Institute and Faculty of Actuaries

Mortality in the 21st century

A toolkit for robust modelling

Andy Harding and Matt Fletcher

Aon Hewitt

22 November 2016

Agenda

- Background
- Modelling challenges
- Modelling solutions
- Practical implementation

Institute and Faculty of Actuaries

Institute and Faculty of Actuaries

Background

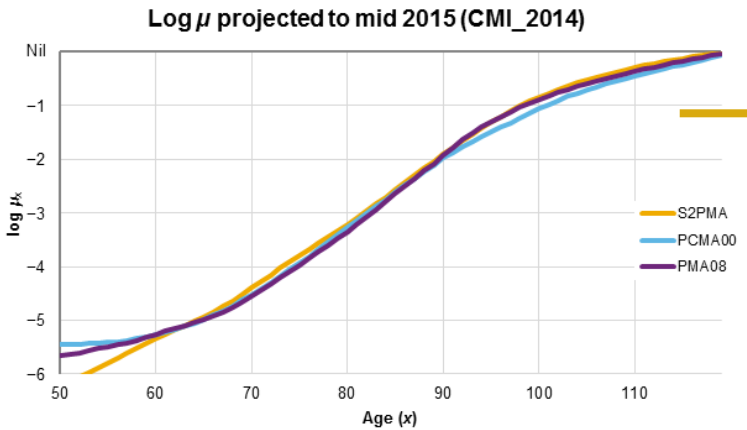
Base mortality analysis

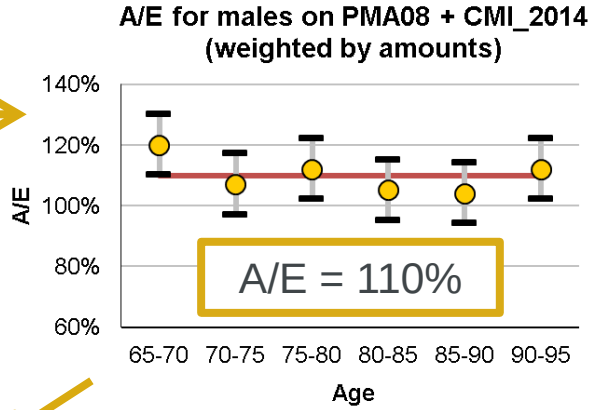
- Purpose
  - Insurance (pricing, reserving, ongoing monitoring)
  - Pension plans (funding, accounting, risk settlement)
- Application
  - Mortality (protection)
  - Longevity (annuity – group and retail)
- Stakeholders
  - Policyholders, management and shareholders
  - Plan members, trustees and sponsoring company

Institute and Faculty of Actuaries

Simplistic approach

- Choose reference table e.g. PMA08 + CMI_2014
- Scale by A/E ratio (amounts-weighted)





Adopt mortality 110% of PMA08 + CMI_2014

Institute and Faculty of Actuaries

Materiality

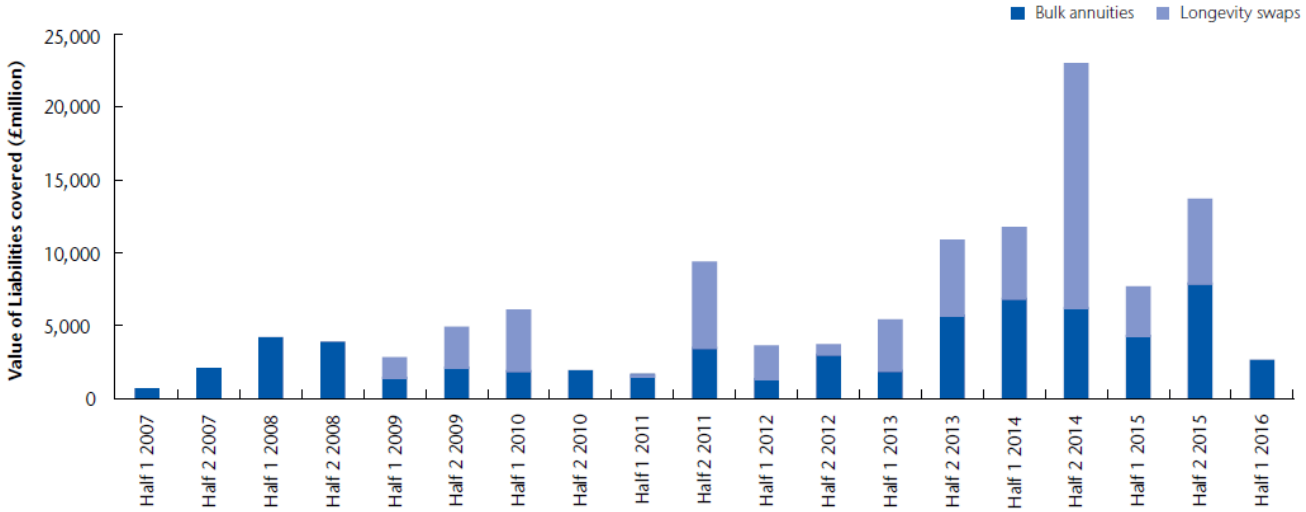
- Mortality rate overestimated by just 1% …
- … e.g. q = 3.03% vs 3.00% …
- … means annuity PV underestimated by ~0.4%

i.e. £10m mis-pricing on £2.5bn portfolio

Institute and Faculty of Actuaries

Growing sophistication – demand

- Market activity



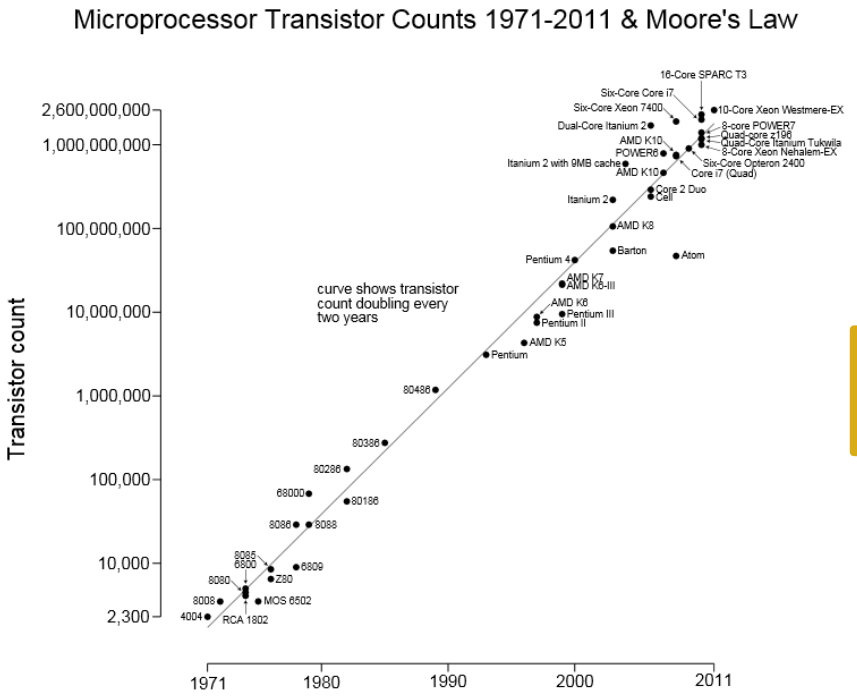
Institute and Faculty of Actuaries

Growing sophistication – supply

- Availability of data
  - Electronic records
  - Postcodes, behaviour and health
- Credibility of data
  - Volume
  - Reliability
- Computational capacity

