

Recherche de l'allocation optimale du nombre de lignes à allouer par classe d'agrégation - Partie 2

Cheick Oumar Kouyaté
Chargé d'études actuarielles chez Allianz France

Résumé

Dans cette étude, notre objectif est de déterminer l'allocation optimale, c'est-à-dire celle qui est associée à la plus petite erreur de compression. Notre approche consiste à : 1) Formuler cette allocation optimale comme la solution d'un programme d'optimisation ; 2) Nous effectuons ensuite une série de transformations, comprenant un changement de variables, un changement d'échelle et une relaxation continue, pour convertir le programme initial, de nature discrète, en un programme continu. Cette première transformation introduit un premier biais d'estimation ; 3) Étant donné que les fonctions composant la fonction objectif du programme continu ne sont pas complètement connues en pratique, nous les interpolons linéairement à partir d'un nombre défini de valeurs intermédiaires, noté p . Cette interpolation entraîne une erreur d'interpolation dépendant de p ; 4) Enfin, nous approximations les interpolations linéaires par des polynômes de Bernstein de degré n , transformant ainsi le programme continu en un programme continu et différentiable. Cette dernière étape induit une erreur d'approximation, dépendant de n .

En analysant les effets des hyperparamètres p et n sur les programmes d'optimisation, nous observons une convergence de la solution du programme continu et différentiable vers l'allocation optimale, l'augmentation de la valeur de p permettant une réduction de l'erreur d'interpolation et l'augmentation de la valeur de n permettant une réduction de l'erreur d'approximation. Ces résultats soulignent l'efficacité de notre approche et fournissent une méthode pratique pour la conception d'allocations optimales dans des contextes similaires.

Mots clés : Compression de données ; Allocation ; Optimisation différentiable ; Interpolation et Approximation

Abréviations : BEL : Best Estimate of Liabilities, ALM : Asset Liability Management, PM : Provision Mathématique

Cette seconde partie présente la mise en œuvre sur un cas pratique, de la démarche exposée dans la partie 1 de l'article publiée précédemment dans Variances.

4. Cas pratique

Pour éclairer les concepts abordés dans cette étude, nous allons utiliser comme illustration deux strates que nous désignerons *Strate 1* et *Strate 2*. Ces strates représentent concrètement différents types de produit d'assurance. Le tableau 2 présente le montant actualisé des engagements futurs (BEL), la Provision Mathématique (PM) ainsi que le nombre de contrats (N) par strate.

Table 2. Résumé des informations par strate

	N	PM	BEL
Strate 1	125	5 482 829	2 124 958
Strate 2	155	7 394 618	3 302 431
Portefeuille	280	12 877 447	5 427 388

Le portefeuille est constitué de 280 contrats. L'objectif consiste à réduire ce nombre de contrats en utilisant un algorithme de compression tel que le kMeans ou la Classification Ascendante Hiérarchique

(CAH), en regroupant les contrats qui partagent des similarités dans leurs caractéristiques, et ce, tout en minimisant l'impact sur le montant de BEL associé au portefeuille. Dans notre exemple, nous optons pour une réduction du portefeuille à 50 lignes, ce qui représente un taux de compression de 82.44%.

Etant donné que *Strate 1* contient 125 lignes et *Strate 2* contient 155 lignes, tout couple de valeurs entières avec le premier élément variant de 1 à 125 et le deuxième élément de 1 à 155, peut être considéré comme une allocation. Par exemple, (100, 35), (4, 10) ou (87, 133) sont des allocations. Comme nous avons décidé de compresser le portefeuille à 50 lignes, les allocations admissibles, potentiellement éligibles comme solution finale, sont les allocations dont la somme des deux éléments est égale à 50. Par exemple (22, 28) ou (7, 43) sont des allocations admissibles. Ainsi, il est évident que la solution finale pour *Strate 1* ne peut excéder 49, et il en va de même pour *Strate 2*. Nous allons donc tronquer les abscisses des graphiques dans cet intervalle pour plus de clarté.

4.1 Fonction d'erreur de compression

Représentons les fonctions h_g fournissant l'erreur de compression en fonction du nombre de lignes alloué à chacune des strates.

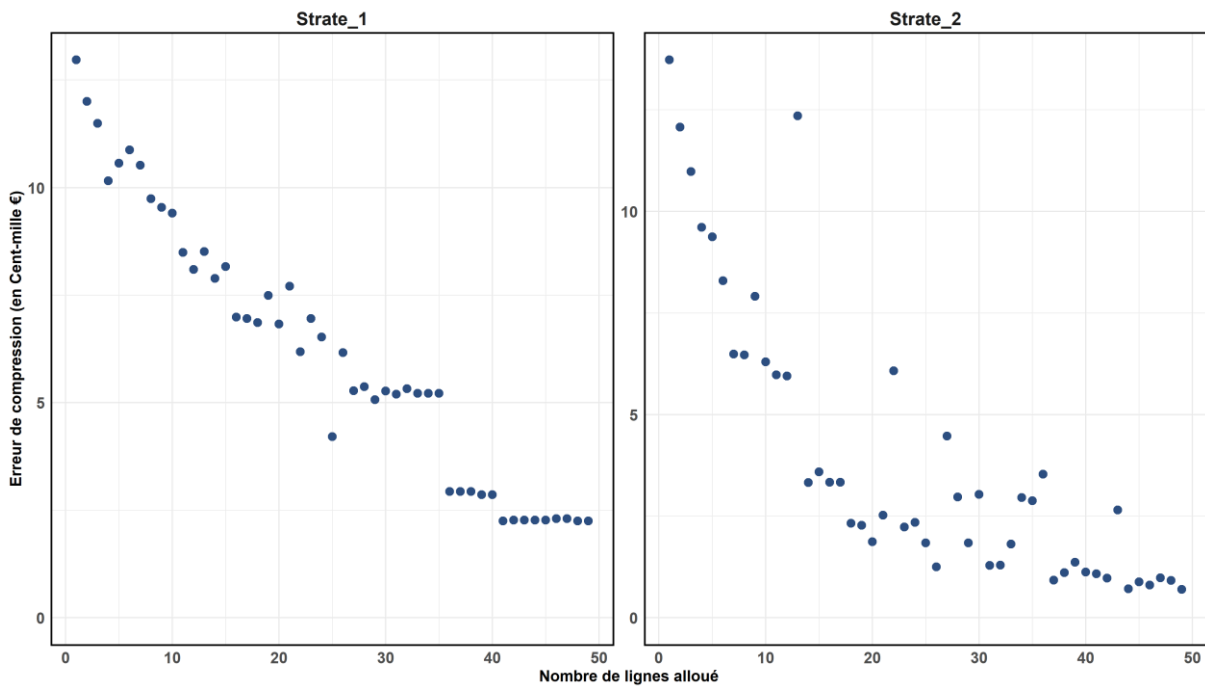


Figure 1. Erreur de compression selon le nombre de lignes alloué

A partir de ces deux graphiques, il est possible de déterminer l'erreur globale de compression pour toute allocation. Par exemple, pour une allocation de $k = (13, 27)$, l'erreur de compression se calcule de la façon suivante :

$$H(13, 27) = h_1(13) + h_2(27) = 851\,357 + 446\,972 = 1\,298\,329$$

4.2 Changement de variables, d'échelle et relaxation continue

Dans cette partie, nous illustrons graphiquement les implications des différentes transformations et leur importance dans notre approche.

