Code for the manuscript titled Causal association between systolic blood pressure and quality-adjusted life years: a mendelian randomisation study.

Tamrat Befekadu Abebe*

April 10, 2025

*Centre for Medicine Use and Safety, Faculty of Pharmacy and Pharmaceutical Sciences, Monash University, tamrat.abebe@monash.edu

Contents

1 Acknowledgement

Firstly, I would like to express my deepest gratitude to **Professor Zanfina Ademi** for her exceptional supervision, unwavering support, and for giving me the opportunity to work on this project. Her guidance has been instrumental throughout the entire journey.

I am also sincerely thankful to **Dr. Jedidiah I. Morton** for his consistent encouragement, insightful advice, and for continually inspiring me to expand my skillset.

My heartfelt thanks go to **Dr. Padraig Dixon** for his invaluable input, particularly in guiding the analysis of the data, which significantly enhanced the quality of this work.

Lastly, I would like to thank **Dr. Jenni Ilomaki** for her helpful feedback and constructive comments on the overall project.

I gratefully acknowledge **Mr. Adam Livori** for being a source of inspiration to work in TeXdoc and LaTeX, and for the time he generously invested in guiding me through the essential materials for setting up this documentation.

In addition, I acknowledge the work of **Sean Harrison and colleagues**[1], from whom the current Stata code was adapted. GitHub link: https://github.com/seanharrison-bristol/Robust-causal-inference-for-long-term-policy-decisions.

This study is supported by the National Health and Medical Research Council Ideas Grants Application ID: 2012582. The funder had no input into the design of the study or decision to submit for publication.

2 Preface

This document presented the code and workflow for the manuscript titled *Causal association between systolic blood pressure and quality-adjusted life years: a mendelian randomisation study*. It detailed the data preparation (cleaning) that was performed using a dataset provided by the UK Biobank. This study was approved by the Research Ethics Committee (REC reference for UK Biobank is 11/NW/0382) of National Committee North West-Haydock, National Health Service, UK.

To generate this document, the Stata package `texdoc` was used, which is available at: http://repec.sowi.unibe.ch/stata/texdoc/.

Our code is also available at:https://github.com/tamrat-works/mr-sbp-qalys.

3 Introduction

Hypertension has long been recognised as a major risk factor for heart disease due to both oxidative and mechanical stress exerted on the arterial wall[2]. A recent study reported that for every 10 mmHg increase in systolic blood pressure (SBP), there was a 53% higher risk of atherosclerotic cardiovascular disease[3]. Mendelian randomisation (MR) studies have also demonstrated a lifetime causal association between SBP and cardiovascular disease[4, 5].

However, a gap remained in the literature regarding the lifetime causal association between SBP and health-related quality of life (HRQoL). One possible approach to demonstrate this causal association could have been a randomised controlled trial (RCT). However, the feasibility of conducting such trials, along with the limited generalisability of their findings, may hinder their applicability. An alternative approach was to conduct an observational study, which is often more cost-effective and may yield findings generalisable to the broader population. Nonetheless, observational studies carry inherent limitations, such as confounding and reverse causation[6].

Mendelian randomisation offered a potential solution to these limitations by using genetic variants as instrumental variables for modifiable traits (i.e., risk factors) associated with outcomes. These outcomes could include clinical conditions (e.g., coronary artery disease) or HRQoL.

This document was developed to showcase the application of MR techniques to investigate the lifetime association between SBP and HRQoL using data from the UK Biobank. HRQoL data were sourced from[7]. The following steps were undertaken to conduct the study.

4 Steps

- Step 1: Working on Phenotype data
- Step 2: Working on Genotype data
- Step 3: Combining Phenotype and Genotype data
- Step 4: Main analysis
- Step 5: Sensitivity analyses
- Step 6: Secondary analysis
- Step 7: Tables and Figures

4.1 Step 1: Working on Phenotype data

4.1.1 Main dataset import

The UK Biobank main dataset contained phenotype data for participants enrolled in the study. Briefly, more than 500,000 individuals were recruited across 22 centres in the UK between 2006 and 2010[8]. Hospital admission data and primary care data (general practice) for UK Biobank participants were linked to Hospital Episode Statistics (HES) in England and Scottish Morbidity Records (SMR) in Scotland up to 31 October 2022, and to the Patient Episode Database for Wales (PEDW) up to 26 May 2022.

Importing the entire dataset into STATA required significant time and computational resources. Therefore, it was more efficient to first select the key variables relevant to the study using the data dictionary.

For those primarily using the UK Biobank Research Analysis Platform (UKB RAP), data extraction was carried out through DNAnexus's JupyterLab environment, specifically using Spark JupyterLab. The following lines of Python code were used to extract the necessary variables.

```
1
2  # Building cohorts using Spark JupyterLab
3
4  # Folders
5  exome_folder = 'Population level exome OQFE variants, PLINK format -
       interim 450k release'
6  exome_field_id = '23149'
7  output_dir = '/Data/'
8
9  # Import important variables
10 import os
11
12 # Set environment variable before importing pyspark
13 os.environ['PYARROW_IGNORE_TIMEZONE'] = '1'
14
15 # Import necessary libraries
16 import pyspark
17 from pyspark.sql import SparkSession
18 from pyspark.sql.functions import col
19 import dxpy
20 import dxdata
21 import pandas as pd
22 import re
23
24 # Initialize Spark
25 # Spark initialization (Done only once; do not rerun this cell unless you
       select Kernel -> Restart kernel).
26 sc = pyspark.SparkContext()
27 spark = pyspark.sql.SparkSession(sc)
28
29 # Automatically discover dispensed dataset ID and load the dataset
30 dispensed_dataset = dxpy.find_one_data_object(
31     typename="Dataset",
```

```
    name="app*.dataset",
    folder="/",
    name_mode="glob"
)
dispensed_dataset_id = dispensed_dataset["id"]
dataset = dxdata.load_dataset(id=dispensed_dataset_id)

dataset.entities

participant = dataset['participant']

main_cohort = dxdata.load_cohort("/cohort/pheno")

field_names = [
    'eid', 'p31', 'p22001', 'p21022', 'p21003_i0', 'p738_i0', 'p22019', 'p22021', 'p53_i0', 'p40000_i0', 'p22018',
    'p22011_a0', 'p22011_a1', 'p22011_a2', 'p22011_a3', 'p22011_a4', 'p22012_a0', 'p22012_a1', 'p22012_a2',
    'p22012_a3', 'p22012_a4', 'p22013_a0', 'p22013_a1', 'p22013_a2', 'p22013_a3', 'p22013_a4', 'p22020',
    'p21000_i0', 'p48_i0', 'p49_i0', 'p50_i0', 'p54_i0', 'p4079_i0_a0', 'p4080_i0_a0', 'p4080_i0_a1',
    'p93_i0_a0', 'p93_i0_a1', 'p4079_i0_a1', 'p94_i0_a0', 'p94_i0_a1', 'p20117_i0', 'p20160_i0', 'p21001_i0',
    'p21002_i0', 'p22000', 'p22007', 'p22008', 'p22003', 'p22027', 'p22004', 'p40007_i0', 'p26201_a0',
    'p26201_a1', 'p26201_a2', 'p26201_a3', 'p22009_a1', 'p22009_a2', 'p22009_a3', 'p22009_a4', 'p22009_a5',
    'p22009_a6', 'p22009_a7', 'p22009_a8', 'p22009_a9', 'p22009_a10', 'p22009_a11', 'p22009_a12', 'p22009_a13',
    'p22009_a14', 'p22009_a15', 'p22009_a16', 'p22009_a17', 'p22009_a18', 'p22009_a19', 'p22009_a20',
    'p22009_a21', 'p22009_a22', 'p22009_a23', 'p22009_a24', 'p22009_a25', 'p22009_a26', 'p22009_a27',
    'p22009_a28', 'p22009_a29', 'p22009_a30', 'p22009_a31', 'p22009_a32', 'p22009_a33', 'p22009_a34',
    'p22009_a35', 'p22009_a36', 'p22009_a37', 'p22009_a38', 'p22009_a39', 'p22009_a40', 'p20002_i0_a0',
    'p20002_i0_a1', 'p20002_i0_a2', 'p20002_i0_a3', 'p20002_i0_a4', 'p20002_i0_a5', 'p20002_i0_a6',
    'p20002_i0_a7', 'p20002_i0_a8', 'p20002_i0_a9', 'p20002_i0_a10', 'p20002_i0_a11', 'p20002_i0_a12',
    'p20002_i0_a13', 'p20002_i0_a14', 'p20002_i0_a15', 'p20002_i0_a16', 'p20002_i0_a17', 'p20002_i0_a18',
    'p20002_i0_a19', 'p20002_i0_a20', 'p20002_i0_a21', 'p20002_i0_a22', 'p20002_i0_a23', 'p20002_i0_a24',
    'p20002_i0_a25', 'p20002_i0_a26', 'p20002_i0_a27', 'p20002_i0_a28', 'p20002_i0_a29', 'p20002_i0_a30',
    'p20002_i0_a31', 'p20002_i0_a32', 'p20002_i0_a33', 'p120098', 'p120099', 'p120100', 'p120101', 'p120102',
    'p120103', 'p120128', 'p29150', 'p29151', 'p29152', 'p29153', 'p29154'
```

```
        , 'p29155', 'p29206', 'fid',
    'p6153_i0', 'p6153_i0_a1', 'p6153_i0_a2', 'p6153_i0_a3', 'p6177_i0', '
        p6177_i0_a1', 'p6177_i0_a2',
    'p6138_i0', 'p34_i0_a0', 'p189_i0_a0', 'p30690_i0', 'p30691_i0', '
        p30760_i0', 'p30761_i0', 'p30780_i0',
    'p30781_i0', 'p30870_i0', 'p30871_i0'
]

df_main_cohort = participant.retrieve_fields(names=field_names, engine=
    dxdata.connect())

print("Initial columns:", df_main_cohort.columns)

# Rename columns for better readability
df_main_cohort = df_main_cohort.withColumnRenamed("eid", "IID")

# Add FID column -- required input format for regenie
print(type(df_main_cohort))

df_main_cohort = df_main_cohort.withColumn('FID', col('IID'))

df_main_cohort.show()

df_main_cohort_pandas = df_main_cohort.toPandas()

df_main_cohort_pandas.shape
df_main_cohort_pandas.p31.value_counts()

print("Initial columns:", df_main_cohort_pandas.columns)

# Get WES
path_to_family_file = f'/mnt/project/Bulk/Exome sequences/{exome_folder}/
    ukb{exome_field_id}_c1_b0_v1.fam'
plink_fam_df = pd.read_csv(path_to_family_file, delimiter='\s', dtype='
    object',
                           names=['FID','IID','Father ID','Mother ID', '
                               sex', 'Pheno'], engine='python')

# Intersect the phenotype file and the 450K WES .fam file
# to generate phenotype DataFrame for the 450K participants
main_wes_450k_df = df_main_cohort_pandas.join(plink_fam_df.set_index('IID'
    ), on='IID', rsuffix='_fam', how='inner')

# Drop unuseful columns from .fam file
main_wes_450k_df.drop(
    columns=['FID_fam','Father ID','Mother ID','sex_fam', 'Pheno'], axis
        =1, inplace=True, errors='ignore'
)

print(type(main_wes_450k_df))

```

```
pheno_IDs = main_wes_450k_df[["IID", "FID"]]

print(pheno_IDs)

pheno_IDs_main = df_main_cohort_pandas[["IID", "FID"]]
print(pheno_IDs_main)

# Write phenotype files to a TSV file
main_wes_450k_df.to_csv('main_wes_450k.phe', sep='\t', na_rep='NA', index=
    False, quoting=3)
main_wes_450k_df.to_csv('main_wes_450k.csv', sep='\t', na_rep='NA', index=
    False, quoting=3, escapechar='\\')
df_main_cohort_pandas.to_csv('main_cohort.csv', sep='\t', na_rep='NA',
    index=False, quoting=3, escapechar='\\')
pheno_IDs.to_csv('pheno_id_450k.phe', sep='\t', na_rep='NA', index=False,
    quoting=3)
pheno_IDs.to_csv('pheno_id_450k.csv', sep='\t', na_rep='NA', index=False,
    quoting=3, escapechar='\\')

# Write phenotype files to a TSV file (for the main (full) cohort)
df_main_cohort_pandas.to_csv('main_cohort.csv', sep='\t', na_rep='NA',
    index=False, quoting=3, escapechar='\\')
pheno_IDs_main.to_csv('pheno_id_main.csv', sep='\t', na_rep='NA', index=
    False, quoting=3, escapechar='\\')

%%bash -s "$output_dir"
dx upload main_wes_450k.phe -p --path $1 --brief

%%bash -s "$output_dir"
dx upload pheno_id_450k.phe -p --path $1 --brief

%%bash -s "$output_dir"
dx upload main_wes_450k.csv -p --path $1 --brief

%%bash -s "$output_dir"
dx upload pheno_id_450k.csv -p --path $1 --brief

%%bash -s "$output_dir"
dx upload main_cohort.csv -p --path $1 --brief

%%bash -s "$output_dir"
dx upload pheno_id_main.csv -p --path $1 --brief
```

4.1.2 Renaming variables

The important phenotype variables were selected and stored in the **main_cohort.csv** dataset. The next step involved downloading the CSV file to the local machine. The downloaded data were saved to the `stata_sbp_input` file path. Once this was completed, the next step was to prepare the data for analysis. For participants prescribed antihypertensive medications, 15 mmHg and 10 mmHg were added to their baseline systolic blood pressure (SBP) and diastolic blood pressure (DBP) measurements, respectively[9].

```
import delimited "$stata_sbp_input\main_cohort.csv", clear

*File paths

global data_source "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou
> \new_sbp_snps\stata\data_source"
global snps "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou\new_sb
> p_snps\stata\snps"
global sbp_snps "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou\ne
> w_sbp_snps\stata\sbp_snps"
global dbp_snps "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou\ne
> w_sbp_snps\stata\dbp_snps"
global sbp_dbp_snps "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelo
> u\new_sbp_snps\stata\sbp_dbp_snps"
global dx_data_sbp "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou
> \new_sbp_snps\stata\dx_data\sbp\output"
global dx_data_dbp "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou
> \new_sbp_snps\stata\dx_data\dbp\output"
global stata_sbp_input "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evang
> elou\new_sbp_snps\stata\stata_sbp_input"
global stata_sbp_output "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evan
> gelou\new_sbp_snps\stata\stata_sbp_output"
global stata_sbp_result "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evan
> gelou\new_sbp_snps\stata\stata_sbp_result"
global stata_dbp_input "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evang
> elou\new_sbp_snps\stata\stata_dbp_input"
global stata_dbp_output "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evan
> gelou\new_sbp_snps\stata\stata_dbp_output"
global stata_dbp_result "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evan
> gelou\new_sbp_snps\stata\stata_dbp_result"
global plot_png "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou\ne
> w_sbp_snps\stata\stata_sbp_plot\png"
global plot_pdf "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou\ne
> w_sbp_snps\stata\stata_sbp_plot\pdf"
global hesin_data "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou\
> new_sbp_snps\stata\hesin_data"
global r_data "C:\Users\tabe0010\OneDrive - Monash University\MR_backup_file\Articles\Evangelou\new_
> sbp_snps\stata\r_codes\r_data"

*rename the variables

 rename (iid p31 p22001 p21022 p21003_i0 p738_i0 p22019 p22021 p53_i0 p40000_i0 p22018 p22011_a0 p22
> 011_a1 p22011_a2 p22011_a3 p22011_a4 p22012_a0 p22012_a1 p22012_a2 p22012_a3 p22012_a4 p22013_a0 p
> 22013_a1 p22013_a2 p22013_a3 p22013_a4 p22020 p21000_i0 p48_i0 p49_i0 p50_i0 p54_i0 p4079_i0_a0 p4
> 080_i0_a0 p4080_i0_a1 p93_i0_a0 p93_i0_a1 p4079_i0_a1 p94_i0_a0 p94_i0_a1 p20117_i0 p20160_i0 p210
> 01_i0 p21002_i0 p22000 p22007 p22008 p22003 p22027 p22004 p40007_i0 p26201_a0 p26201_a1 p26201_a2
> p26201_a3 p22009_a1 p22009_a2 p22009_a3 p22009_a4 p22009_a5 p22009_a6 p22009_a7 p22009_a8 p22009_a
> 9 p22009_a10 p22009_a11 p22009_a12 p22009_a13 p22009_a14 p22009_a15 p22009_a16 p22009_a17 p22009_a
> 18 p22009_a19 p22009_a20 p22009_a21 p22009_a22 p22009_a23 p22009_a24 p22009_a25 p22009_a26 p22009_
> a27 p22009_a28 p22009_a29 p22009_a30 p22009_a31 p22009_a32 p22009_a33 p22009_a34 p22009_a35 p22009
> _a36 p22009_a37 p22009_a38 p22009_a39 p22009_a40 p20002_i0_a0 p20002_i0_a1 p20002_i0_a2 p20002_i0_
> a3 p20002_i0_a4 p20002_i0_a5 p20002_i0_a6 p20002_i0_a7 p20002_i0_a8 p20002_i0_a9 p20002_i0_a10 p20
> 002_i0_a11 p20002_i0_a12 p20002_i0_a13 p20002_i0_a14 p20002_i0_a15 p20002_i0_a16 p20002_i0_a17 p20
> 002_i0_a18 p20002_i0_a19 p20002_i0_a20 p20002_i0_a21 p20002_i0_a22 p20002_i0_a23 p20002_i0_a24 p20
> 002_i0_a25 p20002_i0_a26 p20002_i0_a27 p20002_i0_a28 p20002_i0_a29 p20002_i0_a30 p20002_i0_a31 p20
> 002_i0_a32 p20002_i0_a33 p120098 p120099 p120100 p120101 p120102 p120103 p120128 p29150 p29151 p29
> 152 p29153 p29154 p29155 p29206 fid p6153_i0 p6153_i0_a1 p6153_i0_a2 p6153_i0_a3 p6177_i0 p6177_i0
> _a1 p6177_i0_a2 p6138_i0 p34_i0_a0 p189_i0_a0 p30690_i0 p30691_i0 p30760_i0 p30761_i0 p30780_i0 p3
> 0781_i0 p30870_i0 p30871_i0) (iid n_31_0_0 n_22001_0_0 n_21022_0_0 n_21003_0_0 n_738_0_0 n_22019_0
> _0 n_22021_0_0 n_53_0_0 n_40000_0_0 n_22018_0_0 n_22011_0_0 n_22011_0_1 n_22011_0_2 n_22011_0_3 n_
> 22011_0_4 n_22012_0_0 n_22012_0_1 n_22012_0_2 n_22012_0_3 n_22012_0_4 n_22013_0_0 n_22013_0_1 n_22
> 013_0_2 n_22013_0_3 n_22013_0_4 n_22020_0_0 n_21000_0_0 n_48_0_0 n_49_0_0 n_50_0_0 n_54_0_0 n_4079
> _0_0 n_4080_0_0 n_4080_0_1 n_93_0_0 n_93_0_1 n_4079_0_1 n_94_0_0 n_94_0_1 n_20117_0_0 n_20160_0_0
> n_21001_0_0 n_21002_0_0 n_22000_0_0 n_22007_0_0 n_22008_0_0 n_22003_0_0 n_22027_0_0 n_22004_0_0 n_
> 40007_0_0 n_26201_0_0 n_26201_0_1 n_26201_0_2 n_26201_0_3 n_22009_0_1 n_22009_0_2 n_22009_0_3 n_22
> 009_0_4 n_22009_0_5 n_22009_0_6 n_22009_0_7 n_22009_0_8 n_22009_0_9 n_22009_0_10 n_22009_0_11 n_22
> 009_0_12 n_22009_0_13 n_22009_0_14 n_22009_0_15 n_22009_0_16 n_22009_0_17 n_22009_0_18 n_22009_0_1
> 9 n_22009_0_20 n_22009_0_21 n_22009_0_22 n_22009_0_23 n_22009_0_24 n_22009_0_25 n_22009_0_26 n_220
> 09_0_27 n_22009_0_28 n_22009_0_29 n_22009_0_30 n_22009_0_31 n_22009_0_32 n_22009_0_33 n_22009_0_34
>  n_22009_0_35 n_22009_0_36 n_22009_0_37 n_22009_0_38 n_22009_0_39 n_22009_0_40 n_20002_0_0 n_20002
```

```
> _0_1 n_20002_0_2 n_20002_0_3 n_20002_0_4 n_20002_0_5 n_20002_0_6 n_20002_0_7 n_20002_0_8 n_20002_0
> _9 n_20002_0_10 n_20002_0_11 n_20002_0_12 n_20002_0_13 n_20002_0_14 n_20002_0_15 n_20002_0_16 n_20
> 002_0_17 n_20002_0_18 n_20002_0_19 n_20002_0_20 n_20002_0_21 n_20002_0_22 n_20002_0_23 n_20002_0_2
> 4 n_20002_0_25 n_20002_0_26 n_20002_0_27 n_20002_0_28 n_20002_0_29 n_20002_0_30 n_20002_0_31 n_200
> 02_0_32 n_20002_0_33 n_120098 n_120099 n_120100 n_120101 n_120102 n_120103 n_120128 n_29150 n_2915
> 1 n_29152 n_29153 n_29154 n_29155 n_29206 fid n_6153_0_0 n_6153_0_1 n_6153_0_2 n_6153_0_3 n_6177_0
> _0 n_6177_0_1 n_6177_0_2 n_6138_0_0 n_34_0_0 n_189_0_0 n_30690_0_0 n_30691_0_0 n_30760_0_0 n_30761
> _0_0 n_30780_0_0 n_30781_0_0 n_30870_0_0 n_30871_0_0)

 *Let´s destring the variables
 foreach j of varlist n_* {
        tostring `j´, replace force
        replace `j´ = "88" if `j´ == "NA"
        destring `j´, replace
        capture replace `j´=. if `j´ == 88
 }


  *work on date variables and rename them.
  gen ts_53_0_0 = date(n_53_0_0, "YMD")
  gen ts_40000_0_0 = date(n_40000_0_0, "YMD")
  gen ts_120128 = clock(n_120128, "YMDhms")
  gen ts_29206 = date(n_29206, "YMD")
  format ts_53_0_0 ts_40000_0_0 ts_29206 %td
  format ts_120128 %tc
  replace ts_120128 = dofc(ts_120128)
  format ts_120128 %td

 order ts_53_0_0, a(n_53_0_0)
 order ts_40000_0_0, a(n_40000_0_0)
 order ts_120128, a(n_120128)
 order ts_29206, a(n_29206)

 drop n_53_0_0 n_40000_0_0 n_120128 n_29206

 *Work on systolic blood pressure measurements to take the mean of measurements (n_4080_* are automa
> tice measurements while n_93_* are manual measurements)
 gen phe_sbp=.
 replace phe_sbp = (n_4080_0_0 + n_4080_0_1)/2 if n_4080_0_0 !=. & n_4080_0_1 !=.
 replace phe_sbp = ( n_93_0_0 + n_93_0_1)/2 if n_93_0_0 !=. & n_93_0_1 !=. & n_4080_0_0 ==. & n_4080
> _0_1 ==.
 replace phe_sbp = (n_93_0_0 + n_4080_0_1)/2 if n_93_0_0 !=. & n_93_0_1 ==. & n_4080_0_0 ==. & n_408
> 0_0_1 !=.
 replace phe_sbp = (n_93_0_1 + n_4080_0_0)/2 if n_93_0_0 ==. & n_93_0_1 !=. & n_4080_0_0 !=. & n_408
> 0_0_1 ==.

*the following measurements did not have any observations
 *replace phe_sbp = (n_93_0_1 + n_4080_0_1)/2 if n_93_0_0 ==. & n_93_0_1 !=. & n_4080_0_0 ==. & n_40
> 80_0_1 !=.
 *replace phe_sbp = (n_93_0_0 + n_4080_0_0)/2 if n_93_0_0 !=. & n_93_0_1 ==. & n_4080_0_0 !=. & n_40
> 80_0_1 ==.

 order phe_sbp, a(n_93_0_1)

  *Work on diastolic blood pressure measurements to take the mean of measurements (n_4079_* are auto
> matice measurements while n_94_* are manual measurements)
 gen phe_dbp=.
 replace phe_dbp = (n_4079_0_0 + n_4079_0_1)/2 if n_4079_0_0 !=. & n_4079_0_1 !=.
 replace phe_dbp = ( n_94_0_0 + n_94_0_1)/2 if n_94_0_0 !=. & n_94_0_1 !=. & n_4079_0_0 ==. & n_4079
> _0_1 ==.
 replace phe_dbp = (n_94_0_0 + n_4079_0_1)/2 if n_94_0_0 !=. & n_94_0_1 ==. & n_4079_0_0 ==. & n_407
> 9_0_1 !=.
 replace phe_dbp = (n_94_0_1 + n_4079_0_0)/2 if n_94_0_0 ==. & n_94_0_1 !=. & n_4079_0_0 !=. & n_407
> 9_0_1 ==.

*the following measurements did not have any observations
 *replace phe_dbp = (n_94_0_1 + n_4079_0_1)/2 if n_94_0_0 ==. & n_94_0_1 !=. & n_4079_0_0 ==. & n_40
> 79_0_1 !=.
 *replace phe_dbp = (n_94_0_0 + n_4079_0_0)/2 if n_94_0_0 !=. & n_94_0_1 ==. & n_4079_0_0 !=. & n_40
```

```
> 79_0_1 ==.

 order phe_4bp, a(n_94_0_1)

 *Work on SBP (add 15 mmHg) for those prescribed antihhpertensive medications
gen phe_dbp_adj = phe_dbp+15 if n_6153_0_0 == 2 | n_6153_0_1 == 2 | n_6153_0_2 == 2 | n_6153_0_3 ==
> 2 | n_6177_0_0 == 2 | n_6177_0_1 == 2 | n_6177_0_2 == 2

 replace phe_dbp_adj = phe_dbp if phe_dbp_adj ==.
 order(phe_dbp_adj), a(phe_dbp)

  *Work on DBP (add 10 mmHg) for those prescribed antihhpertensive medications

gen phe_dbp_adj = phe_dbp+10 if n_6153_0_0 == 2 | n_6153_0_1 == 2 | n_6153_0_2 == 2 | n_6153_0_3 ==
> 2 | n_6177_0_0 == 2 | n_6177_0_1 == 2 | n_6177_0_2 == 2

 replace phe_dbp_adj = phe_dbp if phe_dbp_adj ==.
 order(phe_dbp_adj), a(phe_dbp)
save "$stata_sbp_input\main_data.dta", replace
```

4.1.3 Exclusion criteria

The revised phenotype variables were stored in the Phenotype_data dataset. The subsequent step involved excluding participants based on a set of criteria. These exclusion criteria included:

- Sex mismatch between genetic sex and reported sex
- Sex chromosome aneuploidy
- Outliers for heterozygosity or missing rate
- Ethnicity: non-White British participants
- Participants without SBP values
- Related participants (based on kinship)

```
use "$stata_sbp_input\main_data.dta", clear
 gen id_phe = iid
 order fid id_phe, a(iid)
 *Sex mismatch between genetic sex and reported
 replace n_22001_0_0 = n_31_0_0 if n_22001_0_0 ==.
 drop if n_31_0_0 != n_22001_0_0 // sex mismatch between genetic sex and reported sex 372 obs delete
> d
 drop if n_22019_0_0 == 1 // Sex chromosome aneuploidy 470 obs deleted
 drop if n_22027_0_0 == 1 // Outliers for heterozygosity or missing rate 963 obs deleted
 drop if n_21000_0_0 != 1001 // Non white British 59584 obs deleted
 drop if phe_sbp ==.  // Participants without SBP values 1,229 obs deleted
 keep if n_22021_0_0 == 0 // 148,621 observations deleted
 save "$stata_sbp_output\part_1.dta",replace
 keep id_phe ts_53_0_0
 save "$stata_sbp_input\date_attending.dta", replace
 keep id_phe
 save "$stata_sbp_input\id_list.dta", replace
```

The dataset labeled **part_1.dta** contained the variables after applying the exclusion criteria. A separate dataset containing only the date of attending the assessment centre was stored in **date_attending.dta**.

4.1.4 Hospital admission data

The **hesin_diag** dataset was downloaded from the UKB RAP using the following Python code.

```
1
2  #Building cohorts using Spark JupyterLab
3  #Import important variables
4
5  import os
6
7  # Set environment variable before importing pyspark
8
9  os.environ['PYARROW_IGNORE_TIMEZONE'] = '1'
10
11 # Import necessary libraries
12 import pyspark
13 from pyspark import SparkConf, SparkContext
14 from pyspark.sql import SparkSession
```

```
from pyspark.sql.functions import col
import dxpy
import dxdata
import pandas as pd
import re

# Initialize SparkConf with the necessary configurations
conf = SparkConf() \
    .setAppName("HESIN Data Analysis") \
    .set("spark.kryoserializer.buffer.max", "1g")

# Initialize Spark
# Spark initialization (Done only once; do not rerun this cell unless you select Kernel -> Restart kernel).

sc = pyspark.SparkContext()
spark = pyspark.sql.SparkSession(sc)


print(f"Kryo serializer buffer max size set to: {conf.get('spark.kryoserializer.buffer.max')}")


# Automatically discover dispensed dataset ID and load the dataset
dispensed_dataset = dxpy.find_one_data_object(
    typename="Dataset",
    name="app*.dataset",
    folder="/",
    name_mode="glob")
dispensed_dataset_id = dispensed_dataset["id"]
dataset = dxdata.load_dataset(id=dispensed_dataset_id)

dataset.entities

participant = dataset['hesin_diag']

print(type(participant))

help(participant)

field_names = ['eid', 'ins_index', 'arr_index', 'level', 'diag_icd9', 'diag_icd9_nb', 'diag_icd10', 'diag_icd10_nb']

df_hesin_diag = participant.retrieve_fields(names=field_names, engine=dxdata.connect())

print("Initial columns:", df_hesin_diag.columns)

print(type(df_hesin_diag))

df_hesin_diag.show(5)
```

```
df_hesin_diag.count()

df_hesin_diag = df_hesin_diag.repartition(10)

df_hesin_diag_main=df_hesin_diag.toPandas()

print(type(df_hesin_diag_main))

print(df_hesin_diag_main)

df_hesin_diag_main.to_csv('hesin_diag_main.csv', index=False)

%%bash
dx upload hesin_diag_main.csv --dest project-Gkz56gjJx5g1zB269F0ybP63:/
    Data/
```

The **hesin_main** dataset, which included the date of diagnosis for participants admitted to hospitals, was also downloaded. The following Python code was used to create the necessary file from the UKB RAP.

```
#Building cohorts using Spark JupyterLab
#Import important variables

import os

# Set environment variable before importing pyspark

os.environ['PYARROW_IGNORE_TIMEZONE'] = '1'

# Import necessary libraries
import pyspark
from pyspark import SparkConf, SparkContext
from pyspark.sql import SparkSession
from pyspark.sql.functions import col
import dxpy
import dxdata
import pandas as pd
import re

# Initialize SparkConf with the necessary configurations
conf = SparkConf() \
    .setAppName("HESIN Data Analysis") \
    .set("spark.kryoserializer.buffer.max", "1g")

# Initialize Spark
# Spark initialization (Done only once; do not rerun this cell unless you
    select Kernel -> Restart kernel).

sc = pyspark.SparkContext()
spark = pyspark.sql.SparkSession(sc)

```

```
31
32 print(f"Kryo serializer buffer max size set to: {conf.get('spark.
       kryoserializer.buffer.max')}")
33
34
35 # Automatically discover dispensed dataset ID and load the dataset
36 dispensed_dataset = dxpy.find_one_data_object(
37     typename="Dataset",
38     name="app*.dataset",
39     folder="/",
40     name_mode="glob")
41 dispensed_dataset_id = dispensed_dataset["id"]
42 dataset = dxdata.load_dataset(id=dispensed_dataset_id)
43
44
45 dataset.entities
46
47 participant = dataset['hesin']
48
49 print(type(participant))
50
51 help(participant)
52
53 field_names = ['eid', 'ins_index', 'dsource', 'epistart', 'epiend', '
       epidur', 'admidate', 'disdate']
54
55
56 df_hesin = participant.retrieve_fields(names=field_names, engine=dxdata.
       connect())
57
58 print("Initial columns:", df_hesin.columns)
59
60
61 print(type(df_hesin))
62
63
64 df_hesin.show(5)
65
66 df_hesin.count()
67
68 df_hesin = df_hesin.repartition(10)
69
70 df_hesin_main=df_hesin.toPandas()
71
72 print(type(df_hesin_main))
73
74 print(df_hesin_main)
75
76 df_hesin_main.to_csv('hesin_main.csv', index=False)
77
78 %%bash
```

```
79 dx upload hesin_main.csv --dest project-Gkz56gjJx5g1zB269F0ybP63:/Data/
```

Save the **hesin_diag_main.csv** and **hesin_main.csv** files to the **hesin_data** file path on the local machine.

The next step involved working with the ICD codes. The **hesin_diag_main.csv** file contained both ICD-9 and ICD-10 diagnosis codes. The analysis first focused on the ICD-10 codes before proceeding to the ICD-9 codes. In addition to diagnosis codes, the dataset also included variables labeled **instance** (`ins_index`), **array** (`arr_index`), and **level**.

Instance indicates how many occasions participants have measurements performed. There are three categories:

- **Singular**: only one instance can be present, for example sex or year-of-birth
- **Defined**: more than one instance may be present, and each instance represents a fixed identifiable set of results across all participants
- **Variable**: more than one instance may be present, however there is no correspondence between (say) the 3rd instance for one participant and the 3rd instance for another

Array describes whether there are multiple data items for a given participant instance. There are two categories:

- **Single**: only one data item is present for each participant, for instance the answer to "What is your favourite colour of the rainbow?"
- **Multiple**: more than one data item may be present for each participant, for instance the answer to "Which colours of the rainbow do you like?"

level describes whether the diagnosis is primary (1) or secondary (2)

4.1.4.1 Working on ICD-10 diagnosis codes

The next line of stata codes will do the followng tasks:

- filter ICD-10 codes
- merge with the **hesin_main.dta** dataset (we need to convert the csv file to dta for convenience)
- identify admission or episode start date
- identify discharge or episode end date (if required)
- merge with **data_attending.dta** file which contains participants' date of attending the UK Biobank assessment centre
- create 240 dummy variables for the conditions we would like to work on the next step.

```
*Let´s save the hesin_main.csv in stata file format (hesin_main.dta)

 import delimited "$hesin_data\hesin_main.csv", clear
 save "$hesin_data\hesin_main.dta", replace
```

```
* Let´s work on the ICD 10 codes
 import delimited "$hesin_data\hesin_diag_main.csv", clear
 drop diag_icd9_nb diag_icd10_nb diag_icd9
 egen idlong = concat(eid ins_index arr_index level), format(%20.0g) p(",")
 drop eid ins_index level
 reshape wide diag_icd10, i(idlong) j(arr_index)
 split idlong, p(,)
 rename idlong1 eid
 rename idlong2 ins_index
 rename idlong3 arr_index
 rename idlong4 level
 destring eid, replace
 destring ins_index, replace
 destring arr_index, replace
 destring level, replace
 drop idlong
 order eid ins_index arr_index level diag*
 merge m:1 eid ins_index using "$hesin_data\hesin_main.dta"
 drop if _merge == 2
 drop _merge*
 drop if diag_icd100 == "" & diag_icd101 == "" & diag_icd102 == "" & diag_icd103 == "" & diag_icd104
>  == "" & diag_icd105 == "" & diag_icd106 == "" & diag_icd107 == "" & diag_icd108 == "" & diag_icd1
> 09 == "" & diag_icd1010 == "" & diag_icd1011 == "" & diag_icd1012 == "" & diag_icd1013 == "" & dia
> g_icd1014 == "" & diag_icd1015 == "" & diag_icd1016 == "" & diag_icd1017 == "" & diag_icd1018 == "
> " & diag_icd1019 == "" & diag_icd1020 == ""
 gen epistart_1 = date(epistart, "YMD")
 gen epiend_1 = date(epiend, "YMD")
 gen admidate_1 = date(admidate,"YMD")
 gen disdate_1 = date(disdate, "YMD")
 format epistart_1 epiend_1 admidate_1 disdate_1 %td
 drop epistart epiend admidate disdate
 rename epistart_1 epistart
 rename epiend_1 epiend
 rename admidate_1 admidate
 rename disdate_1 disdate
 replace epistart = admidate if epistart ==.
 replace epiend = disdate if epiend ==.
 drop admidate disdate
 rename eid id_phe
 merge m:1 id_phe using "$stata_sbp_input\date_attending.dta", keep(3) nogen
 order ts_53_0_0, a(epistart)
 rename ts_53_0_0 date_baseline
 rename epistart date_episode
 *Generate health condition phenotypes
 forvalues i = 1/240 {
        gen v`i´ = .
        format v`i´ %td
 }
 save "$hesin_data\hesin_icd10_dates.dta", replace
```

