ONLINE SUPPLEMENTARY MATERIALS

The interplay of food insecurity, diet quality and dementia status in their association with all-cause mortality among older US adults in the Health and Retirement Study 2012-2020

SUPPLEMENTARY METHODS 1: ALGORITHMICALLY DEFINED DEMENTIA OUTCOMES

The algorithms use various combinations of sociodemographic characteristics, health and physical functioning variables, social engagement indicators, and cognitive indicators (i.e., cognition test item scores and proxy-reports of cognition) to predict dementia status using race/ethnicity-specific probability thresholds. Each algorithm was developed to minimize differences in prediction accuracy between race/ethnicity groups, with pairwise differences of 3 percentage points for sensitivity and 5 percentage points for specificity, making it appropriate for use in race/ethnicity disparity research.

This data file (hrsdementia_20211109.sas7bdat) was constructed using the 2018 RAND V1 HRS longitudinal file ("randhrs1992_2018v1") and core HRS data; code for recreating this dataset is available in the following Github repository, and the date is 2021_1109.

Note the minor discrepancies in probabilities and classifications for all years in this dataset compared to the previously available dataset (hrsdementia_20191028.sas7bdat), which are due to differences in source data. This previously published dataset, which covered 2000 to 2014, was constructed using the 2014 RAND HRS longitudinal V2 file ("randhrs1992_2014v2") and core HRS data; code for duplicating this prior version of the dataset is available in the following Github repository, dated 2019_0529: https://github.com/powerepilab/AD_algorithm_development

SUPPLEMENTARY METHODS 2: FOOD INSECURITY SCALE AND HEI-2015

2.1. Food Insecurity scale, 2013 WAVE

** storage display value
**variable name type format label variable label
**----------

**HNB1_13 byte %8.0g FOOD DID NOT LAST
**HNB2_13 byte %8.0g CANT AFFORD BALANCED MEALS
**HNB3_13 byte %8.0g CUT OR SKIP MEALS
**HNB4_13 byte %8.0g EAT LESS NOT ENOUGH MONEY
**HNB5_13 byte %8.0g GO HUNGRY NOT ENOUGH MONEY

```
use DATA_HCNS,clear
destring HHID, replace
destring PN, replace

capture drop HHIDPN
egen HHIDPN = concat(HHID PN)

destring HHIDPN, replace
sort HHIDPN

save DATA_HCNSfin, replace



keep HHIDPN HNB1_13 HNB2_13  HNB3_13 HNB4_13 HNB19_13 HNB5_13
save foodinsecurity_data2013, replace
```

use foodinsecurity_data2013,clear

tab1 HNB1_13 HNB2_13 HNB3_13 HNB4_13 HNB19_13 HNB5_13

save foodinsecurity_data2013, replace

**Source: https://www.ers.usda.gov/media/8282/short2012.pdf

**This is what it says

**i. Responses of “often” or “sometimes” on questions HH3 and HH4, and “yes” on AD1, AD2, and AD3 are coded as affirmative (yes). Responses of “almost every month” and “some months but not every month” on AD1a are coded as affirmative (yes).

**Note, there is one question that you did not mention that has a skip pattern: (a) “How often did this happen—almost every month, some months but not every month, or in only 1 or 2 months?” which relies on an affirmative response to (b) “In the **last 12 months, since last (name of current month), did (you/you or other adults in your household) ever cut the size of your meals or skip meals because there wasn't enough money for food?”

**So the algorithm/logic would be as follows:

**1. Convert all character string variable to numeric (1 or 0) based on the above description in (i) for the 6 questions

**2. If there is an NA or missing value for the skip pattern question for those that responded “No” to question (b) above, it should be converted to 0.

