Australian Government
Productivity Commission

FERTMOD: Software for making demographic projections under different fertility assumptions

Technical Paper

Ralph Lattimore

June 2008

This publication is available in PDF format from the Productivity Commission website at www.pc.gov.au. If you require part or all of this publication in a different format, please contact Media and Publications (see below).

Publications Inquiries:
Media and Publications
Productivity Commission
Locked Bag 2 Collins Street East
Melbourne VIC 8003

Tel: (03) 9653 2244
Fax: (03) 9653 2303
Email: maps@pc.gov.au

General Inquiries:
Tel: (02) 6240 3200 or (03) 9653 2100

An appropriate citation for this paper is:

Lattimore, R. 2008, *FERTMOD*: Software for making demographic projections under different fertility assumptions, Technical Paper, Productivity Commission, Canberra, June.

1.1 What is FERTMOD?

FERTMOD is an Excel-based program that projects Australia's population, given assumptions about fertility, mortality, and net migration.[1] It is simple to use and requires little input from users. In many cases, users can simply enter one parameter to conduct experiments — but it also allows considerable flexibility in making demographic projections.

In most instances, users should be able to run the program in Excel without changing their Excel settings. However, in testing we sometimes found users encountered a message 'Can't find project or library'. In that instance, please see section 1.7 at the end of this document about how to solve this problem.

1.2 Purpose of the model

The goal of the model is to indicate how changes in the model's demographic assumptions affect Australia's population and its age structure. FERTMOD uses the standard cohort component model. This is a deterministic model, which for given settings for mortality, net migration, fertility and a starting population, will exactly calculate future population numbers.

At this point, it is important to distinguish projections from forecasts.

Projections

Projections do not aim to predict the future, so that there is no particular view that the underlying settings will be realised. In that sense, it is a demographic 'laboratory' that allows users to conduct 'what if' experiments. For example, what would happen were government to significantly increase migration intakes or to raise fertility through family policies? The outcomes of these experiments may inform policymakers about whether a given set of policies are sensible. That information will mean that those settings are not actually used, thus preventing the realisation of that projection. In its work on projections in its analysis of the

[1] The Commission has also developed a much more sophisticated projection tool, MoDEM, which allows users even greater control over their demographic scenarios, and which also projects key economic variables, such as labour force participation rates and GDP. However, because of its sophistication, this model is larger and requires more knowledge to run properly than FERTMOD. MoDEM and its documentation can be found on the Commission's website at www.pc.gov.au/research/commissionresearch/nationalreformagenda/modem.

implications of population ageing, the Productivity Commission (2005) gave an analogy to this. A large rock is lying on a rail track and 10 kilometres away a train is hurtling towards a collision. The projection is that the train will crash, with all of its tragic consequences. The forecast, taking account of this projection, is that the train will stop and the rock removed from the tracks. No collision occurs. The lack of realisation of the projection is not a fault of the method, but in this context, an obvious strength.

It is important to undertake projections over a long horizon to identify ‘steady-state’ outcomes (when the age structure and population growth has stabilised). Hence, FERTMOD is a long-run model, which provides projections on a year-by-year basis to 2251. The likelihood of realising projections as far forward as 2251 is obviously low. This is because the economic and social circumstances are likely to be fundamentally different from today, altering the mortality, net migration and fertility settings underpinning these projections. Nevertheless, the projections show what *would* happen were the government to preserve its policy settings over the long run, and this can usefully indicate their sustainability and realism. For example, a very substantial increase in family policy might raise fertility above replacement levels to 2.2 babies per woman. With net migration of say, 170 000 per year, population growth and long-run population numbers would be very high. This might suggest that the policy settings underpinning the high fertility or/and net migration could not realistically be sustained.

Forecasts

Forecasts attempt to predict what will actually transpire – taking into account future government policies (for example, migration policy). Accurate forecasts of population numbers can only be made over relatively short horizons. Users can employ FERTMOD to produce such forecasts if they are confident about the accuracy of their underlying assumptions about mortality, fertility and migration.

Even in the presence of uncertainty, FERTMOD can indicate some likely bounds on Australia’s future demographic characteristics by undertaking scenario analysis. For example, users may be reasonably certain about the likely maximum and minimum life expectancy outcomes over the next fifty years, and can then see, for given fertility and migration settings, what this implies for Australia’s future age structure. These are not point forecasts, but they can enable users to test whether certain qualitative outcomes are likely or not – such as the probable magnitude of population ageing. Judicious use of scenario analysis of this kind can emulate some of the advantages of stochastic forecasts.

1.3 The model's options

The model is relatively flexible. Users can specify:

- long-run total fertility rates (which, by definition, are also completed fertility rates);
- life expectancy of males and females (separately);
- net migration levels;
- the transition periods to these long-run values;
- the shape of the age distribution of fertility; and
- whether net migration is endogenous. In this case, users specify a long-run population target and then net migration levels adjust to achieve that target (given all other demographic assumptions). This is an appropriate experiment where governments alter migration policy to achieve a sustainable long run population.

1.4 How to use the model

On opening the FERTMOD excel spreadsheet, there are two (default) worksheets.

The 'Start' worksheet

The most important sheet is the Start worksheet, and this is the one that the program opens in. (The other worksheet – the Input worksheet – is described later, but most users need never look at it.)

The Start worksheet (figure 1.1) serves two purposes:

- It *documents* the key inputs that feed into the experiments (the 'key inputs' part of the sheet) and results of each experiment (the Key Results area of the sheet, including the long-run age-specific fertility rates and a graph showing how they have changed); and
- Through the 'Click here to Start' button, it provides the entry point to the model – where users can specify the parameters they wish to use and the form of outputs generated by the model.

Figure 1.1 **The Start worksheet**

The top part of the sheet

Key inputs used (shows parameter values used in model)

Click here to start program

Long run population (number)	Population endogenous
TFR	3.00
Transition to long run TFR	30.0
Male life expectancy	95.0
Female life expectancy	98.0
Transition to long run life expectancy (males)	100.0
Transition to long run life expectancy (females)	100.0
Annual net migration	135000
Transition to long run net migration	15.0
Lambda (adjusts shape of fertility distribution due to future post	1.0
Theta (adjusts shape of fertility distribution for ΔTFR)	1.0

Key results

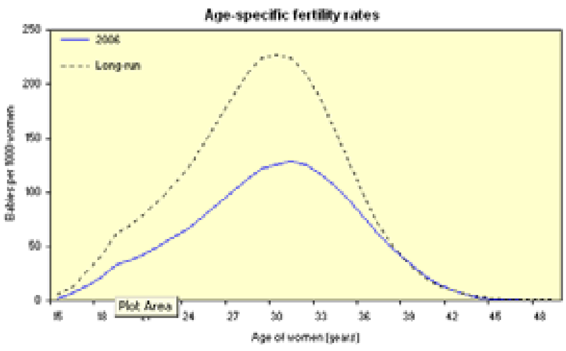
End June	2007	2051	2151	2251
Long run population (millions)	21.0	39.4	173.7	668.3
Long run 65+ share (%)	13.1	21.4	19.9	20.2
Long run <15 share (%)	19.4	23.8	24.9	24.8
Long run total dependency (%)	48.2	82.3	81.1	81.8
Long run 65+ dependency (%)	19.5	39.0	36.1	36.8
Long run <15 dependency (%)	28.7	43.3	45.1	45.1
Long run 25-55 share (%)	43.7	33.1	33.7	33.5
Long run population growth (%)	na	1.4	1.4	1.3
Overseas born share of population (%)	23.8	20.9	5.3	1.4
Prime aged (15-64) millions	14.2	21.6	95.9	367.5
Aged 65+ millions	2.8	8.4	34.6	135.1
Young (<15) millions	4.1	9.4	43.2	165.7

Start / Inputs

The bottom part of the sheet

Age-specific fertility

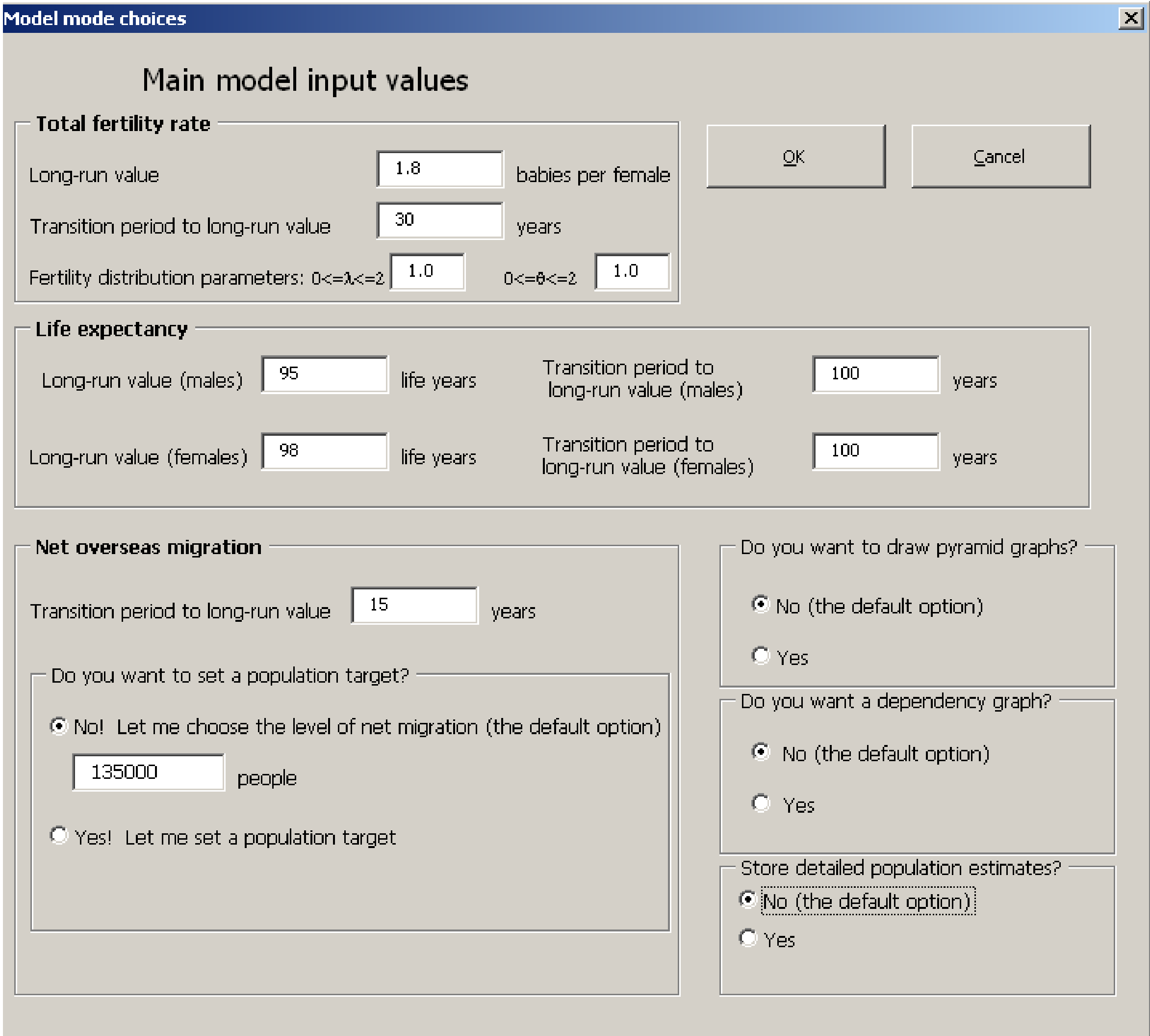
Age	ASFR in 2006	Long-run ASFR	Shift in S distribution due to timing	Shift in S distribution due to DTFR
15	2.4744	5.1635	0.7387	1.4087
16	6.2650	12.7497	0.6863	1.4787
17	13.3103	26.3383	0.6458	1.5278
18	21.7571	41.9203	0.6169	1.5575
19	33.0693	62.3148	0.5988	1.5692
20	38.0263	70.5165	0.5911	1.5644
21	43.8877	80.6387	0.5932	1.5446
22	50.7198	92.9213	0.6045	1.5112
23	58.4203	107.2509	0.6246	1.4656
24	66.7821	123.2328	0.6528	1.4095
25	78.3493	145.4463	0.6887	1.3442
26	89.7304	167.3373	0.7316	1.2711
27	101.2267	188.9611	0.7810	1.1918
28	112.7554	209.5143	0.8364	1.1078
29	122.3636	224.6662	0.8972	1.0204
30	125.9896	226.5590	0.9629	0.9312
31	128.7414	224.4536	1.0329	0.8416
32	125.0972	209.1161	1.1067	0.7532
33	117.0052	185.3374	1.1838	0.6673
34	105.6150	156.6466	1.2634	0.5854
35	91.8410	126.1127	1.3453	0.5090
36	76.8274	96.7560	1.4287	0.4396
37	61.3413	70.4656	1.5131	0.3786
38	47.3554	49.6978	1.5981	0.3275
39	35.3085	34.2879	1.6829	0.2877
40	24.1471	22.3207	1.7672	0.2608
41	15.5601	14.3315	1.8503	0.2482
42	9.8192	9.5618	1.9316	0.2514



Start / Inputs

On clicking the 'Click here to Start' users see a variety of model options (figure 1.2).