The **hesin_icd10_dates.dta** dataset was large, and processing it in full required several hours to complete the analysis. To improve efficiency, the dataset was divided into smaller subsets with reduced sample sizes. Within each subset, the date of diagnosis was identified for 240 predefined comorbidities.

```
local j = 250000
    local k = 1
forval i = 1/39 {

        di `i´ " " `k´ " " `j´
```

```
        use "$hesin_data\hesin_icd10_dates.dta", clear
        keep if _n >= `k´ & _n <= `j´
    save "$hesin_data\hesin_icd10_dates_`i´.dta", replace

        local k = `j´+1
        local j = `j´+250000

        }


 *Loop through HES diagnosis fields to find relevant ICD10 codes

 forval j = 1/39 {
        use "$hesin_data\hesin_icd10_dates_`j´.dta",clear
        forval i = 0/20 {

        dis "Looking through `i´ of 20 diagnosis fields for ICD10 codes of data `j´ "
qui replace v1 = date_episode if strpos(diag_icd10`i´,"B20") > 0
qui replace v2 = date_episode if strpos(diag_icd10`i´,"D86") > 0
qui replace v3 = date_episode if strpos(diag_icd10`i´,"C14") > 0
qui replace v4 = date_episode if strpos(diag_icd10`i´,"C16") > 0
qui replace v5 = date_episode if strpos(diag_icd10`i´,"C18") > 0
qui replace v6 = date_episode if strpos(diag_icd10`i´,"C22") > 0
qui replace v7 = date_episode if strpos(diag_icd10`i´,"C33") > 0
qui replace v7 = date_episode if strpos(diag_icd10`i´,"C34") > 0
qui replace v8 = date_episode if strpos(diag_icd10`i´,"C43") > 0
qui replace v9 = date_episode if strpos(diag_icd10`i´,"C44") > 0
qui replace v10 = date_episode if strpos(diag_icd10`i´,"C50") > 0
qui replace v11 = date_episode if strpos(diag_icd10`i´,"C55") > 0
qui replace v12 = date_episode if strpos(diag_icd10`i´,"C53") > 0
qui replace v13 = date_episode if strpos(diag_icd10`i´,"C61") > 0
qui replace v14 = date_episode if strpos(diag_icd10`i´,"C67") > 0
qui replace v15 = date_episode if strpos(diag_icd10`i´,"C64") > 0
qui replace v15 = date_episode if strpos(diag_icd10`i´,"C65") > 0
qui replace v15 = date_episode if strpos(diag_icd10`i´,"C66") > 0
qui replace v15 = date_episode if strpos(diag_icd10`i´,"C68") > 0
qui replace v16 = date_episode if strpos(diag_icd10`i´,"C71") > 0
qui replace v17 = date_episode if strpos(diag_icd10`i´,"C76") > 0
qui replace v18 = date_episode if strpos(diag_icd10`i´,"C80") > 0
qui replace v19 = date_episode if strpos(diag_icd10`i´,"C81") > 0
qui replace v20 = date_episode if strpos(diag_icd10`i´,"C82") > 0
qui replace v20 = date_episode if strpos(diag_icd10`i´,"C83") > 0
qui replace v20 = date_episode if strpos(diag_icd10`i´,"C84") > 0
qui replace v20 = date_episode if strpos(diag_icd10`i´,"C85") > 0
qui replace v20 = date_episode if strpos(diag_icd10`i´,"C96") > 0
qui replace v21 = date_episode if strpos(diag_icd10`i´,"C95") > 0
qui replace v22 = date_episode if strpos(diag_icd10`i´,"D12") > 0
qui replace v22 = date_episode if strpos(diag_icd10`i´,"D13") > 0
qui replace v22 = date_episode if strpos(diag_icd10`i´,"D20") > 0
qui replace v23 = date_episode if strpos(diag_icd10`i´,"D21") > 0
qui replace v24 = date_episode if strpos(diag_icd10`i´,"D22") > 0
qui replace v24 = date_episode if strpos(diag_icd10`i´,"D23") > 0
qui replace v25 = date_episode if strpos(diag_icd10`i´,"D24") > 0
qui replace v26 = date_episode if strpos(diag_icd10`i´,"D25") > 0
qui replace v27 = date_episode if strpos(diag_icd10`i´,"D36") > 0
qui replace v28 = date_episode if strpos(diag_icd10`i´,"D45") > 0
qui replace v28 = date_episode if strpos(diag_icd10`i´,"D46") > 0
qui replace v28 = date_episode if strpos(diag_icd10`i´,"D47") > 0
qui replace v28 = date_episode if strpos(diag_icd10`i´,"D48") > 0
qui replace v29 = date_episode if strpos(diag_icd10`i´,"D49") > 0
qui replace v30 = date_episode if strpos(diag_icd10`i´,"E01") > 0
qui replace v30 = date_episode if strpos(diag_icd10`i´,"E04") > 0
qui replace v31 = date_episode if strpos(diag_icd10`i´,"E05") > 0
qui replace v32 = date_episode if strpos(diag_icd10`i´,"E02") > 0
qui replace v32 = date_episode if strpos(diag_icd10`i´,"E03") > 0
qui replace v33 = date_episode if strpos(diag_icd10`i´,"E06") > 0
qui replace v34 = date_episode if strpos(diag_icd10`i´,"E07") > 0
```

```
qui replace v35 = date_episode if strpos(diag_icd10`i´,"E10") > 0
qui replace v35 = date_episode if strpos(diag_icd10`i´,"E11") > 0
qui replace v35 = date_episode if strpos(diag_icd10`i´,"E12") > 0
qui replace v35 = date_episode if strpos(diag_icd10`i´,"E13") > 0
qui replace v35 = date_episode if strpos(diag_icd10`i´,"E14") > 0
qui replace v36 = date_episode if strpos(diag_icd10`i´,"E16") > 0
qui replace v37 = date_episode if strpos(diag_icd10`i´,"E20") > 0
qui replace v37 = date_episode if strpos(diag_icd10`i´,"E21") > 0
qui replace v38 = date_episode if strpos(diag_icd10`i´,"E22") > 0
qui replace v38 = date_episode if strpos(diag_icd10`i´,"E23") > 0
qui replace v39 = date_episode if strpos(diag_icd10`i´,"E24") > 0
qui replace v39 = date_episode if strpos(diag_icd10`i´,"E25") > 0
qui replace v39 = date_episode if strpos(diag_icd10`i´,"E26") > 0
qui replace v39 = date_episode if strpos(diag_icd10`i´,"E27") > 0
qui replace v40 = date_episode if strpos(diag_icd10`i´,"E28") > 0
qui replace v41 = date_episode if strpos(diag_icd10`i´,"E30") > 0
qui replace v41 = date_episode if strpos(diag_icd10`i´,"E34") > 0
qui replace v41 = date_episode if strpos(diag_icd10`i´,"E35") > 0
qui replace v42 = date_episode if strpos(diag_icd10`i´,"E73") > 0
qui replace v42 = date_episode if strpos(diag_icd10`i´,"E74") > 0
qui replace v42 = date_episode if strpos(diag_icd10`i´,"E77") > 0
qui replace v43 = date_episode if strpos(diag_icd10`i´,"E71") > 0
qui replace v43 = date_episode if strpos(diag_icd10`i´,"E75") > 0
qui replace v43 = date_episode if strpos(diag_icd10`i´,"E78") > 0
qui replace v44 = date_episode if strpos(diag_icd10`i´,"M10") > 0
qui replace v44 = date_episode if strpos(diag_icd10`i´,"M1A") > 0
qui replace v45 = date_episode if strpos(diag_icd10`i´,"E83") > 0
qui replace v46 = date_episode if strpos(diag_icd10`i´,"E76") > 0
qui replace v46 = date_episode if strpos(diag_icd10`i´,"E79") > 0
qui replace v46 = date_episode if strpos(diag_icd10`i´,"E80") > 0
qui replace v46 = date_episode if strpos(diag_icd10`i´,"E84") > 0
qui replace v46 = date_episode if strpos(diag_icd10`i´,"E85") > 0
qui replace v46 = date_episode if strpos(diag_icd10`i´,"E88") > 0
qui replace v47 = date_episode if strpos(diag_icd10`i´,"E65") > 0
qui replace v47 = date_episode if strpos(diag_icd10`i´,"E66") > 0
qui replace v47 = date_episode if strpos(diag_icd10`i´,"E67") > 0
qui replace v47 = date_episode if strpos(diag_icd10`i´,"E68") > 0
qui replace v48 = date_episode if strpos(diag_icd10`i´,"D80") > 0
qui replace v48 = date_episode if strpos(diag_icd10`i´,"D81") > 0
qui replace v48 = date_episode if strpos(diag_icd10`i´,"D82") > 0
qui replace v48 = date_episode if strpos(diag_icd10`i´,"D83") > 0
qui replace v48 = date_episode if strpos(diag_icd10`i´,"D84") > 0
qui replace v48 = date_episode if strpos(diag_icd10`i´,"D89") > 0
qui replace v49 = date_episode if strpos(diag_icd10`i´,"D50") > 0
qui replace v50 = date_episode if strpos(diag_icd10`i´,"D55") > 0
qui replace v50 = date_episode if strpos(diag_icd10`i´,"D56") > 0
qui replace v50 = date_episode if strpos(diag_icd10`i´,"D57") > 0
qui replace v50 = date_episode if strpos(diag_icd10`i´,"D58") > 0
qui replace v51 = date_episode if strpos(diag_icd10`i´,"D62") > 0
qui replace v51 = date_episode if strpos(diag_icd10`i´,"D63") > 0
qui replace v51 = date_episode if strpos(diag_icd10`i´,"D64") > 0
qui replace v52 = date_episode if strpos(diag_icd10`i´,"D65") > 0
qui replace v52 = date_episode if strpos(diag_icd10`i´,"D66") > 0
qui replace v52 = date_episode if strpos(diag_icd10`i´,"D67") > 0
qui replace v52 = date_episode if strpos(diag_icd10`i´,"D68") > 0
qui replace v53 = date_episode if strpos(diag_icd10`i´,"D69") > 0
qui replace v54 = date_episode if strpos(diag_icd10`i´,"D70") > 0
qui replace v54 = date_episode if strpos(diag_icd10`i´,"D71") > 0
qui replace v54 = date_episode if strpos(diag_icd10`i´,"D72") > 0
qui replace v55 = date_episode if strpos(diag_icd10`i´,"D73") > 0
qui replace v55 = date_episode if strpos(diag_icd10`i´,"D74") > 0
qui replace v55 = date_episode if strpos(diag_icd10`i´,"D75") > 0
qui replace v55 = date_episode if strpos(diag_icd10`i´,"D77") > 0
qui replace v55 = date_episode if strpos(diag_icd10`i´,"D89") > 0
qui replace v55 = date_episode if strpos(diag_icd10`i´,"I88") > 0
qui replace v56 = date_episode if strpos(diag_icd10`i´,"F02") > 0
qui replace v56 = date_episode if strpos(diag_icd10`i´,"F04") > 0
qui replace v56 = date_episode if strpos(diag_icd10`i´,"F06") > 0
```

```
qui replace v56 = date_episode if strpos(diag_icd10`i´,"F09") > 0
qui replace v57 = date_episode if strpos(diag_icd10`i´,"F20") > 0
qui replace v57 = date_episode if strpos(diag_icd10`i´,"F21") > 0
qui replace v57 = date_episode if strpos(diag_icd10`i´,"F23") > 0
qui replace v57 = date_episode if strpos(diag_icd10`i´,"F25") > 0
qui replace v58 = date_episode if strpos(diag_icd10`i´,"F30") > 0
qui replace v58 = date_episode if strpos(diag_icd10`i´,"F31") > 0
qui replace v58 = date_episode if strpos(diag_icd10`i´,"F32") > 0
qui replace v58 = date_episode if strpos(diag_icd10`i´,"F33") > 0
qui replace v58 = date_episode if strpos(diag_icd10`i´,"F34") > 0
qui replace v58 = date_episode if strpos(diag_icd10`i´,"F38") > 0
qui replace v58 = date_episode if strpos(diag_icd10`i´,"F39") > 0
qui replace v59 = date_episode if strpos(diag_icd10`i´,"F22") > 0
qui replace v59 = date_episode if strpos(diag_icd10`i´,"F24") > 0
qui replace v60 = date_episode if strpos(diag_icd10`i´,"F23") > 0
qui replace v60 = date_episode if strpos(diag_icd10`i´,"F28") > 0
qui replace v60 = date_episode if strpos(diag_icd10`i´,"F29") > 0
qui replace v61 = date_episode if strpos(diag_icd10`i´,"F40") > 0
qui replace v61 = date_episode if strpos(diag_icd10`i´,"F41") > 0
qui replace v61 = date_episode if strpos(diag_icd10`i´,"F42") > 0
qui replace v61 = date_episode if strpos(diag_icd10`i´,"F44") > 0
qui replace v61 = date_episode if strpos(diag_icd10`i´,"F45") > 0
qui replace v61 = date_episode if strpos(diag_icd10`i´,"F48") > 0
qui replace v61 = date_episode if strpos(diag_icd10`i´,"F68") > 0
qui replace v62 = date_episode if strpos(diag_icd10`i´,"F60") > 0
qui replace v62 = date_episode if strpos(diag_icd10`i´,"F61") > 0
qui replace v62 = date_episode if strpos(diag_icd10`i´,"F62") > 0
qui replace v62 = date_episode if strpos(diag_icd10`i´,"F68") > 0
qui replace v62 = date_episode if strpos(diag_icd10`i´,"F69") > 0
qui replace v63 = date_episode if strpos(diag_icd10`i´,"F52") > 0
qui replace v63 = date_episode if strpos(diag_icd10`i´,"F64") > 0
qui replace v63 = date_episode if strpos(diag_icd10`i´,"F65") > 0
qui replace v63 = date_episode if strpos(diag_icd10`i´,"F66") > 0
qui replace v64 = date_episode if strpos(diag_icd10`i´,"F10") > 0
qui replace v65 = date_episode if strpos(diag_icd10`i´,"F11") > 0
qui replace v65 = date_episode if strpos(diag_icd10`i´,"F12") > 0
qui replace v65 = date_episode if strpos(diag_icd10`i´,"F13") > 0
qui replace v65 = date_episode if strpos(diag_icd10`i´,"F14") > 0
qui replace v65 = date_episode if strpos(diag_icd10`i´,"F15") > 0
qui replace v65 = date_episode if strpos(diag_icd10`i´,"F16") > 0
qui replace v65 = date_episode if strpos(diag_icd10`i´,"F17") > 0
qui replace v65 = date_episode if strpos(diag_icd10`i´,"F18") > 0
qui replace v65 = date_episode if strpos(diag_icd10`i´,"F19") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F10") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F11") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F12") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F13") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F14") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F15") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F16") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F17") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F18") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F19") > 0
qui replace v66 = date_episode if strpos(diag_icd10`i´,"F55") > 0
qui replace v67 = date_episode if strpos(diag_icd10`i´,"F458") > 0 // changed from "F45.8"
qui replace v68 = date_episode if strpos(diag_icd10`i´,"F50") > 0
qui replace v68 = date_episode if strpos(diag_icd10`i´,"F51") > 0
qui replace v68 = date_episode if strpos(diag_icd10`i´,"F95") > 0
qui replace v68 = date_episode if strpos(diag_icd10`i´,"F98") > 0
qui replace v69 = date_episode if strpos(diag_icd10`i´,"F431") > 0 // changed from "F43.1"
qui replace v69 = date_episode if strpos(diag_icd10`i´,"F432") > 0 // changed from "F43.2"
qui replace v69 = date_episode if strpos(diag_icd10`i´,"F438") > 0 // changed from "F43.8"
qui replace v69 = date_episode if strpos(diag_icd10`i´,"F439") > 0 // changed from "F43.9"
qui replace v69 = date_episode if strpos(diag_icd10`i´,"F930") > 0 // changed from "F93.0"
qui replace v69 = date_episode if strpos(diag_icd10`i´,"F948") > 0 // changed from "F94.8"
qui replace v70 = date_episode if strpos(diag_icd10`i´,"F07") > 0
qui replace v71 = date_episode if strpos(diag_icd10`i´,"F329") > 0 // changed from "F32.9"
qui replace v72 = date_episode if strpos(diag_icd10`i´,"F63") > 0
```

```
qui replace v72 = date_episode if strpos(diag_icd10`i´,"F91") > 0
qui replace v72 = date_episode if strpos(diag_icd10`i´,"F92") > 0
qui replace v73 = date_episode if strpos(diag_icd10`i´,"F90") > 0
qui replace v74 = date_episode if strpos(diag_icd10`i´,"F80") > 0
qui replace v74 = date_episode if strpos(diag_icd10`i´,"F81") > 0
qui replace v74 = date_episode if strpos(diag_icd10`i´,"F82") > 0
qui replace v74 = date_episode if strpos(diag_icd10`i´,"F83") > 0
qui replace v74 = date_episode if strpos(diag_icd10`i´,"F88") > 0
qui replace v74 = date_episode if strpos(diag_icd10`i´,"F89") > 0
qui replace v75 = date_episode if strpos(diag_icd10`i´,"F78") > 0
qui replace v75 = date_episode if strpos(diag_icd10`i´,"F79") > 0
qui replace v76 = date_episode if strpos(diag_icd10`i´,"G30") > 0
qui replace v76 = date_episode if strpos(diag_icd10`i´,"G31") > 0
qui replace v76 = date_episode if strpos(diag_icd10`i´,"G91") > 0
qui replace v76 = date_episode if strpos(diag_icd10`i´,"G93") > 0
qui replace v76 = date_episode if strpos(diag_icd10`i´,"G94") > 0
qui replace v77 = date_episode if strpos(diag_icd10`i´,"G20") > 0
qui replace v77 = date_episode if strpos(diag_icd10`i´,"G21") > 0
qui replace v77 = date_episode if strpos(diag_icd10`i´,"G22") > 0
qui replace v78 = date_episode if strpos(diag_icd10`i´,"G10") > 0
qui replace v78 = date_episode if strpos(diag_icd10`i´,"G23") > 0
qui replace v78 = date_episode if strpos(diag_icd10`i´,"G24") > 0
qui replace v78 = date_episode if strpos(diag_icd10`i´,"G25") > 0
qui replace v78 = date_episode if strpos(diag_icd10`i´,"G26") > 0
qui replace v79 = date_episode if strpos(diag_icd10`i´,"G320") > 0 // changed from "G32.0"
qui replace v79 = date_episode if strpos(diag_icd10`i´,"G95") > 0
qui replace v80 = date_episode if strpos(diag_icd10`i´,"G90") > 0
qui replace v81 = date_episode if strpos(diag_icd10`i´,"G35") > 0
qui replace v82 = date_episode if strpos(diag_icd10`i´,"G80") > 0
qui replace v83 = date_episode if strpos(diag_icd10`i´,"G82") > 0
qui replace v83 = date_episode if strpos(diag_icd10`i´,"G83") > 0
qui replace v84 = date_episode if strpos(diag_icd10`i´,"G40") > 0
qui replace v84 = date_episode if strpos(diag_icd10`i´,"G41") > 0
qui replace v85 = date_episode if strpos(diag_icd10`i´,"G43") > 0
qui replace v86 = date_episode if strpos(diag_icd10`i´,"G474") > 0 // changed from "G47.4"
qui replace v87 = date_episode if strpos(diag_icd10`i´,"G93") > 0
qui replace v87 = date_episode if strpos(diag_icd10`i´,"G94") > 0
qui replace v88 = date_episode if strpos(diag_icd10`i´,"G96") > 0
qui replace v88 = date_episode if strpos(diag_icd10`i´,"G97") > 0
qui replace v88 = date_episode if strpos(diag_icd10`i´,"G98") > 0
qui replace v88 = date_episode if strpos(diag_icd10`i´,"G99") > 0
qui replace v89 = date_episode if strpos(diag_icd10`i´,"G51") > 0
qui replace v90 = date_episode if strpos(diag_icd10`i´,"G54") > 0
qui replace v90 = date_episode if strpos(diag_icd10`i´,"G55") > 0
qui replace v91 = date_episode if strpos(diag_icd10`i´,"G56") > 0
qui replace v91 = date_episode if strpos(diag_icd10`i´,"G587") > 0 // changed from "G58.7"
qui replace v92 = date_episode if strpos(diag_icd10`i´,"G57") > 0
qui replace v92 = date_episode if strpos(diag_icd10`i´,"G580") > 0 // changed from "G58.0"
qui replace v92 = date_episode if strpos(diag_icd10`i´,"G588") > 0 // changed from "G58.8"
qui replace v92 = date_episode if strpos(diag_icd10`i´,"G589") > 0 // changed from "G58.9"
qui replace v92 = date_episode if strpos(diag_icd10`i´,"G59") > 0
qui replace v93 = date_episode if strpos(diag_icd10`i´,"G60") > 0
qui replace v94 = date_episode if strpos(diag_icd10`i´,"G71") > 0
qui replace v94 = date_episode if strpos(diag_icd10`i´,"G72") > 0
qui replace v94 = date_episode if strpos(diag_icd10`i´,"G73") > 0
qui replace v95 = date_episode if strpos(diag_icd10`i´,"H44") > 0
qui replace v95 = date_episode if strpos(diag_icd10`i´,"H45") > 0
qui replace v96 = date_episode if strpos(diag_icd10`i´,"H33") > 0
qui replace v97 = date_episode if strpos(diag_icd10`i´,"H34") > 0
qui replace v97 = date_episode if strpos(diag_icd10`i´,"H35") > 0
qui replace v97 = date_episode if strpos(diag_icd10`i´,"H36") > 0
qui replace v98 = date_episode if strpos(diag_icd10`i´,"H40") > 0
qui replace v98 = date_episode if strpos(diag_icd10`i´,"H42") > 0
qui replace v99 = date_episode if strpos(diag_icd10`i´,"H25") > 0
qui replace v99 = date_episode if strpos(diag_icd10`i´,"H26") > 0
qui replace v99 = date_episode if strpos(diag_icd10`i´,"H28") > 0
qui replace v100 = date_episode if strpos(diag_icd10`i´,"H52") > 0
qui replace v101 = date_episode if strpos(diag_icd10`i´,"H53") > 0
```

```
qui replace v102 = date_episode if strpos(diag_icd10`i´,"H54") > 0
qui replace v103 = date_episode if strpos(diag_icd10`i´,"H16") > 0
qui replace v104 = date_episode if strpos(diag_icd10`i´,"H10") > 0
qui replace v104 = date_episode if strpos(diag_icd10`i´,"H11") > 0
qui replace v104 = date_episode if strpos(diag_icd10`i´,"H13") > 0
qui replace v105 = date_episode if strpos(diag_icd10`i´,"H02") > 0
qui replace v106 = date_episode if strpos(diag_icd10`i´,"H04") > 0
qui replace v106 = date_episode if strpos(diag_icd10`i´,"H060") > 0 // changed from "H06.0"
qui replace v107 = date_episode if strpos(diag_icd10`i´,"H46") > 0
qui replace v107 = date_episode if strpos(diag_icd10`i´,"H47") > 0
qui replace v107 = date_episode if strpos(diag_icd10`i´,"H48") > 0
qui replace v108 = date_episode if strpos(diag_icd10`i´,"H15") > 0
qui replace v108 = date_episode if strpos(diag_icd10`i´,"H19") > 0
qui replace v108 = date_episode if strpos(diag_icd10`i´,"H27") > 0
qui replace v108 = date_episode if strpos(diag_icd10`i´,"H43") > 0
qui replace v108 = date_episode if strpos(diag_icd10`i´,"H55") > 0
qui replace v108 = date_episode if strpos(diag_icd10`i´,"H57") > 0
qui replace v108 = date_episode if strpos(diag_icd10`i´,"H58") > 0
qui replace v109 = date_episode if strpos(diag_icd10`i´,"H72") > 0
qui replace v109 = date_episode if strpos(diag_icd10`i´,"H73") > 0
qui replace v110 = date_episode if strpos(diag_icd10`i´,"H90") > 0
qui replace v110 = date_episode if strpos(diag_icd10`i´,"H91") > 0
qui replace v111 = date_episode if strpos(diag_icd10`i´,"I10") > 0
qui replace v112 = date_episode if strpos(diag_icd10`i´,"I21") > 0
qui replace v112 = date_episode if strpos(diag_icd10`i´,"I22") > 0
qui replace v113 = date_episode if strpos(diag_icd10`i´,"I24") > 0
qui replace v114 = date_episode if strpos(diag_icd10`i´,"I252") > 0 // changed from "I25.2"
qui replace v115 = date_episode if strpos(diag_icd10`i´,"I20") > 0
qui replace v116 = date_episode if strpos(diag_icd10`i´,"I251") > 0 // changed from "I25.1"
qui replace v116 = date_episode if strpos(diag_icd10`i´,"I253") > 0 // changed from "I25.3"
qui replace v116 = date_episode if strpos(diag_icd10`i´,"I254") > 0 // changed from "I25.4"
qui replace v116 = date_episode if strpos(diag_icd10`i´,"I257") > 0 // changed from "I25.7"
qui replace v116 = date_episode if strpos(diag_icd10`i´,"I258") > 0 // changed from "I25.8"
qui replace v116 = date_episode if strpos(diag_icd10`i´,"I259") > 0 // changed from "I25.9"
qui replace v117 = date_episode if strpos(diag_icd10`i´,"I27") > 0
qui replace v118 = date_episode if strpos(diag_icd10`i´,"I34") > 0
qui replace v118 = date_episode if strpos(diag_icd10`i´,"I35") > 0
qui replace v118 = date_episode if strpos(diag_icd10`i´,"I36") > 0
qui replace v118 = date_episode if strpos(diag_icd10`i´,"I37") > 0
qui replace v118 = date_episode if strpos(diag_icd10`i´,"I38") > 0
qui replace v118 = date_episode if strpos(diag_icd10`i´,"I39") > 0
qui replace v119 = date_episode if strpos(diag_icd10`i´,"I42") > 0
qui replace v119 = date_episode if strpos(diag_icd10`i´,"I43") > 0
qui replace v120 = date_episode if strpos(diag_icd10`i´,"I44") > 0
qui replace v120 = date_episode if strpos(diag_icd10`i´,"I45") > 0
qui replace v121 = date_episode if strpos(diag_icd10`i´,"I46") > 0
qui replace v121 = date_episode if strpos(diag_icd10`i´,"I47") > 0
qui replace v121 = date_episode if strpos(diag_icd10`i´,"I48") > 0
qui replace v121 = date_episode if strpos(diag_icd10`i´,"I49") > 0
qui replace v122 = date_episode if strpos(diag_icd10`i´,"I50") > 0
qui replace v123 = date_episode if strpos(diag_icd10`i´,"I51") > 0
qui replace v123 = date_episode if strpos(diag_icd10`i´,"I52") > 0
qui replace v123 = date_episode if strpos(diag_icd10`i´,"I970") > 0 // changed from "I97.0"
qui replace v123 = date_episode if strpos(diag_icd10`i´,"I971") > 0 // changed from "I97.1"
qui replace v124 = date_episode if strpos(diag_icd10`i´,"I63") > 0
qui replace v124 = date_episode if strpos(diag_icd10`i´,"I65") > 0
qui replace v125 = date_episode if strpos(diag_icd10`i´,"G450") > 0 // changed from "G45.0"
qui replace v125 = date_episode if strpos(diag_icd10`i´,"G451") > 0 // changed from "G45.1"
qui replace v125 = date_episode if strpos(diag_icd10`i´,"G452") > 0 // changed from "G45.2"
qui replace v125 = date_episode if strpos(diag_icd10`i´,"G453") > 0 // changed from "G45.3"
qui replace v125 = date_episode if strpos(diag_icd10`i´,"G458") > 0 // changed from "G45.8"
qui replace v125 = date_episode if strpos(diag_icd10`i´,"G459") > 0 // changed from "G45.9"
qui replace v125 = date_episode if strpos(diag_icd10`i´,"G46") > 0
qui replace v126 = date_episode if strpos(diag_icd10`i´,"NONE") > 0
qui replace v127 = date_episode if strpos(diag_icd10`i´,"G454") > 0 // changed from "G45.4"
qui replace v127 = date_episode if strpos(diag_icd10`i´,"I67") > 0
qui replace v127 = date_episode if strpos(diag_icd10`i´,"I68") > 0
qui replace v128 = date_episode if strpos(diag_icd10`i´,"I69") > 0
```

```
qui replace v129 = date_episode if strpos(diag_icd10`i´,"I70") > 0
qui replace v130 = date_episode if strpos(diag_icd10`i´,"I71") > 0
qui replace v130 = date_episode if strpos(diag_icd10`i´,"I790") > 0 // changed from "I79.0"
qui replace v131 = date_episode if strpos(diag_icd10`i´,"I72") > 0
qui replace v132 = date_episode if strpos(diag_icd10`i´,"I73") > 0
qui replace v132 = date_episode if strpos(diag_icd10`i´,"I777") > 0 // changed from "I77.7"
qui replace v133 = date_episode if strpos(diag_icd10`i´,"I74") > 0
qui replace v134 = date_episode if strpos(diag_icd10`i´,"M30") > 0
qui replace v134 = date_episode if strpos(diag_icd10`i´,"M31") > 0
qui replace v135 = date_episode if strpos(diag_icd10`i´,"I77") > 0
qui replace v136 = date_episode if strpos(diag_icd10`i´,"I80") > 0
qui replace v137 = date_episode if strpos(diag_icd10`i´,"I82") > 0
qui replace v138 = date_episode if strpos(diag_icd10`i´,"I83") > 0
qui replace v139 = date_episode if strpos(diag_icd10`i´,"I84") > 0
qui replace v140 = date_episode if strpos(diag_icd10`i´,"I89") > 0
qui replace v140 = date_episode if strpos(diag_icd10`i´,"I972") > 0 // changed from "I97.2"
qui replace v141 = date_episode if strpos(diag_icd10`i´,"I87") > 0
qui replace v141 = date_episode if strpos(diag_icd10`i´,"I98") > 0
qui replace v141 = date_episode if strpos(diag_icd10`i´,"I99") > 0
qui replace v141 = date_episode if strpos(diag_icd10`i´,"R58") > 0
qui replace v142 = date_episode if strpos(diag_icd10`i´,"J32") > 0
qui replace v143 = date_episode if strpos(diag_icd10`i´,"J35") > 0
qui replace v144 = date_episode if strpos(diag_icd10`i´,"J30") > 0
qui replace v145 = date_episode if strpos(diag_icd10`i´,"J41") > 0
qui replace v145 = date_episode if strpos(diag_icd10`i´,"J42") > 0
qui replace v146 = date_episode if strpos(diag_icd10`i´,"J43") > 0
qui replace v147 = date_episode if strpos(diag_icd10`i´,"J45") > 0
qui replace v147 = date_episode if strpos(diag_icd10`i´,"J46") > 0
qui replace v148 = date_episode if strpos(diag_icd10`i´,"J44") > 0
qui replace v149 = date_episode if strpos(diag_icd10`i´,"J60") > 0
qui replace v150 = date_episode if strpos(diag_icd10`i´,"J80") > 0
qui replace v150 = date_episode if strpos(diag_icd10`i´,"J81") > 0
qui replace v150 = date_episode if strpos(diag_icd10`i´,"J82") > 0
qui replace v150 = date_episode if strpos(diag_icd10`i´,"J96") > 0
qui replace v150 = date_episode if strpos(diag_icd10`i´,"J980") > 0 // changed from "J98.0"
qui replace v150 = date_episode if strpos(diag_icd10`i´,"J981") > 0 // changed from "J98.1"
qui replace v150 = date_episode if strpos(diag_icd10`i´,"J982") > 0 // changed from "J98.2"
qui replace v150 = date_episode if strpos(diag_icd10`i´,"J983") > 0 // changed from "J98.3"
qui replace v150 = date_episode if strpos(diag_icd10`i´,"J984") > 0 // changed from "J98.4"
qui replace v151 = date_episode if strpos(diag_icd10`i´,"J95") > 0
qui replace v151 = date_episode if strpos(diag_icd10`i´,"J985") > 0 // changed from "J98.5"
qui replace v151 = date_episode if strpos(diag_icd10`i´,"J986") > 0 // changed from "J98.6"
qui replace v151 = date_episode if strpos(diag_icd10`i´,"J988") > 0 // changed from "J98.8"
qui replace v151 = date_episode if strpos(diag_icd10`i´,"J989") > 0 // changed from "J98.9"
qui replace v152 = date_episode if strpos(diag_icd10`i´,"K02") > 0
qui replace v152 = date_episode if strpos(diag_icd10`i´,"K03") > 0
qui replace v153 = date_episode if strpos(diag_icd10`i´,"K07") > 0
qui replace v154 = date_episode if strpos(diag_icd10`i´,"K08") > 0
qui replace v155 = date_episode if strpos(diag_icd10`i´,"K20") > 0
qui replace v155 = date_episode if strpos(diag_icd10`i´,"K21") > 0
qui replace v155 = date_episode if strpos(diag_icd10`i´,"K22") > 0
qui replace v155 = date_episode if strpos(diag_icd10`i´,"K23") > 0
qui replace v156 = date_episode if strpos(diag_icd10`i´,"K25") > 0
qui replace v157 = date_episode if strpos(diag_icd10`i´,"K29") > 0
qui replace v158 = date_episode if strpos(diag_icd10`i´,"K30") > 0
qui replace v158 = date_episode if strpos(diag_icd10`i´,"K310") > 0 // changed from "K31.0"
qui replace v158 = date_episode if strpos(diag_icd10`i´,"K3183") > 0 // changed from "K31.83"
qui replace v158 = date_episode if strpos(diag_icd10`i´,"K3184") > 0 // changed from "K31.84"
qui replace v158 = date_episode if strpos(diag_icd10`i´,"K942") > 0 // changed from "K94.2"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K311") > 0 // changed from "K31.1"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K312") > 0 // changed from "K31.2"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K313") > 0 // changed from "K31.3"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K314") > 0 // changed from "K31.4"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K315") > 0 // changed from "K31.5"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K316") > 0 // changed from "K31.6"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K317") > 0 // changed from "K31.7"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K3181") > 0 // changed from "K31.81"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K3182") > 0 // changed from "K31.82"
```

```
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K3189") > 0 // changed from "K31.89"
qui replace v159 = date_episode if strpos(diag_icd10`i´,"K319") > 0 // changed from "K31.9"
qui replace v160 = date_episode if strpos(diag_icd10`i´,"K40") > 0
qui replace v161 = date_episode if strpos(diag_icd10`i´,"K412") > 0 // changed from "K41.2"
qui replace v161 = date_episode if strpos(diag_icd10`i´,"K419") > 0 // changed from "K41.9"
qui replace v161 = date_episode if strpos(diag_icd10`i´,"K429") > 0 // changed from "K42.9"
qui replace v161 = date_episode if strpos(diag_icd10`i´,"K432") > 0 // changed from "K43.2"
qui replace v161 = date_episode if strpos(diag_icd10`i´,"K435") > 0 // changed from "K43.5"
qui replace v161 = date_episode if strpos(diag_icd10`i´,"K439") > 0 // changed from "K43.9"
qui replace v161 = date_episode if strpos(diag_icd10`i´,"K449") > 0 // changed from "K44.9"
qui replace v161 = date_episode if strpos(diag_icd10`i´,"K458") > 0 // changed from "K45.8"
qui replace v161 = date_episode if strpos(diag_icd10`i´,"K469") > 0 // changed from "K46.9"
qui replace v162 = date_episode if strpos(diag_icd10`i´,"K50") > 0
qui replace v163 = date_episode if strpos(diag_icd10`i´,"K51") > 0
qui replace v164 = date_episode if strpos(diag_icd10`i´,"K52") > 0
qui replace v165 = date_episode if strpos(diag_icd10`i´,"K57") > 0
qui replace v166 = date_episode if strpos(diag_icd10`i´,"K58") > 0
qui replace v166 = date_episode if strpos(diag_icd10`i´,"K59") > 0
qui replace v166 = date_episode if strpos(diag_icd10`i´,"K910") > 0 // changed from "K91.0"
qui replace v166 = date_episode if strpos(diag_icd10`i´,"K911") > 0 // changed from "K91.1"
qui replace v167 = date_episode if strpos(diag_icd10`i´,"K62") > 0
qui replace v167 = date_episode if strpos(diag_icd10`i´,"K63") > 0
qui replace v167 = date_episode if strpos(diag_icd10`i´,"K93") > 0
qui replace v167 = date_episode if strpos(diag_icd10`i´,"K940") > 0 // changed from "K94.0"
qui replace v167 = date_episode if strpos(diag_icd10`i´,"K941") > 0 // changed from "K94.1"
qui replace v167 = date_episode if strpos(diag_icd10`i´,"K943") > 0 // changed from "K94.3"
qui replace v168 = date_episode if strpos(diag_icd10`i´,"K70") > 0
qui replace v168 = date_episode if strpos(diag_icd10`i´,"K71") > 0
qui replace v168 = date_episode if strpos(diag_icd10`i´,"K73") > 0
qui replace v168 = date_episode if strpos(diag_icd10`i´,"K74") > 0
qui replace v168 = date_episode if strpos(diag_icd10`i´,"K760") > 0 // changed from "K76.0"
qui replace v169 = date_episode if strpos(diag_icd10`i´,"K761") > 0 // changed from "K76.1"
qui replace v169 = date_episode if strpos(diag_icd10`i´,"K763") > 0 // changed from "K76.3"
qui replace v169 = date_episode if strpos(diag_icd10`i´,"K769") > 0 // changed from "K76.9"
qui replace v169 = date_episode if strpos(diag_icd10`i´,"K77") > 0
qui replace v170 = date_episode if strpos(diag_icd10`i´,"K80") > 0
qui replace v171 = date_episode if strpos(diag_icd10`i´,"K81") > 0
qui replace v171 = date_episode if strpos(diag_icd10`i´,"K82") > 0
qui replace v172 = date_episode if strpos(diag_icd10`i´,"K83") > 0
qui replace v172 = date_episode if strpos(diag_icd10`i´,"K915") > 0 // changed from "K91.5"
qui replace v173 = date_episode if strpos(diag_icd10`i´,"K85") > 0
qui replace v173 = date_episode if strpos(diag_icd10`i´,"K86") > 0
qui replace v174 = date_episode if strpos(diag_icd10`i´,"K90") > 0
qui replace v174 = date_episode if strpos(diag_icd10`i´,"K912") > 0 // changed from "K91.2"
qui replace v175 = date_episode if strpos(diag_icd10`i´,"N05") > 0
qui replace v176 = date_episode if strpos(diag_icd10`i´,"N19") > 0
qui replace v177 = date_episode if strpos(diag_icd10`i´,"N20") > 0
qui replace v178 = date_episode if strpos(diag_icd10`i´,"N11") > 0