**3. Take the sum of the six questions

**4. If sum >= 2, 1, else 0

tab1 HNB1_13 HNB2_13 HNB3_13 HNB4_13 HNB5_13

capture drop HNB1_13r

gen HNB1_13r=.

```
replace HNB1_13r=HNB1_13
replace HNB1_13r=. if HNB1_13==99
replace HNB1_13r=4 if HNB1_13==3

capture drop HNB2_13r
gen HNB2_13r=.
replace HNB2_13r=HNB2_13
replace HNB2_13r=. if HNB2_13==99
replace HNB2_13r=4 if HNB2_13==3

capture drop HNB3_13r
gen  HNB3_13r=.
replace HNB3_13r=HNB3_13
replace HNB3_13r=4 if HNB3_13==3
replace HNB3_13r=. if HNB3_13==99

capture drop HNB4_13r
gen HNB4_13r=.
replace HNB4_13r=HNB4_13
replace HNB4_13r=. if HNB4_13==99

capture drop HNB5_13r
gen HNB5_13r=.
replace HNB5_13r=HNB5_13
replace HNB5_13r=. if HNB5_13==99
```

```
capture drop foodinsecuritymiss

egen  foodinsecuritymiss=rowmiss(HNB1_13r HNB2_13r HNB3_13r HNB3_13r HNB4_13r HNB5_13r)


capture drop foodinsecurity_tot

egen foodinsecurity_tot=anycount(HNB1_13r HNB2_13r HNB3_13r HNB3_13r HNB4_13r HNB5_13r), values(1 2 3)

replace foodinsecurity_tot=. if foodinsecuritymiss>0


tab foodinsecurity_tot


capture drop foodinsecurity_totbr

gen foodinsecurity_totbr=.

replace foodinsecurity_totbr=1 if foodinsecurity_tot>=2

replace foodinsecurity_totbr=0 if foodinsecurity_tot<2 & foodinsecurity_tot~=.


sort HHIDPN


save, replace
```

2.2. HEI-2015, 2013 WAVE

```
**********HEI 2015**********

**STEP A: RUN STATA SCRIPTS FOR LEGUMES:


use "E:\FINAL_DATA\HCNS13_R_NT",clear

save "E:\FINAL_DATA\HEI2015", replace
```

**STEP A: RUN STATA SCRIPTS FOR LEGUMES:

capture drop m_mpf m_egg m_nutsd m_soy m_fish_hi m_fish_lo legumes kcal v_total v_drkgr

gen m_mpf=C6D_FF_13+C6E_FF_13+C6F_FF_13+C6G_FF_13+C6H_FF_13+C6I_FF_13+C6J_FF_13+C6K_FF_13+C6L_FF_13+C6M_FF_13+C6N_FF_13+C6O_FF_13+C6P_FF_13+C6R_FF_13+C6S_FF_13+C6T_FF_13+C6U_FF_13+C6V_FF_13+C6W_FF_13+C6Q_FF_13

gen m_egg=C6A_FF_13+C6B_FF_13+C6C_FF_13

gen m_nutsd=C9V_FF_13+C9W_FF_13+C9X_FF_13+C9F_FF_13

gen m_soy=C5E_FF_13+C3D_FF_13

gen m_fish_hi=C6V_FF_13+C6S_FF_13

gen m_fish_lo=C6T_FF_13+C6U_FF_13+C6W_FF_13

gen legumes=C5N_FF_13+C5P_FF_13

gen kcal=CALOR_SUM

gen v_total = C5A_FF_13+C5B_FF_13+C5C_FF_13+C5D_FF_13+ C5F_FF_13+C5G_FF_13+C5H_FF_13+C5I_FF_13+C5J_FF_13+C5K_FF_13+C5L_FF_13+C5M_FF_13+C5N_FF_13+C5O_FF_13+C5P_FF_13+C5Q_FF_13+C5R_FF_13+ C5S_FF_13+C5T_FF_13+C5U_FF_13+C5V_FF_13+C5W_FF_13+C5X_FF_13+C5Y_FF_13+C5Z_FF_13+C5AA_FF_13+C5AB_FF_13