Figure 1.2 **Setting model options**



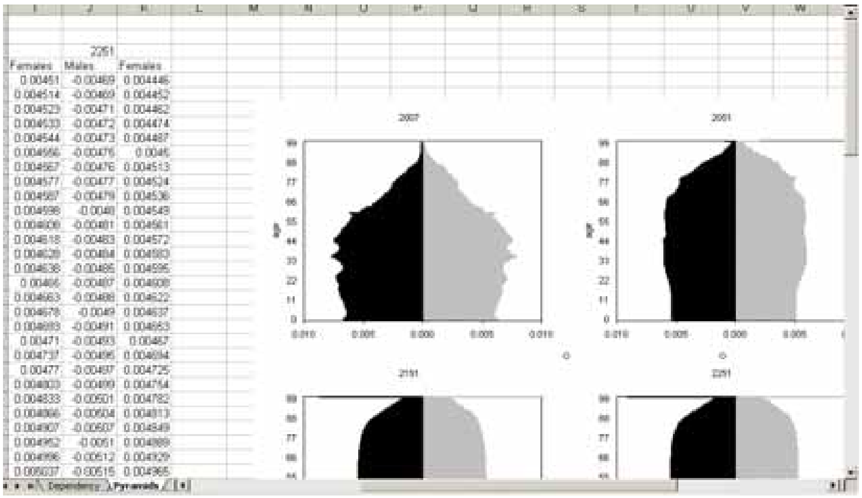
The form has default settings for all of its options, making it easy for users to select just one option, while retaining other settings at their default values. For the purposes of this fertility project, the key option is the long-run value of the total fertility rate. Most of the other options are self-evident, giving options for setting life expectancy, net migration (inwards) and the length of transition periods to the long-run settings.

There are several output options. The default option is the minimalist one. It produces the 'key results' shown in the Start form and no other results. The program runs almost instantaneously when this minimalist option is selected.

However, sometimes users may want more detailed data. The particular options are:

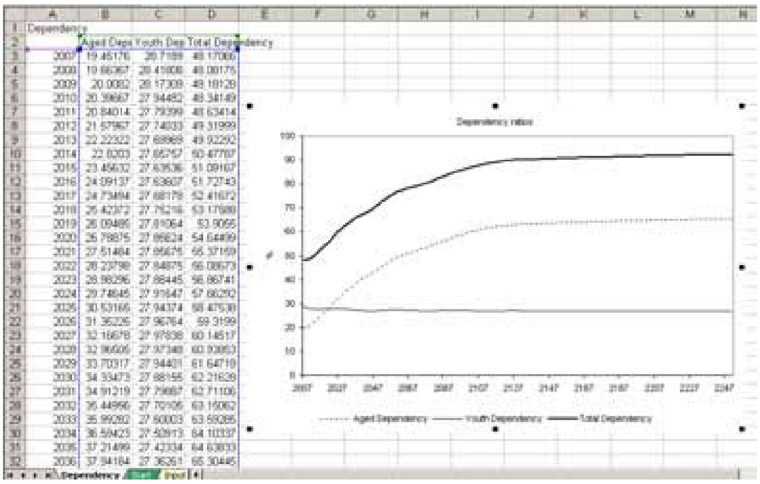
- *‘Do you want to draw pyramid graphs?’*. If yes, this will provide a sheet with graphs of the share of the population by age (so-called ‘pyramid’ graphs). (A sample of part of the output is shown in figure 1.3). It will also provide the data underlying these graphs. The graphs and data are provided for 2007, 2051, 2101, 2151 and 2251.

Figure 1.3 **Pyramid graphs in FERTMOD**



- *‘Do you want a dependency graph?* If yes, this will produce a graph of so-called ‘dependency’ ratios from 2007 to 2251 (A sample is shown in figure 1.4.) Dependency ratios are ratios of various population age groups (0 to 14 and 65+ years) to the population in which workforce participation rates are highest (15 to 64 years). The term ‘dependency’ need not imply that particular people in these young or old populations are financially or otherwise dependent. But it nevertheless will indicate the relative numbers of people who are intensively engaged in labour markets.
- *‘Store detailed population numbers?’* If yes, this will provide year by year data on population numbers by age and gender, and also life expectancies for each year.

Figure 1.4 **Dependency in FERTMOD**



Documenting some key inputs: the ‘Input’ worksheet

The ‘Input’ worksheet is a less important sheet, whose role is only to document the key input data. These data are common to all projections. Users should not change these data unless they wish to update the base numbers. If they wish to do so, the relevant cells should be unlocked by removing worksheet protection — available under the ‘tools’ menu in Excel. We do not recommend this for most users.

The worksheet includes by initial population numbers, death probabilities ($Q_{x,t}$), net migration flows, the current stock of Australians born overseas, and the change in mortality given by $\log(Q_{x,2050}) - \log(Q_{x,2004})$ from the ABS B series projections — all by age and sex. The worksheet also shows the most recent age-specific fertility rates.

1.5 Model construction

The standard features of the cohort-component model are well documented (for example, in PC 2005). However, it is important to indicate how FERTMOD:

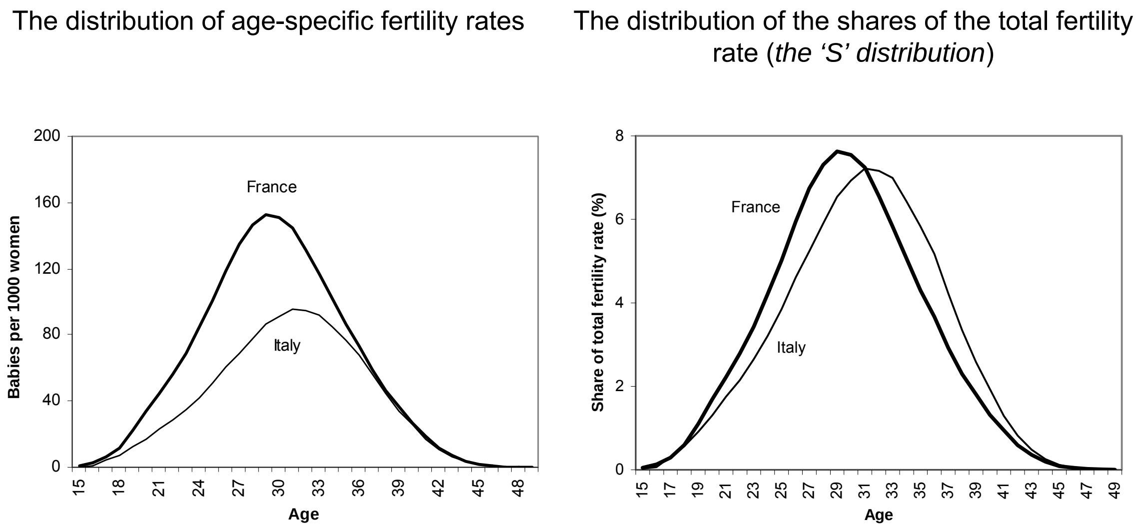
- determines future age-specific fertility rates;
- translates life expectancy settings into Q_x over time; and
- estimates net migration levels to achieve a given population target, where net migration is endogenous rather than exogenous in the model.

Age-specific fertility rates

Empirically, increases or decreases in the total fertility rate (TFR) do not scale up or down the existing age-specific fertility distribution, but change the shape of the distribution. Accordingly, decreasing TFRs have usually resulted in much bigger shifts down in age-specific fertility rates for young women than in older women. In some countries, the age-specific fertility rates of the latter have often risen as women have postponed fertility.

As an illustration, Italy (a low fertility country) and France (a high fertility country) provide revealing contrasts, with quite different distributions of their age-specific fertility rates (ASFRs) (figure 1.5). It is clear from the distribution of age-specific fertility rates that older Italian women have fertility rates similar to French women at those ages. However, younger Italian women have fertility rates much lower than do their French counterparts. Consequently, older women account for a greater share of total fertility in Italy (the right-hand panel of figure 1.5). The age-specific fertility share distribution (the S distribution) for a low TFR country shifts to the right.

Figure 1.5 Italian and French age-specific fertility



[a] Italy is for 2005 and France for 2006. Italy had a TFR of 1.32 and France a TFR of 2.004.

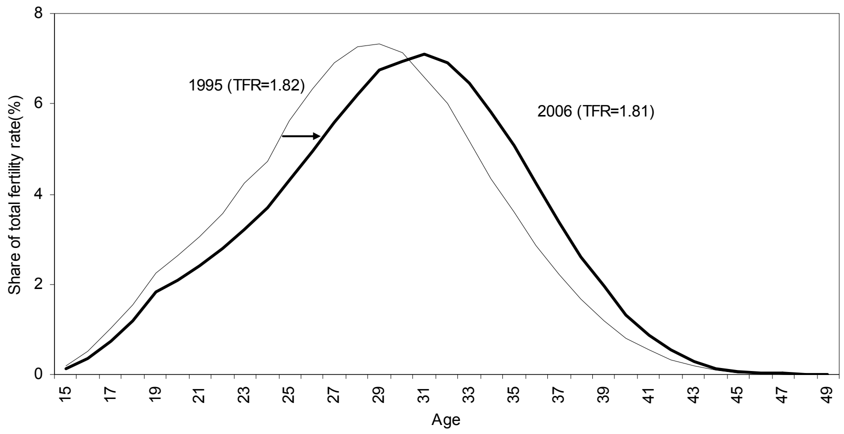
Data source: Eurostat data for Italy and France.

Changing patterns in the timing of children also affect the S distribution, even when the total fertility rate stays the same. For example, Australia's S distribution moved significantly to the right from 1995 to 2006 (as the role of older women in childbearing increased), although the TFR was about the same (figure 1.6).

So in modelling the future S distribution (and its associated age-specific fertility rates), it is important to take account of the likely continuation of these timing effects, as well as the separate impact of changing TFRs.

Figure 1.6 **Australia's age share of fertility is shifting towards older women**

Share of total fertility by age group



Data source: From unpublished data provided by the ABS (based on *Births, Australia,* various issues, Cat. No. 3301.0).

This is achieved in several steps in FERTMOD.

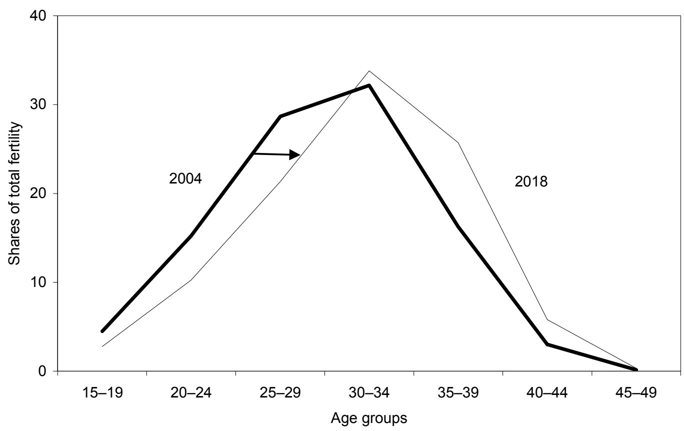
The first step: timing effects

In ABS projections of future age-specific fertility rates, the S distribution shifts rightwards as in figure 1.6 (ABS 2006). This shift can be depicted by the ratio between the two S distributions (R):

$$R_{age} = \frac{ASFR_{age,\,2018} / (1000.\,TFR_{2006})}{ASFR_{age,\,2004} / (1000.\,TFR_{2004})} = \frac{ASFR_{age,\,2018}}{ASFR_{age,\,2004}} \times \frac{TFR_{2004}}{TFR_{2006}}$$

where the ASFR is the age-specific fertility rate and TFR the total fertility rate. The ABS provides the data for calculating R_{age}, but only for five-year age groups (figure 1.7).

