SMM040 – (Group) Coursework No. 3: Sickness-Linked Annuity Model VBA/Excel Program

Introduction

A life insurance company is planning to set up an annuity product that pays out subject to the current level of health of the policyholder. The company is considering three health states to be applicable in this product: ***healthy***, ***light-sickness*** and ***heavy-sickness***. The company intends to investigate the feasibility of the product by a four-state Markov model (i.e. only the current age and state of a life affect the future transition rates), which is made up by three transition states (representing the alive states) and an absorbing state (representing death). Thus, the aim of the investigation is to model and simulate the population dynamics in these four states based on pre-determined transition rates between the Markov states at each age for an overall age range of 65 to 120.

Transitions in the Markov model are calculated as follows:

- All lives are assumed to have their birthday on the 1st January.
- The starting population at age 65 should be allocated as follows:
  a) Starting lives at age $x = 65$ are allocated (randomly) to one of the health states based on some prevalence rates (set by the user);
  b) Consequently, the user is able to set a 100% prevalence rate for any given health state and investigate the model assuming all lives start in the same state at the outset.
- All transitions are assumed to occur halfway through the age-year.
- First, all lives are exposed to mortality risk (i.e. all lives alive in any health state are assumed to have an initial risk of death before any other transition possibilities). That is, all transition probabilities at the current age x are conditional on survival to age x. Thus, first you need to test (through a random simulation based on the probability of death) whether or not the current life survives, before simulating any other transition possibilities.
- Then, those lives that survive (i.e. do not move to the death state) are subject to moving into an alive/health state (i.e. the sickness/recovery rates are applied after mortality). All transitions to a health states are seen as acting at the same time and the total transition can be seen as simply adding up the individual health transitions (see attached methodology).
- During any age-year $(x, x+1)$, a life in a given state can transition to other states (including staying in the same state), subject to the constraints:
  (a) Only one transition is allowed each year;
  (b) It is not possible to return to ***healthy*** from ***heavy-sickness*** during a single year (e.g. it requires a transition through ***light-sickness***);
  (c) The ***death*** state is an absorbing state.

Thus, the company is interested to forecast the population numbers in the three health-states at each age $x \in \{65, 120\}$ using stochastic simulations of the above Markov model. Correspondingly, they want to estimate:

(a) The life-expectancy (i.e. average number of years alive); and

(b) The healthy-life-expectancy (i.e. average number of years alive and healthy)

at age $x = 65$ conditional on the population makeup at the outset (i.e. assumed prevalence rates).

Coursework Tasks

You therefore need to build an Excel macro-enabled workbook that will allow mortality rates and disability rates to be entered (by the means of appropriate spreadsheet tables) so that rates can be tested and adjusted. Further, you should design a VBA program that can be run interactively by a user. The program should be able to read the transition rates, run the required number of simulations and output the results.[1]

Thus, the user should be able to run the Markov model and can choose between:

1. A single deterministic run (i.e. $l_{65} = 1$), which will output the probabilities that a life will occupy each of the states at consecutive ages $x > 65$ (i.e. like in a **deterministic** model). Thus, it corresponds to reproducing the multiple decrement life table (as given) by the means of VBA coding (e.g. avoid using WorksheetFunction(s))
2. Up to 10,000 stochastic runs (i.e. $1 < l_{65} \leq 10{,}000$). In this case the output should summarise the total number of years spent in each state for each age and calculate the predicted life expectancy. The summary data should be automatically graphed in a suitable way (or selected by the user).

   Hint: To calculate the total years spent in any given state you can do it during run time separately for each life (keeping in mind that transitions occur halfway through the year) or at the end of all the runs use the census method on the generated life table (see CT4)

Type of Users

When designing this program there are three types of *users* you should have in mind:

Junior user – They will only need to run the code by selecting the number of simulations and be able to collect the output. Therefore the workings of the model is not their primary concern (e.g. might even be hidden from them).

Senior user – They will need to be able to change the rates if they are updated, but (similarly to the junior user) they will not want to look at the VBA code. Changing the rates should therefore be obvious and not require knowing how the code works.

Programmer – They will need to access the VBA code in order to maintain and update it. They will require documentation to allow them to do this. You can assume they have actuarial knowledge so it is not necessary to go into details about the fundamentals and theory of the Markov model, only about how this code works.

[1] A Userform type interface is not required for this task (although some bonus marks might be given for such extra work).

Assessment

The assessment of your coursework will be done by assuming the roles of each of the above users, giving consideration to the following:

1. As a **junior** user, I will run the code and will make sure that it does what it is supposed to do. Marks will be given for the code running correctly (!) but also that it is easy for a user to use it correctly. That is, the code is ran and the output is generated easily and without errors and is straightforward to interpret.

   [15 marks]

2. As a **senior** user, I will then update the mortality and sickness rates and rerun the code. Marks will be given for the code running correctly after the update and for the ease of carrying out this update.

   [15 marks]

3. As an incompetent/malicious user (both **junior** and **senior**), I will then try to break the program. In this case, marks will be awarded based on how robust the program proves to be. Therefore, I will test how likely the program is to fail/crash when in use (e.g. you can think of it as marks will be deducted if I can break the running of the code).

   [20 marks]

4. Finally, as a **programmer** user, I will need to understand how the code works and to update it with new features and instructions. In this case it will be assessed how easy it is to do so based on your code, comments and documentation provided. Note that excess documentation and comments to the VBA code can be just as confusing as none at all. So striking an appropriate balance is going to be crucial!

   [50 marks]

[Total 100 marks]

Transition Rates Data

The transition rates between the states of the Markov model which you need to make use of are given in an Excel workbook on Moodle (see '**Transition Rates**' worksheet). In addition, the file also contains a corresponding deterministic life table model (see '**Lifetable**') which may be useful to you when checking the output of your model (e.g. in the case of the single run). However, you should only be transferring the transition rates into your program not the deterministic life table model!

Submission

Submit on Moodle a macro enabled Excel WB with your program and, optionally, a Word or pdf document with the description/instructions (alternatively, these can also be given in a separate worksheet of the workbook). There should be only a single submission by each group before the published deadline (see Moodle).

Please note that submissions after the deadline will not be accepted.