

TRABAJO FINAL DE MÁSTER

Título: Métodos estocásticos de cálculo de provisiones incluyendo el efecto del año de calendario. Una aplicación con Shiny.

Autoría: Diego Armando Samacá Amaya

Tutoría: Teresa Costa y Eva Boj

Curso académico: 2020-2021



UNIVERSITAT DE
BARCELONA

Facultat d'Economia
i Empresa

Màster
**de Ciències
Actuarials
i Financeres**

Facultad de Economía y Empresa

Universidad de Barcelona

Trabajo Final de Máster

Máster en Ciencias Actuariales y Financieras

Métodos estocásticos de cálculo de provisiones incluyendo el efecto del año de calendario. Una aplicación con Shiny.

Autoría:

Diego Armando Samacá Amaya

Tutoría:

Teresa Costa y Eva Boj

“El contenido de este documento es de exclusiva responsabilidad del autor, quien declara que no ha incurrido en plagio y que la totalidad de referencias a otros autores han sido expresadas en el texto”

Resumen

El presente trabajo busca desarrollar una aplicación con Shiny de RStudio para el cálculo estocástico de provisiones en seguros no vida que incluyen el efecto del año de calendario y obtener los resultados propuestos en los métodos deterministas de separación Aritmético (Verbeek, 1972) y Geométrico (Taylor, 1979), de forma que los *inputs* y *outputs* puedan ser visualizados a través de una interfaz gráfica interactiva. La metodología de cálculo hace uso del Modelo Lineal Generalizado y la técnica estadística de remuestreo o *Bootstrapping* propuesta por Björkwall et al. (2009, 2010) para la estimación de las reservas.

Palabras clave: Modelo Lineal Generalizado, Bootstrap, cálculo estocástico de provisiones, métodos de separación de Taylor, Shiny (RStudio).

Abstract

The present work seeks to develop an application with Shiny from RStudio for the stochastic calculation of non-life insurance provisions including the effect of the calendar year and obtain the results proposed in the Arithmetic (Verbeek, 1972) and Geometric (Taylor, 1979) deterministic separation methods, so that the inputs and outputs can be viewed through an interactive graphical interface. The calculation methodology makes use of the Generalized Linear Model and the statistical technique of resampling or Bootstrapping proposed by Björkwall et al. (2009, 2010) for estimating claim reserves.

Keywords: Generalized Linear Model, Bootstrap, stochastic claim reserving, Taylor's separation methods, Shiny (RStudio).

Índice

1. Introducción	1
2. Métodos estocásticos de cálculo de provisiones con GLM	2
2.1 Datos	2
2.1.1 Separación Aritmética: Verbeek (1972).....	3
2.1.2 Separación Geométrica: Taylor (1979).....	4
2.2 Descripción GLM.....	5
2.3 Caso particular. Métodos de separación de Taylor	7
2.3.1 GLM para la separación Aritmética	8
2.3.2 GLM para la separación Geométrica.....	8
2.4 Errores de predicción mediante estimación analítica	9
2.5 Bootstrapping	9
3. Desarrollo de una aplicación con Shiny	11
4. Caso práctico	13
5. Conclusiones	21

Lista de Gráficos

Gráfico 1: Esquema Triángulo de desarrollo.....	3
Gráfico 2: Panel de Entrada ShinyApp	11
Gráfico 3: Triángulo con cuantías acumuladas	13
Gráfico 4: Triángulo de desarrollo completo.....	13
Gráfico 5: Resumen de resultados por año de origen.....	14
Gráfico 6: Resumen de resultados por año de calendario	14
Gráfico 7: Histograma de Distribuciones predictivas	15
Gráfico 8: Costos finales de reclamaciones por año de origen	15
Gráfico 9: Últimas reclamaciones por año de origen	16
Gráfico 10: Pagos futuros simulados por año de calendario	16
Gráfico 11: Triángulo con cuantías desacumuladas (Ar).....	17
Gráfico 12: Resumen de resultados por año de origen (Ar).....	17
Gráfico 13: Resumen de resultados por año de calendario (Ar)	18
Gráfico 14: Histograma de Distribuciones predictivas (Ar).....	18
Gráfico 15: Costos finales de reclamaciones por año de origen (Ar).....	19
Gráfico 16: Últimas reclamaciones por año de origen (Ar)	19
Gráfico 17: Pagos futuros simulados por año de calendario (Ar)	20

1. Introducción

El desarrollo del presente trabajo final de máster busca ampliar los conocimientos de los métodos estocásticos del cálculo de provisiones en seguros no vida con el Modelo Lineal Generalizado (GLM por sus siglas en inglés) estudiados en la asignatura de Matemática Actuarial, introduciendo el efecto del año de calendario para obtener como caso particular los resultados de los métodos deterministas de separación de Taylor.

El estudio de estos métodos deterministas obedece a que, a diferencia de otras técnicas como el modelo Chain Ladder, estos permiten introducir como hipótesis de cálculo factores identificables por año de calendario como la inflación, cambios legislativos, compensaciones al asegurado entre otros, que están condicionados de acuerdo con la naturaleza de cada cartera, siendo factores aportantes de incertidumbre a la hora de realizar el cálculo de provisiones. Luego, la introducción del comportamiento de esta inflación de siniestros superpuestos permitirá disminuir el riesgo de reserva (Björkwall, (2009), Björkwall et al. (2010)) obteniendo una cuantificación y valoración de provisiones más cercana a una estimación real en el marco normativo de Solvencia II.

Entretanto, el GLM se ha consolidado como la metodología dominante para el cálculo de reservas y, en ese mismo sentido, la programación informática de la matemática actuarial ha permitido realizar las estimaciones de manera eficiente para obtener los cálculos propuestos en la teoría, haciendo uso de paquetes y funciones actuariales desarrollados en diversos lenguajes de programación como R. Además, la visualización e interpretación de los resultados mediante herramientas disponibles de interfaz gráfica como Shiny, facilitan el análisis y presentación de los outputs arrojados al usuario.

En la segunda sección del documento se revisan el Modelo Lineal Generalizado y su aplicabilidad para obtener los métodos de separación de Taylor, el cálculo de los errores de predicción y el Bootstrapping utilizado en las estimaciones. En la tercera y cuarta sección se describen la estructura y el esquema de la interfaz gráfica de Shiny, con los parámetros de entrada y resultados presentados utilizando el dataset de reclamaciones de Taylor y Ashe (1983). Finalmente, se presentan las conclusiones del desarrollo de la aplicación y futuras líneas de desarrollo.

2. Métodos estocásticos de cálculo de provisiones con GLM

De acuerdo con la clasificación realizada por De Vylder (1986) y Taylor (1986), los métodos actuariales para el cálculo de provisiones en seguros no vida se pueden clasificar en deterministas o estocásticos, en el segundo caso los siniestros se consideran eventos de variables aleatorias.

Se distinguen dentro de los modelos estocásticos (Boj, et al. 2020):

- a) modelos paramétricos, que asumen distribuciones conocidas, pero con la necesidad de estimar algunos parámetros
- b) modelos de distribución libre, que no establecen hipótesis de partida para las distribuciones implicadas.

Algunos métodos deterministas (método de Chain-ladder, método de mínimos cuadrados de De Vylder, método de separación aritmética de Taylor y método de separación geométrica de Taylor) pueden obtenerse como casos particulares de los métodos estocásticos. En concreto, la versión estocástica del método de Chain-ladder ha sido desarrollada tanto a partir de modelos paramétricos (England y Verrall (2002), Kaas et al., (2008)) como de distribución libre (Mack, 1993).

El tratamiento estocástico de los modelos de separación de Taylor se encuentra en Kaas et al. (2008) y Björkwall et al. (2009, 2010, 2011), asumiendo distribuciones paramétricas para el número y las cuantías de los siniestros.

Mediante la aplicación de los métodos deterministas se puede obtener una estimación de las provisiones, mientras que en el caso de los métodos estocásticos adicionalmente se puede calcular el error cometido en la estimación de las mismas.

Según las hipótesis establecidas en los modelos estocásticos, se pueden calcular los errores de predicción de manera analítica o mediante técnicas de remuestreo (*Bootstrapping*).

2.1 Datos

Para el cálculo de provisiones, de manera generalizada, se parte de un triángulo de desarrollo compuesto por valores aleatorios de $c_{i,j}$, que representan las cantidades observadas pagadas en el año de desarrollo (o periodo de retraso) j para $j = 0, 1, \dots, k$ de los siniestros originados en el año i (año de ocurrencia) para $i = 0, 1, \dots, k$, como se presenta en el gráfico 1.

Específicamente para los métodos de separación de Taylor, se utiliza el número de siniestros estimado a final del año de origen i (n_i), obteniendo un triángulo compuesto por los valores de las cuantías medias $s_{i,j}$ con:

$$s_{i,j} = \frac{c_{i,j}}{n_i}.$$

		<i>Año de desarrollo</i>				
		<i>0</i>	<i>1</i>	<i>...</i>	<i>k-1</i>	<i>k</i>
<i>Año de origen</i>	<i>0</i>	$c_{0,0}$	$c_{0,1}$	\dots	$c_{0,k-1}$	$c_{0,k}$
	<i>1</i>	$c_{1,0}$	$c_{1,1}$	\dots	$c_{1,k-1}$	
	\dots	\dots	\dots	\dots		
	<i>k-1</i>	$c_{k-1,0}$	$c_{k-1,1}$			
	<i>k</i>	$c_{k,0}$				

Gráfico 1: Esquema Triángulo de desarrollo. **Fuente:** Elaboración propia

Estos métodos parten de la hipótesis donde las cuantías medias s_{ij} corresponden al producto:

$$s_{ij} = r_j \lambda_{i+j}$$

siendo:

r_j : la proporción de la cuantía de los siniestros ocurridos en el año de origen i pagada en el año de desarrollo j

λ_{i+j} : el índice que mide los factores externos incidentes (p.ej. inflación).

Este último valor es coincidente para todas las cuantías que se encuentran situadas en el mismo momento del tiempo (mismo año de calendario), en la contra diagonal del triángulo de desarrollo.

Adicionalmente, para completar la parte inferior del triángulo (predicciones), se deben estimar los valores de $\hat{\lambda}_h$ para h mayores a k , y el efecto de los factores del año de pago se debe estimar de forma separada.

El cálculo de los estimadores de los parámetros se realiza de acuerdo con las formulaciones del siguiente apartado, revisadas desde Boj et al. (2020).

2.1.1 Separación Aritmética: Verbeek (1972)

En la separación aritmética, se debe cumplir que:

$$\sum_{j=0}^k r_j = 1.$$

Las sumas de las componentes de cuantías medias por columna v_j y diagonal d_j se obtienen a partir de:

$$v_j = \sum_{i=0}^{k-j} s_{i,j}$$

$$d_j = \sum_{i=0}^j s_{i,j-i}.$$

Luego, los estimadores de los parámetros para los valores de h desde 0 hasta k vienen dados por:

$$\hat{r}_h = \frac{v_h}{\sum_{j=h}^k \hat{\lambda}_j}$$

$$\hat{\lambda}_h = \frac{d_h}{1 - \sum_{j=h+1}^k \hat{r}_j}.$$

2.1.2 Separación Geométrica: Taylor (1979)

De forma análoga, se condicionan las proporciones r_j a:

$$\prod_{j=0}^k r_j = 1.$$

Los productos de las componentes de cuantías medias por columna u_j y diagonal w_j se obtienen a partir de:

$$u_j = \prod_{i=0}^{k-j} s_{i,j}$$

$$w_j = \prod_{i=0}^j s_{i,j-i}.$$

Ahora los estimadores de los parámetros para los valores de h desde 0 hasta k vienen dados por:

$$\hat{r}_h = \left(\frac{u_h}{\prod_{j=h}^k \hat{\lambda}_j} \right)^{\frac{1}{k-h+1}}$$

$$\hat{\lambda}_h = \left(w_h \prod_{j=h+1}^k \hat{r}_j \right)^{\frac{1}{h+1}} .$$

2.2 Descripción GLM

Los modelos GLM, estudiados inicialmente por Nelder y Wedderburn (1972), presentan una serie de ventajas a la hora de resolver situaciones de tipo actuarial, en comparación con los modelos lineales ordinarios, puesto que permiten que las desviaciones aleatorias de la media tengan distribuciones diferentes a la Normal, haciendo uso de aquellas pertenecientes a la familia de distribución exponencial (McCullagh y Nelder, 1989), con una función de densidad:

$$f(y_i; \theta_i, \phi_i) = \exp \left\{ \frac{y_i \theta_i - b(\theta_i)}{a(\phi_i)} + c(y_i, \phi_i) \right\}$$

siendo θ_i y ϕ_i los parámetros canónicos y de dispersión respectivamente, como se presenta en Boj y Costa (2014). De igual forma, la media de la variable aleatoria puede ser lineal en cualquier escala (p.ej. logarítmica), y no ser únicamente una función lineal de las variables explicativas.

