Spesifikasjon finansmarked og lønnsdynamikk

Finansmarked

```
μS = 0.10;
μB = 0.05;
σS = 0.20;
σB = 0.05;
ρ = 0.40;
```

Lønnsdynamikk

```
λ = 0.03;
```

Person

Persondata

```
x = 30;
ω = 120;
lønn = 300 000;
```

Dødelighet

GM-parametre underliggende statisk modell

$$\{\alpha, \beta, c\} = \{0.00007809, 0.00000719, 10^{0.04893}\};$$

Dynamisk modell

$$\text{vStat}[y\_] := \alpha + \beta\, c^{y}$$

$$\text{vDyn}[y\_, t\_, l\_] := \text{vStat}\left[y + t - \frac{t}{l}\right]$$

$$\text{pLeve}[y\_, t\_, s\_, l\_] := \text{Exp}\left[-\int_0^s \text{vDyn}[y, t+\tau, l]\, d\tau\right];$$

Kontodynamikk med deterministisk avkastning i kontinuerlig tid

$$\text{spareSaldo}[n\_?\text{NumberQ}, r\_, l\_] := \text{NIntegrate}\left[\frac{(1+\lambda)^{s}\,(1+r)^{n-s}}{\text{pLeve}[x, s, n-s, l]}, \{s, 0, n\}\right]$$

$$\text{engangsPremie}[n\_?\text{NumberQ}, i\_, l\_] := \text{NIntegrate}[(1+i)^{-s}\,\text{pLeve}[x, n, s, l], \{s, 0, \omega - x - n\}];$$

$$\text{spareRate}[n\_, \text{pensjonsProsent}\_, \text{iInnskudd}\_, \text{iEngangspremie}\_, l\_] := \frac{\text{pensjonsProsent}\,(1+\lambda)^{n}\,\text{engangsPremie}[n, \text{iEngangspremie}, l]}{\text{spareSaldo}[n, \text{iInnskudd}, l]}$$

$$\text{pensjonsProsent}[n\_, \text{spareRate}\_, \text{iInnskudd}\_, \text{iEngangspremie}\_, l\_] := \frac{\text{spareRate}\;\text{spareSaldo}[n, \text{iInnskudd}, l]}{(1+\lambda)^{n}\,\text{engangsPremie}[n, \text{iEngangspremie}, l]}$$

Kalibrering av sparerate og mulige pensjonsaldre ved dynamisk dødelighet

```
målsattPensjonsProsent = .2;
kalibrertSparerate = spareRate[37, målsattPensjonsProsent, .06, .03, ∞]
```

```
0.0342806
```

```
muligPensjonsalder[l_] := x + n /. FindRoot[målsattPensjonsProsent ==
     pensjonsProsent[n, kalibrertSparerate, .06, .03, l], {n, 37}];
```

```
eT[a_, b_, l_] := NIntegrate[pLeve[x, 0, v, l], {v, a, b}]
```

```
pensjonsProspekt[l_] := Module[{pVedL, eTAktiv, eTPensjonist},
  pVedL = muligPensjonsalder[l]; eTAktiv = eT[0, pVedL - x, l];
  eTPensjonist = eT[pVedL - x, ω - x, l]; {l, pVedL, eTAktiv, eTPensjonist, eTPensjonist/eTAktiv}]
```

```
pensjonsProspekt /@ {∞, 15, 10, 5}
```

```
{{∞, 67., 35.9895, 14.6297, 0.4065}, {15, 68.5811, 37.5473, 16.0334, 0.427018},
 {10, 69.42, 38.375, 16.8327, 0.438636}, {5, 72.1468, 41.0715, 19.7346, 0.480493}}
```

Kalibrering av sparerate ved helårlig forskuddsvis innskuddsbetaling

$$\texttt{spareSaldoHelårligForskudd[n\_, r\_, l\_]} := \sum_{s=0}^{n-1} \frac{(1+\lambda)^s\,(1+r)^{n-s}}{\texttt{pLeve[x, s, n - s, l]}}$$

$$\texttt{spareRateHelårligForskudd[n\_, pensjonsProsent\_, iInnskudd\_, iEngangspremie\_, l\_]} := \frac{\texttt{pensjonsProsent}\ (1+\lambda)^n\ \texttt{engangsPremie[n, iEngangspremie, l]}}{\texttt{spareSaldoHelårligForskudd[n, iInnskudd, l]}}$$

$$\texttt{pensjonsProsentHelårligForskudd[n\_, spareRate\_, iInnskudd\_, iEngangspremie\_, l\_]} := \frac{\texttt{spareRate spareSaldoHelårligForskudd[n, iInnskudd, l]}}{(1+\lambda)^n\ \texttt{engangsPremie[n, iEngangspremie, l]}}$$

```
målsattPensjonsProsent = .2;
kalibrertHelårligForskuddSparerate =
  spareRateHelårligForskudd[37, målsattPensjonsProsent = .2, .06, .03, ∞]
```

```
0.0337543
```

