

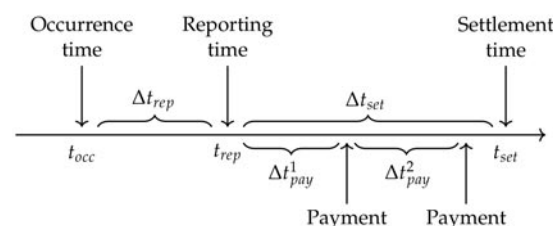
# Schadereserveringen op micro-niveau: Een volledig Bayesiaans model

Donderdag 28 november had ik de eer om voor mijn thesis: 'Micro-level Loss Reserving – A fully Bayesian Approach' de Johan de Witt scriptieprijs in ontvangst te nemen. Deze scriptieprijs wordt jaarlijks uitgereikt door het AG. Dit artikel geeft een beknopt overzicht van mijn scriptie en de Bayesiaanse implementatie van het micro-level model. De volledige scriptie is te downloaden op <http://bit.ly/34LvpMG>

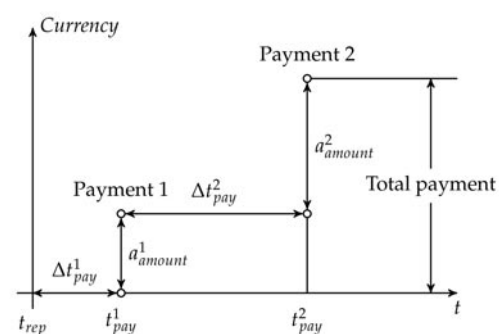
**Klassieke methoden voor het doen van schade-reserveringen maken gebruik van geaggregeerde data op schade- en afwikkeljaar. Met de komst van meer fijnmazige data; de huidige overvloed aan rekenkracht en een actieve gemeenschap voor de ontwikkeling van open-source tooling (PyMC3); komt de vraag of deze klassieke modellen hun actualiteit niet beginnen te verliezen. Ook willen verzekeraars steeds vaker externe variabelen (of covariaten) meenemen in het reserveringsproces, denk aan postcode, gezins-samenstelling of historisch claimgedrag. Daarom wordt de behoefte naar een modern, nauwkeuriger en fijnmaziger model steeds groter. Een micro-level model modelleert alle gebeurtenissen binnen het claim-proces van een polishouder afzonderlijk en afhankelijk van variabelen, waardoor er geen informatie in de data verloren gaat. Verder kan door de toepassing van Bayesiaanse statistiek alle model- en parameter-onzekerheid worden gekwantificeerd in elke component van het model, waarmee vervolgens een voorspelling (lees: reservering) te maken valt die alle informatie omtrent deze onzekerheid bevat.**

## HET CLAIMPROCES

Micro-level modellen maken gebruik van informatie van het claim proces. Het is daarom belangrijk om een goed beeld te hebben van dit proces. Dit proces is bewust zo simpel mogelijk gehouden, zodat het toepasbaar is binnen een breed aantal verzekeringstypes. Figuur 1 en 2 geven een beeld van alle gebeurtenissen in een claim proces. We definiëren hier globaal twee fasen. Allereerst is er een fase waarin de claim nog niet gemeld is aan de verzekeraar (gedurende  $\Delta t_{rep}$ ). Een claim in deze fase wordt aangeduid als *Incurred But Not Settled* (IBNR). Vervolgens is er een fase dat een claim wél gerapporteerd is maar nog niet afgesloten (in  $\Delta t_{set}$ ). Dit wordt aangeduid met *Reported But Not Settled* (RBNS).



**Figuur 1:** De ontwikkeling van een claimproces.  $t_{occ}$  is het moment waarop de gebeurtenis plaatsvindt die leidt tot de claim, de zogenaamde *occurrence time*. De zaak wordt aan de verzekeraar gemeld (reported) op  $t_{rep}$  en gesloten (settled) op  $t_{set}$ . Gedurende  $\Delta t_{set}$  worden (ingaaende en uitgaande) betalingen verricht.



**Figuur 2:** De betalingen binnen de claim gedurende de RBNS fase. Betalingen worden gedefinieerd door een (mogelijk negatieve) hoogte ( $a_{amount}$ ) en de tijd vanaf de vorige gebeurtenis tot de betaling ( $\Delta t_{pay}$ ).

Met een gedefinieerd claimproces, kan de claim stochastisch worden gemodelleerd. De momenten van het ontstaan van de schade  $t_{occ}$  kan worden gemodelleerd d.m.v. een *Poisson* distributie (als in Norberg (1993) en Arjas (1989)), waarin de tijd tussen de schades kan worden gemodelleerd door een *Exponentiele* distributie. Voorts kunnen  $\Delta t_{rep}$ ,  $\Delta t_{pay}$ ,  $\Delta t_{set}$  en  $a_{amount}$  (zie Figuur 1 en 2) stochastisch worden gemodelleerd, waarbij de tijd tussen de betalingen wordt gemodelleerd als een *Renewal Process*, als gedefinieerd by Cook and Lawless (2007). De volledige versie van mijn scriptie bevat een uitgebreide argumentatie voor de keuze van de distributies voor ieder van deze componenten. Ook wordt hier een methode besproken om extra informatie (covariaten) toe te voegen aan het stochastisch model.

## EEN BAYESIAANS MODEL

Standaard Maximum Likelihood Estimation (MLE) methoden zijn reeds toegepast op micro-level loss reserving (zie bijvoorbeeld Antonio en Plat, 2014). Een nadeel van deze methoden is dat parameter en model onzekerheid niet (of slechts gedeeltelijk) zijn meegenomen. Dit zorgt voor onduidelijkheid bij zowel de verzekeraar als de toezichthouder over hoe nauwkeurig de voorspellingen überhaupt zijn. Bayesiaanse modellen aan de andere kant, kunnen alle onzekerheid volledig incorporeren. Hiernaast kan aan een Bayesiaans model een zogenaamde prior worden toegevoegd. Op deze manier kan de actuaaris zijn kennis op een "natuurlijke" manier aan het model toevoegen. Arjas (1989) schreef al: "*Choosing a reasonable prior ... could be viewed as a good opportunity for an actuary to use, in a quantitative fashion, his experience and best hunches.*" Verder kan een goed gekozen prior erg effectief zijn wanneer er weinig data beschikbaar is (de zojuist gestartte verzekeraar) of wanneer de data van een onvoldoende kwaliteit is. Wanneer een Bayesiaans model wordt geschat, spreken we van *Bayesian Inference*. Dit proces levert niet slechts één puntschatting, maar een volledige distributie voor elke parameter van het model (als het model erg slecht is of de data schaars, zullen deze distributies dus erg breed zijn). Als men vervolgens voorspellingen met het Bayesiaanse model doet (*Posterior Predictive Sampling*) worden eerst samples van de parameters genomen, waarna deze weer worden gebruikt om samples te krijgen van het model.

## CORRECTIE VOOR DE NIET-GEOBSERVEERDE IBNR CLAIMS

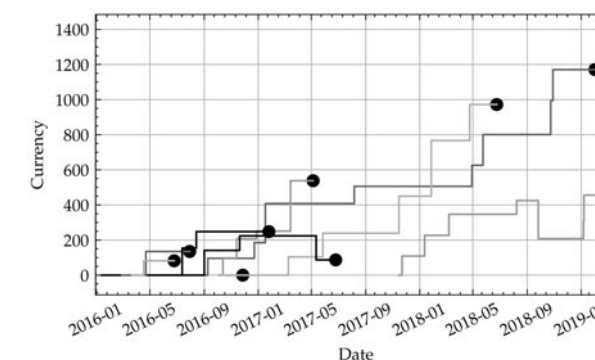
Niet alle schadementen ( $t_{occ}$ ) zijn bekend bij de verzekeraar, dit geldt specifiek voor de *IBNR* claims. Om de intensiteit van het Poisson Process dus goed te kunnen schatten, moet de hoeveelheid claims worden aangepast d.m.v. de distributie van  $\Delta t_{rep}$ . In een volledig Bayesiaans model gebeurt dit door samples van de distributie van  $\Delta t_{rep}$  te nemen, te integreren en hiermee de intensiteit aan te passen. Figuur 3 laat de intensiteit (lees: de snelheid waarmee claims verschijnen) van het Poisson proces voor de *IBNR* claims zien. We kunnen uit dit figuur twee conclusies trekken: (1) er zijn meer *IBNR* claims aan het einde van de dataset en (2) we zien dat het model onzekerder wordt zodra we dichterbij het einde van de dataset komen (hier 2018-12). Beide observaties liggen in lijn met onze verwachtingen en worden bevestigd door het model.