gen v_drkgr=C5T_FF_13+C5U_FF_13+C5V_FF_13

**pf_mps_total: m_mpf

**pf_eggs: m_egg

**pf_nutsds: m_nutsd

*pf_soy: m_soy

/* This program calculates legumes that get counted as meat and those that get

```
counted as veggies*/
/** This macro gets called into the program that calculates HEI 2015 scores**/

capture drop allmeat
capture drop seaplant
capture drop mbmax
capture drop meatleg
capture drop legume_added_*
capture drop meatveg
capture drop extrmeat
capture drop extrleg


gen allmeat=m_mpf+m_egg+m_nutsd+m_soy
gen seaplant=m_fish_hi+m_fish_lo+m_nutsd + m_soy
gen mbmax=2.5*(kcal/1000)
gen needmeat=mbmax-allmeat if allmeat<mbmax
gen meatleg=4*legumes
/*Needs more meat, and all beans go to meat*/
gen all2meat=1 if meatleg<=needmeat /*folks who don't meet meat max and the amount
of legumes they consume is less than the amount they need to reach mbmax*/
foreach var in allmeat seaplant {
gen legume_added_`var'=`var'+meatleg if all2meat==1
}
foreach var in v_total v_drkgr {
gen legume_added_`var'=`var' if all2meat==1
}
/*Needs more meat, and some beans go to meat, some go to veggies*/
gen meatveg=1 if meatleg>needmeat
gen extrmeat=meatleg-needmeat
```

```
gen extrleg=extrmeat/4
foreach var in allmeat seaplant {
replace legume_added_`var'=`var'+needmeat if meatveg==1 /*folks who don't meet
meat max and the amount of legumes they consume is more than the amount they need
to reach mbmax--rest go to veggies*/
}
foreach var in v_total v_drkgr {
replace legume_added_`var'=`var'+extrleg if meatveg==1
}
gen all2veg=1 if allmeat>=mbmax /*Folks who meet the meat requirement so all
legumes count as veggies*/
foreach var in allmeat seaplant {
replace legume_added_`var'=`var' if all2veg==1
}
foreach var in v_total v_drkgr {
replace legume_added_`var'=`var'+legumes if all2veg==1
}

save "E:\FINAL_DATA\HEI2015", replace


**STEP B: RUN STATA SCRIPT FOR HEI-2015


use "E:\FINAL_DATA\HEI2015", clear

capture drop monofat
capture drop polyfat
capture drop add_sug
capture drop discfat_sol
```

```
capture drop alcohol

capture drop f_total

capture drop frtjuice

capture drop wholefrt

capture drop g_whl

capture drop d_total

capture drop Satfat

capture drop sodi

capture drop g_nwhl

capture drop sfat


gen monofat=MONFAT_SUM

gen polyfat=POLY_SUM

gen add_sug=C9AH_FF_13

gen discfat_sol=ADDFAT_SOL_SUM

gen alcohol=ALCO_SUM

gen f_total=
C4A_FF_13+C4B_FF_13+C4C_FF_13+C4D_FF_13+C4E_FF_13+C4F_FF_13+C4G_FF_13+C4H_FF
_13+C4I_FF_13+C4J_FF_13+C4K_FF_13+C4L_FF_13+C4M_FF_13+C4N_FF_13+C4O_FF_13+C4P
_FF_13+C4Q_FF_13+C4R_FF_13+ C4S_FF_13+C4C_FF_13

gen frtjuice=C4I_FF_13+C4K_FF_13+C4L_FF_13+C4N_FF_13+C4O_FF_13

gen wholefrt=f_total-frtjuice

gen
g_whl=C7B_FF_13+C7F_FF_13+C7G_FF_13+C7J_FF_13+C7SA_FF_13+C9AB_FF_13+C9AC_FF_1
3+C9AD_FF_13+C9G_FF_13+C9H_FF_13

gen d_total= C3A_FF_13+C3B_FF_13+C3C_FF_13 +
C3E_FF_13+C3G_FF_13+C3H_FF_13+C3I_FF_13+C3J_FF_13+C3L_FF_13+
C3M_FF_13+C3N_FF_13+C3D_FF_13

gen Satfat=SATFAT_SUM