Simulerte baner for avkastning for de to finansielle eiendelene: Forhåndsgenerer for de senere beregninger på innskuddssparingen, slik at finansmarkedsrealiseringene er de samme for de ulike alternativene med aksjevekting

```
nSim = 10 000;
nTilPensjonsalder67 = 67 - x;
```

```
simAvkastningBane[simNo_] :=
  Module[{noises}, noises = {{1, 0}, {ρ, Sqrt[1 - ρ^2]}}.{RandomReal[NormalDistribution[0, 1],
      nTilPensjonsalder67], RandomReal[NormalDistribution[0, 1], nTilPensjonsalder67]};
   Transpose[{e^((μS - σS^2/2) + σS noises[[1]]) - 1, e^((μB - σB^2/2) + σB noises[[2]]) - 1}]];
```

```
avkastninger = Table[simAvkastningBane[i], {i, 1, nSim}];
```

```
beregnAkkFaktorer[aksjeAndel_] := Module[{porteføljeAvkastning},
   porteføljeAvkastning = avkastninger.{aksjeAndel, 1 - aksjeAndel};
   Table[Reverse[Delete[FoldList[#1 (1 + #2) &, 1, Reverse[porteføljeAvkastning⟦n⟧]], 1]],
    {n, 1, nSim}]];
```

$$\text{aksjeAndelForAvk06} = \frac{.06 - (\text{Exp}[\mu B] - 1)}{(\text{Exp}[\mu S] - 1) - (\text{Exp}[\mu B] - 1)}$$

```
0.161947
```

Innskuddssparing med stokastisk avkastning.

```
spareSaldiHelårligForskuddStokastisk[n_, aksjeAndel_, l_] :=
 Module[{pLeveTab, akkFaktorer}, akkFaktorer = beregnAkkFaktorer[aksjeAndel]; pLeveTab =
   Table[pLeve[x, s, n - s, l], {s, 0, n - 1}];
```

$$\left(\sum_{s=0}^{n-1} \frac{(1+\lambda)^s \, \#1⟦s+1⟧}{\text{pLeveTab}⟦s+1⟧} \;\&\right) \text{/@ akkFaktorer}\Big]$$

```
saldiForAvk06 =
  spareSaldiHelårligForskuddStokastisk[nTilPensjonsalder67, aksjeAndelForAvk06, ∞];
```

$$\text{pNivåForAvk06} = \frac{\text{kalibrertHelårligForskuddSparerate saldiForAvk06}}{(1+\lambda)^{\text{nTilPensjonsalder67}}\ \text{engangsPremie}[\text{nTilPensjonsalder67}, .03, \infty]};$$

```
Mean[pNivåForAvk06]
Median[pNivåForAvk06]
```

$$\text{probPnivåMinst20} = \text{N}\left[\frac{\text{Length}[\text{Select}[\text{pNivåForAvk06}, \#1 \geq 0.2 \;\&]]}{\text{Length}[\text{pNivåForAvk06}]}\right]$$

```
0.199434

0.191972

0.4357
```

$$\text{probPnivåOTPMinst20} = \text{N}\left[\frac{\text{Length}\left[\text{Select}\left[\frac{.02\ \text{pNivåForAvk06}}{\text{kalibrertHelårligForskuddSparerate}}, \#1 \geq 0.2 \;\&\right]\right]}{\text{Length}[\text{pNivåForAvk06}]}\right]$$

