A Retirement Planner Model User’s Guide

Introduction: In this document we will describe the objectives for using ARetirementPanner, some simple cases and how tos. For this there is a section on the models input specification as well as a section on the models output.

Get your feet wet: In this section we will describe a very basic application of ARetirementPlanner.

Invoking the model:

Model input specification:

The basis for the model input specification is the toml specification[[1]](#footnote-1) with its library. The basic format is sections of information ‘[‘ section name ‘]’ where section name can be a category followed by a ‘.’ and a dispriptive name. ‘#’ to the end of the line represents a comment. The global section has no section name header. The input information is represented by an assignment.

General model information (global section):

retirement\_type = 'joint' # defaults to joint, could be single, joint (married), ??? TODO

returns = 6 # defaults to 6.0, return rate of investments as a percent

inflation = 2.5 # defaults to 0, yearly inflation rate as a percent

#maximize = ‘PlusEstate’ # defaults to ‘Spending’, ‘PlusEstate’ maximizes the final estate.

# individual retiree info (one section per retiree, at least one will need an id)

[iam] or [iam.id] where id is some string that represents the retiree

age = 60 # retiree’s age

retire = 65 # age retiree will retire

through = 95 # age through which to plan

primary = true # if there are more than one retiree one should be designated as primary

# Social Security section must specify amount, FRA and an age range

[SocialSecurity] or [SocialSecurity.id] # if more than one retiree at least one needs an id (id should match retiree id)

amount = 31000 # $31,000 at Full Retirement Age (FRA); Assumes inflation, 85% taxed

FRA = 67

age = "68-"

# Income must specify amount (yearly), age range, whether to adjust for inflation, and if it is taxed

[income.mytaxfree] # income after retirement that does not involve retirement accounts

amount = 3000

age = "67-" # starts at age 67 and continue

inflation = false # Adjust for inflation

tax = false # count this as ordinary taxable income (true/false)

[income.rental\_1] # another income source

amount = 36000

age = "67-"

inflation = true

tax = true # count this as income tax

[income.rental\_2] # a third income source

amount = 2400

age = "67-"

inflation = true

tax = true # count this as income tax

# desire income section must define amount, age range, inflation adjustment and whether it is taxed

# desired income should be use in conjunction with ‘PlusEstate’

#[desired.income]

#amount = 45000 # per year

#age = "68-"

#inflation = true

#tax = true # count this as income tax

# max income section must define amount, age range, inflation adjusted

#[max.income]

#amount = 150000

#age = "68-"

#inflation = true

## sections for each account type, IRA and Roth types should have on per retiree as needed

## only one pre-tax account

## Each accounts must define the initial balance. Optionally, an investment return rate can be given

[IRA] or [IRA.id] # where id must match the retiree’s id who is the owner of the account

bal = 2000000

#rate = 7.25

maxcontrib = 00

# roth type accounts

[roth.spouse]

bal = 100000

maxcontrib = 000

# after tax savings/investment type accounts

[aftertax] # assumes joint ownership if multiple retirees

bal = 700000

basis = 400000 # for capital gains tax

Model output specification:

1. The toml specification is archived at: <https://github.com/toml-lang/toml> [↑](#footnote-ref-1)