qui replace v178 = date_episode if strpos(diag_icd10`i´,"N134") > 0 // changed from "N13.4"
qui replace v178 = date_episode if strpos(diag_icd10`i´,"N135") > 0 // changed from "N13.5"
qui replace v178 = date_episode if strpos(diag_icd10`i´,"N137") > 0 // changed from "N13.7"
qui replace v178 = date_episode if strpos(diag_icd10`i´,"N138") > 0 // changed from "N13.8"
qui replace v178 = date_episode if strpos(diag_icd10`i´,"N139") > 0 // changed from "N13.9"
qui replace v178 = date_episode if strpos(diag_icd10`i´,"N28") > 0
qui replace v178 = date_episode if strpos(diag_icd10`i´,"N29") > 0
qui replace v178 = date_episode if strpos(diag_icd10`i´,"R802") > 0 // changed from "R80.2"
qui replace v179 = date_episode if strpos(diag_icd10`i´,"N31") > 0
qui replace v179 = date_episode if strpos(diag_icd10`i´,"N32") > 0
qui replace v180 = date_episode if strpos(diag_icd10`i´,"N02") > 0
qui replace v180 = date_episode if strpos(diag_icd10`i´,"N36") > 0
qui replace v180 = date_episode if strpos(diag_icd10`i´,"N39") > 0
qui replace v180 = date_episode if strpos(diag_icd10`i´,"R31") > 0
qui replace v181 = date_episode if strpos(diag_icd10`i´,"N40") > 0
qui replace v182 = date_episode if strpos(diag_icd10`i´,"N42") > 0
qui replace v183 = date_episode if strpos(diag_icd10`i´,"N48") > 0
qui replace v184 = date_episode if strpos(diag_icd10`i´,"N60") > 0
qui replace v185 = date_episode if strpos(diag_icd10`i´,"N61") > 0
qui replace v185 = date_episode if strpos(diag_icd10`i´,"N62") > 0
```

```
qui replace v185 = date_episode if strpos(diag_icd10`i´,"N63") > 0
qui replace v185 = date_episode if strpos(diag_icd10`i´,"N64") > 0
qui replace v186 = date_episode if strpos(diag_icd10`i´,"N80") > 0
qui replace v187 = date_episode if strpos(diag_icd10`i´,"N81") > 0
qui replace v187 = date_episode if strpos(diag_icd10`i´,"N993") > 0 // changed from "N99.3"
qui replace v188 = date_episode if strpos(diag_icd10`i´,"N83") > 0
qui replace v189 = date_episode if strpos(diag_icd10`i´,"N840") > 0 // changed from "N84.0"
qui replace v189 = date_episode if strpos(diag_icd10`i´,"N85") > 0
qui replace v190 = date_episode if strpos(diag_icd10`i´,"N841") > 0 // changed from "N84.1"
qui replace v190 = date_episode if strpos(diag_icd10`i´,"N86") > 0
qui replace v190 = date_episode if strpos(diag_icd10`i´,"N87") > 0
qui replace v190 = date_episode if strpos(diag_icd10`i´,"N88") > 0
qui replace v191 = date_episode if strpos(diag_icd10`i´,"N842") > 0 // changed from "N84.2"
qui replace v191 = date_episode if strpos(diag_icd10`i´,"N89") > 0
qui replace v192 = date_episode if strpos(diag_icd10`i´,"N91") > 0
qui replace v192 = date_episode if strpos(diag_icd10`i´,"N920") > 0 // changed from "N92.0"
qui replace v192 = date_episode if strpos(diag_icd10`i´,"N921") > 0 // changed from "N92.1"
qui replace v192 = date_episode if strpos(diag_icd10`i´,"N922") > 0 // changed from "N92.2"
qui replace v192 = date_episode if strpos(diag_icd10`i´,"N923") > 0 // changed from "N92.3"
qui replace v192 = date_episode if strpos(diag_icd10`i´,"N925") > 0 // changed from "N92.5"
qui replace v192 = date_episode if strpos(diag_icd10`i´,"N926") > 0 // changed from "N92.6"
qui replace v192 = date_episode if strpos(diag_icd10`i´,"N93") > 0
qui replace v193 = date_episode if strpos(diag_icd10`i´,"N924") > 0 // changed from "N92.4"
qui replace v193 = date_episode if strpos(diag_icd10`i´,"N95") > 0
qui replace v194 = date_episode if strpos(diag_icd10`i´,"N97") > 0
qui replace v195 = date_episode if strpos(diag_icd10`i´,"L305") > 0 // changed from "L30.5"
qui replace v195 = date_episode if strpos(diag_icd10`i´,"L40") > 0
qui replace v195 = date_episode if strpos(diag_icd10`i´,"L41") > 0
qui replace v195 = date_episode if strpos(diag_icd10`i´,"L42") > 0
qui replace v195 = date_episode if strpos(diag_icd10`i´,"L440") > 0 // changed from "L44.0"
qui replace v196 = date_episode if strpos(diag_icd10`i´,"L28") > 0
qui replace v196 = date_episode if strpos(diag_icd10`i´,"L29") > 0
qui replace v196 = date_episode if strpos(diag_icd10`i´,"L981") > 0 // changed from "L98.1"
qui replace v197 = date_episode if strpos(diag_icd10`i´,"L84") > 0
qui replace v198 = date_episode if strpos(diag_icd10`i´,"L110") > 0 // changed from "L11.0"
qui replace v198 = date_episode if strpos(diag_icd10`i´,"L83") > 0
qui replace v198 = date_episode if strpos(diag_icd10`i´,"L85") > 0
qui replace v198 = date_episode if strpos(diag_icd10`i´,"L86") > 0
qui replace v198 = date_episode if strpos(diag_icd10`i´,"L90") > 0
qui replace v198 = date_episode if strpos(diag_icd10`i´,"L91") > 0
qui replace v198 = date_episode if strpos(diag_icd10`i´,"L92") > 0
qui replace v198 = date_episode if strpos(diag_icd10`i´,"L940") > 0 // changed from "L94.0"
qui replace v199 = date_episode if strpos(diag_icd10`i´,"L60") > 0
qui replace v199 = date_episode if strpos(diag_icd10`i´,"L62") > 0
qui replace v200 = date_episode if strpos(diag_icd10`i´,"L63") > 0
qui replace v200 = date_episode if strpos(diag_icd10`i´,"L64") > 0
qui replace v200 = date_episode if strpos(diag_icd10`i´,"L65") > 0
qui replace v200 = date_episode if strpos(diag_icd10`i´,"L66") > 0
qui replace v200 = date_episode if strpos(diag_icd10`i´,"L67") > 0
qui replace v200 = date_episode if strpos(diag_icd10`i´,"L68") > 0
qui replace v200 = date_episode if strpos(diag_icd10`i´,"L73") > 0
qui replace v201 = date_episode if strpos(diag_icd10`i´,"L89") > 0
qui replace v201 = date_episode if strpos(diag_icd10`i´,"L97") > 0
qui replace v201 = date_episode if strpos(diag_icd10`i´,"L984") > 0 // changed from "L98.4"
qui replace v202 = date_episode if strpos(diag_icd10`i´,"M32") > 0
qui replace v202 = date_episode if strpos(diag_icd10`i´,"M33") > 0
qui replace v202 = date_episode if strpos(diag_icd10`i´,"M34") > 0
qui replace v202 = date_episode if strpos(diag_icd10`i´,"M35") > 0
qui replace v202 = date_episode if strpos(diag_icd10`i´,"M36") > 0
qui replace v203 = date_episode if strpos(diag_icd10`i´,"M05") > 0
qui replace v203 = date_episode if strpos(diag_icd10`i´,"M06") > 0
qui replace v203 = date_episode if strpos(diag_icd10`i´,"M08") > 0
qui replace v203 = date_episode if strpos(diag_icd10`i´,"M120") > 0 // changed from "M12.0"
qui replace v204 = date_episode if strpos(diag_icd10`i´,"M15") > 0
qui replace v204 = date_episode if strpos(diag_icd10`i´,"M16") > 0
qui replace v204 = date_episode if strpos(diag_icd10`i´,"M17") > 0
qui replace v204 = date_episode if strpos(diag_icd10`i´,"M18") > 0
qui replace v204 = date_episode if strpos(diag_icd10`i´,"M19") > 0
```

```
qui replace v205 = date_episode if strpos(diag_icd10`i´,"M121") > 0 // changed from "M12.1"
qui replace v205 = date_episode if strpos(diag_icd10`i´,"M122") > 0 // changed from "M12.2"
qui replace v205 = date_episode if strpos(diag_icd10`i´,"M123") > 0 // changed from "M12.3"
qui replace v205 = date_episode if strpos(diag_icd10`i´,"M124") > 0 // changed from "M12.4"
qui replace v205 = date_episode if strpos(diag_icd10`i´,"M125") > 0 // changed from "M12.5"
qui replace v205 = date_episode if strpos(diag_icd10`i´,"M128") > 0 // changed from "M12.8"
qui replace v205 = date_episode if strpos(diag_icd10`i´,"M129") > 0 // changed from "M12.9"
qui replace v205 = date_episode if strpos(diag_icd10`i´,"M13") > 0
qui replace v206 = date_episode if strpos(diag_icd10`i´,"M22") > 0
qui replace v206 = date_episode if strpos(diag_icd10`i´,"M23") > 0
qui replace v207 = date_episode if strpos(diag_icd10`i´,"M24") > 0
qui replace v208 = date_episode if strpos(diag_icd10`i´,"M25") > 0
qui replace v209 = date_episode if strpos(diag_icd10`i´,"M45") > 0
qui replace v209 = date_episode if strpos(diag_icd10`i´,"M460") > 0 // changed from "M46.0"
qui replace v209 = date_episode if strpos(diag_icd10`i´,"M461") > 0 // changed from "M46.1"
qui replace v209 = date_episode if strpos(diag_icd10`i´,"M468") > 0 // changed from "M46.8"
qui replace v209 = date_episode if strpos(diag_icd10`i´,"M469") > 0 // changed from "M46.9"
qui replace v210 = date_episode if strpos(diag_icd10`i´,"M47") > 0
qui replace v210 = date_episode if strpos(diag_icd10`i´,"M481") > 0 // changed from "M48.1"
qui replace v210 = date_episode if strpos(diag_icd10`i´,"M482") > 0 // changed from "M48.2"
qui replace v210 = date_episode if strpos(diag_icd10`i´,"M483") > 0 // changed from "M48.3"
qui replace v210 = date_episode if strpos(diag_icd10`i´,"M488") > 0 // changed from "M48.8"
qui replace v211 = date_episode if strpos(diag_icd10`i´,"M50") > 0
qui replace v211 = date_episode if strpos(diag_icd10`i´,"M51") > 0
qui replace v211 = date_episode if strpos(diag_icd10`i´,"M961") > 0 // changed from "M96.1"
qui replace v212 = date_episode if strpos(diag_icd10`i´,"M436") > 0 // changed from "M43.6"
qui replace v212 = date_episode if strpos(diag_icd10`i´,"M530") > 0 // changed from "M53.0"
qui replace v212 = date_episode if strpos(diag_icd10`i´,"M531") > 0 // changed from "M53.1"
qui replace v212 = date_episode if strpos(diag_icd10`i´,"M5382") > 0 // changed from "M53.82"
qui replace v212 = date_episode if strpos(diag_icd10`i´,"M5383") > 0 // changed from "M53.83"
qui replace v212 = date_episode if strpos(diag_icd10`i´,"M5402") > 0 // changed from "M54.02"
qui replace v212 = date_episode if strpos(diag_icd10`i´,"M5412") > 0 // changed from "M54.12"
qui replace v212 = date_episode if strpos(diag_icd10`i´,"M5413") > 0 // changed from "M54.13"
qui replace v212 = date_episode if strpos(diag_icd10`i´,"M542") > 0 // changed from "M54.2"
qui replace v213 = date_episode if strpos(diag_icd10`i´,"M480") > 0 // changed from "M48.0"
qui replace v213 = date_episode if strpos(diag_icd10`i´,"M539") > 0 // changed from "M53.9"
qui replace v213 = date_episode if strpos(diag_icd10`i´,"M541") > 0 // changed from "M54.1"
qui replace v213 = date_episode if strpos(diag_icd10`i´,"M543") > 0 // changed from "M54.3"
qui replace v213 = date_episode if strpos(diag_icd10`i´,"M545") > 0 // changed from "M54.5"
qui replace v213 = date_episode if strpos(diag_icd10`i´,"M546") > 0 // changed from "M54.6"
qui replace v213 = date_episode if strpos(diag_icd10`i´,"M548") > 0 // changed from "M54.8"
qui replace v213 = date_episode if strpos(diag_icd10`i´,"M549") > 0 // changed from "M54.9"
qui replace v214 = date_episode if strpos(diag_icd10`i´,"M353") > 0 // changed from "M35.3"
qui replace v215 = date_episode if strpos(diag_icd10`i´,"M70") > 0
qui replace v215 = date_episode if strpos(diag_icd10`i´,"M75") > 0
qui replace v215 = date_episode if strpos(diag_icd10`i´,"M76") > 0
qui replace v215 = date_episode if strpos(diag_icd10`i´,"M77") > 0
qui replace v216 = date_episode if strpos(diag_icd10`i´,"M46.2") > 0 // changed from "M46.2"
qui replace v216 = date_episode if strpos(diag_icd10`i´,"M86") > 0
qui replace v216 = date_episode if strpos(diag_icd10`i´,"M89.6") > 0 // changed from "M89.6"
qui replace v217 = date_episode if strpos(diag_icd10`i´,"M88") > 0
qui replace v217 = date_episode if strpos(diag_icd10`i´,"M894") > 0 // changed from "M89.4"
qui replace v217 = date_episode if strpos(diag_icd10`i´,"M906") > 0 // changed from "M90.6"
qui replace v217 = date_episode if strpos(diag_icd10`i´,"M908") > 0 // changed from "M90.8"
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M80") > 0
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M81") > 0
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M84") > 0
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M85") > 0
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M87") > 0
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M890") > 0 // changed from "M89.0"
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M891") > 0 // changed from "M89.1"
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M899") > 0 // changed from "M89.9"
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M940") > 0 // changed from "M94.0"
qui replace v218 = date_episode if strpos(diag_icd10`i´,"M942") > 0 // changed from "M94.2"
qui replace v219 = date_episode if strpos(diag_icd10`i´,"M201") > 0 // changed from "M20.1"
qui replace v219 = date_episode if strpos(diag_icd10`i´,"M202") > 0 // changed from "M20.2"
qui replace v219 = date_episode if strpos(diag_icd10`i´,"M203") > 0 // changed from "M20.3"
qui replace v219 = date_episode if strpos(diag_icd10`i´,"M204") > 0 // changed from "M20.4"
```

```
qui replace v219 = date_episode if strpos(diag_icd10`i´,"M205") > 0 // changed from "M20.5"
qui replace v219 = date_episode if strpos(diag_icd10`i´,"M206") > 0 // changed from "M20.6"
qui replace v220 = date_episode if strpos(diag_icd10`i´,"M40") > 0
qui replace v220 = date_episode if strpos(diag_icd10`i´,"M41") > 0
qui replace v220 = date_episode if strpos(diag_icd10`i´,"M962") > 0 // changed from "M96.2"
qui replace v220 = date_episode if strpos(diag_icd10`i´,"M963") > 0 // changed from "M96.3"
qui replace v220 = date_episode if strpos(diag_icd10`i´,"M964") > 0 // changed from "M96.4"
qui replace v220 = date_episode if strpos(diag_icd10`i´,"M965") > 0 // changed from "M96.5"
qui replace v221 = date_episode if strpos(diag_icd10`i´,"Q05") > 0
qui replace v221 = date_episode if strpos(diag_icd10`i´,"Q0701") > 0 // changed from "Q07.01"
qui replace v221 = date_episode if strpos(diag_icd10`i´,"Q0703") > 0 // changed from "Q07.03"
qui replace v222 = date_episode if strpos(diag_icd10`i´,"Q22") > 0
qui replace v222 = date_episode if strpos(diag_icd10`i´,"Q23") > 0
qui replace v222 = date_episode if strpos(diag_icd10`i´,"Q24") > 0
qui replace v223 = date_episode if strpos(diag_icd10`i´,"Q25") > 0
qui replace v223 = date_episode if strpos(diag_icd10`i´,"Q26") > 0
qui replace v223 = date_episode if strpos(diag_icd10`i´,"Q27") > 0
qui replace v223 = date_episode if strpos(diag_icd10`i´,"Q28") > 0
qui replace v224 = date_episode if strpos(diag_icd10`i´,"Q60") > 0
qui replace v224 = date_episode if strpos(diag_icd10`i´,"Q61") > 0
qui replace v224 = date_episode if strpos(diag_icd10`i´,"Q62") > 0
qui replace v224 = date_episode if strpos(diag_icd10`i´,"Q63") > 0
qui replace v224 = date_episode if strpos(diag_icd10`i´,"Q64") > 0
qui replace v225 = date_episode if strpos(diag_icd10`i´,"Q65") > 0
qui replace v225 = date_episode if strpos(diag_icd10`i´,"Q66") > 0
qui replace v225 = date_episode if strpos(diag_icd10`i´,"Q67") > 0
qui replace v225 = date_episode if strpos(diag_icd10`i´,"Q68") > 0
qui replace v226 = date_episode if strpos(diag_icd10`i´,"Q75") > 0
qui replace v226 = date_episode if strpos(diag_icd10`i´,"Q76") > 0
qui replace v226 = date_episode if strpos(diag_icd10`i´,"Q77") > 0
qui replace v226 = date_episode if strpos(diag_icd10`i´,"Q78") > 0
qui replace v226 = date_episode if strpos(diag_icd10`i´,"Q79") > 0
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q90") > 0
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q91") > 0
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q92") > 0
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q93") > 0
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q95") > 0
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q96") > 0
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q97") > 0
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q98") > 0
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q990") > 0 // changed from "Q99.0"
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q991") > 0 // changed from "Q99.1"
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q998") > 0 // changed from "Q99.8"
qui replace v227 = date_episode if strpos(diag_icd10`i´,"Q999") > 0 // changed from "Q99.9"
qui replace v228 = date_episode if strpos(diag_icd10`i´,"Q851") > 0 // changed from "Q85.1"
qui replace v228 = date_episode if strpos(diag_icd10`i´,"Q858") > 0 // changed from "Q85.8"
qui replace v228 = date_episode if strpos(diag_icd10`i´,"Q87") > 0
qui replace v228 = date_episode if strpos(diag_icd10`i´,"Q89") > 0
qui replace v228 = date_episode if strpos(diag_icd10`i´,"Q992") > 0 // changed from "Q99.2"
qui replace v229 = date_episode if strpos(diag_icd10`i´,"R4181") > 0 // changed from "R41.81"
qui replace v230 = date_episode if strpos(diag_icd10`i´,"S068") > 0 // changed from "S06.8"
qui replace v230 = date_episode if strpos(diag_icd10`i´,"S069") > 0 // changed from "S06.9"
qui replace v231 = date_episode if substr(diag_icd10`i´,1,1) == "S" & substr(diag_icd10`i´,-1,1) ==
> "S"
qui replace v231 = date_episode if strpos(diag_icd10`i´,"T90") > 0
qui replace v231 = date_episode if strpos(diag_icd10`i´,"T91") > 0
qui replace v231 = date_episode if strpos(diag_icd10`i´,"T92") > 0
qui replace v231 = date_episode if strpos(diag_icd10`i´,"T93") > 0
qui replace v231 = date_episode if strpos(diag_icd10`i´,"T98") > 0
qui replace v232 = date_episode if strpos(diag_icd10`i´,"S77") > 0
qui replace v232 = date_episode if strpos(diag_icd10`i´,"S87") > 0
qui replace v232 = date_episode if strpos(diag_icd10`i´,"S97") > 0
qui replace v232 = date_episode if strpos(diag_icd10`i´,"T043") > 0 // changed from "T04.3"
qui replace v233 = date_episode if strpos(diag_icd10`i´,"S141") > 0 // changed from "S14.1"
qui replace v233 = date_episode if strpos(diag_icd10`i´,"S241") > 0 // changed from "S24.1"
qui replace v233 = date_episode if strpos(diag_icd10`i´,"S341") > 0 // changed from "S34.1"
qui replace v233 = date_episode if strpos(diag_icd10`i´,"S343") > 0 // changed from "S34.3"
qui replace v234 = date_episode if strpos(diag_icd10`i´,"Z21") > 0
```

```
qui replace v235 = date_episode if strpos(diag_icd10`i´,"Z85") > 0
qui replace v236 = date_episode if strpos(diag_icd10`i´,"Z720") > 0 // changed from "Z72.0"
qui replace v236 = date_episode if strpos(diag_icd10`i´,"Z91") > 0
qui replace v237 = date_episode if strpos(diag_icd10`i´,"Z865") > 0 // changed from "Z86.5"
qui replace v238 = date_episode if strpos(diag_icd10`i´,"Z94") > 0
qui replace v239 = date_episode if strpos(diag_icd10`i´,"Z95") > 0
qui replace v239 = date_episode if strpos(diag_icd10`i´,"Z96") > 0
qui replace v239 = date_episode if strpos(diag_icd10`i´,"Z97") > 0
qui replace v240 = date_episode if strpos(diag_icd10`i´,"Z49") > 0


*Label variables
label variable v1 "Human immunodeficiency virus (hiv) disease"
label variable v2 "Sarcoidosis"
label variable v3 "Malignant neoplasm of other and ill-defined sites within the lip oral cavity and
> pharynx"
label variable v4 "Malignant neoplasm of stomach"
label variable v5 "Malignant neoplasm of colon"
label variable v6 "Malignant neoplasm of liver"
label variable v7 "Malignant neoplasm of trachea bronchus and lung"
label variable v8 "Malignant melanoma of skin"
label variable v9 "Other malignant neoplasm of skin"
label variable v10 "Malignant neoplasm of female breast"
label variable v11 "Malignant neoplasm of uterus-part unspecified"
label variable v12 "Malignant neoplasm of cervix uteri"
label variable v13 "Malignant neoplasm of prostate"
label variable v14 "Malignant neoplasm of bladder"
label variable v15 "Malignant neoplasm of kidney and other and unspecified urinary organs"
label variable v16 "Malignant neoplasm of brain"
label variable v17 "Malignant neoplasm of other and ill-defined sites"
label variable v18 "Malignant neoplasm without specification of site"
label variable v19 "Hodgkin´s disease"
label variable v20 "Other malignant neoplasms of lymphoid and histiocytic tissue"
label variable v21 "Leukemia of unspecified cell type"
label variable v22 "Benign neoplasm of other parts of digestive system"
label variable v23 "Other benign neoplasm of connective and other soft tissue"
label variable v24 "Benign neoplasm of skin"
label variable v25 "Benign neoplasm of breast"
label variable v26 "Uterine leiomyoma"
label variable v27 "Benign neoplasm of other and unspecified sites"
label variable v28 "Neoplasm of uncertain behavior of other and unspecified sites and tissues"
label variable v29 "Neoplasms Of Unspecified Nature"
label variable v30 "Simple and unspecified goiter"
label variable v31 "Thyrotoxicosis with or without goiter"
label variable v32 "Acquired hypothyroidism"
label variable v33 "Thyroiditis"
label variable v34 "Other disorders of thyroid"
label variable v35 "Diabetes mellitus"
label variable v36 "Other disorders of pancreatic internal secretion"
label variable v37 "Disorders of parathyroid gland"
label variable v38 "Disorders of the pituitary gland and its hypothalamic control"
label variable v39 "Disorders of adrenal glands"
label variable v40 "Ovarian dysfunction"
label variable v41 "Other endocrine disorders"
label variable v42 "Disorders of carbohydrate transport and metabolism"
label variable v43 "Disorders of lipoid metabolism"
label variable v44 "Gout"
label variable v45 "Disorders of mineral metabolism"
label variable v46 "Other and unspecified disorders of metabolism"
label variable v47 "Overweight, obesity and other hyperalimentation"
label variable v48 "Disorders involving the immune mechanism"
label variable v49 "Iron deficiency anemias"
label variable v50 "Hereditary hemolytic anemias"
label variable v51 "Other and unspecified anemias"
label variable v52 "Coagulation defects"
label variable v53 "Purpura and other hemorrhagic conditions"
label variable v54 "Diseases of white blood cells"
label variable v55 "Other diseases of blood and blood-forming organs"
```

```
label variable v56 "Persistent mental disorders due to conditions classified elsewhere"
label variable v57 "Schizophrenic disorders"
label variable v58 "Episodic mood disorders"
label variable v59 "Delusional disorders"
label variable v60 "Other nonorganic psychoses"
label variable v61 "Anxiety, dissociative and somatoform disorders"
label variable v62 "Personality disorders"
label variable v63 "Sexual and gender identity disorders"
label variable v64 "Alcohol dependence syndrome"
label variable v65 "Drug dependence"
label variable v66 "Nondependent abuse of drugs"
label variable v67 "Physiological malfunction arising from mental factors"
label variable v68 "Special symptoms or syndromes not elsewhere classified"
label variable v69 "Adjustment reaction"
label variable v70 "Specific nonpsychotic mental disorders due to brain damage"
label variable v71 "Depressive disorder not elsewhere classified"
label variable v72 "Disturbance of conduct not elsewhere classified"
label variable v73 "Hyperkinetic syndrome of childhood"
label variable v74 "Specific delays in development"
label variable v75 "Unspecified mental retardation"
label variable v76 "Other cerebral degenerations"
label variable v77 "Parkinson´s disease"
label variable v78 "Other extrapyramidal disease and abnormal movement disorders"
label variable v79 "Other diseases of spinal cord"
label variable v80 "Disorders of the autonomic nervous system"
label variable v81 "Multiple sclerosis"
label variable v82 "Infantile cerebral palsy"
label variable v83 "Other paralytic syndromes"
label variable v84 "Epilepsy"
label variable v85 "Migraine"
label variable v86 "Cataplexy and narcolepsy"
label variable v87 "Other conditions of brain"
label variable v88 "Other and unspecified disorders of the nervous system"
label variable v89 "Facial nerve disorders"
label variable v90 "Nerve root and plexus disorders"
label variable v91 "Mononeuritis of upper limb and mononeuritis multiplex"
label variable v92 "Mononeuritis of lower limb and unspecified site"
label variable v93 "Hereditary and idiopathic peripheral neuropathy"
label variable v94 "Muscular dystrophies and other myopathies"
label variable v95 "Disorders of the globe"
label variable v96 "Retinal detachments and defects"
label variable v97 "Other retinal disorders"
label variable v98 "Glaucoma"
label variable v99 "Cataract"
label variable v100 "Disorders of refraction and accommodation"
label variable v101 "Visual disturbances"
label variable v102 "Blindness and low vision"
label variable v103 "Keratitis"
label variable v104 "Disorders of conjunctiva"
label variable v105 "Other disorders of eyelids"
label variable v106 "Disorders of lacrimal system"
label variable v107 "Disorders of optic nerve and visual pathways"
label variable v108 "Other disorders of eye"
label variable v109 "Other disorders of tympanic membrane"
label variable v110 "Hearing loss"
label variable v111 "Essential hypertension"
label variable v112 "Acute myocardial infarction"
label variable v113 "Other acute and subacute forms of ischemic heart disease"
label variable v114 "Old myocardial infarction"
label variable v115 "Angina pectoris"
label variable v116 "Other forms of chronic ischemic heart disease"
label variable v117 "Chronic pulmonary heart disease"
label variable v118 "Other diseases of endocardium"
label variable v119 "Cardiomyopathy"
label variable v120 "Conduction disorders"
label variable v121 "Cardiac dysrhythmias"
label variable v122 "Heart failure"
label variable v123 "Ill-defined descriptions and complications of heart disease"
```

```
label variable v124 "Occlusion and stenosis of precerebral arteries"
label variable v125 "Transient cerebral ischemia"
label variable v126 "Acute but ill-defined cerebrovascular disease"
label variable v127 "Other and ill-defined cerebrovascular disease"
label variable v128 "Late effects of cerebrovascular disease"
label variable v129 "Atherosclerosis"
label variable v130 "Aortic aneurysm and dissection"
label variable v131 "Other aneurysm"
label variable v132 "Other peripheral vascular disease"
label variable v133 "Arterial embolism and thrombosis"
label variable v134 "Polyarteritis nodosa and allied conditions"
label variable v135 "Other disorders of arteries and arterioles"
label variable v136 "Phlebitis and thrombophlebitis"
label variable v137 "Other venous embolism and thrombosis"
label variable v138 "Varicose veins of lower extremities"
label variable v139 "Hemorrhoids"
label variable v140 "Noninfectious disorders of lymphatic channels"
label variable v141 "Other disorders of circulatory system"
label variable v142 "Chronic sinusitis"
label variable v143 "Chronic disease of tonsils and adenoids"
label variable v144 "Allergic rhinitis"
label variable v145 "Chronic bronchitis"
label variable v146 "Emphysema"
label variable v147 "Asthma"
label variable v148 "Chronic airway obstruction not elsewhere classified"
label variable v149 "Coal workers´ pneumoconiosis"
label variable v150 "Other diseases of lung"
label variable v151 "Other diseases of respiratory system"
label variable v152 "Diseases of hard tissues of teeth"
label variable v153 "Dentofacial anomalies including malocclusion"
label variable v154 "Other diseases and conditions of the teeth and supporting structures"
label variable v155 "Diseases of esophagus"
label variable v156 "Gastric ulcer"
label variable v157 "Gastritis and duodenitis"
label variable v158 "Disorders of function of stomach"
label variable v159 "Other disorders of stomach and duodenum"
label variable v160 "Inguinal hernia"
label variable v161 "Other hernia of abdominal cavity without mention of obstruction or gangrene"
label variable v162 "Regional enteritis"
label variable v163 "Ulcerative enterocolitis"
label variable v164 "Other and unspecified noninfectious gastroenteritis and colitis"
label variable v165 "Diverticula of intestine"
label variable v166 "Functional digestive disorders not elsewhere classified"
label variable v167 "Other disorders of intestine"
label variable v168 "Chronic liver disease and cirrhosis"
label variable v169 "Other disorders of liver"
label variable v170 "Cholelithiasis"
label variable v171 "Other disorders of gallbladder"
label variable v172 "Other disorders of biliary tract"
label variable v173 "Diseases of pancreas"
label variable v174 "Intestinal malabsorption"
label variable v175 "Nephritis and nephropathy not specified as acute or chronic"
label variable v176 "Renal failure unspecified"
label variable v177 "Calculus of kidney and ureter"
label variable v178 "Other disorders of kidney and ureter"
label variable v179 "Other disorders of bladder"
label variable v180 "Other disorders of urethra and urinary tract"
label variable v181 "Hyperplasia of prostate"
label variable v182 "Other disorders of prostate"
label variable v183 "Disorders of penis"
label variable v184 "Benign mammary dysplasias"
label variable v185 "Other disorders of breast"
label variable v186 "Endometriosis"
label variable v187 "Genital prolapse"
label variable v188 "Noninflammatory disorders of ovary fallopian tube and broad ligament"
label variable v189 "Disorders of uterus not elsewhere classified"
label variable v190 "Noninflammatory disorders of cervix"
label variable v191 "Noninflammatory disorders of vagina"
```

```
label variable v192 "Disorders of menstruation and other abnormal bleeding from female genital tract
> "
label variable v193 "Menopausal and postmenopausal disorders"
label variable v194 "Female infertility"
label variable v195 "Psoriasis and similar disorders"
label variable v196 "Pruritus and related conditions"
label variable v197 "Corns and callosities"
label variable v198 "Other hypertrophic and atrophic conditions of skin"
label variable v199 "Diseases of nail"
label variable v200 "Diseases of hair and hair follicles"
label variable v201 "Chronic ulcer of skin"
label variable v202 "Diffuse diseases of connective tissue"
label variable v203 "Rheumatoid arthritis and other inflammatory polyarthropathies"
label variable v204 "Osteoarthrosis and allied disorders"
label variable v205 "Other and unspecified arthropathies"
label variable v206 "Internal derangement of knee"
label variable v207 "Other derangement of joint"
label variable v208 "Other and unspecified disorders of joint"
label variable v209 "Ankylosing spondylitis and other inflammatory spondylopathies"
label variable v210 "Spondylosis and allied disorders"
label variable v211 "Intervertebral disc disorders"
label variable v212 "Other disorders of cervical region"
label variable v213 "Other and unspecified disorders of back"
label variable v214 "Polymyalgia rheumatica"
label variable v215 "Peripheral enthesopathies and allied syndromes"
label variable v216 "Osteomyelitis periostitis and other infections involving bone"
label variable v217 "Osteitis deformans and osteopathies associated with other disorders classified
> elsewhere"
label variable v218 "Other disorders of bone and cartilage"
label variable v219 "Acquired deformities of toe"
label variable v220 "Curvature of spine"
label variable v221 "Spina bifida"
label variable v222 "Other congenital anomalies of heart"
label variable v223 "Other congenital anomalies of circulatory system"
label variable v224 "Congenital anomalies of urinary system"
label variable v225 "Certain congenital musculoskeletal deformities"
label variable v226 "Other congenital musculoskeletal anomalies"
label variable v227 "Chromosomal anomalies"
label variable v228 "Other and unspecified congenital anomalies"
label variable v229 "Senility without psychosis"
label variable v230 "Intracranial injury of other and unspecified nature"
label variable v231 "Late effects of other and unspecified injuries"
label variable v232 "Crushing injury of lower limb"
label variable v233 "Spinal cord injury without evidence of spinal bone injury"
label variable v234 "Asymptomatic human immunodeficiency virus (hiv) infection status"
label variable v235 "Personal history of malignant neoplasm"
label variable v236 "Other personal history presenting hazards to health"
label variable v237 "Mental and behavioral problems"
label variable v238 "Organ or tissue replaced by transplant"
label variable v239 "Organ or tissue replaced by other means"
label variable v240 "Encounter for dialysis and dialysis catheter care"

*drop diag* date_episode
        }

save "$hesin_data\hesin_icd10_complete_`j´.dta", replace

}

*Append the icd10_complete data
 use "$hesin_data\hesin_icd10_complete_1.dta", clear
        forval j=2/39{
                append using "$hesin_data\hesin_icd10_complete_`j´.dta"
        }

        save "$hesin_data\hesin_icd10_complete.dta",replace
```

