein ageatt
drop if bmi == .
```

```
drop if bmig == 0       /*removed very few subjects with BMI<18.5*/
tab bmig, gen(dbmi)
tabstat bmi if dbmi1 == 1, statistics(min max)

preserve
keep if gender == 1
tempfile db1_Men
save `db1_Men', replace
restore

preserve
keep if gender == 2
tempfile db1_Women
save `db1_Women', replace
restore

foreach nm in Men Women {

        di "SEX -- `nm' -- $S_TIME  $S_DATE"

        qui use `db1_`nm'', clear

        stset t_bmirel, f(bmirel == 1) id(patid)
        stmt dbmi2 dbmi3 dbmi4 dbmi5,                                       ///
        time1(df(4) tvc(dbmi2 dbmi3 dbmi4 dbmi5) dftvc(4))                  ///
        time2(df(4) tvc(dbmi2 dbmi3 dbmi4 dbmi5) dftvc(4) start(ageatt))    ///
        timeint(t1:t2 4:4)

        preserve
        forvalues j = 0(0.1)20.1 {                  /*time since diagnosis/duration, i.e. timeinc*/
            qui clear
            qui set obs 1201
            qui generate time1 = `j'
            qui range time2 0 120 1201              /*attained age*/
            qui tempfile temppred`n'
            qui save `temppred`n''
            local datalist `datalist' `temppred`n''
            local n = `n' + 1
        }
        qui clear
        qui set obs 0
        qui append using `datalist'
        qui tempfile timedata
        qui save `timedata'
        restore

        qui merge 1:1 _n using `timedata', nogen

        preserve
        qui predict rbmi1, hazard time1var(time1) time2var(time2) at(dbmi2 0 dbmi3 0 dbmi4 0 dbmi5 0) ci
        qui predict rbmi2, hazard time1var(time1) time2var(time2) at(dbmi2 1 dbmi3 0 dbmi4 0 dbmi5 0) ci
        qui predict rbmi3, hazard time1var(time1) time2var(time2) at(dbmi2 0 dbmi3 1 dbmi4 0 dbmi5 0) ci
        qui predict rbmi4, hazard time1var(time1) time2var(time2) at(dbmi2 0 dbmi3 0 dbmi4 1 dbmi5 0) ci
        qui predict rbmi5, hazard time1var(time1) time2var(time2) at(dbmi2 0 dbmi3 0 dbmi4 0 dbmi5 1) ci
        qui gen agedd_p = time2 - time1
        qui renames time1 time2 \ diabdur_p ageatt_p
        qui order agedd_p, before(diabdur_p)
        qui keep agedd_p-ageatt_p rbmi*
        qui cap drop if time1 == .
        qui gen gender = "`nm'"
        qui gen type = "bmirel"
        qui tempfile rates_`nm'_bmirel
        qui save `rates_`nm'_bmirel'
        restore

        qui drop time1 time2
        qui egen float time2 = seq() in 1/6, from(5) to(10) block(1)
```

```
        qui replace time2 = time2*10
        qui gen time1_5 = 5 if time2 !=.
        qui gen time1_20 = 20 if time2 !=.
        qui predictnl lnhr_bmi1  = ln(predict(hazard time1var(time1_20) time2var(time2) at(dbmi2 0 dbmi3 0 dbmi4 0 dbmi5 0))) /*
        */                       - ln(predict(hazard time1var(time1_5)  time2var(time2) at(dbmi2 0 dbmi3 0 dbmi4 0 dbmi5 0))), ci(lnhr_bmi1_lci lnhr_bmi1_uci)
        qui predictnl lnhr_bmi2  = ln(predict(hazard time1var(time1_20) time2var(time2) at(dbmi2 1 dbmi3 0 dbmi4 0 dbmi5 0))) /*
        */                       - ln(predict(hazard time1var(time1_5)  time2var(time2) at(dbmi2 1 dbmi3 0 dbmi4 0 dbmi5 0))), ci(lnhr_bmi2_lci lnhr_bmi2_uci)
        qui predictnl lnhr_bmi3  = ln(predict(hazard time1var(time1_20) time2var(time2) at(dbmi2 0 dbmi3 1 dbmi4 0 dbmi5 0))) /*
        */                       - ln(predict(hazard time1var(time1_5)  time2var(time2) at(dbmi2 0 dbmi3 1 dbmi4 0 dbmi5 0))), ci(lnhr_bmi3_lci lnhr_bmi3_uci)
        qui predictnl lnhr_bmi4  = ln(predict(hazard time1var(time1_20) time2var(time2) at(dbmi2 0 dbmi3 0 dbmi4 1 dbmi5 0))) /*
        */                       - ln(predict(hazard time1var(time1_5)  time2var(time2) at(dbmi2 0 dbmi3 0 dbmi4 1 dbmi5 0))), ci(lnhr_bmi4_lci lnhr_bmi4_uci)
        qui predictnl lnhr_bmi5  = ln(predict(hazard time1var(time1_20) time2var(time2) at(dbmi2 0 dbmi3 0 dbmi4 0 dbmi5 1))) /*
        */                       - ln(predict(hazard time1var(time1_5)  time2var(time2) at(dbmi2 0 dbmi3 0 dbmi4 0 dbmi5 1))), ci(lnhr_bmi5_lci lnhr_bmi5_uci)
        qui keep time2 time1_5 time1_20 lnhr*
        qui cap drop if time2 == .
        qui gen gender = "`nm'"
        qui gen type = "bmirel"
        qui tempfile ratesratio_`nm'_bmirel
        qui save `ratesratio_`nm'_bmirel'
}

clear
foreach nm in Men Women {
    append using `rates_`nm'_bmirel'
}
foreach var of varlist rbmi* {
    replace `var' = `var'*1000
}
foreach var of varlist agedd_p diabdur_p ageatt_p {
    tostring `var', replace force format(%5.1f)
}
foreach var of varlist agedd_p diabdur_p ageatt_p {
    destring `var', replace
}
cap drop if rbmi1 == .
duplicates drop
cap save "...", replace

clear
foreach nm in Men Women {
    append using `ratesratio_`nm'_bmirel'
}
cap save "...", replace
```