4.2.1 Changement de variables

La Figure 2 montre l'impact du changement de variable sur les fonctions des erreurs de compression. Les erreurs de compression ne sont plus représentées en fonction du nombre de lignes allouée, mais en fonction du taux de compression auquel ce nombre alloué correspond.

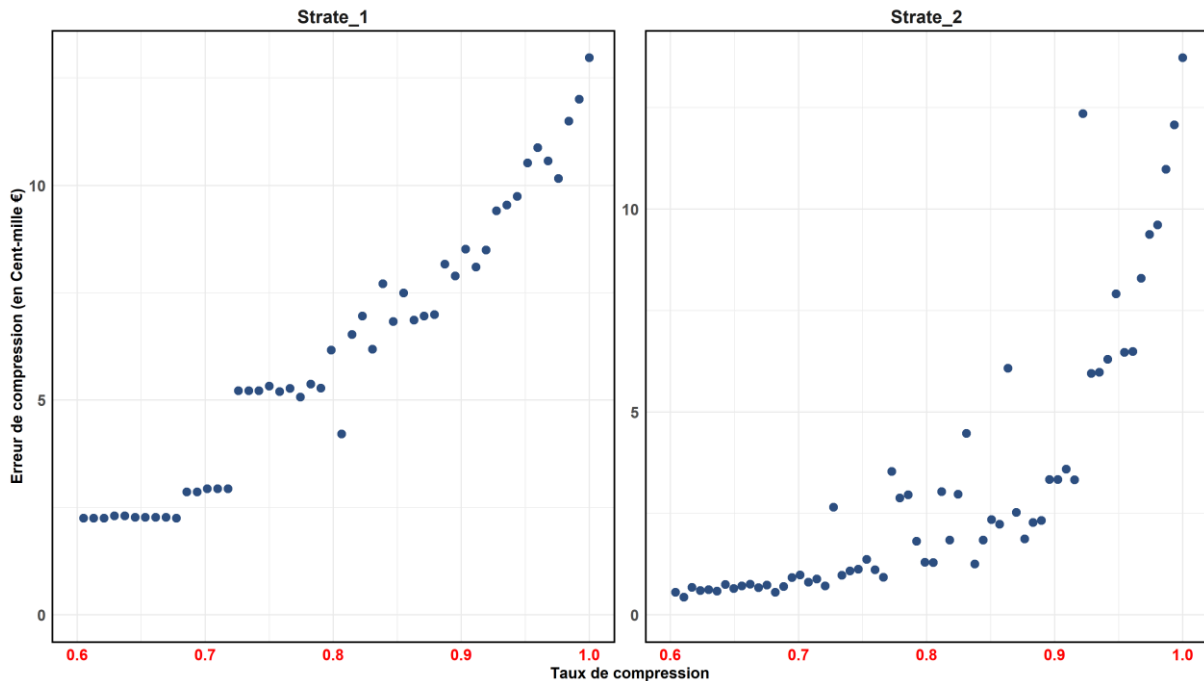


Figure 2. Impact du changement de variables sur les fonctions h_g

Pour reprendre l'exemple de notre allocation $k = (13, 27)$, on sait que cela correspond au couple de taux $t = (0.90, 0.83)$. Ainsi l'erreur de compression peut se calculer également de la façon suivante :

$$H(13, 27) = h_1(125 + 0.90 * (1 - 125)) + h_2(155 + 0.83 * (1 - 155))$$

4.2.2 Changement d'échelle

La Figure 3 montre l'impact du changement d'échelle sur les fonctions des erreurs de compression. Les erreurs de compression ne sont plus représentées sur l'axe des ordonnées en valeur monétaire, mais elles sont rapportées à l'erreur de compression correspondant à la compression maximale que nous avons notée err_g pour une strate g .

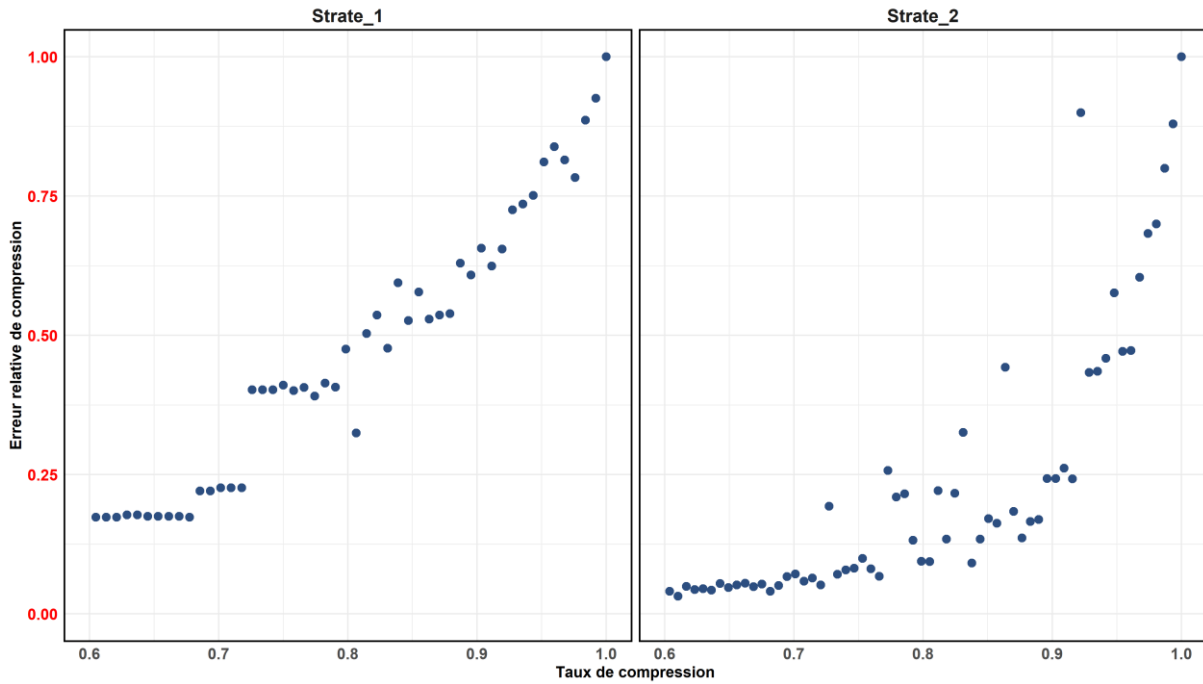


Figure 3. Impact du changement d'échelle sur les fonctions h_g

Cette transformation demande de connaître les erreurs de la compression maximale. Dans notre cas, $err_1 = 1\ 296\ 853$ et $err_2 = 1\ 372\ 977$. Ainsi :

$$H(13,27) = 1\ 296\ 853 * \frac{h_1(125 + 0.90 * (1 - 125))}{1\ 296\ 853} + 1\ 372\ 977 * \frac{h_2(155 + 0.83 * (1 - 155))}{1\ 372\ 977}$$