gen sodi=SODIUM_SUM

gen
g_nwhl=C7A_FF_13+C7C_FF_13+C7E_FF_13+C7H_FF_13+C7I_FF_13+C7K_FF_13+C7L_FF_13+C
7M_FF_13+C7N_FF_13+C7O_FF_13+C7SB_FF_13+C7T_FF_13+C9J_FF_13+C9K_FF_13+C9L_FF_
```

```
13+C9M_FF_13+C9N_FF_13+C9O_FF_13+C9P_FF_13+C9Q_FF_13+C9R_FF_13+C9S_FF_13+C9T
_FF_13+C9U_FF_13+C9Y_FF_13+C9Z_FF_13+C9AA_FF_13

gen sfat=SATFAT_SUM

gen SatFat=SATFAT_SUM


save "E:\FINAL_DATA\HEI2015", replace


capture drop monopoly

capture drop addsugc

capture drop solfatc

capture drop maxalcgr

capture drop ethcal

capture drop exalccal

capture drop emptycal10

capture drop vegden

capture drop hei*

capture drop grbnden

capture drop frtden

capture drop wholefrt

capture drop whfrden

capture drop wgrnden

capture drop monopoly

capture drop farmin

capture drop farmax

capture drop sodden

capture drop sodmin

capture drop sodmax

capture drop rgden

capture drop rgmin

capture drop rgmax
```

```
capture drop sofa*
capture drop addedsugar_perc
capture drop addsugmin
capture dorp addsugmax
capture drop heix12_addedsugar
capture drop saturatedfat_perc
capture drop saturatedfatmin
capture drop saturatedfatmax
capture drop heix13_saturatedfat



/*This do file creates HEI-2015 component densities and scores*/
gen monopoly=monofat+polyfat
gen addsugc=16*add_sug
gen solfatc=9*discfat_sol
gen maxalcgr=13*(kcal/1000)
gen ethcal=7*alcohol
gen exalccal=7*(alcohol-maxalcgr)
replace exalccal=0 if alcohol<=maxalcgr
gen emptycal10=addsugc+solfatc+exalccal
gen vegden=legume_added_v_total/(kcal/1000)
gen heix1_totalveg=5*(vegden/1.1)
replace heix1_totalveg=5 if heix1_totalveg>5
replace heix1_totalveg=0 if heix1_totalveg<0
gen grbnden=legume_added_v_drkgr/(kcal/1000)
gen heix2_greens_and_bean=5*(grbnden/.2)
replace heix2_greens_and_bean=5 if heix2_greens_and_bean>5
replace heix2_greens_and_bean=0 if heix2_greens_and_bean<0
gen frtden=f_total/(kcal/1000)
```

```
gen heix3_totalfruit=5*(frtden/.8)
replace heix3_totalfruit=5 if heix3_totalfruit>5
replace heix3_totalfruit=0 if heix3_totalfruit<0
gen wholefrt=f_total-frtjuice
gen whfrden=wholefrt/(kcal/1000)
gen heix4_wholefruit=5*(whfrden/.4)
replace heix4_wholefruit=5 if heix4_wholefruit>5
replace heix4_wholefruit=0 if heix4_wholefruit<0
gen wgrnden=g_whl/(kcal/1000)
gen heix5_wholegrain=10*(wgrnden/1.5)
replace heix5_wholegrain=10 if heix5_wholegrain>10
replace heix5_wholegrain=0 if heix5_wholegrain<0
gen dairyden=d_total/(kcal/1000)
gen heix6_totaldairy=10*(dairyden/1.3)
replace heix6_totaldairy=10 if heix6_totaldairy>10
replace heix6_totaldairy=0 if heix6_totaldairy<0
gen meatden=legume_added_allmeat/(kcal/1000)
gen heix7_totprot=5*(meatden/2.5)
replace heix7_totprot=5 if heix7_totprot>5
replace heix7_totprot=0 if heix7_totprot<0
gen seaplden=legume_added_seaplant/(kcal/1000)
gen heix8_seaplant_prot=5*(seaplden/.8)
replace heix8_seaplant_prot=5 if heix8_seaplant_prot>5
replace heix8_seaplant_prot=0 if heix8_seaplant_prot<0
gen faratio=monopoly/SatFat if SatFat>0
```