The necessary **date_episodes** were identified for each of the 240 comorbidities and saved as **hesin_icd10_complete.dta**.

4.1.4.2 Working on ICD-9 diagnosis codes

Next, we followed the same approach to work on the ICD-9 codes as we did for the ICD-10 codes.

```
import delimited "$hesin_data\hesin_diag_main.csv", clear

drop diag_icd9_nb diag_icd10_nb diag_icd10
egen idlong = concat(eid ins_index arr_index level), format(%20.0g) p(",")
drop eid ins_index level
reshape wide diag_icd9, i(idlong) j(arr_index)
split idlong, p(,)
rename idlong1 eid
rename idlong2 ins_index
rename idlong3 arr_index
rename idlong4 level
destring eid, replace
destring ins_index, replace
destring arr_index, replace
destring level, replace
drop idlong
order eid ins_index arr_index level diag*

merge m:1 eid ins_index using "$hesin_data\hesin_main.dta"

drop if _merge == 2
drop _merge*

drop if diag_icd90 == "" & diag_icd91 == "" & diag_icd92 == "" & diag_icd93 == "" & diag_icd94 == ""
>  & diag_icd95 == "" & diag_icd96 == "" & diag_icd97 == "" & diag_icd98 == "" & diag_icd99 == "" &
> diag_icd910 == "" & diag_icd911 == "" & diag_icd912 == "" & diag_icd913 == "" & diag_icd914 == ""
> & diag_icd915 == "" & diag_icd916 == "" & diag_icd917 == "" & diag_icd918 == "" & diag_icd919 == "
> " & diag_icd920 == ""

gen epistart_1 = date(epistart, "YMD")
gen epiend_1 = date(epiend, "YMD")
gen admidate_1 = date(admidate,"YMD")
gen disdate_1 = date(disdate, "YMD")

format epistart_1 epiend_1 admidate_1 disdate_1 %td

drop epistart epiend admidate disdate

rename epistart_1 epistart
rename epiend_1 epiend
rename admidate_1 admidate
rename disdate_1 disdate

replace epistart = admidate if epistart ==.
replace epiend = disdate if epiend ==.

drop admidate disdate

rename eid id_phe
merge m:1 id_phe using "$stata_sbp_input\date_attending.dta", keep(3) nogen
order ts_53_0_0, a(epistart)
rename ts_53_0_0 date_baseline
rename epistart date_episode

*Generate health condition phenotypes
forvalues i = 1/240 {
        gen v`i´ = .
        format v`i´ %td
}

save "$hesin_data\hesin_icd9_dates.dta", replace

use "$hesin_data\hesin_icd9_dates.dta", clear

forvalues i = 0/20 {
dis "Looking through `i´ of 20 diagnosis fields for ICD9 codes"

qui replace v1 = date_episode if substr(diag_icd9`i´,1,3) == "042"
qui replace v2 = date_episode if substr(diag_icd9`i´,1,3) == "135"
qui replace v3 = date_episode if substr(diag_icd9`i´,1,3) == "149"
```

```
qui replace v4 = date_episode if substr(diag_icd9`i´,1,3) == "151"
qui replace v5 = date_episode if substr(diag_icd9`i´,1,3) == "153"
qui replace v6 = date_episode if substr(diag_icd9`i´,1,3) == "155"
qui replace v7 = date_episode if substr(diag_icd9`i´,1,3) == "162"
qui replace v7 = date_episode if substr(diag_icd9`i´,1,3) == "162"
qui replace v8 = date_episode if substr(diag_icd9`i´,1,3) == "172"
qui replace v9 = date_episode if substr(diag_icd9`i´,1,3) == "173"
qui replace v10 = date_episode if substr(diag_icd9`i´,1,3) == "174"
qui replace v11 = date_episode if substr(diag_icd9`i´,1,3) == "179"
qui replace v12 = date_episode if substr(diag_icd9`i´,1,3) == "180"
qui replace v13 = date_episode if substr(diag_icd9`i´,1,3) == "185"
qui replace v14 = date_episode if substr(diag_icd9`i´,1,3) == "188"
qui replace v15 = date_episode if substr(diag_icd9`i´,1,3) == "189"
qui replace v15 = date_episode if substr(diag_icd9`i´,1,3) == "189"
qui replace v15 = date_episode if substr(diag_icd9`i´,1,3) == "189"
qui replace v15 = date_episode if substr(diag_icd9`i´,1,3) == "189"
qui replace v16 = date_episode if substr(diag_icd9`i´,1,3) == "191"
qui replace v17 = date_episode if substr(diag_icd9`i´,1,3) == "195"
qui replace v18 = date_episode if substr(diag_icd9`i´,1,3) == "199"
qui replace v19 = date_episode if substr(diag_icd9`i´,1,3) == "201"
qui replace v20 = date_episode if substr(diag_icd9`i´,1,3) == "202"
qui replace v20 = date_episode if substr(diag_icd9`i´,1,3) == "202"
qui replace v20 = date_episode if substr(diag_icd9`i´,1,3) == "202"
qui replace v20 = date_episode if substr(diag_icd9`i´,1,3) == "202"
qui replace v20 = date_episode if substr(diag_icd9`i´,1,3) == "202"
qui replace v21 = date_episode if substr(diag_icd9`i´,1,3) == "208"
qui replace v22 = date_episode if substr(diag_icd9`i´,1,3) == "211"
qui replace v22 = date_episode if substr(diag_icd9`i´,1,3) == "211"
qui replace v22 = date_episode if substr(diag_icd9`i´,1,3) == "211"
qui replace v23 = date_episode if substr(diag_icd9`i´,1,3) == "215"
qui replace v24 = date_episode if substr(diag_icd9`i´,1,3) == "216"
qui replace v24 = date_episode if substr(diag_icd9`i´,1,3) == "216"
qui replace v25 = date_episode if substr(diag_icd9`i´,1,3) == "217"
qui replace v26 = date_episode if substr(diag_icd9`i´,1,3) == "218"
qui replace v27 = date_episode if substr(diag_icd9`i´,1,3) == "229"
qui replace v28 = date_episode if substr(diag_icd9`i´,1,3) == "238"
qui replace v28 = date_episode if substr(diag_icd9`i´,1,3) == "238"
qui replace v28 = date_episode if substr(diag_icd9`i´,1,3) == "238"
qui replace v28 = date_episode if substr(diag_icd9`i´,1,3) == "238"
qui replace v29 = date_episode if substr(diag_icd9`i´,1,3) == "239"
qui replace v30 = date_episode if substr(diag_icd9`i´,1,3) == "240"
qui replace v30 = date_episode if substr(diag_icd9`i´,1,3) == "240"
qui replace v31 = date_episode if substr(diag_icd9`i´,1,3) == "242"
qui replace v32 = date_episode if substr(diag_icd9`i´,1,3) == "244"
qui replace v32 = date_episode if substr(diag_icd9`i´,1,3) == "244"
qui replace v33 = date_episode if substr(diag_icd9`i´,1,3) == "245"
qui replace v34 = date_episode if substr(diag_icd9`i´,1,3) == "246"
qui replace v35 = date_episode if substr(diag_icd9`i´,1,3) == "250"
qui replace v35 = date_episode if substr(diag_icd9`i´,1,3) == "250"
qui replace v35 = date_episode if substr(diag_icd9`i´,1,3) == "250"
qui replace v35 = date_episode if substr(diag_icd9`i´,1,3) == "250"
qui replace v35 = date_episode if substr(diag_icd9`i´,1,3) == "250"
qui replace v36 = date_episode if substr(diag_icd9`i´,1,3) == "251"
qui replace v37 = date_episode if substr(diag_icd9`i´,1,3) == "252"
qui replace v37 = date_episode if substr(diag_icd9`i´,1,3) == "252"
qui replace v38 = date_episode if substr(diag_icd9`i´,1,3) == "253"
qui replace v38 = date_episode if substr(diag_icd9`i´,1,3) == "253"
qui replace v39 = date_episode if substr(diag_icd9`i´,1,3) == "255"
qui replace v39 = date_episode if substr(diag_icd9`i´,1,3) == "255"
qui replace v39 = date_episode if substr(diag_icd9`i´,1,3) == "255"
qui replace v39 = date_episode if substr(diag_icd9`i´,1,3) == "255"
qui replace v40 = date_episode if substr(diag_icd9`i´,1,3) == "256"
qui replace v41 = date_episode if substr(diag_icd9`i´,1,3) == "259"
qui replace v41 = date_episode if substr(diag_icd9`i´,1,3) == "259"
qui replace v41 = date_episode if substr(diag_icd9`i´,1,3) == "259"
qui replace v42 = date_episode if substr(diag_icd9`i´,1,3) == "271"
qui replace v42 = date_episode if substr(diag_icd9`i´,1,3) == "271"
qui replace v42 = date_episode if substr(diag_icd9`i´,1,3) == "271"
```

```
qui replace v43 = date_episode if substr(diag_icd9`i´,1,3) == "272"
qui replace v43 = date_episode if substr(diag_icd9`i´,1,3) == "272"
qui replace v43 = date_episode if substr(diag_icd9`i´,1,3) == "272"
qui replace v44 = date_episode if substr(diag_icd9`i´,1,3) == "274"
qui replace v44 = date_episode if substr(diag_icd9`i´,1,3) == "274"
qui replace v45 = date_episode if substr(diag_icd9`i´,1,3) == "275"
qui replace v46 = date_episode if substr(diag_icd9`i´,1,3) == "277"
qui replace v46 = date_episode if substr(diag_icd9`i´,1,3) == "277"
qui replace v46 = date_episode if substr(diag_icd9`i´,1,3) == "277"
qui replace v46 = date_episode if substr(diag_icd9`i´,1,3) == "277"
qui replace v46 = date_episode if substr(diag_icd9`i´,1,3) == "277"
qui replace v46 = date_episode if substr(diag_icd9`i´,1,3) == "277"
qui replace v47 = date_episode if substr(diag_icd9`i´,1,3) == "278"
qui replace v47 = date_episode if substr(diag_icd9`i´,1,3) == "278"
qui replace v47 = date_episode if substr(diag_icd9`i´,1,3) == "278"
qui replace v47 = date_episode if substr(diag_icd9`i´,1,3) == "278"
qui replace v48 = date_episode if substr(diag_icd9`i´,1,3) == "279"
qui replace v48 = date_episode if substr(diag_icd9`i´,1,3) == "279"
qui replace v48 = date_episode if substr(diag_icd9`i´,1,3) == "279"
qui replace v48 = date_episode if substr(diag_icd9`i´,1,3) == "279"
qui replace v48 = date_episode if substr(diag_icd9`i´,1,3) == "279"
qui replace v48 = date_episode if substr(diag_icd9`i´,1,3) == "279"
qui replace v49 = date_episode if substr(diag_icd9`i´,1,3) == "280"
qui replace v50 = date_episode if substr(diag_icd9`i´,1,3) == "282"
qui replace v50 = date_episode if substr(diag_icd9`i´,1,3) == "282"
qui replace v50 = date_episode if substr(diag_icd9`i´,1,3) == "282"
qui replace v50 = date_episode if substr(diag_icd9`i´,1,3) == "282"
qui replace v51 = date_episode if substr(diag_icd9`i´,1,3) == "285"
qui replace v51 = date_episode if substr(diag_icd9`i´,1,3) == "285"
qui replace v51 = date_episode if substr(diag_icd9`i´,1,3) == "285"
qui replace v52 = date_episode if substr(diag_icd9`i´,1,3) == "286"
qui replace v52 = date_episode if substr(diag_icd9`i´,1,3) == "286"
qui replace v52 = date_episode if substr(diag_icd9`i´,1,3) == "286"
qui replace v52 = date_episode if substr(diag_icd9`i´,1,3) == "286"
qui replace v53 = date_episode if substr(diag_icd9`i´,1,3) == "287"
qui replace v54 = date_episode if substr(diag_icd9`i´,1,3) == "288"
qui replace v54 = date_episode if substr(diag_icd9`i´,1,3) == "288"
qui replace v54 = date_episode if substr(diag_icd9`i´,1,3) == "288"
qui replace v55 = date_episode if substr(diag_icd9`i´,1,3) == "289"
qui replace v55 = date_episode if substr(diag_icd9`i´,1,3) == "289"
qui replace v55 = date_episode if substr(diag_icd9`i´,1,3) == "289"
qui replace v55 = date_episode if substr(diag_icd9`i´,1,3) == "289"
qui replace v55 = date_episode if substr(diag_icd9`i´,1,3) == "289"
qui replace v55 = date_episode if substr(diag_icd9`i´,1,3) == "289"
qui replace v56 = date_episode if substr(diag_icd9`i´,1,3) == "294"
qui replace v56 = date_episode if substr(diag_icd9`i´,1,3) == "294"
qui replace v56 = date_episode if substr(diag_icd9`i´,1,3) == "294"
qui replace v56 = date_episode if substr(diag_icd9`i´,1,3) == "294"
qui replace v57 = date_episode if substr(diag_icd9`i´,1,3) == "295"
qui replace v57 = date_episode if substr(diag_icd9`i´,1,3) == "295"
qui replace v57 = date_episode if substr(diag_icd9`i´,1,3) == "295"
qui replace v57 = date_episode if substr(diag_icd9`i´,1,3) == "295"
qui replace v58 = date_episode if substr(diag_icd9`i´,1,3) == "296"
qui replace v58 = date_episode if substr(diag_icd9`i´,1,3) == "296"
qui replace v58 = date_episode if substr(diag_icd9`i´,1,3) == "296"
qui replace v58 = date_episode if substr(diag_icd9`i´,1,3) == "296"
qui replace v58 = date_episode if substr(diag_icd9`i´,1,3) == "296"
qui replace v58 = date_episode if substr(diag_icd9`i´,1,3) == "296"
qui replace v58 = date_episode if substr(diag_icd9`i´,1,3) == "296"
qui replace v59 = date_episode if substr(diag_icd9`i´,1,3) == "297"
qui replace v59 = date_episode if substr(diag_icd9`i´,1,3) == "297"
qui replace v60 = date_episode if substr(diag_icd9`i´,1,3) == "298"
qui replace v60 = date_episode if substr(diag_icd9`i´,1,3) == "298"
qui replace v60 = date_episode if substr(diag_icd9`i´,1,3) == "298"
qui replace v61 = date_episode if substr(diag_icd9`i´,1,3) == "300"
qui replace v61 = date_episode if substr(diag_icd9`i´,1,3) == "300"
qui replace v61 = date_episode if substr(diag_icd9`i´,1,3) == "300"
qui replace v61 = date_episode if substr(diag_icd9`i´,1,3) == "300"
```

```
qui replace v61 = date_episode if substr(diag_icd9`i´,1,3) == "300"
qui replace v61 = date_episode if substr(diag_icd9`i´,1,3) == "300"
qui replace v61 = date_episode if substr(diag_icd9`i´,1,3) == "300"
qui replace v62 = date_episode if substr(diag_icd9`i´,1,3) == "301"
qui replace v62 = date_episode if substr(diag_icd9`i´,1,3) == "301"
qui replace v62 = date_episode if substr(diag_icd9`i´,1,3) == "301"
qui replace v62 = date_episode if substr(diag_icd9`i´,1,3) == "301"
qui replace v62 = date_episode if substr(diag_icd9`i´,1,3) == "301"
qui replace v63 = date_episode if substr(diag_icd9`i´,1,3) == "302"
qui replace v63 = date_episode if substr(diag_icd9`i´,1,3) == "302"
qui replace v63 = date_episode if substr(diag_icd9`i´,1,3) == "302"
qui replace v63 = date_episode if substr(diag_icd9`i´,1,3) == "302"
qui replace v64 = date_episode if substr(diag_icd9`i´,1,3) == "303"
qui replace v65 = date_episode if substr(diag_icd9`i´,1,3) == "304"
qui replace v65 = date_episode if substr(diag_icd9`i´,1,3) == "304"
qui replace v65 = date_episode if substr(diag_icd9`i´,1,3) == "304"
qui replace v65 = date_episode if substr(diag_icd9`i´,1,3) == "304"
qui replace v65 = date_episode if substr(diag_icd9`i´,1,3) == "304"
qui replace v65 = date_episode if substr(diag_icd9`i´,1,3) == "304"
qui replace v65 = date_episode if substr(diag_icd9`i´,1,3) == "304"
qui replace v65 = date_episode if substr(diag_icd9`i´,1,3) == "304"
qui replace v65 = date_episode if substr(diag_icd9`i´,1,3) == "304"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v66 = date_episode if substr(diag_icd9`i´,1,3) == "305"
qui replace v67 = date_episode if substr(diag_icd9`i´,1,3) == "306"
qui replace v68 = date_episode if substr(diag_icd9`i´,1,3) == "307"
qui replace v68 = date_episode if substr(diag_icd9`i´,1,3) == "307"
qui replace v68 = date_episode if substr(diag_icd9`i´,1,3) == "307"
qui replace v68 = date_episode if substr(diag_icd9`i´,1,3) == "307"
qui replace v69 = date_episode if substr(diag_icd9`i´,1,3) == "309"
qui replace v69 = date_episode if substr(diag_icd9`i´,1,3) == "309"
qui replace v69 = date_episode if substr(diag_icd9`i´,1,3) == "309"
qui replace v69 = date_episode if substr(diag_icd9`i´,1,3) == "309"
qui replace v69 = date_episode if substr(diag_icd9`i´,1,3) == "309"
qui replace v69 = date_episode if substr(diag_icd9`i´,1,3) == "309"
qui replace v70 = date_episode if substr(diag_icd9`i´,1,3) == "310"
qui replace v71 = date_episode if substr(diag_icd9`i´,1,3) == "311"
qui replace v72 = date_episode if substr(diag_icd9`i´,1,3) == "312"
qui replace v72 = date_episode if substr(diag_icd9`i´,1,3) == "312"
qui replace v72 = date_episode if substr(diag_icd9`i´,1,3) == "312"
qui replace v73 = date_episode if substr(diag_icd9`i´,1,3) == "314"
qui replace v74 = date_episode if substr(diag_icd9`i´,1,3) == "315"
qui replace v74 = date_episode if substr(diag_icd9`i´,1,3) == "315"
qui replace v74 = date_episode if substr(diag_icd9`i´,1,3) == "315"
qui replace v74 = date_episode if substr(diag_icd9`i´,1,3) == "315"
qui replace v74 = date_episode if substr(diag_icd9`i´,1,3) == "315"
qui replace v74 = date_episode if substr(diag_icd9`i´,1,3) == "315"
qui replace v75 = date_episode if substr(diag_icd9`i´,1,3) == "319"
qui replace v75 = date_episode if substr(diag_icd9`i´,1,3) == "319"
qui replace v76 = date_episode if substr(diag_icd9`i´,1,3) == "331"
qui replace v76 = date_episode if substr(diag_icd9`i´,1,3) == "331"
qui replace v76 = date_episode if substr(diag_icd9`i´,1,3) == "331"
qui replace v76 = date_episode if substr(diag_icd9`i´,1,3) == "331"
qui replace v76 = date_episode if substr(diag_icd9`i´,1,3) == "331"
qui replace v77 = date_episode if substr(diag_icd9`i´,1,3) == "332"
qui replace v77 = date_episode if substr(diag_icd9`i´,1,3) == "332"
qui replace v77 = date_episode if substr(diag_icd9`i´,1,3) == "332"
qui replace v78 = date_episode if substr(diag_icd9`i´,1,3) == "333"
qui replace v78 = date_episode if substr(diag_icd9`i´,1,3) == "333"
```

```
qui replace v78 = date_episode if substr(diag_icd9`i´,1,3) == "333"
qui replace v78 = date_episode if substr(diag_icd9`i´,1,3) == "333"
qui replace v78 = date_episode if substr(diag_icd9`i´,1,3) == "333"
qui replace v79 = date_episode if substr(diag_icd9`i´,1,3) == "336"
qui replace v79 = date_episode if substr(diag_icd9`i´,1,3) == "336"
qui replace v80 = date_episode if substr(diag_icd9`i´,1,3) == "337"
qui replace v81 = date_episode if substr(diag_icd9`i´,1,3) == "340"
qui replace v82 = date_episode if substr(diag_icd9`i´,1,3) == "343"
qui replace v83 = date_episode if substr(diag_icd9`i´,1,3) == "344"
qui replace v83 = date_episode if substr(diag_icd9`i´,1,3) == "344"
qui replace v84 = date_episode if substr(diag_icd9`i´,1,3) == "345"
qui replace v84 = date_episode if substr(diag_icd9`i´,1,3) == "345"
qui replace v85 = date_episode if substr(diag_icd9`i´,1,3) == "346"
qui replace v86 = date_episode if substr(diag_icd9`i´,1,3) == "347"
qui replace v87 = date_episode if substr(diag_icd9`i´,1,3) == "348"
qui replace v87 = date_episode if substr(diag_icd9`i´,1,3) == "348"
qui replace v88 = date_episode if substr(diag_icd9`i´,1,3) == "349"
qui replace v88 = date_episode if substr(diag_icd9`i´,1,3) == "349"
qui replace v88 = date_episode if substr(diag_icd9`i´,1,3) == "349"
qui replace v88 = date_episode if substr(diag_icd9`i´,1,3) == "349"
qui replace v89 = date_episode if substr(diag_icd9`i´,1,3) == "351"
qui replace v90 = date_episode if substr(diag_icd9`i´,1,3) == "353"
qui replace v90 = date_episode if substr(diag_icd9`i´,1,3) == "353"
qui replace v91 = date_episode if substr(diag_icd9`i´,1,3) == "354"
qui replace v91 = date_episode if substr(diag_icd9`i´,1,3) == "354"
qui replace v92 = date_episode if substr(diag_icd9`i´,1,3) == "355"
qui replace v92 = date_episode if substr(diag_icd9`i´,1,3) == "355"
qui replace v92 = date_episode if substr(diag_icd9`i´,1,3) == "355"
qui replace v92 = date_episode if substr(diag_icd9`i´,1,3) == "355"
qui replace v92 = date_episode if substr(diag_icd9`i´,1,3) == "355"
qui replace v93 = date_episode if substr(diag_icd9`i´,1,3) == "356"
qui replace v94 = date_episode if substr(diag_icd9`i´,1,3) == "359"
qui replace v94 = date_episode if substr(diag_icd9`i´,1,3) == "359"
qui replace v94 = date_episode if substr(diag_icd9`i´,1,3) == "359"
qui replace v95 = date_episode if substr(diag_icd9`i´,1,3) == "360"
qui replace v95 = date_episode if substr(diag_icd9`i´,1,3) == "360"
qui replace v96 = date_episode if substr(diag_icd9`i´,1,3) == "361"
qui replace v97 = date_episode if substr(diag_icd9`i´,1,3) == "362"
qui replace v97 = date_episode if substr(diag_icd9`i´,1,3) == "362"
qui replace v97 = date_episode if substr(diag_icd9`i´,1,3) == "362"
qui replace v98 = date_episode if substr(diag_icd9`i´,1,3) == "365"
qui replace v98 = date_episode if substr(diag_icd9`i´,1,3) == "365"
qui replace v99 = date_episode if substr(diag_icd9`i´,1,3) == "366"
qui replace v99 = date_episode if substr(diag_icd9`i´,1,3) == "366"
qui replace v99 = date_episode if substr(diag_icd9`i´,1,3) == "366"
qui replace v100 = date_episode if substr(diag_icd9`i´,1,3) == "367"
qui replace v101 = date_episode if substr(diag_icd9`i´,1,3) == "368"
qui replace v102 = date_episode if substr(diag_icd9`i´,1,3) == "369"
qui replace v103 = date_episode if substr(diag_icd9`i´,1,3) == "370"
qui replace v104 = date_episode if substr(diag_icd9`i´,1,3) == "372"
qui replace v104 = date_episode if substr(diag_icd9`i´,1,3) == "372"
qui replace v104 = date_episode if substr(diag_icd9`i´,1,3) == "372"
qui replace v105 = date_episode if substr(diag_icd9`i´,1,3) == "374"
qui replace v106 = date_episode if substr(diag_icd9`i´,1,3) == "375"
qui replace v106 = date_episode if substr(diag_icd9`i´,1,3) == "375"
qui replace v107 = date_episode if substr(diag_icd9`i´,1,3) == "377"
qui replace v107 = date_episode if substr(diag_icd9`i´,1,3) == "377"
qui replace v107 = date_episode if substr(diag_icd9`i´,1,3) == "377"
qui replace v108 = date_episode if substr(diag_icd9`i´,1,3) == "379"
qui replace v108 = date_episode if substr(diag_icd9`i´,1,3) == "379"
qui replace v108 = date_episode if substr(diag_icd9`i´,1,3) == "379"
qui replace v108 = date_episode if substr(diag_icd9`i´,1,3) == "379"
qui replace v108 = date_episode if substr(diag_icd9`i´,1,3) == "379"
qui replace v108 = date_episode if substr(diag_icd9`i´,1,3) == "379"
qui replace v108 = date_episode if substr(diag_icd9`i´,1,3) == "379"
qui replace v109 = date_episode if substr(diag_icd9`i´,1,3) == "384"
qui replace v109 = date_episode if substr(diag_icd9`i´,1,3) == "384"
qui replace v110 = date_episode if substr(diag_icd9`i´,1,3) == "389"
```

```
qui replace v110 = date_episode if substr(diag_icd9`i´,1,3) == "389"
qui replace v111 = date_episode if substr(diag_icd9`i´,1,3) == "401"
qui replace v112 = date_episode if substr(diag_icd9`i´,1,3) == "410"
qui replace v112 = date_episode if substr(diag_icd9`i´,1,3) == "410"
qui replace v113 = date_episode if substr(diag_icd9`i´,1,3) == "411"
qui replace v114 = date_episode if substr(diag_icd9`i´,1,3) == "412"
qui replace v115 = date_episode if substr(diag_icd9`i´,1,3) == "413"
qui replace v116 = date_episode if substr(diag_icd9`i´,1,3) == "414"
qui replace v116 = date_episode if substr(diag_icd9`i´,1,3) == "414"
qui replace v116 = date_episode if substr(diag_icd9`i´,1,3) == "414"
qui replace v116 = date_episode if substr(diag_icd9`i´,1,3) == "414"
qui replace v116 = date_episode if substr(diag_icd9`i´,1,3) == "414"
qui replace v116 = date_episode if substr(diag_icd9`i´,1,3) == "414"
qui replace v117 = date_episode if substr(diag_icd9`i´,1,3) == "416"
qui replace v118 = date_episode if substr(diag_icd9`i´,1,3) == "424"
qui replace v118 = date_episode if substr(diag_icd9`i´,1,3) == "424"
qui replace v118 = date_episode if substr(diag_icd9`i´,1,3) == "424"
qui replace v118 = date_episode if substr(diag_icd9`i´,1,3) == "424"
qui replace v118 = date_episode if substr(diag_icd9`i´,1,3) == "424"
qui replace v118 = date_episode if substr(diag_icd9`i´,1,3) == "424"
qui replace v119 = date_episode if substr(diag_icd9`i´,1,3) == "425"
qui replace v119 = date_episode if substr(diag_icd9`i´,1,3) == "425"
qui replace v120 = date_episode if substr(diag_icd9`i´,1,3) == "426"
qui replace v120 = date_episode if substr(diag_icd9`i´,1,3) == "426"
qui replace v121 = date_episode if substr(diag_icd9`i´,1,3) == "427"
qui replace v121 = date_episode if substr(diag_icd9`i´,1,3) == "427"
qui replace v121 = date_episode if substr(diag_icd9`i´,1,3) == "427"
qui replace v121 = date_episode if substr(diag_icd9`i´,1,3) == "427"
qui replace v122 = date_episode if substr(diag_icd9`i´,1,3) == "428"
qui replace v123 = date_episode if substr(diag_icd9`i´,1,3) == "429"
qui replace v123 = date_episode if substr(diag_icd9`i´,1,3) == "429"
qui replace v123 = date_episode if substr(diag_icd9`i´,1,3) == "429"
qui replace v123 = date_episode if substr(diag_icd9`i´,1,3) == "429"
qui replace v124 = date_episode if substr(diag_icd9`i´,1,3) == "433"
qui replace v124 = date_episode if substr(diag_icd9`i´,1,3) == "433"
qui replace v125 = date_episode if substr(diag_icd9`i´,1,3) == "435"
qui replace v125 = date_episode if substr(diag_icd9`i´,1,3) == "435"
qui replace v125 = date_episode if substr(diag_icd9`i´,1,3) == "435"
qui replace v125 = date_episode if substr(diag_icd9`i´,1,3) == "435"
qui replace v125 = date_episode if substr(diag_icd9`i´,1,3) == "435"
qui replace v125 = date_episode if substr(diag_icd9`i´,1,3) == "435"
qui replace v125 = date_episode if substr(diag_icd9`i´,1,3) == "435"
qui replace v126 = date_episode if substr(diag_icd9`i´,1,3) == "436"
qui replace v127 = date_episode if substr(diag_icd9`i´,1,3) == "437"
qui replace v127 = date_episode if substr(diag_icd9`i´,1,3) == "437"
qui replace v127 = date_episode if substr(diag_icd9`i´,1,3) == "437"
qui replace v128 = date_episode if substr(diag_icd9`i´,1,3) == "438"
qui replace v129 = date_episode if substr(diag_icd9`i´,1,3) == "440"
qui replace v130 = date_episode if substr(diag_icd9`i´,1,3) == "441"
qui replace v130 = date_episode if substr(diag_icd9`i´,1,3) == "441"
qui replace v131 = date_episode if substr(diag_icd9`i´,1,3) == "442"
qui replace v132 = date_episode if substr(diag_icd9`i´,1,3) == "443"
qui replace v132 = date_episode if substr(diag_icd9`i´,1,3) == "443"
qui replace v133 = date_episode if substr(diag_icd9`i´,1,3) == "444"
qui replace v134 = date_episode if substr(diag_icd9`i´,1,3) == "446"
qui replace v134 = date_episode if substr(diag_icd9`i´,1,3) == "446"
qui replace v135 = date_episode if substr(diag_icd9`i´,1,3) == "447"
qui replace v136 = date_episode if substr(diag_icd9`i´,1,3) == "451"
qui replace v137 = date_episode if substr(diag_icd9`i´,1,3) == "453"
qui replace v138 = date_episode if substr(diag_icd9`i´,1,3) == "454"
qui replace v139 = date_episode if substr(diag_icd9`i´,1,3) == "455"
qui replace v140 = date_episode if substr(diag_icd9`i´,1,3) == "457"
qui replace v140 = date_episode if substr(diag_icd9`i´,1,3) == "457"
qui replace v141 = date_episode if substr(diag_icd9`i´,1,3) == "459"
qui replace v141 = date_episode if substr(diag_icd9`i´,1,3) == "459"
qui replace v141 = date_episode if substr(diag_icd9`i´,1,3) == "459"
qui replace v141 = date_episode if substr(diag_icd9`i´,1,3) == "459"
qui replace v142 = date_episode if substr(diag_icd9`i´,1,3) == "473"
```

```
qui replace v143 = date_episode if substr(diag_icd9`i´,1,3) == "474"
qui replace v144 = date_episode if substr(diag_icd9`i´,1,3) == "477"
qui replace v145 = date_episode if substr(diag_icd9`i´,1,3) == "491"
qui replace v145 = date_episode if substr(diag_icd9`i´,1,3) == "491"
qui replace v146 = date_episode if substr(diag_icd9`i´,1,3) == "492"
qui replace v147 = date_episode if substr(diag_icd9`i´,1,3) == "493"
qui replace v147 = date_episode if substr(diag_icd9`i´,1,3) == "493"
qui replace v148 = date_episode if substr(diag_icd9`i´,1,3) == "496"
qui replace v149 = date_episode if substr(diag_icd9`i´,1,3) == "500"
qui replace v150 = date_episode if substr(diag_icd9`i´,1,3) == "518"
qui replace v150 = date_episode if substr(diag_icd9`i´,1,3) == "518"
qui replace v150 = date_episode if substr(diag_icd9`i´,1,3) == "518"
qui replace v150 = date_episode if substr(diag_icd9`i´,1,3) == "518"
qui replace v150 = date_episode if substr(diag_icd9`i´,1,3) == "518"
qui replace v150 = date_episode if substr(diag_icd9`i´,1,3) == "518"
qui replace v150 = date_episode if substr(diag_icd9`i´,1,3) == "518"
qui replace v150 = date_episode if substr(diag_icd9`i´,1,3) == "518"
qui replace v150 = date_episode if substr(diag_icd9`i´,1,3) == "518"
qui replace v151 = date_episode if substr(diag_icd9`i´,1,3) == "519"
qui replace v151 = date_episode if substr(diag_icd9`i´,1,3) == "519"
qui replace v151 = date_episode if substr(diag_icd9`i´,1,3) == "519"
qui replace v151 = date_episode if substr(diag_icd9`i´,1,3) == "519"
qui replace v151 = date_episode if substr(diag_icd9`i´,1,3) == "519"
qui replace v152 = date_episode if substr(diag_icd9`i´,1,3) == "521"
qui replace v152 = date_episode if substr(diag_icd9`i´,1,3) == "521"
qui replace v153 = date_episode if substr(diag_icd9`i´,1,3) == "524"
qui replace v154 = date_episode if substr(diag_icd9`i´,1,3) == "525"
qui replace v155 = date_episode if substr(diag_icd9`i´,1,3) == "530"
qui replace v155 = date_episode if substr(diag_icd9`i´,1,3) == "530"
qui replace v155 = date_episode if substr(diag_icd9`i´,1,3) == "530"
qui replace v155 = date_episode if substr(diag_icd9`i´,1,3) == "530"
qui replace v156 = date_episode if substr(diag_icd9`i´,1,3) == "531"
qui replace v157 = date_episode if substr(diag_icd9`i´,1,3) == "535"
qui replace v158 = date_episode if substr(diag_icd9`i´,1,3) == "536"
qui replace v158 = date_episode if substr(diag_icd9`i´,1,3) == "536"
qui replace v158 = date_episode if substr(diag_icd9`i´,1,3) == "536"
qui replace v158 = date_episode if substr(diag_icd9`i´,1,3) == "536"
qui replace v158 = date_episode if substr(diag_icd9`i´,1,3) == "536"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v159 = date_episode if substr(diag_icd9`i´,1,3) == "537"
qui replace v160 = date_episode if substr(diag_icd9`i´,1,3) == "550"
qui replace v161 = date_episode if substr(diag_icd9`i´,1,3) == "553"
qui replace v161 = date_episode if substr(diag_icd9`i´,1,3) == "553"
qui replace v161 = date_episode if substr(diag_icd9`i´,1,3) == "553"
qui replace v161 = date_episode if substr(diag_icd9`i´,1,3) == "553"
qui replace v161 = date_episode if substr(diag_icd9`i´,1,3) == "553"
qui replace v161 = date_episode if substr(diag_icd9`i´,1,3) == "553"
qui replace v161 = date_episode if substr(diag_icd9`i´,1,3) == "553"
qui replace v161 = date_episode if substr(diag_icd9`i´,1,3) == "553"
qui replace v161 = date_episode if substr(diag_icd9`i´,1,3) == "553"
qui replace v162 = date_episode if substr(diag_icd9`i´,1,3) == "555"
qui replace v163 = date_episode if substr(diag_icd9`i´,1,3) == "556"
qui replace v164 = date_episode if substr(diag_icd9`i´,1,3) == "558"
qui replace v165 = date_episode if substr(diag_icd9`i´,1,3) == "562"
qui replace v166 = date_episode if substr(diag_icd9`i´,1,3) == "564"
qui replace v166 = date_episode if substr(diag_icd9`i´,1,3) == "564"
qui replace v166 = date_episode if substr(diag_icd9`i´,1,3) == "564"
qui replace v166 = date_episode if substr(diag_icd9`i´,1,3) == "564"
qui replace v167 = date_episode if substr(diag_icd9`i´,1,3) == "569"
```

```
qui replace v167 = date_episode if substr(diag_icd9`i´,1,3) == "569"
qui replace v167 = date_episode if substr(diag_icd9`i´,1,3) == "569"
qui replace v167 = date_episode if substr(diag_icd9`i´,1,3) == "569"
qui replace v167 = date_episode if substr(diag_icd9`i´,1,3) == "569"
qui replace v167 = date_episode if substr(diag_icd9`i´,1,3) == "569"
qui replace v168 = date_episode if substr(diag_icd9`i´,1,3) == "571"
qui replace v168 = date_episode if substr(diag_icd9`i´,1,3) == "571"
qui replace v168 = date_episode if substr(diag_icd9`i´,1,3) == "571"
qui replace v168 = date_episode if substr(diag_icd9`i´,1,3) == "571"
qui replace v168 = date_episode if substr(diag_icd9`i´,1,3) == "571"
qui replace v169 = date_episode if substr(diag_icd9`i´,1,3) == "573"
qui replace v169 = date_episode if substr(diag_icd9`i´,1,3) == "573"
qui replace v169 = date_episode if substr(diag_icd9`i´,1,3) == "573"
qui replace v169 = date_episode if substr(diag_icd9`i´,1,3) == "573"
qui replace v170 = date_episode if substr(diag_icd9`i´,1,3) == "574"
qui replace v171 = date_episode if substr(diag_icd9`i´,1,3) == "575"
qui replace v171 = date_episode if substr(diag_icd9`i´,1,3) == "575"
qui replace v172 = date_episode if substr(diag_icd9`i´,1,3) == "576"
qui replace v172 = date_episode if substr(diag_icd9`i´,1,3) == "576"
qui replace v173 = date_episode if substr(diag_icd9`i´,1,3) == "577"
qui replace v173 = date_episode if substr(diag_icd9`i´,1,3) == "577"
qui replace v174 = date_episode if substr(diag_icd9`i´,1,3) == "579"
qui replace v174 = date_episode if substr(diag_icd9`i´,1,3) == "579"
qui replace v175 = date_episode if substr(diag_icd9`i´,1,3) == "583"
qui replace v176 = date_episode if substr(diag_icd9`i´,1,3) == "586"
qui replace v177 = date_episode if substr(diag_icd9`i´,1,3) == "592"
qui replace v178 = date_episode if substr(diag_icd9`i´,1,3) == "593"
qui replace v178 = date_episode if substr(diag_icd9`i´,1,3) == "593"
qui replace v178 = date_episode if substr(diag_icd9`i´,1,3) == "593"
qui replace v178 = date_episode if substr(diag_icd9`i´,1,3) == "593"
qui replace v178 = date_episode if substr(diag_icd9`i´,1,3) == "593"
qui replace v178 = date_episode if substr(diag_icd9`i´,1,3) == "593"
qui replace v178 = date_episode if substr(diag_icd9`i´,1,3) == "593"
qui replace v178 = date_episode if substr(diag_icd9`i´,1,3) == "593"
qui replace v178 = date_episode if substr(diag_icd9`i´,1,3) == "593"
qui replace v179 = date_episode if substr(diag_icd9`i´,1,3) == "596"
qui replace v179 = date_episode if substr(diag_icd9`i´,1,3) == "596"
qui replace v180 = date_episode if substr(diag_icd9`i´,1,3) == "599"
qui replace v180 = date_episode if substr(diag_icd9`i´,1,3) == "599"
qui replace v180 = date_episode if substr(diag_icd9`i´,1,3) == "599"
qui replace v180 = date_episode if substr(diag_icd9`i´,1,3) == "599"
qui replace v181 = date_episode if substr(diag_icd9`i´,1,3) == "600"
qui replace v182 = date_episode if substr(diag_icd9`i´,1,3) == "602"
qui replace v183 = date_episode if substr(diag_icd9`i´,1,3) == "607"
qui replace v184 = date_episode if substr(diag_icd9`i´,1,3) == "610"
qui replace v185 = date_episode if substr(diag_icd9`i´,1,3) == "611"
qui replace v185 = date_episode if substr(diag_icd9`i´,1,3) == "611"
qui replace v185 = date_episode if substr(diag_icd9`i´,1,3) == "611"
qui replace v185 = date_episode if substr(diag_icd9`i´,1,3) == "611"
qui replace v186 = date_episode if substr(diag_icd9`i´,1,3) == "617"
qui replace v187 = date_episode if substr(diag_icd9`i´,1,3) == "618"
qui replace v187 = date_episode if substr(diag_icd9`i´,1,3) == "618"
qui replace v188 = date_episode if substr(diag_icd9`i´,1,3) == "620"
qui replace v189 = date_episode if substr(diag_icd9`i´,1,3) == "621"
qui replace v189 = date_episode if substr(diag_icd9`i´,1,3) == "621"
qui replace v190 = date_episode if substr(diag_icd9`i´,1,3) == "622"
qui replace v190 = date_episode if substr(diag_icd9`i´,1,3) == "622"
qui replace v190 = date_episode if substr(diag_icd9`i´,1,3) == "622"
qui replace v190 = date_episode if substr(diag_icd9`i´,1,3) == "622"
qui replace v191 = date_episode if substr(diag_icd9`i´,1,3) == "623"
qui replace v191 = date_episode if substr(diag_icd9`i´,1,3) == "623"
qui replace v192 = date_episode if substr(diag_icd9`i´,1,3) == "626"
qui replace v192 = date_episode if substr(diag_icd9`i´,1,3) == "626"
qui replace v192 = date_episode if substr(diag_icd9`i´,1,3) == "626"
qui replace v192 = date_episode if substr(diag_icd9`i´,1,3) == "626"
qui replace v192 = date_episode if substr(diag_icd9`i´,1,3) == "626"
qui replace v192 = date_episode if substr(diag_icd9`i´,1,3) == "626"
qui replace v192 = date_episode if substr(diag_icd9`i´,1,3) == "626"
```

```
qui replace v192 = date_episode if substr(diag_icd9`i´,1,3) == "626"
qui replace v193 = date_episode if substr(diag_icd9`i´,1,3) == "627"
qui replace v193 = date_episode if substr(diag_icd9`i´,1,3) == "627"
qui replace v194 = date_episode if substr(diag_icd9`i´,1,3) == "628"
qui replace v195 = date_episode if substr(diag_icd9`i´,1,3) == "696"
qui replace v195 = date_episode if substr(diag_icd9`i´,1,3) == "696"
qui replace v195 = date_episode if substr(diag_icd9`i´,1,3) == "696"
qui replace v195 = date_episode if substr(diag_icd9`i´,1,3) == "696"
qui replace v195 = date_episode if substr(diag_icd9`i´,1,3) == "696"
qui replace v196 = date_episode if substr(diag_icd9`i´,1,3) == "698"
qui replace v196 = date_episode if substr(diag_icd9`i´,1,3) == "698"
qui replace v196 = date_episode if substr(diag_icd9`i´,1,3) == "698"
qui replace v197 = date_episode if substr(diag_icd9`i´,1,3) == "700"
qui replace v198 = date_episode if substr(diag_icd9`i´,1,3) == "701"
qui replace v198 = date_episode if substr(diag_icd9`i´,1,3) == "701"
qui replace v198 = date_episode if substr(diag_icd9`i´,1,3) == "701"
qui replace v198 = date_episode if substr(diag_icd9`i´,1,3) == "701"
qui replace v198 = date_episode if substr(diag_icd9`i´,1,3) == "701"
qui replace v198 = date_episode if substr(diag_icd9`i´,1,3) == "701"
qui replace v198 = date_episode if substr(diag_icd9`i´,1,3) == "701"
qui replace v198 = date_episode if substr(diag_icd9`i´,1,3) == "701"
qui replace v199 = date_episode if substr(diag_icd9`i´,1,3) == "703"
qui replace v199 = date_episode if substr(diag_icd9`i´,1,3) == "703"
qui replace v200 = date_episode if substr(diag_icd9`i´,1,3) == "704"
qui replace v200 = date_episode if substr(diag_icd9`i´,1,3) == "704"
qui replace v200 = date_episode if substr(diag_icd9`i´,1,3) == "704"
qui replace v200 = date_episode if substr(diag_icd9`i´,1,3) == "704"
qui replace v200 = date_episode if substr(diag_icd9`i´,1,3) == "704"
qui replace v200 = date_episode if substr(diag_icd9`i´,1,3) == "704"
qui replace v200 = date_episode if substr(diag_icd9`i´,1,3) == "704"
qui replace v201 = date_episode if substr(diag_icd9`i´,1,3) == "707"
qui replace v201 = date_episode if substr(diag_icd9`i´,1,3) == "707"
qui replace v201 = date_episode if substr(diag_icd9`i´,1,3) == "707"
qui replace v202 = date_episode if substr(diag_icd9`i´,1,3) == "710"
qui replace v202 = date_episode if substr(diag_icd9`i´,1,3) == "710"
qui replace v202 = date_episode if substr(diag_icd9`i´,1,3) == "710"
qui replace v202 = date_episode if substr(diag_icd9`i´,1,3) == "710"
qui replace v202 = date_episode if substr(diag_icd9`i´,1,3) == "710"
qui replace v203 = date_episode if substr(diag_icd9`i´,1,3) == "714"
qui replace v203 = date_episode if substr(diag_icd9`i´,1,3) == "714"
qui replace v203 = date_episode if substr(diag_icd9`i´,1,3) == "714"
qui replace v203 = date_episode if substr(diag_icd9`i´,1,3) == "714"
qui replace v204 = date_episode if substr(diag_icd9`i´,1,3) == "715"
qui replace v204 = date_episode if substr(diag_icd9`i´,1,3) == "715"
qui replace v204 = date_episode if substr(diag_icd9`i´,1,3) == "715"
qui replace v204 = date_episode if substr(diag_icd9`i´,1,3) == "715"
qui replace v204 = date_episode if substr(diag_icd9`i´,1,3) == "715"
qui replace v205 = date_episode if substr(diag_icd9`i´,1,3) == "716"
qui replace v205 = date_episode if substr(diag_icd9`i´,1,3) == "716"
qui replace v205 = date_episode if substr(diag_icd9`i´,1,3) == "716"
qui replace v205 = date_episode if substr(diag_icd9`i´,1,3) == "716"
qui replace v205 = date_episode if substr(diag_icd9`i´,1,3) == "716"
qui replace v205 = date_episode if substr(diag_icd9`i´,1,3) == "716"
qui replace v205 = date_episode if substr(diag_icd9`i´,1,3) == "716"
qui replace v205 = date_episode if substr(diag_icd9`i´,1,3) == "716"
qui replace v206 = date_episode if substr(diag_icd9`i´,1,3) == "717"
qui replace v206 = date_episode if substr(diag_icd9`i´,1,3) == "717"
qui replace v207 = date_episode if substr(diag_icd9`i´,1,3) == "718"
qui replace v208 = date_episode if substr(diag_icd9`i´,1,3) == "719"
qui replace v209 = date_episode if substr(diag_icd9`i´,1,3) == "720"
qui replace v209 = date_episode if substr(diag_icd9`i´,1,3) == "720"
qui replace v209 = date_episode if substr(diag_icd9`i´,1,3) == "720"
qui replace v209 = date_episode if substr(diag_icd9`i´,1,3) == "720"
qui replace v209 = date_episode if substr(diag_icd9`i´,1,3) == "720"
qui replace v210 = date_episode if substr(diag_icd9`i´,1,3) == "721"
qui replace v210 = date_episode if substr(diag_icd9`i´,1,3) == "721"
qui replace v210 = date_episode if substr(diag_icd9`i´,1,3) == "721"
qui replace v210 = date_episode if substr(diag_icd9`i´,1,3) == "721"
```

```
qui replace v210 = date_episode if substr(diag_icd9`i´,1,3) == "721"
qui replace v211 = date_episode if substr(diag_icd9`i´,1,3) == "722"
qui replace v211 = date_episode if substr(diag_icd9`i´,1,3) == "722"
qui replace v211 = date_episode if substr(diag_icd9`i´,1,3) == "722"
qui replace v212 = date_episode if substr(diag_icd9`i´,1,3) == "723"
qui replace v212 = date_episode if substr(diag_icd9`i´,1,3) == "723"
qui replace v212 = date_episode if substr(diag_icd9`i´,1,3) == "723"
qui replace v212 = date_episode if substr(diag_icd9`i´,1,3) == "723"
qui replace v212 = date_episode if substr(diag_icd9`i´,1,3) == "723"
qui replace v212 = date_episode if substr(diag_icd9`i´,1,3) == "723"
qui replace v212 = date_episode if substr(diag_icd9`i´,1,3) == "723"
qui replace v212 = date_episode if substr(diag_icd9`i´,1,3) == "723"
qui replace v212 = date_episode if substr(diag_icd9`i´,1,3) == "723"
qui replace v213 = date_episode if substr(diag_icd9`i´,1,3) == "724"
qui replace v213 = date_episode if substr(diag_icd9`i´,1,3) == "724"
qui replace v213 = date_episode if substr(diag_icd9`i´,1,3) == "724"
qui replace v213 = date_episode if substr(diag_icd9`i´,1,3) == "724"
qui replace v213 = date_episode if substr(diag_icd9`i´,1,3) == "724"
qui replace v213 = date_episode if substr(diag_icd9`i´,1,3) == "724"
qui replace v213 = date_episode if substr(diag_icd9`i´,1,3) == "724"
qui replace v213 = date_episode if substr(diag_icd9`i´,1,3) == "724"
qui replace v214 = date_episode if substr(diag_icd9`i´,1,3) == "725"
qui replace v215 = date_episode if substr(diag_icd9`i´,1,3) == "726"
qui replace v215 = date_episode if substr(diag_icd9`i´,1,3) == "726"
qui replace v215 = date_episode if substr(diag_icd9`i´,1,3) == "726"
qui replace v215 = date_episode if substr(diag_icd9`i´,1,3) == "726"
qui replace v216 = date_episode if substr(diag_icd9`i´,1,3) == "730"
qui replace v216 = date_episode if substr(diag_icd9`i´,1,3) == "730"
qui replace v216 = date_episode if substr(diag_icd9`i´,1,3) == "730"
qui replace v217 = date_episode if substr(diag_icd9`i´,1,3) == "731"
qui replace v217 = date_episode if substr(diag_icd9`i´,1,3) == "731"
qui replace v217 = date_episode if substr(diag_icd9`i´,1,3) == "731"
qui replace v217 = date_episode if substr(diag_icd9`i´,1,3) == "731"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v218 = date_episode if substr(diag_icd9`i´,1,3) == "733"
qui replace v219 = date_episode if substr(diag_icd9`i´,1,3) == "735"
qui replace v219 = date_episode if substr(diag_icd9`i´,1,3) == "735"
qui replace v219 = date_episode if substr(diag_icd9`i´,1,3) == "735"
qui replace v219 = date_episode if substr(diag_icd9`i´,1,3) == "735"
qui replace v219 = date_episode if substr(diag_icd9`i´,1,3) == "735"
qui replace v219 = date_episode if substr(diag_icd9`i´,1,3) == "735"
qui replace v220 = date_episode if substr(diag_icd9`i´,1,3) == "737"
qui replace v220 = date_episode if substr(diag_icd9`i´,1,3) == "737"
qui replace v220 = date_episode if substr(diag_icd9`i´,1,3) == "737"
qui replace v220 = date_episode if substr(diag_icd9`i´,1,3) == "737"
qui replace v220 = date_episode if substr(diag_icd9`i´,1,3) == "737"
qui replace v220 = date_episode if substr(diag_icd9`i´,1,3) == "737"
qui replace v221 = date_episode if substr(diag_icd9`i´,1,3) == "741"
qui replace v221 = date_episode if substr(diag_icd9`i´,1,3) == "741"
qui replace v221 = date_episode if substr(diag_icd9`i´,1,3) == "741"
qui replace v222 = date_episode if substr(diag_icd9`i´,1,3) == "746"
qui replace v222 = date_episode if substr(diag_icd9`i´,1,3) == "746"
qui replace v222 = date_episode if substr(diag_icd9`i´,1,3) == "746"
qui replace v223 = date_episode if substr(diag_icd9`i´,1,3) == "747"
qui replace v223 = date_episode if substr(diag_icd9`i´,1,3) == "747"
qui replace v223 = date_episode if substr(diag_icd9`i´,1,3) == "747"
qui replace v223 = date_episode if substr(diag_icd9`i´,1,3) == "747"
qui replace v224 = date_episode if substr(diag_icd9`i´,1,3) == "753"
qui replace v224 = date_episode if substr(diag_icd9`i´,1,3) == "753"
qui replace v224 = date_episode if substr(diag_icd9`i´,1,3) == "753"
```

```
qui replace v224 = date_episode if substr(diag_icd9`i´,1,3) == "753"
qui replace v224 = date_episode if substr(diag_icd9`i´,1,3) == "753"
qui replace v225 = date_episode if substr(diag_icd9`i´,1,3) == "754"
qui replace v225 = date_episode if substr(diag_icd9`i´,1,3) == "754"
qui replace v225 = date_episode if substr(diag_icd9`i´,1,3) == "754"
qui replace v225 = date_episode if substr(diag_icd9`i´,1,3) == "754"
qui replace v226 = date_episode if substr(diag_icd9`i´,1,3) == "756"
qui replace v226 = date_episode if substr(diag_icd9`i´,1,3) == "756"
qui replace v226 = date_episode if substr(diag_icd9`i´,1,3) == "756"
qui replace v226 = date_episode if substr(diag_icd9`i´,1,3) == "756"
qui replace v226 = date_episode if substr(diag_icd9`i´,1,3) == "756"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v227 = date_episode if substr(diag_icd9`i´,1,3) == "758"
qui replace v228 = date_episode if substr(diag_icd9`i´,1,3) == "759"
qui replace v228 = date_episode if substr(diag_icd9`i´,1,3) == "759"
qui replace v228 = date_episode if substr(diag_icd9`i´,1,3) == "759"
qui replace v228 = date_episode if substr(diag_icd9`i´,1,3) == "759"
qui replace v228 = date_episode if substr(diag_icd9`i´,1,3) == "759"
qui replace v229 = date_episode if substr(diag_icd9`i´,1,3) == "797"
qui replace v230 = date_episode if substr(diag_icd9`i´,1,3) == "854"
qui replace v230 = date_episode if substr(diag_icd9`i´,1,3) == "854"
qui replace v231 = date_episode if substr(diag_icd9`i´,1,3) == "908"
qui replace v231 = date_episode if substr(diag_icd9`i´,1,3) == "908"
qui replace v231 = date_episode if substr(diag_icd9`i´,1,3) == "908"
qui replace v231 = date_episode if substr(diag_icd9`i´,1,3) == "908"
qui replace v231 = date_episode if substr(diag_icd9`i´,1,3) == "908"
qui replace v231 = date_episode if substr(diag_icd9`i´,1,3) == "908"
qui replace v232 = date_episode if substr(diag_icd9`i´,1,3) == "928"
qui replace v232 = date_episode if substr(diag_icd9`i´,1,3) == "928"
qui replace v232 = date_episode if substr(diag_icd9`i´,1,3) == "928"
qui replace v232 = date_episode if substr(diag_icd9`i´,1,3) == "928"
qui replace v233 = date_episode if substr(diag_icd9`i´,1,3) == "952"
qui replace v233 = date_episode if substr(diag_icd9`i´,1,3) == "952"
qui replace v233 = date_episode if substr(diag_icd9`i´,1,3) == "952"
qui replace v233 = date_episode if substr(diag_icd9`i´,1,3) == "952"
qui replace v234 = date_episode if substr(diag_icd9`i´,1,3) == "V08"
qui replace v235 = date_episode if substr(diag_icd9`i´,1,3) == "V10"
qui replace v236 = date_episode if substr(diag_icd9`i´,1,3) == "V15"
qui replace v236 = date_episode if substr(diag_icd9`i´,1,3) == "V15"
qui replace v237 = date_episode if substr(diag_icd9`i´,1,3) == "V40"
qui replace v238 = date_episode if substr(diag_icd9`i´,1,3) == "V42"
qui replace v239 = date_episode if substr(diag_icd9`i´,1,3) == "V43"
qui replace v239 = date_episode if substr(diag_icd9`i´,1,3) == "V43"
qui replace v239 = date_episode if substr(diag_icd9`i´,1,3) == "V43"
qui replace v240 = date_episode if substr(diag_icd9`i´,1,3) == "V56"
}
*drop diag* date_episode
save "$hesin_data\hesin_icd9_complete.dta", replace
```

The necessary **date_episodes** were added to the relevant ICD-9 codes for the 240 comorbidities and saved as **hesin_icd9_complete.dta**.

Finally, **hesin_icd10_complete.dta** and **hesin_icd9_complete.dta** datasets were merged into **hesin_icd_complete.dta**.

```
use "$hesin_data\hesin_icd10_complete.dta", clear
```

```
        append using "$hesin_data\hesin_icd9_complete.dta"

        *Replace all v`i´ values with the minimum, so when duplicates are dropped, any instance of t
> he ICD code is kept
forvalues i = 1/240 {
        dis "Sorting through `i´ of 240 variables"
        qui {
                bysort id_phe: egen x = min(v`i´)
        }
        qui replace v`i´ = x
        qui drop x
}
*Keep only a single row per participant
 duplicates drop id_phe, force

 save "$hesin_data\hesin_icd_complete.dta",replace
```

4.1.5 Setting up the data for HRQoL prediction

We compiled the phenotype data, including both ICD-9 and ICD-10 diagnosis dates. The next step was to prepare the data for HRQoL prediction based on the types of comorbidities that a participant might have developed.

The next stata codes will execute:

- Merge the **part_1.dta** dataset with **hesin_icd_compelet.dta** dataset
- Generate variables labeled "z" to make sure if a study participant have developed a condition before the utility day.
- Generate variables labeled "ncc" to count the number of comorbidities a particiapant may have
- Rename some variables
- Reclassify the "qualification" variable for the purpose of analysis

```
use "$stata_sbp_output\part_1.dta", clear
merge 1:1 id_phe using "$hesin_data\hesin_icd_complete.dta", nogen keep(1 3)
*Generate dummy variables for each condition
*These will be 1 if the participant had the condition before the utility day
forvalues i = 1/240 {
        gen z`i´ = 0
}
*Generate dummy variables for number of comorbidities
forvalues i = 2/10 {
        gen ncc`i´ = 0
}
*Rename some variables
rename n_31_0_0 sex
rename n_54_0_0 centre
rename n_189_0_0 eco_tdi
rename n_738_0_0 eco_household_income
rename n_6138_0_0 eco_qualifications
rename n_20160_0_0 ever_smoked
rename n_21001_0_0 phe_bmi
rename n_21003_0_0 age
forvalues i = 1/40 {
        rename n_22009_0_`i´ pc`i´
}
replace date_baseline = ts_53_0_0 if date_baseline == .
```

```
drop ts_53_0_0
rename ts_40000_0_0 date_death
*Make qualifications into dummy variables
forvalues i = 1/3 {
        gen qual_`i´ = 0
}
*Assume everyone who prefered not to answer and missing responses have no qualifications
*High school = A levels, O levels, CSEs & GCSEs
replace qual_1 = 1 if eco_qualifications == 2 | eco_qualifications == 3 | eco_qualifications == 4
*Other degree = NVQs and other
replace qual_2 = 1 if eco_qualifications == 5 | eco_qualifications == 6
*Bachelor degree = College or university degree
replace qual_3 = 1 if eco_qualifications == 1
*Not coding ma_phd
*Remove all participants who died before entry
qui drop if date_death < date_baseline
save "$stata_sbp_output\part_2a.dta", replace
```

Dividing the part_2a.dta Dataset into Small Blocks

The **part_2a.dta** dataset was large, making utility prediction slow. To expedite the analysis, the following steps were taken:

First, the data were divided into 12 blocks, each containing 25,000 observations. Then, the utility was predicted for each block.

4.1.5.1 Predicting Utility

The initial plan was to predict the daily utilities for the participants. However, running the *for loop* for daily predictions would have taken a significant amount of time. Therefore, a monthly utility prediction was chosen by converting the dates into month values. The following steps were taken for the prediction:

- Step 1: Changed variables with date values to monthly values.
- Step 2: Determined the length of the follow-up window for total QALYs (denoted as `fu`).
- Step 3: Created a *for loop* to predict utility using coefficients sourced from Sullivan et al.[7] for the covariates adjusted for prediction and comorbidities.
- Step 4: Predicted the utilities for the participants, considering:
  - The 240 conditions - the main prediction model
  - The four major conditions: cancer, cardiovascular disease, cerebrovascular disease, and diabetes

N.B. Since the Stata code was long and somewhat complicated, comments were added throughout to improve the interpretability of the code.