Kaas et al. (2008) presentan 3 características fundamentales del GLM:

- a. Un **componente estocástico** del modelo que determina que las observaciones de

$$Y_i \quad \forall i = 1, 2, \dots, n$$

son variables aleatorias independientes, que presentan una densidad perteneciente a la familia de las distribuciones exponenciales (Normal, Poisson, Poisson sobre dispersa, Gamma, Gaussiana inversa, etc.) En cada una de estas, existe un valor esperado μ_i que se corresponde con una varianza específica ϕ_i , de modo que:

$$\text{Var} [Y_i] = \phi_i V(\mu_i),$$

siendo $V(\mu_i)$ una *función de varianza* que depende del valor esperado.

- b. Un **componente sistemático** del modelo que atribuye a cada observación un *predictor lineal* η_i dado por:

$$\eta_i = \sum_j x_{ij} \beta_j$$

siendo x_{ij} los regresores y $\beta_j \quad \forall j = 1, 2, \dots, p$ los parámetros lineales. De aquí se puede expresar que:

$$E[Y_i] = \mu_i.$$

- c. Una **función de enlace** que vincula el valor esperado de Y_i con el predictor lineal:

$$\eta_i = g(\mu_i),$$

siendo esta una función inversa del valor esperado,

$$g^{-1}(\eta_i) = E[Y_i].$$

En algunos casos, el parámetro canónico coincide con el predictor lineal:

$$\eta_i = \theta(\mu_i).$$

Las **funciones de varianza** $V(\cdot)$ vienen dadas por los parámetros de media μ y dispersión \emptyset , que permiten modelar adecuadamente diversos problemas actuariales con diferentes comportamientos heterocedásticos (Kaas et.al. 2018):

- Para la distribución Normal, $V(\cdot) = \emptyset\mu^0 = \emptyset$.
- En la distribución Poisson donde media y varianza son iguales, $V(\cdot) = \mu$. Para la clase con múltiplos de Poisson que tienen varianza proporcional a su media, esta es igual a $V(\cdot) = \emptyset\mu$.
- Para la distribución Gamma, con parámetro de forma fijo y coeficiente de variación constante, $V(\cdot) = \emptyset\mu^2$.
- En la distribución gaussiana inversa, $V(\cdot) = \emptyset\mu^3$.

Para cada distribución existe una **función de enlace natural o canónica** (ver Fox, 2016):

- En la distribución Normal, coincide con la identidad, $\eta_i = \mu_i$, que induce a modelos de tipo aditivo.
- Para la distribución Poisson, la función es logarítmica, $\eta_i = \log(\mu_i)$, conduciendo a modelos multiplicativos log-lineales.
- En el caso de la distribución Gamma, es el recíproco.

En términos generales, y de acuerdo con lo presentado por Boj et al. (2020), se define la familia de enlaces paramétricos como:

$$\eta_i = g(\mu_i) = \begin{cases} \log(\mu_i) & \text{si } \lambda = 0 \\ \mu_i^\lambda & \text{si } \lambda \neq 0. \end{cases}$$

La **estimación de los parámetros** en el GLM se efectúa maximizando el logaritmo de la función de verosimilitud (desarrollo detallado en Boj y Costa, 2014), que, para el caso de la familia de exponencial, corresponde a las funciones de cuasi verosimilitud. Aun cuando el modelo lineal generalizado permite elegir la distribución del error y la función de enlace, lo que supone una mayor simplificación del modelo, la calidad del GLM está dada por su log-verosimilitud, anteponiéndose a su parsimonia.

Para determinar la **calidad de ajuste del modelo**, es necesario evaluar sus residuos estandarizados, teniendo en cuenta los parámetros estimados y la función de varianza. De manera similar al análisis de la varianza-ANOVA realizado para los modelos lineales en los modelos logit y probit, en el modelo lineal generalizado se calculan los residuos de la desviación o *desviación residual* (Fox, 2016) como sigue:

$$D_m = 2(\ln L_s - \ln L_m)$$

siendo L_s la máxima verosimilitud del modelo propuesto y L_m la del modelo saturado¹.

En cuanto al **análisis de los residuos**, Boj y Costa (2014) destacan:

- a) los residuos de Pearson, que, para el caso de la distribución Normal con una función de enlace identidad, tienen una distribución χ^2 , que es proporcional a la diferencia de las máximas verosimilitudes:

$$r_i^{Pearson} = \frac{(y_i - \hat{\mu}_i)}{\sqrt{V(\hat{\mu}_i)/w_i}}$$

- b) los residuos de desviación², que indican cuanto contribuye cada observación a la desviación escalada:

$$r_i^{Desviación} = \text{signo}(y_i - \hat{\mu}_i) \sqrt{d_i}.$$

Según Boj y Costa (2014), se establece que, de forma similar al tratamiento utilizado para la función de enlace, también se utiliza la familia paramétrica de distribuciones exponenciales para la **distribución del error**:

$$V(\mu_i) = \mu_i^\zeta.$$

donde ζ determina la estructura del error, entre una Poisson ($\zeta = 1$), una Gamma ($\zeta = 2$) o una inversa Gaussiana ($\zeta = 3$).

2.3 Caso particular. Métodos de separación de Taylor

Gracias a la aplicabilidad del Modelo Lineal Generalizado para cálculo de provisiones, algunas instancias del modelo se han convertido en técnicas actuariales relevantes. Mediante el uso del GLM, se pueden obtener los resultados de los métodos deterministas, especificando los parámetros característicos de cada método, como se desarrolla a continuación.

En los métodos de separación, se parte del supuesto que en cada **año de desarrollo** j se paga un porcentaje fijo del total, y que existe un **efecto adicional** (inflación, velocidad de liquidación de los siniestros, variación del tamaño de la cartera, regulaciones, propensiones de reclamación, etc.) *constante* para cada año de calendario k , es decir, para las observaciones en las contra diagonales del triángulo, determinado, en este caso, por $k = i + j + 1$. Un ejemplo de su asertividad se presenta en los seguros como los de responsabilidad civil para la profesión médica, donde el riesgo aumenta cada año y puede presentarse una tendencia de mayor compensación como resultado de un veredicto.

¹ Un modelo saturado asigna un parámetro a cada observación, con lo que se obtiene un elevado nivel de ajuste de los datos.

² Desviación definida por las autoras como la suma de las diferencias para cada uno de los datos:
 $D(y, \hat{\mu}) = \sum_{i=1}^n d_i(y_i, \hat{\mu}_i)$

2.3.1 GLM para la separación Aritmética

Para el cálculo de las componentes faltantes c_{ij} , $\forall i + j + 1 > k$ del triángulo de cuantías, se toma como punto de partida un modelo multiplicativo con parámetros independientes para cada columna j y diagonal (año de calendario) k , en el que las componentes s_{ij} se distribuyen:

$$s_{ij} \sim \text{Poisson}(\beta_j \gamma_k); \text{ con } \alpha_i = 1$$

deduciendo así una función de enlace de tipo logarítmico. Luego, el valor esperado de s_{ij} es igual a:

$$E[s_{ij}] = \beta_j \gamma_{i+j+1}$$

manteniendo la restricción inicial del método:

$$\sum_{j=0}^t \beta_j = 1$$

donde β_j es la proporción liquidada en cada año de desarrollo.

2.3.2 GLM para la separación Geométrica

Para obtener los resultados de la separación geométrica a partir del GLM, se asume que los logaritmos de las componentes del triángulo (independientes) se distribuyen según una Normal con media y varianza:

$$\log(s_{ij}) \sim N(\log(\beta_j \gamma_{i+j+1}), \sigma^2); \text{ con } \alpha_i = 1.$$

Luego, se obtiene una regresión de tipo aditivo donde el valor esperado de las log-componentes corresponde a la suma de los logaritmos de los parámetros de los efectos por año de desarrollo j y año de calendario k :

$$E[\log s_{ij}] = \log \beta_j + \log \gamma_{i+j+1}.$$

En este caso, los parámetros pueden calcularse de manera recursiva, teniendo en cuenta la restricción del método de separación geométrica para el efecto del año de desarrollo:

$$\prod_{j=0}^t \beta_j = 1.$$

Por su parte, el parámetro estimado $\hat{\gamma}_t$ que refleja el efecto del año de calendario para años futuros puede ser calculado mediante diferentes métodos, como extrapolación log-lineal (ver Kaas et al, 2008) o a partir de una tasa de incremento anual que se aplica sobre el último valor estimado (Björkwall et al., 2010).

2.4 Errores de predicción mediante estimación analítica

El error de predicción mediante estimación analítica se obtiene como la raíz cuadrada del error cuadrático medio (MSE), aproximado desde la suma de la varianza del proceso (varianza de las observaciones) y la varianza de la estimación, como sigue:

$$E[(c_{ij} - \hat{c}_{ij})^2] \cong \text{Var}[c_{ij}] + \text{Var}[\hat{c}_{ij}].$$

Asumiendo que la varianza de proceso es $\text{Var}[c_{ij}] = \emptyset \mu_{ij}^{\zeta}$ para la familia de distribuciones paramétricas, para la varianza de la estimación se utiliza la aproximación³:

$$\text{Var}[\hat{c}_{ij}] \cong \left| \frac{\partial \mu_{ij}}{\partial \eta_{ij}} \right|^2 \text{Var}[\hat{\eta}_{ij}]$$

con lo cual se podrán obtener los MSE para cada estimación de los pagos futuros, para la estimación por año de origen, para la estimación por año de calendario y para la estimación por totales (ver Boj y Costa, (2014) y Boj y Costa, (2017)).

2.5 Bootstrapping

Aplicando la metodología *Bootstrapping* se puede estimar la varianza de la estimación $\text{Var}[\hat{c}_{ij}]$ a partir del error estándar de la distribución predictiva de c_{ij} . En concreto, se utilizan los *Bootstrapping residuals* basado en los residuos de Pearson (ver Boj y Costa, 2014) para los modelos lineales generalizados, a excepción de los métodos de separación.

La evaluación de los errores de predicción y la distribución predictiva obtenida mediante los métodos estocásticos de separación se realizarán de acuerdo con lo propuesto por Björkwall et al. (2010). El método se basa en el principio de conexión (*plug-in principle*) propuesto por Efron y Tibshirani (1993), en donde los resultados obtenidos con simulaciones en un escenario Bootstrap (*Bootstrap world*), permiten de forma análoga obtener valores válidos para el escenario real (*Real world*).

En este caso, el método parte del triángulo de desarrollo de cuantías, de forma que los parámetros estimados por año de desarrollo y año calendario (\hat{r}_j y $\hat{\lambda}_k$) de las observaciones se obtienen de la misma forma que en los métodos de separación presentados anteriormente. Para el cálculo del factor que mide el efecto de la inflación en la parte inferior del triángulo, dado un valor predeterminado K , se asume una relación de tipo porcentual:

$$\hat{\lambda}_{k+1} = (1 + 0.01K) \hat{\lambda}_k, \quad \forall t < k \leq 2t$$

siendo t el último año de calendario en el triángulo de desarrollo.

³ Aproximación realizada mediante el método delta, basado en la aproximación de media asintótica, demostrada mediante las expansiones de Taylor alrededor de la media. (Kaas et al, 2008).

Luego, se asigna una distribución Gamma a las cuantías a estimar (condicionadas a un número de siniestros dado por año de origen n_i), teniendo en cuenta su flexibilidad y positividad, con sus respectivos parámetros de forma y escala:

$$c_{ij}|n_i \in \Gamma \left(\frac{n_i}{\hat{\phi}}, \hat{r}_j \hat{\lambda}_k \hat{\phi} \right)$$

con esperanza y varianza:

$$E [c_{ij}|n_i] = n_i \hat{r}_j \hat{\lambda}_k = \hat{m}_{ij}$$

$$Var [c_{ij}|n_i] = \hat{\phi} n_i (\hat{r}_j \hat{\lambda}_k)^2$$

correspondientes a un GLM ponderado con una función de enlace logarítmico. Entretanto, el parámetro ϕ es obtenido mediante una estimación tipo Pearson de modo que:

$$\hat{\phi} = \frac{1}{T - q} \sum n_i \frac{(c_{ij} - n_i \hat{r}_j \hat{\lambda}_k)^2}{(n_i \hat{r}_j \hat{\lambda}_k)^2}$$

donde $T = (t + 1)(t + 2) / 2$ y $q = 2t - 1$.