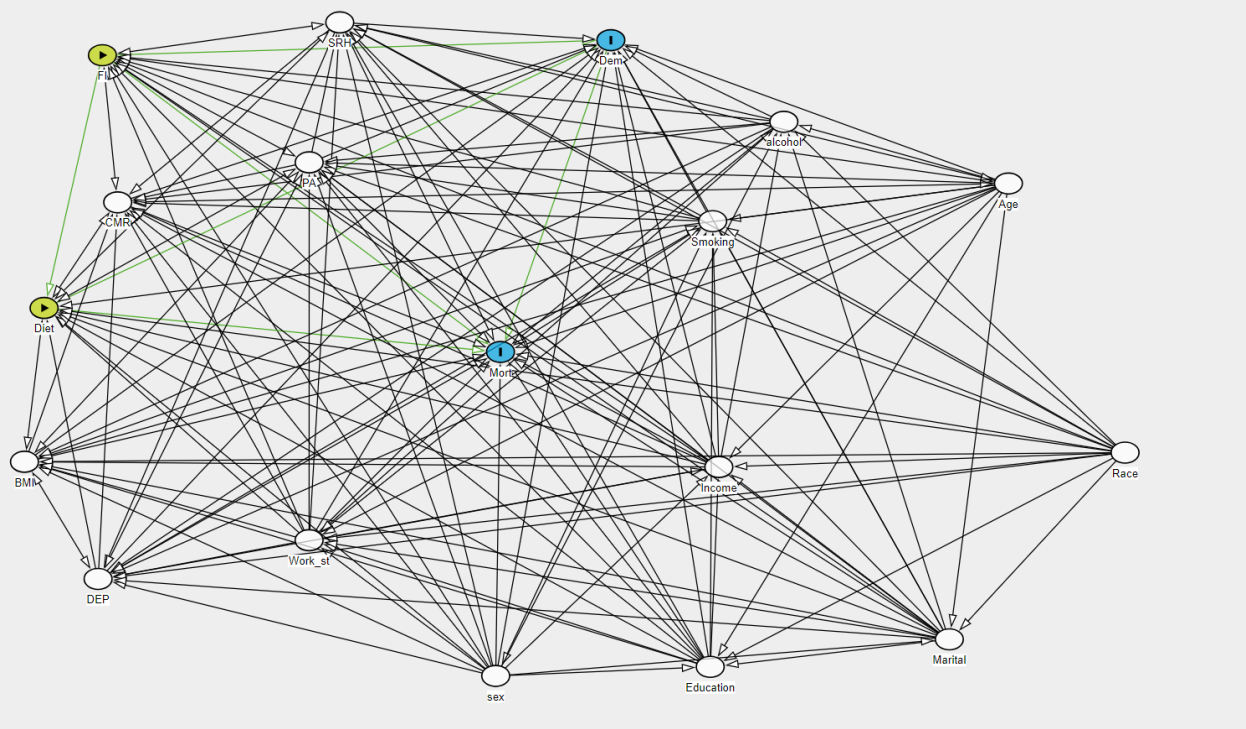
```
gen farmin=1.2
gen farmax=2.5
gen heix9_fattyacid=0 if SatFat==0 & monopoly==0
replace heix9_fattyacid=10 if SatFat==0 & monopoly>0
replace heix9_fattyacid=10 if faratio>=farmax & faratio !=.
replace heix9_fattyacid=0 if faratio<=farmin & faratio !=.
replace heix9_fattyacid=10*((faratio-farmin)/(farmax-farmin)) if faratio !=.
gen sodden=sodi/kcal
gen sodmin=1.1
gen sodmax=2
gen heix10_sodium=10
replace heix10_sodium=0 if sodden>=sodmax
replace heix10_sodium=10-(10*(sodden-sodmin)/(sodmax-sodmin))
gen rgden=g_nwhl/(kcal/1000)
gen rgmin=1.8
gen rgmax=4.3
gen heix11_refinedgrain=10
replace heix11_refinedgrain=0 if rgden>=rgmax
replace heix11_refinedgrain=10-(10*(rgden-rgmin)/(rgmax-rgmin))


gen addedsugar_perc=100*add_sug*16/kcal
gen addsugmin=6.5
gen addsugmax=26
gen heix12_addedsugar=0 if addedsugar_perc>=addsugmax
replace heix12_addedsugar=10 if addedsugar_perc<=addsugmin
replace heix12_addedsugar=10-(10*(addedsugar_perc-addsugmin)/(addsugmax-addsugmin))
```

```
gen saturatedfat_perc=100*sfat*9/kcal

gen saturatedfatmin=7

gen saturatedfatmax=15

gen heix13_saturatedfat=0 if saturatedfat_perc>=saturatedfatmax

replace heix13_saturatedfat=10 if saturatedfat_perc<=saturatedfatmin

replace heix13_saturatedfat=10-(10*(saturatedfat_perc-saturatedfatmin)/(saturatedfatmax-
saturatedfatmin))



foreach var in vegden grbnden frtden whfrden wgrnden dairyden meatden seaplden faratio sodden rgden {

replace `var'=0 if `var'==.

}


foreach var in 1_totalveg 2_greens_and_bean 3_totalfruit 4_wholefruit 5_wholegrain 6_totaldairy
7_totprot 8_seaplant 9_fattyacid 10_sodium 11_refinedgrain 12_addedsugar 13_saturatedfat {

replace heix`var'=0 if kcal==0

}

foreach var in 1_totalveg 2_greens_and_bean 3_totalfruit 4_wholefruit 5_wholegrain 6_totaldairy
7_totprot 8_seaplant 9_fattyacid 10_sodium 11_refinedgrain 12_addedsugar 13_saturatedfat {

replace heix`var'=0 if heix`var'<0 & heix`var'!=.

}

foreach var in 9_fattyacid 10_sodium 11_refinedgrain {

replace heix`var'=10 if heix`var'>10 & heix`var'!=.

}

replace heix12_addedsugar=10 if heix12_addedsugar>10 & heix12_addedsugar!=.

replace heix13_saturatedfat=10 if heix13_saturatedfat>10 & heix13_saturatedfat!=.
```

```
gen hei2015_total_score=heix1_totalveg+heix2_greens_and_bean+heix3_totalfruit+ ///