N.B. Since the Stata command used a *local macro*, the entire command was run at once.

```
use "$stata_sbp_output\part_2a.dta", clear
foreach j of varlist date* v* epiend {
        replace `j´ = mofd(`j´)
        format `j´ %tm
}
```

```
save "$stata_sbp_output\part_2a_months.dta", replace
    local j = 25000
    local k = 1
forval i = 1/12 {

        di `i´ " " `k´ " " `j´
        use part_2a_months, clear

        keep if _n >= `k´ & _n<= `j´
    save "$stata_sbp_output\part_2a_months_`i´.dta", replace

        local k = `j´+1
        local j = `j´+25000

        }

*Let´s look at the maximum follow-up dates by data source (HES, SMR, and PEDW)
use "$stata_sbp_output\part_2a_months.dta", clear
bysort dsource: su date_episode
*The maximum follow up dates for hospitalization were:
        * HES: upto 31 Oct 2022 (753 months)
        * SMR: upto 31 Oct 2022 (753 months)
        * PEDW: upto 26 May 2022 (748 months)
        * For participant without hospital reported data follow-up assumed until 31 Oct 2022 (753 mo
> nths)
*******************************************************************************
*Let´s work on follow-up reported by HES, SMR or no hospital data reported
*******************************************************************************
forval x = 1/12 {

 use "$stata_sbp_output\part_2a_months_`x´.dta", clear

 keep if dsource != "PEDW"

*Find out how long the follow-up window should be for total QALYs (`fu´)
*This is the longest amount of follow-up, i.e. the first entry to 31/10/2022
su date_baseline
qui su date_baseline
local min = r(min)
local fu = 753 -`min´
gen months = 0
forvalues i = 0/`fu´ {
        *Utility variables are the estimated utility for each day after baseline (1 is the end of ba
> seline). For convenience we converted each date to monthly format
        *Start the utility as baseline, adjusted for age
        gen utility_`i´ = 0.9630907 + ((age+0.5+`i´/12)*-0.0002747) + (sex*0.0010046) /// constant +
>  age (half a year added to baseline age, plus days after baseline/365.25) + sex
        + 0.0396568 /// constant added for income (mid income), as this isn´t mappable
        + (qual_1*0.0028418) + (qual_2*0.0056836) + (qual_3*0.0060444) // qualifications

        *Update the dummy variables to 1 if the participant had the condition on or before the utili
> ty day
        forvalues v = 1/240 {
                qui replace z`v´ = 1 if v`v´ <= (date_baseline + `i´)
        }

        *Generate d4_utility just using the 4 main conditions (stroke, heart disease, cancer, type 2
>  diabetes)
        *Cancer = v3-v21, diabetes = v35, cardiovascular disease = v112-v123, stroke = v124-v128
        {
        gen d4_utility_`i´ = utility_`i´
        qui replace d4_utility_`i´ = d4_utility_`i´ ///
        + z3*-0.0278367 /// Cancer
        + z4*-0.0705565 ///
        + z5*-0.0673908 ///
        + z6*-0.0929452 ///
```

```
        + z7*-0.1192427 ///
        + z8*-0.0020176 ///
        + z9*-0.0134124 ///
        + z10*-0.0194279 ///
        + z11*-0.1132353 ///
        + z12*-0.0513159 ///
        + z13*-0.049392 ///
        + z14*-0.056553 ///
        + z15*-0.0479982 ///
        + z16*-0.0414255 ///
        + z17*-0.0858906 ///
        + z18*-0.0335121 ///
        + z19*-0.0584308 ///
        + z20*-0.0099273 ///
        + z21*-0.0500121 ///
        + z35*-0.0714349 /// Diabetes
        + z113*-0.0866826 /// Cardiovascular disease
        + z114*-0.0367975 ///
        + z115*-0.0854255 ///
        + z116*-0.0626527 ///
        + z117*0.0387891 ///
        + z118*-0.0288829 ///
        + z119*-0.1556403 ///
        + z120*-0.0866662 ///
        + z121*-0.0383929 ///
        + z122*-0.1166656 ///
        + z123*-0.0867575 ///
        + z124*-0.0348978 /// Cerebrovascular disease
        + z125*-0.0330382 ///
        + z126*-0.1170501 ///
        + z127*-0.0310476 ///
        + z128*-0.0731964

        *Account for number of comorbidities
        egen ncc = rowtotal(z3-z21 z35 z112-z123 z124-z128)
        forvalues j = 2/10 {
                qui replace ncc`j´ = 0 // reset the ncc counters to 0 each time, so if switching fro
> m 2 to 3 comorbidities, you don´t double count the nccs
                qui replace ncc`j´ = 1 if ncc == `j´
        }
        *ncc10 if 10 or more, so account for that
        qui replace ncc10 = 1 if ncc > 10 & ncc < .
        drop ncc

        *Update the utility variable for number of comorbidities
        qui replace d4_utility_`i´ = d4_utility_`i´+(ncc2*-0.0528484) ///
        +(ncc3*-0.0415352) ///
        +(ncc4*-0.0202969) ///
        +(ncc5*0.0083033) ///
        +(ncc6*0.0408673) ///
        +(ncc7*0.0668729) ///
        +(ncc8*0.1158895) ///
        +(ncc9*0.1344392) ///
        +(ncc10*0.183614)

        *Account for people who have died: set utility to 0 if death is on or before the utility day
        qui replace d4_utility_`i´ = 0 if date_death <= (date_baseline + `i´)

        *Account for utilities after the end date
        qui replace d4_utility_`i´ = . if date_baseline + `i´ > 753

        *End of d4_utility code
        }

        *Account for number of comorbidities
        egen ncc = rowtotal(z*)
        forvalues j = 2/10 {
                qui replace ncc`j´ = 0 // reset the ncc counters to 0 each time, so if switching fro
```

```
> m 2 to 3 comorbidities, you don´t double count the nccs
                qui replace ncc`j´ = 1 if ncc == `j´
        }
        *ncc10 if 10 or more, so account for that
        qui replace ncc10 = 1 if ncc > 10 & ncc < .
        drop ncc

        *Update the utility variable for number of comorbidities
        qui replace utility_`i´ = utility_`i´+(ncc2*-0.0528484) ///
        +(ncc3*-0.0415352) ///
        +(ncc4*-0.0202969) ///
        +(ncc5*0.0083033) ///
        +(ncc6*0.0408673) ///
        +(ncc7*0.0668729) ///
        +(ncc8*0.1158895) ///
        +(ncc9*0.1344392) ///
        +(ncc10*0.183614)

        *Main equation code
        {
        qui replace utility_`i´ = utility_`i´ ///
        + z1*-0.0844983 ///
        + z2*-0.0957076 ///
        + z3*-0.0278367 ///
        + z4*-0.0705565 ///
        + z5*-0.0673908 ///
        + z6*-0.0929452 ///
        + z7*-0.1192427 ///
        + z8*-0.0020176 ///
        + z9*-0.0134124 ///
        + z10*-0.0194279 ///
        + z11*-0.1132353 ///
        + z12*-0.0513159 ///
        + z13*-0.049392 ///
        + z14*-0.056553 ///
        + z15*-0.0479982 ///
        + z16*-0.0414255 ///
        + z17*-0.0858906 ///
        + z18*-0.0335121 ///
        + z19*-0.0584308 ///
        + z20*-0.0099273 ///
        + z21*-0.0500121 ///
        + z22*0.0035188 ///
        + z23*-0.0427221 ///
        + z24*-0.0005495 ///
        + z25*-0.0012877 ///
        + z26*0.0025671 ///
        + z27*-0.0457617 ///
        + z28*0.0021036 ///
        + z29*-0.0356286 ///
        + z30*-0.0320323 ///
        + z31*-0.0345585 ///
        + z32*-0.0064666 ///
        + z33*0.0048511 ///
        + z34*-0.0471758 ///
        + z35*-0.0714349 ///
        + z36*-0.0041212 ///
        + z37*-0.1202905 ///
        + z38*-0.043536 ///
        + z39*-0.1555965 ///
        + z40*0.0251649 ///
        + z41*-0.0302843 ///
        + z42*0.0374719 ///
        + z43*-0.0065353 ///
        + z44*-0.0400731 ///
        + z45*-0.0094726 ///
        + z46*-0.0299147 ///
        + z47*-0.0708597 ///
```

```
+ z48*-0.0807239 ///
+ z49*-0.035499 ///
+ z50*-0.1858287 ///
+ z51*-0.0218789 ///
+ z52*-0.1290684 ///
+ z53*-0.007524 ///
+ z54*0.001099 ///
+ z55*-0.0606417 ///
+ z56*-0.0679451 ///
+ z57*-0.1129732 ///
+ z58*-0.1269103 ///
+ z59*-0.3272071 ///
+ z60*-0.0860614 ///
+ z61*-0.093641 ///
+ z62*-0.087199 ///
+ z63*0.0075985 ///
+ z64*-0.0314791 ///
+ z65*-0.0514317 ///
+ z66*-0.0284652 ///
+ z67*-0.0537585 ///
+ z68*-0.0367998 ///
+ z69*-0.0634499 ///
+ z70*-0.1445047 ///
+ z71*-0.1123433 ///
+ z72*-0.1196535 ///
+ z73*-0.0020176 ///
+ z74*-0.025434 ///
+ z75*-0.2098308 ///
+ z76*-0.2165659 ///
+ z77*-0.227641 ///
+ z78*-0.0968456 ///
+ z79*-0.1496606 ///
+ z80*-0.3915088 ///
+ z81*-0.2114183 ///
+ z82*-0.2459921 ///
+ z83*-0.5262339 ///
+ z84*-0.0398664 ///
+ z85*-0.0438483 ///
+ z86*-0.0128252 ///
+ z87*-0.0855975 ///
+ z88*-0.1495273 ///
+ z89*-0.0761332 ///
+ z90*-0.1432425 ///
+ z91*-0.0757759 ///
+ z92*-0.0653636 ///
+ z93*-0.1491389 ///
+ z94*-0.3675736 ///
+ z95*-0.0296473 ///
+ z96*-0.0282526 ///
+ z97*-0.0357925 ///
+ z98*-0.0384753 ///
+ z99*-0.0334403 ///
+ z100*0.0014598 ///
+ z101*-0.0408324 ///
+ z102*-0.0642188 ///
+ z103*0.0206044 ///
+ z104*-0.0002747 ///
+ z105*-0.0116543 ///
+ z106*-0.0253964 ///
+ z107*-0.1127119 ///
+ z108*-0.0143062 ///
+ z109*-0.0003608 ///
+ z110*-0.0217701 ///
+ z111*-0.0460375 ///
+ z112*-0.0625727 ///
+ z113*-0.0866826 ///
+ z114*-0.0367975 ///
+ z115*-0.0854255 ///
```

```
+ z116*-0.0626527 ///
+ z117*0.0387891 ///
+ z118*-0.0288829 ///
+ z119*-0.1556403 ///
+ z120*-0.0866662 ///
+ z121*-0.0383929 ///
+ z122*-0.1166656 ///
+ z123*-0.0867575 ///
+ z124*-0.0348978 ///
+ z125*-0.0330382 ///
+ z126*-0.1170501 ///
+ z127*-0.0310476 ///
+ z128*-0.0731964 ///
+ z129*-0.0364444 ///
+ z130*-0.0351324 ///
+ z131*-0.0983128 ///
+ z132*0.0012497 ///
+ z133*-0.0390077 ///
+ z134*-0.0101898 ///
+ z135*-0.0408994 ///
+ z136*-0.0849659 ///
+ z137*-0.0646133 ///
+ z138*-0.0105162 ///
+ z139*-0.0049371 ///
+ z140*-0.0625365 ///
+ z141*-0.0633589 ///
+ z142*-0.0021036 ///
+ z143*0.0125577 ///
+ z144*-0.0012877 ///
+ z145*-0.0443551 ///
+ z146*-0.1090566 ///
+ z147*-0.0463398 ///
+ z148*-0.133609 ///
+ z149*-0.1262862 ///
+ z150*-0.0776235 ///
+ z151*-0.0372242 ///
+ z152*-0.0021036 ///
+ z153*-0.0473768 ///
+ z154*-0.001099 ///
+ z155*-0.0438595 ///
+ z156*-0.0568783 ///
+ z157*-0.0439011 ///
+ z158*-0.043736 ///
+ z159*-0.068438 ///
+ z160*-0.018698 ///
+ z161*-0.0641628 ///
+ z162*-0.1218071 ///
+ z163*-0.0608405 ///
+ z164*-0.0752718 ///
+ z165*-0.0574825 ///
+ z166*-0.0726647 ///
+ z167*-0.0516125 ///
+ z168*-0.0831554 ///
+ z169*-0.0955822 ///
+ z170*-0.0583732 ///
+ z171*-0.0277284 ///
+ z172*-0.1269329 ///
+ z173*-0.1753948 ///
+ z174*0.0367826 ///
+ z175*-0.0013305 ///
+ z176*-0.1103664 ///
+ z177*-0.0232626 ///
+ z178*-0.1005789 ///
+ z179*-0.0900796 ///
+ z180*-0.0053797 ///
+ z181*-0.0022102 ///
+ z182*-0.0408141 ///
+ z183*-0.015596 ///
```

```
+ z184*-0.0019232 ///
+ z185*-0.003297 ///
+ z186*-0.0659663 ///
+ z187*-0.0097629 ///
+ z188*-0.0024727 ///
+ z189*-0.0206471 ///
+ z190*-0.0205704 ///
+ z191*-0.044975 ///
+ z192*-0.0013737 ///
+ z193*-0.0088572 ///
+ z194*0.0010046 ///
+ z195*-0.0037521 ///
+ z196*-0.0479614 ///
+ z197*-0.0482219 ///
+ z198*0.003297 ///
+ z199*-0.0335269 ///
+ z200*0.0011933 ///
+ z201*-0.0705302 ///
+ z202*-0.0832538 ///
+ z203*-0.1659431 ///
+ z204*-0.1144509 ///
+ z205*-0.1179321 ///
+ z206*-0.0692656 ///
+ z207*-0.0330332 ///
+ z208*-0.0796054 ///
+ z209*-0.0280394 ///
+ z210*-0.1168881 ///
+ z211*-0.1442472 ///
+ z212*-0.057222 ///
+ z213*-0.0865975 ///
+ z214*-0.0932371 ///
+ z215*-0.071455 ///
+ z216*-0.1250332 ///
+ z217*-0.1696578 ///
+ z218*-0.0362949 ///
+ z219*-0.0594539 ///
+ z220*-0.0809203 ///
+ z221*-0.271901 ///
+ z222*-0.0181141 ///
+ z223*-0.2049198 ///
+ z224*-0.020492 ///
+ z225*-0.0186058 ///
+ z226*-0.0366582 ///
+ z227*0.0105347 ///
+ z228*-0.0257964 ///
+ z229*-0.2136477 ///
+ z230*-0.122754 ///
+ z231*-0.0045418 ///
+ z232*-0.0019232 ///
+ z233*-0.1567364 ///
+ z234*-0.0667603 ///
+ z235*-0.0347814 ///
+ z236*-0.0575569 ///
+ z237*-0.0946193 ///
+ z238*-0.1420118 ///
+ z239*-0.0651164 ///
+ z240*-0.0414367 ///
}

*Account for people who have died: set utility to 0 if death is on or before the utility day
qui replace utility_`i´ = 0 if date_death <= (date_baseline + `i´)

*Account for utilities after the end date
qui replace utility_`i´ = . if date_baseline + `i´ > 753

*Add one to the count of days (this is equivalent to adding 0.03285 months)
qui replace months = months + 1 if date_baseline + `i´ <= 753
```

```
        dis "Completed day `i´ of `fu´"
}
*Utilities are created for all participants for all days between their registration date and the fol
> low-up window
*Total QALYs are the sum of all utilities, divided by months of follow-up (accounting for the number
>  of utility observations)
egen qaly_hes = rowtotal(utility*)
replace qaly_hes = qaly_hes/months
label variable qaly_hes "Average Utility to October 2022 (i.e. QALYs per year) [HES]"
*And for d4_utility
egen d4_qaly_hes = rowtotal(d4_utility*)
replace d4_qaly_hes = d4_qaly_hes/months
label variable d4_qaly_hes "Average D4 Utility to October 2022 (i.e. QALYs per year) [HES]"
drop z* ncc* utility* v* d4_utility*

compress

save "$stata_sbp_output\part_3a_753_`x´.dta", replace
}
```

The QALYs were predicted over an average follow-up period and the data were saved into **part_3a_753_x.dta** for England and Scotland, where **x** represents the data block (ranging from 1 to 12). Next, the analysis was focused on the Wales data.

```
*******************************************************************************
*Let´s work on follow-up reported by PEDW
*******************************************************************************
forval x = 1/12 {

 use "$stata_sbp_output\part_2a_months_`x´.dta", clear

 keep if dsource == "PEDW"

*Find out how long the follow-up window should be for total QALYs (`fu´)
*This is the longest amount of follow-up, i.e. the first entry to 26/05/2022.
su date_baseline
qui su date_baseline
local min = r(min)
local fu = 748 -`min´
gen months = 0
forvalues i = 0/`fu´ {
        *Utility variables are the estimated utility for each day after baseline (1 is the end of ba
> seline). For convenience we converted each date to monthly format
        *Start the utility as baseline, adjusted for age
        gen utility_`i´ = 0.9630907 + ((age+0.5+`i´/12)*-0.0002747) + (sex*0.0010046) /// constant +
>  age (half a year added to baseline age, plus days after baseline/365.25) + sex
        + 0.0396568 /// constant added for income (mid income), as this isn´t mappable
        + (qual_1*0.0028418) + (qual_2*0.0056836) + (qual_3*0.0060444) // qualifications

        *Update the dummy variables to 1 if the participant had the condition on or before the utili
> ty day
        forvalues v = 1/240 {
                qui replace z`v´ = 1 if v`v´ <= (date_baseline + `i´)
        }

        *Generate d4_utility just using the 4 main conditions (stroke, heart disease, cancer, type 2
>  diabetes)
        *Cancer = v3-v21, diabetes = v35, cardiovascular disease = v112-v123, stroke = v124-v128
        {
        gen d4_utility_`i´ = utility_`i´
        qui replace d4_utility_`i´ = d4_utility_`i´ ///
        + z3*-0.0278367 /// Cancer
        + z4*-0.0705565 ///
        + z5*-0.0673908 ///
        + z6*-0.0929452 ///
        + z7*-0.1192427 ///
```

```
        + z8*-0.0020176 ///
        + z9*-0.0134124 ///
        + z10*-0.0194279 ///
        + z11*-0.1132353 ///
        + z12*-0.0513159 ///
        + z13*-0.049392 ///
        + z14*-0.056553 ///
        + z15*-0.0479982 ///
        + z16*-0.0414255 ///
        + z17*-0.0858906 ///
        + z18*-0.0335121 ///
        + z19*-0.0584308 ///
        + z20*-0.0099273 ///
        + z21*-0.0500121 ///
        + z35*-0.0714349 /// Diabetes
        + z113*-0.0866826 /// Cardiovascular disease
        + z114*-0.0367975 ///
        + z115*-0.0854255 ///
        + z116*-0.0626527 ///
        + z117*0.0387891 ///
        + z118*-0.0288829 ///
        + z119*-0.1556403 ///
        + z120*-0.0866662 ///
        + z121*-0.0383929 ///
        + z122*-0.1166656 ///
        + z123*-0.0867575 ///
        + z124*-0.0348978 /// Cerebrovascular disease
        + z125*-0.0330382 ///
        + z126*-0.1170501 ///
        + z127*-0.0310476 ///
        + z128*-0.0731964

        *Account for number of comorbidities
        egen ncc = rowtotal(z3-z21 z35 z112-z123 z124-z128)
        forvalues j = 2/10 {
                qui replace ncc`j´ = 0 // reset the ncc counters to 0 each time, so if switching fro
> m 2 to 3 comorbidities, you don´t double count the nccs
                qui replace ncc`j´ = 1 if ncc == `j´
        }
        *ncc10 if 10 or more, so account for that
        qui replace ncc10 = 1 if ncc > 10 & ncc < .
        drop ncc

        *Update the utility variable for number of comorbidities
        qui replace d4_utility_`i´ = d4_utility_`i´+(ncc2*-0.0528484) ///
        +(ncc3*-0.0415352) ///
        +(ncc4*-0.0202969) ///
        +(ncc5*0.0083033) ///
        +(ncc6*0.0408673) ///
        +(ncc7*0.0668729) ///
        +(ncc8*0.1158895) ///
        +(ncc9*0.1344392) ///
        +(ncc10*0.183614)

        *Account for people who have died: set utility to 0 if death is on or before the utility day
        qui replace d4_utility_`i´ = 0 if date_death <= (date_baseline + `i´)

        *Account for utilities after the end date
        qui replace d4_utility_`i´ = . if date_baseline + `i´ > 748

        *End of d4_utility code
        }

        *Account for number of comorbidities
        egen ncc = rowtotal(z*)
        forvalues j = 2/10 {
                qui replace ncc`j´ = 0 // reset the ncc counters to 0 each time, so if switching fro
> m 2 to 3 comorbidities, you don´t double count the nccs
```

```
        qui replace ncc`j´ = 1 if ncc == `j´
}
*ncc10 if 10 or more, so account for that
qui replace ncc10 = 1 if ncc > 10 & ncc < .
drop ncc

*Update the utility variable for number of comorbidities
qui replace utility_`i´ = utility_`i´+(ncc2*-0.0528484) ///
+(ncc3*-0.0415352) ///
+(ncc4*-0.0202969) ///
+(ncc5*0.0083033) ///
+(ncc6*0.0408673) ///
+(ncc7*0.0668729) ///
+(ncc8*0.1158895) ///
+(ncc9*0.1344392) ///
+(ncc10*0.183614)

*Main equation code
{
qui replace utility_`i´ = utility_`i´ ///
+ z1*-0.0844983 ///
+ z2*-0.0957076 ///
+ z3*-0.0278367 ///
+ z4*-0.0705565 ///
+ z5*-0.0673908 ///
+ z6*-0.0929452 ///
+ z7*-0.1192427 ///
+ z8*-0.0020176 ///
+ z9*-0.0134124 ///
+ z10*-0.0194279 ///
+ z11*-0.1132353 ///
+ z12*-0.0513159 ///
+ z13*-0.049392 ///
+ z14*-0.056553 ///
+ z15*-0.0479982 ///
+ z16*-0.0414255 ///
+ z17*-0.0858906 ///
+ z18*-0.0335121 ///
+ z19*-0.0584308 ///
+ z20*-0.0099273 ///
+ z21*-0.0500121 ///
+ z22*0.0035188 ///
+ z23*-0.0427221 ///
+ z24*-0.0005495 ///
+ z25*-0.0012877 ///
+ z26*0.0025671 ///
+ z27*-0.0457617 ///
+ z28*0.0021036 ///
+ z29*-0.0356286 ///
+ z30*-0.0320323 ///
+ z31*-0.0345585 ///
+ z32*-0.0064666 ///
+ z33*0.0048511 ///
+ z34*-0.0471758 ///
+ z35*-0.0714349 ///
+ z36*-0.0041212 ///
+ z37*-0.1202905 ///
+ z38*-0.043536 ///
+ z39*-0.1555965 ///
+ z40*0.0251649 ///
+ z41*-0.0302843 ///
+ z42*0.0374719 ///
+ z43*-0.0065353 ///
+ z44*-0.0400731 ///
+ z45*-0.0094726 ///
+ z46*-0.0299147 ///
+ z47*-0.0708597 ///
+ z48*-0.0807239 ///
```

```
+ z49*-0.035499 ///
+ z50*-0.1858287 ///
+ z51*-0.0218789 ///
+ z52*-0.1290684 ///
+ z53*-0.007524 ///
+ z54*0.001099 ///
+ z55*-0.0606417 ///
+ z56*-0.0679451 ///
+ z57*-0.1129732 ///
+ z58*-0.1269103 ///
+ z59*-0.3272071 ///
+ z60*-0.0860614 ///
+ z61*-0.093641 ///
+ z62*-0.087199 ///
+ z63*0.0075985 ///
+ z64*-0.0314791 ///
+ z65*-0.0514317 ///
+ z66*-0.0284652 ///
+ z67*-0.0537585 ///
+ z68*-0.0367998 ///
+ z69*-0.0634499 ///
+ z70*-0.1445047 ///
+ z71*-0.1123433 ///
+ z72*-0.1196535 ///
+ z73*-0.0020176 ///
+ z74*-0.025434 ///
+ z75*-0.2098308 ///
+ z76*-0.2165659 ///
+ z77*-0.227641 ///
+ z78*-0.0968456 ///
+ z79*-0.1496606 ///
+ z80*-0.3915088 ///
+ z81*-0.2114183 ///
+ z82*-0.2459921 ///
+ z83*-0.5262339 ///
+ z84*-0.0398664 ///
+ z85*-0.0438483 ///
+ z86*-0.0128252 ///
+ z87*-0.0855975 ///
+ z88*-0.1495273 ///
+ z89*-0.0761332 ///
+ z90*-0.1432425 ///
+ z91*-0.0757759 ///
+ z92*-0.0653636 ///
+ z93*-0.1491389 ///
+ z94*-0.3675736 ///
+ z95*-0.0296473 ///
+ z96*-0.0282526 ///
+ z97*-0.0357925 ///
+ z98*-0.0384753 ///
+ z99*-0.0334403 ///
+ z100*0.0014598 ///
+ z101*-0.0408324 ///
+ z102*-0.0642188 ///
+ z103*0.0206044 ///
+ z104*-0.0002747 ///
+ z105*-0.0116543 ///
+ z106*-0.0253964 ///
+ z107*-0.1127119 ///
+ z108*-0.0143062 ///
+ z109*-0.0003608 ///
+ z110*-0.0217701 ///
+ z111*-0.0460375 ///
+ z112*-0.0625727 ///
+ z113*-0.0866826 ///
+ z114*-0.0367975 ///
+ z115*-0.0854255 ///
+ z116*-0.0626527 ///
```

```
+ z117*0.0387891 ///
+ z118*-0.0288829 ///
+ z119*-0.1556403 ///
+ z120*-0.0866662 ///
+ z121*-0.0383929 ///
+ z122*-0.1166656 ///
+ z123*-0.0867575 ///
+ z124*-0.0348978 ///
+ z125*-0.0330382 ///
+ z126*-0.1170501 ///
+ z127*-0.0310476 ///
+ z128*-0.0731964 ///
+ z129*-0.0364444 ///
+ z130*-0.0351324 ///
+ z131*-0.0983128 ///
+ z132*0.0012497 ///
+ z133*-0.0390077 ///
+ z134*-0.0101898 ///
+ z135*-0.0408994 ///
+ z136*-0.0849659 ///
+ z137*-0.0646133 ///
+ z138*-0.0105162 ///
+ z139*-0.0049371 ///
+ z140*-0.0625365 ///
+ z141*-0.0633589 ///
+ z142*-0.0021036 ///
+ z143*0.0125577 ///
+ z144*-0.0012877 ///
+ z145*-0.0443551 ///
+ z146*-0.1090566 ///
+ z147*-0.0463398 ///
+ z148*-0.133609 ///
+ z149*-0.1262862 ///
+ z150*-0.0776235 ///
+ z151*-0.0372242 ///
+ z152*-0.0021036 ///
+ z153*-0.0473768 ///
+ z154*-0.001099 ///
+ z155*-0.0438595 ///
+ z156*-0.0568783 ///
+ z157*-0.0439011 ///
+ z158*-0.043736 ///
+ z159*-0.068438 ///
+ z160*-0.018698 ///
+ z161*-0.0641628 ///
+ z162*-0.1218071 ///
+ z163*-0.0608405 ///
+ z164*-0.0752718 ///
+ z165*-0.0574825 ///
+ z166*-0.0726647 ///
+ z167*-0.0516125 ///
+ z168*-0.0831554 ///
+ z169*-0.0955822 ///
+ z170*-0.0583732 ///
+ z171*-0.0277284 ///
+ z172*-0.1269329 ///
+ z173*-0.1753948 ///
+ z174*0.0367826 ///
+ z175*-0.0013305 ///
+ z176*-0.1103664 ///
+ z177*-0.0232626 ///
+ z178*-0.1005789 ///
+ z179*-0.0900796 ///
+ z180*-0.0053797 ///
+ z181*-0.0022102 ///
+ z182*-0.0408141 ///
+ z183*-0.015596 ///
+ z184*-0.0019232 ///
```

```
+ z185*-0.003297 ///
+ z186*-0.0659663 ///
+ z187*-0.0097629 ///
+ z188*-0.0024727 ///
+ z189*-0.0206471 ///
+ z190*-0.0205704 ///
+ z191*-0.044975 ///
+ z192*-0.0013737 ///
+ z193*-0.0088572 ///
+ z194*0.0010046 ///
+ z195*-0.0037521 ///
+ z196*-0.0479614 ///
+ z197*-0.0482219 ///
+ z198*0.003297 ///
+ z199*-0.0335269 ///
+ z200*0.0011933 ///
+ z201*-0.0705302 ///
+ z202*-0.0832538 ///
+ z203*-0.1659431 ///
+ z204*-0.1144509 ///
+ z205*-0.1179321 ///
+ z206*-0.0692656 ///
+ z207*-0.0330332 ///
+ z208*-0.0796054 ///
+ z209*-0.0280394 ///
+ z210*-0.1168881 ///
+ z211*-0.1442472 ///
+ z212*-0.057222 ///
+ z213*-0.0865975 ///
+ z214*-0.0932371 ///
+ z215*-0.071455 ///
+ z216*-0.1250332 ///
+ z217*-0.1696578 ///
+ z218*-0.0362949 ///
+ z219*-0.0594539 ///
+ z220*-0.0809203 ///
+ z221*-0.271901 ///
+ z222*-0.0181141 ///
+ z223*-0.2049198 ///
+ z224*-0.020492 ///
+ z225*-0.0186058 ///
+ z226*-0.0366582 ///
+ z227*0.0105347 ///
+ z228*-0.0257964 ///
+ z229*-0.2136477 ///
+ z230*-0.122754 ///
+ z231*-0.0045418 ///
+ z232*-0.0019232 ///
+ z233*-0.1567364 ///
+ z234*-0.0667603 ///
+ z235*-0.0347814 ///
+ z236*-0.0575569 ///
+ z237*-0.0946193 ///
+ z238*-0.1420118 ///
+ z239*-0.0651164 ///
+ z240*-0.0414367 ///
}

*Account for people who have died: set utility to 0 if death is on or before the utility day
qui replace utility_`i´ = 0 if date_death <= (date_baseline + `i´)

*Account for utilities after the end date
qui replace utility_`i´ = . if date_baseline + `i´ > 748

*Add one to the count of days (this is equivalent to adding 0.03285 months)
qui replace months = months + 1 if date_baseline + `i´ <= 748

dis "Completed day `i´ of `fu´"
```

```
}
*Utilities are created for all participants for all days between their registration date and the fol
> low-up window
*Total QALYs are the sum of all utilities, divided by months of follow-up (accounting for the number
>  of utility observations)
egen qaly_hes = rowtotal(utility*)
replace qaly_hes = qaly_hes/months
label variable qaly_hes "Average Utility to May 2022 (i.e. QALYs per year) [HES]"
*And for d4_utility
egen d4_qaly_hes = rowtotal(d4_utility*)
replace d4_qaly_hes = d4_qaly_hes/months
label variable d4_qaly_hes "Average D4 Utility to May 2022 (i.e. QALYs per year) [HES]"
drop z* ncc* utility* v* d4_utility*
compress

save "$stata_sbp_output\part_3a_748_`x´.dta", replace
}
```

The QALYs were also predicted over an average follow-up period and the data were saved into **part_3a_748_x.dta** for Wales, where **x** represents the data block (ranging from 1 to 12). Next, we merged these two datasets.

```
use "$stata_sbp_output\part_3a_753_1.dta", clear
forval i == 2/12 {
        append using "$stata_sbp_output\part_3a_753_`i´.dta"
}
save "$stata_sbp_output\part_3a_753.dta", replace
use "$stata_sbp_output\part_3a_748_1.dta", clear
forval i == 2/12 {
        append using "$stata_sbp_output\part_3a_748_`i´.dta"
}
save "$stata_sbp_output\part_3a_748.dta", replace
append using "$stata_sbp_output\part_3a_753.dta"
tabstat qaly_hes d4_qaly_hes, statistics(mean sd median p25 p75 min max)
replace qaly_hes = 1 if qaly_hes >1
replace d4_qaly_hes = 1 if d4_qaly_hes >1
save "$stata_sbp_output\part_3a.dta", replace
```

4.1.6 Working on EQ-5D data collected by UK Biobank

Web based **EQ-5D-5L** questionnaires were administered to the UK Biobank participants as part of the chronic pain (administered in 2019–20) and mental well-being (administered in 2022–23) surveys. We calculated the **EQ-5D index** using the **UK tariffs** (i.e., value set) for each survey[10]. Once the EQ-5D-indexes were calculated, we took the average EQ-5D-index for participants who had EQ-5D data for both surveys (124,830). The remaining participants had EQ-5D data for either chronic pain survey (42,281) or mental well-being survey (44,707); hence, the average EQ-5D index was not calculated. The next line of codes will prepare our data for EQ-5D index calculation, then apply the UK tariffs.

```
use "$stata_sbp_output\part_3a.dta", clear
 *Work on EQ_5D data collected by UKB
 rename (n_120098 n_120099 n_120100 n_120101 n_120102) (mobility selfcare activity pain anxiety)
 rename (n_29150 n_29151 n_29152 n_29153 n_29154) (mobility_1 selfcare_1 activity_1 pain_1 anxiety_1
> )
foreach j in mobility selfcare activity pain anxiety {

        replace `j´ = 1 if `j´ == -521
        replace `j´ = 2 if `j´ == -522
```

```
        replace `j´ = 3 if `j´ == -523
        replace `j´ = 4 if `j´ == -524
        replace `j´ = 5 if `j´ == -525

}
foreach j in mobility_1 selfcare_1 activity_1 pain_1 anxiety_1 {
        replace `j´ = `j´+1 if `j´ !=.

}

*Work on the health states
egen health_states = concat(mobility selfcare activity pain anxiety)
tostring health_states, replace force
replace health_states = "88" if health_states =="....."
destring health_states, replace
replace health_states =. if health_states ==88
order health_states, a(anxiety)
egen health_states_1 = concat(mobility_1 selfcare_1 activity_1 pain_1 anxiety_1)
tostring health_states_1, replace force
replace health_states_1 = "88" if health_states_1 =="....."
destring health_states_1, replace
replace health_states_1 =. if health_states_1 ==88
order health_states_1, a(anxiety_1)

/*
Computing EQ-5D-5L index values with STATA using the English (ENG) Devlin value set
Version 1.1 (Updated 01/12/2020)
The variables for the 5 dimensions of the EQ-5D-5L descriptive system should be named ´mobility´,
´selfcare´, ´activity´, ´pain´, and ´anxiety´. If they are given different names the syntax code
below will not work properly. The 5 variables should contain the values for the different dimensions
>
in the EQ-5D health profile (i.e. 1, 2, 3, 4 or 5). The variable ´EQindex´ contains the values of th
> e
EQ-5D-5L index values on the basis of the ENG set of weights.
You can copy and paste the syntax below directly into a STATA syntax window.
*/
*********************************************************************
*STATA syntax code for the computation of index*
*values with the English value set*
*********************************************************************

 gen disut_mo= .
 replace disut_mo= 0 if missing(disut_mo) ///
 & mobility == 1
 replace disut_mo= 0.058 if missing(disut_mo) ///
 & mobility == 2
 replace disut_mo= 0.076  if missing(disut_mo) ///
 & mobility == 3
 replace disut_mo= 0.207 if missing(disut_mo) ///
 & mobility == 4
 replace disut_mo= 0.274  if missing(disut_mo) ///
 & mobility == 5

 gen disut_sc= .
 replace disut_sc= 0 if missing(disut_sc) ///
 & selfcare == 1
 replace disut_sc= 0.050 if missing(disut_sc) ///
 & selfcare == 2
 replace disut_sc= 0.080  if missing(disut_sc) ///
 & selfcare == 3
 replace disut_sc= 0.164 if missing(disut_sc) ///
 & selfcare == 4
 replace disut_sc= 0.203  if missing(disut_sc) ///
 & selfcare == 5

 gen disut_ua= .
 replace disut_ua= 0 if missing(disut_ua) ///
```

```
 & activity == 1
 replace disut_ua= 0.050 if missing(disut_ua) ///
 & activity == 2
 replace disut_ua= 0.063  if missing(disut_ua) ///
 & activity == 3
 replace disut_ua= 0.162 if missing(disut_ua) ///
 & activity == 4
 replace disut_ua= 0.184  if missing(disut_ua) ///
 & activity == 5

 gen disut_pd= .
 replace disut_pd= 0 if missing(disut_pd) ///
 & pain == 1
 replace disut_pd= 0.063 if missing(disut_pd) ///
 & pain == 2
 replace disut_pd= 0.084  if missing(disut_pd) ///
 & pain == 3
 replace disut_pd= 0.276 if missing(disut_pd) ///
 & pain == 4
 replace disut_pd= 0.335  if missing(disut_pd) ///
 & pain == 5

 gen disut_ad= .
 replace disut_ad= 0 if missing(disut_ad) ///
 & anxiety == 1
 replace disut_ad= 0.078 if missing(disut_ad) ///
 & anxiety == 2
 replace disut_ad= 0.104  if missing(disut_ad) ///
 & anxiety == 3
 replace disut_ad= 0.285 if missing(disut_ad) ///
 & anxiety == 4
 replace disut_ad= 0.289  if missing(disut_ad) ///
 & anxiety == 5
gen disut_total=disut_mo+disut_sc+disut_ua+disut_pd+disut_ad
gen EQindex=.
replace EQindex=1-disut_total
replace EQindex=round(EQindex,.001)
order disut_mo disut_sc disut_ua disut_pd disut_ad disut_total EQindex, b(mobility_1)

 gen disut_mo_1= .
 replace disut_mo_1= 0 if missing(disut_mo_1) ///
 & mobility_1 == 1
 replace disut_mo_1= 0.058 if missing(disut_mo_1) ///
 & mobility_1 == 2
 replace disut_mo_1= 0.076  if missing(disut_mo_1) ///
 & mobility_1 == 3
 replace disut_mo_1= 0.207 if missing(disut_mo_1) ///
 & mobility_1 == 4
 replace disut_mo_1= 0.274  if missing(disut_mo_1) ///
 & mobility_1 == 5

 gen disut_sc_1= .
 replace disut_sc_1= 0 if missing(disut_sc_1) ///
 & selfcare_1 == 1
 replace disut_sc_1= 0.050 if missing(disut_sc_1) ///
 & selfcare_1 == 2
 replace disut_sc_1= 0.080  if missing(disut_sc_1) ///
 & selfcare_1 == 3
 replace disut_sc_1= 0.164 if missing(disut_sc_1) ///
 & selfcare_1 == 4
 replace disut_sc_1= 0.203  if missing(disut_sc_1) ///
 & selfcare_1 == 5

 gen disut_ua_1= .
 replace disut_ua_1= 0 if missing(disut_ua_1) ///
 & activity_1 == 1
 replace disut_ua_1= 0.050 if missing(disut_ua_1) ///
 & activity_1 == 2
```

```
replace disut_ua_1= 0.063  if missing(disut_ua_1) ///
& activity_1 == 3
replace disut_ua_1= 0.162 if missing(disut_ua_1) ///
& activity_1 == 4
replace disut_ua_1= 0.184  if missing(disut_ua_1) ///
& activity_1 == 5

gen disut_pd_1= .
replace disut_pd_1= 0 if missing(disut_pd_1) ///
& pain_1 == 1
replace disut_pd_1= 0.063 if missing(disut_pd_1) ///
& pain_1 == 2
replace disut_pd_1= 0.084  if missing(disut_pd_1) ///
& pain_1 == 3
replace disut_pd_1= 0.276 if missing(disut_pd_1) ///
& pain_1 == 4
replace disut_pd_1= 0.335  if missing(disut_pd_1) ///
& pain_1 == 5

gen disut_ad_1= .
replace disut_ad_1= 0 if missing(disut_ad_1) ///
& anxiety_1 == 1
replace disut_ad_1= 0.078 if missing(disut_ad_1) ///
& anxiety_1 == 2
replace disut_ad_1= 0.104  if missing(disut_ad_1) ///
& anxiety_1 == 3
replace disut_ad_1= 0.285 if missing(disut_ad_1) ///
& anxiety_1 == 4
replace disut_ad_1= 0.289  if missing(disut_ad_1) ///
& anxiety_1 == 5
gen disut_total_1=disut_mo_1+disut_sc_1+disut_ua_1+disut_pd_1+disut_ad_1
gen EQindex_1=.
replace EQindex_1=1-disut_total_1
replace EQindex_1 =round(EQindex_1,.001)
order disut_mo_1 disut_sc_1 disut_ua_1 disut_pd_1 disut_ad_1 disut_total_1 EQindex_1, a(health_state
> s_1)

rename n_120103 EQ_VAS
rename n_29155  EQ_VAS_1
rename ts_120128 qol_date
rename ts_29206 qol_date_1
tabstat EQindex EQindex_1 EQ_VAS EQ_VAS_1, statistics(mean sd median p25 p75 min max)
count if (EQindex !=. & EQindex_1 !=.)
count if (EQindex !=. & EQindex_1 ==.)
count if (EQindex ==. & EQindex_1 !=.)

save "$stata_sbp_output\part_3b.dta", replace
```

We have now completed the first step. We will work on the **genotype data**.

4.2 Step 2: Working on genotype data

4.2.1 Preparing our data

To run a mendelian randomisation (MR) analysis, a type of instrumental variable analysis, to estimate the causal association of SBP (i.e., exposure) with QALYs (i.e., outcome) using genetic variants as instruments, we need genetic variants that are strongly associated with the exposure (i.e., trait) of interest. In this case, we need genetic variants (also known as single nucleotide polymorphisms) that are associated with SBP. There are a number of assumptions genetic variants should meet to run an MR analysis[6, 11, 12].

- **relevance**: the variant is associated with the exposure
- **exchangeability**: the variant is not associated with the outcome via a confounding pathway
- **exclusion restriction**: the variant does not affect the outcome directly, only possibly indirectly via exposure

For this we look into a study conducted by Evangelou E. et al[13]. This genome-wide association study (GWAS) included data from the UK Biobank, International Consortium for Blood Pressure (ICBP), the US Million Veteran Program (MVP) and Estonian Genome Centre, University of Tartu (EGCUT). The UK Biobank and ICBP data were used for discovery meta-analysis, while the MVP and EGCUT data were used for replication meta-analysis. A combined meta-analysis was also performed using all data sources[13]. The GWAS study has identified 535 novel loci (1 variant per locus) that have reached the significant threshold to one of the blood pressure traits[13]. The criteria for significance threshold for genetic variants with a specific trait was based on one-stage or two-stage analysis design set by the GWAS study[13]. SNPs were filtered to meet criteria of genotype missingness below 0.015 and minor allele frequency above 0.01[13]. They were also tested for Hardy-Weinberg equilibrium and linkage disequilibrium within the GWAS[13]. Linkage disequilibrium (LD) was calculated for all variants within a 500kb window on either side of the reference SNP[13]. Variants in linkage disequilibrium with the reference SNP, reaching an r^2 threshold of 0.1 or higher, were identified[13].

The GWAS study also included 92 sentinel SNPs previously known but replicated for the first time and 357 SNPs already known and validated to have association with one of the blood pressure traits.

Among the 984 SNPs, 282 were primarily related to SBP[13]. After LD clumping (r^2 0.001 and 10,000 kb window) and removing ambiguous SNPs, 181 genetic instruments were candidates for building polygenic risk scores (PRS). If a sentinel SNP was not available in the UK Biobank, a proxy SNP was substituted ($r^2 \geq 0.8$)[13]. PRS was constructed by summing the effects of the 181 SNPs on SBP, each weighted by its effect size derived from non-UK Biobank cohorts (the ICBP meta-analysis and replication meta-analysis). The ICBP and replication meta-analyses were selected to avoid cohort overlap with the UK Biobank population.

We also need diastolic blood pressure for the multivariable MR (MVMR)analysis, an extention of standard MR analysis. The GWAS study reported sentinel SNPs association with primary and secondary blood pressure traits[13]. For the 187 sentinel SNPs primarily associated with SBP (after LD clumping), we also identified association with DBP as a secondary trait. Similarly, we also identified 208 sentinel SNPs (after LD clumping) primarily associated with DBP that were

also linked to SBP. The combined 395 SNPs were candidates for the MVMR analysis. After the exclusion of ambiguous and missing effect size SNPs, 384 and 382 SNPs were used to construct PRS for SBP and DBP, respectively. Effect estimates for the SNPs were sourced from either ICBP or replication meta-analysis.