Institute and Faculty of Actuaries

Growing sophistication – supply



Source: Wikipedia https://en.wikipedia.org/wiki/Moore%27s_law

Transistor count has doubled every 2 years

Institute and Faculty of Actuaries

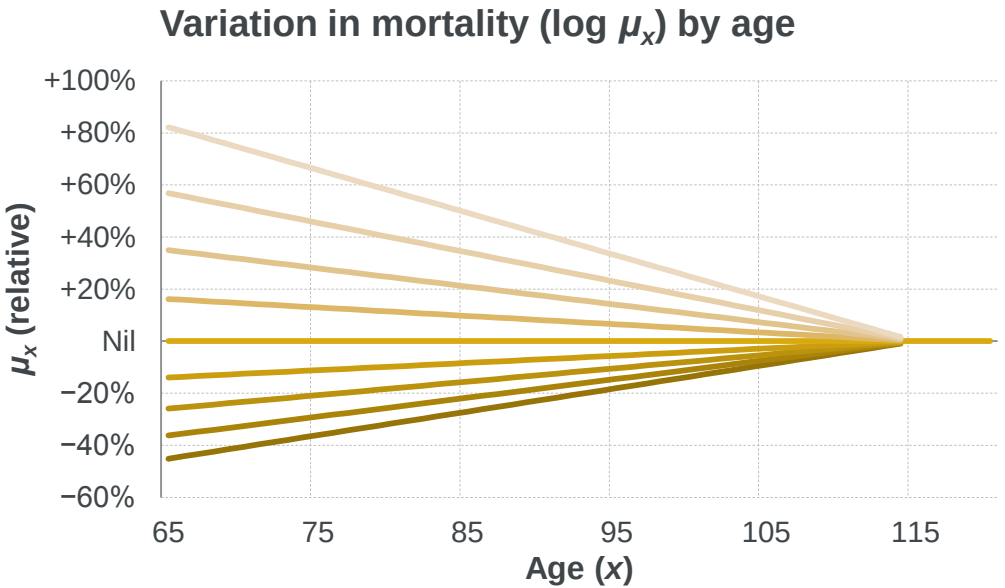
Institute and Faculty of Actuaries

Modelling challenges

22 November 2016

Heterogeneity

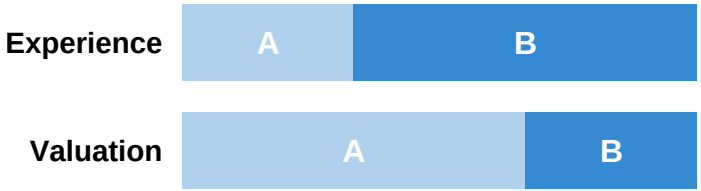
- Change in profile
- Intrinsic PV impact
- Age shape



Institute and Faculty of Actuaries

Heterogeneity – change in profile

- Population to be valued may have different profile to experience data

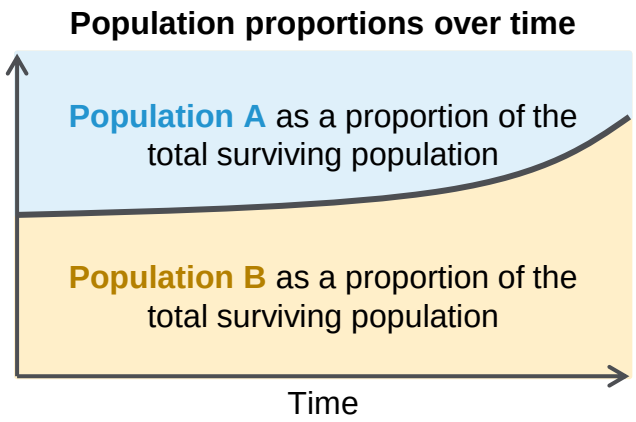


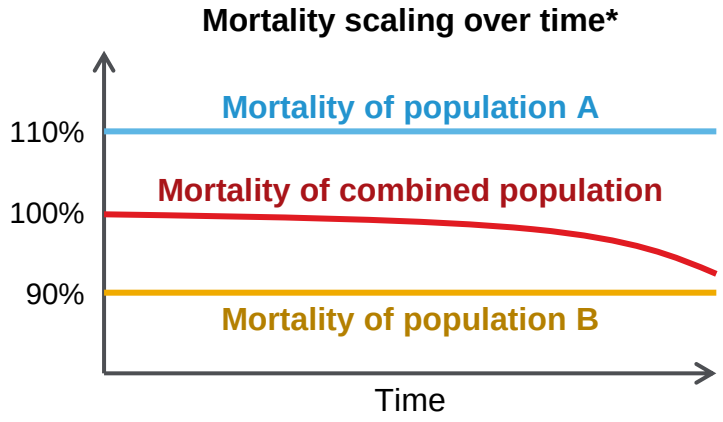
- Real-life example
  - Longevity transaction – 2 business units, distinct characteristics
  - Longer-lived business unit represented
    - 67% of the portfolio to be valued…
    - … but only 33% of the experience data
  - Simple A/E would have understated PV by 3%

Institute and Faculty of Actuaries

Heterogeneity – intrinsic PV impact

- A/E is a recent snapshot of mortality…
- … but PV is a forward-looking measure

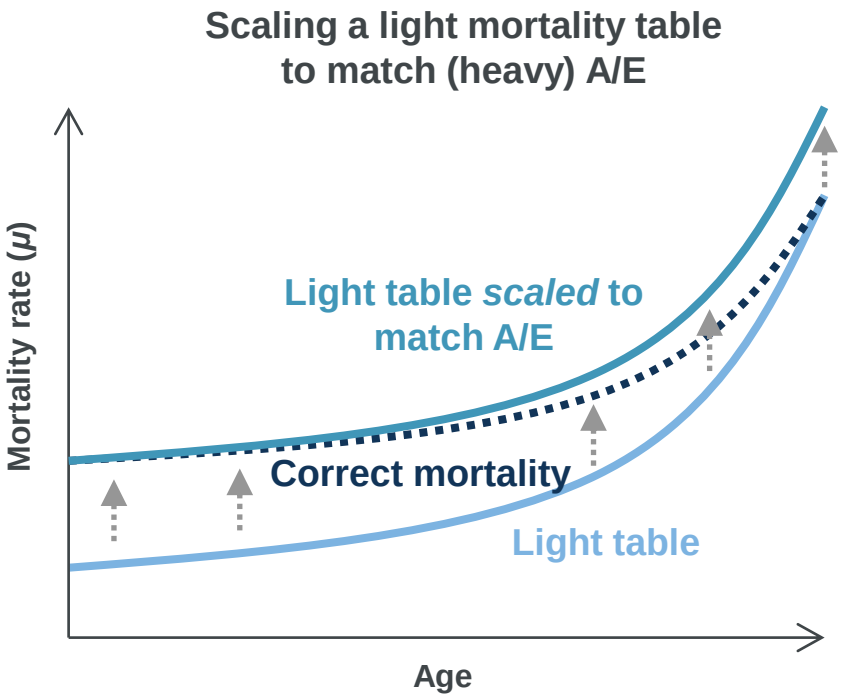




* We have ignored mortality convergence by age in order to simplify the presentation

Institute and Faculty of Actuaries

Heterogeneity – age shape

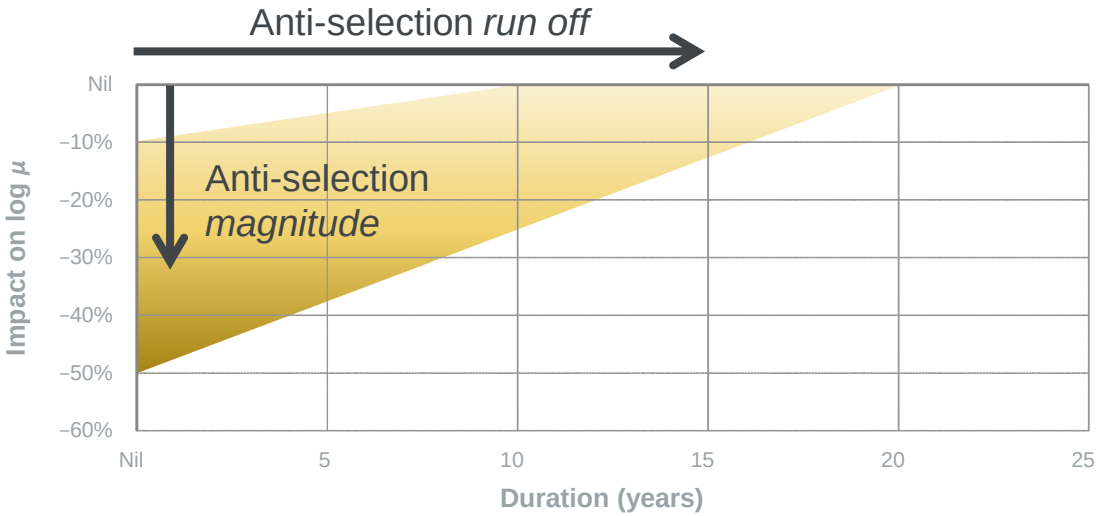


If you scale a **light table** to match A/E (which is typically weighted towards lower ages than the liabilities) then you may *understate* liabilities because the mortality shape of the **scaled table** is wrong. (The correct shape is the **dotted line**.)

