***ICD-10 Mapping Steps***

1. An initial translation was obtained by applying mappings from the 2018 CMS GEMs and the CMS Chronic Conditions Warehouse (CCW) (14) to map ICD-9 codes for each health deficit in VA-FI-9 to ICD-10 codes.

2. The initial translation was validated by manual review of whether candidate ICD-10 codes were correctly classified to each health deficit.

3. The [full ICD-10 hierarchy and list of active CPT codes](https://github.com/bostoninformatics/va_frailty_index/blob/main/icd_code_vafi_mapping.csv) in the VHA Corporate Data Warehouse (CDW) were then manually reviewed to check whether additional ICD, CPT, or HCPCS codes should be included in the VA-FI-10.

***The updated VA-FI-10 includes 31 health deficits:***

* Atrial fibrillation
* Anemia
* Anxiety
* Arthritis
* Coronary artery disease
* Cancer
* Chronic pain
* Cerebrovascular disease
* Dementia
* Depression
* Diabetes
* Durable medical equipment
* Falls
* Fatigue
* Failure to thrive
* Gait abnormality
* Hearing impairment/loss
* Heart failure
* Hypertension
* Incontinence
* Kidney disease
* Liver disease or cirrhosis
* Lung disease (COPD, asthma)
* Muscular issue
* Osteoporosis
* Parkinson's disease
* Peripheral neuropathy
* Peripheral vascular disease
* Thyroid disease
* Vision comorbidity
* Weight loss

***VA-FI yields the following 5 categories of frailty (VA-FI = sum of the above health deficits / 31):***

* non frail (VA-FI ≤ 0.1),
* prefrail (>0.1–0.2),
* mildly frail (>0.2–0.3),
* moderately frail (>0.3–0.4),
* severely frail (>0.4)

***Programming code and ICD mappings are available on GitHub*** - [bostoninformatics/va\_frailty\_index](https://github.com/bostoninformatics/va_frailty_index)

Calculating the VA-FI requires the following steps:

* Step 1: Defining an index date and lookback period (typically 3 years) relative to which the VA-FI will be calculated.
* Step 2: Pulling diagnosis and procedure codes needed from the data source (within the VA, this is typically the VA Corporate Data Warehouse and possibly VA's CMS data).
* Step 3: Actually calculating the VA-FI based on the above.

This repo provides code to execute Step 3. Step 1 is inherently project specific and must be done by you. For Step 2, there are two options:

* Step 2, Option 1: You may pull diagnosis and procedure codes in a superset of the lookback period using your favorite method, e.g., writing a query in SQL Server Management Studio and downloading the results to a CSV file.
* Step 2, Option 2: For VA employees or WOCs, we have written a very efficient push-button code to do steps 2 and 3 together, which is available at <https://github.ec.va.gov/EPMO/va_frailty_index>. We can also email you this code at your va.gov email address. *This is the recommended option for internal VA users.*

In addition, if you'd like to reimplement Step 3 yourself for some reason, CSV versions of the code sets are available in icd\_code\_vafi\_mapping.csv and procedure\_code\_vafi\_mapping.csv.

* *Note: Exercise caution if you edit these files in Excel. Opening the file and saving in Excel will likely corrupt ICD codes beginning or ending in 0s (e.g., 035.40 will be incorrectly rewritten as 35.4).*

For further information or assistance with this code, please reach out to the authors, which for senior authors Ariela Orkaby and Nathanael Fillmore are both Firstname.Lastname@va.gov.

*# The following should be edited as appropriate – see documentation of the make\_vafi and pull\_icd\_and\_procedure\_codes functions.*

conn <- odbcConnect("vhacdwrb03") *# or vhacdwrb02, as appropriate for the study*

database <- “ORD\_PIName\_2019XXXXXX”

patient\_tb <- read.csv(“vafi\_patient\_tb.csv”) *# table with one row per patient and two columns, PatientICN and IndexDate*

lookback\_days <- 3\*365

tmp\_dir <- “vafi\_tmp\_dir”

vafi\_code\_path <- “.” *# location of directory where icd\_prcd\_vafi.fst is stored*

vafi\_boolean <- pull\_icd\_and\_procedure\_codes(conn, database, patient\_tb, lookback\_days, tmp\_dir, vafi\_code\_path)

vafi <- make\_vafi (patient\_tb, vafi\_boolean, vafi\_code\_path)

write.csv(vafi, file = "vafi.csv", row.names = FALSE)