Figure 1.7 **The shift in the age share of fertility to 2018**
ABS population projections[a]



[a] 2018 is the long-run in the ABS projections in that after that time, the distribution no longer alters.

Data source: ABS 2006.

In order to interpolate the value of R_{age} for all ages between 15 and 49 years, we ran a regression of R_{age} on age, age^2 and age^3:

$$R'_{age} = 1.0 + \lambda \times \{2.338288 - 0.308566\ age + 0.0103884\ age^2 - 0.00009140508\ age^3\}$$

R'_{age} provides a close approximation to the observed ABS value of R, as shown in figure 1.8. It is used in FERTMOD to determine the long-run impact of timing effects on the S distribution, independent of any effects on that distribution arising from changes in the TFR.

λ is a parameter such that $0 \leq \lambda \leq 0$, which allows users to determine how much weight they wish to give to shifts in the S distribution due to timing changes in the TFR. If $\lambda = 0$, any shift is disregarded. With $\lambda = 1$ (the default), the full impact of changes to the S distribution arising from timing are taken into account. Users can specify amounts between 0 and 1 if they wish to choose an intermediate position, or specify $1 < \lambda < 2$ if they wish to accentuate shifts in the S distribution due to timing.

The second step: effects of changing TFRs

FERTMOD uses the relationship between the S distributions of France and Italy as the basis for estimating the effects of variations in the TFR on the S distribution. The ratio (Y, shown below in figure 1.9) can be approximated as:

$$\hat{Y}_{age} = 1.0 + \{\mu + \gamma\ age + \beta\ age^2 - \alpha\ age^3\} \Rightarrow$$

$\hat{Y}_{age} = 1.0 + \{2.167 - 0.3038\ age + 0.01159\ age^2 - 0.0001277\ age^3\}$

Figure 1.8 **Ratio of age shares, Australia**

2018 age share distribution on 2004 age share distributions

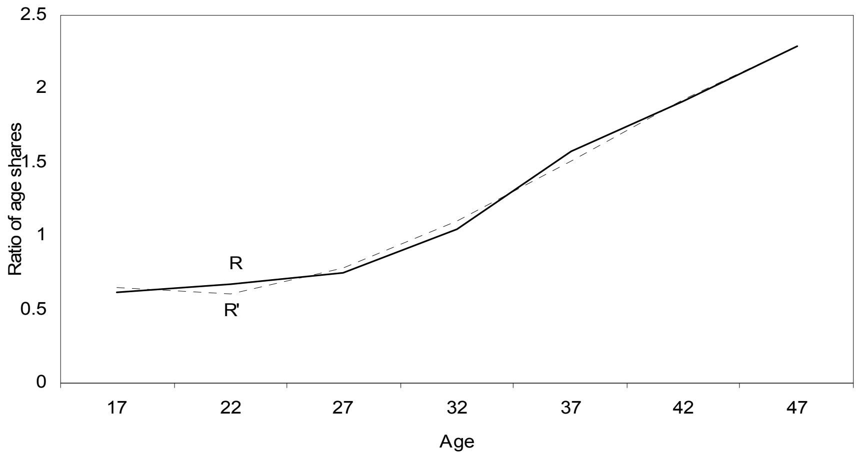
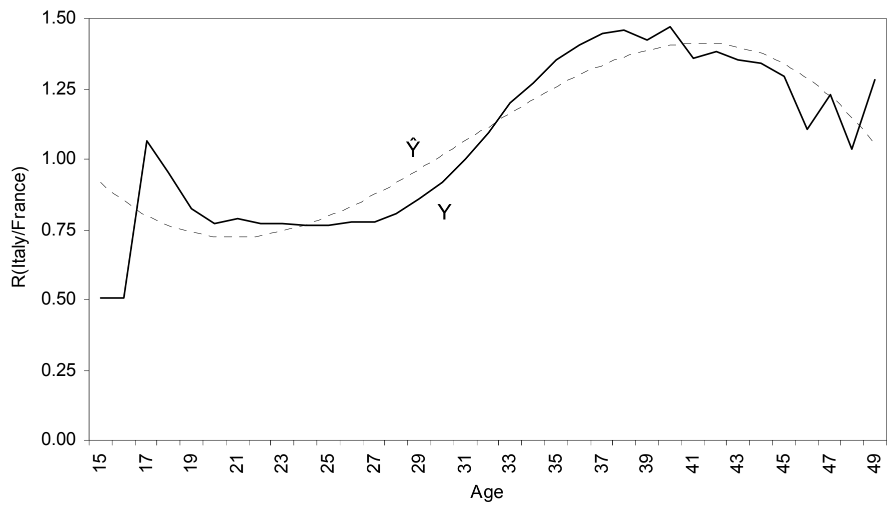


Figure 1.9 **Ratio of Age shares**

Italy compared with France



Data source: As in figure 1.5.

$\hat{Y}_{age}$ is the ratio of the S distributions for two countries whose TFRs are 1.32 and 2.004 — that is, where the low-fertility country's TFR is 66 per cent of the high-fertility country's TFR.

The most recent official estimate of the TFR for Australia is 1.814 for 2006 (TFR_{2006}). In an Australian context, the position of Italy relative to France is akin to comparing an alternative level of TFR (Alt_TFR) of 0.66 of the current TFR, that is:

$$Alt\_TFR = (TFR_{Italy} / TFR_{France}) \times TFR_{2006} = 1.2$$

We assume that were the Australian TFR to fall to 1.2 in the long run, the S distribution would shift by exactly the same as Italian S distribution currently does with respect to the French distribution (i.e. by $\hat{Y}_{age}$). But what might happen for other values of the TFR? To approximate that, we scale α, β, μ and γ by Ω, where:

$$\Omega = \frac{(LRT - TFR_{2006})}{Alt\_TFR - TFR_{2006}}$$

where LRT is the Longrun_TFR (the projected long-run TFR selected by the user) *or* 3.35, whichever is the smaller. It is apparent that if the TFR does not change over time Ω=0. Were LRT to be equal to Alt_TFR (that is, around 1.2) then Ω=1 and the scaled values of α, β, μ and γ would be the same as estimated above.

The reason for specifying a maximum of LRT is that if Longrun_TFR is much greater than 3.35, some age-specific fertility rates become negative at older ages. Setting a limit prevents this. Users can still nominate a value of the long-run TFR above 3.35. If they do, then age-specific fertility rates will still add up to that higher TFR. So setting Longrun_TFR = 4.0 will still result in age-specific fertility rates that add up to 4.0 (when divided by 1000). However, the shape of the distribution of age-specific fertility rates (the *S distribution)* is fixed for Longrun_TFR >= 3.35.

Given Ω, the appropriate ratio to shift the S distribution in line with a new TFR is: $AdjRatio_{age} = 1.0 + \theta \times \Omega \times \{\mu. + \gamma\, age + \beta\, age^2 + \alpha\, age^3\}$. Given the definition of Ω above, other than the effects of timing, the S distribution will stay the same as in 2006 if the TFR does not change from its current level.

θ fulfils a similar function to λ for timing effects. θ is a parameter such that $0 \le \theta \le 2$ that allows users to determine how much weight they wish to give to shifts in the S distribution due to changes in the TFR. If θ = 0, any shift is disregarded, regardless of the choice of the longrun_TFR. This is the modelling option implicit in ABS projections.[2] With θ = 1 (the default), the full impact of changes to the S distribution arising from changes in TFR are taken into account. Users can specify

[2] Since, the age-specific fertility shares of the TFR in the ABS projections are invariant to the long-run choice of the TFR.

amounts between 0 and 1 if they wish to choose an intermediate position, or specify $1 < \theta < 2$ if they wish to accentuate shifts in the S distribution.

Notably, if $\theta=\lambda=0$, then both timing and TFR effects on the S distribution are ignored and the long-run age specific shares of the TFR will remain at their 2006 settings. In FERTMOD, this setting means that any changes in the TFR will simply scale up or down proportionately the current age-specific fertility rates.

Step 3 bringing the steps together

First, we calculate the current age-specific shares of the TFR for Australia:

$$AUSH_{age} = ASFR_{age,\,2006} / (1000.\,TFR_{2006})$$

Then the long-run age specific fertility rates resulting from the combined effects of timing and any changes in the TFR are calculated, and are normalised so that they must sum to one:

$$LR\_ASFR_{age} = \frac{AdjRatio_{age} \times R'_{age} \times AUSH_{age}}{\sum_{age=15}^{49} AdjRatio_{age} \times R'_{age} \times AUSH_{age}} \times Longrun\_TFR \times 1000$$

Finally, the path to the long-run age-specific fertility rate is calculated, where PeriodM is the time it takes to reach the long-run values:

$$\text{If } (year \le PeriodM + 2006) \text{ Then}$$
$$ASFR_{age,\,year} = ASFR_{age,\,year\text{-}1} + \{LR\_ASFR_{age} - ASFR_{age,2006}\}/PeriodM$$
$$\text{Else}$$
$$ASFR_{age,\,year} = LR\_ASFR_{age}$$

Life expectancies

Historically, age-specific probabilities of death have declined steeply — and the default presumption of FERTMOD is that this continues unabated, albeit over a protracted period.

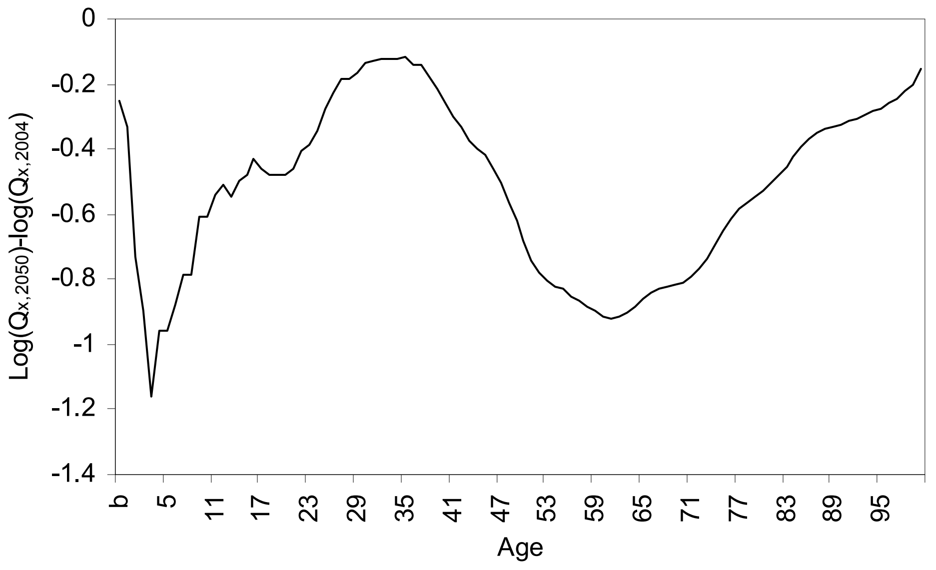
In FERTMOD the user nominates the long-run life expectancies of males (desired_M) and females (desired_F).

FERTMOD then finds a set of long-run mortality probabilities by age (Q_x ,where x is an age from birth to 100+) consistent with these nominated life expectancies. A simple solution would be a fixed scaling up or down of the Q_x values for 2006.

However, historically, probabilities of death have declined more for some ages than others. For example, males aged 65-69 have experienced larger reductions in death probabilities than males aged 35-39. Consequently, the achievement of a given life expectancy figure in any future year will involve larger reductions in mortality for some ages than others. ABS projections of future populations assume the continuation of this pattern. Figure 1.10 shows $DABSQ_x = \log(Q_{x,2050}) - \log(Q_{x,2004})$.