Institute and Faculty of Actuaries

Anti-selection

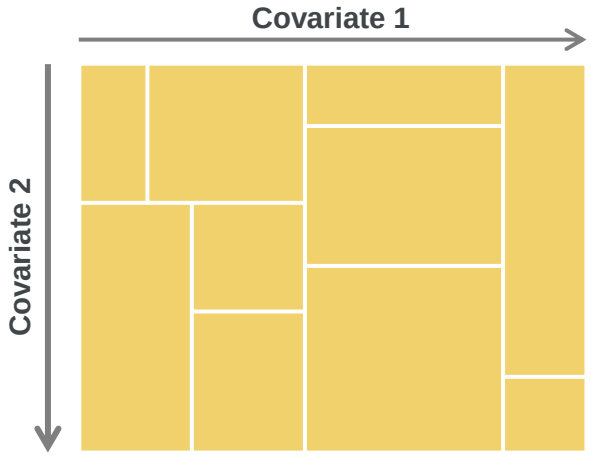
- Annuity purchase selects for mortality characteristics
- Impact depends heavily on book



Institute and Faculty of Actuaries

Covariates

- Discrete
  - Sections (exec, staff, works)
  - Product / retirement type
  - Sales channel
- Continuous
  - Benefit amount
  - High-resolution socio-economic profiling
- Co-graduation – e.g. ill-health impact across sections
- Risk of data-mining

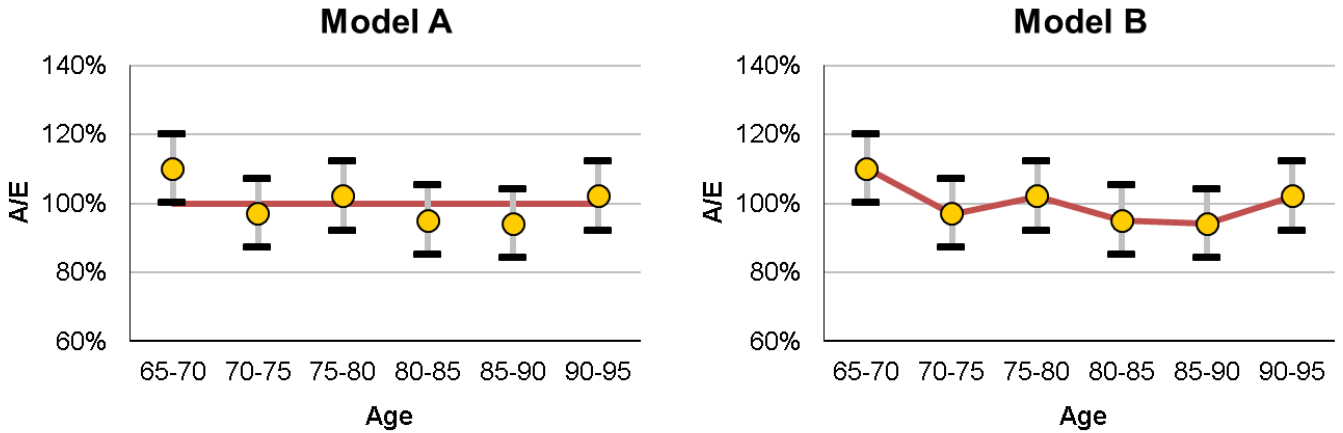


Institute and Faculty of Actuaries

Model selection

- Statistical impact (beware overfitting)
- PV impact

Which is better?



What is 'better'?

Institute and Faculty of Actuaries

Institute and Faculty of Actuaries

Modelling solutions

22 November 2016

Framework

- Proportional hazards model

$$\mu_{it}(\beta) = \mu^{Ref}{}_{it} \, exp(\beta^T \varphi_{it})$$

Reference table

Vector of factor weights (‘model parameters’)

Vector of covariates/rating factors

- This is an example of a *Generalised Linear Model*
- Advantages
  - Enormously flexible and tractable
  - Derivatives of log likelihood are trivial, so straightforward to fit
  - Mortality improvements can be built in (e.g. to the reference table)

Institute and Faculty of Actuaries

Fitting and diagnostics

- Maximise the log likelihood

$$LL = -E + A \log\mu$$

where

Death indicator

$$A = \sum_{i \in individuals} D_i$$

Date of death (random)

Mortality

$$E = \sum_{i \in individuals} \int_{v_i}^{T_i} \mu_{it} \, dt$$

Institute and Faculty of Actuaries

Fitting and diagnostics

- Maximise the log likelihood

$$LL = -E\omega + A\omega \log\mu$$

where

Death indicator

$$A\omega = \sum_{i \in individuals} D_i \omega_{iT_i}$$

Date of death (random)

Mortality

$$E\omega = \sum_{i \in individuals} \int_{v_i}^{T_i} \mu_{it}\, \omega_{it}\, dt$$

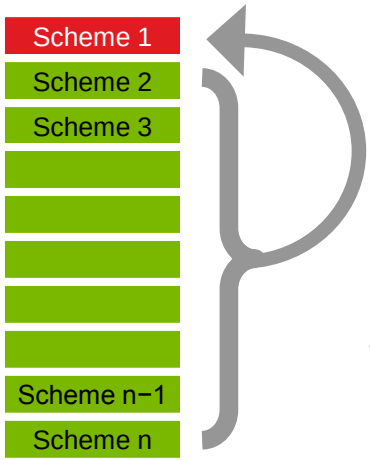
- Simple example

Flat scaling, i.e. mortality	$\mu_{it} = \alpha \mu^{Ref}{}_{it}$
Maximising *LL* leads to	$\alpha = A\omega / \left(E^{Ref}\omega\right)$
with 90% confidence ≈	$\pm 1.645 \sqrt{E\omega^2}/E\omega$

Institute and Faculty of Actuaries

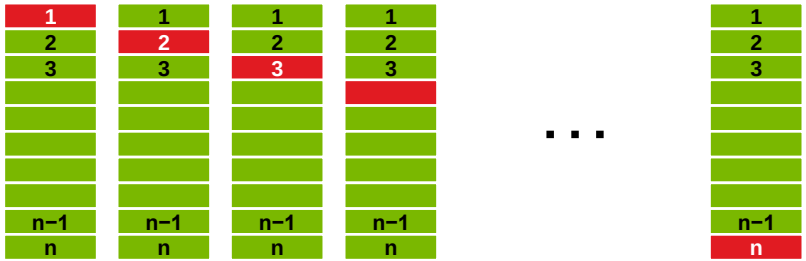
Model selection – cross-validation

- Tests actual *predictivity* across groups of interest
- But computationally onerous…
- … and not always possible to partition



Is this mortality…

…predicted by the model fitted to these schemes?



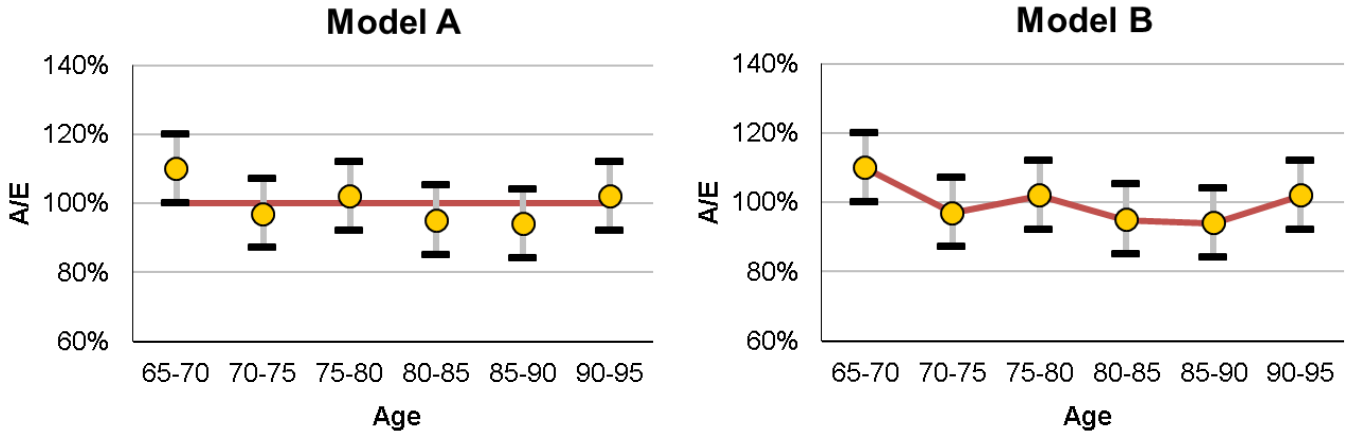
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$

Which is better?