Una vez se han determinado los parámetros necesarios, se completan los valores del triángulo superior e inferior (ajustados y futuros, respectivamente) obtenidos mediante los métodos de separación, que sumados darán como resultado los valores totales estimados de provisiones por año de origen (\hat{R}_i), por año de calendario (\hat{R}_k) y total.

Luego, se trasladan los datos al escenario *Bootstrap* para obtener la parte superior (observada) del triángulo mediante la distribución Gamma planteada, y con estos nuevos datos, se estiman de nuevo los valores \hat{m}_{ij} de las reclamaciones pendientes futuras (triángulo inferior). Se almacenan los residuos de predicción (estandarizados o no) y se repite el proceso de *Bootstrapping* B veces.

Finalmente, para analizar los resultados, se obtienen las distribuciones predictivas total y/o por año de origen y año de calendario del escenario real (estandarizados o no), graficando los B resultados obtenidos a partir del *Bootstrapping* (ver Björkwall et al., 2009 y 2010).

3. Desarrollo de una aplicación con Shiny

El objetivo del trabajo es desarrollar una aplicación con Shiny que permita visualizar los siguientes resultados del cálculo estocástico de provisiones mediante los métodos de separación revisados previamente:

- Triángulo inicial con cuantías (acumuladas o incrementales).
- Triángulo de desarrollo completo de acuerdo con alguno de los métodos de separación disponibles.
- Provisiones, pagos futuros, errores de predicción, coeficiente de variación y cuantiles de la distribución (por año de origen, año de calendario y totales)
- Gráficos para representar las distribuciones predictivas con Bootstrap, los costos finales de reclamaciones por año de origen, últimas reclamaciones por año de origen y los pagos futuros simulados por año de calendario

El usuario podrá seleccionar los inputs (*Sidebar panel*):

- Método de separación aritmética o geométrica de Taylor
- Efecto sobre el año de calendario (%)
- Número de remuestreos - Bootstrap (*B*)
- Semilla para replicar resultados.

Visualización de los parámetros de entrada en el panel de control:

The screenshot shows a Shiny app control panel titled "GLM for Taylor's Separation Methods". It includes the following elements:

- Triangle:** Two radio buttons, "Incremental" (unselected) and "Acumulative" (selected).
- Taylor Separation Method:** A dropdown menu currently set to "Geometric".
- Calendar Year Effect (%):** A text input field containing the value "0".
- Enter a percentage value:** A label below the input field.
- Samples Bootstrapping:** A text input field containing the value "1000".
- Set seed:** A text input field containing the value "7".
- Enter an specific seed #:** A label below the input field.
- Calculate:** A button to execute the calculation.
- Instructions:** A note at the bottom stating "Click the 'Calculate' button to see the results on the main panel".

Gráfico 2: Panel de Entrada ShinyApp. **Fuente:** Elaboración propia

Una vez establecidos los parámetros de entrada para el cálculo, se definen las funciones que permitirán la compilación y ejecución de la interfaz gráfica. La arquitectura básica de las aplicaciones en Shiny está compuesta por 3 componentes: un objeto *ui* (user interface), una función *server* y la ejecución de la función invocando la función *ShinyApp*.

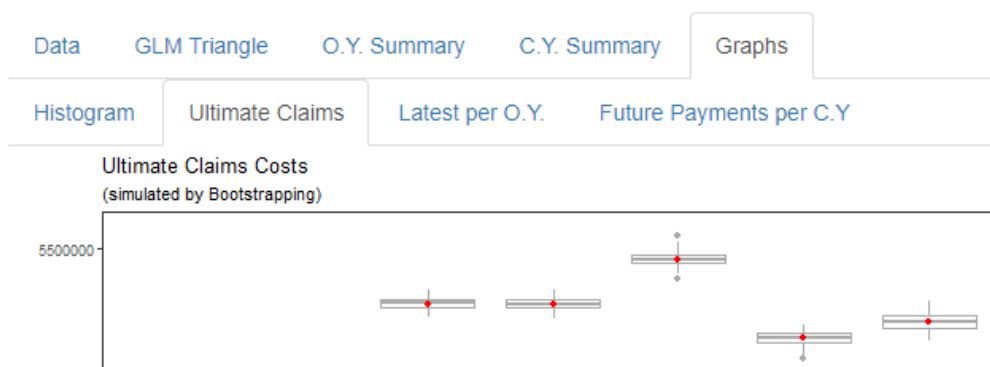
El diseño y controles de la aplicación son definidos en el objeto *ui*, mientras que en *server* se declaran las instrucciones necesarias para crear la aplicación. La función *ShinyApp* (*ui*, *server*) crea objetos de tipo Shiny para cada par *ui/server*. Por su parte, las ejecuciones son efectuadas a través de las funciones *observeEvent* y *eventReactive*, que trasladan los datos de los inputs hacia la función *server*.

En el panel de salida (*main panel*), los resultados serán visualizados a través de diferentes pestañas mediante el uso de la función *tabPanel*, de modo que permita una lectura visual más clara y diferenciada para el usuario. Los triángulos indicarán los años de origen y de desarrollo, mientras que el resumen de resultados (*Summary by Origin Year* y *Summary by Calendar Year*) presentarán los outputs por año de origen y año de calendario, respectivamente. Teniendo en cuenta las utilidades de la aplicación y haciendo uso de la función *tabBox*, los gráficos se presentan de forma individual y nominada, facilitando así su visualización.

Visualización de las pestañas con resúmenes de resultados:

	Data	GLM Triangle	O.Y. Summary	C.Y. Summary	Graphs			
Initial Triangle								
year	1	2	3	4	5	6	7	8
1	357848	766940	610542	482940	527326	574398	146342	139950
2	352118	884021	933894	1183289	445745	320996	527804	266172

Subpestañas con gráficos:



4. Caso práctico

El cálculo de provisiones mediante los métodos de separación obtenidos a partir del modelo lineal generalizado ha sido realizado con el *dataset* propuesto por Taylor y Ashe (1983), disponible en el paquete *ChainLadder* de R. El número de siniestros por año de origen fue estimado a partir de un triángulo de cuantías propuesto por los mismos autores.

A continuación, se presentan los resultados de la ejecución seleccionando el método Geométrico de separación de Taylor, un efecto de año de calendario nulo, 1000 muestras para *Bootstrapping* y semilla 7:

- Datos iniciales:

Data	GLM Triangle	O.Y. Summary	C.Y. Summary	Graphs						
Initial Triangle										
year	1	2	3	4	5	6	7	8	9	10
1	357848	1124788	1735330	2218270	2745596	3319994	3466336	3606286	3833515	3901463
2	352118	1236139	2170033	3353322	3799067	4120063	4647867	4914039	5339085	NA
3	290507	1292306	2218525	3235179	3985995	4132918	4628910	4909315	NA	NA
4	310608	1418858	2195047	3757447	4029929	4381982	4588268	NA	NA	NA
5	443160	1136350	2128333	2897821	3402672	3873311	NA	NA	NA	NA
6	396132	1333217	2180715	2985752	3691712	NA	NA	NA	NA	NA
7	440832	1288463	2419861	3483130	NA	NA	NA	NA	NA	NA
8	359480	1421128	2864498	NA	NA	NA	NA	NA	NA	NA
9	376686	1363294	NA	NA	NA	NA	NA	NA	NA	NA
10	344014	NA	NA	NA	NA	NA	NA	NA	NA	NA

Gráfico 3: Triángulo con cuantías acumuladas. **Fuente:** Elaboración propia

- Triángulo de desarrollo estimado mediante GLM:

Data	GLM Triangle	O.Y. Summary	C.Y. Summary	Graphs						
Full GLM Triangle										
year	1	2	3	4	5	6	7	8	9	10
1	357848	766940	610542	482940	527326	574398	146342	139950	227229	67948
2	352118	884021	933894	1183289	445745	320996	527804	266172	425046	80281.8
3	290507	1001799	926219	1016654	750816	146923	495992	280405	342048.82	77254.41
4	310608	1108250	776189	1562400	272482	352053	206286	222968.3	305808.52	69069.25
5	443160	693190	991983	769488	504851	470639	316032.64	215729.07	295879.67	66826.75
6	396132	937085	847498	805037	705960	336991.09	276263.43	188581.96	258646.49	58417.34
7	440832	847631	1131398	1063269	512838.53	314352.53	257704.47	175913.3	241271.01	54492.95
8	359480	1061648	1443370	826894.3	414702.76	254198.65	208390.65	142250.88	195101.87	44065.29
9	376686	986608	813516.16	732211.74	367217.71	225091.93	184529.12	125962.61	172761.96	39019.64
10	344014	900605.31	827542.3	744836.08	373549.05	228972.83	187710.66	128134.38	175740.61	39692.4

Gráfico 4: Triángulo de desarrollo completo. **Fuente:** Elaboración propia

- Resultados obtenidos a partir del Bootstrapping por año de origen:

Data GLM Triangle O.Y. Summary C.Y. Summary Pred. Distributions										
Summary by Origin Year (Boots)										
OY	Latest	dev.to.date	Ultimate	IBNR	IBNR mean	Pred.Err (Abs)	CV	IBNR quant75	IBNR quant95	IBNR quant99
1	3772717.04	1	3772717.04	0	0	0	0	0	0	0
2	4777195.61	0.98	4857461.85	80281.8	80266.24	1156.97	0.01	80983.35	81986.43	82716.87
3	4797779.89	0.92	5215131.8	419303.23	417351.9	9864.72	0.02	419620.87	422397.18	425127.29
4	4588794.46	0.89	5181030.34	597846.08	592235.88	42075.8	0.07	594505.99	597672.78	599700.46
5	4399240.35	0.83	5282544.05	894468.13	883303.7	81069.7	0.09	885751.61	889602.49	892425.21
6	3823879.74	0.78	4926514.22	1118900.31	1102634.49	110467.78	0.1	1105340.51	1109525.55	1112011.28
7	3400717.12	0.69	4938058.12	1556572.79	1537341	141338.2	0.09	1540772	1545947.38	1549358.15
8	2235565.71	0.52	4321867.36	2085604.39	2086301.66	171454.17	0.08	2091211.43	2098153.78	2101896.31
9	1327510.47	0.33	4010201.66	2660310.89	2682691.2	216862.16	0.08	2689407.22	2699245.24	2704598.91
10	419727.97	0.1	4079617.82	3606783.63	3659889.85	276933.11	0.08	3669473.67	3684293.92	3691590.14
Total	33543128.34	0.72	46585144.26	13020071.24	13042015.92	1049870.55	0.08	13070502.54	13110333.83	13135154.42

Gráfico 5: Resumen de resultados por año de origen. **Fuente:** Elaboración propia

- Resultados a partir del Bootstrapping por año de calendario:

Data GLM Triangle O.Y. Summary C.Y. Summary Pred. Distributions							
Summary by Calendar Year (Boots)							
CY	IBNR	IBNR mean	PredErr Abs	CV	IBNR quant75	IBNR quant95	IBNR quant99
11	4352176.94	4738620.38	340210.52	0.07	4758095.08	4783873.47	4808897.61
12	3163864.77	3441406.64	260867.48	0.08	3457223.34	3482170.15	3499579.89
13	2177487.8	2351813.84	177139.16	0.08	2363163.72	2380487.85	2390463.54
14	1308418.18	1407735.67	114929.21	0.08	1414276.77	1424663.72	1429619.56
15	855441.19	922813.61	78986.03	0.09	927520.53	934452.74	940284.65
16	563268.09	605200.67	48638.63	0.08	609329.16	614366.86	618933.36
17	344961.63	365650.74	23897.14	0.07	368157.67	372311.04	375779.13
18	214760.26	218981.57	5466.14	0.02	221073	224149.19	226598.28
19	39692.4	39663.34	663.15	0.02	40125.81	40737.2	41282.54
Total	13020071.24	14091886.46	1050797.45	0.07	14121766.02	14165164.98	14191794.48

