Modèle PGG en Python
IMPERIO ASSURANCES

Joël Da Costa Oliveira,
Robin Wengi,
Frédéric Tauxe

7 janvier 2020

Table des matières

1 Introduction

L'objectif de cette documentation est d'expliquer le fonctionnement du nouveau modèle de calcul de la provision global de gestion. Le nouveau modèle est alors basé sur l'ancien qui se trouvait sur le logiciel Prophet, nous avons alors migrer la totalité du modèle sur Python. Premièrement nous allons répliquer les résultats obtenu de notre ancien modèle. Ensuite une fois que la totalité des résultats sont identiques au modèle Prophet, nous validons ces calculs et nous procédons aux changements des incohérences que nous avions remarqué durant la réplication de l'ancien modèle.

Nous avons 5 différents scénarios r où l'on va stresser différentes hypothèses du modèle :

- Best Estimate
- Rendement et longévité
- Annulation plus
- Annulation moins
- Best Estimate plus marge
- Biométrie et frais

Les calculs des projection de cash flows seront effectué sur des vecteurs composés de trois différentes dimensions : la police i du portefeuille pour la projection au temps t du scénario r.

2 Méthodologie

Afin d'obtenir les même résultats du modèle Prophet, nous avons commencé par modélisé un des produits que nous estimions le plus facilement calculable. Nous effectuons la totalité des calculs pour une police, si le résultat de cette police est identique à Prophet, nous calculons pour la totalité de notre portefeuille pour le produit en question. S'il s'avère que le résultat total n'est pas égal à Prophet, nous testions alors une autre police afin de comprendre d'où pourrait venir l'écart. Une fois que les résultats de la totalité des polices de notre portefeuille pour un produit répliquent parfaitement les résultats de tout les scénarios Prophet, nous validions les résultats du nouveau modèle et passions au produit suivant.

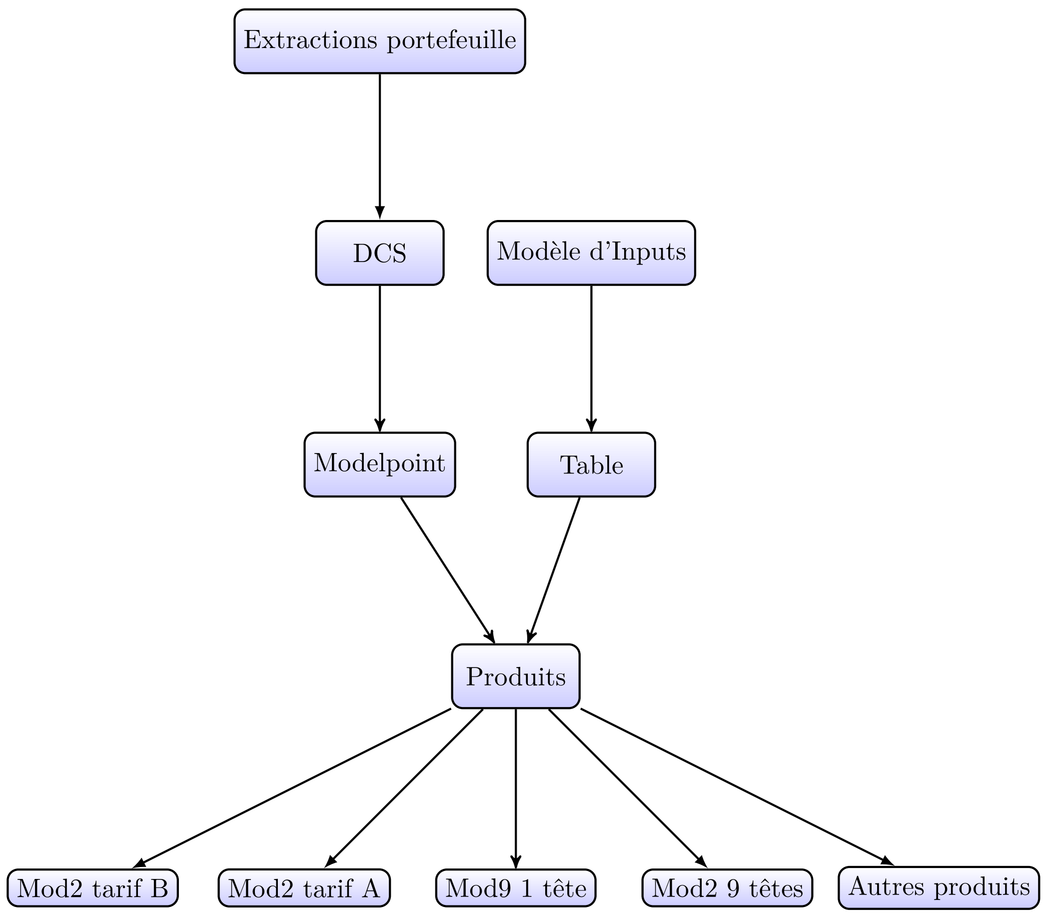
Le résultat servant à calculer la provision globale de gestion est la variable "Best Estimated Liabilities" (BEL). Nous commencions par calculer les variables des composantes du BEL afin de remonter le calcul jusqu'à obtenir le BEL. Cette variable est composée de la totalité des variables calculées dans le modèle, nous utilisons alors le BEL afin de comparer nos résultats avec ceux de Prophet.