4.2.2 Association of Genetic Variants with blood pressure traits

The following stata code will help us to format the data suitable for the next analysis.

```
*******************************************************************************
*Let´s work on known SNPs
*******************************************************************************
import excel "$data_source\known_bp_snps.xlsx", firstrow clear

save "$snps\known_bp_snps.dta",replace

import excel "$data_source\known_bp_snps_association.xlsx", firstrow clear

save "$snps\known_bp_snps_association.dta",replace

merge 1:1 rsID using "$snps\known_bp_snps.dta"

replace Trait = "SBP" if rsID == "rs2076328"
replace Trait = "PP" if rsID == "rs2157597"
replace Trait = "PP" if rsID == "rs28427409"
replace Trait = "SBP" if rsID == "rs6783086"
replace Trait = "DBP" if rsID == "rs73030266"
replace Trait = "DBP" if rsID == "rs73091767"
replace Trait = "PP" if rsID == "rs7480089"
replace Trait = "DBP" if rsID == "rs7777128"
replace Trait = "DBP" if rsID == "rs7810028"
replace Trait = "PP" if rsID == "rs9479200"

count if SNP_Gwsig == "FALSE" // not reached GWAS significant level for the primary trait (16)
count if SNP_Gwsig == "n/a" // no locus covarage, rare variant (MAF<1%), low frequency variant (MAF
> 1%-5%), or  not in Haplotype Reference Consortium (28)

drop if _merge == 2

drop _merge*
* the snps included in the analysis are low frequency or common varinats, and LD prunned.
preserve

keep if Trait == "SBP"

gen status = "known"

gen effect_source = "icbp"

replace P_min = Pmin_ICBP_UKBmeta if P_min == ""
drop *PP Pmin_ICBP_UKBmeta

foreach var of varlist EAF_SBP Beta_SBP se_SBP P_SBP EAF_DBP Beta_DBP se_DBP P_DBP MAF P_min {
        destring `var´, replace
}

save "$sbp_snps\known_snps_sbp.dta",replace
export delimited "$sbp_snps\known_snps_sbp.csv", replace

restore

keep if Trait == "DBP"

gen status = "known"

gen effect_source = "icbp"

replace P_min = Pmin_ICBP_UKBmeta if P_min == ""
drop *PP Pmin_ICBP_UKBmeta

foreach var of varlist EAF_SBP Beta_SBP se_SBP P_SBP EAF_DBP Beta_DBP se_DBP P_DBP MAF P_min {
        destring `var´, replace
}

save "$dbp_snps\known_snps_dbp.dta",replace
export delimited "$dbp_snps\known_snps_dbp.csv", replace

*******************************************************************************
*Let´s work on novel (1stage210 and 2stage325) previously known but replicated
*for the first time SNPs (non) associated with SBP
```

```
*************************************************************************
import excel "$data_source\1stage_snps_secondary_effect.xlsx", firstrow clear
rename (SBPICBPBETA SBPICBPSE SBPICBPPVAL DBPICBPBETA DBPICBPSE DBPICBPPVAL) (SBP_ICBP_BETA SBP_ICBP
> _SE SBP_ICBP_PVAL DBP_ICBP_BETA DBP_ICBP_SE DBP_ICBP_PVAL)
foreach x in SBP_ICBP_BETA SBP_ICBP_SE SBP_ICBP_PVAL DBP_ICBP_BETA DBP_ICBP_SE DBP_ICBP_PVAL {
        destring `x´, replace
}
save "$snps\one_stage_snps_secondary_effect.dta",replace
import excel "$data_source\2stage_snps_secondary_effect.xlsx", firstrow clear
rename (SBPrepBETA SBPrepSE SBPrepPVAL DBPrepBETA DBPrepSE DBPrepPVAL) (SBP_rep_BETA SBP_rep_SE SBP_
> rep_PVAL DBP_rep_BETA DBP_rep_SE DBP_rep_PVAL)
foreach x in SBP_rep_BETA SBP_rep_SE SBP_rep_PVAL DBP_rep_BETA DBP_rep_SE DBP_rep_PVAL {
        destring `x´, replace
}
save "$snps\two_stage_snps_secondary_effect.dta",replace

import excel "$sbp_snps\novel_sbp_snps.xlsx", firstrow clear


 gen rsID = ""
 gen chrpos = ""
 gen A1 = ""
 gen A2 = ""
 gen EAF_SBP =.
 gen Beta_SBP =.
 gen se_SBP =.
 gen P_SBP =.
 gen EAF_DBP =.
 gen Beta_DBP =.
 gen se_DBP =.
 gen P_DBP =.
 gen MAF =.
 gen P_min =.
 gen SNP_Gwsig = ""
 gen status = ""
 gen effect_source = ""

 replace rsID = rsID_proxy_SNP
 replace chrpos = ChrPos_RepSNP
 replace A1 = A1_rep if Type == "2stage325" | Type == "non"
 replace A1 = A1_icbp if Type == "1stage210"
 replace A2 = A2_rep if Type == "2stage325" | Type == "non"
 replace A2 = A2_icbp if Type == "1stage210"
 replace EAF_SBP = EAF_rep if Type == "2stage325" | Type == "non"
 replace EAF_SBP = EAF_icbp if Type == "1stage210"
 replace Beta_SBP = BETA_rep if Type == "2stage325" | Type == "non"
 replace Beta_SBP = BETA_icbp if Type == "1stage210"
 replace se_SBP = SE_rep if  Type == "2stage325" | Type == "non"
 replace se_SBP = SE_icbp if  Type == "1stage210"
 replace P_SBP = P_rep if Type == "2stage325" | Type == "non"
 replace P_SBP = P_icbp if Type == "1stage210"
 replace MAF = MAF_uk
 replace P_min = P_comb if Type == "2stage325" | Type == "non"
 replace P_min = P_disc if Type == "1stage210"
 replace SNP_Gwsig = "TRUE"
 replace status = "novel" if Type == "1stage210" | Type == "2stage325"
 replace status = "replicated" if Type == "non"
 replace effect_source = "replicaton_study" if Type == "2stage325" | Type == "non"
 replace effect_source = "icbp" if Type == "1stage210"
 keep  rsID chrpos A1 A2 EAF_SBP Beta_SBP se_SBP P_SBP EAF_DBP Beta_DBP se_DBP P_DBP MAF Trait P_min
>  SNP_Gwsig status effect_source Type
 order rsID chrpos A1 A2 EAF_SBP Beta_SBP se_SBP P_SBP EAF_DBP Beta_DBP se_DBP P_DBP MAF Trait P_min
>  SNP_Gwsig status effect_source Type

 save "$sbp_snps\novel_rep_sbp_snps.dta", replace
 export delimited "$sbp_snps\novel_rep_sbp_snps.csv", replace
```

```
use "$sbp_snps\novel_rep_sbp_snps.dta", clear
append using "$sbp_snps\known_snps_sbp.dta"
merge 1:1 rsID using "$snps\one_stage_snps_secondary_effect.dta"
drop if _merge == 2
drop _merge*

merge 1:1 rsID using "$snps\two_stage_snps_secondary_effect.dta"
drop if _merge == 2
drop _merge*

replace Beta_DBP = DBP_ICBP_BETA if Beta_DBP==. & Type == "1stage210"
replace se_DBP = DBP_ICBP_SE if se_DBP ==. & Type == "1stage210"
replace P_DBP = DBP_ICBP_PVAL if P_DBP ==. & Type == "1stage210"

replace Beta_DBP = DBP_rep_BETA if Beta_DBP==. & Type == "2stage325"
replace se_DBP = DBP_rep_SE if se_DBP ==. & Type == "2stage325"
replace P_DBP = DBP_rep_PVAL if P_DBP ==. & Type == "2stage325"

replace Beta_DBP = DBP_rep_BETA if Beta_DBP==. & Type == "non"
replace se_DBP = DBP_rep_SE if se_DBP ==. & Type == "non"
replace P_DBP = DBP_rep_PVAL if P_DBP ==. & Type == "non"
replace EAF_DBP = EAF_SBP if EAF_DBP ==.

drop  a1_icbp a2_icbp SBP_ICBP_BETA SBP_ICBP_SE SBP_ICBP_PVAL DBP_ICBP_BETA DBP_ICBP_SE DBP_ICBP_PV
> AL a1_rep a2_rep SBP_rep_BETA SBP_rep_SE SBP_rep_PVAL DBP_rep_BETA DBP_rep_SE DBP_rep_PVAL

save "$sbp_snps\all_sbp_snps.dta", replace
export delimited "$sbp_snps\all_sbp_snps.csv", replace

*******************************************************************************
*Let´s work on novel (1stage210 and 2stage325) previously known but replicated
*for the first time SNPs (non) associated with DBP
*******************************************************************************
import excel "$dbp_snps\novel_dbp_snps.xlsx", firstrow clear


gen rsID = ""
gen chrpos = ""
gen A1 = ""
gen A2 = ""
gen EAF_SBP =.
gen Beta_SBP =.
gen se_SBP =.
gen P_SBP =.
gen EAF_DBP =.
gen Beta_DBP =.
gen se_DBP =.
gen P_DBP =.
gen MAF =.
gen P_min =.
gen SNP_Gwsig = ""
gen status = ""
gen effect_source = ""

replace rsID = rsID_proxy_SNP
replace chrpos = ChrPos_RepSNP
replace A1 = A1_rep if Type == "2stage325" | Type == "non"
replace A1 = A1_icbp if Type == "1stage210"
replace A2 = A2_rep if Type == "2stage325" | Type == "non"
replace A2 = A2_icbp if Type == "1stage210"
replace EAF_DBP = EAF_rep if Type == "2stage325" | Type == "non"
replace EAF_DBP = EAF_icbp if Type == "1stage210"
replace Beta_DBP = BETA_rep if Type == "2stage325" | Type == "non"
replace Beta_DBP = BETA_icbp if Type == "1stage210"
replace se_DBP = SE_rep if  Type == "2stage325" | Type == "non"
replace se_DBP = SE_icbp if  Type == "1stage210"
replace P_DBP = P_rep if Type == "2stage325" | Type == "non"
replace P_DBP = P_icbp if Type == "1stage210"
replace MAF = MAF_uk
```

```
replace P_min = P_comb if Type == "2stage325" | Type == "non"
replace P_min = P_disc if Type == "1stage210"
replace SNP_Gwsig = "TRUE"
replace status = "novel" if Type == "1stage210" | Type == "2stage325"
replace status = "replicated" if Type == "non"
replace effect_source = "replicaton_study" if Type == "2stage325" | Type == "non"
replace effect_source = "icbp" if Type == "1stage210"
keep  rsID chrpos A1 A2 EAF_SBP Beta_SBP se_SBP P_SBP EAF_DBP Beta_DBP se_DBP P_DBP MAF Trait P_min
>  SNP_Gwsig status effect_source Type
order rsID chrpos A1 A2 EAF_SBP Beta_SBP se_SBP P_SBP EAF_DBP Beta_DBP se_DBP P_DBP MAF Trait P_min
>  SNP_Gwsig status effect_source Type

save "$dbp_snps\novel_rep_dbp_snps.dta", replace
export delimited "$dbp_snps\novel_rep_dbp_snps.csv", replace

use "$dbp_snps\novel_rep_dbp_snps.dta", clear
append using "$dbp_snps\known_snps_dbp.dta"
merge 1:1 rsID using "$snps\one_stage_snps_secondary_effect.dta"
drop if _merge == 2
drop _merge*

merge 1:1 rsID using "$snps\two_stage_snps_secondary_effect.dta"
drop if _merge == 2
drop _merge*

replace Beta_SBP = SBP_ICBP_BETA if Beta_SBP==. & Type == "1stage210"
replace se_SBP = SBP_ICBP_SE if se_SBP ==. & Type == "1stage210"
replace P_SBP = SBP_ICBP_PVAL if P_SBP ==. & Type == "1stage210"

replace Beta_SBP = SBP_rep_BETA if Beta_SBP==. & Type == "2stage325"
replace se_SBP = SBP_rep_SE if se_SBP ==. & Type == "2stage325"
replace P_SBP = SBP_rep_PVAL if P_SBP ==. & Type == "2stage325"

replace Beta_SBP = SBP_rep_BETA if Beta_SBP==. & Type == "non"
replace se_SBP = SBP_rep_SE if se_SBP ==. & Type == "non"
replace P_SBP = SBP_rep_PVAL if P_SBP ==. & Type == "non"
replace EAF_SBP = EAF_DBP if EAF_SBP ==.

drop  a1_icbp a2_icbp SBP_ICBP_BETA SBP_ICBP_SE SBP_ICBP_PVAL DBP_ICBP_BETA DBP_ICBP_SE DBP_ICBP_PV
> AL a1_rep a2_rep SBP_rep_BETA SBP_rep_SE SBP_rep_PVAL DBP_rep_BETA DBP_rep_SE DBP_rep_PVAL


save "$dbp_snps\all_dbp_snps.dta", replace
export delimited "$dbp_snps\all_dbp_snps.csv", replace

*******************************************************************************
*Let´s combine all SBP and DBP SNPs
*******************************************************************************
use "$sbp_snps\all_sbp_snps.dta", clear
append using "$dbp_snps\all_dbp_snps.dta"

save "$sbp_dbp_snps\all_sbp_dbp_snps.dta"
export delimited "$sbp_dbp_snps\all_sbp_dbp_snps.csv", replace
```

4.2.3 LD Clumping

The previous Stata codes provided CSV files for the next step: LD clumping using the **TwoSampleMR** and **ieugwas** packages from R. First, an Application Programming Interface (API) was set up to access the **IEU GWAS** database. Then, the SNPs were clumped.

```r
#This is to set up the API
# Get the location of your .Renviron file
renviron_path <- Sys.getenv("R_ENVIRON_USER")
```

```r

# If the .Renviron file doesn't exist, create it
if (renviron_path == "") {
  renviron_path <- file.path(Sys.getenv("HOME"), ".Renviron")
  file.create(renviron_path)
}

# Append your token to the .Renviron file
cat("OPENGWAS_JWT=<Add Your Token Here>", file = renviron_path, append =
    TRUE)

# Print the path to verify
print(renviron_path)


library(ieugwasr)

# Check if the token is loaded
jwt <- ieugwasr::get_opengwas_jwt()
if (nzchar(jwt)) {
  cat("Token is recognized:\n", jwt, "\n")
} else {
  cat("Token is not recognized. Check your .Renviron file.\n")
}


# Retrieve user information
user_info <- ieugwasr::user()

# Check if user information is retrieved
if (!is.null(user_info)) {
  print("Token is working. User information:")
  print(user_info)
} else {
  print("Token is not working. Check your token and internet connection.")
}

######################################################################

#Load the necessary packages

library(MRPracticals)
library(TwoSampleMR)
library(ieugwasr)
library(MRInstruments)
library(dplyr)
library(readxl)

vignette("MRBase")

######################################################################
```

```
54
55 #Run the analysis from here onwards
56 setwd("C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/
      Articles/Evangelou/new_sbp_snps/stata/sbp_snps")
57 getwd()
58
59
60
61 sbp_data<-read.csv("all_sbp_snps.csv")
62 sbp_data[c("Chromosome", "Position")]<-do.call(rbind,strsplit(sbp_data$
      chrpos, ":"))
63 sbp_data$Chromosome<-as.numeric(sbp_data$Chromosome)
64 sbp_data$Position<-as.numeric(sbp_data$Position)
65
66
67 sbp_data_2<-sbp_data%>%select(Chromosome, Position, rsID, Beta_SBP, se_SBP
      , A1, A2, EAF_SBP, P_min, Trait, Beta_DBP, se_DBP, EAF_DBP)%>%mutate(id
      .exposure = "icbp_rep")
68 colnames(sbp_data_2)
69
70 colnames(sbp_data_2)<-c("chr.exposure", "pos.exposure", "SNP", "beta.
      exposure", "se.exposure","effect_allele.exposure", "other_allele.
      exposure", "eaf.exposure", "pval.exposure", "exposure", "Beta_DBP", "se
      _DBP", "EAF_DBP", "id.exposure")
71 head(sbp_data_2)
72
73 sbp_data_2<-sbp_data_2[order(sbp_data_2$chr.exposure),]
74
75 clumped_sbp_data_2 <- clump_data(sbp_data_2,
76                                  clump_kb = 10000,  # Clumping window
                                     (10,000 kb)
77                                  clump_r2 = 0.001,  # LD threshold (r2 <
                                     0.001)
78                                  pop = "EUR")  # European LD reference
79
80 sbp_snplist<-clumped_sbp_data_2%>%select(SNP)
81
82 #sbp_effect_list<-clumped_sbp_data_2%>%select(SNP,effect_allele.exposure,
      beta.exposure)
83
84
85
86 write.csv(clumped_sbp_data_2, "sbp_exposure.csv", row.names = FALSE)
87 write.table(sbp_snplist, "C:/Users/tabe0010/OneDrive - Monash University/
      MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/sbp/data/
      sbp_snplist.txt",row.names = FALSE, col.names =FALSE,quote = FALSE,sep
      = " ")
```

We have clumped the SNPs, and the necessary genetic variants primarily associated with SBP were selected, totaling 187 SNPs. The **snp_snplist.txt** file was then used to select the required genetic variants from the UK Biobank imputed BGEN file.

4.2.4 Selecting the genetic variants from the BGEN file

PLINK2 was used via the **Swiss Army Knife** to select the necessary SNPs from the UK Biobank. The UKB RAP served as the platform to access the SNPs from the UK Biobank. Alternatively, a *bash* command was used on the local machine to run the process. First, log in with your UKB RAP credentials using the Command Prompt (on a Windows machine). Then, open Git Bash, select your project, and run the analysis. A job request was sent, and users were notified when the process was complete.

```
##########################################################################
#Run the following PLINK2 codes on Git Bash
##########################################################################

#Login through Command Prompt on your Windows machine
dx login

#Run the code below on Git Bash

dx select --level VIEW
#select your project
# make sure your sbp_snplist.txt file is uploaded to the UKB RAP.
#select the "instance type" you want: this makes sure you have enough computation
    power (CPU and GPU).
#In the command below, I put chromosome 1 to 22 to loop through all autosomal
    chromosomes just to show the code. But in actuality, I put two chromosomes at a
     time. This makes sure I have enough computational space and if there is any
    error, I could adjust the code.

# Loop over chromosomes 1 to 22 and process each one with the SNP list
run_merge=""
for chr in {1..22}; do
    run_merge+="cp /mnt/project/Bulk/Imputation/UKB\ imputation\ from\ genotype/
        ukb22828_c${chr}_b0_v3.bgen .; "
    run_merge+="cp /mnt/project/Bulk/Imputation/UKB\ imputation\ from\ genotype/
        ukb22828_c${chr}_b0_v3.sample .; "
    run_merge+="plink2 --bgen ukb22828_c${chr}_b0_v3.bgen ref-first --sample
        ukb22828_c${chr}_b0_v3.sample --extract sbp_snplist.txt --make-pgen --
        autosome-xy --out ukb22828_c${chr}_v3; "
done

dx run swiss-army-knife -iin="project-GpbQqBjJb7jb1vQjf8ZxVpVY:/SBP_data/sbp_txt/
    sbp_snplist.txt" -icmd="${run_merge}" --tag="Step1" --instance-type "
    mem1_ssd1_v2_x36" --destination="project-GpbQqBjJb7jb1vQjf8ZxVpVY:/SBP_data/
    sbp_geno_data/" --brief --yes

##########################################################################
#Run the following PLINK2 codes via Swiss Army Knife on UKB RAP platform
##########################################################################
#Make sure you have uploaded the sbp_merge_list.txt to UKB RAP file path.
#The merge list should have a sigle column list containing the following text, "
    ukb22828_cx_v3" (without the quotations). The column wil have 22 rows for each
    autosomal chromosomes. Replace "x" with 1-22.

#merging the pgen files
```

```
#Execute the command on Swiss Army Knife interface
#inputs are the plink files for the chromosomes and the txt file for the merging
    chromosomes
plink2 --pmerge-list sbp_merge_list.txt pfile --make-pgen --out
    ukb22828_c1_22_v3_sbp_merged

#Calculate the allele dosage
#creating a .raw file for participants with the number of effect allele (0, 1, or
    2) for each snp
# Input the for code below is the merged plink files (pfiles)

plink2 --pfile ukb22828_c1_22_v3_sbp_merged --export A --out ukb22828_sbp_alleles

#Calculate allele frequency

plink2 --pfile ukb22828_c1_22_v3_sbp_merged --freq --out ukb22828_sbp_allele_freq
```

The final PLINK output files, **ukb22828_sbp_alleles.raw** and **ukb22828_sbp_allele_freq.afreq**, contained the allele dosage and allele frequency for each of the 187 SNPs. These files were downloaded to the local machine and saved to the `$dx_data_sbp` file path.

4.2.5 Association of Genetic Variants with Quality-Adjusted Life Years

We then worked on the association between the genetic variants and QALYs. The QALYs were regressed on the allele dosages, adjusting for age, sex, and the first 10 genetic principal components to account for population stratification.

```
*******************************************************************************
*SNP-QALYs association
*******************************************************************************
import delimited "$dx_data_sbp\ukb22828_sbp_alleles.raw",clear
keep iid rs*
rename iid id_phe
merge 1:1 id_phe using "$stata_sbp_input\id_list.dta", keep(3) nogen
save "$stata_sbp_input\snp_alleles_sbp.dta", replace

use "$stata_sbp_output\part_3b.dta", clear
merge 1:1 id_phe using "$stata_sbp_input\snp_alleles_sbp.dta", keep(1 3) nogen
*gen imputation = .
gen snp = ""
gen effect_allele = ""
gen eaf = .
gen outcome = ""
gen beta = .
gen se = .
gen variance = .
gen p = .
gen n = .
local i = 1
local outcomes = "qaly_hes"


        foreach outcome in `outcomes´ {

                qui regress `outcome´ rs* age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10

                foreach snp of varlist rs* {
                        local snpx = substr("`snp´",1,length("`snp´")-2)
                        *qui replace imputation = `imputation´ in `i´
```

```
                        qui replace snp = "`snpx´" in `i´
                        qui replace outcome = "`outcome´" in `i´
                        qui replace beta = _b[`snp´] if snp == "`snpx´" & outcome == "`outcome´"
                        qui replace se = _se[`snp´] if snp == "`snpx´" & outcome == "`outcome´"
                        qui sum `snp´
                        qui replace eaf = r(mean)/2 if snp == "`snpx´"
                        local effect_allele = upper(substr("`snp´",length("`snp´"),1))
                        qui replace effect_allele = "`effect_allele´" if snp == "`snpx´"
                        local i = `i´+1
                }

                *Ns
                qui sum `outcome´
                qui replace n = r(N) if outcome == "`outcome´"
        }

keep snp-n
keep if snp != ""
qui replace variance = se^2
qui replace p = 2*normal(-abs(beta/se))
save "$stata_sbp_result\Results_snp_qalys.dta", replace
use "$stata_sbp_result\Results_snp_qalys.dta", clear
keep if outcome == "qaly_hes"
save "$stata_sbp_result\Results_snp_qaly_hes.dta"
*******************************************************************************
*merge with allele frequncy data
*******************************************************************************
import delimited "$dx_data\ukb22828_sbp_allele_freq.afreq",clear
rename id snp
rename alt other_allele
merge 1:1 snp using "$stata_sbp_result\Results_snp_qaly_hes.dta"
drop chrom ref _merge* alt_freqs obs_ct
export delimited using "$sbp_snps\snp_qaly_hes.csv",replace
```

4.2.6 Data harmonisation

We now had the SNP-exposure and SNP-outcome association data. The next task was to harmonize these two datasets. For this, we continued working in the previous R environment. Data harmonization ensured that the effect alleles between the two datasets were properly aligned and removed any ambiguous SNPs. Ambiguous SNPs were those with A/T or C/G allele pairs, as they could create strand alignment issues due to their complementarity, making it difficult to determine the correct effect direction.

```
# we will continue working in R environment
# clumped_dbp_data_2 contains the SNP-SBP association

# Upload the SNP-outcome csv file: snp_qaly_hes.csv
qaly_hes_data<-read.csv("C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/sbp_snps/snp_qaly_hes.csv")

# rename the column names
colnames(qaly_hes_data)<-c("SNP", "other_allele.outcome", "effect_allele.outcome", "eaf.outcome", "outcome", "beta.outcome", "se.outcome", "variance.outcome", "pval.outcome", "samplesize.outcome")

#add outcome ID
qaly_hes_data$id.outcome = "ukb" # "ukb" added here just a reminder the SNP-outcome association is from UKB cohort.

#Harmonise the data
harmonise_data <- harmonise_data(
  exposure_dat = clumped_sbp_data_2,
  outcome_dat = qaly_hes_data
)

#save the file
write.csv(harmonise_data, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/sbp_snps/snp_sbp_qaly_hes_harmonised.csv", row.names = FALSE)

#select the SNP-exposure associatoin column for the next analysis
sbp_effect_list<-harmonise_data%>%select(SNP,effect_allele.exposure,beta.exposure)

#Filter ambiguous to exclude for the next analysis
sbp_snplist_exclude<-harmonise_data%>%filter(mr_keep == "FALSE")%>%select(SNP)

#save both files in a .txt format
write.table(sbp_effect_list, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/sbp/data/sbp_effect_list.txt",row.names = FALSE, col.names =FALSE, quote = FALSE,sep = " ")
write.table(sbp_snplist_exclude, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data
```

```
/sbp/data/sbp_snplist_exclude.txt", row.names = FALSE, col.names =FALSE
,quote = FALSE,sep = " ")
```

We have harmonised our data and created .txt files for the next analysis.

4.2.7 Calculating polygenic risk scores

Using the **sbp_effect_list.txt** file and excluding ambiguous SNPs based on the **sbp_snplist_exclude.txt** file, the **polygenic risk score (PRS)** was calculated. PRS is a weighted sum of risk alleles across many genetic variants (SNPs). In our study, it was calculated across 181 SNPs after excluding the ambiguous ones.

The two .txt files were uploaded to the UKB RAP platform, and PLINK2 was used via the Swiss Army Knife to calculate the PRS.

```
# Upload sbp_effect_list.txt and sbp_snplist_exclude.txt files to UKB RAP
# We will also use the merged plink files we created before
# Use the two .txt files and the merged plink flies to run the following code to
    calculate the PRS via Swiss Army Knife

plink2 --pfile ukb22828_c1_22_v3_sbp_merged --score sbp_effect_list.txt cols=+
    scoresums --exclude sbp_snplist_exclude.txt --out ukb22828_sbp_prs
```

We have now calculated the PRS for SNPs effect on SBP. Download the **ukb22828_sbp_prs.sscore** file to your local machine and save them to the `$dx_data_sbp` file path.

4.3 Step 3: Combining Phenotype and Genotype data

Now we will combine our phenotype data with PRS. The **part_3b.dta** contains the phenotype data for our cohor while the **ukb22828_sbp_prs.sscore** contains the PRS for each participants in the UK Biobank. The next stata line of codes will merge the two datasets. In addition prepare our data for the main and sensitivity analyses.

```
import delimited "$dx_data_sbp\ukb22828_sbp_prs.sscore", clear
 gen id_phe = iid // IID: Individual ID
 save "$dx_data_sbp\ukb22828_sbp_prs.sscore.dta", replace

use "$dx_data_sbp\ukb22828_sbp_prs.sscore.dta", clear
 merge 1:1 id_phe using "$stata_sbp_output\part_3b.dta"
 keep if _merge == 3
 drop _merge*
 rename score1_sum prs_sbp
 drop fid iid allele_ct named_allele_dosage_sum score1_avg

* Create a genotype array type variable for the next analysis
gen geno_array =.
        drop if  n_22000_0_0 ==.
        replace geno_array = 0 if n_22000_0_0 <0
        replace geno_array = 1 if n_22000_0_0 >0

*Estimate prs-free SBP for the next analysis
        su prs_sbp
        local mean = r(mean)
        reg phe_sbp_adj prs_sbp
        gen gf_sbp = phe_sbp_adj - _b[prs_sbp]*prs_sbp + _b[prs_sbp]*`mean´, a(phe_sbp_adj)
*50 xtiles of prs-free SBP
        qui xtile cat_gf_sbp = gf_sbp, nq(50)

        save "$stata_sbp_output\part_4a.dta", replac
```

We have prepared our dataset for the main and sensitivity analyses.

4.4 Step 4: Main analysis

We estimated the causal association of SBP with QALY using **MR technique**[14, 11, 6]. Specifically, **two-stage least square (2SLS)** run by regressing the exposure variable (SBP) on the PRS for SBP at the first stage followed by regressing the outcome variable (QALYs) on the predicted SBP from the first stage. Age, sex, UK Biobank assessment centre, genotyping array, and the first 10 genetic principal components for population stratification were used as covariates in the model. F-statistics was used to assess for weak instrument bias. Outputs of the model interpreted as change in QALYs caused by a 1 mmHg increase in SBP over an average year of follow-up. For convenience, the final output was presented as percentage change in QALY per 10 mmHg increase in SBP.

We also performed **multivariable linear regression model** fitting the QALY outcome on SBP exposure data adjusting for age, sex, assessment centre, genotyping array and the first 10 genetic principal components. Then compared the estimate from 2SLS with the multivariable linear regression model and test for presence of **endogeneity (Hausman test)**[15, 16, 17]. A low p value in the Hausman test indicated difference in the estimates between the 2SLS and multivariable linear regression model.

```
use "$stata_sbp_output\part_4a.dta", clear
*******************************************************************************
*Main analysis
*******************************************************************************
*Create table
gen outcome = ""
gen type = ""
*gen imputation = .
gen n = .
gen beta = .
gen variance = .
gen se = .
gen double p = .
gen double p_endog = .
gen f_stat = .

local x = 1

        foreach var in qaly_hes {
                dis "Outcome = `var´"

                *MR analysis
                ivreg2 `var´ (phe_sbp_adj = prs_sbp) age i.sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 p
> c10 i.centre i.geno_array, robust endog(phe_sbp_adj)

                matrix a = e(b)
                matrix b = e(V)
                local beta = a[1,1]
                local variance = b[1,1]

                local n = e(N)
                local f_stat = e(widstat)
                local p_endog = e(estatp)

                replace outcome = "`var´" in `x´
                replace type = "Main Analysis MR" in `x´
                foreach z in beta variance n p_endog f_stat {
                        replace `z´ = ``z´´ in `x´
                }

                local x = `x´ + 1

                *Linear regression
                reg `var´ phe_sbp_adj age i.sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre i.
```

```
> geno_array

                matrix a = e(b)
                matrix b = e(V)
                local reg_beta = a[1,1]
                local reg_variance = b[1,1]

                local reg_n = e(N)

                *Linear regression estimates
                replace outcome = "`var´" in `x´
                replace type = "Multivariable Adjusted" in `x´
                *qui replace imputation = `j´ in `x´
                foreach z in beta variance n {
                        replace `z´ = `reg_`z´´ in `x´
                }

                local x = `x´ + 1

        }

keep outcome-f_stat
keep if outcome != ""
replace outcome = "QALYs per year (with 240 comorbidities)" if outcome == "qaly_hes"
qui replace se = sqrt(var)
qui replace p = 2*normal(-abs(beta/se))
sort outcome type
*save "Result_sbp_exclusive_table.dta", replace
 save "$stata_sbp_result\Result_sbp_table.dta", replace

*******************************************************************************
*SBP variations explained by PRS
*******************************************************************************
clear
set obs 1
gen r2 = .
        preserve
        use "$stata_sbp_output\part_4a.dta", clear

        corr phe_sbp_adj prs_sbp
        local r2 = r(rho)^2
        restore
        replace r2 = `r2´ in 1
su r2, d
save "$stata_sbp_result\R2_value_sbp.dta", replace
```

4.5 Step 5: Sensitivity analyses

To test the robustness of the main analyses outcome, a number of sensitivity analyses were perfomed.

- **Untreated population**: Rerun the main analysis for the cohort without antihhpertensive medications
- **Two-sample MR**: Run a number of summary level MR analyses
- **Sub-group analysis**: Stratified by age, sex, PRS-free SBP
- **Non-linear MR**
- **EQ-5D index from UK Biobank survey**

4.5.1 Untreated populaton

We repeated the main analysis excluding participants prescribed with antihypertensive medications.

```
use "$stata_sbp_output\part_4a.dta", clear
gen sbp_treat =.
replace sbp_treat = 1 if n_6153_0_0 == 2 | n_6153_0_1 == 2 | n_6153_0_2 == 2 | n_6153_0_3 == 2 | n_6
> 177_0_0 == 2 | n_6177_0_1 == 2 | n_6177_0_2 == 2
replace sbp_treat = 2 if sbp_treat ==.
label define treatlbl 1 "medication" 2 "no medication"
label values sbp_treat treatlbl
keep if sbp_treat == 2
*Create table
gen outcome = ""
gen type = ""
gen n = .
gen beta = .
gen variance = .
gen se = .
gen double p = .
gen double p_endog = .
gen f_stat = .
local x = 1
*local outcomes = "cost qaly qaly_cost_20k"

        foreach var in qaly_hes {
                dis "Outcome = `var´"

                *MR analysis
                ivreg2 `var´ (phe_sbp_adj = prs_sbp) age i.sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 p
> c10 i.centre i.geno_array, robust endog(phe_sbp_adj)

                matrix a = e(b)
                matrix b = e(V)
                local beta = a[1,1]
                local variance = b[1,1]

                local n = e(N)
                local f_stat = e(widstat)
                local p_endog = e(estatp)

                replace outcome = "`var´" in `x´
                replace type = "Main Analysis MR" in `x´
```

```
                foreach z in beta variance n p_endog f_stat {
                        replace `z´ = ``z´´ in `x´
                }

                local x = `x´ + 1

                *Linear regression
                reg `var´ phe_sbp_adj age i.sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre i.
> geno_array

                matrix a = e(b)
                matrix b = e(V)
                local reg_beta = a[1,1]
                local reg_variance = b[1,1]

                local reg_n = e(N)

                *Linear regression estimates
                replace outcome = "`var´" in `x´
                replace type = "Multivariable Adjusted" in `x´
                foreach z in beta variance n {
                        replace `z´ = `reg_`z´´ in `x´
                }

                local x = `x´ + 1

        }

keep outcome-f_stat
keep if outcome != ""
replace outcome = "QALYs per year (with 240 comorbidities)" if outcome == "qaly_hes"
qui replace se = sqrt(var)
qui replace p = 2*normal(-abs(beta/se))
sort outcome type
save "$stata_sbp_result\Result_sbp_exclusive_table_no_medication.dta", replace
```

4.5.2 Two-sample MR

To test the robustness of the main analysis estimate, 2SLS, we also performed a two-sample MR (i.e., summary level MR)[11]. The first sample data was sourced from the ICBP and replication meta-analyses which included the effect of the 181 SNPs on SBP[13]. For the second sample, we estimated the effect of the 181 SNPs on QALYs using the UK Biobank population by regressing the QALYs on the 181 SNPs allele dosage (the number of effect allele each participant has) adjusting for age, sex and the first 10 genetic principal components[8, 18]. Inverse variance weighting (IVW), MR Egger, weighted median and weighted mode analyses were performed using the summary level data[11].

- IVW
  - IVW method combines the individual ratio estimates from multiple genetic instruments (i.e., SNPs) into a single overall estimate, using a weighted average approach, where the weights are the inverse of the variance of the SNP-outcome association[11].
  - Presence of heterogeneity among the instruments was tested using Cochran's Q statistics, and a statistically significant p-value indicates evidence of heterogeneity[19].
  - To quantify the degree of heterogeneity, I^2 was also reported, with 0% indicating no observed heterogeneity, $0\% < I^2 \leq 25\%$ representing low heterogeneity, $25\% < I^2 \leq 50\%$ indicating moderate heterogeneity, $50\% < I^2 \leq 75\%$ denoting substantial heterogeneity, and $I^2 > 75\%$ reflecting considerable heterogeneity[19].
- MR Egger
  - The MR Egger method used to test for and account for directional pleiotropy, a potential source of bias in causal estimates[11]. It's an extension of the IVW approach, with additional flexibility to model horizontal (i.e., directional) pleiotropy, which occurs when genetic variants (SNPs) influence the outcome through pathways other than the exposure of interest[11]. The main difference between MR Egger and IVW is that MR Egger includes an intercept term in the regression model. This allows for testing and adjusting for directional pleiotropy[11].
- Weighted median
  - The weighted median method is useful when some of the genetic variants (SNPs) used as instrumental variables may be invalid due to pleiotropy or other issues[11]. This method gives consistent causal estimate if at least 50
- Weighted mode
  - The weighted mode method estimates the causal effect by finding the most common (modal) value of SNP-specific causal estimates, with more weight given to SNPs that are more precise[11]. This method is robust to invalid instruments and can produce valid estimates even when most instruments are invalid[11].

```
*******************************************************************************
*Prepare the data
*******************************************************************************
*import the harmonised data
```

```
import delimited "$sbp_snps\snp_sbp_qaly_harmonised.csv", clear
        rename betaexposure beta_exposure
        rename seexposure   se_exposure
        rename betaoutcome beta
        rename seoutcome se
        rename samplesizeoutcome n
*Make all the exposure betas positive
        replace beta = -beta if beta_exposure < 0
        replace beta_exposure = -beta_exposure if beta_exposure < 0
        count if beta_exposure <0
        sort outcome snp
        keep snp beta_exposure se_exposure beta se outcome pvaloutcome pvalexposure n
save "$stata_sbp_output\MR_data_sbp.dta",replace
*******************************************************************************
*Run the Two sample MR analyses
*******************************************************************************
use "$stata_sbp_output\MR_data_sbp.dta", clear
gen out = ""
gen ivw = .
gen ivw_se = .
gen ivw_p = .
gen egger_slope = .
gen egger_slope_se = .
gen egger_slope_p = .
gen egger_cons = .
gen egger_cons_se = .
gen egger_cons_p = .
gen double heterogeneity_p = .
gen median = .
gen median_se = .
gen median_p = .
gen mode = .
gen mode_se = .
gen mode_p = .
qui levelsof outcome, local(outcome)
local i = 1
foreach out in `outcome´ {
        *MR robust takes the outcome first
        replace out = "`out´" in `i´
        mregger beta beta_exposure [aw=1/(se^2)] if outcome == "`out´", ivw heterogi
        if !_rc {
                replace heterogeneity_p = r(pval) in `i´
        }
        else {
                mregger beta beta_exposure [aw=1/(se^2)] if outcome == "`out´", ivw
        }
        replace ivw = _b[beta_exposure] in `i´
        replace ivw_se = _se[beta_exposure] in `i´

        mregger beta beta_exposure [aw=1/(se^2)] if outcome == "`out´"
        replace egger_slope = _b[slope] in `i´
        replace egger_slope_se = _se[slope] in `i´
        replace egger_cons = _b[_cons] in `i´
        replace egger_cons_se = _se[_cons] in `i´
        mrmedian beta se beta_exposure se_exposure if outcome == "`out´"
        replace median = _b[beta] in `i´
        replace median_se = _se[beta] in `i´
        mrmodal beta se beta_exposure se_exposure if outcome == "`out´"
        replace mode = _b[beta] in `i´
        replace mode_se = _se[beta] in `i´

        local i = `i´ + 1
}

foreach var of varlist ivw egger_slope egger_cons median mode {
```

```
        replace `var´_p = 2*normal(-abs(`var´/`var´_se))
}
keep out-mode_p
keep if out != ""
rename out outcome
*Make things look better
replace outcome = "QALYs per year (HES only)" if outcome == "qaly_hes"
sort outcome
save "$stata_sbp_result\Results_table_sensitivity_sbp.dta", replace
```

4.5.3 Sub-group analysis

Sub-group analysis was performed by rerunning the main MR stratified by age categories (< 50 years, 50-54 years, 55-59 years, 60-64 years, and 65+ years), sex (Male and Female), and PRS-free SBP categories (<120 mmHg, 120-139 mmHg and 140+ mmHg). PRS-free SBP was estimated by first regressing observed SBP (i.e., mean SBP) on the PRS for SBP and then predicting each participant's SBP as if they had the average PRS for SBP[1].

```
*******************************************************************************
*Sex-, SBP category- and Age-Specific Analyses
*******************************************************************************
{
use "$stata_sbp_output\part_4a.dta", clear
*Mark participants depending on sex, age and SBP level on genetic-free SBP
gen all = 1
*replace age = age+38
gen age_cat1 = 1 if age < 50
gen age_cat2 = 1 if age < 55 & age >= 50
gen age_cat3 = 1 if age < 60 & age >= 55
gen age_cat4 = 1 if age < 65 & age >= 60
gen age_cat5 = 1 if age >=65
gen sbp_cat1 = 1 if gf_sbp <120
gen sbp_cat2 = 1 if gf_sbp >= 120 & gf_sbp <140
gen sbp_cat3 = 1 if gf_sbp >= 140
*Create table
gen outcome = ""
gen type = ""
*gen imputation = .
gen sensitivity = ""
foreach sex in all male female {
  gen n_`sex´ = .
  gen beta_`sex´ = .
  gen variance_`sex´ = .
  gen se_`sex´ = .
  gen double p_`sex´ = .
  gen double p_endog_`sex´ = .
  gen f_stat_`sex´ = .
}
foreach sex in all male female {
local x = 1
foreach sens in all age_cat1 age_cat2 age_cat3 age_cat4 age_cat5 sbp_cat1 sbp_cat2 sbp_cat3 {
foreach var in qaly_hes {
if "`sex´" == "all" {
        qui ivreg2 `var´ (phe_sbp_adj = prs_sbp) age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.
> centre geno_array if `sens´ == 1, robust endog(phe_sbp_adj)
}
else if "`sex´" == "female" {
        qui ivreg2 `var´ (phe_sbp_adj = prs_sbp) age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.
> centre geno_array if `sens´ == 1 & sex == 0, robust endog(phe_sbp_adj)
}
else {
        qui ivreg2 `var´ (phe_sbp_adj = prs_sbp) age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.
> centre geno_array if `sens´ == 1 & sex == 1, robust endog(phe_sbp_adj)
}
matrix a = e(b)
matrix b = e(V)
local beta = a[1,1]
local variance = b[1,1]
local n = e(N)
local f_stat = e(widstat)
local p_endog = e(estatp)
*MR estimates
```

```
qui replace outcome = "`var´" in `x´
qui replace type = "Main Analysis MR" in `x´
qui replace sensitivity = "`sens´" in `x´

foreach z in beta variance n p_endog f_stat {
qui replace `z´_`sex´ = ``z´´ in `x´
}

local x = `x´ + 1

*Linear regression
if "`sex´" == "all" {
        qui reg `var´ phe_sbp_adj age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre geno_arr
> ay if `sens´ == 1
}
else if "`sex´" == "female" {
        qui reg `var´ phe_sbp_adj age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre geno_arr
> ay if `sens´ == 1 & sex == 0
}
else {
        qui reg `var´ phe_sbp_adj age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre geno_arr
> ay if `sens´ == 1 & sex == 1
}

matrix a = e(b)
matrix b = e(V)
local beta = a[1,1]
local variance = b[1,1]

local n = e(N)

*Linear regression estimates
qui replace outcome = "`var´" in `x´
qui replace type = "Multivariable Adjusted" in `x´
qui replace sensitivity = "`sens´" in `x´

foreach z in beta variance n {
qui replace `z´_`sex´ = ``z´´ in `x´
}

local x = `x´ + 1
}
}
}

keep outcome-f_stat_female
drop if outcome == ""
sort outcome type sensitivity

foreach sex in all male female {
qui replace se_`sex´ = sqrt(variance_`sex´)
qui replace p_`sex´ = 2*normal(-abs(beta_`sex´/se_`sex´))
}

*Altman-Bland/Fisher tests
*Male-female
gen double p_sex = 2*normal(-abs((beta_female-beta_male)/sqrt(se_female^2+se_male^2)))

save "$stata_sbp_result\Results_sbp_subgroup.dta", replace

}
```

4.5.4 Non-linear MR

Non-linear MR was performed by running the main MR within fifty quantiles of PRS-free SBP, estimating quantile specific **local average causal effects**. These local average estimates were used in the **variance weighted least squares (VWLS)** models to determine whether there was a stable or incremental change in the effect of SBP on QALYs as SBP increased. Both linear and cubic models (with respect to the mean PRS-free SBP in each quantile) used to describe the shape of the effect of the increase in SBP over the range of PRS-free SBP values. We followed the next steps to execute the non-linear MR analysis.