Figure 1.10 **Reductions in the probability of death**

By age, 2004 to 2050, ABS B projections



Data source: Unpublished data from the ABS.

FERTMOD finds a scale factor, Φ (for males and females separately) to apply to $DABSQ_x$, such that the resulting long-run Q_x series (in 2251) are consistent with the nominated life expectancy. This means that $\log(Q_{x,2251}) - \log(Q_{x,2005}) = \Phi \times DABSQ_x$.

The long-run value of Q_x for a given gender is then given by:

$$LRQ_x = e^{(\Phi \times DABSQ_x)} \times Q_{x,2005}$$

$Q_{x,2005}$ is used as the base, since the most recent lifetables relate to 2005. We find Φ for any given life expectancy using a non-linear numerical technique (the secant method). We increase the speed of the routine by seeding the routine with close approximations to the solution. These close approximations are based on an estimated generalised logistic curve of the relationship between life expectancies (desired_M and desired_F) and the scale factor.

As in the case of fertility, FERTMOD allows users to determine the transition period to the realisation of the life expectancies (T).

If year < T+2005 then

$$Q_{x,year} = Q_{x,2005} \times e^{\left\{\frac{(year - 2005) \times \Phi \times DABSQ_x}{T}\right\}} \text{ else } Q_{x,year} = Q_{x,2005} \times e^{\{\Phi \times DABSQ_x\}}$$

Endogenous net migration

The Australian Government has a considerable capacity to choose the level of net migration into Australia. Making choices in this area depend, on the one hand, any sustainability issues associated with larger populations, and on the other, concerns to bolster the size of the workforce or the commercial desire for a bigger population.

As a consequence, were the TFR to fall, the government may well counter its dampening effect on population growth by increasing net migration. Conversely, were the TFR to rise, the government may well reduce its emphasis on net migration to achieve a sustainable population.

FERTMOD allows users to consider the demographic outcomes if governments choose net migration levels to ultimately achieve a certain population target (for given fertility levels).

The model is solved iteratively for differing net migration levels to achieve the population target. As in modelling mortality, a non-linear numerical technique (again the secant method) is used to find the solution.

1.6 Model code

The full model code is provided below.

Module 1 code

```
Public Life_Exp As Double
Public PopM(100, 250) As Double
Public PopF(100, 250) As Double
Public BornOS_male_age_dist(101) As Double
Public BornOS_female_age_dist(101) As Double
Public NOM_male_age_dist(101) As Double
Public NOM_female_age_dist(101) As Double
Public Age_specific_TFR(49, 250) As Double
Public year, age As Integer
Public PeriodM As Integer
Public longrun_TFR As Double
```

```
Public PopM_ABorn(100, 250) As Double, PopF_ABorn(100, 250) As Double ' note need to declare type of each variable, even when on the same line
Public TotPopM(250) As Double, TotPopF(250) As Double
Public NOM_Fix As Double
Public DQx_Male(101) As Double
Public DQx_Female(101) As Double
Public Qx_ABSLR_Male(101) As Double, Qx_ABSLR_Female(101) As Double
Public Qx_Male(101, 250) As Double
Public Qx_Female(101, 250) As Double
Public Transition_yrs_mortality_male As Double, Transition_yrs_mortality_female As Double
Public i As Integer
Public desired_M As Double, desired_F As Double
Public scale_male As Double, scale_female As Double
Public guess_male As Double
Public guess_female As Double
Public x0 As Double, x1 As Double, x2 As Double, Dx As Double
Public k As Integer
Public scale_M As Double, scale_F As Double
Public err1 As Double, err0 As Double
Public endogenous_migration As String, Transition_yrs_NOM As Double
Public AgeShare_NOM_Male(100) As Double, AgeShare_NOM_Female(100) As Double, POPNOM_Male(100, 250) As Double, POPNOM_Female(100, 250) As Double, POPNOM(250) As Double
Public SURVIVE_POPNOM_Male(100, 250), SURVIVE_POPNOM_Female(100, 250) As Double
Public Target_pop As Double
Public Survivors_male_Aborn(100, 250) As Double, Survivors_female_Aborn(100, 250) As Double, Survivors_male(100, 250) As Double, Survivors_female(100, 250) As Double
Public Survivors_male_OSborn(100, 250) As Double, Survivors_female_OSborn(100, 250) As Double
Public Births(49, 250) As Double, TotalBirths(250) As Double
Public Pop(250) As Double, Aust_Born(250) As Double, Prime(250) As Double, Aged(250) As Double, Under10(250) As Double, Youth(250) As Double, Aged_Dependency(250) As Double, Youth_Dependency(250) As Double, Dependency(250) As Double, AgedShare(250) As Double, YouthShare(250) As Double, OS_Share(250) As Double
Public POP2555(250) As Double, POP2555Share(250) As Double
Public PopLR0 As Double, PopLR1 As Double, PopTotal As Double
Public LR_POP As Variant
Public epsilon As Double
Public LambDa, ThetaVal As Double
Public Const Male_share As Double = 0.501477272727273
Public Const mig_share As Double = 23.8398553979533
Public Const male_birthshare As Double = 105.5 / 205.5
Public Const TFR2006 As Double = 1.81442959142374
Public BigL_m(101) As Double, BigL_f(101) As Double
Public T_m As Double, T_f As Double
Public Life_m As Double, Life_f As Double
Public Const Nmax As Integer = 1000
Public male_M As Double, male_B As Double, male_T As Double, male_A As Double, male_C As Double
Public female_M As Double, female_B As Double, female_T As Double, female_A As Double, female_C As Double
Public tot_sh, Lmb, Theta As Double
Public TFR_Ratio As Double, Alpha As Double, Beta As Double, Gamma As Double, mu As Double
Public LR_Age_specific_TFR(49) As Double

Sub Auto_Open()
   MsgBox "Welcome to FERTMOD!"
   Range("H6").Select
End Sub

Sub start()
StartFrm.Show
End Sub
```