3 Structure du modèle

La structure du nouveau modèle sur Python est semblable à celui de Prophet. Il y a quelques différences au niveau de la granularité ainsi que la façon de traité les différents tarifs et nombre d'assurés pour un produit donné.

3.1 Structure Prophet

Voici l'ancienne structure utilisée par Prophet pour calculer la provision globale de gestion et donc le BEL pour la totalité de notre portefeuille :

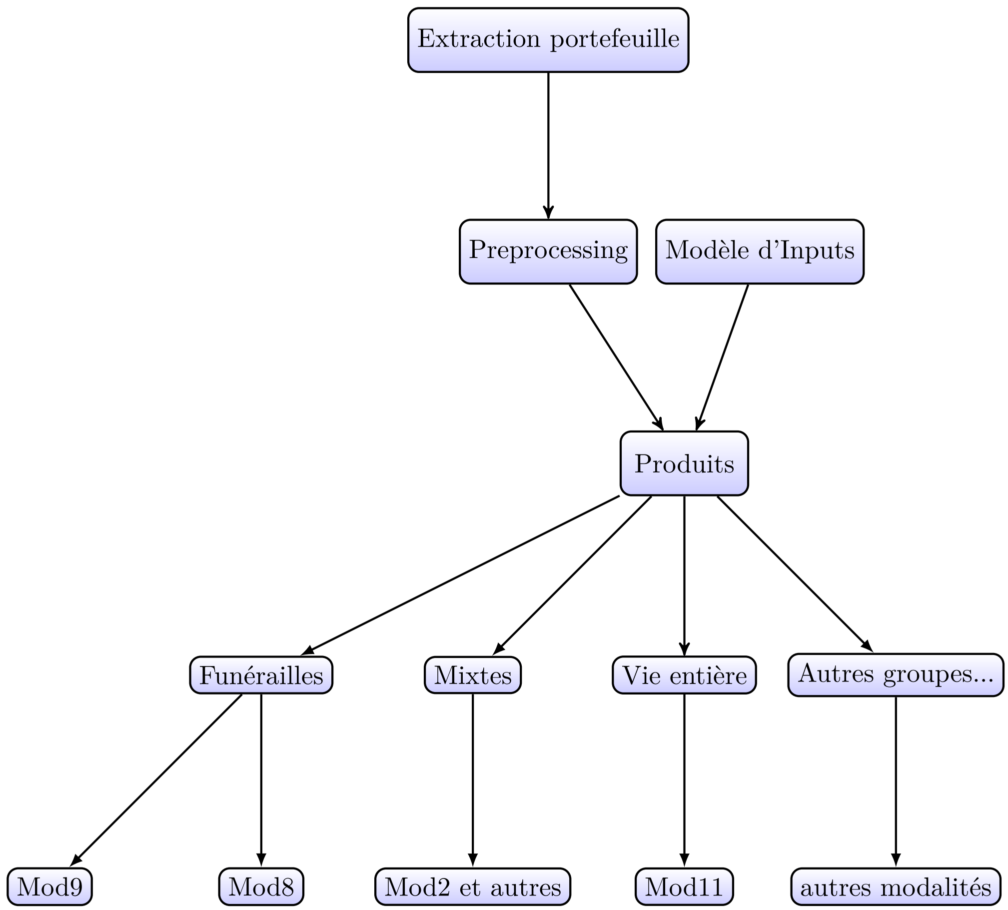


L'extraction de l'entier de notre portefeuille est formatté à travers les codes DCS qui permettent ensuite de créer plusieurs Modelpoints par produit. À l'aide des Modelpoint ainsi que des Tables Prophet qui sont créer directement à partir des modèles d'Inputs où se trouvent notamment les hypothèses, Prophet procède aux calculs du BEL pour tout les types de produits pour tout les scénarios.

La granularité de Prophet est plus élevée que celle que nous utilisons pour le nouveau modèle. Il existe en effet de nombreux types de produits (La modalité 2 est découpée en un sous-produit par Tarif, la modalité 9 possède deux sous-produits : Mod9 1 tête et Mod9 2 têtes). Ce qui nous fait un total de 50 produits Prophet. Ensuite la totalité des variables sont calculée pour chaque produit Prophet.

3.2 Structure Python

Voici la structure utilisée dans le nouveau modèle basé sur Python :



Dans le nouveau modèle nous avons un processus nommée "Preprocessing", celui-ci nous sert à extraire les données de notre portefeuille, formater/calculer les données extraites afin d'obtenir des données pouvant être traitées par la suite. Ce processus effectue également le chargement des hypothèses.

Ensuite il existe une super classe "Hypo" (Structure : Paramètres) qui va dimensionner les hypothèses utilisées en fonction du portefeuille extrait. La dimension des scénarios est connue et reste fixe (cinq scénarios), la dimension des polices ainsi que la dimension temporelle est connue grace au Preprocessing (nombre de police et nombre de mois projetés). La sous classe "Portfolio" (Structure : Produits) contient les variables qui se calculent de manières identiques