4.5.4.1 Rerun the main MR within the fifty quantiles of PRS-free SBP

```
use "$stata_sbp_output\part_4a.dta", clear
gen outcome = ""
gen type = ""
gen gf_sbp_cat = .
foreach sex in all male female {
local x = 1
gen n_`sex´ = .
gen beta_`sex´ = .
gen variance_`sex´ = .
gen se_`sex´ = .
gen double p_`sex´ = .
foreach var in qaly_hes {
forvalues k = 1/50 {
        display "working on `k´ of 50"
if "`sex´" == "all" {
 qui ivreg2 `var´ (phe_sbp_adj = prs_sbp) age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre
> geno_array if cat_gf_sbp == `k´,  robust
}
else if "`sex´" == "female" {

 qui ivreg2 `var´ (phe_sbp_adj = prs_sbp) age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre
> geno_array if  sex ==0 & cat_gf_sbp == `k´, robust
}
else {
 qui ivreg2 `var´ (phe_sbp_adj = prs_sbp) age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre
> geno_array if sex == 1 & cat_gf_sbp == `k´, robust
}
matrix a = e(b)
matrix b = e(V)
local beta = a[1,1]
local variance = b[1,1]
local se = sqrt(b[1,1])
local p = 2*normal(-abs(`beta´/`se´))
local n = e(N)
*MR estimates
qui replace outcome = "`var´" in `x´
qui replace type = "Main Analysis MR" in `x´
qui replace gf_sbp_cat = `k´ in `x´
foreach z in beta se variance p n {
qui replace `z´_`sex´ = ``z´´ in `x´
}
local x = `x´ + 1
*Linear regression
if "`sex´" == "all" {
qui reg `var´ phe_sbp_adj age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre geno_array if ca
> t_gf_sbp == `k´
}
```

```
else if "`sex´" == "female" {
qui reg `var´ phe_sbp_adj age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre geno_array if se
> x == 0 & cat_gf_sbp == `k´
}
else {
qui reg `var´ phe_sbp_adj age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre geno_array if se
> x == 1 & cat_gf_sbp == `k´
}
matrix a = e(b)
matrix b = e(V)
local beta = a[1,1]
local variance = b[1,1]
local se = sqrt(b[1,1])
local p = 2*normal(-abs(`beta´/`se´))
local n = e(N)
*Linear regression estimates
qui replace outcome = "`var´" in `x´
qui replace type = "Multivariable Adjusted" in `x´
qui replace gf_sbp_cat = `k´ in `x´
*qui replace imputation = `j´ in `x´
foreach z in beta se variance p n {
qui replace `z´_`sex´ = ``z´´ in `x´
}
local x = `x´ + 1
}
}
}
keep outcome-p_female
drop if outcome == ""
*sort outcome imputation type gf_sbp_cat
sort outcome type gf_sbp_cat
*Altman-Bland/Fisher tests
*Male-female
gen double p_sex = 2*normal(-abs((beta_female-beta_male)/sqrt(se_female^2+se_male^2)))
save "$stata_sbp_result\Results_sensitivity_sbp_nl_50q.dta", replace
```

4.5.4.2 Sub-group analysis mainly stratified by PRS-free SBP

For each PRS-free SBP category, we further stratified by age-group and sex.

```
{
use "$stata_sbp_output\part_4a.dta", clear
su gf_sbp
gen sbp_cat =.
replace sbp_cat = 1 if gf_sbp <120
replace sbp_cat = 2 if gf_sbp >= 120 & gf_sbp <140
replace sbp_cat = 3 if gf_sbp >= 140
label define sbplbl 1 "<120 mmHg" 2 "120-139 mmHg" 3 "140+ mmHg"
label values sbp_cat sbplbl
*keep if sbp_cat == 1
*Mark participants depending on sex, age and overweight on genetic-free SBP
gen all = 1
gen age_cat1 = 1 if age < 50
gen age_cat2 = 1 if age < 55 & age >= 50
gen age_cat3 = 1 if age < 60 & age >= 55
gen age_cat4 = 1 if age < 65 & age >= 60
gen age_cat5 = 1 if age >=65
gen sbp_cat1 = 1 if gf_sbp <120
gen sbp_cat2 = 1 if gf_sbp >=120 & gf_sbp <=139
gen sbp_cat3 = 1 if gf_sbp >=140
*Create table
```

```
gen outcome = ""
gen type = ""
gen sbp_category = ""
*gen imputation = .
gen sensitivity = ""
foreach sex in all male female {
  gen n_`sex´ = .
  gen beta_`sex´ = .
  gen variance_`sex´ = .
  gen se_`sex´ = .
  gen double p_`sex´ = .
  gen double p_endog_`sex´ = .
  gen f_stat_`sex´ = .
}
foreach sex in all male female {
local x = 1
levelsof sbp_cat, local(sbp)
foreach g in `sbp´ {
foreach sens in all age_cat1 age_cat2 age_cat3 age_cat4 age_cat5 {
foreach var in qaly_hes {
        display "Group = `g´, Sex = `sex´, Sensitivity `sens´, and Outcome = `var´"
if "`sex´" == "all" {
        qui ivreg2 `var´ (phe_sbp_adj = prs_sbp) age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.
> centre geno_array if `sens´ == 1 &
        sbp_cat == `g´, robust endog(phe_sbp_adj)
}
else if "`sex´" == "female" {
   qui ivreg2 `var´ (phe_sbp_adj = prs_sbp) age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centr
> e geno_array if `sens´ == 1 &
   sex == 0  & sbp_cat == `g´, robust endog(phe_sbp_adj)
}
else {
  qui ivreg2 `var´ (phe_sbp_adj = prs_sbp) age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre
>  geno_array if `sens´ == 1 &
  sex == 1  & sbp_cat == `g´, robust endog(phe_sbp_adj)
}
matrix a = e(b)
matrix b = e(V)
local beta = a[1,1]
local variance = b[1,1]
local n = e(N)
local f_stat = e(widstat)
local p_endog = e(estatp)
*MR estimates
qui replace outcome = "`var´" in `x´
qui replace type = "Main Analysis MR" in `x´
qui replace sbp_category = "`g´" in `x´
qui replace sensitivity = "`sens´" in `x´
foreach z in beta variance n p_endog f_stat {
qui replace `z´_`sex´ = ``z´´ in `x´
}
local x = `x´ + 1
*Linear regression
if "`sex´" == "all" {
 qui reg `var´ phe_sbp_adj age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre geno_array if `
> sens´ == 1 &
 sbp_cat == `g´
}
else if "`sex´" == "female" {
qui reg `var´ phe_sbp_adj age pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre geno_array if `sens´
>  == 1 &
sex == 0  & sbp_cat == `g´
}
else {
```

```
qui reg `var´ phe_sbp_adj age pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre geno_array if `sens´
>  == 1 &
sex == 1  & sbp_cat == `g´
}

matrix a = e(b)
matrix b = e(V)
local beta = a[1,1]
local variance = b[1,1]

local n = e(N)

*Linear regression estimates
qui replace outcome = "`var´" in `x´
qui replace type = "Multivariable Adjusted" in `x´
qui replace sbp_category = "`g´" in `x´
qui replace sensitivity = "`sens´" in `x´

foreach z in beta variance n {
qui replace `z´_`sex´ = ``z´´ in `x´
}

local x = `x´ + 1

}
}
}
}

keep outcome-f_stat_female
drop if outcome == ""
sort outcome type sbp_category sensitivity

foreach sex in all male female {
qui replace se_`sex´ = sqrt(variance_`sex´)
qui replace p_`sex´ = 2*normal(-abs(beta_`sex´/se_`sex´))
}

save "$stata_sbp_result\Results_sbp_subgroup_1.dta", replace
}
```

4.5.4.3 Select observations for all participants in each SBP

```
use "$stata_sbp_result\Results_sbp_subgroup_1.dta", clear

replace sbp_category = "Normal" if sbp_category == "1"
replace sbp_category = "Pre_hypertension" if sbp_category == "2"
replace sbp_category = "Hypertension" if sbp_category == "3"

keep if sensitivity == "all"
drop if type == "Multivariable Adjusted"
keep outcome beta* sbp_category

replace sbp_category = lower(sbp_category)
foreach var of varlist beta* {
        rename `var´ `var´_
}

reshape wide beta*, i(outcome) j(sbp_category) string

save "$stata_sbp_result\Sensitivity_analysis_graph_data.dta", replace
```

4.5.4.4 Estimate the mean SBP for each quantile

```
use "$stata_sbp_output\Part_4a.dta",clear
su gf_sbp
gen touse = 1
gen gf_sbp_cat = _n in 1/50
foreach sex in all male female {
        gen gf_sbp_mean_`sex´ = .
        replace touse = 1
        if "`sex´" == "male" {
                replace touse = . if sex == 0
```

```
        }
        if "`sex´" == "female" {
                replace touse = . if sex == 1
        }
        forvalues i = 1/50 {
                qui su gf_sbp if cat_gf_sbp == `i´ & touse == 1
                qui replace gf_sbp_mean_`sex´ = r(mean) in `i´
        }
}
keep if gf_sbp_cat != .
keep gf_sbp_cat gf_sbp_mean*
        histogram gf_sbp_mean_all, bin(20) normal percent color(red) xlabel(100(5)200) title("SBP Di
> stribution")
        count if gf_sbp_mean_all >180

save "$stata_sbp_result\Sensitivity_analysis_PRS_free_SBP_data.dta", replace
```

4.5.4.5 Merge the previous datasets for VWLS analyses and plots

We run the analyses here

```
use "$stata_sbp_result\Results_sensitivity_sbp_nl_50q.dta", clear
merge m:1 gf_sbp_cat using "$stata_sbp_result\Sensitivity_analysis_PRS_free_SBP_data.dta", nogen
sort outcome type gf_sbp_cat
merge m:1 outcome using "$stata_sbp_result\Sensitivity_analysis_graph_data.dta", nogen

replace outcome = "QALYs per year" if outcome == "qaly_hes"
*Meta-regress using linear regression
encode outcome, gen(outcome2)
encode type, gen(type2)
foreach sex in all male female {
        gen l95_`sex´ = beta_`sex´ - se_`sex´*1.96
        gen u95_`sex´ = beta_`sex´ + se_`sex´*1.96

        gen gf_sbp_mean_`sex´_2 = gf_sbp_mean_`sex´^2
        gen gf_sbp_mean_`sex´_3 = gf_sbp_mean_`sex´^3

        gen b0_`sex´ = .
        gen b1_`sex´ = .
        gen b2_`sex´ = .
        gen b3_`sex´ = .

        gen se0_`sex´ = .
        gen se1_`sex´ = .
        gen se2_`sex´ = .
        gen se3_`sex´ = .
}
local obs = c(N)+1
local obs2 = c(N) + 10000
set obs `obs2´
qui gen n = _n
gen test = 1 in `obs´/`obs2´
foreach sex in all male female {
        qui replace gf_sbp_mean_`sex´ = 100 + (n-`obs´)/100 in `obs´/`obs2´
        qui replace gf_sbp_mean_`sex´_2 = (100 + (n-`obs´)/100)^2 in `obs´/`obs2´
        qui replace gf_sbp_mean_`sex´_3 = (100 + (n-`obs´)/100)^3 in `obs´/`obs2´
}

forvalues outcome = 1/1 {
        qui su n if outcome2 == `outcome´
        local outcome_label = outcome[r(min)]

        foreach sex in all male female {

                local xtitle = "SBP (mmHg)"
```

```
                if "`sex´" == "male" {
                        local xtitle = "SBP (mmHg) (male)"
                }
                if "`sex´" == "female" {
                        local xtitle = "SBP (mmHg) (female)"
                }

                vwls beta_`sex´ gf_sbp_mean_`sex´ gf_sbp_mean_`sex´_2 gf_sbp_mean_`sex´_3 if outcome
> 2 == `outcome´ & type2 == 1, sd(se_`sex´)
                local b1 = _b[gf_sbp_mean_`sex´]
                local b2 = _b[gf_sbp_mean_`sex´_2]
                local b3 = _b[gf_sbp_mean_`sex´_3]
                local cons = _b[_cons]

                foreach x in normal pre_hypertension hypertension {
                        qui su n if outcome2 == `outcome´
                        local `x´ = beta_`sex´_`x´[r(min)]
                }
                scatter beta_`sex´ gf_sbp_mean_`sex´ if outcome2 == `outcome´ & type == "Main Analys
> is MR" &  gf_sbp_mean_`sex´ <=160 || rcap l95_`sex´ u95_`sex´ gf_sbp_mean_`sex´ if outcome2 == `ou
> tcome´ & type == "Main Analysis MR"  &  gf_sbp_mean_`sex´ <=160 || ///
                function y = `cons´ + `b1´*x + `b2´*x^2 + `b3´*x^3, range(100 160) || function y = `
> normal´, range (100 119.9) lcolor(navy) lpattern(dash) || ///
                function y = `pre_hypertension´, range(120 139.9) lcolor(navy) lpattern(dash) || fun
> ction y = `hypertension´, range(140 160) lcolor(navy) lpattern(dash) ///
                xtitle("`xtitle´") ytitle("Effect of a unit increase in SBP on `outcome_label´", siz
> e(small)) legend(off) xscale(range(100 160))
        graph export "$plot_png\`outcome_label´ [`sex´].png", as(png) width(1200) replace

                        *Just trend line and 95% CI
                *Maybe create a small dataset with fixed values to remove

                predict x if test == 1
                predict x_se if test == 1, stdp
                gen x_l95 = x - 1.96*x_se
                gen x_u95 = x + 1.96*x_se

                twoway rarea x_l95 x_u95 gf_sbp_mean_`sex´ if test == 1 & gf_sbp_mean_`sex´ <=160, l
> color(green%50) color(green%50) || line x gf_sbp_mean_`sex´ if test == 1 & gf_sbp_mean_`sex´ <=160
> , lcolor(dkgreen) ///
                legend(off) xtitle("`xtitle´") ytitle("Effect of a unit increase in SBP on `outcome_
> label´", size(small)) ///
                plotregion(fcolor(white)) graphregion(fcolor(white)) xline(120,lcolor(maroon%80) lpa
> ttern(-)) xline(140,lcolor(maroon%80) lpattern(-)) yline(0,lcolor(gs4))

                graph export "$plot_png\`outcome_label´ [`sex´] v2.png", as(png) width(1200) replace

                drop x-x_u95
        }
}
*Figures
su n if outcome2 == 1
local outcome_label = outcome[r(min)]
local xtitle = "SBP (mmHg)"

vwls beta_all gf_sbp_mean_all gf_sbp_mean_all_2 gf_sbp_mean_all_3 if outcome2 == 1 & type2 == 1, sd(
> se_all)
local b1 = _b[gf_sbp_mean_all]
local b2 = _b[gf_sbp_mean_all_2]
local b3 = _b[gf_sbp_mean_all_3]
local cons = _b[_cons]
predict x if test == 1
predict x_se if test == 1, stdp
gen x_l95 = x - 1.96*x_se
gen x_u95 = x + 1.96*x_se
twoway rarea x_l95 x_u95 gf_sbp_mean_all if test == 1 & gf_sbp_mean_all<=160, lcolor(green%50) color
```

```
> (green%50) || line x gf_sbp_mean_all if test == 1 & gf_sbp_mean_all<=160, lcolor(dkgreen) ///
legend(off) xtitle("`xtitle´") ytitle("Effect of a unit increase in SBP on `outcome_label´" "{&uarr}
>  SBP leads to {&darr} QALYs {&uarr} SBP leads to {&uarr} QALYs", size(small)) ///
plotregion(fcolor(white)) graphregion(fcolor(white)) xline(120,lcolor(maroon%80) lpattern(-)) xline(
> 140,lcolor(maroon%80) lpattern(-)) yline(0,lcolor(gs4)) ///
xscale(range (100 160))
graph export "$plot_png\nl_MR.png", as(png) width(1200) replace
drop x-x_u95

*Meta-regression
forvalues outcome = 1/2 {
        forvalues type = 1/2 {
                foreach sex in all male female {
                        qui vwls beta_`sex´ gf_sbp_mean_`sex´ gf_sbp_mean_`sex´_2 gf_sbp_mean_`sex´_
> 3 if outcome2 == `outcome´ & type2 == `type´, sd(se_`sex´)

                        qui replace b1_`sex´ = _b[gf_sbp_mean_`sex´] if outcome2 == `outcome´ & type
> 2 == `type´
                        qui replace b2_`sex´ = _b[gf_sbp_mean_`sex´_2] if outcome2 == `outcome´ & ty
> pe2 == `type´
                        qui replace b3_`sex´ = _b[gf_sbp_mean_`sex´_3] if outcome2 == `outcome´ & ty
> pe2 == `type´

                        qui replace se1_`sex´ = _se[gf_sbp_mean_`sex´] if outcome2 == `outcome´ & ty
> pe2 == `type´
                        qui replace se2_`sex´ = _se[gf_sbp_mean_`sex´_2] if outcome2 == `outcome´ &
> type2 == `type´
                        qui replace se3_`sex´ = _se[gf_sbp_mean_`sex´_3] if outcome2 == `outcome´ &
> type2 == `type´

                        qui vwls beta_`sex´ gf_sbp_mean_`sex´ if outcome2 == `outcome´ & type2 == `t
> ype´, sd(se_`sex´)

                        qui replace b0_`sex´ = _b[gf_sbp_mean_`sex´] if outcome2 == `outcome´ & type
> 2 == `type´
                        qui replace se0_`sex´ = _se[gf_sbp_mean_`sex´] if outcome2 == `outcome´ & ty
> pe2 == `type´
                }
        }
}
keep outcome type b0* b1* b2* b3* se0* se1* se2* se3*
duplicates drop
foreach sex in all male female {
        forvalues i = 0/3 {
                qui gen p`i´_`sex´ = 2*normal(-abs(b`i´_`sex´/se`i´_`sex´))
        }
}
order outcome type *0* *_all *_male *_female
save "$stata_sbp_result\Results_table_sensitivity_Stata_vwls_raw.dta", replace
keep outcome type *0*
foreach x in b se p {
        foreach sex in all male female {
                rename `x´0_`sex´ `x´1_`sex´
        }
}
gen model = "Linear"
save "$stata_sbp_result\Results_table_sensitivity_Stata_vwls_append.dta", replace
use "$stata_sbp_result\Results_table_sensitivity_Stata_vwls_append.dta", clear
append using "$stata_sbp_result\Results_table_sensitivity_Stata_vwls_raw.dta"
drop *0*
replace model = "Cubic" if model == ""
order b1* se1* p1* b2* se2* p2* b3* se3* p3*
order outcome type model *_all *_male *_female
save "$stata_sbp_result\Results_table_sensitivity_Stata_vwls.dta", replace
```

4.5.5 EQ-5D index from UK Biobank survey

We performed an additional sensitivity analysis using the EQ-5D-5L survey data from the UK Biobank[20, 21] to assess whether the direction of the effect observed in the main analysis would be similar when using direct EQ-5D-5L data from UK Biobank. Web-based EQ-5D-5L questionnaires were administered to the UK Biobank participants as part of the chronic pain (administered in 2019–20) and mental well-being (administered in 2022–23) surveys[20, 21]. There were 167,111 participants responded to the chronic pain survey[20] and 169,537 participants responded to the mental well-being survey[21]. We calculated the EQ-5D index using the UK tariffs (i.e., value set) for each survey[10]. Once the EQ-5D-indexes were calculated, we took the average of both EQ-5D-indexes for participants who had EQ-5D data for both surveys (124,830). The remaining participants had EQ-5D data for either chronic pain survey (42,281) or mental well-being survey (44,707); hence, the average EQ-5D index was not calculated. In total, there were 211,818 participants with EQ-5D index data from the combined survey. Of these participants, 128,635 had met the initial inclusion criteria and were included in this sensitivity analysis.

We rerun 2SLS analysis by regressing the SBP trait on the genetic risk scores for SBP on the first stage following by fitting the EQ-5D index on the predicted SBP values from the first stage. We adjusted for age, sex, UK Biobank assessment centre, genotyping array, and the first 10 genetic principal components for population stratification. Outputs from the 2SLS model were presented as the effect of 10 mmHg increase in SBP on percentage change in EQ-5D index.

```
*UKB Utilities
use "$stata_sbp_output\part_4a.dta", clear

gen ukb_utility =.
replace ukb_utility = (EQindex_1 + EQindex)/2 if EQindex !=. & EQindex_1 !=.
replace ukb_utility = EQindex if EQindex !=. & EQindex_1 ==.
replace ukb_utility = EQindex_1 if EQindex ==. & EQindex_1 !=.

*Create table
gen outcome = ""
gen type = ""

*gen imputation = .
gen n = .
gen beta = .
gen variance = .
gen se = .
gen double p = .
gen double p_endog = .
gen f_stat = .

*Number of imputations
*local m = 100

local x = 1

*local outcomes = "cost qaly qaly_cost_20k"


        foreach var in ukb_utility {
                dis "Outcome = `var´"

                *MR analysis
                ivreg2 `var´ (phe_sbp_adj = prs_sbp) age i.sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 p
> c10 i.centre i.geno_array,   robust endog(phe_sbp_adj)

                matrix a = e(b)
                matrix b = e(V)
                local beta = a[1,1]
                local variance = b[1,1]

                local n = e(N)
```

```
                local f_stat = e(widstat)
                local p_endog = e(estatp)

                replace outcome = "`var´" in `x´
                replace type = "Main Analysis MR" in `x´
                *qui replace imputation = `j´ in `x´
                foreach z in beta variance n p_endog f_stat {
                        replace `z´ = ``z´´ in `x´
                }

                local x = `x´ + 1

                *Linear regression
                reg `var´ phe_sbp_adj age i.sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre i.
> geno_array

                matrix a = e(b)
                matrix b = e(V)
                local reg_beta = a[1,1]
                local reg_variance = b[1,1]

                local reg_n = e(N)

                *Linear regression estimates
                replace outcome = "`var´" in `x´
                replace type = "Multivariable Adjusted" in `x´
                foreach z in beta variance n {
                        replace `z´ = `reg_`z´´ in `x´
                }

                local x = `x´ + 1

        }

keep outcome-f_stat
keep if outcome != ""
replace outcome = "EQ-5D-index" if outcome == "ukb_utility"
qui replace se = sqrt(var)
qui replace p = 2*normal(-abs(beta/se))
sort outcome type
save "$stata_sbp_result\Result_sbp_table_ukb_utility.dta", replace
```

4.6 Step 6: Secondary analysis

4.6.1 Multivariable Mendelian Randomisation

We performed a multivariable MR(MVMR), an extension of a simple MR that accounts for multiple exposures that are potentially related and may influence the outcome of interest[22, 23, 24]. For this analysis, SBP and DBP were considered as exposures. The mean DBP calculated the same way as the mean SBP described above (see exposure and covariate section). For participants who reported taking antihypertensive medications, a 10 mmHg add to the mean DBP measurement[13, 5, 9, 25].

The GWAS study reported sentinel SNPs association with primary and secondary traits[13]. For the 187 sentinel SNPs primarily associated with SBP (after LD clumping), we also identified association with DBP as a secondary trait. Similarly, we also identified 208 sentinel SNPs (after LD clumping) primarily associated with DBP that were also linked to SBP. The combined 395 SNPs were candidates for the MVMR analysis. After the exclusion of ambiguous and missing effect size SNPs, 384 and 382 SNPs were used to construct PRS for SBP and DBP, respectively. Effect estimates for the SNPs were sourced from either ICBP or replication meta-analysis.

4.6.1.1 Working on the DBP data

We had already worked on known and validated SNPs, known but validated (i.e., replicated) for the first time, and novel SNPs associated with the DBP trait. We continued to work on these SNPs.

These SNPs were clumped using the **TwoSampleMR** and **ieugwasr** packages in the **R environment**.

4.6.1.2 Clumping SNPs in LD

The previous stata codes provided csv files for the next step, LD clumping using **TwoSampleMR** and **ieugwas** packages. First, we need to set up an Application Programming Interface (API) to access the **IEU GWAS** database. Then we clump the SNPs in LD.

```
# We will continue working on in the R environment.
# We clump the SNPs in LD.

dbp_data<-read.csv("C:/Users/tabe0010/OneDrive - Monash University/MR_
    backup_file/Articles/Evangelou/new_sbp_snps/stata/dbp_snps/all_dbp_snps
    .csv")
dbp_data[c("Chromosome", "Position")]<-do.call(rbind,strsplit(dbp_data$
    chrpos, ":"))
dbp_data$Chromosome<-as.numeric(dbp_data$Chromosome)
dbp_data$Position<-as.numeric(dbp_data$Position)

dbp_data_2<-dbp_data%>%select(Chromosome, Position, rsID, Beta_DBP, se_DBP
    , A1, A2, EAF_DBP, P_min, Trait, Beta_SBP, se_SBP, EAF_SBP)%>%mutate(id
    .exposure = "icbp_rep")
colnames(dbp_data_2)

colnames(dbp_data_2)<-c("chr.exposure", "pos.exposure", "SNP", "beta.
    exposure", "se.exposure","effect_allele.exposure", "other_allele.
```

```
    exposure", "eaf.exposure", "pval.exposure", "exposure", "Beta_SBP", "se
    _SBP", "EAF_SBP", "id.exposure")
head(dbp_data_2)

dbp_data_2<-dbp_data_2[order(dbp_data_2$chr.exposure),]

clumped_dbp_data_2 <- clump_data(dbp_data_2,
                                 clump_kb = 10000,  # Clumping window
                                    (10,000 kb)
                                 clump_r2 = 0.001,  # LD threshold (r2 <
                                    0.001)
                                 pop = "EUR")  # European LD reference

dbp_snplist<-clumped_dbp_data_2%>%select(SNP)
#sbp_effect_list<-clumped_sbp_data_2%>%select(SNP,effect_allele.exposure,
    beta.exposure)



write.csv(clumped_dbp_data_2, "C:/Users/tabe0010/OneDrive - Monash
    University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dbp_
    snps/dbp_exposure.csv", row.names = FALSE)
write.table(dbp_snplist, "C:/Users/tabe0010/OneDrive - Monash University/
    MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/dbp/data/
    dbp_snplist.txt",row.names = FALSE, col.names =FALSE,quote = FALSE,sep
    = " ")
```

4.6.1.3 Selecting the genetic variants from the BGEN file

PLINK2 was used via the **Swiss Army Knife** to select the necessary SNPs from the UK Biobank. The UKB RAP served as the platform to access the SNPs from the UK Biobank. Alternatively, a *bash* command was used on the local machine to run the process. First, users logged in with their UKB RAP credentials using the Command Prompt (on a Windows machine). Then, Git Bash was opened, the project was selected, and the analysis was run. A job request was sent, and users were notified when it was completed.

```
#############################################################################
#Run the following PLINK2 codes on Git Bash
#############################################################################

#Login through Command Prompt on your Windows machine
dx login

#Run the code below on Git Bash

dx select --level VIEW
#select your project
# make sure your sbp_snplist.txt file is uploaded to the UKB RAP.
#select the "instance type" you want: this makes sure you have enough computation
    power (CPU and GPU).
#In the command below, I put chromosome 1 to 22 to loop through all autosomal
    chromosomes just to show the code. But in actuality, I put two chromosomes at a
```

```
    time. This makes sure I have enough computational space and if there is any
    error, I could adjust the code.

# Loop over chromosomes 1 to 22 and process each one with the SNP list
run_merge=""
for chr in {1..22}; do
    run_merge+="cp /mnt/project/Bulk/Imputation/UKB\ imputation\ from\ genotype/
        ukb22828_c${chr}_b0_v3.bgen .; "
    run_merge+="cp /mnt/project/Bulk/Imputation/UKB\ imputation\ from\ genotype/
        ukb22828_c${chr}_b0_v3.sample .; "
    run_merge+="plink2 --bgen ukb22828_c${chr}_b0_v3.bgen ref-first --sample
        ukb22828_c${chr}_b0_v3.sample --extract dbp_snplist.txt --make-pgen --
        autosome-xy --out ukb22828_c${chr}_v3; "
done

dx run swiss-army-knife -iin="project-GpbQqBjJb7jb1vQjf8ZxVpVY:/DBP_data/dbp_txt/
    dbp_snplist.txt" -icmd="${run_merge}" --tag="Step1" --instance-type "
    mem1_ssd1_v2_x36" --destination="project-GpbQqBjJb7jb1vQjf8ZxVpVY:/DBP_data/
    dbp_geno_data/" --brief --yes

#############################################################################
#Run the following PLINK2 codes via Swiss Army Knife on UKB RAP platform
#############################################################################
#Make sure you have uploaded the sbp_merge_list.txt to UKB RAP file path.
#The merge list should have a sigle column list containing the following text, "
    ukb22828_cx_v3" (without the quotations). The column wil have 22 rows for each
    autosomal chromosomes. Replace "x" with 1-22.

#merging the pgen files

#Execute the command on Swiss Army Knife interface
#inputs are the plink files for the chromosomes and the txt file for the merging
    chromosomes
plink2 --pmerge-list dbp_merge_list.txt pfile --make-pgen --out
    ukb22828_c1_22_v3_dbp_merged

#Calculate the allele dosage
#creating a .raw file for participants with the number of effect allele (0, 1, or
    2) for each snp
# Input the for code below is the merged plink files (pfiles)

plink2 --pfile ukb22828_c1_22_v3_dbp_merged --export A --out ukb22828_dbp_alleles

#Calculate allele frequency

plink2 --pfile ukb22828_c1_22_v3_dbp_merged --freq --out ukb22828_dbp_allele_freq
```

The last PLINK output files, **ukb22828_dbp_alleles.raw** and **ukb22828_dbp_allele_freq.afreq**, contain the allele dosage and allele frequency for each of the 208 SNPs. Download these files to your local machine and save them to the `$dx_data_dbp` file path.

4.6.1.4 Association of Genetic Variants with Quality-Adjusted Life Years

We then worked on the association between the genetic variants and QALYs. The QALYs were regressed on the allele dosages, adjusting for age, sex, and the first 10 genetic principal components to account for population stratification.

```
*************************************************************************
*SNP-QALYs association
*************************************************************************
import delimited "$dx_data_dbp\ukb22828_dbp_alleles.raw",clear
keep iid rs*
rename iid id_phe
merge 1:1 id_phe using "$stata_sbp_input\id_list.dta", keep(3) nogen
save "$stata_dbp_input\snp_alleles_dbp.dta", replace

use "$stata_sbp_output\part_3b.dta", clear
merge 1:1 id_phe using "$stata_dbp_input\snp_alleles_dbp.dta", keep(1 3) nogen
*gen imputation = .
gen snp = ""
gen effect_allele = ""
gen eaf = .
gen outcome = ""
gen beta = .
gen se = .
gen variance = .
gen p = .
gen n = .
local i = 1
local outcomes = "qaly_hes"

        foreach outcome in `outcomes´ {

                qui regress `outcome´ rs* age sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10

                foreach snp of varlist rs* {
                        local snpx = substr("`snp´",1,length("`snp´")-2)
                        *qui replace imputation = `imputation´ in `i´
                        qui replace snp = "`snpx´" in `i´
                        qui replace outcome = "`outcome´" in `i´
                        qui replace beta = _b[`snp´] if snp == "`snpx´" & outcome == "`outcome´"
                        qui replace se = _se[`snp´] if snp == "`snpx´" & outcome == "`outcome´"
                        qui sum `snp´
                        qui replace eaf = r(mean)/2 if snp == "`snpx´"
                        local effect_allele = upper(substr("`snp´",length("`snp´"),1))
                        qui replace effect_allele = "`effect_allele´" if snp == "`snpx´"
                        local i = `i´+1
                }

                *Ns
                qui sum `outcome´
                qui replace n = r(N) if outcome == "`outcome´"
        }

keep snp-n
keep if snp != ""
qui replace variance = se^2
qui replace p = 2*normal(-abs(beta/se))
save "$stata_dbp_result\Results_snp_qalys_dbp.dta", replace
```

```
use "$stata_dbp_result\Results_snp_qalys_dbp.dta", clear
keep if outcome == "qaly_hes"
save "$stata_dbp_result\Results_snp_qaly_hes_dbp.dta"
*******************************************************************************
*merge with allele frequncy data
*******************************************************************************
import delimited "$dx_data_dbp\ukb22828_dbp_allele_freq.afreq",clear
rename id snp
rename alt other_allele
merge 1:1 snp using "$stata_dbp_result\Results_snp_qaly_hes_dbp.dta"
drop chrom ref _merge* alt_freqs obs_ct
export delimited using "$dbp_snps\snp_qaly_hes_dbp.csv",replace
```

4.6.1.5 Data Harmonisation

We now had the SNP-exposure and SNP-outcome association data. The next task was to harmonize these two datasets. For this, we continued working in the previous R environment.

```
qaly_hes_data_dbp<-read.csv("C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dbp_snps/snp_qaly_hes_dbp.csv")
colnames(qaly_hes_data_dbp)<-c("SNP", "other_allele.outcome", "effect_allele.outcome", "eaf.outcome", "outcome", "beta.outcome", "se.outcome", "variance.outcome", "pval.outcome", "samplesize.outcome")
qaly_hes_data_dbp$id.outcome = "ukb"

harmonise_data_dbp <- harmonise_data(
  exposure_dat = clumped_dbp_data_2,
  outcome_dat = qaly_hes_data_dbp
)


dbp_effect_list<-harmonise_data_dbp%>%select(SNP,effect_allele.exposure,beta.exposure)
dbp_snplist_exclude<-harmonise_data_dbp%>%filter(mr_keep == "FALSE")%>%select(SNP)

write.table(dbp_effect_list, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/dbp/data/dbp_effect_list.txt",row.names = FALSE, col.names =FALSE, quote = FALSE,sep = " ")
write.table(dbp_snplist_exclude, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/dbp/data/dbp_snplist_exclude.txt", row.names = FALSE, col.names =FALSE,quote = FALSE,sep = " ")
```

4.6.1.6 Combining the genetic data

We had now identified the genetic variants primarily associated with SBP and secondarily with DBP, and vice versa. These datasets were then combined.

```

#We will continue working on the previous environment

```

```
#Let's combine the SNPs that are LD clumped and associted with SBP and DBP to create the SNP list (this is the total sets of SNPs)

sbp_dbp_snplist<-rbind(sbp_snplist, dbp_snplist)

#Let's work on combining SNPs primarily associated with SBP and seconarily associated with DBP

temp_sbp<-harmonise_data%>%select(chr.exposure, pos.exposure, SNP, effect_allele.exposure,other_allele.exposure,beta.exposure,se.exposure,eaf.exposure)
temp_sbp_2<-harmonise_data_dbp%>%select(chr.exposure, pos.exposure, SNP, effect_allele.exposure,other_allele.exposure, Beta_SBP, se_SBP, EAF_SBP)%>%rename(beta.exposure = Beta_SBP, se.exposure = se_SBP, eaf.exposure = EAF_SBP)
temp_sbp_append<-rbind(temp_sbp, temp_sbp_2)
temp_sbp_append<-temp_sbp_append[order(temp_sbp_append$chr.exposure, temp_sbp_append$pos.exposure),]
temp_sbp_append$beta.exposure<-as.numeric(temp_sbp_append$beta.exposure)
temp_sbp_append$se.exposure<-as.numeric(temp_sbp_append$se.exposure)
n_distinct(temp_sbp_append$SNP)
temp_sbp_na<-temp_sbp_append%>%filter(is.na(beta.exposure))%>%select(SNP)
mvmr_sbp_snplist_exclude<-rbind(sbp_snplist_exclude, dbp_snplist_exclude, temp_sbp_na)
mvmr_sbp_effect_list<-temp_sbp_append%>%select(SNP, effect_allele.exposure, beta.exposure)

#Let's work on combining SNPs primarily associated with DBP and seconarily associated with SBP

temp_dbp<-harmonise_data_dbp%>%select(chr.exposure, pos.exposure, SNP, effect_allele.exposure,other_allele.exposure,beta.exposure,se.exposure, eaf.exposure)
temp_dbp_2<-harmonise_data%>%select(chr.exposure, pos.exposure, SNP, effect_allele.exposure,other_allele.exposure, Beta_DBP, se_DBP, EAF_DBP)%>%rename(beta.exposure = Beta_DBP, se.exposure = se_DBP, eaf.exposure = EAF_DBP)
temp_dbp_append<-rbind(temp_dbp, temp_dbp_2)
temp_dbp_append<-temp_dbp_append[order(temp_dbp_append$chr.exposure, temp_dbp_append$pos.exposure),]
temp_dbp_append$beta.exposure<-as.numeric(temp_dbp_append$beta.exposure)
temp_dbp_append$se.exposure<-as.numeric(temp_dbp_append$se.exposure)
temp_dbp_na<-temp_dbp_append%>%filter(is.na(beta.exposure))%>%select(SNP)
mvmr_dbp_snplist_exclude<-rbind(sbp_snplist_exclude, dbp_snplist_exclude, temp_dbp_na)
mvmr_dbp_effect_list<-temp_dbp_append%>%select(SNP, effect_allele.exposure, beta.exposure)

#Save the files

```

```
write.table(sbp_dbp_snplist, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/sbp_dbp/data/sbp_dbp_snplist.txt",row.names = FALSE, col.names =FALSE, quote = FALSE,sep = " ")
write.table(mvmr_sbp_snplist_exclude, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/sbp_dbp/data/mvmr_sbp_snplist_exclude.txt",row.names = FALSE, col.names =FALSE,quote = FALSE,sep = " ")
write.table(mvmr_dbp_snplist_exclude, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/sbp_dbp/data/mvmr_dbp_snplist_exclude.txt",row.names = FALSE, col.names =FALSE,quote = FALSE,sep = " ")
write.table(mvmr_sbp_effect_list, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/sbp_dbp/data/mvmr_sbp_effect_list.txt",row.names = FALSE, col.names = FALSE,quote = FALSE,sep = " ")
write.table(mvmr_dbp_effect_list, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/dx_data/sbp_dbp/data/mvmr_dbp_effect_list.txt",row.names = FALSE, col.names = FALSE,quote = FALSE,sep = " ")
write.csv(temp_sbp_append, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/sbp_dbp_snps/mvmr_sbp.csv", row.names = FALSE)
write.csv(temp_dbp_append, "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/sbp_dbp_snps/mvmr_dbp.csv", row.names = FALSE)
```

4.6.1.7 Working on SNPs primarily associated with SBP and seconarily associated with DBP

We calculated the PRS, allele frequency, and allele dosage for these SNPs using the **sbp_dbp_snplist.txt**, **mvmr_sbp_snplist_exclude.txt**, and **mvmr_sbp_effect_list.txt** files.

We had 395 total SNPs; of these, 382 SNPs were associated with SBP as a primary or secondary trait, and 384 SNPs were associated with DBP as a primary or secondary trait. The remaining SNPs were either ambiguous SNPs or had missing effect sizes.

```
#########################################################################
#Run the following PLINK2 codes on Git Bash
#########################################################################

#Login through Command Prompt on your Windows machine
dx login

#Run the code below on Git Bash

dx select --level VIEW
#select your project
# make sure your sbp_dbp_snplist.txt file is uploaded to the UKB RAP.
#select the "instance type" you want: this makes sure you have enough computation power (CPU and GPU).
#In the command below, I put chromosome 1 to 22 to loop through all autosomal chromosomes just to show the code. But in actuality, I put two chromosomes at a
```

```
    time. This makes sure I have enough computational space and if there is any
    error, I could adjust the code.

# Loop over chromosomes 1 to 22 and process each one with the SNP list

run_merge=""
for chr in {1..22}; do
    run_merge+="cp /mnt/project/Bulk/Imputation/UKB\ imputation\ from\ genotype/
        ukb22828_c${chr}_b0_v3.bgen .; "
    run_merge+="cp /mnt/project/Bulk/Imputation/UKB\ imputation\ from\ genotype/
        ukb22828_c${chr}_b0_v3.sample .; "
    run_merge+="plink2 --bgen ukb22828_c${chr}_b0_v3.bgen ref-first --sample
        ukb22828_c${chr}_b0_v3.sample --extract sbp_dbp_snplist.txt --exclude
        mvmr_sbp_snplist_exclude.txt --make-pgen --autosome-xy --out ukb22828_c${
        chr}_v3; "
done

dx run swiss-army-knife -iin="project-GpbQqBjJb7jb1vQjf8ZxVpVY:/SBP_DBP_data/
    mvmr_sbp_txt/sbp_dbp_snplist.txt" -iin="project-GpbQqBjJb7jb1vQjf8ZxVpVY:/
    SBP_DBP_data/mvmr_sbp_txt/mvmr_sbp_snplist_exclude.txt" -icmd="${run_merge}" --
    tag="mvmr_chr1_22" --instance-type "mem1_ssd1_v2_x36" --destination="project-
    GpbQqBjJb7jb1vQjf8ZxVpVY:/SBP_DBP_data/mvmr_sbp_geno_data/mvmr_sbp_chromosomes/
    " --brief --yes


#########################################################################
#Run the following PLINK2 codes via Swiss Army Knife on UKB RAP platform
#########################################################################
#Make sure you have uploaded the sbp_dbp_merge_list.txt to UKB RAP file path.
#The merge list should have a sigle column list containing the following text, "
    ukb22828_cx_v3" (without the quotations). The column wil have 22 rows for each
    autosomal chromosomes. Replace "x" with 1-22.

#merging the pgen files

#Execute the command on Swiss Army Knife interface
#inputs are the plink files for the chromosomes and the txt file for the merging
    chromosomes
plink2 --pmerge-list sbp_dbp_merge_list.txt pfile --make-pgen --out
    ukb22828_c1_22_v3_mvmr_sbp_merged

#PRS mvmr_SBP
plink2 --pfile ukb22828_c1_22_v3_mvmr_sbp_merged --score mvmr_sbp_effect_list.txt
    cols=+scoresums --out ukb22828_mvmr_sbp_prs

#Calculate the allele dosage
#creating a .raw file for participants with the number of effect allele (0, 1, or
    2) for each snp
# Input the for code below is the merged plink files (pfiles)

plink2 --pfile ukb22828_c1_22_v3_mvmr_sbp_merged --export A --out
    ukb22828_mvmr_sbp_alleles

#Calculate allele frequency

plink2 --pfile ukb22828_c1_22_v3_mvmr_sbp_merged --freq --out
```

```
    ukb22828_mvmr_sbp_allele_freq
```

4.6.1.8 Working on SNPs primarily associated with DBP and seconarily associated with SBP

We calculated the PRS, allele frequency, and allele dosage for these SNPs using the **sbp_dbp_snplist.txt**, **mvmr_dbp_snplist_exclude.txt**, and **mvmr_dbp_effect_list.txt** files.

```
##############################################################################
#Run the following PLINK2 codes on Git Bash
##############################################################################

#Login through Command Prompt on your Windows machine
dx login

#Run the code below on Git Bash

dx select --level VIEW
#select your project
# make sure your sbp_dbp_snplist.txt file is uploaded to the UKB RAP.
#select the "instance type" you want: this makes sure you have enough computation
    power (CPU and GPU).
#In the command below, I put chromosome 1 to 22 to loop through all autosomal
    chromosomes just to show the code. But in actuality, I put two chromosomes at a
     time. This makes sure I have enough computational space and if there is any
    error, I could adjust the code.

# Loop over chromosomes 1 to 22 and process each one with the SNP list

run_merge=""
for chr in {1..22}; do
    run_merge+="cp /mnt/project/Bulk/Imputation/UKB\ imputation\ from\ genotype/
        ukb22828_c${chr}_b0_v3.bgen .; "
    run_merge+="cp /mnt/project/Bulk/Imputation/UKB\ imputation\ from\ genotype/
        ukb22828_c${chr}_b0_v3.sample .; "
    run_merge+="plink2 --bgen ukb22828_c${chr}_b0_v3.bgen ref-first --sample
        ukb22828_c${chr}_b0_v3.sample --extract sbp_dbp_snplist.txt --exclude
        mvmr_dbp_snplist_exclude.txt --make-pgen --autosome-xy --out ukb22828_c${
        chr}_v3; "
done

dx run swiss-army-knife -iin="project-GpbQqBjJb7jb1vQjf8ZxVpVY:/SBP_DBP_data/
    mvmr_dbp_txt/sbp_dbp_snplist.txt" -iin="project-GpbQqBjJb7jb1vQjf8ZxVpVY:/
    SBP_DBP_data/mvmr_dbp_txt/mvmr_dbp_snplist_exclude.txt" -icmd="${run_merge}" --
    tag="mvmr_chr1_22" --instance-type "mem1_ssd1_v2_x36" --destination="project-
    GpbQqBjJb7jb1vQjf8ZxVpVY:/SBP_DBP_data/mvmr_dbp_geno_data/mvmr_dbp_chromosomes/
    " --brief --yes


##############################################################################
#Run the following PLINK2 codes via Swiss Army Knife on UKB RAP platform
##############################################################################
#Make sure you have uploaded the sbp_dbp_merge_list.txt to UKB RAP file path.
#The merge list should have a sigle column list containing the following text, "
    ukb22828_cx_v3" (without the quotations). The column wil have 22 rows for each
```

```
    autosomal chromosomes. Replace "x" with 1-22.

#merging the pgen files

#Execute the command on Swiss Army Knife interface
#inputs are the plink files for the chromosomes and the txt file for the merging
    chromosomes
plink2 --pmerge-list sbp_dbp_merge_list.txt pfile --make-pgen --out
    ukb22828_c1_22_v3_mvmr_dbp_merged

#PRS mvmr_DBP
plink2 --pfile ukb22828_c1_22_v3_mvmr_dbp_merged --score mvmr_dbp_effect_list.txt
    cols=+scoresums --out ukb22828_mvmr_dbp_prs

#Calculate the allele dosage
#creating a .raw file for participants with the number of effect allele (0, 1, or
    2) for each snp
# Input the for code below is the merged plink files (pfiles)

plink2 --pfile ukb22828_c1_22_v3_mvmr_dbp_merged --export A --out
    ukb22828_mvmr_dbp_alleles

#Calculate allele frequency

plink2 --pfile ukb22828_c1_22_v3_mvmr_dbp_merged --freq --out
    ukb22828_mvmr_dbp_allele_freq
```

We have now calculated the PRS for SNPs effect on SBP. Download the **ukb22828_sbp_prs.sscore** file to your local machine and save them to the `$dx_data_sbp` file path.

4.6.1.9 Combining Phenotype and Genotype data

We then combined our phenotype data with the PRS. The **part_3b.dta** file contained the phenotype data for our cohort, while the **ukb22828_mvmr_sbp_prs.sscore** and **ukb22828_mvmr_dbp_prs.sscore** files contained the PRS for each participant in the UK Biobank. The next Stata code merged the two datasets and prepared the data for the MVMR analysis. We saved the data to the **part_4a.dta** Stata file that we had previously created.

```
import delimited "$dx_data_sbp\ukb22828_mvmr_sbp_prs.sscore", clear
 gen id_phe = iid // IID: Individual ID
 save "$dx_data_sbp\ukb22828_mvmr_sbp_prs.sscore.dta", replace

import delimited "$dx_data_sbp_dbp\ukb22828_mvmr_dbp_prs.sscore", clear
 gen id_phe = iid // IID: Individual ID
 save "$dx_data_sbp_dbp\ukb22828_mvmr_dbp_prs.sscore.dta", replace

use "$stata_sbp_output\part_3b.dta", clear

 merge 1:1 id_phe using "$dx_data_sbp_dbp\ukb22828_mvmr_sbp_prs.sscore.dta"
 keep if _merge == 3
 drop _merge*
 rename score1_sum prs_mvmr_sbp
 drop fid iid allele_ct named_allele_dosage_sum score1_avg

 merge 1:1 id_phe using "$dx_data_sbp_dbp\ukb22828_mvmr_dbp_prs.sscore.dta"
 keep if _merge == 3
 drop _merge*
```

```
 rename score1_sum prs_mvmr_dbp
 drop fid iid allele_ct named_allele_dosage_sum score1_avg

save "$stata_sbp_output\part_4a.dta", replace
```

4.6.1.10 Analysis

To estimate the direct effect of SBP on QALYs conditional on DBP, we employed the difference method approach [24]. The total effect (β_1^*) of SBP on QALYs was estimated in the main analysis. The direct effect (β_1) of SBP on QALYs was estimated by considering both SBP and DBP traits. At the first stage, we regressed the exposure trait (SBP) on the polygenic risk score (PRS) for SNPs that were associated with SBP (i.e., 382 SNPs). We repeated the same approach for the DBP exposure by regressing it on the PRS for SNPs that were associated with DBP (i.e., 384 SNPs). In the second stage, the outcome (QALYs) was regressed on the predicted values for SBP and DBP from the first-stage models. For the direct effect models, age, sex, UK Biobank assessment centre, genotyping array, and the first 10 genetic principal components for population stratification were used as covariates. The indirect effect was calculated as the difference between the total effect estimate and the direct effect estimate, i.e., $\beta_1^* - \beta_1$.

```
*MVMR
use "$stata_sbp_output\part_4a.dta", clear
*Create table
gen outcome = ""
gen type = ""
gen n = .
gen beta = .
gen variance = .
gen se = .
gen double p = .
*gen double p_endog = .
gen f_stat = .
local x = 1

        foreach var in qaly_hes {
                dis "Outcome = `var´"

                *MR analysis (main)
                ivreg2 `var´ (phe_sbp_adj = prs_sbp) age i.sex pc1 pc2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 p
> c10 i.centre i.geno_array, robust

                matrix a = e(b)
                matrix b = e(V)
                local beta_total = a[1,1]
                local variance_total = b[1,1]

                local n_total = e(N)
                local f_stat_total = e(widstat)
                *local p_endog_total = e(estatp)

                replace outcome = "`var´" in `x´
                replace type = "Total effect" in `x´
                *qui replace imputation = `j´ in `x´
                foreach z in beta variance n f_stat {
                        replace `z´ = ``z´_total´ in `x´
                }

                local x = `x´ + 1

                *MV MR
                 ivreg2 `var´ (phe_sbp_adj phe_dbp_adj = prs_mvmr_sbp prs_mvmr_dbp) age i.sex pc1 pc
> 2 pc3 pc4 pc5 pc6 pc7 pc8 pc9 pc10 i.centre i.geno_array, robust

                matrix a = e(b)
```

```
                matrix b = e(V)
                local beta_direct = a[1,1]
                local variance_direct = b[1,1]

                local n_direct = e(N)
                local f_stat_direct = e(widstat)
                *local p_endog = e(estatp)

                replace outcome = "`var´" in `x´
                replace type = "Direct effect" in `x´
                *qui replace imputation = `j´ in `x´
                foreach z in beta variance n f_stat {
                        replace `z´ = ``z´_direct´ in `x´
                }


                local x = `x´ + 1

        }
keep outcome-f_stat
keep if outcome != ""
replace outcome = "QALYs per year (with 240 comorbidities)" if outcome == "qaly_hes"
qui replace se = sqrt(var)
qui replace p = 2*normal(-abs(beta/se))
sort outcome type
save "$stata_sbp_result\Result_mvmr_sbp_table.dta", replace
```

4.7 Step 7: Tables and Figures

The codes that were used to generate the main results' tables and figures are provided here.