Gráfico 6: Resumen de resultados por año de calendario. **Fuente:** Elaboración propia

- Histograma de las provisiones estimadas a partir de los B remuestreos:

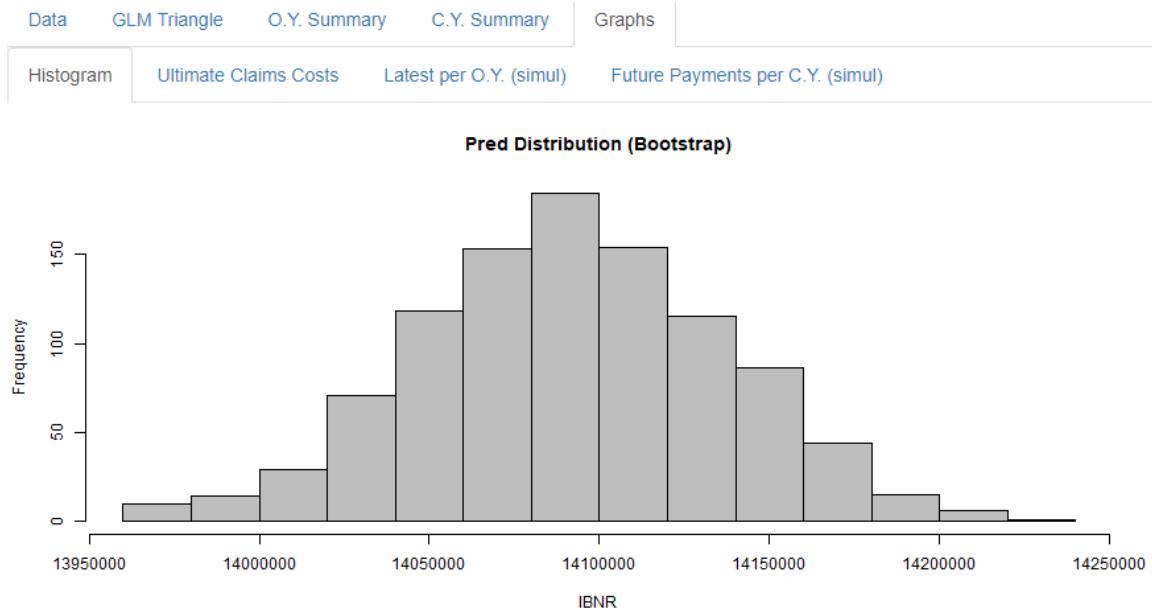


Gráfico 7: Histograma de Distribuciones predictivas. **Fuente:** Elaboración propia

- Costos finales de reclamación simulados a partir del Bootstrapping:

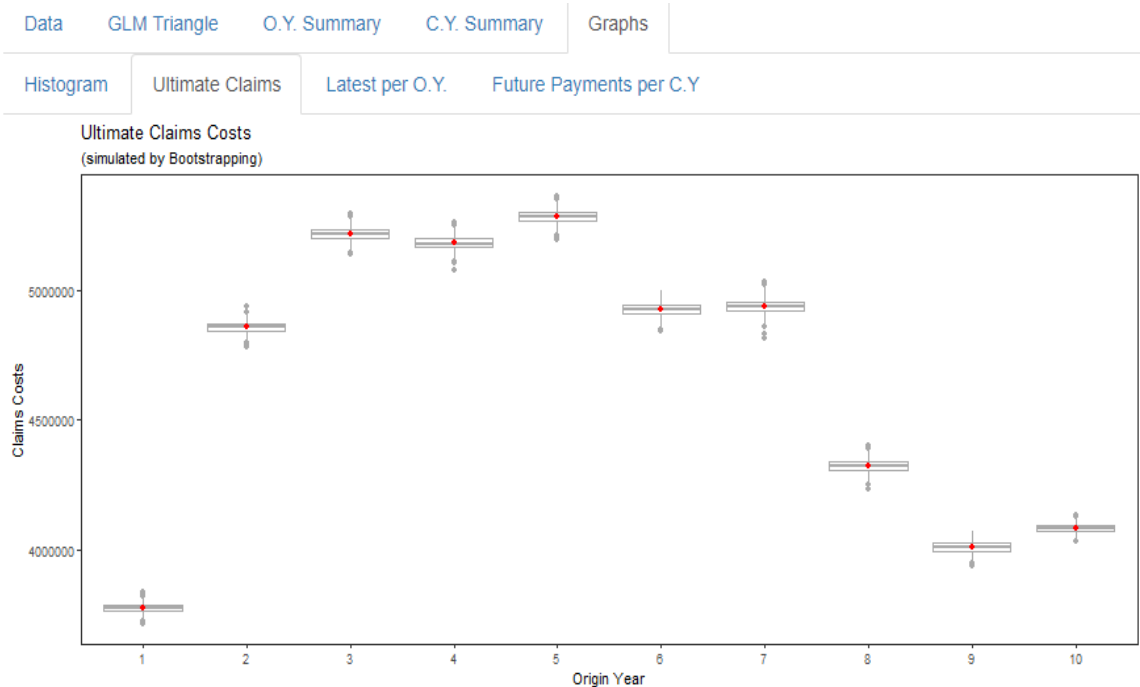


Gráfico 8: Costos finales de reclamaciones por año de origen. **Fuente:** Elaboración propia

- Últimos valores simulados por año de origen:

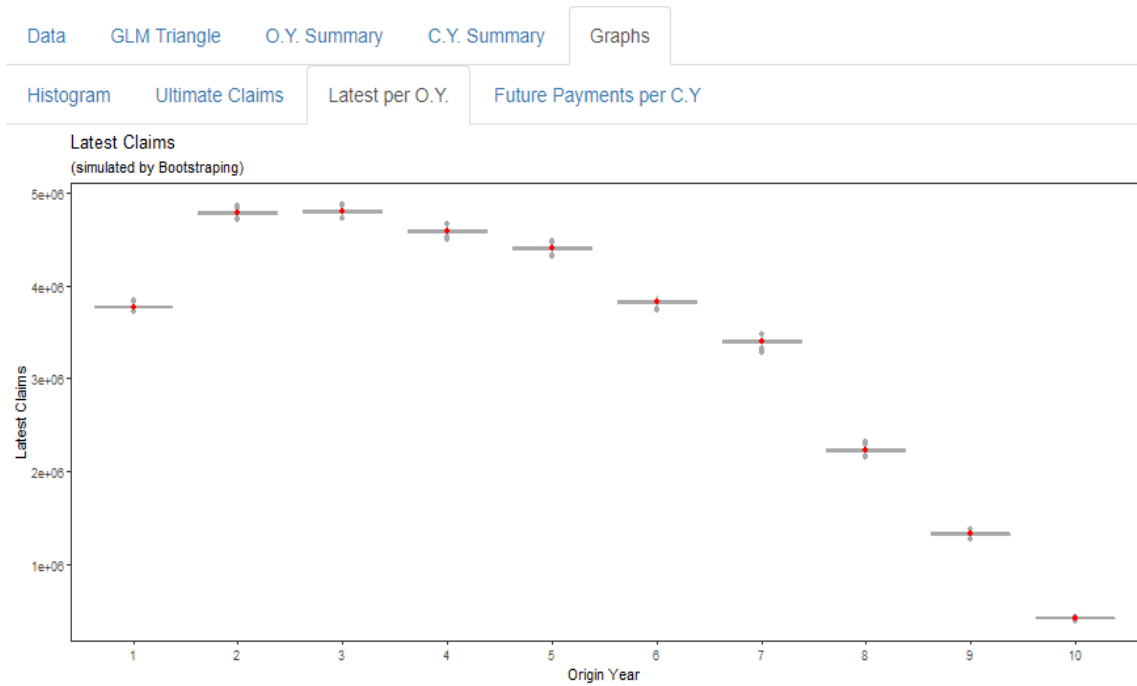


Gráfico 9: Últimas reclamaciones por año de origen. **Fuente:** Elaboración propia

- Pagos futuros simulados por año de calendario a partir del Bootstrapping:

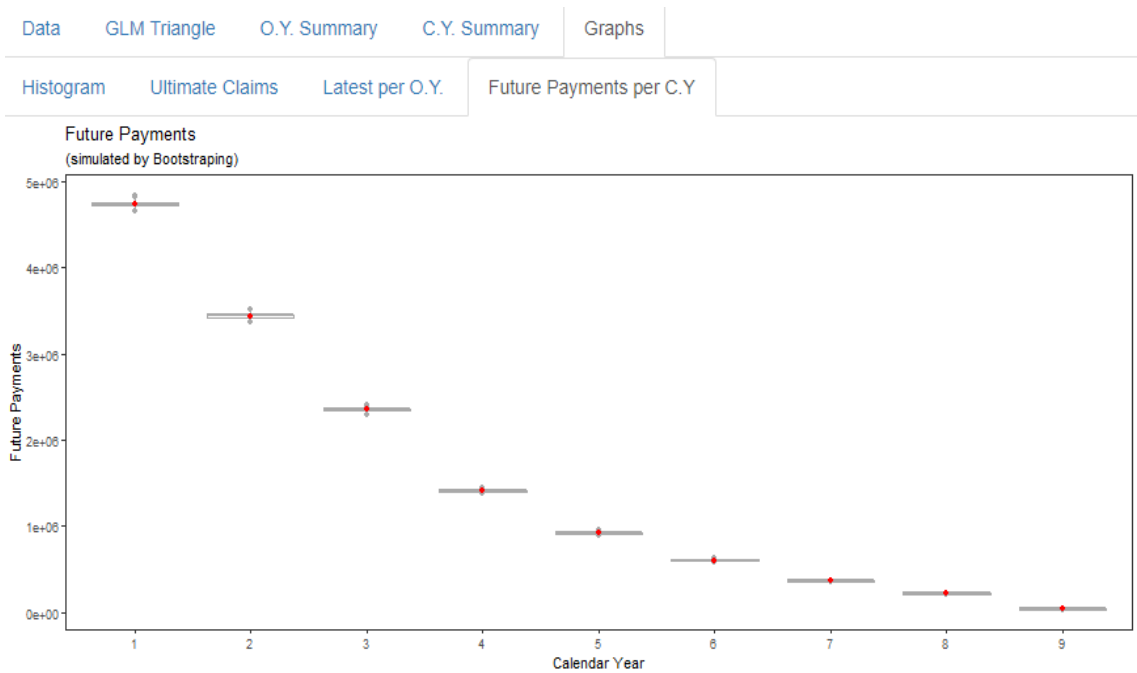


Gráfico 10: Pagos futuros simulados por año de calendario. **Fuente:** Elaboración propia

De forma similar, se presentan los resultados obtenidos cambiando el método Aritmético de separación de Taylor, el efecto de año de calendario al 1% (1000 muestras para *Bootstrapping*, semilla 7):

Data	GLM Triangle	O.Y. Summary	C.Y. Summary	Graphs						
Initial Triangle										
year	1	2	3	4	5	6	7	8	9	10
1	357848	766940	610542	482940	527326	574398	146342	139950	227229	67948
2	352118	884021	933894	1183289	445745	320996	527804	266172	425046	NA
3	290507	1001799	926219	1016654	750816	146923	495992	280405	NA	NA
4	310608	1108250	776189	1562400	272482	352053	206286	NA	NA	NA
5	443160	693190	991983	769488	504851	470639	NA	NA	NA	NA
6	396132	937085	847498	805037	705960	NA	NA	NA	NA	NA
7	440832	847631	1131398	1063269	NA	NA	NA	NA	NA	NA
8	359480	1061648	1443370	NA	NA	NA	NA	NA	NA	NA
9	376686	986608	NA	NA	NA	NA	NA	NA	NA	NA
10	344014	NA	NA	NA	NA	NA	NA	NA	NA	NA

Gráfico 11: Triángulo con cuantías desacumuladas (Ar). **Fuente:** Elaboración propia