Startfrm code

```
Private Sub CmdOK_mode_Click()
Call Cohort_component_model
End Sub
Private Sub CmdCancel_Mode_Click()
Unload Me
End Sub

Public Sub Cohort_component_model()
Application.ScreenUpdating = False

Application.DisplayAlerts = False
If SheetExists("detailed") = True Then
   Sheets("detailed").Delete
End If

If SheetExists("Pyramids") = True Then
   Sheets("Pyramids").Delete
End If

If SheetExists("Dependency") = True Then
   Sheets("Dependency").Delete
End If
Application.DisplayAlerts = True

'Input variables from form
Transition_yrs_NOM = Val(TxtTransNOM) ' years taken for net migration to shift from current level to new level
Target_pop = Val(TxtPOP)
desired_M = Val(TxtMaleLE)
desired_F = Val(TxtFemaleLE)
PeriodM = Val(TxtTransTFR)
longrun_TFR = Val(TxtTFR)
Transition_yrs_mortality_male = Val(TxtTransLE_male)
Transition_yrs_mortality_female = Val(TxtTransLE_female)
LambDa = Val(TxtBoxLambda)
ThetaVal = Val(TxtBoxTheta)
If LambDa < 0 Then
   LambDa = 1#
End If

If ThetaVal < 0 Then
   ThetaVal = 1#
End If

Call input_start 'input some starting values of the population
Call fertility 'Fertility calculations
'----------------------------------------------
' Mortality calculation
'Qx in 2005 (the known Qx values from the ABS life tables) and
'Qx in 2251 were the ABS life expectancy projections for 2050 to be fixed thereafter (these will be scaled up or down later)
'Define life expectancy at 2251

'Richard's curve (or generalised logistic) parameters for aprox solutions to scale variable
male_M = 113.698766239251
male_T = 0.938136023303275
male_B = 0.069632942708231
male_A = -1.9713940319551
male_C = 27.4843883317946
guess_male = male_A + male_C / (1# + male_T * Exp(-male_B * (desired_M - male_M)) ^ (1# / male_T))
```

```
female_M = 144.22013437038
female_T = 0.288362741928448
female_B = 1.88346551267379E-02
female_A = -3.51761174917085
female_C = 56.9098369841528
guess_female = female_A + female_C / (1# + female_T * Exp(-female_B * (desired_F - female_M)) ^ (1# / female_T))
epsilon = 0.000001
'---------------------------------
'Calcs for males (secant method)
   x0 = guess_male * 0.95
   x1 = guess_male * 1.05
   k = 0
   Do While Abs(LifeGoal(x1, DQx_Male, Qx_Male) - desired_M) > epsilon And k < Nmax
      k = k + 1
      If Abs(LifeGoal(x1, DQx_Male, Qx_Male) - LifeGoal(x0, DQx_Male, Qx_Male)) < 0.0000001 Then
         MsgBox ("Values of life expectancy are not changing any more.")
      Exit Sub
      End If
   err1 = LifeGoal(x1, DQx_Male, Qx_Male) - desired_M
   err0 = LifeGoal(x0, DQx_Male, Qx_Male) - desired_M
   x2 = x1 + err1 * (x1 - x0) / (err0 - err1)
   x0 = x1
   x1 = x2 ' use x2 as the solution
   Loop
   scale_M = x2

'Calcs for females
   x0 = guess_female * 0.9
   x1 = guess_female * 1.1
   k = 0
   Do While Abs(LifeGoal(x1, DQx_Female, Qx_Female) - desired_F) > epsilon And k < Nmax
      k = k + 1
      If Abs(LifeGoal(x1, DQx_Female, Qx_Female) - LifeGoal(x0, DQx_Female, Qx_Female)) < 0.0000001 Then
         MsgBox ("Values of life expectancy are not changing any more.")
      Exit Sub
      End If
   err1 = LifeGoal(x1, DQx_Female, Qx_Female) - desired_F
   err0 = LifeGoal(x0, DQx_Female, Qx_Female) - desired_F
   x2 = x1 + err1 * (x1 - x0) / (err0 - err1)
   x0 = x1
   x1 = x2 ' use x2 as the solution
   Loop
   scale_F = x2
'-----------------------------------
'Now work out scaled values of Qx for all intervening years
For i = 0 To 101
   For year = 2006 To 2251
      If (year - 2005) < Transition_yrs_mortality_male Then
         Qx_Male(i, year - 2005) = Qx_Male(i, 0) * Exp((year - 2005) * scale_M * DQx_Male(i) / Transition_yrs_mortality_male) ' Qx_Male(i, 2)
is 2007 values of Qx
      Else
         Qx_Male(i, year - 2005) = Qx_Male(i, 0) * Exp(scale_M * DQx_Male(i))
      End If
   Next year
Next i

For i = 0 To 101
   For year = 2006 To 2251
      If (year - 2005) < Transition_yrs_mortality_female Then
         Qx_Female(i, year - 2005) = Qx_Female(i, 0) * Exp((year - 2005) * scale_F * DQx_Female(i) / Transition_yrs_mortality_female)
```

```
        Else
            Qx_Female(i, year - 2005) = Qx_Female(i, 0) * Exp(scale_F * DQx_Female(i))
        End If
    Next year
Next i
'----------------------------------------------------
'Net migration

If OptButNo_Mode = False Then
    Call secant
Else
    NOM_Fix = Val(TxtNOM)
End If
Call LongRunPop(NOM_Fix) 'does cohort model

'display parameter settings for model
If OptButYes_Mode = True Then
    Worksheets("Start").Cells(4, 5).Value = Target_pop
    Worksheets("Start").Cells(11, 5).Value = x2
Else
    Worksheets("Start").Cells(4, 5).Value = "Population endogenous"
    Worksheets("Start").Cells(11, 5).Value = Val(TxtNOM)
End If

Worksheets("Start").Cells(5, 5).Value = longrun_TFR
Worksheets("Start").Cells(6, 5).Value = PeriodM
Worksheets("Start").Cells(7, 5).Value = desired_M
Worksheets("Start").Cells(8, 5).Value = desired_F
Worksheets("Start").Cells(9, 5).Value = Transition_yrs_mortality_male
Worksheets("Start").Cells(10, 5).Value = Transition_yrs_mortality_female
Worksheets("Start").Cells(12, 5).Value = Transition_yrs_NOM
Worksheets("Start").Cells(13, 5).Value = LambDa
Worksheets("Start").Cells(14, 5).Value = ThetaVal

'get useful output results
For year = 2007 To 2251
    For age = 0 To 100
        Pop(year - 2005) = PopF(age, year - 2005) + PopM(age, year - 2005) + Pop(year - 2005)
        Aust_Born(year - 2005) = Aust_Born(year - 2005) + PopF_ABorn(age, year - 2005) + PopM_ABorn(age, year - 2005)

        If age >= 15 And age <= 64 Then
            Prime(year - 2005) = PopF(age, year - 2005) + PopM(age, year - 2005) + Prime(year - 2005)
            If age >= 25 And age <= 55 Then
                POP2555(year - 2005) = PopF(age, year - 2005) + PopM(age, year - 2005) + POP2555(year - 2005)
            End If
        ElseIf age >= 65 Then Aged(year - 2005) = Aged(year - 2005) + PopF(age, year - 2005) + PopM(age, year - 2005)
        ElseIf age < 15 Then
            Youth(year - 2005) = Youth(year - 2005) + PopF(age, year - 2005) + PopM(age, year - 2005)
            If age < 10 Then
            Under10(year - 2005) = Under10(year - 2005) + PopF(age, year - 2005) + PopM(age, year - 2005)
            End If
        End If
    Next age
    Aged_Dependency(year - 2005) = Aged(year - 2005) / Prime(year - 2005) * 100#
    Youth_Dependency(year - 2005) = Youth(year - 2005) / Prime(year - 2005) * 100#
    Dependency(year - 2005) = Youth_Dependency(year - 2005) + Aged_Dependency(year - 2005)
    AgedShare(year - 2005) = Aged(year - 2005) / Pop(year - 2005) * 100#
    YouthShare(year - 2005) = Youth(year - 2005) / Pop(year - 2005) * 100#
    OS_Share(year - 2005) = (1# - Aust_Born(year - 2005) / Pop(year - 2005)) * 100#
    POP2555Share(year - 2005) = POP2555(year - 2005) / Pop(year - 2005) * 100#
```

```
Next year

Worksheets("Start").Cells(21, 6).Value = Pop(2007 - 2005) / 1000000#
Worksheets("Start").Cells(22, 6).Value = AgedShare(2007 - 2005)
Worksheets("Start").Cells(23, 6).Value = YouthShare(2007 - 2005)
Worksheets("Start").Cells(24, 6).Value = Dependency(2007 - 2005)
Worksheets("Start").Cells(25, 6).Value = Aged_Dependency(2007 - 2005)
Worksheets("Start").Cells(26, 6).Value = Youth_Dependency(2007 - 2005)
Worksheets("Start").Cells(27, 6).Value = POP2555Share(2007 - 2005)
Worksheets("Start").Cells(28, 6).Value = "na"
Worksheets("Start").Cells(29, 6).Value = OS_Share(2007 - 2005)
Worksheets("Start").Cells(30, 6).Value = Prime(2007 - 2005) / 1000000#
Worksheets("Start").Cells(31, 6).Value = Aged(2007 - 2005) / 1000000#
Worksheets("Start").Cells(32, 6).Value = Youth(2007 - 2005) / 1000000#
Worksheets("Start").Cells(33, 6).Value = Under10(2007 - 2005) / Pop(2007 - 2005) * 100#

Worksheets("Start").Cells(21, 7).Value = Pop(2051 - 2005) / 1000000#
Worksheets("Start").Cells(22, 7).Value = AgedShare(2051 - 2005)
Worksheets("Start").Cells(23, 7).Value = YouthShare(2051 - 2005)
Worksheets("Start").Cells(24, 7).Value = Dependency(2051 - 2005)
Worksheets("Start").Cells(25, 7).Value = Aged_Dependency(2051 - 2005)
Worksheets("Start").Cells(26, 7).Value = Youth_Dependency(2051 - 2005)
Worksheets("Start").Cells(27, 7).Value = POP2555Share(2051 - 2005)
Worksheets("Start").Cells(28, 7).Value = (Pop(2051 - 2005) / Pop(2051 - 2005 - 1) - 1#) * 100#
Worksheets("Start").Cells(29, 7).Value = OS_Share(2051 - 2005)
Worksheets("Start").Cells(30, 7).Value = Prime(2051 - 2005) / 1000000#
Worksheets("Start").Cells(31, 7).Value = Aged(2051 - 2005) / 1000000#
Worksheets("Start").Cells(32, 7).Value = Youth(2051 - 2005) / 1000000#
Worksheets("Start").Cells(33, 7).Value = Under10(2051 - 2005) / Pop(2051 - 2005) * 100#

Worksheets("Start").Cells(21, 8).Value = Pop(2151 - 2005) / 1000000#
Worksheets("Start").Cells(22, 8).Value = AgedShare(2151 - 2005)
Worksheets("Start").Cells(23, 8).Value = YouthShare(2151 - 2005)
Worksheets("Start").Cells(24, 8).Value = Dependency(2151 - 2005)
Worksheets("Start").Cells(25, 8).Value = Aged_Dependency(2151 - 2005)
Worksheets("Start").Cells(26, 8).Value = Youth_Dependency(2151 - 2005)
Worksheets("Start").Cells(27, 8).Value = POP2555Share(2151 - 2005)
Worksheets("Start").Cells(28, 8).Value = (Pop(2151 - 2005) / Pop(2151 - 2005 - 1) - 1#) * 100#
Worksheets("Start").Cells(29, 8).Value = OS_Share(2151 - 2005)
Worksheets("Start").Cells(30, 8).Value = Prime(2151 - 2005) / 1000000#
Worksheets("Start").Cells(31, 8).Value = Aged(2151 - 2005) / 1000000#
Worksheets("Start").Cells(32, 8).Value = Youth(2151 - 2005) / 1000000#
Worksheets("Start").Cells(33, 8).Value = Under10(2151 - 2005) / Pop(2151 - 2005) * 100#

Worksheets("Start").Cells(21, 9).Value = Pop(2251 - 2005) / 1000000#
Worksheets("Start").Cells(22, 9).Value = AgedShare(2251 - 2005)
Worksheets("Start").Cells(23, 9).Value = YouthShare(2251 - 2005)
Worksheets("Start").Cells(24, 9).Value = Dependency(2251 - 2005)
Worksheets("Start").Cells(25, 9).Value = Aged_Dependency(2251 - 2005)
Worksheets("Start").Cells(26, 9).Value = Youth_Dependency(2251 - 2005)
Worksheets("Start").Cells(27, 9).Value = POP2555Share(2251 - 2005)
Worksheets("Start").Cells(28, 9).Value = (Pop(2251 - 2005) / Pop(2251 - 2005 - 1) - 1#) * 100#
Worksheets("Start").Cells(29, 9).Value = OS_Share(2251 - 2005)
Worksheets("Start").Cells(30, 9).Value = Prime(2251 - 2005) / 1000000#
Worksheets("Start").Cells(31, 9).Value = Aged(2251 - 2005) / 1000000#
Worksheets("Start").Cells(32, 9).Value = Youth(2251 - 2005) / 1000000#
Worksheets("Start").Cells(33, 9).Value = Under10(2251 - 2005) / Pop(2251 - 2005) * 100#

If OptButNo_StoreMode = False Then
  Sheets.Add.Name = "Detailed"
```

```
    Call MoveSheets
    Worksheets("detailed").Cells(309, 1).Value = "Total population"
    For year = 2007 To 2251
        Worksheets("detailed").Cells(1, year - 2005).Value = year
        Worksheets("detailed").Cells(309, year - 2005).Value = Pop(year - 2005)

' Calculates Life expectancy from Qx values
' It assumes first value of Q is Qb, then Q0 to Qmax
        BigL_m(0) = (1# - Qx_Male(0, year - 2005)) * 100000#  'L0 = (1-Qb)*100,000 noting that Qxdata(0) is Qb
        BigL_f(0) = (1# - Qx_Female(0, year - 2005)) * 100000#  'L0 = (1-Qb)*100,000 noting that Qxdata(0) is Qb
        For i = 1 To 100
            BigL_m(i) = (1# - Qx_Male(i, year - 2005)) * BigL_m(i - 1) ' L1=(1-Q0)L0 to L100=(1-Q99)*L99
            BigL_f(i) = (1# - Qx_Female(i, year - 2005)) * BigL_f(i - 1) ' L1=(1-Q0)L0 to L100=(1-Q99)*L99
        Next i
        BigL_m(101) = BigL_m(100) * (1# - Qx_Male(101, year - 2005)) / Qx_Male(101, year - 2005)
        BigL_f(101) = BigL_f(100) * (1# - Qx_Female(101, year - 2005)) / Qx_Female(101, year - 2005)

        T_m = 0#
        T_f = 0#
        For i = 0 To 101
            T_m = BigL_m(i) + T_m
            T_f = BigL_f(i) + T_f
        Next i
        Life_m = T_m / 100000#
        Life_f = T_f / 100000#
        Worksheets("detailed").Cells(350, year - 2005).Value = Life_m
        Worksheets("detailed").Cells(351, year - 2005).Value = Life_f

        For age = 0 To 100
            Worksheets("detailed").Cells(age + 3, year - 2005).Value = PopM(age, year - 2005)
            Worksheets("detailed").Cells(age + 105, year - 2005).Value = PopF(age, year - 2005)
            Worksheets("detailed").Cells(age + 208, year - 2005).Value = PopF(age, year - 2005) + PopM(age, year - 2005)
        Next age

        For age = 15 To 49
            Worksheets("detailed").Cells(age + 297, year - 2005).Value = Age_specific_TFR(age, year - 2005)
        Next age

    Next year

    For age = 0 To 100
        Worksheets("detailed").Cells(age + 3, 1).Value = age
        Worksheets("detailed").Cells(age + 105, 1).Value = age
        Worksheets("detailed").Cells(age + 208, 1).Value = age
    Next age

    For age = 15 To 49
        Worksheets("detailed").Cells(age + 297, 1).Value = age
    Next age

    Worksheets("detailed").Cells(2, 1).Value = "Male population (End June)"