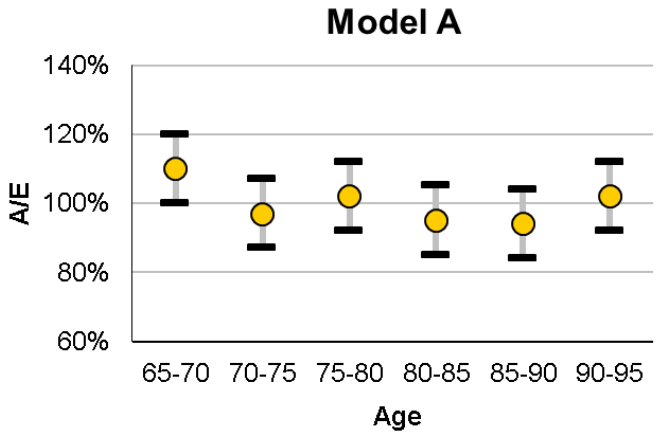


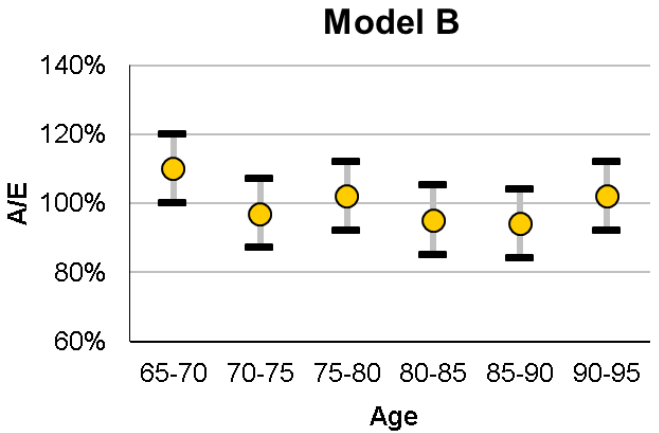
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



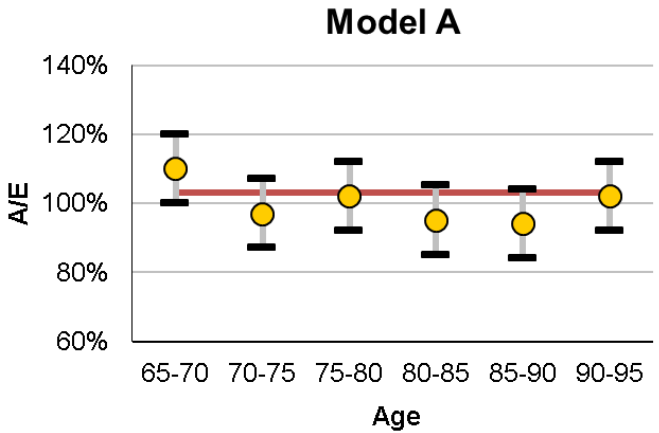


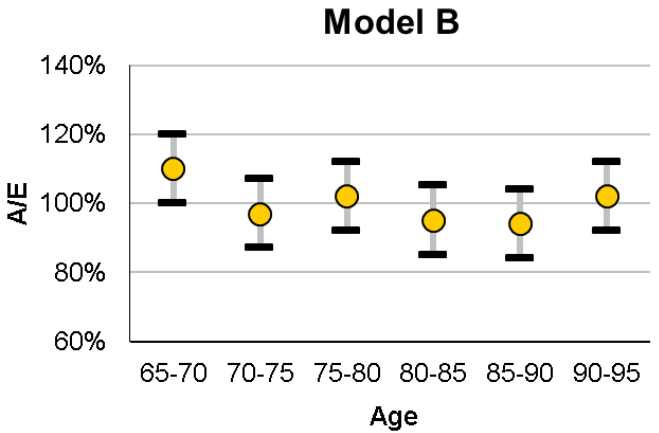
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



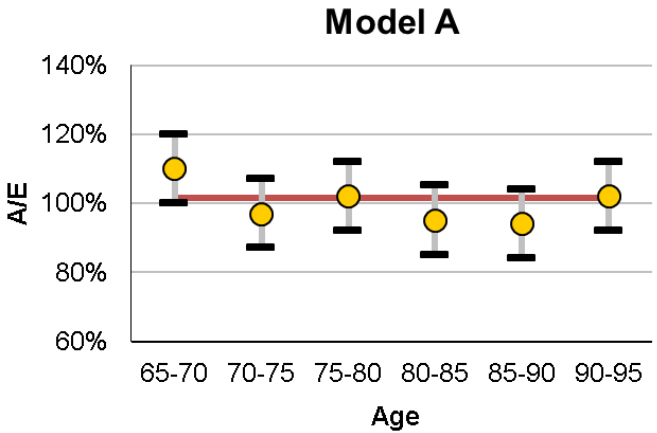


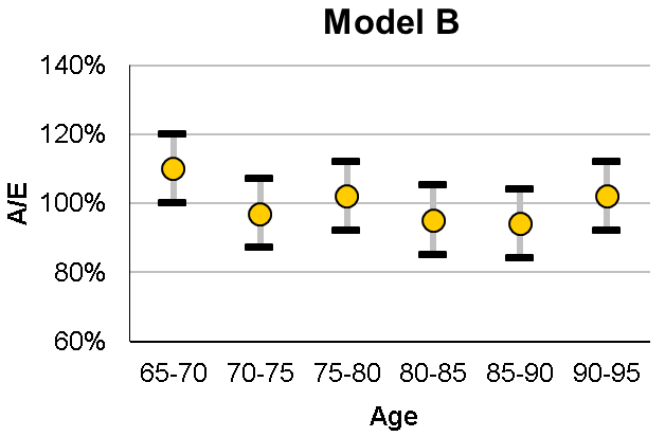
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



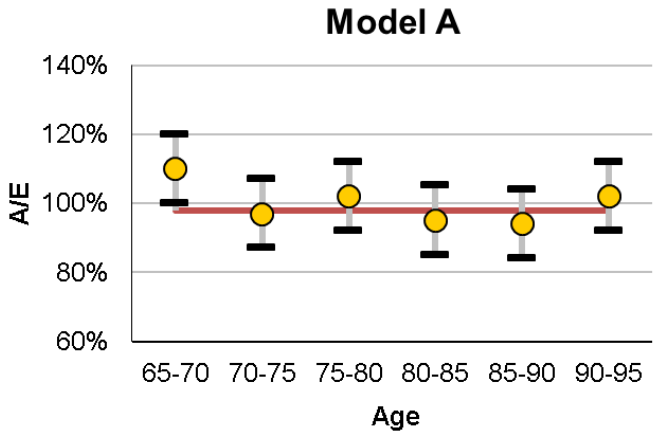


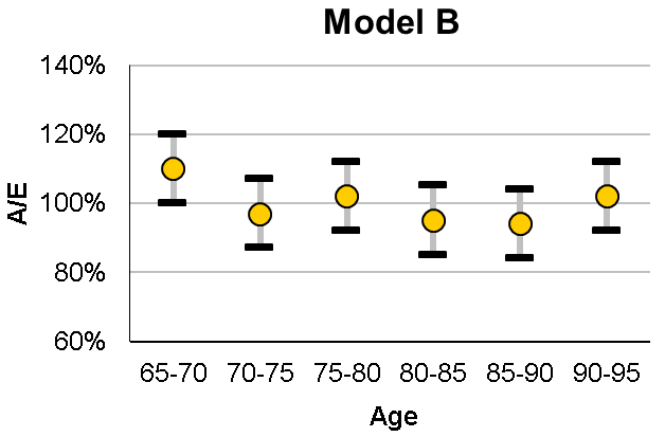
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