Ou encore

$$H(13,27) = 1\ 296\ 853 * f_1(0.90) + 1\ 372\ 977 * f_2(0.83) = F(0.90, 0.83) \quad (4)$$

Les fonctions f présentes dans l'équation (4) sont encore les versions discrètes de ces fonctions. Le but du point suivant est justement d'obtenir les versions continues de ces fonctions.

4.2.3 Relaxation continue

La Figure 4 montre l'impact de la relaxation continue sur les fonctions des erreurs de compression. Les taux de compression ne sont plus considérés dans un espace discret, mais sont pris dans l'ensemble continu $[0,1]$ tout entier.

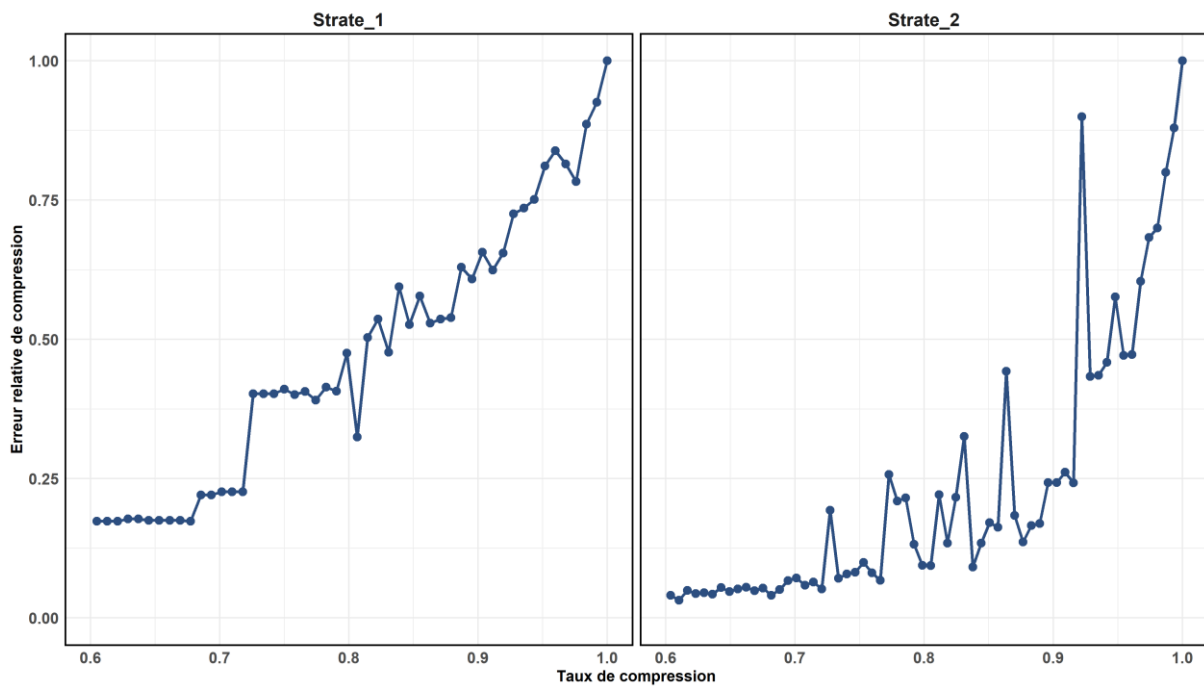


Figure 4. Impact de la relaxation continue sur les fonctions h_g

Ces trois transformations permettent de passer des fonctions h_g aux fonctions f_g . A la suite de la relaxation continue, les fonctions f présentes dans l'équation (4) sont dorénavant les versions continues de ces fonctions.

4.3 Résultats et discussions

L'objectif de cette section est de calculer les différentes allocations présentées dans cet article et de mettre en lumière les phénomènes de convergence abordés à travers les hyperparamètres. Dans un premier temps, les approximations par les polynômes de Bernstein seront effectuées avec une valeur de n fixée à 150. Ensuite, nous démontrerons la convergence des polynômes vers les fonctions sous-jacentes à mesure que n augmente. Pour ce qui est du nombre de valeurs intermédiaires p , nous effectuerons successivement des estimations avec p égal à 5, 10 et 15.

4.3.1 L'allocation optimale

L'allocation optimale, solution du programme discret (*), est celle qui minimise l'erreur de compression. Dans notre cas, avec seulement deux strates, il est relativement simple de déterminer manuellement cette allocation. Il suffit de dresser la liste exhaustive de toutes les allocations admissibles, d'associer à chacune son erreur de compression, puis de classer cette liste par ordre croissant d'erreur. L'allocation optimale correspond alors à la première allocation de cette liste.

	Strate 1	Strate 2	Ecart de BEL	Strate 1	Strate 2	Ecart de BEL	Strate 1	Strate 2	Ecart de BEL		
1	25	25	605 165	21	40	10	915 703	41	3	47	1 247 246
2	36	14	626 612	22	12	38	920 993	42	2	48	1 292 224
3	30	20	713 711	23	13	37	943 653	43	7	43	1 317 607
4	31	19	747 520	24	21	29	955 028	44	1	49	1 366 416
5	27	23	751 034	25	11	39	985 627	45	48	2	1 433 179
6	29	21	759 722	26	20	30	986 641	46	37	13	1 529 048
7	32	18	765 137	27	16	34	994 655	47	49	1	1 598 072
8	24	26	777 814	28	41	9	1 015 629	48	46	4	2 605 721
9	18	32	815 614	29	10	40	1 053 644	49	47	3	2 801 822
10	26	24	851 197	30	44	6	1 056 876				
11	34	16	854 987	31	9	41	1 062 541				
12	33	17	855 179	32	8	42	1 071 803				
13	42	8	873 844	33	4	46	1 096 132				
14	43	7	875 951	34	15	35	1 104 203				
15	17	33	877 041	35	23	27	1 142 281				
16	19	31	877 782	36	14	36	1 142 468				
17	35	15	881 224	37	28	22	1 145 101				
18	39	11	884 070	38	5	45	1 145 173				
19	38	12	888 378	39	6	44	1 159 198				
20	22	28	915 454	40	45	5	1 165 006				