    Worksheets("detailed").Cells(104, 1).Value = "Female population (End June)"
    Worksheets("detailed").Cells(207, 1).Value = "Total population(End June)"
    Worksheets("detailed").Cells(311, 1).Value = "Age-specific fertility rates (babies per 1000 women)(End June)"
    Worksheets("detailed").Cells(350, 1).Value = "Male life expectancy (years)"
    Worksheets("detailed").Cells(351, 1).Value = "Female life expectancy (years)"

End If

If OptButPyramid_No = False Then  'Chart pyramids
```

```
   Sheets.Add.Name = "Pyramids"
   Call MoveSheets
   Worksheets("Pyramids").Cells(1, 1).Value = "Population pyramids"
   Worksheets("Pyramids").Cells(3, 2).Value = "2007"
   Worksheets("Pyramids").Cells(3, 4).Value = "2051"
   Worksheets("Pyramids").Cells(3, 6).Value = "2101"
   Worksheets("Pyramids").Cells(3, 8).Value = "2151"
   Worksheets("Pyramids").Cells(3, 10).Value = "2251"
   Worksheets("Pyramids").Cells(4, 2).Value = "Males"
   Worksheets("Pyramids").Cells(4, 3).Value = "Females"
   Worksheets("Pyramids").Cells(4, 4).Value = "Males"
   Worksheets("Pyramids").Cells(4, 5).Value = "Females"
   Worksheets("Pyramids").Cells(4, 6).Value = "Males"
   Worksheets("Pyramids").Cells(4, 7).Value = "Females"
   Worksheets("Pyramids").Cells(4, 8).Value = "Males"
   Worksheets("Pyramids").Cells(4, 9).Value = "Females"
   Worksheets("Pyramids").Cells(4, 10).Value = "Males"
   Worksheets("Pyramids").Cells(4, 11).Value = "Females"

For age = 0 To 100
   Worksheets("Pyramids").Cells(age + 5, 1).Value = age
   Worksheets("Pyramids").Cells(age + 5, 2).Value = -PopM(age, 2) / Pop(2)
   Worksheets("Pyramids").Cells(age + 5, 3).Value = PopF(age, 2) / Pop(2)
   Worksheets("Pyramids").Cells(age + 5, 4).Value = -PopM(age, 46) / Pop(46)
   Worksheets("Pyramids").Cells(age + 5, 5).Value = PopF(age, 46) / Pop(46)
   Worksheets("Pyramids").Cells(age + 5, 6).Value = -PopM(age, 96) / Pop(96)
   Worksheets("Pyramids").Cells(age + 5, 7).Value = PopF(age, 96) / Pop(96)
   Worksheets("Pyramids").Cells(age + 5, 8).Value = -PopM(age, 146) / Pop(146)
   Worksheets("Pyramids").Cells(age + 5, 9).Value = PopF(age, 146) / Pop(146)
   Worksheets("Pyramids").Cells(age + 5, 10).Value = -PopM(age, 246) / Pop(246)
   Worksheets("Pyramids").Cells(age + 5, 11).Value = PopF(age, 246) / Pop(246)
Next age
Call Pyramid_Charting
End If

If OptButDepend_No = False Then
   Sheets.Add.Name = "Dependency"
   Call MoveSheets
   Worksheets("Dependency").Cells(1, 1).Value = "Dependency"
   Worksheets("Dependency").Cells(2, 2).Value = "Aged Dependency"
   Worksheets("Dependency").Cells(2, 3).Value = "Youth Dependency"
   Worksheets("Dependency").Cells(2, 4).Value = "Total Dependency"
   For year = 2007 To 2251
      Worksheets("Dependency").Cells(year - 2007 + 3, 1).Value = year
      Worksheets("Dependency").Cells(year - 2007 + 3, 2).Value = Aged_Dependency(year - 2005)
      Worksheets("Dependency").Cells(year - 2007 + 3, 3).Value = Youth_Dependency(year - 2005)
      Worksheets("Dependency").Cells(year - 2007 + 3, 4).Value = Dependency(year - 2005)
   Next year
   Call Dependency_chart
End If

Application.ScreenUpdating = True
CmdCancel_Mode = True
Worksheets("start").Select
End
End Sub
Function SheetExists(SheetName As String) As Boolean
' returns TRUE if the sheet exists in the active workbook
   SheetExists = False
   On Error GoTo NoSuchSheet
```

```
    If Len(Sheets(SheetName).Name) > 0 Then
        SheetExists = True
        Exit Function
    End If
NoSuchSheet:
End Function

Sub secant()
Dim z0 As Double, z1 As Double
    x0 = 100000
    x1 = 110000
    k = 0
    epsilon = 50 / 30000000 * Target_pop ' small % error around target pop allowed
    Do While Abs(LongRunPop(x1) - Target_pop) > epsilon And k < Nmax
        k = k + 1

        If Abs(LongRunPop(x1) - LongRunPop(x0)) < 0.000000001 Then
            MsgBox ("Population is no longer changing")
        Exit Sub
        End If
    z1 = LongRunPop(x1)
    z0 = LongRunPop(x0)
    err1 = z1 - Target_pop
    err0 = z0 - Target_pop
    x2 = x1 + err1 * (x1 - x0) / (err0 - err1)
    x0 = x1
    x1 = x2 ' use x2 as the solution

    Loop
    NOM_Fix = x2

End Sub

Public Function LongRunPop(Nom As Variant) As Double 'this is intended to model the long run population
'Overall cohort modelling

'cohort behaviour of migrants
For year = 2008 To 2251
    If year - 2007 < Transition_yrs_NOM Then
    POPNOM(year - 2005) = (Nom - POPNOM(2)) / Transition_yrs_NOM + POPNOM(year - 2005 - 1) ' popNOM(3) is 2008
    Else
    POPNOM(year - 2005) = Nom
    End If
Next year

For year = 2008 To 2251
    POPNOM_Male(0, year - 2005) = AgeShare_NOM_Male(0) / 100# * Male_share * POPNOM(year - 2005) ' year 3 is 2008
    POPNOM_Female(0, year - 2005) = AgeShare_NOM_Female(0) / 100# * (1# - Male_share) * POPNOM(year - 2005)
    SURVIVE_POPNOM_Male(0, year - 2005) = 0.5 * POPNOM_Male(0, year - 2005) * (1# - 0.5 * Qx_Male(1, year - 2006)) ' year 3 is 2008
    SURVIVE_POPNOM_Female(0, year - 2005) = 0.5 * POPNOM_Female(0, year - 2005) * (1# - 0.5 * Qx_Female(1, year - 2006))

    For age = 1 To 99
        POPNOM_Male(age, year - 2005) = AgeShare_NOM_Male(age) / 100# * Male_share * POPNOM(year - 2005) ' year 3 is 2008
        POPNOM_Female(age, year - 2005) = AgeShare_NOM_Female(age) / 100# * (1# - Male_share) * POPNOM(year - 2005)
        SURVIVE_POPNOM_Male(age, year - 2005) = 0.5 * (POPNOM_Male(age - 1, year - 2005) * (1# - 0.5 * Qx_Male(age, year - 2006)) +
POPNOM_Male(age, year - 2005) * (1 - 0.5 * Qx_Male(age + 1, year - 2006))) ' year 3 is 2008
        SURVIVE_POPNOM_Female(age, year - 2005) = 0.5 * (POPNOM_Female(age - 1, year - 2005) * (1# - 0.5 * Qx_Female(age, year -
2006)) + POPNOM_Female(age, year - 2005) * (1 - 0.5 * Qx_Female(age + 1, year - 2006))) ' year 3 is 2008
    Next age
        POPNOM_Male(100, year - 2005) = AgeShare_NOM_Male(100) / 100# * Male_share * POPNOM(year - 2005) ' year 3 is 2008
        POPNOM_Female(100, year - 2005) = AgeShare_NOM_Female(100) / 100# * (1# - Male_share) * POPNOM(year - 2005)
```

```
        SURVIVE_POPNOM_Male(100, year - 2005) = 0.5 * POPNOM_Male(99, year - 2005) * (1# - 0.5 * Qx_Male(100, year - 2006)) + POPNOM_Male(100, year - 2005) * (1 - 0.5 * Qx_Male(101, year - 2006)) ' year 3 is 2008
        SURVIVE_POPNOM_Female(100, year - 2005) = 0.5 * POPNOM_Female(99, year - 2005) * (1# - 0.5 * Qx_Female(100, year - 2006)) + POPNOM_Female(100, year - 2005) * (1 - 0.5 * Qx_Female(101, year - 2006)) ' year 3 is 2008
Next year
'end of cohort behaviour of migrants
'-----------------------------------------------------------------------------------

'cohort component calcs of whole population

For year = 2008 To 2251
        TotalBirths(year - 2005) = 0#
        For age = 1 To 99
        Survivors_male(age, year - 2005) = (1# - Qx_Male(age, year - 2006)) * PopM(age - 1, year - 2006)
        Survivors_female(age, year - 2005) = (1# - Qx_Female(age, year - 2006)) * PopF(age - 1, year - 2006)
        PopM(age, year - 2005) = Round(SURVIVE_POPNOM_Male(age, year - 2005) + Survivors_male(age, year - 2005), 0) ' year 3 is 2008
        PopF(age, year - 2005) = Round(SURVIVE_POPNOM_Female(age, year - 2005) + Survivors_female(age, year - 2005), 0) ' year 3 is 2008

        Survivors_male_Aborn(age, year - 2005) = (1# - Qx_Male(age, year - 2006)) * PopM_ABorn(age - 1, year - 2006) ' noting Qx(1,2) is Qx for age=0 in 2007
        Survivors_female_Aborn(age, year - 2005) = (1# - Qx_Female(age, year - 2006)) * PopF_ABorn(age - 1, year - 2006)
        PopM_ABorn(age, year - 2005) = Round(Survivors_male_Aborn(age, year - 2005), 0)
        PopF_ABorn(age, year - 2005) = Round(Survivors_female_Aborn(age, year - 2005), 0)

        If age >= 15 And age <= 49 Then
          Births(age, year - 2005) = 0.5 * (Age_specific_TFR(age, year - 2006) * PopF(age, year - 2006) + Age_specific_TFR(age, year - 2005) * PopF(age, year - 2005)) / 1000# 'births(15,3) are births for 15yr olds in 2008
          TotalBirths(year - 2005) = TotalBirths(year - 2005) + Births(age, year - 2005)

        End If
    Next age

        Survivors_male(100, year - 2005) = (1# - Qx_Male(100, year - 2006)) * PopM(99, year - 2006) + (1# - Qx_Male(101, year - 2006)) * PopM(100, year - 2006)
        Survivors_female(100, year - 2005) = (1# - Qx_Female(100, year - 2006)) * PopF(99, year - 2006) + (1# - Qx_Female(101, year - 2006)) * PopF(100, year - 2006)
        PopM(100, year - 2005) = Round(SURVIVE_POPNOM_Male(100, year - 2005) + Survivors_male(100, year - 2005), 0) ' year 3 is 2008
        PopF(100, year - 2005) = Round(SURVIVE_POPNOM_Female(100, year - 2005) + Survivors_female(100, year - 2005), 0)  ' year 3 is 2008
        PopM(0, year - 2005) = Round(TotalBirths(year - 2005) * male_birthshare * (1# - Qx_Male(0, year - 2006)) + SURVIVE_POPNOM_Male(0, year - 2005), 0)
        PopF(0, year - 2005) = Round(TotalBirths(year - 2005) * (1# - male_birthshare) * (1# - Qx_Female(0, year - 2006)) + SURVIVE_POPNOM_Female(0, year - 2005), 0)

        Survivors_male_Aborn(100, year - 2005) = (1# - Qx_Male(100, year - 2006)) * PopM_ABorn(99, year - 2006) + (1# - Qx_Male(101, year - 2006)) * PopM_ABorn(100, year - 2006) ' noting Qx(100,2) is Qx for age=99 in 2007
        Survivors_female_Aborn(100, year - 2005) = (1# - Qx_Female(100, year - 2006)) * PopF_ABorn(99, year - 2006) + (1# - Qx_Female(101, year - 2006)) * PopF_ABorn(100, year - 2006) ' noting Qx(100,2) is Qx for age=99 in 2007
        PopM_ABorn(100, year - 2005) = Round(Survivors_male_Aborn(100, year - 2005), 0)
        PopF_ABorn(100, year - 2005) = Round(Survivors_female_Aborn(100, year - 2005), 0)
        PopM_ABorn(0, year - 2005) = Round(TotalBirths(year - 2005) * male_birthshare * (1# - Qx_Male(0, year - 2006)), 0)
        PopF_ABorn(0, year - 2005) = Round(TotalBirths(year - 2005) * (1# - male_birthshare) * (1# - Qx_Female(0, year - 2006)), 0)

Next year

' long run population
Dim Tot As Double
Tot = 0#
For age = 0 To 100
Tot = Tot + PopM(age, 2251 - 2005) + PopF(age, 2251 - 2005)
Next age
LongRunPop = Tot
```

```
'End of cohort model

End Function

Sub input_start()

With Worksheets("inputs")
   For i = 0 To 101
      Qx_Male(i, 0) = .Cells(4 + i, 3).Value ' male Qx IN 2005 running from birth, age 0, age 1 to age 100+ (ie year 0 is 2005)
      Qx_Female(i, 0) = .Cells(108 + i, 3).Value
      DQx_Male(i) = .Cells(4 + i, 6).Value 'This is the value of Dlog(Qx) males to 2050 from the ABS
      DQx_Female(i) = .Cells(108 + i, 6).Value
   Next i
End With

For age = 0 To 100
   PopM(age, 2) = Worksheets("Inputs").Cells(5 + age, 2).Value   ' 2007 end June pop of males
   PopF(age, 2) = Worksheets("Inputs").Cells(109 + age, 2).Value  ' 2007 end June pop of females
   TotPopM(2) = TotPopM(2) + PopM(age, 2) 'pop of males in 2007
   TotPopF(2) = TotPopF(2) + PopF(age, 2) 'pop of males in 2007
Next age

For age = 0 To 100
   BornOS_male_age_dist(age) = Worksheets("Inputs").Cells(5 + age, 5).Value  ' share of males born overseas by age
   BornOS_female_age_dist(age) = Worksheets("Inputs").Cells(109 + age, 5).Value  ' share of females born overseas by age
   NOM_male_age_dist(age) = Worksheets("Inputs").Cells(5 + age, 4).Value  ' share of new male migrants by age
   NOM_female_age_dist(age) = Worksheets("Inputs").Cells(109 + age, 4).Value  ' share of new female migrants by age
   PopM_ABorn(age, 2) = PopM(age, 2) - BornOS_male_age_dist(age) / 100# * TotPopM(2) * mig_share / 100#
   PopF_ABorn(age, 2) = PopF(age, 2) - BornOS_female_age_dist(age) / 100# * TotPopF(2) * mig_share / 100# ' POPF_OSBorn(age,2) is
data for end 2007
Next age

For age = 0 To 100
   AgeShare_NOM_Male(age) = Worksheets("Inputs").Cells(age + 5, 4).Value
   AgeShare_NOM_Female(age) = Worksheets("Inputs").Cells(age + 109, 4).Value
Next age
POPNOM(2) = 177600  ' NOM in 2007 (ie year=2 is 2007, year =0 is ignored as not needed)

End Sub

Public Function LifeGoal(scale_guess As Variant, arrayDQx As Variant, arrayQx As Variant) As Double ' this undertakes the calculations for
the 2251 life expectancy for 2251
Dim L(101) As Double
Dim T As Double
Dim i As Integer
Dim Qx_ABSLR(101) As Double

   For i = 0 To 101
      Qx_ABSLR(i) = Exp(scale_guess * arrayDQx(i)) * arrayQx(i, 0) ' Qx IN 2251 with scalar adjustment of the (Dlog of ABS life expectancy
from 2005 to 2050) - running from birth, age 0, age 1 to age 100+
   Next i

   L(0) = (1# - Qx_ABSLR(0)) * 100000# ' L0 = (1-Qb)*100,000 noting that Qx_ABSLR(0) is Qb
   For i = 1 To 100
      L(i) = (1# - Qx_ABSLR(i)) * L(i - 1) ' L1=(1-Q0)L0  to L100=(1-Q99)*L99
   Next i

  L(101) = L(100) * (1# - Qx_ABSLR(101)) / Qx_ABSLR(101)
  T = 0#
  For i = 0 To 101
     T = L(i) + T
```

```
  Next i
  LifeGoal = T / 100000#

End Function

Sub Pyramid_Charting()
'
'
Application.ScreenUpdating = False
Sheets("Pyramids").Select
Dim Rangename As String
Dim Titlename As String
For i = 1 To 5

If i = 1 Then
   Rangename = "a4:a105,b4:c105"