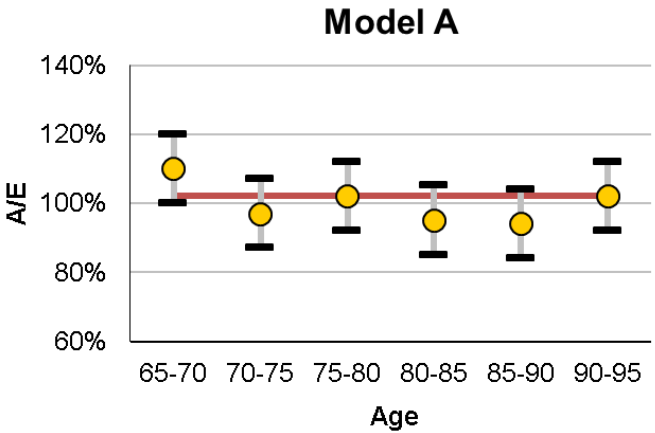


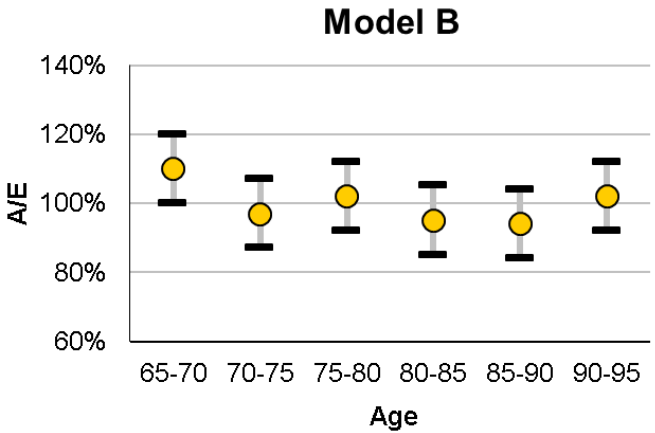
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



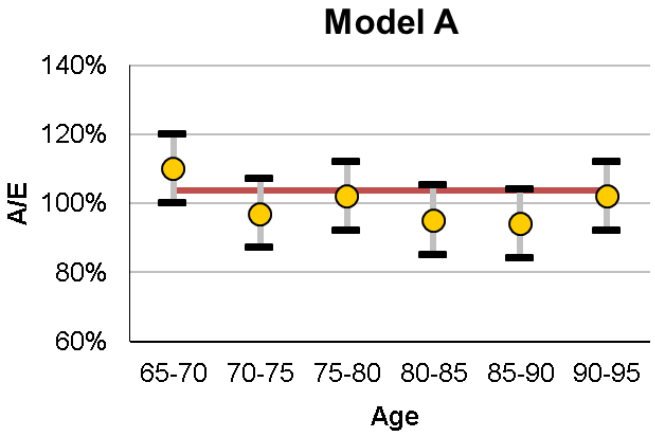


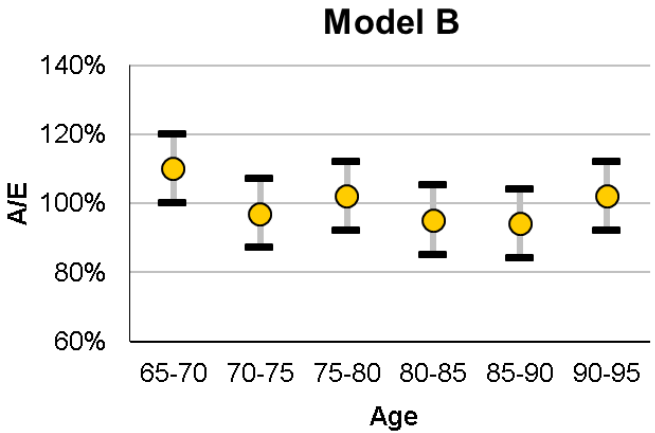
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



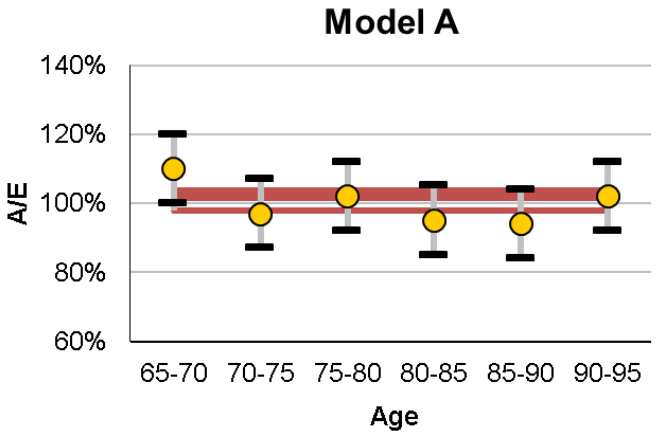


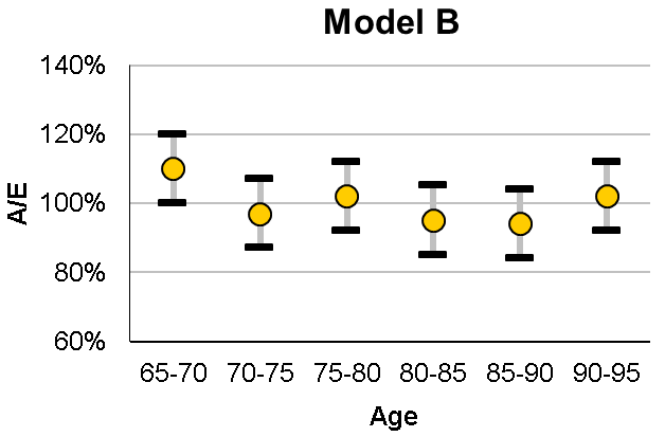
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



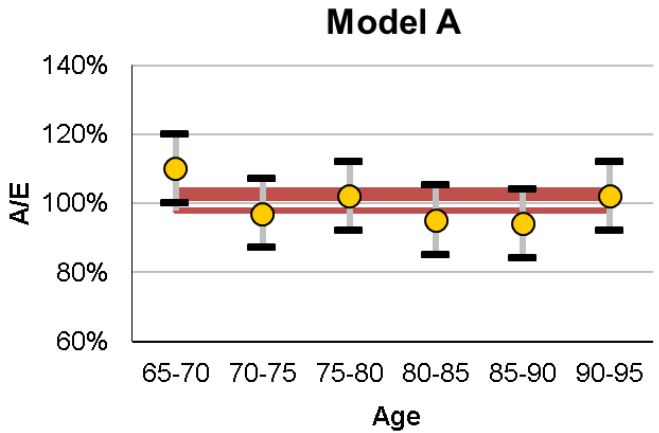


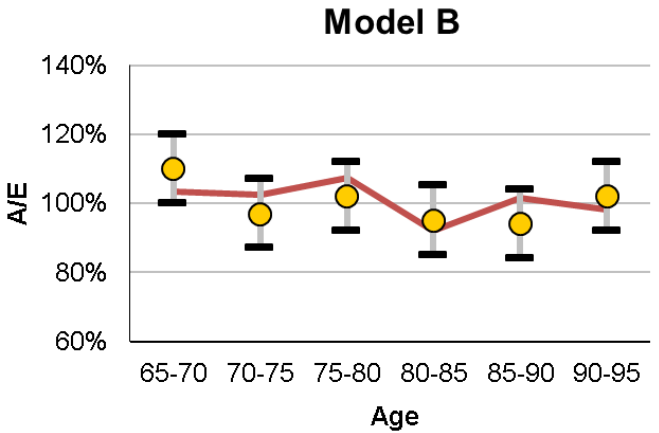
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



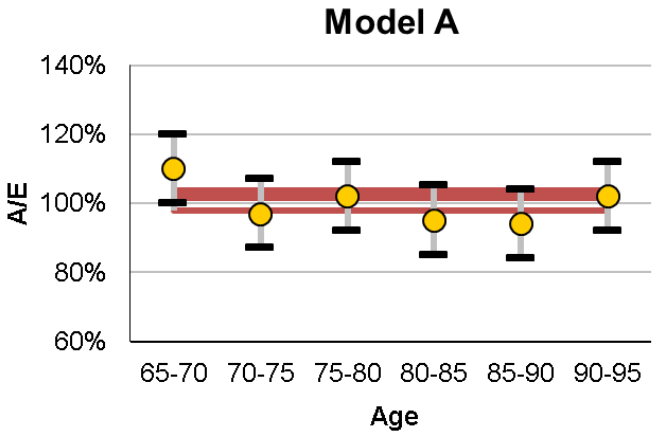


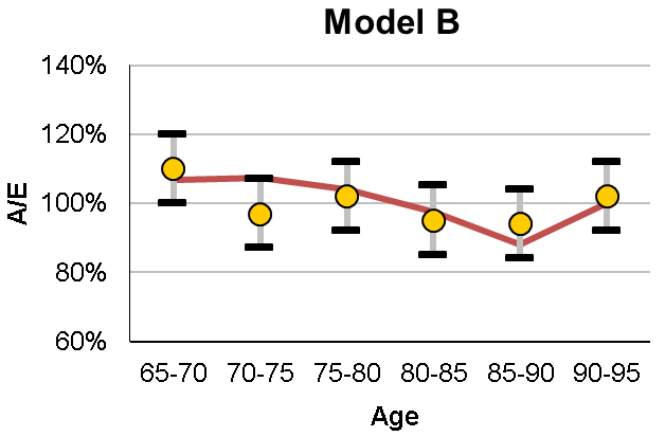
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