4.7.1 Tables

4.7.1.1 Table 1: Background characteristics

```
{
use "$stata_sbp_output\part_4a.dta", clear

gen Variable = ""
gen All = ""
gen N_All = ""
gen Men = ""
gen N_Men = ""
gen Women = ""
gen N_Women = ""
order Variable-N_Women, first

gen years = months/12
gen qual_0 = 0
replace qual_0 = 1 if qual_1 == 0 & qual_2 == 0 & qual_3 == 0
gen death = 0

*bysort dsource: su date_death
replace death = 1 if date_death <= 753 & dsource != "PEDW"
replace death = 1 if date_death <= 748 & dsource == "PEDW"

qui replace Variable = "N" in 1
qui replace Variable = "Age at recruitment, years [Median (IQR)]" in 2
qui replace Variable = "Body Mass Index, kg/m2 [Median (IQR)]" in 3
qui replace Variable = "Systolic blood pressure, mmHg [Median (IQR)]" in 4
qui replace Variable = "Years of follow-up [Median (IQR)]" in 5
qui replace Variable = "Death [N (%)]*" in 6
qui replace Variable = "Qualification: None [N (%)]" in 7
qui replace Variable = "Qualification: A levels, O level, GCSE or CSE [N (%)]" in 8
qui replace Variable = "Qualification: NVQ or other [N (%)]" in 9
qui replace Variable = "Qualification: College or university degree [N (%)]" in 10
qui replace Variable = "Average QALYs per year, HES only [Median (IQR)]*" in 11

local i = 1

*Number of participants
{
qui sum sex
local x = r(N)
local x: dis %9.0fc `x´
local x = strtrim("`x´")
local x1 = r(N)*r(mean)
local x1: dis %9.0fc `x1´
local x1 = strtrim("`x1´")
local x2 = r(N)*(1-r(mean))
local x2: dis %9.0fc `x2´
local x2 = strtrim("`x2´")
qui replace All = "`x´" in `i´
qui replace Men = "`x1´" in `i´
qui replace Women = "`x2´" in `i´
local i = `i´ + 1
}

*Age, SBP, systolic blood pressure, follow-up
foreach var of varlist age phe_bmi phe_sbp years  {
        sum `var´,d
        local median = r(p50)
        local median: dis %9.2f `median´
        local median = strtrim("`median´")
        local N = r(N)
        local N: dis %9.0fc `N´
        local N = strtrim("`N´")
```

```
        local p25 = r(p25)
        local p25: dis %9.2f `p25´
        local p25 = strtrim("`p25´")
        local p75 = r(p75)
        local p75: dis %9.2f `p75´
        local p75 = strtrim("`p75´")
        local x = "`median´ (`p25´ to `p75´)"
        qui replace All = "`x´" in `i´
        qui replace N_All = "`N´" in `i´

        sum `var´ if sex == 1,d
        local median = r(p50)
        local median: dis %9.2f `median´
        local median = strtrim("`median´")
        local N = r(N)
        local N: dis %9.0fc `N´
        local N = strtrim("`N´")
        local p25 = r(p25)
        local p25: dis %9.2f `p25´
        local p25 = strtrim("`p25´")
        local p75 = r(p75)
        local p75: dis %9.2f `p75´
        local p75 = strtrim("`p75´")
        local x = "`median´ (`p25´ to `p75´)"
        qui replace Men = "`x´" in `i´
        qui replace N_Men = "`N´" in `i´
        sum `var´ if sex == 0,d
        local median = r(p50)
        local median: dis %9.2f `median´
        local median = strtrim("`median´")
        local N = r(N)
        local N: dis %9.0fc `N´
        local N = strtrim("`N´")
        local p25 = r(p25)
        local p25: dis %9.2f `p25´
        local p25 = strtrim("`p25´")
        local p75 = r(p75)
        local p75: dis %9.2f `p75´
        local p75 = strtrim("`p75´")
        local x = "`median´ (`p25´ to `p75´)"
        qui replace Women = "`x´" in `i´
        qui replace N_Women = "`N´" in `i´

        local i = `i´ + 1
}
*Primary care, death & qualifications
foreach var of varlist death qual_0 qual_1-qual_3 {

        qui sum `var´
        local N = r(mean)*r(N)
        local N: dis %9.0fc `N´
        local N = strtrim("`N´")
        local percent = r(mean)*100
        local percent: dis %9.2f `percent´
        local percent = strtrim("`percent´")
        local x = "`N´ (`percent´)"
        qui replace All = "`x´" in `i´

        local N2 = r(N)
        local N2: dis %9.0fc `N2´
        local N2 = strtrim("`N2´")
        qui replace N_All = "`N2´" in `i´

        qui sum `var´ if sex == 1
        local N = r(mean)*r(N)
        local N: dis %9.0fc `N´
        local N = strtrim("`N´")
        local percent = r(mean)*100
```

```
        local percent: dis %9.2f `percent´
        local percent = strtrim("`percent´")
        local x = "`N´ (`percent´)"
        qui replace Men = "`x´" in `i´

        local N2 = r(N)
        local N2: dis %9.0fc `N2´
        local N2 = strtrim("`N2´")
        qui replace N_Men = "`N2´" in `i´
        qui sum `var´ if sex == 0
        local N = r(mean)*r(N)
        local N: dis %9.0fc `N´
        local N = strtrim("`N´")
        local percent = r(mean)*100
        local percent: dis %9.2f `percent´
        local percent = strtrim("`percent´")
        local x = "`N´ (`percent´)"
        qui replace Women = "`x´" in `i´

        local N2 = r(N)
        local N2: dis %9.0fc `N2´
        local N2 = strtrim("`N2´")
        qui replace N_Women = "`N2´" in `i´

        local i = `i´+1
}

*HES QALYs
foreach var of varlist qaly_hes {
        *All

        sum `var´, d
        local median = r(p50)
        local median: dis %9.2f `median´
        local median = strtrim("`median´")
        local p25 = r(p25)
        local p25: dis %9.2f `p25´
        local p25 = strtrim("`p25´")
        local p75 = r(p75)
        local p75: dis %9.2f `p75´
        local p75 = strtrim("`p75´")
        local x = "`median´ (`p25´ to `p75´)"
        qui replace All = "`x´" in `i´

        *Men
        su `var´ if sex == 1, d
        local median = r(p50)
        local median: dis %9.2f `median´
        local median = strtrim("`median´")
        local p25 = r(p25)
        local p25: dis %9.2f `p25´
        local p25 = strtrim("`p25´")
        local p75 = r(p75)
        local p75: dis %9.2f `p75´
        local p75 = strtrim("`p75´")
        local x = "`median´ (`p25´ to `p75´)"
        qui replace Men = "`x´" in `i´

        *Women
        su `var´ if sex == 0, d
        local median = r(p50)
        local median: dis %9.2f `median´
        local median = strtrim("`median´")
        local p25 = r(p25)
        local p25: dis %9.2f `p25´
        local p25 = strtrim("`p25´")
        local p75 = r(p75)
        local p75: dis %9.2f `p75´
```

```
        local p75 = strtrim("`p75´")
        local x = "`median´ (`p25´ to `p75´)"
        qui replace Women = "`x´" in `i´

        local i = `i´ + 1
}

keep Var-N_Women
drop N*
drop if Variable == ""
save "$stata_sbp_result\Table_1.dta", replace
}
```

4.7.1.2 Table 2: Main analysis

```
use "$stata_sbp_result\Result_sbp_table.dta", clear

replace beta = 10*beta
replace se = 10*se

*Change to percentages
replace beta = 100*beta
replace se = 100*se

gen lower = beta-1.96*se
gen upper = beta+1.96*se

format beta se lower upper %9.2f

*Effect estimate variable - sort of decimal places
foreach var of varlist beta se lower upper {
        tostring `var´, gen(`var´_x) force

        replace `var´_x = substr(`var´_x,1,4) if `var´ < 0 & `var´ > -1
        replace `var´_x = substr(`var´_x,1,5) if `var´ <= -1 & `var´ > -10
        replace `var´_x = substr(`var´_x,1,6) if `var´ <= -10 & `var´ > -100
        replace `var´_x = substr(`var´_x,1,7) if `var´ <= -100 & `var´ > -1000
        replace `var´_x = substr(`var´_x,1,3) if `var´ > 0 & `var´ < 1
        replace `var´_x = substr(`var´_x,1,4) if `var´ >= 1 & `var´ < 10
        replace `var´_x = substr(`var´_x,1,5) if `var´ >= 10 & `var´ < 100
        replace `var´_x = substr(`var´_x,1,6) if `var´ >= 100 & `var´ < 1000

        replace `var´_x = subinstr(`var´_x,".","0.",.) if substr(`var´_x,1,1) == "." | substr(`var´_
> x,1,2) == "-."
        replace `var´_x = `var´_x + "%"

        *replace `var´_x = "£"+`var´_x if strpos(outcome,"QALYs") == 0
}


gen effect = beta_x + " (" + lower_x + " to " + upper_x + ")"

keep outcome type n effect se_x p p_endog f_stat
rename se_x se
order outcome type n effect se p p_endog f_stat

save "$stata_sbp_result\Table_2.dta", replace
```

4.7.1.3 Table 3: Sensitivity analyses - No history of antihypertensive medication cohort

```
use "$stata_sbp_result\Result_sbp_exclusive_table_no_medication.dta", clear

replace beta = 10*beta
replace se = 10*se

*Change to percentages
replace beta = 100*beta
replace se = 100*se

gen lower = beta-1.96*se
gen upper = beta+1.96*se

format beta se lower upper %9.2f

*Effect estimate variable - sort of decimal places
foreach var of varlist beta se lower upper {
        tostring `var´, gen(`var´_x) force

        replace `var´_x = substr(`var´_x,1,4) if `var´ < 0 & `var´ > -1
        replace `var´_x = substr(`var´_x,1,5) if `var´ <= -1 & `var´ > -10
        replace `var´_x = substr(`var´_x,1,6) if `var´ <= -10 & `var´ > -100
        replace `var´_x = substr(`var´_x,1,7) if `var´ <= -100 & `var´ > -1000
        replace `var´_x = substr(`var´_x,1,3) if `var´ > 0 & `var´ < 1
        replace `var´_x = substr(`var´_x,1,4) if `var´ >= 1 & `var´ < 10
        replace `var´_x = substr(`var´_x,1,5) if `var´ >= 10 & `var´ < 100
        replace `var´_x = substr(`var´_x,1,6) if `var´ >= 100 & `var´ < 1000

        replace `var´_x = subinstr(`var´_x,".","0.",.) if substr(`var´_x,1,1) == "." | substr(`var´_
> x,1,2) == "-."
        replace `var´_x = `var´_x + "%"

        *replace `var´_x = "£"+`var´_x if strpos(outcome,"QALYs") == 0
}


gen effect = beta_x + " (" + lower_x + " to " + upper_x + ")"

keep outcome type n effect se_x p p_endog f_stat
rename se_x se
order outcome type n effect se p p_endog f_stat

save "$stata_sbp_result\Table_3.dta", replace
```

4.7.1.4 Table 4: Sensitivity analyses - Two-sample MR

```
use "$stata_sbp_result\Results_table_sensitivity_sbp.dta", clear

gen ll_ivw = ivw - ivw_se*1.96
gen ul_ivw = ivw + ivw_se*1.96
gen ll_egger_slope = egger_slope - egger_slope_se*1.96
gen ul_egger_slope = egger_slope + egger_slope_se*1.96
gen ll_egger_cons = egger_cons - egger_cons_se*1.96
gen ul_egger_cons = egger_cons + egger_cons_se*1.96
gen ll_median = median - median_se*1.96
gen ul_median = median + median_se*1.96
gen ll_mode = mode - mode_se*1.96
gen ul_mode = mode + mode_se*1.96

foreach var of varlist ivw ivw_se ll_ivw ul_ivw egger_slope egger_slope_se ll_egger_slope ul_egger_s
> lope egger_cons egger_cons_se ll_egger_cons ul_egger_cons median median_se ll_median ul_median mod
> e mode_se ll_mode ul_mode {
        replace `var´ = `var´*10
        replace `var´ = `var´*100
        tostring `var´, gen(`var´_x) force

                replace `var´_x = substr(`var´_x,1,4) if `var´ < 0 & `var´ > -1
        replace `var´_x = substr(`var´_x,1,5) if `var´ <= -1 & `var´ > -10
        replace `var´_x = substr(`var´_x,1,6) if `var´ <= -10 & `var´ > -100
        replace `var´_x = substr(`var´_x,1,7) if `var´ <= -100 & `var´ > -1000
        replace `var´_x = substr(`var´_x,1,3) if `var´ > 0 & `var´ < 1
        replace `var´_x = substr(`var´_x,1,4) if `var´ >= 1 & `var´ < 10
        replace `var´_x = substr(`var´_x,1,5) if `var´ >= 10 & `var´ < 100
        replace `var´_x = substr(`var´_x,1,6) if `var´ >= 100 & `var´ < 1000

        replace `var´_x = subinstr(`var´_x,".","0.",.) if substr(`var´_x,1,1) == "." | substr(`var´_
> x,1,2) == "-."
        replace `var´_x = `var´_x + "%"
}

drop ivw ivw_se egger_slope egger_slope_se egger_cons egger_cons_se median median_se mode mode_se ll
> _ivw ul_ivw ll_egger_slope ul_egger_slope ll_egger_cons ul_egger_cons ll_median ul_median ll_mode
> ul_mode

rename (ivw_x ivw_se_x egger_slope_x egger_slope_se_x egger_cons_x egger_cons_se_x median_x median_s
> e_x mode_x mode_se_x ll_ivw_x ul_ivw_x ll_egger_slope_x ul_egger_slope_x ll_egger_cons_x ul_egger_
> cons_x ll_median_x ul_median_x ll_mode_x ul_mode_x) (ivw ivw_se egger_slope egger_slope_se egger_c
> ons egger_cons_se median median_se mode mode_se ll_ivw ul_ivw ll_egger_slope ul_egger_slope ll_egg
> er_cons ul_egger_cons ll_median ul_median ll_mode ul_mode)

gen ivw_effect = ivw + " (" + ll_ivw + " to " + ul_ivw + ")"
gen egger_slope_effect =  egger_slope + " (" + ll_egger_slope + " to " + ul_egger_slope + ")"
gen egger_cons_effect =  egger_cons + " (" + ll_egger_cons + " to " + ul_egger_cons + ")"
gen median_effect = median + " (" + ll_median + " to " + ul_median + ")"
gen mode_effect = mode + " (" + ll_mode + " to " + ul_mode + ")"

keep outcome ivw_effect ivw_se ivw_p egger_slope_effect egger_slope_se egger_slope_p egger_cons_effe
> ct egger_cons_se egger_cons_p median_effect median_se median_p mode_effect mode_se mode_p
order outcome ivw_effect ivw_se ivw_p egger_slope_effect egger_slope_se egger_slope_p egger_cons_eff
> ect egger_cons_se egger_cons_p median_effect median_se median_p mode_effect mode_se mode_p

save "$stata_sbp_result\Table_4.dta",replace
```

4.7.1.5 Table 5: Sensitivity analyses - EQ-5D-index from UK Biobank survey

```
use "$stata_sbp_result\Result_sbp_table_ukb_utility.dta", clear

replace beta = 10*beta
replace se = 10*se

*Change to percentages
replace beta = 100*beta
replace se = 100*se

gen lower = beta-1.96*se
gen upper = beta+1.96*se

format beta se lower upper %9.2f

*Effect estimate variable - sort of decimal places
foreach var of varlist beta se lower upper {
        tostring `var´, gen(`var´_x) force

        replace `var´_x = substr(`var´_x,1,4) if `var´ < 0 & `var´ > -1
        replace `var´_x = substr(`var´_x,1,5) if `var´ <= -1 & `var´ > -10
        replace `var´_x = substr(`var´_x,1,6) if `var´ <= -10 & `var´ > -100
        replace `var´_x = substr(`var´_x,1,7) if `var´ <= -100 & `var´ > -1000
        replace `var´_x = substr(`var´_x,1,3) if `var´ > 0 & `var´ < 1
        replace `var´_x = substr(`var´_x,1,4) if `var´ >= 1 & `var´ < 10
        replace `var´_x = substr(`var´_x,1,5) if `var´ >= 10 & `var´ < 100
        replace `var´_x = substr(`var´_x,1,6) if `var´ >= 100 & `var´ < 1000

        replace `var´_x = subinstr(`var´_x,".","0.",.) if substr(`var´_x,1,1) == "." | substr(`var´_
> x,1,2) == "-."
        replace `var´_x = `var´_x + "%"

        *replace `var´_x = "£"+`var´_x if strpos(outcome,"QALYs") == 0
}


gen effect = beta_x + " (" + lower_x + " to " + upper_x + ")"

keep outcome type n effect se_x p p_endog f_stat
rename se_x se
order outcome type n effect se p p_endog f_stat

save "$stata_sbp_result\Table_5.dta", replace
```

4.7.1.6 Table 6: Secondary analysis

```
use "$stata_sbp_result\Result_mvmr_sbp_table.dta", clear

replace beta = 10*beta
replace se = 10*se

*Change to percentages
replace beta = 100*beta
replace se = 100*se

gen lower = beta-1.96*se
gen upper = beta+1.96*se

format beta se lower upper %9.2f

*Effect estimate variable - sort of decimal places
foreach var of varlist beta se lower upper {
        tostring `var´, gen(`var´_x) force

        replace `var´_x = substr(`var´_x,1,4) if `var´ < 0 & `var´ > -1
        replace `var´_x = substr(`var´_x,1,5) if `var´ <= -1 & `var´ > -10
        replace `var´_x = substr(`var´_x,1,6) if `var´ <= -10 & `var´ > -100
        replace `var´_x = substr(`var´_x,1,7) if `var´ <= -100 & `var´ > -1000
        replace `var´_x = substr(`var´_x,1,3) if `var´ > 0 & `var´ < 1
        replace `var´_x = substr(`var´_x,1,4) if `var´ >= 1 & `var´ < 10
        replace `var´_x = substr(`var´_x,1,5) if `var´ >= 10 & `var´ < 100
        replace `var´_x = substr(`var´_x,1,6) if `var´ >= 100 & `var´ < 1000

        replace `var´_x = subinstr(`var´_x,".","0.",.) if substr(`var´_x,1,1) == "." | substr(`var´_
> x,1,2) == "-."
        replace `var´_x = `var´_x + "%"
}


gen effect = beta_x + " (" + lower_x + " to " + upper_x + ")"

keep outcome n type effect se_x p f_stat
rename se_x se
order outcome n type effect se p f_stat

save "$stata_sbp_result\Table_6.dta", replace
```

4.7.2 Figures

4.7.2.1 Figure 1

Flow chart showing selection criteria for genetic instruments primarily associated with systolic blood pressure

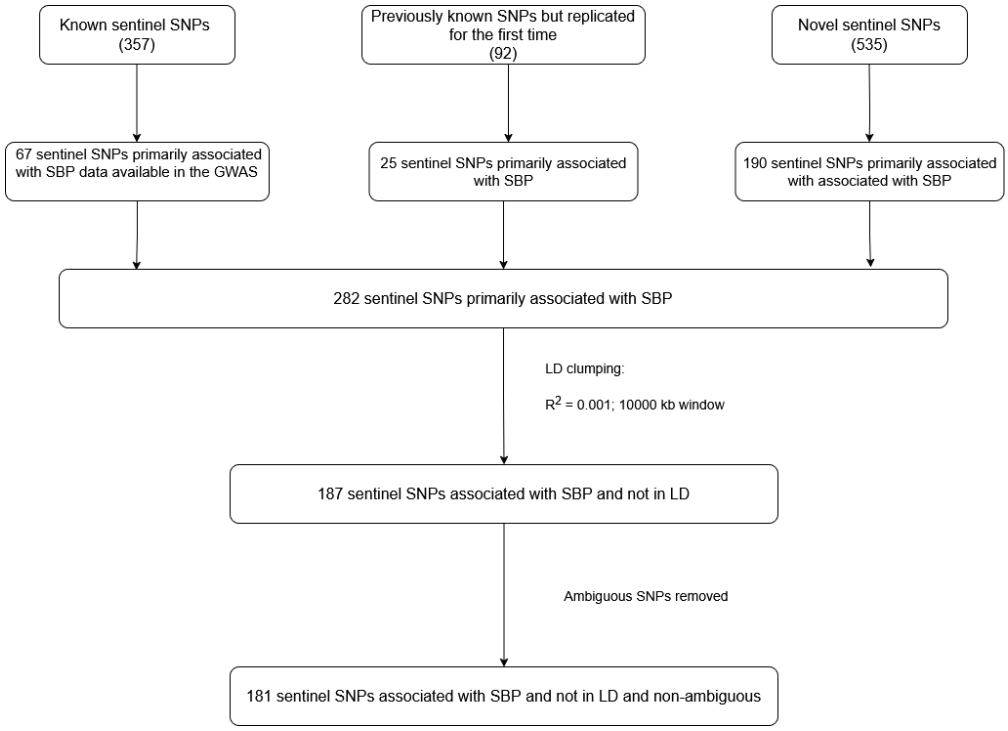

Figure 1: Flow chart showing selection criteria for genetic instruments primarily associated with systolic blood pressure

4.7.2.2 Figure 2

Flow chart showing selection criteria for genetic instruments primarily associated with diastolic blood pressure

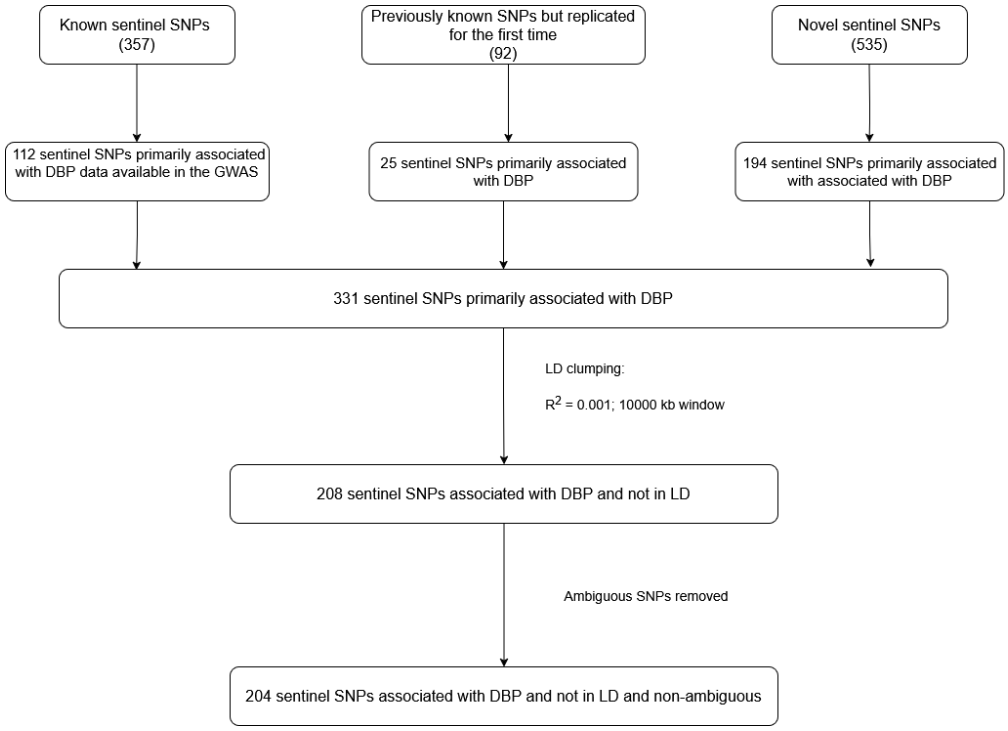

Figure 2: Flow chart showing selection criteria for genetic instruments primarily associated with diastolic blood pressure

4.7.2.3 Figure 3

Figure showing main analysis, and sub-group analyses by age, sex and PRS-free SBP.

```

#R plots

getwd()

#Folder paths
wd_path = "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/r_codes/r_data"
data_path = "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/r_codes/r_data"
png_plot_path = "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/stata_sbp_plot/png"
pdf_plot_path = "C:/Users/tabe0010/OneDrive - Monash University/MR_backup_file/Articles/Evangelou/new_sbp_snps/stata/stata_sbp_plot/pdf"

#Create MR graphs from main analysis
setwd(wd_path)

#install.packages("TwoSampleMR", repos = c("https://mrcieu.r-universe.dev", "https://cloud.r-project.org"))
#install.packages("devtools")
#devtools::install_github("MRCIEU/MRInstruments")
#install.packages("data.table")

library(TwoSampleMR)
library(MRInstruments)
library(data.table)
library("ggplot2")
library(plyr); library(dplyr)
#install.packages("forestplot")
library(forestplot)

#########################################################################

#while (dev.cur() > 1) dev.off()
master_data = read.csv("metan_sbp_r.csv",stringsAsFactors = FALSE)
qaly_hes<-master_data%>%filter(outcome=="qaly_hes")

x_list_qaly_1 = unique(qaly_hes$outcome)
qaly_hes = qaly_hes[order(qaly_hes$outcome,qaly_hes$type),]
tabletext_1 = cbind(unique(qaly_hes$label),(paste(qaly_hes$effect[qaly_hes$type=="Main Analysis MR"],"\n","\n", qaly_hes$effect[qaly_hes$type=="Multivariable Adjusted"],sep="")))
colnames(tabletext_1)<-c("variable", "effect")
hrzl_lines = list("1"=gpar(lty=0), "2"=gpar(lty=1),"3"=gpar(lty=1),"4"=gpar(lty=2),"5"=gpar(lty=1),"6"=gpar(lty=2),
                  "7"=gpar(lty=2),"8"=gpar(lty=2),"9"=gpar(lty=2), "10" = gpar(lty=1), "11" = gpar(lty=2), "12"=gpar(lty=2))
```

```r


#png("qaly_hes.png", width = 1000, height = 1000)
#pdf("qaly_hes.pdf", width = 12, height = 12)
png(file.path(png_plot_path, "qaly_hes.png"), width = 1000, height = 1000)
#pdf(file.path(pdf_plot_path, "qaly_hes.pdf"), width = 12, height = 12)

forestplot(tabletext_1,
           legend = c("Main Analysis MR","Multivariable Adjusted"),
           title = "QALYs per year",
           mean = cbind(qaly_hes$beta[qaly_hes$type == "Main Analysis MR"
              ], qaly_hes$beta[qaly_hes$type == "Multivariable Adjusted"])
              ,
           lower = cbind(qaly_hes$lower[qaly_hes$type == "Main Analysis MR
              "], qaly_hes$lower[qaly_hes$type == "Multivariable Adjusted"
              ]),
           upper = cbind(qaly_hes$upper[qaly_hes$type == "Main Analysis MR
              "], qaly_hes$upper[qaly_hes$type == "Multivariable Adjusted"
              ]),
           col=fpColors(box=c("blue", "darkred"),
                        zero=c("darkblue")),
           boxsize = 0.1,
           line.margin = 0.2,
           #xticks =  c(-5,-4,-3,-2,-1,0,1,2,3),
           xticks =  c(-3,-2,-1,0,1,2,3),
           grid = FALSE,
           hrzl_lines=hrzl_lines,
           txt_gp = fpTxtGp(xlab=gpar(cex=1.2),
                            ticks = gpar(cex=1),
                            summary = gpar(cex=1.5),
                            title = gpar(cex=1.75),
                            legend = gpar(cex=1.3),
                            label = list(gpar(cex=1.2),gpar(1.5),gpar(cex
                               =0.8))),
           xlab = "Percentage change in QALYs for 10mmHg increase in SBP"


) %>%
  fp_add_header(effect = "Estimate (95% CI)")  # Add custom header
dev.off()
```

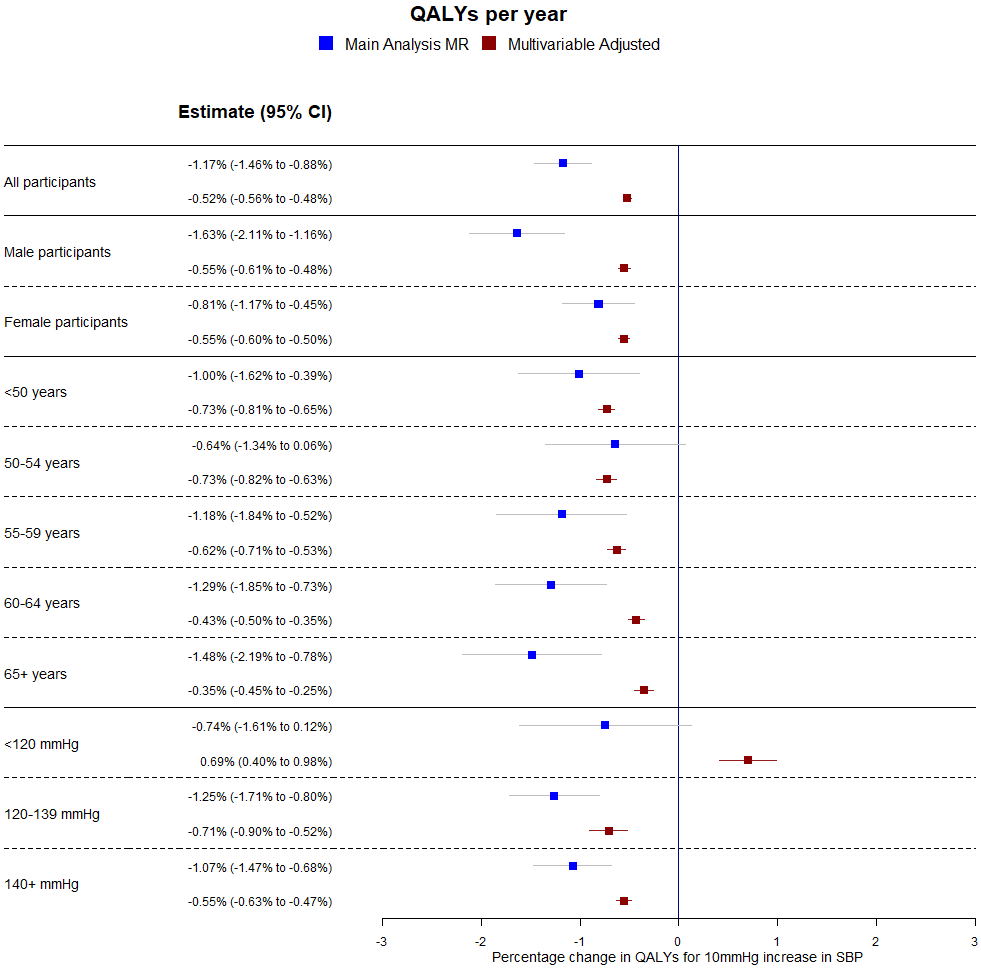

Figure 3: Mendelian randomisation estimates for QALYs on average year of follow-up

4.7.2.4 Figure 4

Associations between SNPs associated with systolic blood pressure and quality adjusted life years.

```
1
2  #R plots
3  # We will continue working on the previous R plot environment
4
5  dat <- read.csv("mr_analysis_sbp.csv", stringsAsFactors = FALSE)
6  dat <- rename(dat, beta.outcome = beta_outcome, se.outcome = se_outcome,
7                pval.outcome = pval_outcome, beta.exposure = beta_exposure,
8                se.exposure = se_exposure, id.exposure = id_exposure, id.
                     outcome = id_outcome)
9
10 dat$mr_keep <- TRUE
11 dat$exposure <- "SBP"
12
13 # MR analysis
14 setwd(paste(wd_path, sep=""))
15 res <- mr(dat)
16 p1 <- mr_scatter_plot(res, dat)
17
18 x = res[res$method == "Inverse variance weighted" | res$method == "Wald
       ratio",c("outcome","exposure")]
19
20 # Update axis labels, background theme, move legend to bottom, and remove
       grid for each plot
21 for(i in 1:length(p1)){
22   outcome <- res[res$method == "Inverse variance weighted" | res$method ==
         "Wald ratio", "outcome"][i]
23
24   # Customize the plot with new labels, white background, legend at the
       bottom, and no grid
25   p1[[i]] <- p1[[i]] +
26     labs(x = "SNP effect on SBP", y = "SNP effect on QALYs") +  # Change
         these to your desired labels
27     theme_bw() +  # Set background theme to white
28     theme(legend.position = "bottom",  # Move legend to bottom
29           panel.grid = element_blank())  # Remove grid
30
31   # Save the modified plot
32   #ggsave(p1[[i]], file=paste("IVW - ", outcome, ".png", sep=""), width=7,
         height=7)
33   ggsave(p1[[i]], filename = paste(png_plot_path, "/IVW - ", outcome, ".
       png", sep=""), width=7, height=7)
34 }
```

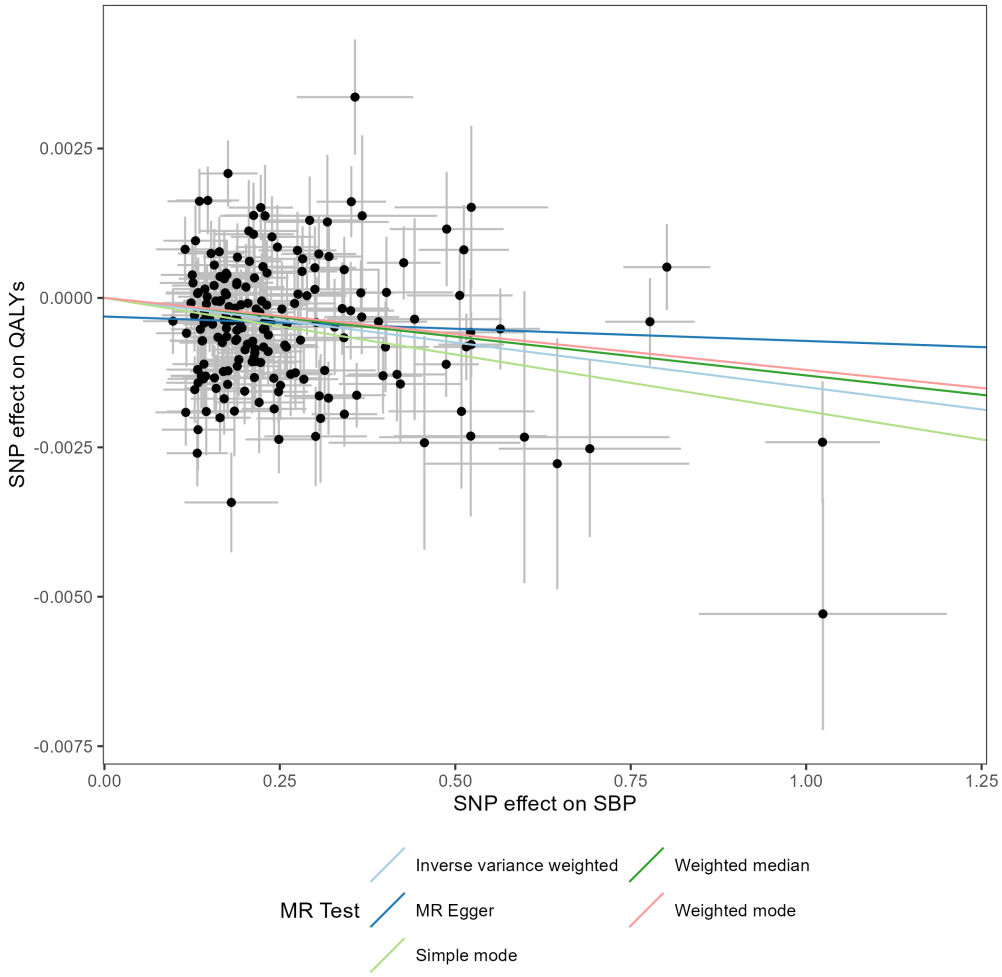

Figure 4: Associations between SNPs associated with systolic blood pressure and quality adjusted life years.

4.7.2.5 Figure 5

Non-linear MR showing the estimated effects of 1 mmHg increase in SBP on QALYs over an average year of follow-up, across SBP levels.

A positive value indicates an increase in SBP would increase in QALYs, and vice versa. There was little evidence of nonlinearity in the effect of SBP on QALYs. The SBP thresholds of 120 mmHg (for pre-hypertension SBP) and 140 mmHg (hypertension) are represented with dashed red lines. The green shaded area represents the 95

The code for this figure already done in **Section 4.5.4**.

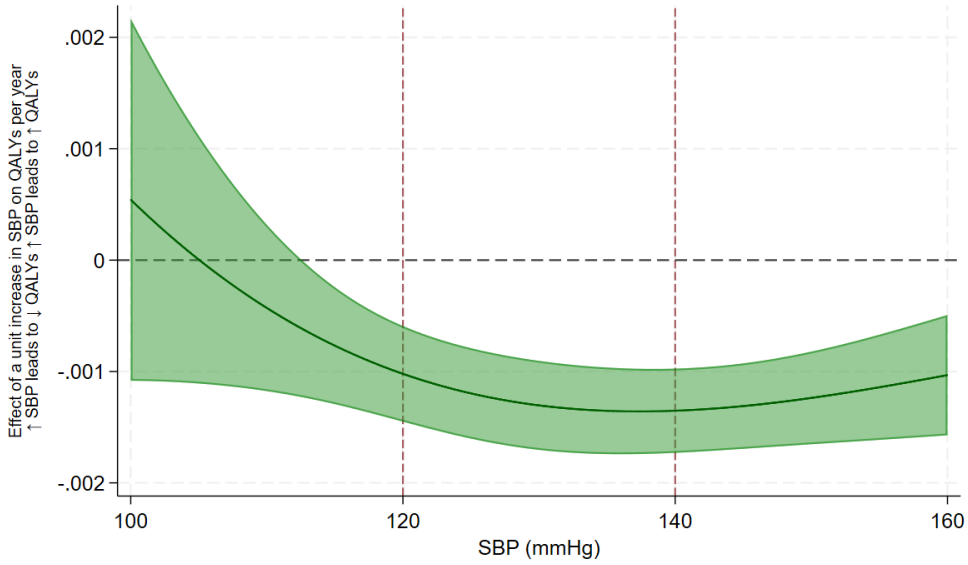

Figure 5: Estimated effects of 1 mmHg increase in SBP on QALYs over an average year of follow-up, across SBP levels.

References

[1] Sean Harrison, Padraig Dixon, Hayley E Jones, Alisha R Davies, Laura D Howe, and Neil M Davies. Long-term cost-effectiveness of interventions for obesity: A mendelian randomisation study. *PLoS medicine*, 18(8):e1003725, 2021.

[2] Jonathan C Brown, Thomas E Gerhardt, and Edward Kwon. Risk factors for coronary artery disease. `https://www.ncbi.nlm.nih.gov/books/NBK554410/`, 2023. Accessed 20-03-2025.

[3] Seamus P Whelton, John W McEvoy, Leslee Shaw, Bruce M Psaty, Joao AC Lima, Matthew Budoff, Khurram Nasir, Moyses Szklo, Roger S Blumenthal, and Michael J Blaha. Association of normal systolic blood pressure level with cardiovascular disease in the absence of risk factors. *JAMA cardiology*, 5(9):1011–1018, 2020.

[4] Brian A Ference, Deepak L Bhatt, Alberico L Catapano, Chris J Packard, Ian Graham, Stephen Kaptoge, Thatcher B Ference, Qi Guo, Ulrich Laufs, Christian T Ruff, et al. Association of genetic variants related to combined exposure to lower low-density lipoproteins and lower systolic blood pressure with lifetime risk of cardiovascular disease. *Jama*, 322(14):1381–1391, 2019.

[5] Eric Yuk Fai Wan, Wing Tung Fung, C Mary Schooling, Shiu Lun Au Yeung, Man Ki Kwok, Esther Yee Tak Yu, Yuan Wang, Esther Wai Yin Chan, Ian Chi Kei Wong, and Cindy Lo Kuen Lam. Blood pressure and risk of cardiovascular disease in uk biobank: a mendelian randomization study. *Hypertension*, 77(2):367–375, 2021.

[6] Debbie A Lawlor, Roger M Harbord, Jonathan AC Sterne, Nic Timpson, and George Davey Smith. Mendelian randomization: using genes as instruments for making causal inferences in epidemiology. *Statistics in medicine*, 27(8):1133–1163, 2008.

[7] Patrick W Sullivan, Julia F Slejko, Mark J Sculpher, and Vahram Ghushchyan. Catalogue of eq-5d scores for the united kingdom. *Medical Decision Making*, 31(6):800–804, 2011.

[8] Cathie Sudlow, John Gallacher, Naomi Allen, Valerie Beral, Paul Burton, John Danesh, Paul Downey, Paul Elliott, Jane Green, Martin Landray, et al. Uk biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS medicine*, 12(3):e1001779, 2015.

[9] Martin D Tobin, Nuala A Sheehan, Katrina J Scurrah, and Paul R Burton. Adjusting for treatment effects in studies of quantitative traits: antihypertensive therapy and systolic blood pressure. *Statistics in medicine*, 24(19):2911–2935, 2005.

[10] Nancy J Devlin, Koonal K Shah, Yan Feng, Brendan Mulhern, and Ben Van Hout. Valuing health-related quality of life: An eq-5 d-5 l value set for e ngland. *Health economics*, 27(1):7–22, 2018.

[11] Stephen Burgess and Simon G Thompson. *Mendelian randomization: methods for using genetic variants in causal estimation*. CRC Press, 2015.

[12] Neil M Davies, Michael V Holmes, and George Davey Smith. Reading mendelian randomisation studies: a guide, glossary, and checklist for clinicians. *bmj*, 362, 2018.

[13] Evangelos Evangelou, Helen R Warren, David Mosen-Ansorena, Borbala Mifsud, Raha Pazoki, He Gao, Georgios Ntritsos, Niki Dimou, Claudia P Cabrera, Ibrahim Karaman, et al. Genetic analysis of over 1 million people identifies 535 new loci associated with blood pressure traits. *Nature genetics*, 50(10):1412–1425, 2018.

[14] George Davey Smith and Shah Ebrahim. 'mendelian randomization': can genetic epidemiology contribute to understanding environmental determinants of disease? *International journal of epidemiology*, 32(1):1–22, 2003.

[15] James Durbin. Errors in variables. *Revue de l'institut International de Statistique*, pages 23–32, 1954.

[16] De-Min Wu. Alternative tests of independence between stochastic regressors and disturbances. *Econometrica: journal of the Econometric Society*, pages 733–750, 1973.

[17] Jerry A Hausman. Specification tests in econometrics. *Econometrica: Journal of the econometric society*, pages 1251–1271, 1978.

[18] Clare Bycroft, Colin Freeman, Desislava Petkova, Gavin Band, Lloyd T Elliott, Kevin Sharp, Allan Motyer, Damjan Vukcevic, Olivier Delaneau, Jared O'Connell, et al. The uk biobank resource with deep phenotyping and genomic data. *Nature*, 562(7726):203–209, 2018.

[19] William G Cochran. The combination of estimates from different experiments. *Biometrics*, 10(1):101–129, 1954.

[20] UK Biobank. Pain web questionnaire version 2.1, 2022. Accessed: 30-Mar-2025.

[21] UK Biobank. Mental well-being web questionnaire version 1.1, 2023. Accessed: 30-Mar-2025.

[22] Stephen Burgess and Simon G Thompson. Multivariable mendelian randomization: the use of pleiotropic genetic variants to estimate causal effects. *American journal of epidemiology*, 181(4):251–260, 2015.

[23] Eleanor Sanderson, George Davey Smith, Frank Windmeijer, and Jack Bowden. An examination of multivariable mendelian randomization in the single-sample and two-sample summary data settings. *International journal of epidemiology*, 48(3):713–727, 2019.

[24] Eleanor Sanderson. Multivariable mendelian randomization and mediation. *Cold Spring Harbor perspectives in medicine*, 11(2):a038984, 2021.

[25] Helen R Warren, Evangelos Evangelou, Claudia P Cabrera, He Gao, Meixia Ren, Borbala Mifsud, Ioanna Ntalla, Praveen Surendran, Chunyu Liu, James P Cook, et al. Genome-wide association analysis identifies novel blood pressure loci and offers biological insights into cardiovascular risk. *Nature genetics*, 49(3):403–415, 2017.