Data	GLM Triangle	O.Y. Summary	C.Y. Summary	Graphs						
Summary by Origin Year (Boots)										
OY	Latest	dev.to.date	Ultimate	IBNR	IBNR mean	Pred.Err (Abs)	CV	IBNR quant75	IBNR quant95	IBNR quant99
1	3875594.42	1	3875594.42	0	0	0	0	0	0	0
2	4893707.9	0.98	4974777.14	81084.61	81069.24	1148.17	0.01	81780.92	82776.24	83500.97
3	4864037.38	0.92	5298898.22	434745.92	434860.83	17960.69	0.04	437204.27	439973.57	442594.6
4	4678009.06	0.88	5303963.75	625883.23	625954.68	47745.93	0.08	628295.95	631503.36	633901.18
5	4506673.64	0.82	5477701.73	970967.72	971028.09	84156.66	0.09	973881.89	977784.58	981218.92
6	3955003.17	0.76	5198883.92	1243791.98	1243880.75	118670.64	0.1	1247010.64	1251567.46	1254464.2
7	3543851.03	0.67	5271775.44	1727864.12	1727924.41	162896.8	0.09	1732050.36	1737658.13	1741479.04
8	2311932.64	0.5	4612869.19	2300957.38	2300936.55	223693.71	0.1	2306710.5	2314804.28	2321391.35
9	1369005.58	0.32	4321753.93	2952944.88	2952748.35	297612.17	0.1	2960328.69	2971973.36	2979452
10	441325.77	0.1	4430280.46	3989285.81	3988954.7	398837.29	0.1	4000527.29	4016855.15	4028309.67
Total	34439140.6	0.71	48766498.2	14327525.67	14327357.6	1351569.01	0.09	14360471.06	14406780.08	14438318.77

Gráfico 12: Resumen de resultados por año de origen (Ar). **Fuente:** Elaboración propia

Summary by Calendar Year (Boots)

CY	IBNR	IBNR mean	PredErr Abs	CV	IBNR quant75	IBNR quant95	IBNR quant99
11	4712465.19	5176207.53	464034.67	0.09	5196362.73	5225806.49	5249566.66
12	3479143.14	3823849.83	344875.39	0.09	3841281.95	3867583.05	3886411.7
13	2406598.93	2634337.42	227727.3	0.09	2647173.29	2665857.32	2677133.81
14	1463908.24	1598241.55	134271.13	0.08	1605634.46	1617327.8	1623047.05
15	969186.3	1057943.64	88675.78	0.08	1063451.2	1071433.31	1077869.2
16	634082.64	687325.33	53200.9	0.08	691908.25	697729.21	702769.8
17	380408.58	409079.63	28630.09	0.07	411883.22	416423.71	420264.45
18	238321.65	248560.97	10205.91	0.04	250931.33	254417.59	257114.55
19	43410.99	43380.7	723.25	0.02	43876.83	44533.73	45119.6
Total	14327525.67	15678926.61	1352344.43	0.09	15711817.8	15756938.07	15788652.45

Gráfico 13: Resumen de resultados por año de calendario (Ar). **Fuente:** Elaboración propia

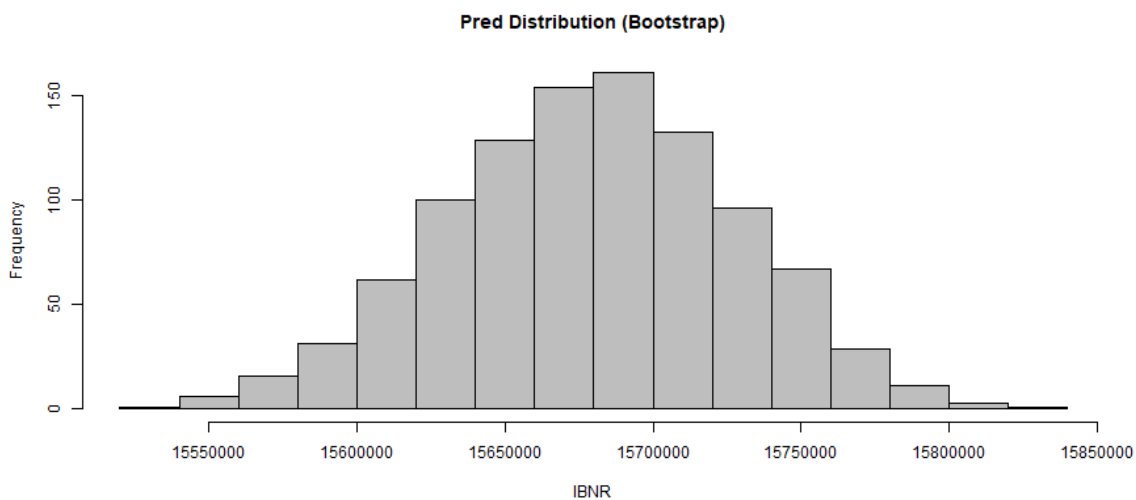


Gráfico 14: Histograma de Distribuciones predictivas (Ar). **Fuente:** Elaboración propia

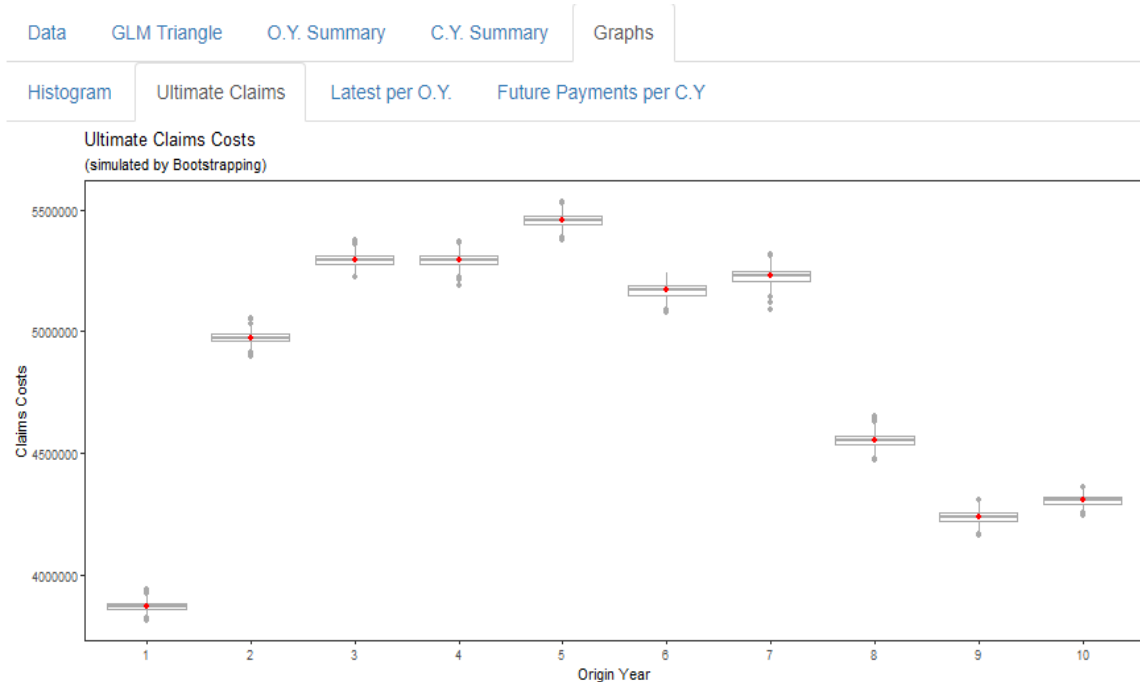


Gráfico 15: Costos finales de reclamaciones por año de origen (Ar). **Fuente:** Elaboración propia

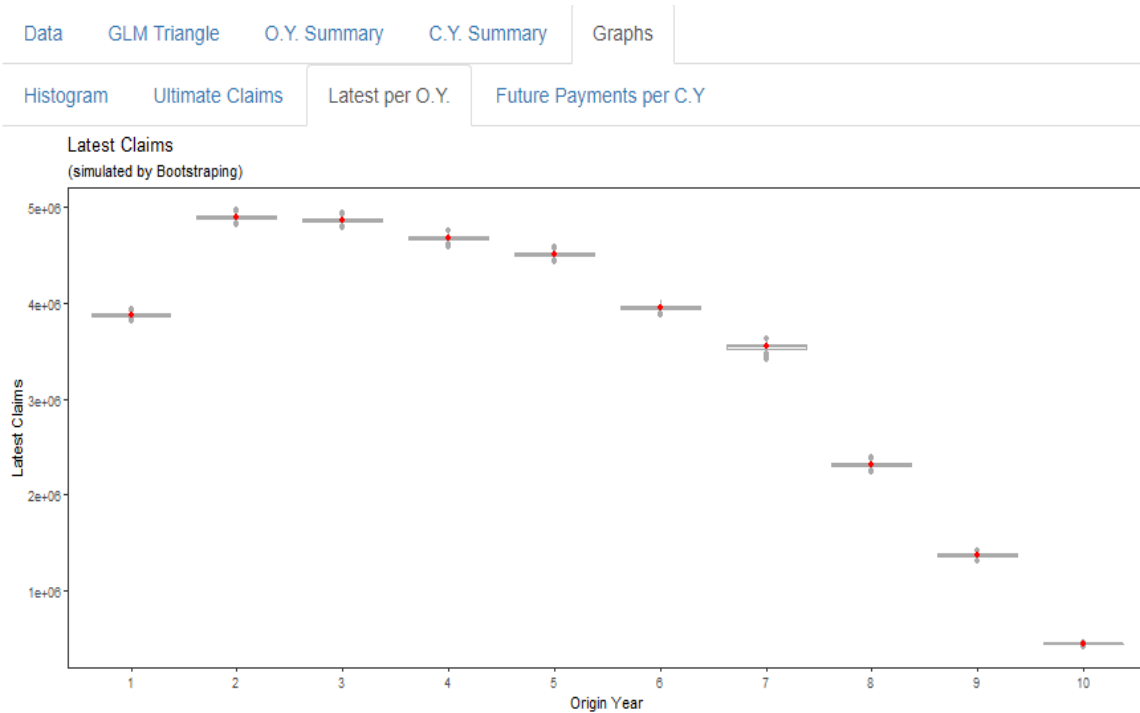


Gráfico 16: Últimas reclamaciones por año de origen (Ar). **Fuente:** Elaboración propia

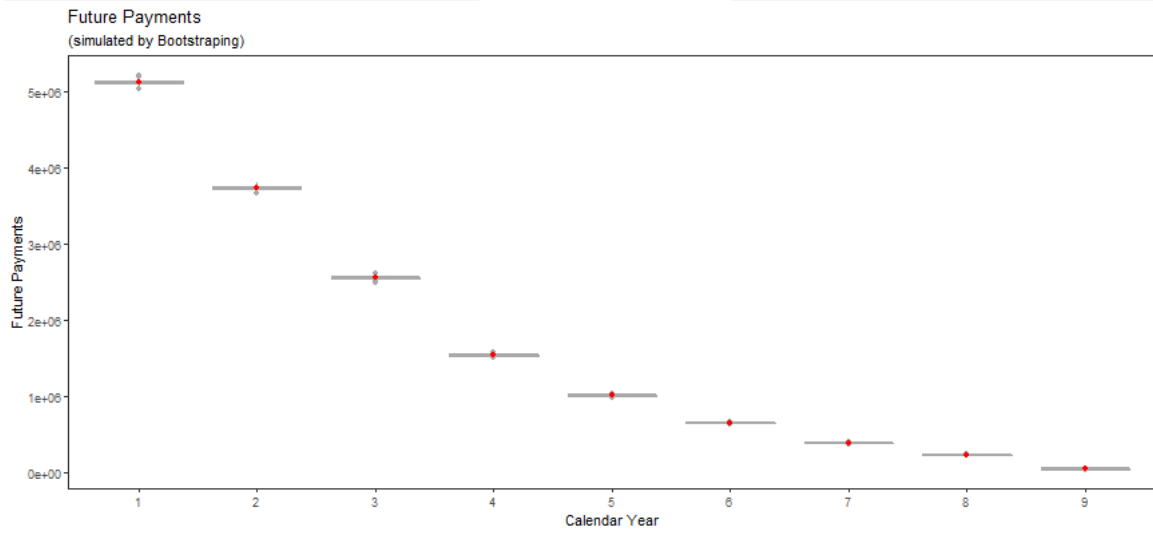


Gráfico 17: Pagos futuros simulados por año de calendario. **Fuente:** Elaboración propia

5. Conclusiones

El desarrollo de una aplicación interactiva para el cálculo estocástico de provisiones permite una manipulación inmediata de las condiciones iniciales haciendo que la visualización de los resultados se presente de manera más dinámica e integrada, al tiempo que se consolida como una herramienta complementaria para el aprendizaje y obtención de los resultados de los métodos de separación de Taylor mediante el uso del GLM.