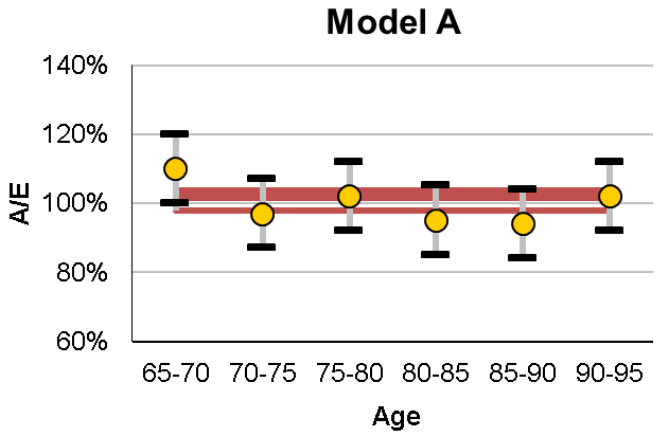


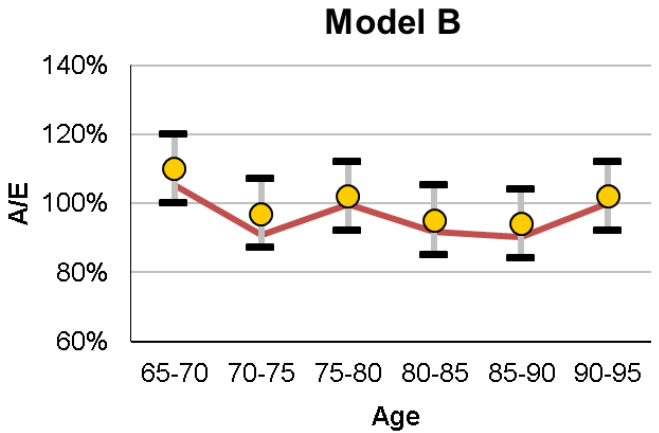
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



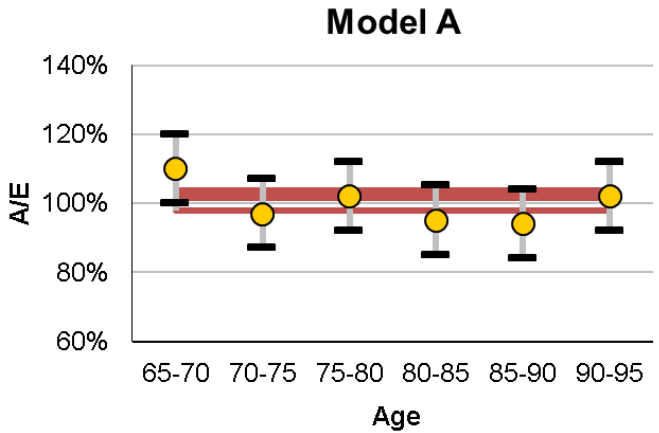


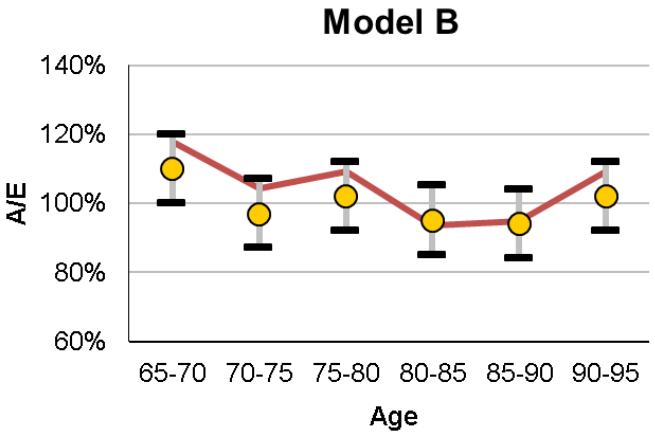
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



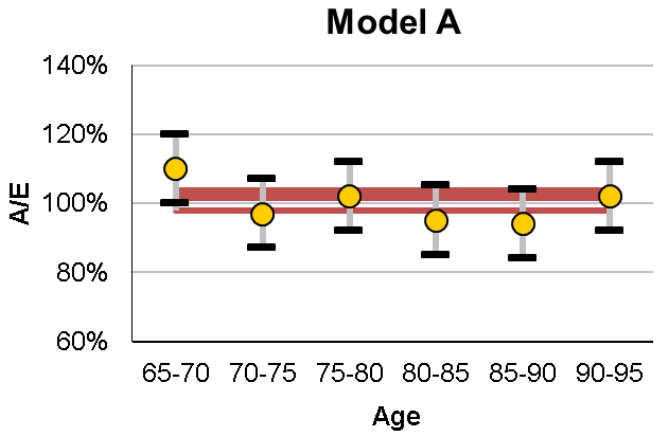


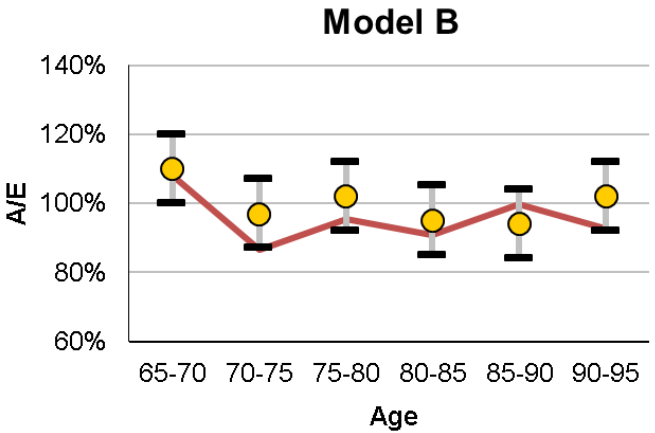
Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood

$$LL = -E\omega + A\omega \log\mu$$



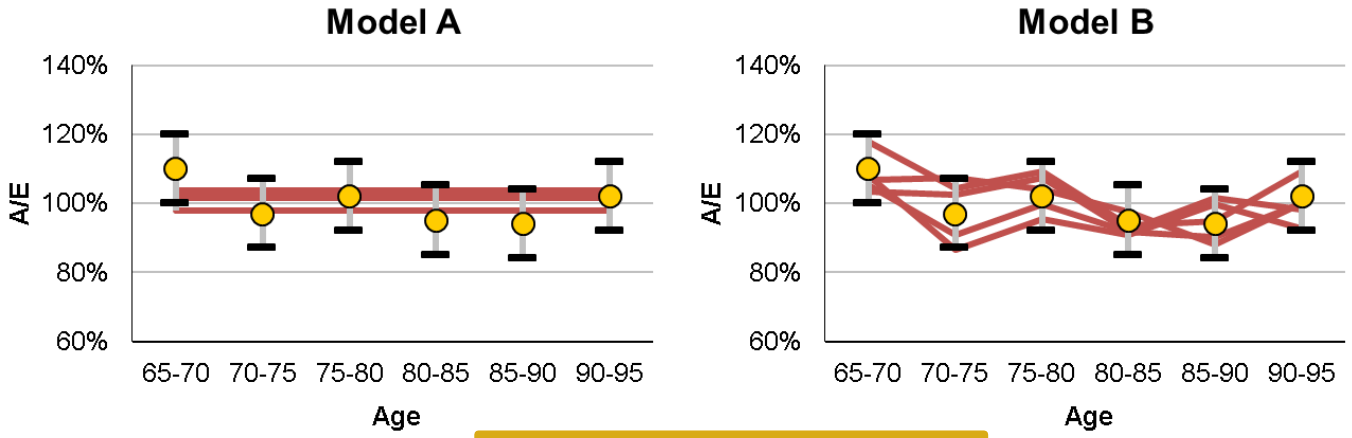


Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise log likelihood $LL = -E\omega + A\omega \log\mu$

Which is better?