heix4_wholefruit+heix5_wholegrain+heix6_totaldairy+heix7_totprot+heix8_seaplant ///
+heix9_fattyacid+heix10_sodium+heix11_refinedgrain+heix12_addedsugar+heix13_saturatedfat

label var hei2015_total_score "total hei-2015 score"
label var heix1_totalveg "hei-2015 component 1 total vegetables"
label var heix2_greens_and_bean "hei-2015 component 2 greens and beans"
label var heix3_totalfruit "hei-2015 component 3 total fruit"
label var heix4_wholefruit "hei-2015 component 4 whole fruit"
label var heix5_wholegrain "hei-2015 component 5 whole grains"
label var heix6_totaldairy "hei-2015 component 6 dairy"
label var heix7_totprot "hei-2015 component 7 total protein foods"

label var heix8_seaplant_prot "hei-2015 component 8 seafood and plant protein"
label var heix9_fattyacid "hei-2015 component 9 fatty acid ratio"
label var heix10_sodium "hei-2015 component 10 sodium"
label var heix11_refinedgrain "hei-2015 component 11 refined grains"
label var heix12_addedsugar "hei-2015 component 12 added sugar"
label var heix13_saturatedfat "hei-2015 component 13 saturated fat"

label var vegden "density of mped total vegetables per 1000 kcal"
label var grbnden "density of mped of dark green veg and beans per 1000 kcal"
label var frtden "density of mped total fruit per 1000 kcal"
label var whfrden "density of mped whole fruit per 1000 kcal"
label var wgrnden "density of mped of whole grain per 1000 kcal"
label var dairyden "density of mped of dairy per 1000 kcal"
label var meatden "density of mped total meat/protein per 1000 kcal"
label var seaplden "denstiy of mped of seafood and plant protein per 1000 kcal"
label var faratio "fatty acid ratio"
```

```
label var sodden "density of sodium per 1000 kcal"

label var rgden "density of mped of refined grains per 1000 kcal"

label var addedsugar_perc "percent of calories from added sugar"

label var saturatedfat_perc "percent of calories from saturated fat"


save "E:\FINAL_DATA\HEI2015", replace


keep HHID PN hei* vegden grbnden frtden whfrden dairyden meatden seaplden faratio sodden rgden
addedsugar_perc saturatedfat_perc-saturatedfat_perc


destring HHID, replace

destring PN, replace


capture drop HHIDPN

egen HHIDPN = concat(HHID PN)


destring HHIDPN, replace

sort HHIDPN

su hei*


save "E:\FINAL_DATA\HEI2015_small", replace
```