Los métodos de separación de Taylor adquieren relevancia al incluir en la estimación de las provisiones futuras un parámetro para el efecto por el año de calendario, debido a que este tipo de factores externos inciden en la valoración de las provisiones, y no son tenidos en cuenta por otro tipo de métodos deterministas.

Futuras líneas de desarrollo en aplicaciones dinámicas para la estimación y valoración de las provisiones pueden incluir características adicionales como curvas de tipo de interés que conducirán a escenarios más realistas. De igual forma, la inclusión del cálculo de resultados del desarrollo de las reclamaciones-CDR, permitirá analizar la evolución de las provisiones en el corto plazo. A nivel de empresa, estas aplicaciones se han consolidado como una importante herramienta informática a la hora de presentar resultados de una forma más entendible en contextos más generales.

Bibliografía

- Beeley, C., (2013). *Web Application Development with R Using Shiny*. Packt Publishing, Birmingham.
- Björkwall, S., (2011). *Stochastic claims reserving in non-life insurance*. Bootstrap and smoothing models. Department of Mathematics, Stockholm University, Stockholm.
- Björkwall, S., Hössjer, O., Ohlsson, E., (2009). *Non-parametric and parametric bootstrap techniques for age-to-age development factor methods in stochastic claims reserving*. Scandinavian Actuarial Journal 4, 306–331.
- Björkwall, S., Hössjer, O., Ohlsson, E., (2010). *Bootstrapping the separation method in claims reserving*. Astin Bulletin 40:2, 845–869.
- Björkwall, S., Hössjer, O., Ohlsson, E., Verrall, R., (2011). *A generalized linear model with smoothing effects for claims reserving*. Insurance: Mathematics and Economics 49:1, 27–37.
- Boj, E., Costa, T., (2014). *Modelo lineal generalizado y cálculo de la provisión técnica*. Colección OMADO. Depósito digital de la Universidad de Barcelona. [URL:http://hdl.handle.net/2445/49068](http://hdl.handle.net/2445/49068).
- Boj, E., Costa, T. (2017). *Provisions for claims outstanding, incurred but not reported, with generalized linear models: prediction error formulated according to calendar year*. Cuadernos de gestión 17:2, 157-174.
- Boj, E., Claramunt, M.M., Costa, T. (2020). *Tarifificación y provisiones*. Tercera edición. Depósito Digital de la Universidad de Barcelona. Colección de objetos y materiales docentes (OMADO). [URL:http://hdl.handle.net/2445/149241](http://hdl.handle.net/2445/149241)
- Canadell, A., Boj, E., Costa, T. (2018). *provision: statistical models for claim reserving*. R package version 0.1.0. <https://github.com/acanadis/provision/>
- Charpentier, A., (2014). *Computational actuarial science with R*. Chapman and Hall, New York.
- Efron, B., Tibshirani, R. J., (1993). *An introduction to the bootstrap*. NY: Chapman & Hall.
- England, P., Verrall, R., (1999). *Analytic and bootstrap estimates of prediction errors in claims reserving*. Insurance: Mathematics and Economics 25, 281–293.
- England, P.D., Verrall, R.J., (2002). *Stochastic claims reserving in general insurance*. British Actuarial Journal 8, 443-518.
- Fox, J., (2016). *Applied Regression Analysis and Generalized Linear Models*. McMaster University. Third Edition. SAGE publications, Inc, CA, USA.
- Gesmann, M., Murphy, D., Zhang, Y., Carrato, A., Crupi, G., Dutang, C., Lacoume, A., Charpentier, A., Wüthrich, M., Concina, F., (2021). *ChainLadder: Statistical*

methods and models for claims reserving in general insurance. R package v. 0.2.12.
URL: <http://cran.r-project.org/package=ChainLadder>

Grolemund, G., (2015). *Shiny Tutorial*. URL: <https://shiny.rstudio.com/tutorial/>

Hindley, D., (2018). *Claims reserving in general insurance*. Cambridge University Press, Cambridge.

Kaas, R., Goovaerts, M., Dhaene, J., Denuit, M., (2008). *Modern actuarial risk theory: using R*. Second edition. Springer-Verlag, Heidelberg.

Mack, T., (1993). *Distribution free calculation of the standard error of chain ladder reserve estimates*. *Astin Bulletin* 23:2, 213–225.

Mulero, J., (2016). *Aplicaciones Interactivas Diseñadas con Shiny*. Colección de Docencia del Repositorio Institucional de la Universidad de Alicante. <http://hdl.handle.net/10045/54325>.

Nelder, J.A., Wedderburn, W.M., (1972). *Generalized Linear Models*. *Journal of the Royal Statistical Society. Series A (General)*. Vol. 135, No. 3, 370-384.

Plot.ly (2021). *Plotly R Open Source Graphing Library.pdf*. URL: <https://plot.ly/r/>

R Development Core Team (2021) *R: a language and environment for statistical computing*. Vienna, Austria. URL: <http://www.R-project.org/>

Sievert, C., Parmer, C., Hocking, T., Chamberlain, S., Ram, K., Corvellec, M., Despouy, P., (2021). *plotly: Create Interactive Web Graphics via 'plotly.js'*. R package version 4.9.3 URL: <https://CRAN.R-project.org/package=plotly>.

ANEXOS: Programación en RStudio

```
install.packages("devtools")
install.packages("usethis")
install.packages("ChainLadder")
install.packages("statmod")
install.packages("shiny")
install.packages("shinydashboard")
install.packages("ggplot2")
install.packages("dplyr")
library(dplyr)
library(ggplot2)
library(shiny)
library(shinydashboard)
library(devtools)
library(usethis)
library(ChainLadder)
library(statmod)
```

```
glmTaylorsep <- function(incraccum=1, metodo= "Geometric", lambdaini = 0, B = 1000,
seed = 7){
```

```
  lambdaini<-lambdaini/100
```

```
  #1 obtener vector con número de siniestros, Talor & Ashe 1983
```

```
  Nij <- c(40, 124, 157, 93, 141, 22, 14, 10, 3, 2,
    37, 186, 130, 239, 61, 26, 23, 6, 6,
    35, 158, 243, 153, 48, 26, 14, 5,
    41, 155, 218, 100, 67, 17, 6,
    30, 187, 166, 120, 55, 13,
    33, 121, 204, 87, 37,
    32, 115, 146, 103,
    43, 111, 83,
    17, 92,
    22)
```

```
  TT <- trunc(sqrt(2*length(Nij)))
```

```
  i <- rep(1:TT,TT:1)
```

```
  j <- sequence(TT:1)
```

```
  #matrices de Ni:
```

```
  Nij.mat.cum <- Nij.tri <- matrix(NA, nrow=TT, ncol=TT)
```

```
  for (k in 1:length(Nij)) Nij.tri[i[k],j[k]] <- Nij[k]
```

```
  for (k in 1:TT) Nij.mat.cum[k,] <- cumsum(Nij.tri[k,])
```

```
  #paso 1: MUNDO REAL
```

```
  #paso 1.1 triángulo de número de siniestros
```

```

Nij.mat.cum.tri <- as.triangle(Nij.mat.cum)
glmtriN <- glmReserve(Nij.mat.cum.tri,mse.method = "formula")

#VECTOR DE SINIESTROS POR AÑO DE ORIGEN
Ni <- as.vector(glmtriN$FullTriangle[,10])
Ni.mat <- matrix(Ni,TT,TT)
Ni.ao <- rep(Ni, TT:1)

#2 Cuantías de siniestros, Taylor & Ashe

Xij.mat.acum <- Xij.mat <- Xij.prom.mat <- Xij.prom <- matrix(NA,nrow =
nrow(GenIns), ncol = ncol(GenIns))

Xij.mat.acum[,] <- GenIns[,]
colnames(Xij.mat.acum )<-c(1:10)
row.names(Xij.mat.acum )<-c(1:10)
#names(dimnames(Xij.mat.acum)) <- c("Orig.Year", "Dev.Year")

for (i in 2:10) {
  Xij.mat[,i] <- GenIns[,i]-GenIns[,i-1]
}
Xij.mat[,1] <- GenIns[,1]

colnames(Xij.mat)<-c(1:10)
row.names(Xij.mat)<-c(1:10)
#names(dimnames(Xij.mat)) <- c("Orig.Year", "Dev.Year")

Xij<- as.vector(t(Xij.mat))
Xij<- Xij[!is.na(Xij)]

rm(TT,i,j)

TT <- trunc(sqrt(2*length(Xij)))
i <- rep(1:TT,TT:1)
j <- sequence(TT:1)

#matriz cuantías medias
Xij.prom.mat <- Xij.mat/Ni.mat
Xij.prom <- as.vector(t(Xij.prom.mat))
Xij.prom <- Xij.prom[!is.na(Xij.prom)]

if (incraccum==1){triangu=Xij.mat}

if (incraccum==2){triangu=Xij.mat.acum}

#3 GLM PARA CÁLCULO DE RESULTADOS POR MÉTODOS DE
SEPARACIÓN
#paso 1.2 estimar triángulo de reclamaciones con métodos de separación
ao <- rep(1:TT,TT:1)

```

```

ad <- sequence(TT:1)
ac <- ao+ad-1
ao <- as.factor(ao)
ad <- as.factor(ad)
ac <- as.numeric(ao) + as.numeric(ad)

```

#3.1 SEP ARITMÉTICA

```

if(metodo == "Arithmetic"){
  glm.Aritm <- glm(Xij.prom ~ ad + ac, family = statmod::tweedie(var.power = 1,
link.power = 0))

```

```

Xij.prom.fit.mat <- matrix(0, TT,TT)
for (i in 1:TT){
  Xij.prom.fit.mat[i, ] <- c(glm.Aritm$fitted.values[ao == i],rep(times = i-1, NA))
}
#triáng-matrix datos ajustados, multip por Ni
Xij.fitted <- Xij.prom.fit.mat * Ni

```

```

triangulo.glm <- matrix(0, TT, TT)
for (i in 2:TT){
  tal <- glm.Aritm$fitted.values[((ao == (TT-i+1)) & (ad == i))] * Ni[(TT-i+2):TT] *
  (1+lambdaini)^(1:(i-1))
  triangulo.glm[, i] <- c(Xij.mat[1:(TT-i+1), i], tal)
  rm(tal)
}
#TRIÁNGULO FINAL COMPLETO
triangulo.glm[, 1] <- Xij.mat[, 1]
colnames(triangulo.glm) <- c(1:TT)
row.names(triangulo.glm) <- c(1:TT)
#names(dimnames(triangulo.glm)) <- c("Orig.Year", "Dev.Year")
}

```

#3.2 SEPARACIÓN GEOMÉTRICA

```

if(metodo == "Geometric"){
  #distr norm sobre log de las cuantías medias, enlace 1
  glm.Geomet <- glm(log(Xij.prom) ~ ad + ac, family = statmod::tweedie(var.power =
0,link.power = 1))

```

```

Xij.prom.fit.mat <- matrix(0, TT, TT)
for (i in 1:TT){
  Xij.prom.fit.mat[i, ] <- c(glm.Geomet$fitted.values[ao == i], rep(times = i-1, NA))
}

```

```

Xij.fitted <- exp(Xij.prom.fit.mat) * Ni

```

```

triangulo.glm <- matrix(0, TT, TT)

```

```

for (i in 2:TT){
  tal <- exp(glm.Geomet$fitted.values[((ao == (TT-i+1)) & (ad == i)]) * Ni[(TT-
i+2):TT] *
  (1 + lambdaini)^(1:(i-1))
  triangulo.glm[, i] <- c(Xij.mat[1:(TT-i+1), i], tal)
  rm(tal)
}

#TRIÁNGULO FINAL COMPLETO
triangulo.glm[, 1] <- Xij.mat[, 1]
colnames(triangulo.glm) <-c (1:TT)
row.names(triangulo.glm) <-c (1:TT)
#names(dimnames(triangulo.glm)) <- c("Orig.Year", "Dev.Year")
}

#paso 1.2: CÁLCULO DE PHI
if (metodo == "Arithmetic"){
  phi <- (sum(glm.Aritm$residuals^2/glm.Aritm$df.residual))
}
if (metodo == "Geometric"){
  phi <- (sum(glm.Geomet$residuals^2/glm.Geomet$df.residual))
}

#4 RESERVAS POR AÑO DE ORIGEN Y TOTAL
#paso 1.2 Cálculos de reservas por año de origen y totales

resanyo <- rep(0, TT-1)
for (i in 2:TT){
  resanyo[i-1] <- sum(triangulo.glm[i, (TT-i+2):TT])
}
#TOTAL-restotal=totres
restotal <- sum(resanyo)

#VECTOR PAGOS FUTUROS
vecpagos <- rep(0, dim(triangulo.glm)[1] - 1)
for (v in 1:dim(triangulo.glm)[1] - 1) {
  diagtf <- row(triangulo.glm) + col(triangulo.glm) - 1 == dim(triangulo.glm)[1] + v
  vecpagos[v] <- sum(triangulo.glm[diagtf])
}

#datos reserva total y vpf en matrix
resanyo.matrix <- matrix(data = resanyo, B, TT-1, byrow = TRUE)
vpf.matrix <- matrix(data=vecpagos, B, TT-1, byrow = TRUE)

#5 CÁLCULO DE LAMBDA Y R - parámetros de incidencia por año de desarrollo y
de calendario
#Hipótesis de factor de incidencia establecido en lambdaini

# 5.1. Por Separación Aritmética