## CORRECTIE VOOR HET CENSURING EFFECT

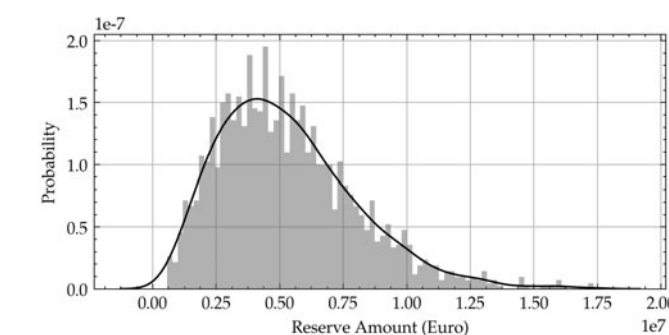
Een karakteristieke eigenschap van analyses waarbij een tijdsduur gemeten wordt, is wanneer bepaalde gebeurtenissen (nog) niet geobserveerd zijn wanneer de studie is afgelopen. Dit zorgt voor incomplete informatie (*censoring*), waar rekening mee moet worden gehouden in het model. Dit is specifiek in het geval bij het modelleren van  $\Delta t_{set}$ . Aan het einde van de dataset zullen er immers nog veel claims zijn die wel geopend zijn, maar (nog) niet gesloten zijn (*IBNR*). Als vervolgens alleen de gesloten  $\Delta t_{set}$  meegenomen worden in het model, zal het model deze tijd over het algemeen *onderschatten*. Het model is aangepast zodat deze effecten meegenomen zijn.

## RESULTATEN

Om te resultaten te vergaren uit het model, gebruiken we de zogenaamde 'Posterior Predictive' distributie. Deze trekt éerst samples uit de verdeling uit de parameters van het model. Hierna worden samples uit het model getrokken, gegeven deze parameters. Voor de huidige simulatie worden 1000 paden getrokken voor elke *IBNR* en *RBNS* claim. Figuur 3 laat een aantal verschillende paden zien. Figuur 4 aggregeert de paden uit de simulatie en zorgt op deze manier voor samples uit de reservering.



**Figuur 3:** Een aantal gesimuleerde paden voor *IBNR* zaken waarbij  $t_{occ}$  in het jaar 2016 ligt. De verticale as geeft het geldbedrag weer, de horizontale as de datum. De lijnen representeren de ontwikkeling van de claim. De sluiting van de claim wordt gerepresenteerd door een zwarte stip.



**Figuur 4:** Een histogram van de samples van de *RBNS* reservering. Op de y-as wordt de waarschijnlijkheid weergegeven, op de x-as de hoogte van de reservering in Euro.

## IMPLEMENTATIE IN PYTHON

Bayesiaanse modellen zijn berucht om het feit dat ze vaak veel reken capaciteit vragen of ingewikkeld te programmeren zijn. Dit was zeker in het verleden vaak een probleem. Echter, recente ontwikkelingen in open-source pakketten zoals *PyMC3* voor Python, maken Bayesiaanse analyses implementeerbaar en meer toepasbaar dan ooit tevoren. Zo maakt *PyMC3* het bijvoorbeeld mogelijk om meerdere processor-kernen of zelfs de grafische kaart te gebruiken. Verder heeft de programmeertaal Python de laatste jaren bijoorlijk aan invloed gewonnen ten opzichte van R, wat vooral merkbaar is in het aantal en de kwaliteit van de software-pakketten (de zogenaamde *packages*) die de laatste jaren zijn verschenen. Verder is Python code veel beter te onderhouden en is robuster dan R, mits gestructureerd en modulair geschreven. Ook kan binnen Python uitgebreid worden getest (d.m.v. unittests) en is het vaak eenvoudiger om code in productie te zetten (d.m.v. Docker, CI/CD pipelines). Het ziet er dus naar uit dat de toekomst voor Python is. ■