   Titlename = "2007"
ElseIf i = 2 Then
   Rangename = "a4:a105,d4:e105"
   Titlename = "2051"

ElseIf i = 3 Then
   Rangename = "a4:a105,f4:g105"
   Titlename = "2101"

ElseIf i = 4 Then
   Rangename = "a4:a105,h4:i105"
   Titlename = "2151"

ElseIf i = 5 Then
   Rangename = "a4:a105,j4:k105"
   Titlename = "2251"
End If

Range(Rangename).Select
   Charts.Add
   ActiveChart.ChartType = xlBarClustered
   ActiveChart.SetSourceData Source:=Sheets("Pyramids").Range(Rangename), _
      PlotBy:=xlColumns
   ActiveChart.Location Where:=xlLocationAsObject, Name:="Pyramids"
   With ActiveChart
      .HasTitle = True
      .ChartTitle.Characters.Text = Titlename
      .Axes(xlCategory, xlPrimary).HasTitle = True
      .Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = "age"
      .Axes(xlValue, xlPrimary).HasTitle = False
   End With
   With ActiveChart.Axes(xlCategory)
      .HasMajorGridlines = False
      .HasMinorGridlines = False
   End With
   With ActiveChart.Axes(xlValue)
      .HasMajorGridlines = False
      .HasMinorGridlines = False
   End With
   ActiveChart.HasLegend = False

   ActiveChart.Axes(xlValue).Select
   With ActiveChart.Axes(xlValue)
      .MinimumScale = -0.01
```

```
    .MaximumScale = 0.01
    .MinorUnitIsAuto = True
    .MajorUnitIsAuto = True
    .Crosses = xlAutomatic
    .ReversePlotOrder = False
    .ScaleType = xlLinear
    .DisplayUnit = xlNone
End With

ActiveChart.PlotArea.Select
With Selection.Border
    .ColorIndex = 16
    .Weight = xlThin
    .LineStyle = xlContinuous
End With
Selection.Interior.ColorIndex = xlNone
ActiveChart.ChartArea.Select
With Selection.Border
    .Weight = 2
    .LineStyle = 0
End With

ActiveChart.PlotArea.Select
ActiveChart.Axes(xlCategory).Select
With Selection.Border
    .Weight = xlHairline
    .LineStyle = xlAutomatic
End With
With Selection
    .MajorTickMark = xlOutside
    .MinorTickMark = xlNone
    .TickLabelPosition = xlLow
End With
ActiveChart.SeriesCollection(1).Select
With Selection.Border
    .Weight = xlThin
    .LineStyle = xlNone
End With
Selection.Shadow = False
Selection.InvertIfNegative = False
With Selection.Interior
    .ColorIndex = 1
    .Pattern = xlSolid
End With
With ActiveChart.ChartGroups(1)
    .Overlap = 100
    .GapWidth = 0
    .HasSeriesLines = False
End With
ActiveChart.SeriesCollection(2).Select
With Selection.Border
    .Weight = xlThin
    .LineStyle = xlNone
End With
Selection.Shadow = False
Selection.InvertIfNegative = False
Selection.Fill.Patterned Pattern:=msoPatternLightUpwardDiagonal
With Selection
    .Fill.Visible = True
    .Fill.ForeColor.SchemeColor = 1
```

```
        .Fill.BackColor.SchemeColor = 2
    End With
    ActiveChart.PlotArea.Select
    With Selection.Border
        .ColorIndex = 1
        .Weight = xlHairline
        .LineStyle = xlContinuous
    End With
    Selection.Interior.ColorIndex = xlNone
    ActiveChart.ChartArea.Select
    ActiveChart.Axes(xlValue).Select
    With Selection.Border
        .ColorIndex = 1
        .Weight = xlHairline
        .LineStyle = xlContinuous
    End With
    With Selection
        .MajorTickMark = xlOutside
        .MinorTickMark = xlNone
        .TickLabelPosition = xlNextToAxis
    End With

    ActiveChart.Axes(xlValue).Select
    Selection.TickLabels.NumberFormat = "#,###0.000;#,###0.000"


    ActiveChart.ChartArea.Select
    Selection.Interior.ColorIndex = xlAutomatic
    Selection.AutoScaleFont = False
    With Selection.Font
        .Name = "Arial"
        .FontStyle = "Regular"
        .Size = 8
    End With

    Next i
    Call ArrangeMyCharts
    Application.ScreenUpdating = True
End Sub
Sub ArrangeMyCharts()
    Dim iChart As Long
    Dim nCharts As Long
    Dim dTop As Double
    Dim dLeft As Double
    Dim dHeight As Double
    Dim dWidth As Double
    Dim nColumns As Long

    dTop = 75      ' top of first row of charts
    dLeft = 600    ' left of first column of charts
    dHeight = 225  ' height of all charts
    dWidth = 275   ' width of all charts
    nColumns = 3   ' number of columns of charts
    nCharts = ActiveSheet.ChartObjects.Count

    For iChart = 1 To nCharts
        With ActiveSheet.ChartObjects(iChart)
            .Height = dHeight
            .Width = dWidth
            .Top = dTop + Int((iChart - 1) / nColumns) * dHeight
```

```
            .Left = dLeft + ((iChart - 1) Mod nColumns) * dWidth
        End With
    Next
End Sub
Sub Dependency_chart()
'
' Dependency_chart Macro
    Range("A2:D247").Select
    ActiveWindow.LargeScroll Down:=-6
    Charts.Add
    ActiveChart.ChartType = xlLine
    ActiveChart.SetSourceData Source:=Sheets("Dependency").Range("A2:D247"), _
        PlotBy:=xlColumns
    ActiveChart.Location Where:=xlLocationAsObject, Name:="Dependency"
    With ActiveChart
        .HasTitle = True
        .ChartTitle.Characters.Text = "Dependency ratios"
        .Axes(xlCategory, xlPrimary).HasTitle = False
        .Axes(xlValue, xlPrimary).HasTitle = True
        .Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "%"
    End With
    With ActiveChart.Axes(xlCategory)
        .HasMajorGridlines = False
        .HasMinorGridlines = False
    End With
    With ActiveChart.Axes(xlValue)
        .HasMajorGridlines = False
        .HasMinorGridlines = False
    End With
    ActiveChart.HasLegend = True
    ActiveChart.Legend.Select
    Selection.Position = xlBottom
    ActiveChart.PlotArea.Select
    With Selection.Border
        .ColorIndex = 16
        .Weight = xlThin
        .LineStyle = xlContinuous
    End With
    Selection.Interior.ColorIndex = xlNone
    ActiveChart.ChartArea.Select
    With Selection.Border
        .Weight = 2
        .LineStyle = 0
    End With
    Selection.Interior.ColorIndex = xlAutomatic
    ActiveChart.SeriesCollection(3).Select
    With Selection.Border
        .ColorIndex = 1
        .Weight = xlMedium
        .LineStyle = xlContinuous
    End With
    With Selection
        .MarkerBackgroundColorIndex = xlNone
        .MarkerForegroundColorIndex = xlNone
        .MarkerStyle = xlNone
        .Smooth = False
        .MarkerSize = 3
        .Shadow = False
    End With
    ActiveChart.SeriesCollection(1).Select
```

```
With Selection.Border
    .ColorIndex = 1
    .Weight = xlThin
    .LineStyle = xlDot
End With
With Selection
    .MarkerBackgroundColorIndex = xlNone
    .MarkerForegroundColorIndex = xlNone
    .MarkerStyle = xlNone
    .Smooth = False
    .MarkerSize = 3
    .Shadow = False
End With
ActiveChart.SeriesCollection(2).Select
With Selection.Border
    .ColorIndex = 1
    .Weight = xlHairline
    .LineStyle = xlContinuous
End With
With Selection
    .MarkerBackgroundColorIndex = xlNone
    .MarkerForegroundColorIndex = xlNone
    .MarkerStyle = xlNone
    .Smooth = False
    .MarkerSize = 3
    .Shadow = False
End With
ActiveChart.Legend.Select
With Selection.Border
    .Weight = xlHairline
    .LineStyle = xlNone
End With
Selection.Shadow = False
Selection.Interior.ColorIndex = xlAutomatic
ActiveChart.Axes(xlCategory).Select
With Selection.Border
    .Weight = xlHairline
    .LineStyle = xlAutomatic
End With
With Selection
    .MajorTickMark = xlOutside
    .MinorTickMark = xlNone
    .TickLabelPosition = xlLow
End With
With ActiveChart.Axes(xlCategory)
    .CrossesAt = 1
    .TickLabelSpacing = 20
    .TickMarkSpacing = 20
    .AxisBetweenCategories = True
    .ReversePlotOrder = False
End With
ActiveChart.PlotArea.Select
Selection.Top = 20
Selection.Height = 150

With ActiveSheet.ChartObjects(1)
    .Left = 200
    .Width = 450
    .Top = 75
    .Height = 300
```

```
    End With

    ActiveChart.ChartArea.Select
    Selection.AutoScaleFont = False
    With Selection.Font
        .Name = "Arial"
        .FontStyle = "Regular"
        .Size = 8
        .Strikethrough = False
        .Superscript = False
        .Subscript = False
        .OutlineFont = False
        .Shadow = False
        .Underline = xlUnderlineStyleNone
        .ColorIndex = xlAutomatic
        .Background = xlAutomatic
    End With
End Sub

Private Sub Label35_Click()

End Sub

Private Sub LifeFrame_Click()

End Sub

Private Sub OptButNo_Mode_Click()
If OptButNo_Mode = False Then
    StartFrm.TxtNOM.Visible = False
    StartFrm.LblNOM.Visible = False
    Else
    StartFrm.TxtNOM.Visible = True
    StartFrm.LblNOM.Visible = True
    StartFrm.TxtPOP.Visible = False
    StartFrm.LblNOM.Visible = False

End If

End Sub

Private Sub OptButNo_StoreMode_Click()

End Sub

Private Sub OptButYes_Mode_Click()
If OptButYes_Mode = True Then
    StartFrm.TxtPOP.Visible = True
    StartFrm.LblNOM.Visible = True
    StartFrm.TxtNOM.Visible = False
    StartFrm.LblNOM.Visible = False

End If

End Sub

Private Sub TxtTransLE_Change()

End Sub
Private Sub TFRFrame_Click()
```

```
End Sub

Private Sub TxtBoxLambda_Change()
Lmb = Val(TxtBoxLambda)
If Lmb < 0# Then
   MsgBox ("Lambda cannot be less than zero! FERTMOD is a bit bossy. It will set Lambda to one")
   TxtBoxLambda.Value = 1#
   Exit Sub
   ElseIf Lmb > 2# Then MsgBox ("Sorry, but Lambda cannot exceed 2. FERTMOD is intolerant of this error. Lambda will be set to one")
   TxtBoxLambda.Value = 1#
   Exit Sub
   ElseIf Lmb > 1# And Lmb <= 2 Then MsgBox ("FERTMOD hopes you know what you are doing - be cautious with Lambda in range 1 to 2")
End If

End Sub

Private Sub TxtBoxTheta_Change()
Theta = Val(TxtBoxTheta)
If Theta < 0# Then
   MsgBox ("Theta cannot be less than zero! Theta will be set to unity")
   TxtBoxTheta.Value = 1#
   Exit Sub
   ElseIf Theta > 2 Then MsgBox ("Theta cannot exceed 2! Sorry Dave, but FERTMOD has taken control and will set Theta to one")
   TxtBoxTheta.Value = 1#
   Exit Sub
   ElseIf Theta > 1# And Theta <= 2 Then MsgBox ("FERTMOD hopes you know what you are doing - be cautious with Theta > 1!")
End If
End Sub

Private Sub TxtTransNOM_Change()
End Sub

Private Sub UserForm_Click()
End Sub

Sub MoveSheets()
   ActiveSheet.Move _
      After:=ActiveWorkbook.Sheets(ActiveWorkbook.Sheets.Count)
      'Moves active sheet to end of active workbook.
End Sub

Sub fertility()
Dim tot_sh, LRT, Omega As Double
Const Italy As Double = 1.32 'Italian TFR in 2005 (Eurostat)
Const France As Double = 2.004 'French TFR in 2006 (Eurostat)
Dim ASFR_share(49) As Double
Dim AusF_share(49) As Double
Dim ABS_ratio(49) As Double ' This indicates the extent to which, all other things being equal,
' age-specific fertility rates can be expected to shift in the long-run evn if fertility stays at
' its 2006 value(ie 1.81).
' ABS_ratio is put into effect by multiplying the share of each ASFR in the TFR by ABS_ratio.
Dim AdjRatio(49) As Double
' This is an adjustment ratio that shifts the long-run ASFRs as the TFR deviates away from its 2006 value.
' It is based on the difference in the ASFR shares of the TFR between Italy (a low fertility country) and France (a high fertility country).
' This difference is used to provide a function that indicates how the shares change as TFR moves away from the TFR in 2006 (TFR2006)
' If TFR long-run is at TFR2006, then AdjRatio is 1 for all ages. It grows with age if TFRs fall below TFR2006
' or it goes down with age if the TFR rises. This is the typical pattern revealed by time series and cross-country ASFRs of countries with different fertility levels.
```

```
mu = 2.166916836
Gamma = -0.303822102
Beta = 0.011590909
Alpha = -0.000127709
LRT = Application.Min(longrun_TFR, 3.35)
Omega = (LRT - TFR2006) / (Italy / France * TFR2006 - TFR2006)
tot_sh = 0#

For age = 15 To 49
   ABS_ratio(age) = 1# + LambDa * (2.338288 - 0.308566 * age + 0.0103884 * age ^ 2# - 0.00009140508 * age ^ 3#)
   AdjRatio(age) = 1# + ThetaVal * Omega * (mu + Gamma * age + Beta * age ^ 2# + Alpha * age ^ 3#)
   Age_specific_TFR(age, 1) = Worksheets("Inputs").Cells(age - 11, 8).Value ' age specific fertility rate in 2006 - (15,1) is 15yr olds in 2006
   AusF_share(age) = Age_specific_TFR(age, 1) / (1000# * TFR2006) ' proportion of total TFR of each asfr in 2006
   ASFR_share(age) = AdjRatio(age) * ABS_ratio(age) * AusF_share(age)
   tot_sh = tot_sh + ASFR_share(age)
Next age

For age = 15 To 49
   LR_Age_specific_TFR(age) = longrun_TFR * 1000# * ASFR_share(age) / tot_sh
   Worksheets("Start").Cells(39 + age - 15, 2).Value = Age_specific_TFR(age, 1)
   Worksheets("Start").Cells(39 + age - 15, 3).Value = LR_Age_specific_TFR(age)
   Worksheets("Start").Cells(39 + age - 15, 4).Value = ABS_ratio(age)
   Worksheets("Start").Cells(39 + age - 15, 5).Value = AdjRatio(age)
Next age

For age = 15 To 49
   For year = 2007 To 2251

      If (year - 2006 <= PeriodM) Then
      Age_specific_TFR(age, year - 2005) = Age_specific_TFR(age, year - 2006) + (LR_Age_specific_TFR(age) - Age_specific_TFR(age, 1))
/ PeriodM
      Else
      Age_specific_TFR(age, year - 2005) = LR_Age_specific_TFR(age) 'Age_specific_TFR(15, 2) is ASFR for 15yr olds in 2007
      End If

   Next year
Next age
End Sub
```