```



```

if (metodo == "Arithmetic"){
  sij<-Xij.prom.mat

  suma.col <- colSums(sij, na.rm = TRUE)

  suma.diag <- rep(0, TT)
  for (k in 1:TT){
    diagonalf <- row(sij) + col(sij) - 1 == k
    suma.diag[k] <- sum(sij[diagonalf])
  }
  rm(diagonalf)
  lambda <- rep(0, TT)
  r <- rep(0, TT)
  lambda[TT] <- suma.diag[TT]
  r[TT] <- suma.col[TT]/lambda[TT]
  for (i in 1:(TT-1)){
    lambda[TT-i] <- suma.diag[TT-i]/(1-sum(r[(TT-i+1):TT]))
    r[TT-i] <- suma.col[TT-i]/sum(lambda[(TT-i):TT])
  }
  lambdainf <- lambda[TT] * (1+lambdaini)^(1:(TT-1))
  lambdatotal <- c(lambda, lambdainf)
}

```

5.2. Por Separación Geométrica

```

if (metodo == "Geometric"){
  #traer matrix de cuantías medias
  sij<-Xij.prom.mat

  prod.col <- rep(0, TT)
  for (k in 1:TT){
    prod.col[k] <- prod(sij[(1:(TT+1-k)), k])
  }

  prod.diag <- rep(1, TT)
  for (k in 1:TT){
    diagtf <- row(sij) + col(sij) - 1 == k
    prod.diag[k] <- prod(sij[diagtf])
  }

  lambda <- rep(0, TT)
  r <- rep(0, TT)
  lambda[TT] <- prod.diag[TT]^(1/TT)
  r[TT] <- prod.col[TT]/lambda[TT]
  for (i in 1:(TT-1)){
    lambda[TT-i] <- (prod.diag[TT-i]*prod(r[(TT-i+1):TT]))^(1/(TT-i))
    r[TT-i] <- (prod.col[TT-i]/prod(lambda[(TT-i):TT]))^(1/(i+1))
  }
  lambdainf <- lambda[TT] * (1+lambdaini)^(1:(TT-1))
  lambdatotal <- c(lambda, lambdainf)
}

```

```
}
```

#PASO 2: MUNDO BOOTSTRAPPING

6. Bootstrapping a partir de lambdas obtenidos

```
lambda.mat <- matrix(0, TT, TT)
for (i in 1:TT){
  lambda.mat[, i] <- lambdatotal[i:(TT+i-1)]
}
r.vector <- rep(r, each = TT)
r.mat <- matrix(r.vector, TT, TT)
cij.boot <- array(dim = c(TT, TT, B), data = 0)
```

#paso 2.1

#obtener B pseudo-triángulos superiores con cuantías de siniestros bajo una distribución Gamma

```
for (i in 1:TT){
  for (j in 1:TT){
    cij.boot[i, j, ] <- rgamma(B, Ni.mat[i, j]/phi, 1/(r.mat[i,j] * lambda.mat[i,j] * phi))
  }
}
```

TRIANGULOS CON CUANTIAS iniciales OBTENIDOS POR BOOTSTRAP

```
tri.sup.boot <- array(dim = c(TT, TT, B), data = 0)
for (boots in 1:B){
  if(! is.null(seed)){
    set.seed(seed)
  }
  for(i in 1:TT){
    tri.sup.boot[i, , boots] <- c(cij.boot[i, 1:(TT-i+1), boots],
                                rep(times = i-1, NA))
  }
}
```

#paso 2.2

```
tri.inf.boot <- array(dim = c(TT, TT, B), data = NA)
for (boots in 1:B){
  if(! is.null(seed)){
    set.seed(seed)
  }
  for (i in 2:TT){
    tri.inf.boot[i, , boots] <- c(rep(times = (TT-i+1), NA), cij.boot[i, (TT-i+2):TT,
boots])
  }
}
```

```

boot.mat <- matrix(0, TT, TT)
boot.tal <- matrix(NA, TT, TT)
tri.glm.boot <- array(data = 0, dim = c(TT, TT, B)) #arreglo para almacenar triángulos
inferiores

```

```
#paso 2.2
```

```
# 6.1 Separación Aritmética con triángulos obtenidos por Bootstrapping
```

```

if (metodo == "Arithmetic"){
  for (boots in 1:B) {
    if(! is.null(seed)){
      set.seed(seed)
    }
    boot.mat <- t(tri.sup.boot[, , boots])
    boot.vector <- as.vector(boot.mat)
    boot.vector <- boot.vector[!is.na(boot.vector)]

    boot.prom <- boot.vector/Ni.ao

    glm.Aritm.boot <- glm(boot.prom ~ ad + ac, family = statmod::tweedie(var.power =
1, link.power = 0))

    for (i in 2:TT){
      tal <- glm.Aritm.boot$fitted.values[((ao == (TT-i+1)) & (ad == i))] * Ni[(TT-
i+2):TT] *
      (1+lambdaini)^(1:(i-1))
      tri.glm.boot[, i, boots] <- c(boot.tal[1:(TT-i+1), i], tal)
      rm(tal)
    }
    tri.glm.boot[, 1, boots] <- boot.tal[, 1]
  }
}

```

```
# 6.2 Separación Geométrica con triángulos Boots
```

```
#paso 2.2
```

```

if (metodo == "Geometric"){
  for (boots in 1:B) {
    if(! is.null(seed)){
      set.seed(seed)
    }
    boot.mat <- t(tri.sup.boot[, , boots])
    boot.vector <- as.vector(boot.mat)
    boot.vector <- boot.vector[!is.na(boot.vector)]

    boot.prom <- boot.vector/Ni.ao

    glm.Geomet.boot <- glm(log(boot.prom) ~ ad + ac, family =
statmod::tweedie(var.power = 1,link.power = 0))

```

```

for (i in 2:TT){
  tal <- exp(glm.Geomet.boot$fitted.values[((ao==(TT-i+1))&(ad==i))]) * Ni[(TT-
i+2):TT] *
  (1+lambdaini)^(1:(i-1))
  tri.glm.boot[, i, boots] <- c(boot.tal[1:(TT-i+1), i], tal)
  rm(tal)
}
tri.glm.boot[, 1, boots] <- boot.tal[,1]
}
}

```

#PASO 3: ANÁLISIS DE LAS SIMULACIONES # 7 ERRORES DE PREDICCIÓN

```

resanyorboot.mat <- matrix(0, B, TT-1)
restotal.boot <- rep(0, times = B)
vecpagos.boot <- matrix(0, B, TT-1)

```

```

for (boots in 1:B){
  if(! is.null(seed)){
    set.seed(seed)
  }

```

```

for (i in 2:TT){
  resanyorboot.mat[boots, i-1] <- sum(tri.glm.boot[i, (TT-i+2):TT, boots])
}
# Total reserve
restotal.boot[boots] <- sum(resanyorboot.mat[boots, ])

```

```

g <- ncol(tri.glm.boot) + 1
vpfaux <- NULL
for (z in 1:(nrow(tri.glm.boot)-1)){
  au <- 0
  for (i in 1:nrow(tri.glm.boot)){
    for (j in 1:ncol(tri.glm.boot)){
      if ((i+j-1) == g){
        au <- sum(au, tri.glm.boot[i, j, boots], na.rm = TRUE)
      }
    }
  }
  vpfaux[z] <- au
  g <- g + 1
}
vecpagos.boot[boots, ] <-vpfaux
}

```

```

# Obtener pagos futuros con bootstrap
resanyorfut.boot <- matrix(0, B, TT-1)
restotalfut.boot <- rep(0, times = B)

```

```

vecpagosfut.boot <- matrix(0, B, TT-1)

for (boots in 1:B){
  if(! is.null(seed)){
    set.seed(seed)
  }
  tri.boot <- tri.inf.boot[, , boots]

  for (i in 2:TT){
    resanyorfut.boot[boots, i-1] <- sum(tri.boot[i, (TT-i+2):TT])
  }

  restotalfut.boot[boots] <- sum(resanyorfut.boot[boots, ])

  a <- ncol(tri.inf.boot) + 1
  vpfutaux <- NULL
  for (z in 1:(nrow(tri.inf.boot)-1)){
    aux <- 0
    for (i in 1:nrow(tri.inf.boot)){
      for (j in 1:ncol(tri.inf.boot)){
        if ((i+j-1) == a){
          aux <- sum(aux, tri.inf.boot[i, j, boots], na.rm = TRUE)
        }
      }
    }
    vpfutaux[z] <- aux
    a <- a + 1
  }
  vecpagofut.boot[boots, ] <- vpfutaux
  rm(aux)
}

#vecpagosfut.boot.df
vecpagosfut.boot.df <- matrix(vecpagosfut.boot, ncol=ncol(vecpagosfut.boot), nrow =
nrow(vecpagosfut.boot))

aux.vpfb <- reshape2::melt(vecpagosfut.boot.df)
colnames(aux.vpfb) <- c("B", "cy", "vpfb")
aux.vpfb$cy <- aux.vpfb$cy
aux.vpfb$cy <- factor(aux.vpfb$cy)
aux.vpfb.cy <- aux.vpfb %>% dplyr::group_by(cy) %>%
dplyr::summarise(mu=mean(vpfb))

#CÁLCULO DE ERRORES DE PREDICCIÓN

erp.orig <- erp.vecpagfut <- matrix(0, B, TT-1)

for (boots in 1:B){
  if(! is.null(seed)){

```

```

    set.seed(seed)
  }
  for(i in 1:(TT-1)){
    erp.orig[boots, ] <- (resanyorfut.boot[boots, ] - resanyorboot.mat[boots, ])
    erp.vecpagfut[boots, ] <- vecpagosfut.boot[boots, ] - vecpagos.boot[boots, ]
  }
}
erp.total <- restotalfut.boot - restotal.boot

```

```

#paso 3
#Distribuciones predictivas
resanyo.pred <- resanyo.matrix + erp.orig
vpfut.pred <- vpf.matrix + erp.vecpagfut
restotal.pred <- restotal + erp.total

```

8. Resultados

```

#Resumen por año de origen
out.sum <- matrix(NA, ncol = 10, nrow = TT+1,
  dimnames = list(c(rownames(Xij.mat), "O.Y. TOTAL"),
    c("Latest", "dev.to.date", "Ultimate",
      "IBNR", "IBNR mean", "Pred.Err (Abs)", "CV",
      "IBNR quant75", "IBNR quant95",
      "IBNR quant99"))))

```

```

aux <- array(0, c(TT, TT, B))
auxup <- tri.sup.boot; auxdn <- tri.glm.boot
auxup[is.na(auxup)] <- 0; auxdn[is.na(auxdn)] <- 0
for (i in 1:B){
  aux[, , i] <- auxup[, , i] + auxdn[, , i]
}
rm(auxup, auxdn, i)

```

```

increm.tri <- array(0, c(TT, TT, B))
for (i in 1:B){
  increm.tri[, , i] <- t(apply(aux[, , i], 1, cumsum))
}

```

```

## Latest
diagon <- row(increm.tri[, , 1]) + col(increm.tri[, , 1]) - 1 == nrow(increm.tri)
latest <- matrix(increm.tri[diagon], ncol = nrow(Xij.mat), nrow = B, byrow=TRUE)
latest.df <- matrix(rev(latest), ncol=ncol(latest), nrow = nrow(latest))