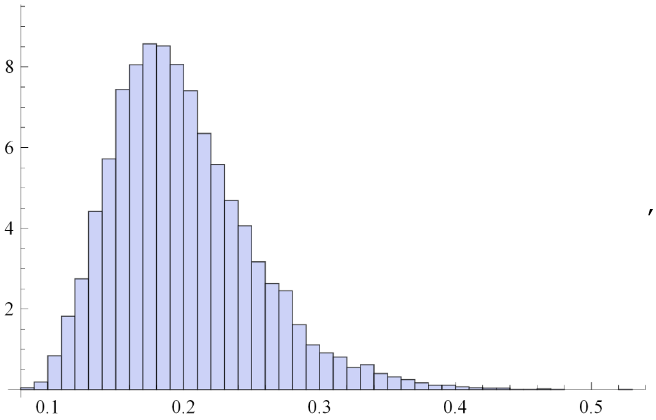
```
0.017
```

```
testModul[aksjeAndel_, l_] :=
 Module[{testPNivå}, testPNivå = (kalibrertHelårligForskuddSparerate
      spareSaldiHelårligForskuddStokastisk[37, aksjeAndel, l]) /
    ((1 + λ)^nTilPensjonsalder67 engangsPremie[nTilPensjonsalder67, .03, l]);
  {l, aksjeAndel, Histogram[testPNivå, Automatic, "ProbabilityDensity"],
   Mean[testPNivå], Median[testPNivå], StandardDeviation[testPNivå],
   N[Length[Select[testPNivå, #1 ≥ 0.2 &]] / Length[testPNivå]]}]
```

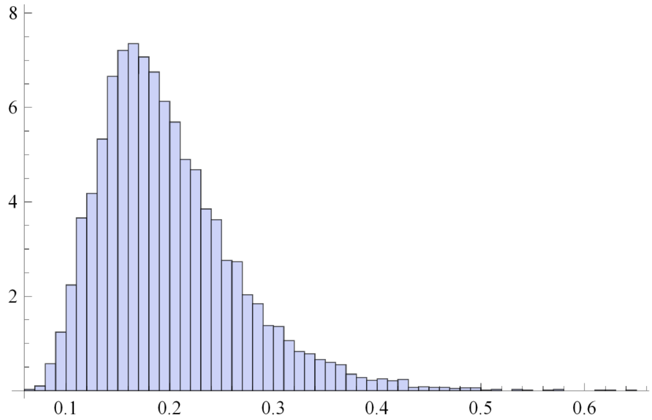
```
testPar = {{aksjeAndelForAvk06, Infinity}, {aksjeAndelForAvk06 + .116, 15},
  {aksjeAndelForAvk06 + .171, 10}, {aksjeAndelForAvk06 + .320, 5}}
```

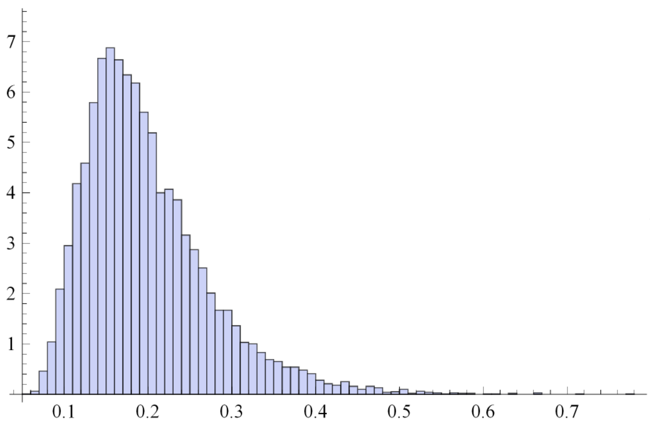
```
{{0.161947, ∞}, {0.277947, 15}, {0.332947, 10}, {0.481947, 5}}
```

```
testerForUlikeL = (testModul[#1⟦1⟧, #1⟦2⟧] &) /@ testPar
```

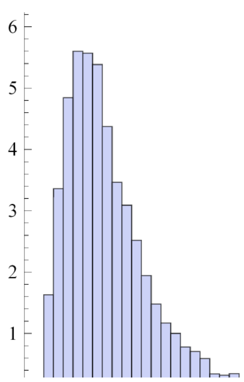
{{∞, 0.161947, 

, 0.199434, 0.191972, 0.0521767, 0.4357},

{15, 0.277947, 

, 0.198679, 0.186562, 0.0671367, 0.4148}, {10, 0.332947, 

, 0.198569, 0.183653, 0.075512, 0.4052}, {5, 0.481947,



, 0.197889, 0.17339, 0.101218, 0.3803}}

```
(Transpose[testerForUlikeL][[3]]) // Show
```