à tout les produits sous-jacent (exemple : age, qx, etc...). Ensuite nous avons une classe par type de produit où l'on procède au calcul des variables qui sont identiques à toutes les modalités comprises dans ce type de produit. Le calcul des variables spécifiques à chaque produits se calculeront dans une sous-classe avec le nom de cette modalité.

Nous pouvons constater que la structure de notre modèle est en quelques points identiques au modèle Prophet. Les DCS correspondent au Preprocessing, les Tables aux Paramètres et les Produits Prophet sont semblable aux produits du nouveau modèle, la différence vient du fait que nous avons tout de même certaines variables qui sont calculées aux niveau ”Produits”, contrairement à Prophet qui effectue l'entier de ces calculs à l'echelle de la modalité. Sans oublier le fait que Prophet possède une granularité plus élevée en traitant séparément les polices avec un assurés des polices avec deux assurés, tandis que dans le nouveau modèle proposé, ceux-ci sont traités dans le même niveau.

4 Hypothèses

Les inputs utilisés pour le nouveau model sont resté inchangé.
Les différents calculs et explications des hypothèses sont expliqués dans la documentation (voir doc ANNEXE ??)

Taux d'annulation : [Documentation lapse.docx]

Hypothèse de rendement : [Hypothèses RendementsNEW.docx]

Sinistralité : [Hypothèses Sinistralité.docx]

5 Calcul des variables utilisées

Les diverses variables utilisée pour calculer le "Best estimated liabilities" varient en fonction du produit.

En effet, le calcul des probabilités que la police soit toujours en vigueur va dépendre si il y a possibilité de réduction pour le produit en question. La sinistralité va également dépendre si celle-ci est calculée avec un taux de sinistre sur primes ou alors simplement avec les probabilités de décès (pour les assurances temporaires décès). En ce qui concerne les reserves, celles-ci sont spécifiques à chaque produits, ces valeurs seront alors calculées différemment pour chacun d'entre eux. Les coûts varieront si le produit possède des reserves et si celui-ci est sujet à des primes en cours.

5.1 Inforce probability

Il faut noté d'abord que Les composantes des probabilités inforce pour la polices i du scénario r sont les suivantes :

— $nbMaturities_t$ nombre de polices arrivant à maturité
— $nbPolIfSM_t$ nombre de polices en début de mois
— $nbrDeath_t$ nombre de morts
— $nbSurrender_t$ nombre d'annulation
— $nbrPolIf_t$ nombre de polices en vigueur

Tout d'abord nous devons connaître la probabilité de décès mensuel et la probabilité de décès jointe mensuel si la police en question concerne deux assurés.

$$\begin{aligned} qx^m &= 1 - (1 - qx)^{1/12}, \\ qxy &= qx + qy - qxqy, \\ qxy^m &= 1 - (1 - qxy)^{1/12} \end{aligned} \tag{1}$$

Nous aurons également besoin de connaître w_t^m, la probabilité d'annulation mensuel au temps t, qui vient des hypothèses (w_t étant la probabilité d'annulation annuel)

$$w_t^m = 1 - (1 - w_t)^{1/12}. \tag{2}$$

Le calcul des probabilités inforce est un calcul recursif dont la totalité des variables composant les inforces sont toutes interdépendantes. L'ensemble des variables doit être connu afin de procéder au calcul des inforce. Nous forçons uniquement la variable $nbrPolIf_t = 0$ pour $t = 0$, ensuite la recursion peut avoir lieu.
Premièrement nous voulons connaître le nombre de maturités. Le nombre de maturité pour une police donnée est égal au nombre de polices inforce au terme de la police. Nous avons $polTermM$ le nombre de mois que va durer la police i jusqu'à son échéance, nous avons une valeur uniquement au temps $t = polTermM + 1$:

$$nbrMaturities_{polTermM+1} = nbrPolIf_{polTermM} \tag{3}$$

et pour tout autre t nous aurons $nbMaturities = 0$.

Nous cherchons ensuite le nombre de polices inforce en début de mois $nbrPolIfSM_t$

$$nbrPolIfSM_t = nbrPolIf_{t-1} - nbrMaturities_t \tag{4}$$

Le nombre de mort ainsi que le nombre d'annulation est calculé de la manière suivante :

$$nbrDeath_t = nbrPolIfSM_t \cdot qxy_t \cdot (1 - (w_t^m \cdot (1 - lapseTiming))) \tag{5}$$

$$nbrSurrender_t = nbrPolIfSM_t \cdot w_t^m \cdot (1 - (qxy_t \cdot lapseTiming)) \tag{6}$$

avec $lapseTiming = 1$.

Nous pouvons alors trouver de manière récursive $nbrPolIf_t$, la probabilité inforce :

$$nbrPolIf_t = nbrPolIf_{t-1} - nbrMaturities_t - nbrDeath_t - nbrSurrender_t \tag{7}$$

5.2 Primes funérailles

Le calcul des primes inforce est le suivant :

$$totalPremium_t = prem_t * nbrPolIfSM_t \tag{8}$$

Les montant des primes ainsi que son occurence va dépendre du fractionnement de la police. Si le fractionnement est annuel, nous aurons une prime par année, soit $prem_t = 0$ pour $\frac{11}{12}$ du temps t.

5.3 Sinistralité décès funérailles

Pour connaître la sinistralité principale $deathClaim_t$ pour les modalités 8 et 9, il suffit de multiplier le nombre de décès au temps t par le capital décès $capital_i$ (celui-ci est le même pour tout temps, mais varie si c'est une modalité 8 ou 9).

$$deathClaim_t = nbrDeath_t \cdot capital_i \tag{9}$$

5.4 Sinistralité complémentaire frais de visite funérailles

Pour cette sinistralité nous appliquons le principe du taux de sinistre $claimRate$ sur prime complémentaire $primeCompl$ (qui sont le même pour tout t).

$$fraisVisiteClaim_t = claimRate \cdot primeCompl \tag{10}$$

pour tout t où il y a un paiement de prime, autrement $fraisVisiteClaim = 0$. Le total des sinistralités $totaClaim_t$ sera donc l'addition de $fraisVisiteClaim_t + deathClaim_t$.

5.5 Coût totaux

Le total des coûts est égal à l'addition du coût par police inforce venant des hypothèses et le coût inforce des gestions de placements qui sont appliqués sur les reserves ainsi que les risques en cours.

$$totalExpense_t = unitExpense_t + reserveExpense_t \tag{11}$$

5.5.1 Coût de gestion de placement

Les coûts de gestion de placement est un taux appliqué sur les reserves d'un produit ainsi que sur les risques en cours de celui-ci, cette addition nous donne la variable $adjustedReserve_t$ (inforce en début de mois). Avec le taux $fraisGestionPlacement$ nous avons

$$reserveExpense_t = adjustedReserve_t \cdot fraisGestionPlacement \cdot nbrPolIfSM_t \tag{12}$$

Le calcul des reserves ajustée vont dépendre du produit, $adjustedReserve_t$ est calculé de la manière suivante :

$$adjustedReserve_t = risqueEnCour_t + reservesForExp_t \tag{13}$$

S'il n'y a pas de reserves constituées pour ce produit, $reservesForExp_t = 0$, alors les reserves ajustées seront égales au risque en cours

5.5.2 Coût par police

Le coût unitaire par police, $coutParPolice$ pour tout t, vient des hypothèses, c'est un coût mensuel appliqué en début de mois. Ce coût est aussi impacté par l'inflation des coûts, $inflation_t$, variable calculée à l'aide d'un paramètre provenant également des hypothèses. Nous avons

$$unitExpense_t = coutParPolice \cdot inflation_t \cdot nbrPolIfSM_t \tag{14}$$

avec le taux d'inflation $txInfl$, nous pouvons calculer mensuellement $inflation_t$

$$inflation_t = (1 + txInfl)^{t/12} \tag{15}$$

5.6 Commissions

Les commissions sont calculées selon les taux de commissionnement venant des hypothèses (voir section SECTION A AJOUTER). Les commissions inforce sera égal au primes inforce multiplié par le taux de commissionnement

$$inflation_t = (1 + txInfl)^{t/12} \tag{16}$$

5.7 Nombre de rachat

Le calcul du nombre de rachat sera différent si un produit permet la réduction ou non.

5.7.1 Nombre de rachat : Produits sans réduction possible

Nous avons la probabilité de décès mensuel qx^m qui est défini par

$$qx^m = 1 - (1 - qx)^{1/12} \tag{17}$$

avec IF_t^m la probabilité mensuelle au temps t que la police soit toujours en vigueur (In Force) au temps $t + 1$ (au mois prochain), w_t la probabilité au temps t qu'une police soit annulée l'année d'après ainsi que w_t^m la probabilité mensuelle au temps t que la police soit annulée au temps $t + 1$ (au mois prochain).

w_t vien des hypothèses de rachat. Nous trouvons la probabilité d'annulation mensuel

$$w_t^m = 1 - (1 - w_t)^{1/12} \tag{18}$$

nous avons ensuite le nombre d'annulation $Surr_t$ au temps t

$$Surr_t = IF_{t-1} w_t^m (1 - \frac{qx^m}{2}) \tag{19}$$

en sachant que $IF_{t-1} = 1$ pour $t = 1$, ensuite pour calculer $Surr_t$ avec $t > 1$ il faudra connaître le nombre de police en vigueur au temps t (voir section 5.7.1, page 14 pour le calcul du nombre de police en vigueur au temps t)

5.7.2 Nombre de rachat : Produits avec réduction possible

Lorsque la réduction est possible nous aurons différents états possible :
- Police réduite
- Police en vigueur
- Police annulée

Nous devons donc connaître :
- $IFnotRed_t$ Le nombre de police en vigueur et non-réduite
- $Rnew_t$ Le nombre de nouvelles réductions pour une police en vigueur
- $RedTot_t$ Le nombre total de polices réduites
- $Surr_t$ Le nombre d'annulation pour une police en vigueur non-réduite
- $SurrRed_t$ Le nombre d'annulation pour une police réduite
- $Death_t$ Le nombre de décès pour une police en vigueur non-réduite
- $DeathRed_t$ Le nombre de décès pour une police réduite

Nous avons les variables qx, w_t, r_t venant des hypothèses, avec r_t la probabilité qu'une police en vigueur soit réduite l'année d'après et $frac$ étant le fractionnement de la police.

Nous trouvons d'abord $death_t^m$ le nombre de décès mensuel au temps t.

$$death_t^m = qx^m IFnotRed_{t-1}(1 - \frac{w_t^m}{2}) \tag{20}$$

pour tout $t > 0$ en sachant que $IFnotRed_t = 1$ au temps $t = 0$

Sachant r_t les probabilités de réduction venant des hypothèses, nous trouvons

$$r_t^m = 1 - (1 - r_t)^{1/12} \tag{21}$$

la probabilité de réduction mensuel.

Pour la suite des calculs il faut savoir que certaines variables sont déjà connues au temps $t = 1$:

$$IFnotRed_0 = \begin{cases} 1 & \text{si la police n'est pas réduite} \\ 0 & \text{si la police est déjà réduite} \end{cases} \tag{22}$$

$$IFred_0 = \begin{cases} 0 & \text{si la police n'est pas réduite} \\ 1 & \text{si la police est déjà réduite} \end{cases} \tag{23}$$

Nous avons besoin ensuite de connaître le nombre de nouvelles réductions $Rnew_t$. Nous connaissons $IFnotRed_{t-1}$ pour $t = 1$ et donc nous avons

$$Rnew_1 = r_1^m(IFnotRed_0 - Death_1 - Surr_1) \tag{24}$$

nous pouvons ensuite calculer $IFnotRed_1$

$$IFnotRed_1 = IFnotRed_0 - Death_1 - Surr_1 - Rnew_1 \tag{25}$$

ainsi par récursivité nous avons pour tout t

$$IFnotRed_t = IFnotRed_{t-1} - Death_t - Surr_t - Rnew_t \tag{26}$$

Nous connaissons désormais le nombre de police en vigueur non-réduite ainsi que le nombre de nouvelles réductions, donc nous pouvons à présent calculer le nombre d'annulation pour les polices non-réduite $SurrNotRed_t$

$$SurrNotRed_t = IFnotRed_{t-1}r_t^m(1 - q_{x+t}^m) \tag{27}$$

Afin de connaître le nombre total d'annulation, il nous manque à calculer le nombre d'annulation pour des polices qui sont déjà réduites $SurrRed_t$. Pour cela il nous faut connaître le nombre de polices réduites qui va dépendre du nombre de décès pour des polices déjà réduite $DeathRed_t$

$$DeathRed_t = IFred_{t-1} \cdot q_{x+t}^m \cdot (1 - \frac{w_t^m}{2}) \tag{28}$$

nous trouvons alors

$$SurrRed_t = IFred_{t-1} * r_t^m(1 - q_{x+t}^m) \tag{29}$$

Pour calculer $IFred$ il faut à nouveau procéder à un calcul récursif en s'aidant de la formule (22)

$$IFred_t = IFred_{t-1} + Rnew_t - DeathRed_t - SurrRed_t \tag{30}$$

5.8 Inforce probability

Calcul des inforce probability. Il existe deux façon de calculer les inforce probability qui va dépendre du produits. Il y a donc les produits sans possibilité de réduction et les produits avec possibilité de réduction.

5.8.1 Inforce pour les produits sans possibilité de réduction

Nous savons que $IF_{t-1} = 1$ pour $t = 1$ et donc nous pouvons calculer IF_{t-1} pour tout $t > 1$

$$IF_t = IF_{t-1} - death_t - Surr_t \tag{31}$$

5.8.2 Inforce pour les produits avec possibilité de réduction

Ici on insère le calcul

```
class Portfolio:


    def __init__(self,po=p,runs=[0,1,2,3,4,5], \
              LapseNew=True,RateNew=True,SinistralityNew=True,CommissionNew=True,
CostNew=True):
        self.tout=po
        self.p=po
        self.runs=runs
        self.un=self.one()
        self.zero=self.zeros()
        self.vide=self.vides()
        self.template= self.templateProjection()
        self.shape=list(self.un.shape)


```

Exemple insertion code python

Exemple d'ajout de calcul Latex :

La fonction f est définie par

$$f(x) = x - 1 \tag{32}$$

On a alors

$$f(x) = 0 \iff x = 1 \tag{33}$$

.

A Annexes

qx : probabilité de décès annuel pour une personne agée de x années
w_t : probabilité qu'une police en vigueur au temps t soit annulée au temps $t+1$
$frac$: fractionnement de la police
$deathR_t$: nombre de décès pour des polices déjà réduites