1.7 Troubleshooting

First, it is important to ensure macros are enabled in Excel. When the macro security level in Excel is set to Low, macros can be run without prompting. When macro security is set to Medium, Excel displays a dialog box asking if you want to enable macros. When macro security is set to High, Excel allows you to run only those macros that are digitally signed or stored in the Excel startup (XLStart) folder. FERTMOD is not digitally signed and will not run in High security mode. To change macro enabling, go to the 'Tools' menu in Excel, select Options, select the 'Security' tab, click the 'Macro security' button, and choose 'Medium' or 'Low'.

Second, some users have reported that they have encountered the error message 'Can't find project or library' when running the program. The problem appears to be caused by the program not being able to reference (i.e locate) the 'Solver' object.

To restore this or any other missing references take the following steps.

1. Open the Visual basic editor (select the Tools menu, then Macros, and then Visual Basic Editor) (figure 1.11). Open the Tools menu and select ‘References’

2. The References box will open (figure 1.12). Check whether Microsoft Excel has nominated any references as ‘missing’. If so, the label ‘missing’ will appear alongside the relevant reference. For example in the screen shown in figure 1.12, if ‘SOLVER’ was missing the box next to it would be unchecked and ‘missing’ would be printed beside it. This may mean that the Solver Add-In has not been installed on your computer.

Figure 1.11 **In *Excel*, choose the ‘Tools’, then Macro, then *Visual Basic Editor*, then select the ‘Tools’ menu in the *Visual Basic Editor* and then select ‘References’**

In Excel In the Visual Basic Editor

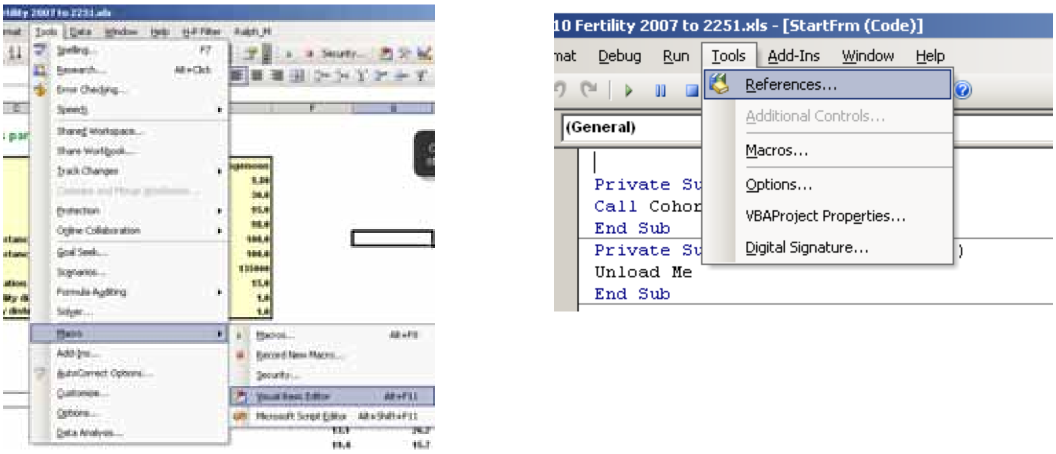
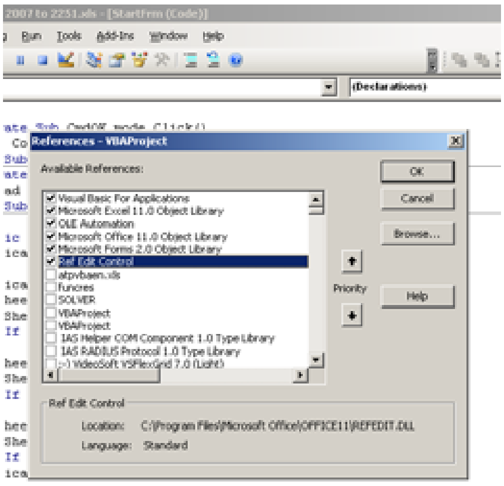


Figure 1.12 **References - VBAProject**



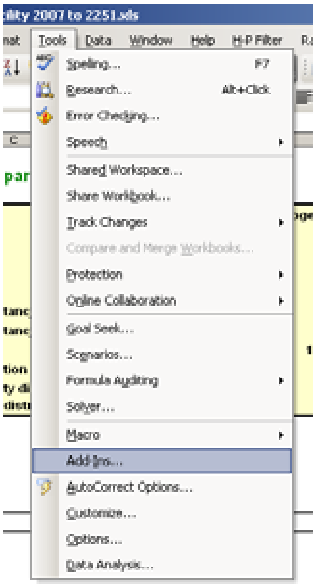
3. To install the Solver Add-In, close the VBA editor. This will return you to the standard Excel worksheet screen. Then select the Tools menu, Add-ins and ensure that the Solver Add-in is ticked (figure 1.13). Click OK.

4. Then return to Visual Basic Editor as before, open ‘Tools’ menu, References and tick the box next to SOLVER.

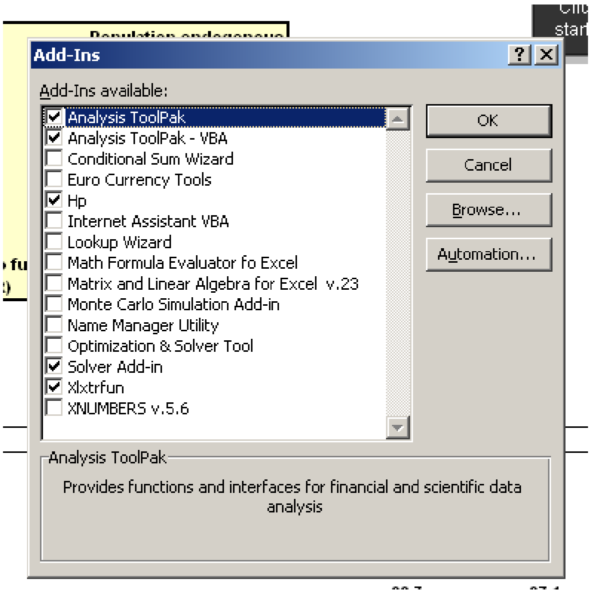
The program should now run.

Figure 1.13 **Checking Add-ins**

Choose Tools and 'Add_Ins'



Check Solver Add-in is ticked



References

ABS (Australian Bureau of Statistics) 2006, *Population Projections, Australia, 2004 to 2101*, Cat. No. 3222.0, 14 June.

PC (productivity Commission), *Demographic Projections*, Technical Paper 1, Economic Implications of an Ageing Australia, accessed from http://www.pc.gov.au/study/ageing/docs/finalreport/technicalpapers.