```

```

aux.lat <- reshape2::melt(latest.df)
colnames(aux.lat) <- c("B", "oy", "latest")
aux.lat$oy <- aux.lat$oy
aux.lat$oy <- factor(aux.lat$oy)

```

```

aux.lat.oy <- aux.lat %>% dplyr::group_by(oy) %>%
dplyr::summarise(mu=mean(latest))

out.sum[, 1] <- c(rev(colMeans(latest)), sum(colMeans(latest)))
rm(i)
#rm(latest, i, diagon)

## Ultimate
ultimate <- NULL
for (i in 1:nrow(aux)){
  ultimate <- c(ultimate, mean(increm.tri[i, ncol(increm.tri), ]))
}
out.sum[, 3] <- c(ultimate, sum(ultimate))
#rm(ultimate, i)

##ultimate.df
ultimate.df <- matrix(0, ncol = nrow(increm.tri),
  nrow = dim(increm.tri)[3])
for (i in 1:dim(increm.tri)[3]){
  ultimate.df[i, ] <- increm.tri[, ncol(increm.tri), i]
}

aux.ult <- reshape2::melt(ultimate.df)
colnames(aux.ult) <- c("B", "oy", "ultimate")
aux.ult$oy <- aux.ult$oy
aux.ult$oy <- factor(aux.ult$oy)
aux.ult.oy <- aux.ult %>% dplyr::group_by(oy) %>%
dplyr::summarise(mu=mean(ultimate))

## IBNR
out.sum[, 4]<-c(0, resanyo, sum(resanyo))

## IBNR medio
out.sum[, 5] <- c(0, colMeans(resanyorboot.mat), mean(restotal.boot))

## deviation to date
out.sum[, 2] <- out.sum[, 1]/out.sum[, 3]

## PE
out.sum[, 6] <- c(0, colMeans(abs(erp.orig)), mean(abs(erp.total)))

## CV
out.sum[, 7] <- out.sum[, 6]/out.sum[, 4]

#c(0, colMeans(abs(erp.orig)), mean(abs(erp.total))) / c(0, resanyo, sum(resanyo))

#Cuantiles

```

```

out.sum[, 8] <- c(0, apply(resanyorboot.mat, 2, quantile, 0.75), quantile(restotal.boot,
0.75))
out.sum[, 9] <- c(0, apply(resanyorboot.mat, 2, quantile, 0.95), quantile(restotal.boot,
0.95))
out.sum[, 10] <- c(0, apply(resanyorboot.mat, 2, quantile, 0.99), quantile(restotal.boot,
0.99))

out.sum[is.nan(out.sum)] <- 0

#Resumen por año de calendario
labelac <- paste0( (TT+1):(TT+ncol(Xij.mat)-1))
out.sum2 <- matrix(NA, ncol = 7, nrow = TT,
dimnames = list(c(labelac, "C.Y. TOTAL "),
c("IBNR", "IBNR mean", "PredErr Abs", "CV",
"IBNR quant75", "IBNR quant95",
"IBNR quant99")))

out.sum2[, 1] <- c(vecpagos, sum(vecpagos))

out.sum2[, 2] <- c(apply(vecpagosfut.boot,2,mean),
sum(apply(vecpagosfut.boot,2,mean)))

out.sum2[, 3] <- c(apply(abs(erp.vecpagfut),2,mean),
sum(apply(abs(erp.vecpagfut),2,mean)))

out.sum2[, 4] <- out.sum2[,3]/out.sum2[,2]

out.sum2[, 5] <- c(apply(vecpagosfut.boot, 2, quantile, 0.75, na.rm = TRUE),
quantile(restotalfut.boot, 0.75, na.rm = TRUE))

out.sum2[, 6] <- c(apply(vecpagosfut.boot, 2, quantile, 0.95, na.rm = TRUE),
quantile(restotalfut.boot, 0.95, na.rm = TRUE))

out.sum2[, 7] <- c(apply(vecpagosfut.boot, 2, quantile, 0.99, na.rm = TRUE),
quantile(restotalfut.boot, 0.99, na.rm = TRUE))

out.sum2[is.nan(out.sum2)] <- 0

if (metodo == "Arithmetic"){
resumen <- list(call = match.call(expand.dots = FALSE),
sepmethod = metodo,
triangle = triangu,
freqData = Ni,
glm.triangle = triangulo.glm,
reserve.boot = restotalfut.boot,
glm.triangle.bootstrap = tri.glm.boot,
OYearSumm = out.sum,

```



```

    CYearSumm = out.sum2,
    ultimate = ultimate,
    labelac = labelac,
    aux.ult = aux.ult,
    aux.ult.oy = aux.ult.oy,
    aux.lat = aux.lat,
    aux.lat.oy = aux.lat.oy,
    aux.vpfb = aux.vpfb,
    aux.vpfb.cy = aux.vpfb.cy
  )
}
if (metodo == "Geometric"){
  resumen <- list(call = match.call(expand.dots = FALSE),
    sepmethod = metodo,
    triangle = triangu,
    freqData = Ni,
    glm.triangle = triangulo.glm,
    reserve.boot = restotalfut.boot,
    glm.triangle.bootstrap = tri.glm.boot,
    OYearSumm = out.sum,
    CYearSumm = out.sum2,
    ultimate = ultimate,
    labelac = labelac,
    aux.ult = aux.ult,
    aux.ult.oy = aux.ult.oy,
    aux.lat = aux.lat,
    aux.lat.oy = aux.lat.oy,
    aux.vpfb = aux.vpfb,
    aux.vpfb.cy = aux.vpfb.cy
  )
}
class(resumen) <- "glmTaylor"
return(resumen)
}

```

```

ui <- fluidPage(
  titlePanel("GLM for Taylor's Separation Methods"),

  sidebarLayout(
    sidebarPanel(

      radioButtons("cumul", h5("Triangle"),
        choices = list("Incremental" = 1, "Acumulative" = 2)),

      selectInput("metodo", h5("Taylor Separation Method"),
        choices = list("Arithmetic" = "Arithmetic", "Geometric" = "Geometric")),

      numericInput("efecto", h5("Calendar Year Effect (%)"),
        min = 0, max = 100, value = 0, step = 0.5),

```

```

helpText("Enter a percentage value"),

numericInput("muestras",h5("Samples Bootstrapping"),
             min = 0, max = 1000, value = 100, step = 50),

numericInput("semilla",h5("Set seed"),
             min = 0, max = 10000, value = 7, step = 1),
helpText("Enter an specific seed #"),

actionButton("calc","Calculate"),
p("Click the 'Calculate' button to see the results on the main panel")
),

mainPanel(
  tabsetPanel(
    tabPanel("Data",
             h5("Initial Triangle"),tableOutput("sep_triangu")),
    tabPanel("GLM Triangle",
             h5("Full GLM Triangle"),tableOutput("sep_glmtri")),
    tabPanel("O.Y. Summary",
             h5("Summary by Origin Year (Boots)"),tableOutput("sep_oyresu")),
    tabPanel("C.Y. Summary",
             h5("Summary by Calendar Year (Boots)"),tableOutput("sep_cyresu")),
    tabPanel("Graphs",
             tabBox(title="", id="PredicDist",width=NULL,
                   tabPanel("Histogram",
                             plotOutput("sep_hist")
                           ),
                   tabPanel("Ultimate Claims",
                             plotOutput("sep_ultim")
                           ),
                   tabPanel("Latest per O.Y.",
                             plotOutput("sep_latest")
                           ),
                   tabPanel("Future Payments per C.Y",
                             plotOutput("sep_vpfb")
                           )
                 )
             #h5("Graphs"),plotOutput("sep_hist"))
    )
  )
)
)
)
)

# Define server logic ----
server <- function(input, output) {
  sep_triangu <- observeEvent(input$calc,{
    input$cumul

```

```

})
sep_glmtri <- eventReactive(input$calc,{
  input$metodo
})
sep_oyresu <- eventReactive(input$calc,{
  input$metodo
  input$efecto
  input$muestras
  input$semilla
})
sep_cyresu <- eventReactive(input$calc,{
  input$metodo
  input$efecto
  input$muestras
  input$semilla
})
sep_hist <- eventReactive(input$calc,{
  input$metodo
  input$efecto
  input$muestras
  input$semilla
})
sep_ultim <- eventReactive(input$calc,{
  input$metodo
  input$efecto
  input$muestras
  input$semilla
})
sep_latest <- eventReactive(input$calc,{
  input$metodo
  input$efecto
  input$muestras
  input$semilla
})
sep_vpfb <- eventReactive(input$calc,{
  input$metodo
  input$efecto
  input$muestras
  input$semilla
})

output$sep_triangu <- renderTable({
  tri1 <- round(glmTaylorsep(incraccum = input$cumul)$triangle, digits=2)
  year <- as.character(c(1:nrow(tri1)))
  cbind(year,tri1)
})

output$sep_glmtri <- renderTable({
  tri2 <- round(glmTaylorsep(metodo = input$metodo)$glm.triangle, digits=2)

```

```

year <- as.character(c(1:nrow(tri2)))
cbind(year,tri2)

})
output$sep_oyresu <- renderTable({
  table1 <- round(glmTaylorsep(metodo = input$metodo, lambdaini = input$efecto,
                             B=input$muestras, seed = input$semilla )$OYearSumm, digits=2)
  OY<-c(1:(nrow(table1)-1),"Total")
  cbind(OY,table1)
})
output$sep_cyresu <- renderTable({
  table2 <- round(glmTaylorsep(metodo = input$metodo, lambdaini = input$efecto,
                             B=input$muestras, seed = input$semilla )$CYearSumm, digits=2)
  years <- seq(from=(nrow(table2)+1), length=(nrow(table2)-1))
  CY <- c(years,"Total")
  cbind(CY,table2)
})
output$sep_hist <- renderPlot({
  hist(glmTaylorsep(metodo = input$metodo, lambdaini = input$efecto,
                  B = input$muestras, seed = input$semilla )$reserve.boot,
       main="Pred Distribution (Bootstrap)",
       xlab="IBNR",
       col="grey",
       freq=TRUE)
})
output$sep_ultim <- renderPlot({
  ggplot(data = glmTaylorsep(metodo = input$metodo, lambdaini = input$efecto,
                            B = input$muestras, seed = input$semilla)$aux.ult, aes(x = oy, y =
ultimate)) +
  geom_boxplot(color = "dark grey") +
  geom_point(data = glmTaylorsep(metodo = input$metodo, lambdaini =
input$efecto,
                              B = input$muestras, seed = input$semilla)$aux.ult.oy, mapping
= aes(x = oy, y = mu),
            inherit.aes = FALSE, color = "red") +
  labs(title = "Ultimate Claims Costs", subtitle = "(simulated by Bootstrap)", x =
"Origin Year", y = "Claims Costs") +
  theme_bw() +
  theme(panel.grid = element_blank())
})
output$sep_latest <- renderPlot({
  ggplot(data = glmTaylorsep(metodo = input$metodo, lambdaini = input$efecto,
                            B = input$muestras, seed = input$semilla)$aux.lat, aes(x = oy, y =
latest)) +
  geom_boxplot(color = "dark grey") +
  geom_point(data = glmTaylorsep(metodo = input$metodo, lambdaini =
input$efecto,
                              B = input$muestras, seed = input$semilla)$aux.lat.oy, mapping
= aes(x = oy, y = mu),
            inherit.aes = FALSE, color = "red") +

```

```

  labs(title = "Latest Claims ", subtitle = "(simulated by Bootstrap)", x = "Origin
Year", y = "Latest Claims") +
  theme_bw() +
  theme(panel.grid = element_blank())
})
output$sep_vpfb<- renderPlot({
  ggplot(data = glmTaylorsep(metodo = input$metodo, lambdainsi = input$efecto,
    B = input$muestras, seed = input$semilla)$aux.vpfb, aes(x = cy, y =
vpfb)) +
  geom_boxplot(color = "dark grey") +
  geom_point(data = glmTaylorsep(metodo = input$metodo, lambdainsi =
input$efecto,
    B = input$muestras, seed = input$semilla)$aux.vpfb.cy, mapping
= aes(x = cy, y = mu),
  inherit.aes = FALSE, color = "red") +
  labs(title = "Future Payments", subtitle = "(simulated by Bootstrap)", x = "Calendar
Year", y = "Future Payments") +
  theme_bw() +
  theme(panel.grid = element_blank())
})
}

```

```

# Run the app ----
shinyApp(ui=ui, server=server)

```