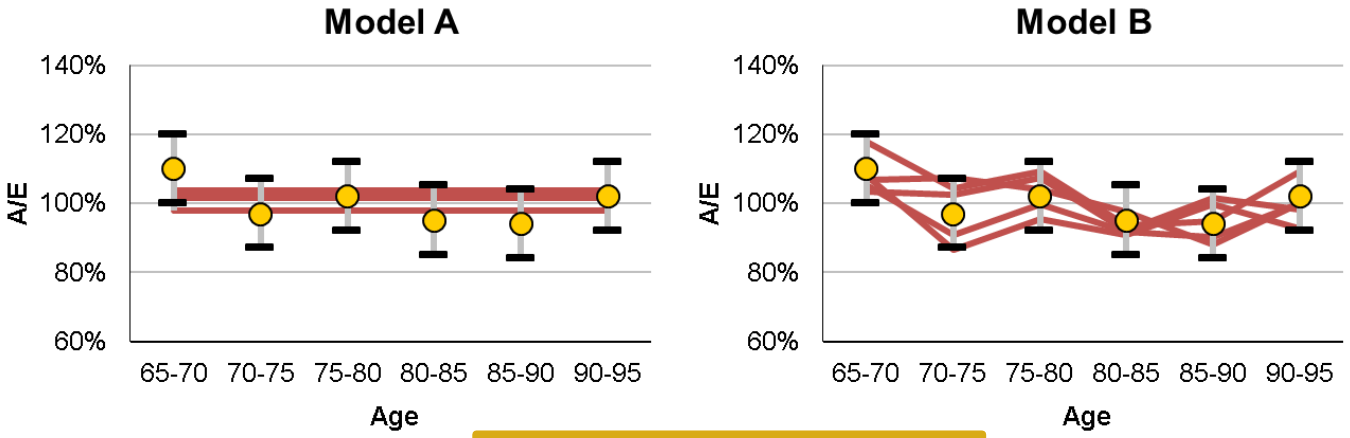
Bias-variance tradeoff

Institute and Faculty of Actuaries

Model selection – information criteria

- Maximise *penalised* log likelihood $LLP = LL - Penalty$

Which is better?



Bias-variance tradeoff

Institute and Faculty of Actuaries

Model selection – information criteria

- What is ‘better’?
- AIC: *Out-of-sample predictivity*
- Aim to maximise: $\boldsymbol{E}_{X_1}\boldsymbol{E}_{X_2}(LL(X_2 \mid \hat{\beta}(X_1))$
- Leads to: $LLP = LL - k$

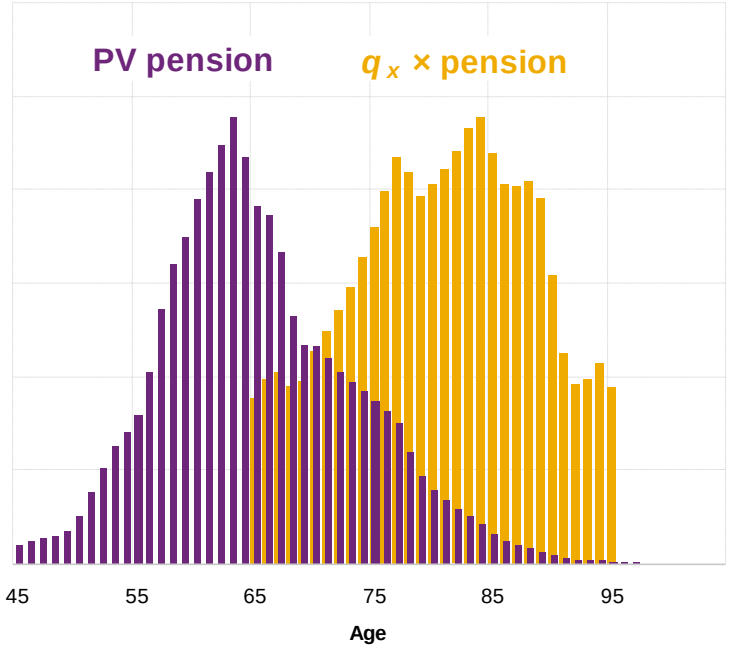
  where k is the no. of parameters
- Need to adjust for amounts-weighting and overdispersion
- Asymptotically equivalent to cross-validation…

  … BUT only if samples are unbiased!

Institute and Faculty of Actuaries

PV equivalence

- A/E scaling
  - implies a simplistic model
  - weights to the wrong population
- Better to
  - choose a sensible model
  - fit parameters:
    - point estimates
    - (co)variance
  - apply fitted model to give PV impact
- Can re-express as PV-equivalent q-scaling



Institute and Faculty of Actuaries

Example

- Candidate models for annuity portfolio
- LLP based on amounts-weighted AIC (overdispersion 3)

Fitting results for each model

Model	Penalised log likelihood (relative to best; Ω=3)	Log likelihood (relative to best; Ω=3)	Penalty	Number of covariates	Normalised penalised log likelihood (relative to best; Ω=3)	PV impact
Model A	−125,218	−178,968	42,788	2	−6.1	−0.12%
Model B	−135,554	−178,335	53,758	3	−6.6	−0.14%
Model C	−99,708	−153,360	42,886	2	−4.9	nil
Model D	−112,087	−153,107	55,519	3	−5.5	−0.03%
Model E	nil	nil	96,539	4	nil	+0.39%
Model F	−7,699	10,838	115,076	6	−0.4	+0.41%

Institute and Faculty of Actuaries

Institute and Faculty of Actuaries

Practical implementation

22 November 2016

Key features

- Safe
- Flexible
- Efficient

- Need flexible type-safe object-oriented programming environment

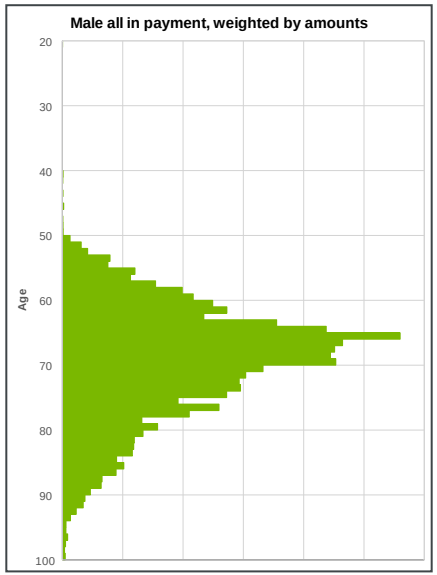
- Object-oriented programming (OOP)
  - Class – object template
  - Fields, properties and methods

Institute and Faculty of Actuaries

Simple example

Valuation data summary for male annuitants

Group	Total pension (£'000 p.a.)
15-25	54
25-35	0
35-45	25
45-55	2,145
55-65	20,164
65-75	26,341
75-85	19,889
85-95	6,373
95-105	343
Total	**75,347**




```
Member[] members = ReadMemberData(file);
double sumPension65_75 = 0.0;
double sumPension75_85 = 0.0;
foreach (var m in members)
{
   if (m.Age >= 65 && m.Age < 75)
   {
      if (m.Sex == "M") { sumPension65_75 += m.Pension; }
   }
   if (m.Age >= 75 && m.Age < 85)
   {
      if (m.Sex == "M") { sumPension75_85 += m.Pension; }
   }
}
```

Institute and Faculty of Actuaries

Safe

- DRY (Don’t Repeat Yourself)

```
Member[] members = ReadMemberData(file);
double sumPension65_75 = 0.0;
double sumPension75_85 = 0.0;
foreach (var m in members)
{
    if (m.Age >= 65 && m.Age < 75)
    {
        if (m.Sex == "M") { sumPension65_75 += m.Pension; }
    }
    if (m.Age >= 75 && m.Age < 85)
    {
        if (m.Sex == "M") { sumPension75_85 += m.Pension; }
    }
}
```

Institute and Faculty of Actuaries

Safe

- DRY (Don't Repeat Yourself)

```
Member[] members = ReadMemberData(file);
int ageStart = 65;
int ageStep = 10;
int groupCount = 2;
var sumByGroup = new double[groupCount];
foreach (var m in members)
{
   int ageLow = ageStart;
   for (int i = 0; i < groupCount ; ++i)
   {
      if (m.Age >= ageLow && m.Age < ageLow + ageStep)
      {
         if (m.Sex == "M") { sumByGroup[i] += m.Pension; }
      }
      ageLow += ageStep;
   }
}
```

Institute and Faculty of Actuaries

Safe

- DRY (Don’t Repeat Yourself)
- Type safety

```
Member[] members = ReadMemberData(file);
int ageStart = 65;
int ageStep = 10;
int groupCount = 2;
var sumByGroup = new double[groupCount];
foreach (var m in members)
{
   int ageLow = ageStart;
   for (int i = 0; i < groupCount ; ++i)
   {
      if (m.Age >= ageLow && m.Age < ageLow + ageStep)
      {
         if (m.Sex == "M") { sumByGroup[i] += m.Pension; }
      }
      ageLow += ageStep;
   }
}
```

Institute and Faculty of Actuaries

Safe

- DRY (Don't Repeat Yourself)
- Type safety

```
Member[] members = ReadMemberData(file);
int ageStart = 65;
int ageStep = 10;
int groupCount = 2;
var sumByGroup = new double[groupCount];
foreach (var m in members)
{
   int ageLow = ageStart;
   for (int i = 0; i < groupCount ; ++i)
   {
      if (m.Age >= ageLow && m.Age < ageLow + ageStep)
      {
         if (m.Sex == Sex.Male) { sumByGroup[i] += m.Pension; }
      }
      ageLow += ageStep;
   }
}
```

Institute and Faculty of Actuaries

Safe

- DRY (Don’t Repeat Yourself)
- Type safety
- Time safety

```
Member[] members = ReadMemberData(file);
int ageStart = 65;
int ageStep = 10;
int groupCount = 2;
var sumByGroup = new double[groupCount];
foreach (var m in members)
{
   int ageLow = ageStart;
   for (int i = 0; i < groupCount ; ++i)
   {
      if (m.Age >= ageLow && m.Age < ageLow + ageStep)
      {
         if (m.Sex == Sex.Male) { sumByGroup[i] += m.Pension; }
      }
      ageLow += ageStep;
   }
}
```

Institute and Faculty of Actuaries

Safe

- DRY (Don't Repeat Yourself)
- Type safety
- Time safety

```
Member[] members = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
var sumByGroup = new double[groupCount];
foreach (var m in members)
{
   var ageLow = ageStart;
   for (int i = 0; i < groupCount; ++i)
   {
      var age = asAtDate - m.BirthDate;
      if (age >= ageLow && age < ageLow + ageStep)
      {
         if (m.Sex == Sex.Male) { sumByGroup[i] += m.Pension; }
      }
      ageLow += ageStep;
   }
}
```

Institute and Faculty of Actuaries

Safe

- DRY (Don't Repeat Yourself)
- Type safety
- Time safety
- Immutability

```
Member[] members = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
var sumByGroup = new double[groupCount];
foreach (var m in members)
{
   var ageLow = ageStart;
   for (int i = 0; i < groupCount; ++i)
   {
      var age = asAtDate - m.BirthDate;
      if (age >= ageLow && age < ageLow + ageStep)
      {
         if (m.Sex == Sex.Male) { sumByGroup[i] += m.Pension; }
      }
      ageLow += ageStep;
   }
}
```

Institute and Faculty of Actuaries

Safe

- DRY (Don't Repeat Yourself)
- Type safety
- Time safety
- Immutability

```
MemberData memberData = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
var sumByGroup = new double[groupCount];
foreach (var m in memberData)
{
   var ageLow = ageStart;
   for (int i = 0; i < groupCount; ++i)
   {
      var age = asAtDate - m.BirthDate;
      if (age >= ageLow && age < ageLow + ageStep)
      {
         if (m.Sex == Sex.Male) { sumByGroup[i] += m.Pension; }
      }
      ageLow += ageStep;
   }
}
```

Institute and Faculty of Actuaries

Flexible

- Modular programming

```
MemberData memberData = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
var sumByGroup = new double[groupCount];
foreach (var m in memberData)
{
   var ageLow = ageStart;
   for (int i = 0; i < groupCount; ++i)
   {
      var age = asAtDate - m.BirthDate;
      if (age >= ageLow && age < ageLow + ageStep)
      {
         if (m.Sex == Sex.Male) { sumByGroup[i] += m.Pension; }
      }
      ageLow += ageStep;
   }
}
```

Institute and Faculty of Actuaries

Flexible

- Modular programming

```
MemberData memberData = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
Document.AddPenTable(memberData, asAtDate, Sex.Male,
    ageStart, ageStep, groupCount);
```

Institute and Faculty of Actuaries

Flexible

- Modular programming

```
MemberData memberData = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
Document.AddPenTable(memberData, asAtDate, Sex.Male,
    ageStart, ageStep, groupCount);
```

Institute and Faculty of Actuaries

Flexible

- Modular programming

```
MemberData memberData = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
Document.AddTable(memberData, asAtDate, Sex.Male,
    ageStart, ageStep, groupCount, m => m.Pension);
```

Institute and Faculty of Actuaries

Flexible

- Modular programming

```
MemberData memberData = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
Document.AddTable(memberData, asAtDate, Sex.Male,
    ageStart, ageStep, groupCount, m => 1.0);
```

Institute and Faculty of Actuaries

Flexible

- Modular programming

```
MemberData memberData = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
    foreach (var weight in weights)
    {
        Document.AddTable(memberData, asAtDate, sex,
            ageStart, ageStep, groupCount, weight);
    }
}
```

Institute and Faculty of Actuaries

Flexible

- Modular programming
- Alternative file formats

```
MemberData memberData = ReadMemberData(file);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
   foreach (var weight in weights)
   {
      Document.AddTable(memberData, asAtDate, sex,
         ageStart, ageStep, groupCount, weight);
   }
}
```

Institute and Faculty of Actuaries

Flexible

- Modular programming
- Alternative file formats

```
MemberData memberData = ReadMemberData(file, Format.Xlsx);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
   foreach (var weight in weights)
   {
      Document.AddTable(memberData, asAtDate, sex,
         ageStart, ageStep, groupCount, weight);
   }
}
```

Institute and Faculty of Actuaries

Flexible

- Modular programming
- Alternative file formats

```
MemberData memberData = ReadMemberData(file, Format.Accdb);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
   foreach (var weight in weights)
   {
      Document.AddTable(memberData, asAtDate, sex,
         ageStart, ageStep, groupCount, weight);
   }
}
```

Institute and Faculty of Actuaries

Flexible

- Modular programming
- Alternative file formats

```
MemberData memberData = ReadMemberData(file, Format.Txt);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
   foreach (var weight in weights)
   {
      Document.AddTable(memberData, asAtDate, sex,
         ageStart, ageStep, groupCount, weight);
   }
}
```

Institute and Faculty of Actuaries

Flexible

- Modular programming
- Alternative file formats

```
MemberData memberData = ReadMemberData(file, Format.Csv);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
   foreach (var weight in weights)
   {
      Document.AddTable(memberData, asAtDate, sex,
         ageStart, ageStep, groupCount, weight);
   }
}
```

Institute and Faculty of Actuaries

Efficient

- Data compression
- Parallel processing

```
MemberData memberData = ReadMemberData(file, Format.Csv);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
   foreach (var weight in weights)
   {
      Document.AddTable(memberData, asAtDate, sex,
         ageStart, ageStep, groupCount, weight);
   }
}
```

Institute and Faculty of Actuaries

Efficient

- Data compression
- Parallel processing

```
MemberData memberData = ReadMemberData(file, Format.Csv);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
    Parallel.For(0, weights.Length, i =>
    {
        var weight = weights[i];
        Document.AddTable(memberData, asAtDate, sex
            , ageStart, ageStep, groupCount, weight);
    });
}
```

Institute and Faculty of Actuaries

Efficient

- Data compression
- Parallel processing

```
MemberData memberData = ReadMemberData(file, Format.Csv);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
    Parallel.For(0, weights.Length, i =>
    {
        var weight = weights[i];
        Document.AddTable(memberData, asAtDate, sex
            , ageStart, ageStep, groupCount, weight);
    });
}
```

Institute and Faculty of Actuaries

Efficient

- Data compression
- Parallel processing

```
MemberData memberData = ReadMemberData(file, Format.Csv);
var asAtDate = Time.BeginDay(2016, 12, 1);
var ageStart = TimeDiff.FromYears(65);
var ageStep = TimeDiff.FromYears(10);
int groupCount = 2;
foreach (var sex in new[] { Sex.Male, Sex.Female })
{
   var tables = new Table[weights.Length];
   Parallel.For(0, weights.Length, i =>
   {
      var weight = weights[i];
      tables[i] = CreateTable(memberData, asAtDate, sex
         , ageStart, ageStep, groupCount, weight);
   });
   for (int i = 0; i < weights.Length; ++i)
   {
      Document.Add(tables[i]);
   }
}
```

Institute and Faculty of Actuaries

Institute and Faculty of Actuaries

Recap

22 November 2016

Recap

- Background
- Modelling challenges
- Modelling solutions
- Practical implementation

Institute and Faculty of Actuaries

Questions

Comments

Expressions of individual views by members of the Institute and Faculty of Actuaries and its staff are encouraged.

The views expressed in this presentation are those of the presenter.

Institute and Faculty of Actuaries