Figure 5. Liste exhaustive des allocations admissibles classée par ordre croissant d'erreur de compression

Dans cette liste on peut voir que l'allocation associée à la plus petite erreur de compression est l'allocation (25, 25). L'erreur de compression de 605 165 observée au niveau du portefeuille est le meilleur résultat théoriquement possible. Cette erreur est uniquement imputable à l'algorithme utilisé pour compresser les données. Elle peut être éventuellement améliorée, soit en modifiant les autres hyperparamètres de l'algorithme comme la métrique utilisée ou la méthode d'initialisation des centroïdes par exemple, s'il s'agit d'un kMeans, soit en changeant d'algorithme.

4.3.2 L'allocation proportionnelle

En l'absence de l'application de notre méthode, l'allocation utilisée serait déterminée par la proratisation par rapport à la Provision Mathématique (PM) de chaque strate. Elle sera utilisée pour initialiser nos différents programmes d'optimisation. Le tableau 3 présente les résultats de cette proratisation, ainsi que les erreurs de compression associées à cette allocation.

Table 3. Erreur de compression de l'allocation initiale

	PM	Allocation	Ecart de BEL
Strate 1	5 482 829	21	770 796
Strate 2	7 394 618	29	184 232
Portefeuille	12 877 447	50	955 028

On peut remarquer que cette allocation est située au 24^{ème} rang dans la liste des allocations ordonnée par efficacité.

4.3.3 L'allocation optimale réalisable

Dans cette partie, les bases d'information étant complètes, les fonctions sont totalement déterminées. Nous optons pour une approximation directe de ces fonctions à l'aide des polynômes de Bernstein, dans le but de garantir la différentiabilité de la fonction objectif. À partir de cette nouvelle fonction objectif, nous procédons au calcul du vecteur de taux de compression optimal, lequel est ensuite transformé en allocation. Le graphique ci-dessous illustre le résultat de l'approximation par les polynômes de Bernstein, avec un paramètre n fixé à 150.

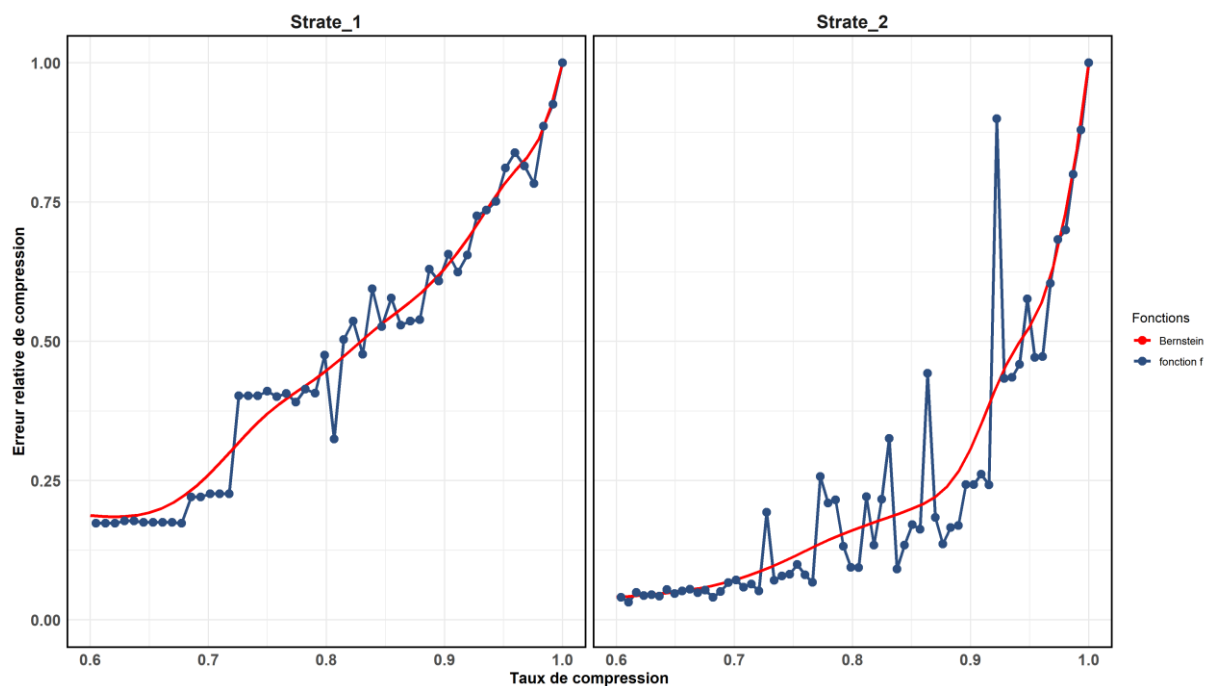


Figure 6. Approximation des fonctions f_g par les polynômes de Bernstein avec $n = 150$

On peut observer que les fonctions obtenues à partir de l'approximation par les polynômes de Bernstein sont significativement plus régulières que les fonctions f , ce qui témoigne de leur caractère dérivable. Ces approximations sont fournies au programme d'optimisation (***) pour minimisation. Les résultats de cette minimisation sont présentés dans le tableau ci-dessous.

Table 4. Allocation optimale réalisable et Erreur de compression pour $n = 150$

	Taux de comp. t^*	Allocation k^*	Ecart de BEL
Strate 1	0,766	30	526 874
Strate 2	0,876	20	186 837
Portefeuille	0,824	50	713 711

L'allocation optimale réalisable est le couple $(30, 20)$ avec une erreur de compression associée égale à 713 711. Cela signifie que si nous n'étions pas en mesure de résoudre le programme discret, cette allocation est la meilleure qu'il nous serait possible d'atteindre pour une valeur de $n = 150$ et ce quel que soit le nombre de valeurs intermédiaires p choisi. L'erreur de compression associée de 713 711 agit donc comme une limite infranchissable en dessous de laquelle il est impossible de descendre. On peut remarquer que cette allocation est au troisième rang dans la liste des allocations ordonnée par efficacité. Cependant, augmenter la valeur de n pourrait la rapprocher de l'allocation optimale située au premier rang et réduire l'erreur de compression limite.

4.3.4 L'allocation optimale réalisée

L'allocation optimale réalisée est la solution du programme (****) où l'on suppose que nous ne disposons d'aucune information a priori à propos des fonctions f . Dans ce cas, les seules informations dont nous serions sûrs sont que pour chacune des strates, la fonction f évaluée en 0 vaut 0 et évaluée en 1 vaut 1.

Pour obtenir un peu plus d'informations sur ces fonctions, il faut choisir un certain nombre p de valeurs intermédiaires de taux de compression (c.-à-d. comprises entre 0 et 1) pour lesquelles l'algorithme de compression sera réalisé afin d'obtenir les erreurs de compression correspondantes. Ce nombre p doit être suffisamment grand pour permettre de meilleures approximations des fonctions, mais aussi pas trop grand pour éviter une explosion des coûts relatifs à l'exécution de l'algorithme de compression. Dans notre exemple, nous allons choisir un nombre de point de plus en plus élevé afin de mettre en évidence un phénomène de convergence des solutions vers la solution optimale. Plus précisément, nous choisirons successivement $p = 5, 10$ et 15 .

Une fois le nombre de valeurs intermédiaires choisi, l'algorithme de compression est réalisé autant de fois que nécessaire, à la suite de quoi nous disposons de la base d'information nécessaire pour le calcul de la fonction d'interpolation linéaire f^{lin} . La Figure 7 montre les résultats du calcul de la fonction d'interpolation linéaire f^{lin} dans le cas où $p = 15$. Les graphiques pour les cas $p = 5$ et $p = 10$ peuvent être consultés en annexe (voir Figures 10 et 11).

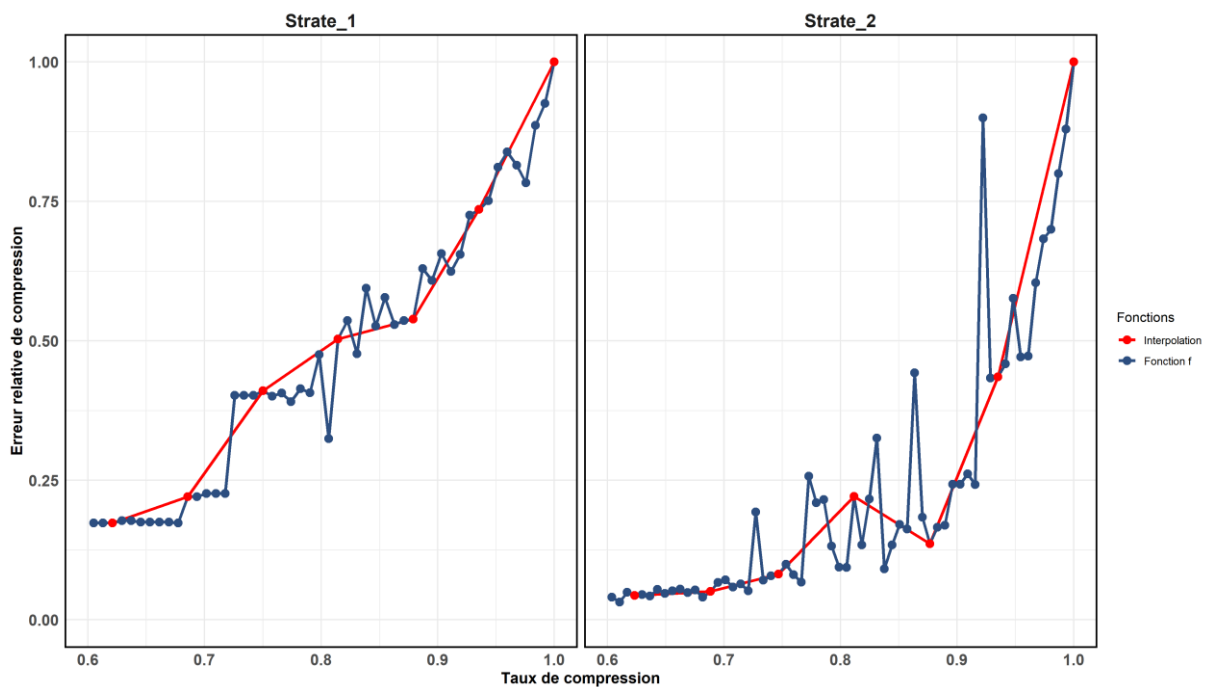


Figure 7. Interpolations linéaires des fonctions f_g avec $p = 15$

Une fois les interpolations linéaires obtenues, nous les approximations par des polynômes de Bernstein avec $n = 150$ afin de les rendre différentiables. La Figure 8 montre les résultats de cette approximation, encore une fois dans le cas où $p = 15$. Les graphiques pour les cas $p = 5$ et $p = 10$ peuvent être consultés en annexe (voir Figures 12 et 13).

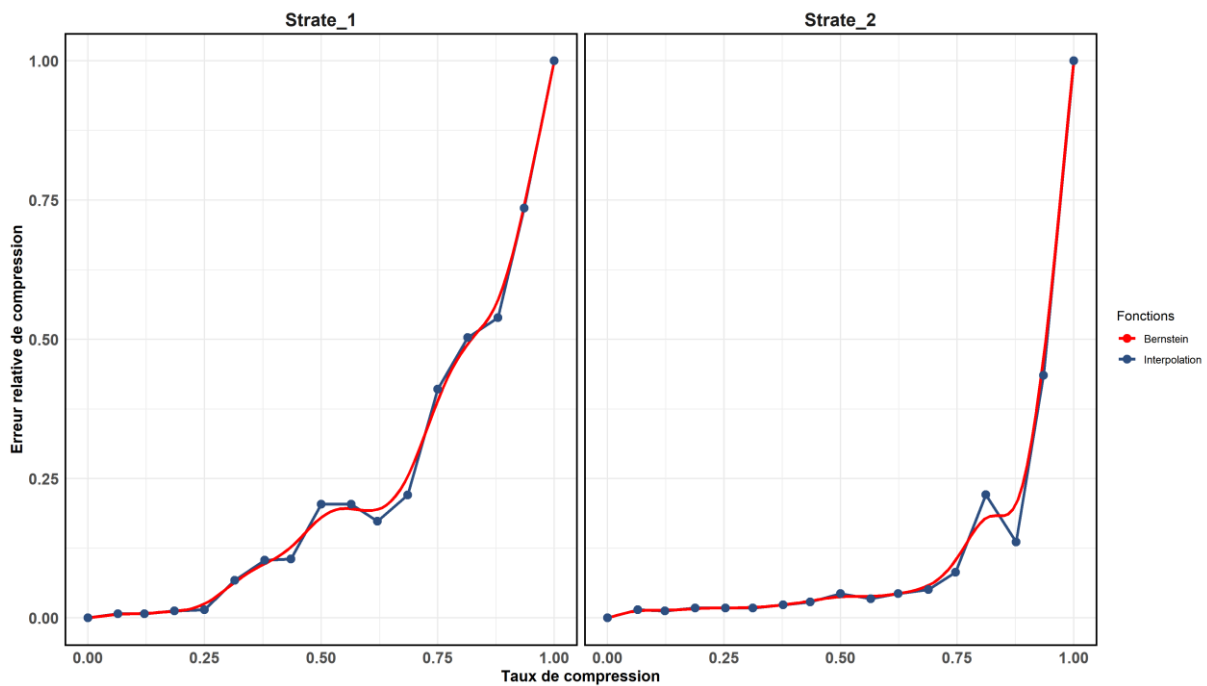


Figure 8. Approximation des fonctions f_g^{lin} par les polynômes de Bernstein avec $p = 15$ et $n = 150$

Enfin, ces approximations sont fournies au programme d'optimisation (***) pour minimisation. Les résultats de cette minimisation sont présentés dans les tableaux 5 et 6. Nous y présentons également les résultats de l'allocation proportionnelle et l'allocation optimale réalisable ($p = 153$).

Table 5. Allocations réalisées selon le nombre de valeurs intermédiaires p avec $n = 150$

	Alloc. Prop.	$p = 5$	$p = 10$	$p = 15$	$p = 153$
Strate 1	21	23	27	31	30
Strate 2	29	27	23	19	20
Portefeuille	50	50	50	50	50

Table 6. Erreurs de compression associées aux allocations réalisées selon p avec $n = 150$

	Alloc. Prop.	$p = 5$	$p = 10$	$p = 15$	$p = 153$
Strate 1	770 796	695 310	527 842	519 904	526 874
Strate 2	184 232	446 972	223 193	227 616	186 837
Portefeuille	955 028	1 142 282	751 035	747 520	713 711

Le tableau 6 montre qu'utiliser 5 valeurs intermédiaires n'est pas optimal. L'allocation issue de l'approche proportionnelle est meilleure que celle obtenue avec notre méthode. Cela est dû au fait que les interpolations linéaires des fonctions f réalisées à partir de ces 5 valeurs sont beaucoup trop grossières. Cependant, les interpolations linéaires deviennent meilleures pour $p = 10$ et $p = 15$ et cela est perceptible dans les résultats où les erreurs de compression de notre méthode sont nettement inférieures à l'erreur de compression de l'allocation proportionnelle. Par ailleurs, les erreurs de compression diminuent à mesure que p augmente et tendent vers l'erreur de compression limite. Cela est une illustration de la convergence de l'allocation optimale réalisée vers l'allocation optimale réalisable ou plus généralement de la convergence du programme (***) vers le programme (**).

$$\lim_{p \rightarrow 153} \text{lin_interp_error}(p) = 0$$

Pour terminer, nous rappelons que tous les polynômes de Bernstein utilisés jusqu'ici ont pour degré $n = 150$. Pour obtenir de meilleures approximations des fonctions sous-jacentes, il peut être intéressant d'augmenter ce degré. L'erreur d'approximation entre une fonction f et son approximation est calculée par la formule suivante :

$$\text{dist}(f, B_n(f)) = \int_0^1 |f(x) - B_n(f)(x)| dx$$

La Figure 9 montre l'évolution des erreurs d'approximation entre les fonctions f et leur approximation pour chaque strate selon des valeurs de n entre 150 et 10 000.

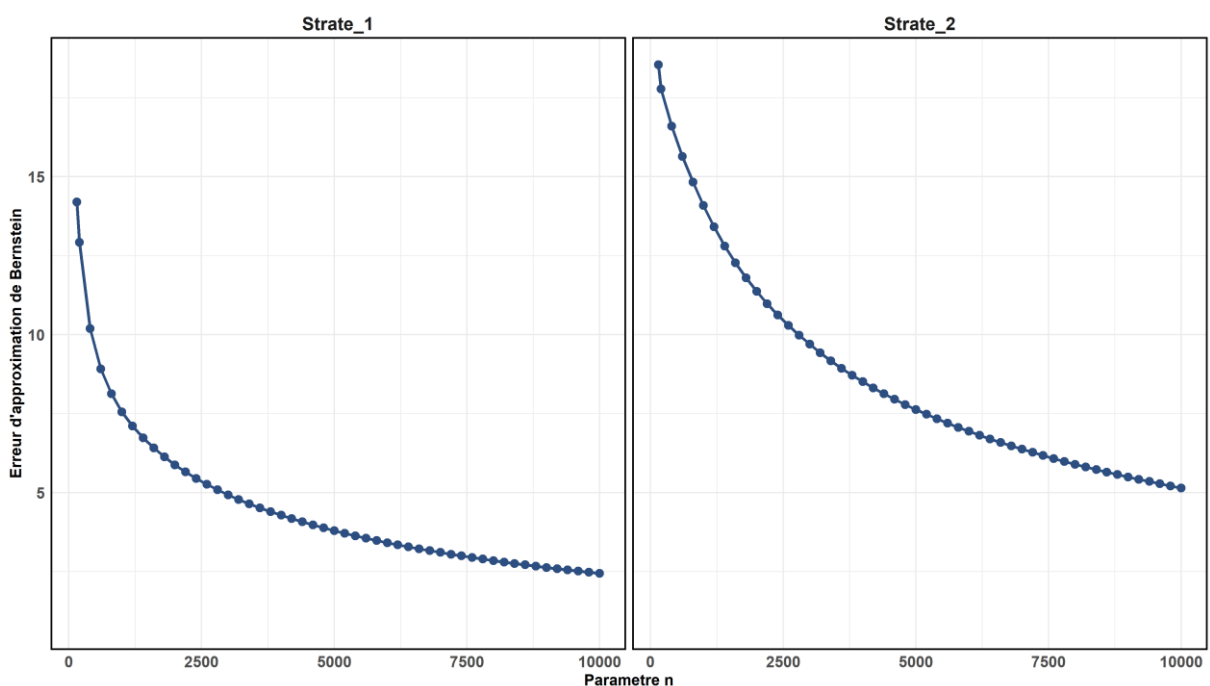


Figure 9. Erreur d'approximation des fonctions f par les polynômes de Bernstein selon la valeur de n

On constate que les erreurs d'approximation diminuent à mesure que l'on augmente le degré n ce qui signifie que les polynômes de Bernstein donnent des représentations de plus en plus fidèles des fonctions f . Cependant, les gains deviennent de moins en moins importants lorsque que la valeur de n devient trop élevée.

L'intérêt de réduire ces erreurs d'approximation réside dans le fait que des erreurs d'approximation de plus en plus faibles assurent que la fonction objectif $F_{bern}(t)$ du programme (***) se rapproche de la fonction objectif $F(t)$ du programme (**), ou en termes d'allocations, que l'allocation optimale réalisable converge vers l'allocation optimale à une erreur d'estimation μ près.

Les résultats de l'optimisation pour un degré des polynômes $n = 5\,000$ donnent une allocation optimale réalisable de $(36, 14)$, soit celle située au deuxième rang dans la liste des allocations triée par ordre décroissant d'efficacité.

5. Conclusion

En conclusion, notre étude a démontré l'efficacité de notre approche pour résoudre le défi de la détermination de l'allocation optimale dans le contexte de la compression des contrats. En utilisant des techniques de théorie de l'approximation de fonctions et d'optimisation différentiable, nous avons pu formuler et résoudre un problème d'optimisation visant à minimiser l'erreur de compression globale, tout en tenant compte des contraintes opérationnelles telles que le temps de calcul.

L'analyse approfondie des effets des hyperparamètres p pour le nombre de valeurs intermédiaires et n pour le degré des polynômes de Bernstein sur les programmes d'optimisation a révélé des tendances significatives : l'augmentation de la valeur de p réduit l'erreur d'interpolation, tandis que l'augmentation de la valeur de n diminue l'erreur d'approximation. Ces résultats fournissent des indications précieuses pour la conception pratique d'allocations optimales dans des contextes similaires.

En remplaçant l'approche traditionnelle de proratisation par la PM par une méthodologie plus précise et adaptable, notre approche ouvre de nouvelles perspectives pour améliorer la qualité des projections de cash-flows et optimiser les modèles Actif-Passif en approche Full ALM dans le secteur de l'assurance. En synthèse, notre recherche contribue à combler une lacune importante dans les pratiques actuelles en offrant une méthode plus sophistiquée et plus précise pour l'allocation des contrats, avec des implications significatives pour la prise de décision stratégique et opérationnelle.

Annexe : figures supplémentaires

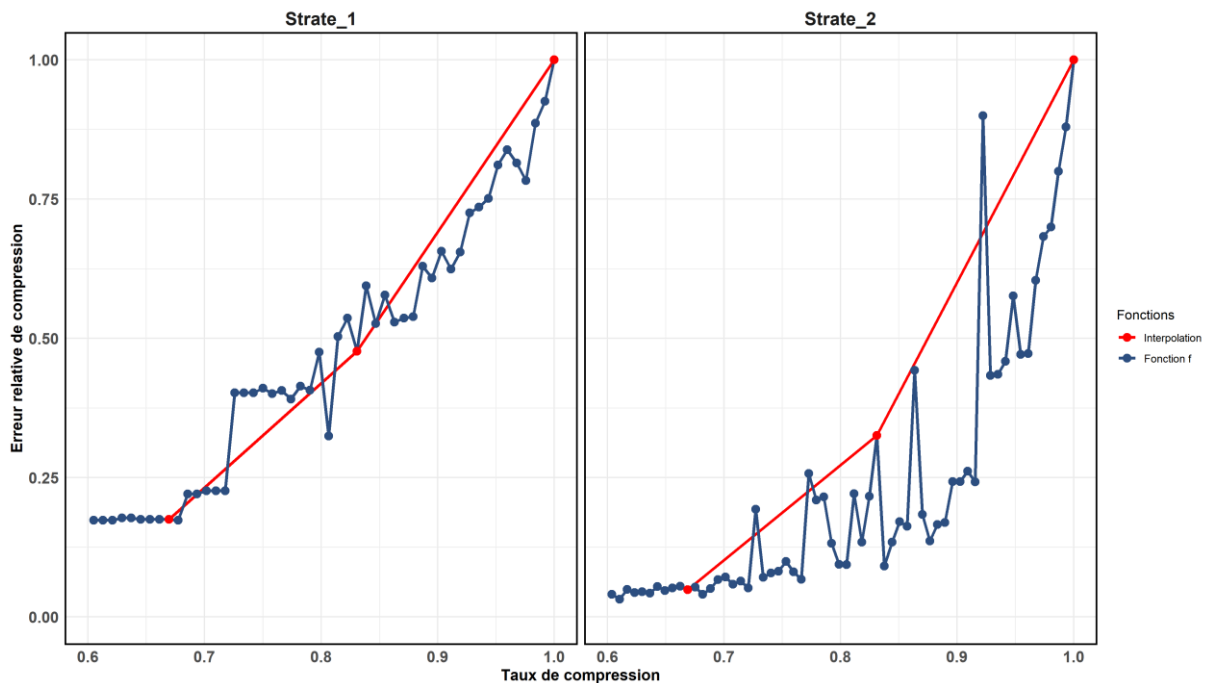


Figure 10. Interpolations linéaires des fonctions f_g avec $p = 5$

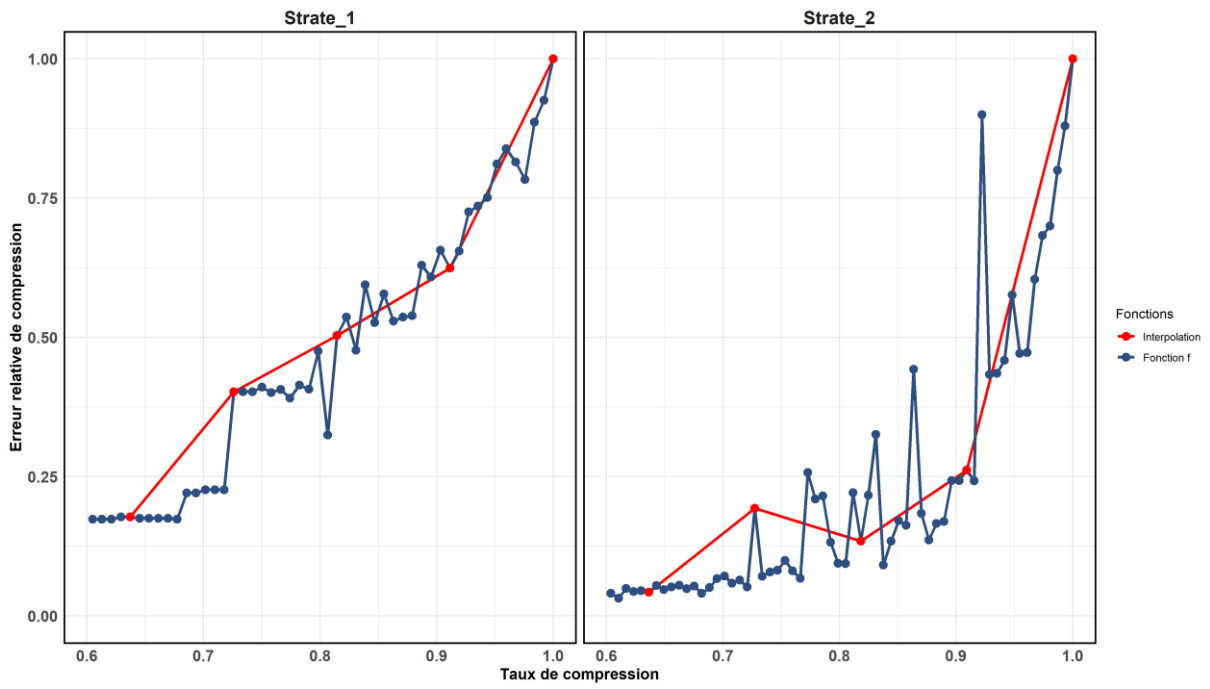


Figure 11. Interpolations linéaires des fonctions f_g avec $p = 10$

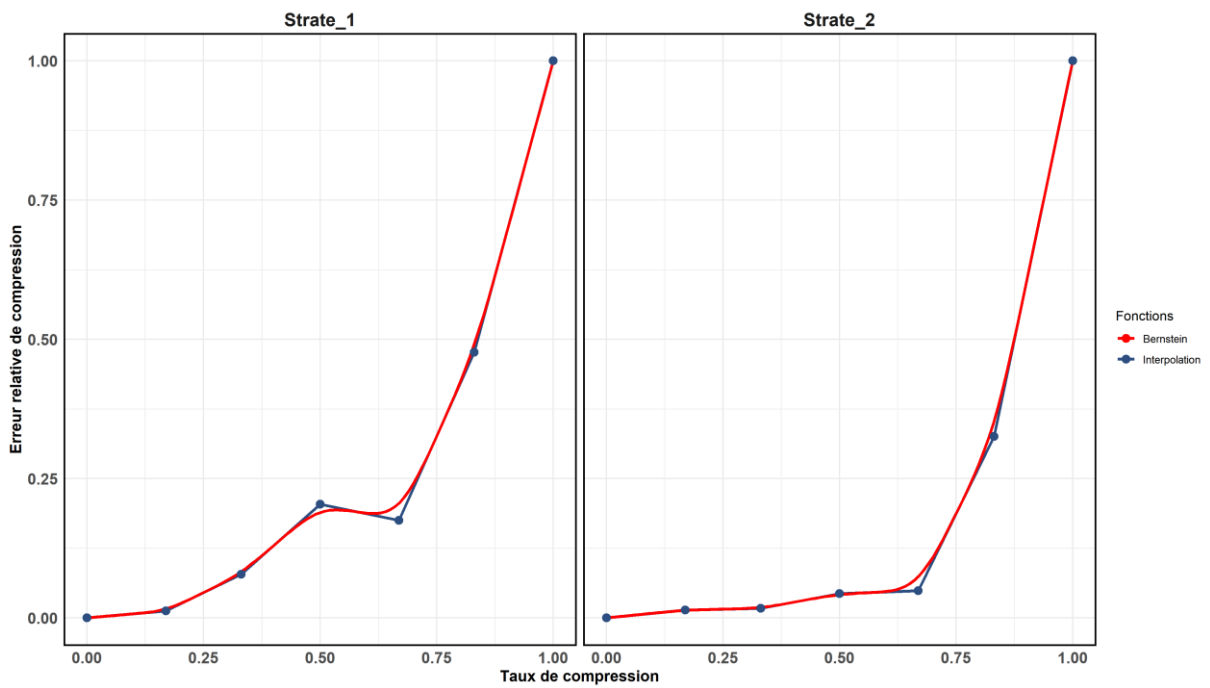


Figure 12. Approximation des fonctions f_g^{lin} par les polynômes de Bernstein avec $p = 5$ et $n = 150$

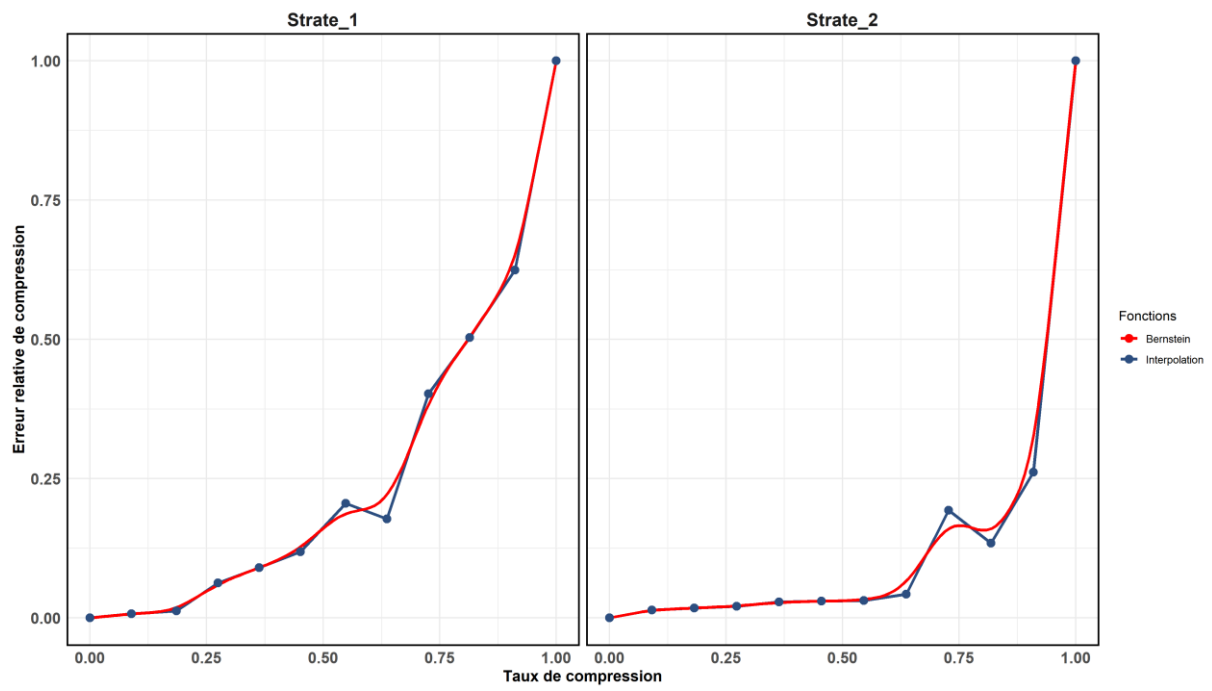


Figure 13. Approximation des fonctions f_g^{lin} par les polynômes de Bernstein avec $p = 10$ et $n = 150$