SUPPLEMENTARY METHODS 3: Theoretical directed acyclic graph connecting various exposures, outcomes, mediators and covariates to be adjusted.



Minimal sufficient adjustment sets containing Age, BMI, CMR, DEP, Education, Income, Marital, PA, Race, SRH, Smoking, Work_st, alcohol, sex for estimating the total effect of FI, Diet on Mort, Dem:

- Age, BMI, CMR, DEP, Education, Income, Marital, PA, Race, SRH, Smoking, Work_st, alcohol, sex

dag {

bb="-5.398,-6.638,5.3,7.026"

Age [adjusted,pos="3.194,-3.146"]

BMI [adjusted,pos="-5.169,1.960"]

CMR [adjusted,pos="-4.376,-2.800"]

DEP [adjusted,pos="-4.543,4.109"]

Dem [outcome,pos="-0.185,-5.774"]

Diet [exposure,pos="-5.002,-0.862"]

Education [adjusted,pos="0.660,5.721"]

FI [exposure,pos="-4.506,-5.499"]

Income [adjusted,pos="0.733,2.056"]

Marital [adjusted,pos="2.692,5.217"]

Mort [outcome,pos="-1.123,-0.055"]

PA [adjusted,pos="-2.749,-3.533"]

Race [adjusted,pos="4.186,1.791"]

SRH [adjusted,pos="-2.489,-6.101"]

Smoking [adjusted,pos="0.681,-2.454"]

Work_st [adjusted,pos="-2.750,3.399"]

alcohol [adjusted,pos="1.286,-4.278"]

sex [adjusted,pos="-1.164,5.887"]

Age -> BMI

Age -> DEP

Age -> Dem

Age -> Diet

Age -> Education

Age -> Income

Age -> Marital

Age -> Mort

Age -> PA

Age -> SRH

Age -> Smoking

Age -> Work_st

Age <-> CMR

Age <-> FI

Age <-> alcohol

BMI -> CMR

BMI -> Dem

BMI -> Mort

BMI <-> DEP

BMI <-> Diet

BMI <-> PA

BMI <-> Smoking

CMR -> Dem

CMR -> Mort

CMR <-> Diet

CMR <-> FI

DEP -> CMR

DEP -> Dem

DEP -> Diet

DEP -> Mort

DEP <-> PA

DEP <-> Smoking

Dem -> Mort

Diet -> Dem

Diet -> Mort

Diet <-> PA

Diet <-> SRH

Education -> BMI

Education -> CMR

Education -> Dem

Education -> Diet

Education -> FI

Education -> Income

Education -> Mort

Education -> PA

Education -> SRH

Education -> Smoking

Education -> Work_st

FI -> Dem

FI -> Diet

FI -> Mort

FI <-> PA

FI <-> SRH

Income -> BMI

Income -> CMR

Income -> DEP

Income -> Dem

Income -> Diet

Income -> FI

Income -> Mort

Income -> PA

Income -> SRH

Income -> Smoking

Income -> alcohol

Marital -> BMI

Marital -> CMR

Marital -> DEP

Marital -> Dem

Marital -> Diet

Marital -> Education

Marital -> FI

Marital -> Income

Marital -> Mort

Marital -> PA

Marital -> SRH

Marital -> Smoking

Marital -> Work_st

Marital -> alcohol

PA -> CMR

PA -> Dem

PA -> Mort

PA -> SRH

Race -> BMI

Race -> DEP

Race -> Dem

Race -> Diet

Race -> Education

Race -> FI

Race -> Income

Race -> Marital

Race -> Mort

Race -> PA

Race -> SRH

Race -> Smoking

Race -> Work_st

Race -> alcohol

SRH -> CMR

SRH -> DEP

SRH -> Dem

SRH -> Mort

Smoking -> CMR

Smoking -> Dem

Smoking -> FI

Smoking -> Mort

Smoking -> PA

Smoking -> SRH

Smoking <-> Work_st

Smoking <-> sex

Work_st -> BMI

Work_st -> CMR

Work_st -> DEP

Work_st -> Dem

Work_st -> Diet

Work_st -> FI

Work_st -> Income

Work_st -> Mort

Work_st -> PA

Work_st -> SRH

Work_st <-> alcohol

alcohol -> BMI

alcohol -> CMR

alcohol -> DEP

alcohol -> Dem

alcohol -> FI

alcohol -> Mort

alcohol -> PA

alcohol -> SRH

sex -> BMI

sex -> CMR

sex -> DEP

sex -> Dem

sex -> Diet

sex -> Education

sex -> FI

sex -> Income

sex -> Marital

sex -> Mort

sex -> PA

sex -> SRH

sex -> Work_st

sex -> alcohol

}

QR code for DAGitty website: