Doble Grado en
Matemáticas y Estadística

Departamento de Estadística e Investigación Operativa

Facultad de Matemáticas

Trabajo Fin de Grado

Modelos PLS-PM

Realizado por **Irene González Huelva**

Dirigido por Rafael Pino Mejías

Sevilla, Junio de 2018

Índice general

Agradecimientos

Este Trabajo de Fin de Grado, que pone fin a cinco años de formación, es el reflejo de aquello que siempre me decían: “Con tu responsabilidad, trabajo y ganas podrás conseguir lo que te propongas en la vida”. A todas esas personas que creyeron en mí desde el primer momento y que me han apoyado quiero agradecerles todo lo que me han aportado.

Agradecer especialmente a mi tutor D.Rafael Pino Mejías su disposición desde el primer momento en lo que respecta a la dirección de este Trabajo de Fin de Grado. Agradecerle también su tiempo y dedicación en cada una de las tutorías, consultas y correcciones y, además, por la oportunidades brindadas.

El mayor agradecimiento es para mi familia, por el esfuerzo para que fuera posible lo que siempre quise, por apoyarme en todas las decisiones, por animarme en los momentos más difíciles de la carrera y por hacer que los buenos resultados lo fueran por partida doble: por ver la recompensa al esfuerzo y por ver el orgullo reflejado en vuestras caras.

Resumen

La modelización PLS-PM (Partial Least Squares Path Modeling), incluida dentro de los Modelos de Ecuaciones Estructurales, se basa en la idea de la reducción de la dimensionalidad de los datos con el objetivo de estimar una red de causalidad, definida de acuerdo a un modelo teórico donde se consideran conceptos o variables latentes (no observables) que se pueden medir por medio de indicadores observables que llamaremos variables manifiestas. La red de causalidad queda definida por el modelo estructural (que tiene en cuenta las relaciones entre las variables latentes) y el modelo de medida (que recoge las relaciones entre cada variable latente y sus variables manifiestas). Dicha red se estima por medio de un procedimiento iterativo y gracias a diversos indicadores y contrastes de hipótesis es posible evaluar la calidad de dicha estimación.

Todas estas nociones son presentadas en los tres primeros capítulos de este escrito en los que además se tratan temas muy frecuentes en la práctica como son: la comparación de diferentes modelos (en caso de que exista más de uno) y el cómo lidiar con la heterogeneidad de los datos que se usan para crear el modelo. En el capítulo final de este proyecto, con el fin de ilustrar los contenidos previos, se presentan diversas aplicaciones de la modelización PLS-PM con R.

Abstract

The PLS-PM modeling (Partial Least Squares Path Modeling), included in the Structural Equations Models, is based on the idea of reducing the dimensionality of the data in order to estimate a network of causality, defined according to a theoretical model where latent (unobservable) concepts or variables that can be measured by means of observable indicators that we will call manifest variables. The network of causality is defined by the structural model (which takes into account the relationships between latent variables) and the measurement model (which includes the relationships between each latent variable and its manifest variables). This network is estimated by means of an iterative procedure and thanks to various indicators and hypothesis contrasts, it is possible to assess the quality of the estimate.

All these notions are presented in the first three chapters of this paper in which are also treated very common issues in practice such as: the comparison of different models (in the case that there is more than one) and how to deal with the heterogeneity of the data that is used to create the model. In the final chapter of this project, in order to illustrate the previous contents, several applications of PLS-PM modeling are presented with R.

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

1.1. ¿Qué es el PLS-PM?

Los métodos de mínimos cuadrados parciales (PLS) son herramientas analíticas con origen en algoritmos destinados a resolver modelos en situaciones muy prácticas. No derivan de razonamientos probabilísticos o de la optimización numérica aunque están orientados al razonamiento predictivo. Realmente no dependen de la inferencia clásica aunque no por ello deja de tener una base estadística sólida.

Partial Least Squares Path Modeling (PLS-PM) es una metodología del análisis de datos estadísticos que surge como la intersección de modelos de regresión, modelos de ecuaciones estructurales y los métodos de análisis multivariante. Fue desarrollado originalmente por Herman Wold y su grupo de investigación durante los setenta y principios de los ochenta del siglo pasado.

Desde el punto de vista del modelo de ecuaciones estructurales (SEM), los métodos PLS ofrecen diferentes enfoques que no imponen hipótesis distribucionales sobre los datos y que llevan implícita la idea de la reducción de la dimensionalidad. Además de la descripción del PLS-PM como un enfoque alternativo al análisis de estructura de la covarianza (CSA) del SEM, el PLS-PM se considera como una técnica para analizar un sistema de relaciones entre múltiples bloques de variables.

Desde el punto de vista del análisis multivariante, PLS-PM ofrece herramientas para analizar datos con gran dimensionalidad ya que usa la idea de la reducción de la dimensionalidad.

Podemos encontrar diferentes descripciones sobre el PLS-PM y son las siguientes:

- Es el método de mínimos cuadrados parciales en los modelos de ecuaciones estructurales.
- Es un método estadístico para el estudio de las relaciones multivariantes entre variables observadas y latentes.
- Es un enfoque del análisis de datos para el estudio de múltiples relaciones entre bloques de variables manifiestas (observadas) en el que cada bloque juega el papel de un concepto teórico que aparece en forma de variable latente (no observada), entre las cuales existen relaciones lineales.

Aunque existan diversas formas de entender el PLS Path Modeling (PLS-PM) y no necesariamente excluyentes, entenderemos el PLS-PM como una metodología del análisis de datos que se acerca a los modelos de ecuaciones estructurales (SEM) debido a que utiliza diversos métodos estadísticos para estimar un sistema de relaciones de causalidad, definido de acuerdo a un modelo teórico, entre conceptos latentes no observables (que llamaremos variables latentes, denotadas por LV), siendo cada uno de ellos medidos a través de un número de indicadores observables (que llamaremos variables manifiestas, denotadas por MV). Una aplicación común del PLS-PM es calcular índices que cuantifican alguna noción de importancia.

PLS Path Modeling ha sido propuesto como un procedimiento de estimación basado en componentes distinto del enfoque basado en covarianzas de tipo LISREL ya que no reproduce la matriz de covarianza de la muestra sino que, en el mejor de los casos, explica la varianza residual de las variables latentes y, potencialmente, también de las variables manifiestas en cualquier regresión realizada en el modelo puesto que el PLS-PM se basa en un algoritmo iterativo que resuelve por separado los bloques del modelo de medida (que comprende las relaciones entre las LVs y sus MVs) y luego, en un segundo paso, estima los coeficientes de ruta del modelo estructural (que comprende las relaciones entre las LVs). Aunque haya sido propuesto como un procedimiento de estimación, está más orientado a optimizar las predicciones que la precisión estadística de las estimaciones.

PLS-PM se considera como un enfoque de modelado "suave" donde no se requieren hipótesis fuertes (con respecto a las distribuciones, el tamaño de la muestra y la escala de medida). Esta es una característica muy interesante, especialmente en los campos de aplicación donde tales hipótesis no son sostenibles, al menos en su totalidad y además, marca una importante diferencia con el SEM que supone fuertes hipótesis distribucionales y necesita mayor tamaño muestral. Por otro lado, esto implica una falta del marco inferencial paramétrico clásico que se reemplaza por intervalos de confianza y procedimientos de contrastes de hipótesis basados en métodos de remuestreo, como jackknife y bootstrap. También conduce a propiedades estadísticas menos ambiciosas para las estimaciones.

1.2. Breve reseña histórica sobre PLS-PM

El desarrollo de los métodos PLS se llevó a cabo en un período de aproximadamente veinte años, desde mediados de los años sesenta hasta principios de los ochenta, de la mano de Herman Wold y su grupo de investigación.

Inicialmente algunos aspectos de los métodos PLS surgieron de otras técnicas (principalmente de la regresión de mínimos cuadrados ordinarios) y luego se adaptaron para abordar distintas tareas del análisis de datos. En un primer período estos métodos se aplicaban a ciencias sociales y a principios de los ochenta, Svante Wold (hijo de Herman Wold) desarrolló principios y aplicaciones en Química y la industria de la alimentación. La evolución de los métodos PLS ha dado lugar a dos campos de aplicación del análisis PLS: los modelos de regresión PLS (PLS-R) y el modelado de ruta PLS (PLS-PM).

Inicialmente, los métodos PLS nacen del trabajo del estadístico sueco Herman Wold. Nació el 25 de diciembre de 1908 en Skien donde estuvo hasta que en 1912 se mudara a Suecia.

En 1927 comenzó sus estudios en la Universidad de Estocolmo donde estudió Física, Matemáticas y Economía. En 1930, tras graduarse, encontró su primer trabajo aunque finalmente decidió estudiar un doctorado bajo la tutela de Harald Cramer.

En 1938 terminó su doctorado y fue profesor de Estadística Matemática y Matemática Actuarial hasta que en 1942 obtuvo la Cátedra de Estadística en la Universidad de Uppsala en la que permanecío hasta 1970, año en el que se fue a la Universidad de Gotemburgo hasta que se retirara en 1975. Ese mismo año volvió a la ciudad de Uppsala donde continuó con su labor de investigación como profesor emérito.

Muchos de los trabajos de Herman Wold estuvieron relacionados con la Econometría cuya evolución estuvo marcada por el análisis de series temporales y el análisis de demanda. En estos dos hechos Herman Wold hizo bastantes aportaciones ya que durante los estudios de grado, Wold trató el tema de las series temporales y enfocó sus actividades postdoctorales al análisis de demanda aunque en ambas situaciones se basó en el principio de mínimos cuadrados. En su período de estudio de las series temporales, sus primeras aportaciones fueron el estudio de la predicción en un sólo paso de una serie temporal y el conocido teorema de descomposición que aparece en su tesis doctoral: *Un estudio en el análisis de series temporales estacionarias*. A partir de la obtención de su doctorado, entre 1938 y 1940, llevó a cabo el análisis de la demanda del consumidor.

Directamente relacionado con el trabajo realizado en la Econometría, Wold se vió involucrado en una confrontación ocurrida dentro de este campo durante la década de 1940 y fue la disputa entre mínimos cuadrados ordinarios y máxima verosimilitud ya que los mínimos cuadrados ordinarios era la herramienta utilizada hasta el momento pero a medida que los modelos comenzaron a ser más sofisticados, empezó a ser cuestionado. Este período es crucial para el desarrollo del PLS porque en este tiempo Wold defiende el principio de mínimos cuadrados en contra de otros métodos, especialmente máxima verosimilitud.

El protagonista principal de este período fue Trygve Haavelmo, quien destacó los problemas de los mínimos cuadrados cuando se aplica para estimar un modelo de ecuaciones simultáneas. Para resolver el problema, Haavelmo recurrió a la máxima verosimilitud. Como Wold había utilizado la regresión de mínimos cuadrados ordinarios en sus anteriores trabajos, debido al rechazo a dicha técnica por parte de Haavelmo, en 1946, Bentzel y Wold primero distinguieron si un sistema de ecuaciones simultáneas era recursivo (más tarde llamado sistemas de cadenas causales) o no recursivo (más tarde llamado sistemas interdependientes) y luego demostraron la equivalencia de la estimación de mínimos cuadrados con la estimación de máxima verosimilitud en un sistema recursivo cuando las perturbaciones de las diferentes ecuaciones se distribuían de forma independiente y normal.

Durante las décadas de 1950 y 1960, el interés de investigación de Wold se amplió desde la Econometría a otros análisis no experimentales y a la filosofía de la ciencia. En particular, gran parte de su trabajo en este campo giraba en torno a la noción de causalidad.

Durante la década de 1950, Herman Wold dedicó una parte de su trabajo a discutir las nociones de construcción de modelos y causalidad. A principios de la década de 1950, Wold puso énfasis en dos puntos:

- la interpretación causal y el uso operativo de la forma estructural de los sistemas recursivos (sistemas de cadenas causales)
- las dificultades alrededor de la forma estructural de los sistemas interdependientes

Como a Wold le fue imposible encontrar una solución por mínimos cuadrados parciales para sistemas no recursivos, a finales de la década de 1950, al darse cuenta de que su discurso sobre las nociones de causalidad y modelado tuvo poco o ningún impacto en sus análisis, finalmente se dio por vencido. Durante un período de visita a la Universidad de Columbia entre 1958 y 1959, sufrió lo que podríamos llamar una "crisis intelectual" ya que renunció a todo lo que había hecho y comenzó de nuevo desde una nueva perspectiva conceptual y filosófica. Wold comenzó de nuevo sobre la base de lo que denominó *especificación del predictor* que tiene en cuenta la parte sistemática de una relación predictiva expresada como la expectativa condicional correspondiente. Este enfoque le condujo al método de punto fijo que se desarrolló como su nueva propuesta para estimar sistemas interdependientes por mínimos cuadrados. En diciembre de 1964, Wold viaja a los Estados Unidos con el fin de presentar su método de punto fijo para analizar los sistemas interdependientes generalizados. Durante uno de sus seminarios en la Universidad de Carolina del Norte, el profesor G. S. Tolley le preguntó a Wold si era posible aplicar su procedimiento para calcular los componentes principales lo que le permitió abrir nuevas líneas de investigación y desarrollar el marco para los métodos de mínimos cuadrados parciales.

El primer documento que presenta la base de lo que una década más tarde daría lugar a PLS Path Modeling es un artículo de 1966 escrito por Wold *Nonlinear Estimation by Iterative Least Squares Procedures* (NILES) con el que introduce el marco y las aplicaciones que pueden tratarse mediante diferentes algoritmos iterativos basados en regresiones de mínimos cuadrados. Este repertorio de aplicaciones ilustra los problemas que su equipo en el Instituto de Estadística de la Universidad de Uppsala ha estado trabajando, tales como:

- Componentes principales
- Correlaciones canónicas
- Modelos híbridos de componentes principales, correlaciones canónicas y regresión múltiple
- Componentes principales en el caso de información parcial
- Regresión cociente
- Análisis factorial

Directamente relacionado con la primera publicación de NILES, existe otra publicación ese mismo año sobre estimación de componentes principales y modelos relacionados por mínimos cuadrados iterativos.

El acrónimo NILES es utilizado por Wold como un término genérico para agrupar diferentes dispositivos de linealización para estimar modelos no lineales. Aunque todavía no usa la palabra "Parcial", este concepto ya se refleja en sus procedimientos: la idea de dividir los parámetros de un modelo en subconjuntos para que puedan estimarse en partes.

Unos años más tarde, en 1969, aparece otro documento *Nonlinear iterative partial least squares (NIPALS) estimation procedures*. Esta es la publicación donde las palabras mínimos cuadrados parciales se usan por primera vez, pero aún no implica un enfoque para los modelos de variables latentes.

Wold comienza a usar el acrónimo NIPALS en lugar del acrónimo NILES utilizado anteriormente en 1966. Como las primeras publicaciones estaban relacionadas con el análisis de componentes principales (PCA), actualmente cuando se habla de NIPALS nos referimos al algoritmo PLS para el PCA.

En 1971, Karl Joreskog presenta su enfoque LISREL para modelos de ecuaciones estructurales con variables latentes. Motivado por este tipo de modelos, Herman Wold adapta su modelo NIPALS para acercarse a los modelos SEM. Una serie de publicaciones en la década de 1970 reflejan la transformación y evolución de las ideas detrás de los métodos PLS:

- *From hard to soft modelling* (1975)
- *Path models with latent variables: The Non-Linear Iterative Partial Least Squares (NIPALS) approach* (1975)
- *Open path models with latent variables: The NIPALS (Nonlinear Iterative Partial Least Squares) approach* (1976)

A finales de la década de 1970, el término NIPALS fue reemplazado por el acrónimo PLS.

A mediados de la década de 1980, los métodos PLS parecían estar viviendo una era de expansión y consolidación. Entre 1982 y 1985 se describieron las etapas específicas del algoritmo PLS. Además, los métodos de regresión PLS se están desarrollando con éxito en Quimiometría, término inventado por Svante Wold en 1971. Por otro lado, Jan-Bernd Lohmöller en 1987 desarrolla LVPLS, el primer programa de computadora que integra el algoritmo básico PLS-PM y las extensiones hechas por él mismo.

Sin embargo, la década de 1990 fue un período que fue testigo de una metamorfosis del marco PLS en el que se olvidó el enfoque PLS para los modelos de ruta con variable latente y solo el método de regresión PLS se conservó para un mayor desarrollo.
Pese a ello, algunos profesionales (destacando un grupo de estadísticos y analistas bajo la dirección de Michael Tenenhaus) siguieron activos en el marco del PLS-PM.

A pesar de ello, PLS-PM adquirió una popularidad creciente durante la primera década de los años 2000 debido principalmente al proyecto del Sistema de Índice de Satisfacción Europeo (ESIS) que impulsó el desarrollo del software PLS-PM y su aplicación en estudios de marketing. Además, la serie de simposios internacionales exclusivamente desechados a los métodos PLS contribuyó a su difusión en todo el mundo la cual se vió apoyada por los programas y sofwares que permiten a los usuarios aplicar los métodos PLS entre los que destacan *PLS-Graph*, el primer programa PLS-PM con una interfaz gráfica de usuario para dibujar diagramas de ruta en el que además propone una validación cruzada de los parámetros del modelo PLS que fue desarrollado por Wynne Chin, *SmartPLS* y el módulo PLSPM de *XLSTAT* aunque el que se usará en esta memoria será *R* con el paquete *plspm*.

Capítulo 2

Marco teórico del PLS-PM

2.1. Estructura

Los modelos PLS-PM tienen como objetivo estimar las relaciones entre Q $(q = 1, \ldots, Q)$ bloques de variables, que son expresión de variables latentes no observables, las cuales se miden a través de variables manifiestas asociadas. Con el fin de conseguir ese objetivo, PLS-PM construye un sistema de ecuaciones interdependientes basadas en regresiones simples y múltiples donde un sistema estima las relaciones entre las variables latentes así como las de las variables manifiestas con sus propias variables latentes.

Formalmente, supongamos que tenemos P variables $(p = 1, ..., P)$ observadas sobre N unidades experimentales $(n = 1, ..., N)$. Como resultado tenemos el conjunto de datos $(x_{n\times p})$ agrupado en una tabla de datos $(\boldsymbol{X})$ dividida en bloques mutuamente excluyentes: $\boldsymbol{X} = [\boldsymbol{X_1}, ..., \boldsymbol{X_q}, ...\boldsymbol{X_Q}]$ donde $\boldsymbol{X}_q$ son los datos del q-ésimo bloque que tiene P_q variables y está asociado a la variable latente LV_q.

Todo modelo PLS-PM está compuesto por dos submodelos: el modelo de medida y el modelo estructural.

2.1.1. El modelo estructural

El modelo estructural o también llamado modelo interno, tiene en cuenta las relaciones entre las variables latentes y puede escribirse matemáticamente como:

$$\boldsymbol{\xi}_j = \beta_{0j} + \sum_{q:\xi_q \to \xi_j} \beta_{qj}\boldsymbol{\xi}_q + \boldsymbol{\zeta}_j$$

donde $\boldsymbol{\xi}_j$ $(j = 1, ..., J)$ es la j-ésima variable endógena latente, β_{qj} es el coeficiente de ruta que relaciona la q-ésima variable exógena latente con la j-ésima endógena y $\boldsymbol{\zeta}_j$ es el error de la relación interna, es decir, el término que indica la perturbación producida en la predicción de la j-ésima variable latente endógena por sus variables latentes explicativas. Las hipótesis del modelo son: la de regresión $(E(\boldsymbol{\xi}_q|\boldsymbol{\xi}_j) = \beta_{0j} + \sum_{j:\xi_j \to \xi_q} \beta_{jq}\boldsymbol{\xi}_j)$ y la de incorrelación de las variables latentes con sus errores $(cov(\boldsymbol{\xi}_q, \boldsymbol{\zeta}_q) = 0)$.

2.1.2. El modelo de medida

El modelo de medida o también llamado modelo externo, tiene en cuenta las relaciones entre cada variable latente y sus correspondientes variables manifiestas. Su formulación depende de la dirección de dichas relaciones por lo que se distinguen tres tipos de modelos de medida: el **modelo reflexivo**, el **modelo formativo** y el **modelo MIMIC**.

2.1.2.1. El modelo reflexivo

En este modelo, se supone que el bloque de variables manifiestas relacionadas con una variable latente mide un concepto subyacente único. Este modelo se caracteriza porque cada variable manifiesta es efecto de la correspondiente variable latente y juega el papel de variable endógena en el modelo de medida específico del bloque. Formalmente, en el modelo reflexivo cada variable manifiesta está relacionada con la variable latente correspondiente mediante un modelo de regresión dado por:

$$\boldsymbol{x}_{pq} = \lambda_{p0} + \lambda_{pq}\boldsymbol{\xi}_q + \boldsymbol{\epsilon}_{pq}$$

donde λ_{pq} es la carga asociada a la p-ésima variable manifiesta del q-ésimo bloque y $\boldsymbol{\epsilon}_{pq}$ es el término de error que representa la imprecisión en el proceso de medida.
Una de las hipótesis de este modelo es que las variables manifiestas vinculadas a la misma variable latente deben estar correlacionadas. Este es el motivo por el que normalmente, se usan las cargas estandarizadas ya que son preferibles para la interpretación porque representan la correlación entre cada variable manifiesta y su correspondiente variable latente. Otra hipotesis es la denominada *especificación del predictor* en la que el error $\boldsymbol{\epsilon}_{pq}$ debe tener media cero y estar incorrelado con la variable latente del mismo bloque :

$$E(\boldsymbol{x}_{pq}|\boldsymbol{\xi}_q) = \lambda_{p0} + \lambda_{pq}\boldsymbol{\xi}_q$$

lo que nos permite asegurar que se cumplen las condiciones de estimación deseables en el modelo de mínimos cuadrados ordinarios. Nótese que para el modelo reflexivo, el modelo de medida reproduce el modelo de análisis factorial en el que cada variable es una función del factor subyacente.

En el modelo reflexivo, es necesario comprobar la consistencia interna, es decir, comprobar la suposición de que cada bloque es homogéneo y unidimensional, esto es, que las variables manifiestas en un bloque miden el mismo concepto subyacente (único).

Existen varias herramientas para comprobar la unidimensinalidad y la homogeneidad de un bloque y son:

- **El alpha de Cronbach**: es un índice que sirve como medida de consistencia interna. Un bloque se considera homogéneo si este índice es mayor que 0.7. El alpha de Cronbach viene dado por:

  $$\alpha = \frac{\sum_{p\neq p'} cor(\boldsymbol{x}_{pq}, \boldsymbol{x}_{p'q})}{P_q + \sum_{p\neq p'} cor(\boldsymbol{x}_{pq}, \boldsymbol{x}_{p'q})} \times \frac{P_q}{P_q - 1}$$

  donde P_q es el número de variables manifiestas en el q-ésimo bloque.
  El cálculo de este coeficiente requiere que las variables manifiestas estén estandarizadas y positivamente correlacionadas.

- **El rho de Dillon-Goldstein**: también es conocido como fiabilidad compuesta. Un bloque se considera homogéneo si el índice es mayor que 0.7. Se define como:

$$\rho = \frac{(\sum_{p=1}^{P_q} \lambda_{pq})^2}{(\sum_{p=1}^{P_q} \lambda_{pq})^2 + \sum_{p=1}^{P_q} (1 - \lambda_{pq}^2)}$$

- **El análisis de componentes principales de un bloque**: un bloque se considera unidimensional si el primer autovalor de su matriz de correlación es mayor que 1 mientras que los otros son menor que la unidad. Se puede implementar un procedimiento bootstrap para evaluar si la estructura de autovalores es significativa o si es debida a fluctuaciones de muestreo. Si se rechaza la unidimensionalidad, se pueden identificar subbloques unidimensionales mediante los patrones de correlación entre variable-factor que se ven en los gráficos de carga.

El rho de Dillon-Goldstein se considera un mejor indicador que el alpha de Cronbach. En efecto, este último asume la conocida equivalencia tau (o paralelismo) de las variables manifiestas, es decir, se asume que cada variable manifiesta es igualmente importante en la definición de la variable latente. El rho de Dillon-Goldstein no hace esta suposición ya que se basa en los resultados del modelo (las cargas) en lugar de las correlaciones observadas entre las variables manifiestas en el conjunto de datos. En realidad, el alpha de Cronbach proporciona una cota inferior de la estimación de la fiabilidad interpretando ésta como la varianza de $\boldsymbol{x}_{pq}$ que es explicada por $\boldsymbol{\xi}_q$,es decir, $rel(\boldsymbol{x}_{pq}) = \frac{\lambda_{pq}^2 var(\boldsymbol{\xi}_q)}{var(\boldsymbol{x}_{pq})} = cor^2(\boldsymbol{x}_{pq}, \boldsymbol{\xi}_q)$.

2.1.2.2. El modelo formativo

Este modelo se caracteriza porque cada variable manifiesta es la causa de la correspondiente variable latente. En este modelo, cada variable manifiesta o cada subbloque de variables manifiestas representan una dimensión diferente del concepto subyacente. Por lo tanto, a diferencia del modelo reflexivo, el modelo formativo no asume la homogeneidad ni la unidimensionalidad del bloque. La variable latente es definida como una combinación lineal de las correspondientes variables manifiestas, así cada variable manifiesta es una variable exógena en el modelo de medida. Estas variables no necesitan estar correladas. Así el modelo de medida se expresa como:

$$\boldsymbol{\xi}_q = \sum_{p=1}^{P_q} \omega_{pq} \boldsymbol{x}_{pq} + \boldsymbol{\delta}_q$$

donde ω_{pq} es el coeficiente que une cada variable manifiesta con su variable latente correspondiente y el término de error $\boldsymbol{\delta}_q$ representa la fracción de la variable latente correspondiente no explicada para por el bloque de variables manifiestas. La hipótesis de este modelo es conocida como *especificación del predictor* y es la siguiente:

$$E(\boldsymbol{\xi}_q | \boldsymbol{x}_{1q}, ..., \boldsymbol{x}_{P_q q}) = \sum_{p=1}^{P_q} \omega_{pq} \boldsymbol{x}_{pq}$$

2.1.2.3. El modelo MIMIC

Es una mezcla de los modelos reflexivo y formativo dentro del mismo bloque de variables manifiestas.

Independientemente del tipo de modelo de medida, en cuanto a la convergencia del algoritmo, las puntuaciones de las variables latentes estandarizadas ($\widehat{\boldsymbol{\xi}}_q$) asociadas a la q-ésima variable latente ($\boldsymbol{\xi}_q$) se calculan como una combinación lineal de su propio bloque de variables manifiestas a través de la llamada *relación de pesos* definida como:

$$\widehat{\boldsymbol{\xi}}_q = \sum_{p=1}^{P_q} w_{pq} \boldsymbol{x}_{pq}$$

donde las variables $\boldsymbol{x}_{pq}$ están centradas y w_{pq} son los pesos externos. Estos pesos se obtienen tras la convergencia del algoritmo y luego se transforman para producir puntuaciones variables latentes estandarizadas. Sin embargo, cuando todas las variables manifiestas son observadas mediante la misma escala de medida y todos los pesos externos son positivos, es interesante y factible expresar estas puntuaciones en la escala original. Esto se consigue usando pesos normalizados $\widetilde{w}_{pq}$ definidos como:

$$\widetilde{w}_{pq} = \frac{w_{pq}}{\sum_{p=1}^{P_q} w_{pq}} \quad con \quad \sum_{p=1}^{P_q} \widetilde{w}_{pq} = 1 \quad \forall q : P_q > 1$$

Es muy importante no confundir la *relación de pesos* con un modelo formativo ya que en la relación de pesos no afecta el tipo de relación entre los dos tipos de variables en el modelo externo.

2.2. Algoritmo PLS-PM

2.2.1. Descripción

El algoritmo PLS-PM es un proceso iterativo que trabaja sobre variables manifiestas centradas o estandarizadas y que nos permite estimar los pesos externos (w_{pq}), las puntuaciones de las variables latentes ($\widehat{\boldsymbol{\xi}}_q$) y las cargas ($\lambda_{pq}$). El procedimiento de estimación se denomina parcial ya que resuelve los bloques de uno en uno mediante regresiones lineales simples y múltiples alternas. El algoritmo lo podremos dividir en tres etapas:

- **Etapa 1**: Obtención de los pesos para conseguir las puntuaciones de las variables latentes. En esta etapa, la estimación de los pesos externos se consigue a través del intercambio de pasos de estimaciones internas y externas, iterados hasta la convergencia.
- **Etapa 2**: Estimación de coeficientes de ruta (β_{qj}) mediante regresiones entre las puntuaciones estimadas de las variables latentes de acuerdo con el sistema de relaciones estructurales especificadas.
- **Etapa 3**: Obtención de las cargas mediante correlaciones entre las variables latentes y manifiestas.

2.2.2. Algoritmo

Etapa 1: Obtener los pesos para conseguir las puntuaciones de las LVs
Es la parte más importante de la metodología del PLS-PM. Esta etapa se basa en un proceso iterativo con el objetivo de conseguir las puntuaciones de las variables latentes. El algoritmo del proceso iterativo es el siguiente:

- **Paso inicial: Pesos iniciales arbitrarios**
  El algoritmo se inicia eligiendo unos pesos iniciales arbitrarios w_{pq}. Realmente, necesitaríamos aplicarles una transformación (normalización) para que las variables latentes estén estandarizadas, o lo que es lo mismo, que las puntuaciones tengan varianza unidad. Por tanto, se toman los w_{pq} iguales a uno.
- **Paso 1: Aproximación externa de las variables latentes**
  En esta etapa de estimación externa, cada variable latente se estima como combinación lineal ponderada de sus variables manifiestas:

$$\boldsymbol{\nu}_q \propto \pm \sum_{p=1}^{P_q} w_{pq} \boldsymbol{x}_{pq} = \pm \boldsymbol{X_q w_q}$$

  donde $\boldsymbol{\nu}_q$ es la estimación externa estandarizada de la q-ésima variable latente $\boldsymbol{\xi}_q$, el símbolo $\propto$ significa que el lado izquierdo de la ecuación corresponde al lado derecho estandarizado y el símbolo $\pm$ muestra ambigüedad en el signo que normalmente se elimina tomando el signo que hace que la estimación externa se correlacione positivamente con la mayoría de sus variables manifiestas, es decir, el signo será $signo[\sum_q signo(cor(\boldsymbol{x}_{pq}, \boldsymbol{\nu}_q))]$. En cualquier caso, el usuario puede invertir el signo de los pesos de un bloque completo para hacer que sea coherente con la definición de la variable latente.
- **Paso 2: Obtener los pesos internos**
  Los pesos internos son los coeficientes $e_{qq'}$ que aparecen en la estimación externa de las variables latentes. Pueden calcularse de distinta forma atendiendo a distintos esquemas:
  - **Esquema de centroide** Fue el esquema original de Wold. Este esquema toma el signo de la correlación entre la estimación externa $\boldsymbol{\nu}_q$ de la q-ésima variable latente y la estimación externa de la variable latente q'-ésima $\boldsymbol{\nu}_{q'}$ conectada con $\boldsymbol{\nu}_q$ y en caso contrario, toma el valor 0. El problema de este esquema puede darse cuando las correlaciones son cercanas a cero dando lugar a cambios de signo de +1 a -1 durante las iteraciones.
  - **Esquema factorial**
    Fue propuesto por Lohmöller. Este esquema toma la correlación entre la estimación externa $\boldsymbol{\nu}_q$ de la q-ésima variable latente y la estimación externa de la variable latente q'-ésima $\boldsymbol{\nu}_{q'}$ conectada con $\boldsymbol{\nu}_q$ y en caso contrario, toma el valor 0. Este esquema no sólo tiene en cuenta si una variable latente está asociada a otra sino cuánto.
  - **Esquema de ponderación estructural o de ruta**
    Este esquema toma el coeficiente de regresión de $\boldsymbol{\nu}_q$ y $\boldsymbol{\nu}_{q'}$ conectada con $\boldsymbol{\nu}_q$ si $\boldsymbol{\nu}_q$ juega el papel de variable dependiente en la ecuación estructural específica, o toma el coeficiente de correlación en caso de que sea variable independiente.

En el marco teórico, la elección del esquema es de gran importancia, especialmente para entender cómo el PLS-PM puede aplicarse para diferentes técnicas del análisis multivariante aunque en la práctica, no tiene mucha relevancia en el proceso de estimación. Si realizamos una comparación de los tres esquemas, el de ponderación de ruta parece ser el más coherente por usar la dirección de las relaciones estructurales entre variables latentes aunque el esquema de centroide se usa muy a menudo ya que se adapta bien a los casos en que las variables manifiestas en un bloque están fuertemente correlacionadas entre sí. El esquema factorial, en cambio, se adapta mejor a los casos donde tales correlaciones son más débiles. Por lo general, se recomienda utilizar el esquema de ponderación de ruta ya que es el único esquema de estimación que considera explícitamente la dirección de las relaciones tal como se especifica en el modelo de ruta predictivo aunque presenta problemas en el caso de que la matriz de correlaciones de las variables latentes sea singular.

- **Paso 3: Calcular la aproximación interna de las LVs**
  Una vez que tengamos calculados los pesos internos, en el paso de estimación interna, cada variable latente es estimada considerando sus conexiones con las otras Q' variables latentes adyacentes sin importar cuáles son las variables dependientes o independientes:

$$\boldsymbol{\vartheta}_q \propto \sum_{q'=1}^{Q'} e_{qq'} \boldsymbol{\nu}_{q'}$$

  donde $\boldsymbol{\vartheta}_q$ es la estimación interna estandarizada de la q-ésima variable latente $\boldsymbol{\xi}_q$ y $e_{qq'}$ son los pesos internos.

- **Paso 4: Calcular los nuevos pesos externos**
  Una vez que se obtiene una primera estimación de las variables latentes, el algoritmo continúa actualizando los pesos externos w_{pq}. Para actualizar los pesos externos hay dos formas estrechamente relacionadas que dependen de la relación de las variables latentes con sus variables manifiestas y son:

  - **Modo A**: cada peso externo w_{pq} se actualiza como el coeficiente de regresión en la regresión simple de la p-ésima variable manifiesta del q-ésimo bloque ($\boldsymbol{x}_{pq}$) sobre la estimación interna de la q-ésima variable latente $\boldsymbol{\vartheta}_q$.
    De hecho, como $\boldsymbol{\vartheta}_q$ está estandarizada, el peso externo w_{pq} se obtiene como: $w_{pq} = cov(\boldsymbol{x}_{pq}, \boldsymbol{\vartheta}_q)$ es decir, el coeficiente de regresión coincide con la covarianza entre cada variable manifiesta y la estimación interna de su variable latente correspondiente. En el caso de que las variables manifiestas también estén estandarizadas, las covarianzas pasan a ser correlaciones.

  - **Modo B**: el vector $\boldsymbol{w}_q$ de pesos w_{pq} asociados a las variables manifiestas del q-ésimo bloque se actualiza como el vector de coeficientes de regresión en la regresión múltiple de la estimación interna de la q-ésima variable latente $\boldsymbol{\vartheta}_q$ sobre las variables manifiestas en $\boldsymbol{X}_q$:

$$\boldsymbol{w}_q = (\boldsymbol{X}'_q \boldsymbol{X}_q)^{-1} \boldsymbol{X}'_q \boldsymbol{\vartheta}_q$$

    donde $\boldsymbol{X}_q$ comprende las P_q variables manifiestas $\boldsymbol{x}_{pq}$ previamente centradas y escaladas por $\sqrt{1/N}$.

La elección del modo de estimación del peso exterior está estrechamente relacionada con la naturaleza del modelo de medida. Para un modelo reflexivo, el *Modo A* es más apropiado, mientras que el *Modo B* es mejor para un modelo formativo. Además, se sugiere el *Modo A* para las variables latentes endógenas y el *Modo B* para las exógenas. El *Modo A* y el *Modo B* pueden usarse simultáneamente para el modelo MIMIC usando el *A* para la parte reflexiva y el *B* para la formativa.

Observación

Podemos destacar las siguientes características de cada uno de los modos:

$$\text{Modo A} = \begin{cases} \text{Las variables manifiestas deben estar correladas} \\ \text{Se realizan varias regresiones de mínimos cuadrados ordinarios} \\ \text{Varianza explicada} \\ \text{Consistencia interna} \\ \text{Estabilidad de resultados con bloques bien definidos} \\ \text{Si sólo hay un bloque, los resultados coinciden con la 1}^{\text{a}}\text{CP el ACP} \end{cases}$$

$$\text{Modo B} = \begin{cases} \text{Las variables manifiestas no deben estar correladas} \\ \text{Se realiza una regresión múltiple de mínimos cuadrados ordinarios} \\ \text{Valores altos de } R^2 \text{ para variables latentes endógenas} \\ \text{Multidimensionalidad} \\ \text{Inestabilidad de resultados si hay bloques que no estén bien definidos} \end{cases}$$

Vale la pena observar que el modo B puede verse afectado por la multicolinealidad entre las variables manifiestas que pertenecen al mismo bloque. Si esto sucede, la regresión PLS se puede usar como una alternativa más estable y mejor interpretable que la regresión de mínimos cuadrados ordinarios (OLS) para estimar los pesos externos en un modelo formativo, definiendo así un nuevo modo, que llamaremos **Modo PLS** y que se adapta bien a los modelos formativos donde los bloques son multidimensionales pero con menos dimensiones que el número de variables manifiestas.

Convergencia

Este proceso se repite hasta la convergencia de los pesos externos. Para comprobar dicha convergencia, se sigue el siguiente criterio: en cada iteración, llamémosla S, comprobamos la convergencia comparando los pesos externos de la iteración S (w_{pq}^{S}) con los de la iteración anterior (w_{pq}^{S-1}). Diremos que tenemos convergencia de los pesos externos si $|w_{pq}^{S-1} - w_{pq}^{S}| < 10^{-5}$

Hasta ahora, la convergencia se ha demostrado solo para diagramas de ruta con uno o dos bloques. Sin embargo, para los modelos multibloque, es habitual que se produzca la convergencia de los pesos. En cuanto a la convergencia, las estimaciones de las puntuaciones de las variables latentes se obtienen de acuerdo a la siguiente fórmula: $\widehat{\boldsymbol{\xi}}_q = \sum_{p=1}^{P_q} w_{pq}\boldsymbol{x}_{pq}$. Por lo tanto, el modelo de ruta PLS proporciona una estimación directa de las puntuaciones individuales de las variables latentes como suma de variables manifiestas que naturalmente implican errores de medida. El precio de obtener estas puntuaciones es la inconsistencia de las estimaciones.

Etapa 2: Estimar los coeficientes de ruta

A continuación, los coeficientes estructurales o coeficientes de ruta se estiman a través de regresiones de mínimos cuadrados ordinarios simples o múltiples entre las puntuaciones de las variables latentes. La regresión PLS puede reemplazar a la regresión de mínimos cuadrados ordinarios en el caso de que nos encontremos frente a los siguientes problemas: puntuaciones perdidas, variables latentes fuertemente correlacionadas, un número limitado de unidades en comparación con el número de predictores de la ecuación estructural más compleja.

Etapa 3: Obtener las cargas
Finalmente, las cargas se obtienen mediante el cálculo de correlaciones entre una variable latente y sus variables manifiestas: $\lambda_{pq} = cor(\boldsymbol{\xi}_q, \boldsymbol{x}_{pq})$.

Este algoritmo PLS-PM, propuesto por Lohmöller, es el procedimiento más conocido para calcular puntuaciones de las variables latentes, posee más ventajas y es más fácil de interpretar que otro procedimiento menos conocido que fue propuesto inicialmente por Wold. Sin embargo, para asegurar la convergencia, es mejor el procedimiento de Wold ya que el algoritmo es monótonamente convergente.

2.3. Validación del modelo

La validación del modelo se refiere únicamente a la forma en que se modelan las relaciones, tanto en el modelo estructural como en el modelo de medida; en particular, las siguientes hipótesis nulas deben ser **rechazadas**:

a) $\lambda_{pq} = 0$ ya que se supone que cada variable manifiesta se correlaciona con su variable latente correspondiente.

b) $w_{pq} = 0$ ya que se supone que cada variable latente se verá afectada por todas las variables manifiestas de su bloque.

c) $\beta_{qq'} = 0$ ya que se supone que el predictor latente es explicativo con respecto a su respuesta latente.

d) $cor(\boldsymbol{\xi}_q, \boldsymbol{\xi}_{q'}) = 0$. Rechazar esta hipótesis significa aceptar la **validez nomológica** del PLS-PM ya que se supone que las variables latentes están conectadas por una correlación estadísticamente significativa.

e) $cor(\boldsymbol{\xi}_q, \boldsymbol{\xi}_{q'}) = 1$. Rechazar esta hipótesis significa aceptar la **validez discriminante** del PLS-PM ya que se supone que cada variable latente mide un concepto diferente.

f) AVE_q y $AVE_{q'}$ deben ser menor que $cor^2(\boldsymbol{\xi}_q, \boldsymbol{\xi}_{q'})$ ya que una variable latente debe estar más estrechamente relacionada con su bloque de variables manifiestas que con otro bloque. AVE_q viene dado por:

$$AVE_q = \frac{\sum_p [\lambda_{pq}^2 var(\boldsymbol{\xi}_q)]}{\sum_p [\lambda_{pq}^2 var(\boldsymbol{\xi}_q)] + \sum_q (1 - \lambda_{pq}^2)}$$

2.4. Evaluación de la bondad del modelo

PLS-PM carece de un criterio de optimización global bien identificado, por lo que no existe una función de ajuste global para evaluar la bondad del modelo. Además, es un modelo basado en la varianza orientado a realizar predicciones. Por lo tanto, la validación del modelo se centra principalmente en la capacidad predictiva del mismo. De acuerdo con la estructura PLS-PM, se deben comprobar una serie de hipótesis y además, validar cada una de las partes del modelo: el modelo de medida, el modelo estructural y el modelo completo.

2.4.1. Evaluación del modelo de medida

Para validar el modelo de medida, además de comprobar la unidimensionalidad (con las herramientas dadas en la sección 2.1.2.1), usaremos el índice de comunalidad así como las comunalidades individuales y las cargas cruzadas.

2.4.1.1. Comunalidades

Para comprobar que las variables latentes están bien explicadas por sus variables manifiestas usaremos las cargas y las comunalidades, ambas estrechamente relacionadas.

Las **cargas** (λ_{pq}) son las correlaciones entre una variable latente y sus variables manifiestas, es decir,

$$\lambda_{pq} = cor(\boldsymbol{\xi}_q, \boldsymbol{x}_{pq})$$

Las **comunalidades** son definidas de la siguiente forma: para la p-ésima variable manifiesta del q-ésimo bloque se define como:

$$Com(\boldsymbol{\xi}_q, \boldsymbol{x}_{pq}) = cor^2(\boldsymbol{\xi}_q, \boldsymbol{x}_{pq}) = \lambda_{pq}^2$$

Cargas superiores a 0.7 son aceptables debido a que las comunalidades representarían el $0.7^2 \approx 50\,\%$ de la variabilidad común entre una variable manifiesta y su variable latente. En el caso en que alguna variable manifiesta tenga comunalidad baja, se suele eliminar del modelo.

2.4.1.2. Cargas cruzadas

Para evaluar el grado en que una variable latente es diferente a otras se utilizan las **cargas cruzadas** que son las cargas de las variables manifiestas con el resto de variables latentes, distintas a la que está asociada. Esto se hace para saber si la mayor carga de la variable manifiesta la presenta con su variable latente asociada o con otra. En caso de que la carga sea mayor con otra variable (distinta a la que está asociada), habría que considerar si realmente está bien asociada en el modelo.

2.4.1.3. Índice de comunalidad

Para cada q-ésimo bloque en el modelo con más de una variable manifiesta ($P_q > 1$), la calidad del modelo de medida es evaluada mediante el **índice de comunalidad** que está dado por:

$$Com_q = \frac{1}{P_q} \sum_{p=1}^{P_q} cor^2(\boldsymbol{x}_{pq}, \widehat{\boldsymbol{\xi}}_q) \qquad \forall q : P_q > 1$$

Este índice mide qué parte de la variabilidad de las variables manifiestas del q-ésimo bloque es explicada por la puntuación de su variable latente correspondiente $\widehat{\boldsymbol{\xi}}_q$. Además, el índice de comunalidad para el q-ésimo bloque no es más que la media del cuadrado de las correlaciones entre cada variable manifiesta y la puntuación de su variable latente correspondiente. Si las variables manifiestas están estandarizadas es la media de las cargas al cuadrado.

También es posible evaluar la calidad del modelo de medida completo a través de la **comunalidad media**:

$$\overline{Com} = \frac{1}{\sum_{q:P_q>1} P_q} \sum_{q:P_q>1} P_q Com_q$$

Es una media ponderada de las Q comunalidades específicas de cada bloque con pesos iguales al número de variables manifiestas en cada bloque. No es más que la media de todos los cuadrados de las correlaciones entre cada variable manifiesta y la puntuación de su correspondiente variable latente, es decir:

$$\overline{Com} = \frac{1}{\sum_{q:P_q>1} P_q} \sum_{q:P_q>1} \sum_{p=1}^{P_q} cor^2(\boldsymbol{x}_{pq}, \widehat{\boldsymbol{\xi}}_q)$$

Observación

Estas herramientas para estudiar la adecuación del modelo de medida sólo son válidas cuando el modelo de medida es reflexivo ya que en el modelo formativo no suponemos la existencia de correlación entre variables manifiestas y latentes. En el caso de que el modelo sea formativo, comparamos los pesos externos de cada variable manifiesta para determinar cuáles de ellas contribuyen de manera más eficaz en las variables latentes. Además, en el caso de que existar alta multicolinealidad, deberíamos eliminar alguna variable manifiesta del estudio.

2.4.2. Evaluación del modelo estructural

Una vez evaluada la calidad del modelo de medida, pasamos a evaluar la calidad del modelo estructural estudiando los resultados obtenidos en cada regresión de las ecuaciones estructurales. La calidad del modelo estructural se puede medir en base a dos índices: el coeficiente de determinación R^2 y el índice de redundancia.

2.4.2.1. Coeficiente de determinación

El coeficiente de determinación R^2 sólo se aplica sobre las variables latentes endógenas o dependientes. El coeficiente de determinación R^2 mide la cantidad de varianza en la variable latente endógena explicada por sus variables latentes independientes. A partir de un valor de 0.6 se considera aceptable aunque cabe destacar que este coeficiente no es suficiente para evaluar el modelo completo ya que solo tiene en cuenta el ajuste de cada ecuación de regresión en el modelo estructural.

2.4.2.2. Índice de redundancia

El índice de redundancia surge con el fin de vincular el rendimiento predictivo del modelo de medida con el estructural.
El **índice de redundancia** para el j-ésimo bloque endógeno mide la parte de la variabilidad de las variables manifiestas conectadas con la j-ésima variable latente endógena explicada por las variables latentes conectadas con el bloque en cuestión. Otra definición alternativa es que la redundancia es la cantidad de varianza de una variable latente endógena explicada por sus variables latentes independientes. En definitiva, la redundancia refleja la capacidad de un conjunto de variables latentes independientes para explicar la variación de una variable latente dependiente. Matemáticamente, se define como:

$$Red_j = Com_j \times R^2(\widehat{\boldsymbol{\xi}}_j, \widehat{\boldsymbol{\xi}}_{q:\boldsymbol{\xi}_q \to \boldsymbol{\xi}_j})$$

Un índice de redundancia alto indica que la capacidad de predicción del modelo es elevada.

Una medida de la calidad global del modelo estructural es la que proporciona el **índice de redundancia media** que se calcula como:

$$\overline{Red} = \frac{1}{J} \sum_{j=1}^{J} Red_j$$

donde J es el número de variables latentes endógenas en el modelo.

2.4.3. Evaluación del modelo completo

El el marco del PLS, no existe un criterio único para medir la calidad de un modelo, por lo que no se pueden realizar las pruebas estadísticas inferenciales de bondad de ajuste. Como alternativa, las pruebas no paramétricas se pueden aplicar para la evaluación del modelo estructural. En este caso, se utiliza el índice GoF para medir la calidad del modelo PLS-PM global.

2.4.3.1. Índice GoF

El índice *GoF* (Goodness of Fit) es un índice que tiene en cuenta el modelo estructural y de medida y proporciona un valor único para la calidad del modelo global. El **índice *GoF*** se obtiene como la media geométrica del índice de medio de comunalidad y el R^2 medio:

$$GoF = \sqrt{\overline{Com} \times \overline{R^2}}$$

donde el R^2 medio viene dado por la siguiente expresión:

$$\overline{R^2} = \frac{1}{J} R^2(\widehat{\boldsymbol{\xi}}_j, \widehat{\boldsymbol{\xi}}_{q:\boldsymbol{\xi}_q \to \boldsymbol{\xi}_j})$$

El principal inconveniente es que no existe un umbral que nos permita determinar su significación estadística ni a partir de qué valor podemos considerar que el GoF es bueno. Teniendo en cuenta la comunalidad del PLS, podemos considerar valores aceptables del GoF si son superiores a 0.7.

Como se basa en parte en la comunalidad media, el índice GoF es conceptualmente apropiado cuando el modelo de medida es reflexivo. Sin embargo, las comunidades también pueden ser calculadas e interpretadas en el caso de que el modelo sea formativo sabiendo que, en tal caso, las comunidades serán más bajas pero el R^2 será más alto en comparación con el caso del modelo reflexivo. Por lo tanto, para fines prácticos, el índice GoF puede usarse para cualquier tipo de modelo de medida.

Según los valores de $\overline{Com}$ y $\overline{R^2}$, el índice *GoF* puede reescribirse como:

$$GoF = \sqrt{\frac{\sum_{q:P_q>1} \sum_{p=1}^{P_q} cor^2(\boldsymbol{x}_{pq}, \widehat{\boldsymbol{\xi}}_q)}{\sum_{q:P_q>1} P_q} \times \frac{\sum_{j=1}^{J} R^2(\widehat{\boldsymbol{\xi}}_j, \widehat{\boldsymbol{\xi}}_{q:\boldsymbol{\xi}_q \to \boldsymbol{\xi}_j})}{J}}$$

Se obtiene una versión normalizada relacionando cada término de la expresión anterior con el valor máximo correspondiente. En particular, se sabe que en el análisis de componentes principales, la mejor aproximación de un conjunto de variables $\boldsymbol{X}$ viene dada por el autovector asociado al mayor autovalor de la matriz $\boldsymbol{X'X}$. Además, la suma de los cuadrados de las correlaciones entre cada variable y la primera componente principal de $\boldsymbol{X}$ es un máximo.

Si los datos están normalizados, el primer término de dentro de la raíz del *GoF* es tal que $\sum_{p=1}^{P_q} cor^2(\boldsymbol{x}_{pq}, \widehat{\boldsymbol{\xi}}_q) \leq \lambda^1_{(q)}$ donde $\lambda^1_{(q)}$ es el primer autovalor del análisis de componentes principales del q-ésimo bloque de variables manifiestas. Así, el primer término normalizado del índice GoF es:

$$T_1 = \frac{1}{\sum_{q:P_q>1} P_q} \sum_{q:P_q>1} \frac{cor^2(\boldsymbol{x}_{pq}, \widehat{\boldsymbol{\xi}}_q)}{\lambda^1_{(q)}}$$

En otras palabras, aquí la suma de las comunalidades en cada bloque se divide por el primer autovalor del propio bloque.

Del mismo modo, el segundo término de dentro de la raíz del GoF normalizado es:

$$T_2 = \frac{1}{J}\sum_{j=1}^{J}\frac{R^2(\widehat{\boldsymbol{\xi}}_j, \widehat{\boldsymbol{\xi}}_{q:\boldsymbol{\xi}_q\to\boldsymbol{\xi}_j})}{\rho_j^2}$$

donde ρ_j es la primera correlación canónica del análisis canónico de correspondencia entre $\boldsymbol{X}_j$ que contiene las variables manifiestas asociadas a la j-ésima variable latente endógena, y una matriz que contiene las variables manifiestas asociadas a todas las variables latentes que explican $\boldsymbol{\xi}_j$.

Así, el índice GoF es:

$$GoF_{rel} = \sqrt{\frac{1}{\sum_{q:P_q>1}P_q}\sum_{q:P_q>1}\frac{cor^2(\boldsymbol{x}_{pq}, \widehat{\boldsymbol{\xi}}_q)}{\lambda^1_{(q)}} \times \frac{1}{J}\sum_{j=1}^{J}\frac{R^2(\widehat{\boldsymbol{\xi}}_j, \widehat{\boldsymbol{\xi}}_{q:\boldsymbol{\xi}_q\to\boldsymbol{\xi}_j})}{\rho_j^2}}$$

Este índice toma valores entre 0 y 1. Tanto el índice GoF como GoF_{rel} son índices descriptivos, es decir, no está basado en la inferencia y no hay un límite de significación estadística para estos valores. Como regla general, valores de GoF_{rel} superiores a 0.9 son señal de que el modelo es adecuado.

2.4.3.2. Índices mediante validación cruzada

Como PLS-PM no realiza hipótesis distribucionales, es posible estimar la significación de los parámetros mediante métodos de remuestreo como jackknife y bootstrap. También es posible obtener los índices de calidad (como versión del proceso de validación cruzada) a través del procedimiento de *blindfolding* que consiste en:

1. Dividir la matriz de datos en G grupos. El valor recomendado por Herman Wold fue de $G = 7$.

2. Eliminar un grupo de los datos.

3. Calcular el modelo PLS-PM para el conjunto de datos menos el grupo.

El paso 2 y 3 se repiten hasta que cada grupo se haya eliminado una vez de los datos.

4. Evaluar la calidad del modelo (el completo) midiendo el índice de redundancia y comunalidad mediante validación cruzada. Son un Q^2 basado en la comunalidad y un Q^2 basado en la redundancia. En ambos casos, se obtiene una medida de relevancia para la variable latente endógena que se pretende predecir. Esta medida de relevancia se considera más informativa que R^2 y AVE, ya que tienen sesgo inherente y plantea el problema del sobreajuste de datos. Esta medida pretende ayudar a evaluar la validez predictiva a nivel de variables latentes.

Los índices vienen dados por la siguiente expresión:

Índice de CV-comunalidad

$$H_q^2 = 1 - \frac{\sum_q\sum_i\left(x_{pqi} - \overline{x}_{pq} - \widehat{\lambda}_{pq(-i)}\widehat{\boldsymbol{\xi}}_{q(-i)}\right)^2}{\sum_q\sum_i\left(x_{pqi} - \overline{x}_{pq}\right)}$$

La media de este índice (para bloques endógenos) se puede usar para medir la calidad global del modelo de medida si son positivos para todos los bloques.

Índice de CV-redundancia

$$F_q^2 = 1 - \frac{\sum_q \sum_i (x_{pqi} - \overline{x}_{pq} - \widehat{\lambda}_{pq(-i)} Pred(\widehat{\boldsymbol{\xi}}_{q(-i)}))^2}{\sum_q \sum_i (x_{pqi} - \overline{x}_{pq})}$$

La media de este índice (para bloques endógenos) se puede usar para medir la calidad global del modelo de medida si son positivos para todos los bloques endógenos.

Otros índices obtenidos mediante bootstrap

Debido a la necesidad de índices que ayuden a proporcionar información sobre la validez predictiva de un modelo PLS a nivel de variables latentes, se presenta un procedimiento de remuestreo bootstrap para la validación cruzada de los pesos obtenidos en un análisis PLS para predecir variables latentes endógenas. Está destinado a dar información sobre el valor de los pesos proporcionados en un análisis PLS en lo que se refiere a maximizar el R^2 de las variables latentes dependientes de un modelo. La validación cruzada estándar implica el uso de un conjunto de datos de entrenamiento seguido de un conjunto de datos test (no necesariamente del mismo tamaño) de la misma población para evaluar la predictibilidad de las estimaciones del modelo.
Distinguimos el uso del conjunto muestral original como el conjunto de entrenamiento para estimar un modelo de PLS dado y el uso del remuestreo bootstrap para crear nuevos conjuntos de datos.
Los pesos de las variables manifiestas derivadas del conjunto de muestra original se utilizan en las nuevas muestras bootstrap y las distintas medidas de R^2 se estudian para cada variable latente endógena en el modelo.

Las distintas medidas de R^2 son:

- **R^2 sumado ponderado (WSD)** representa qué tan bien predicen los pesos originales de la muestra dados los nuevos datos (es decir, una nueva muestra bootstrap).
- **R^2 sumado simple (SSD)** refleja la predictibilidad usando el enfoque más simple de pesos unitarios.
- **R^2 sumado optimizado (OSD)** son los R^2 obtenidos al calcular el modelo en cada conjunto de datos bootstrap. Generalmente deberían ser mayores que los R^2 de WSD o SSD.
- **El índice de rendimiento relativo (RPI)** basado en los R^2 WSD y SSD puede calcularse para representar el grado en que los pesos PLS de la muestra original proporcionan una mayor predictibilidad para variables latentes endógenas que el procedimiento más simple de regresión después de la suma simple de variables manifiestas. Para cada conjunto de muestras bootstrap, también se puede completar una ejecución PLS estándar.
- **El índice sumado optimizado (PFO)** cuya interpretación se obtiene al contrastar el WSD con los OSD.

Procedimiento general de cálculo de los índices RPI y PFO

Los pasos específicos para calcular los índices RPI y PFO son los siguientes:

1. Calcular el modelo para la muestra original, anotar los pesos originales y el R^2 para cada variable latente endógena en el modelo.
2. Crear N muestras bootstrap donde cada muestra se utilizará para obtener tres R^2 diferentes (OSD, WSD y SSD) para cada variable latente endógena.
3. Para cada muestra bootstrap, realizar el algoritmo PLS y anotar el R^2 para cada variable latente endógena. Esto se denominará R^2 sumado optimizado (OSD).
4. Estandarizar cada muestra bootstrap y usar los pesos originales para calcular el conjunto WSD de las puntuaciones de las variables latentes. Los pesos unitarios se usan para calcular el conjunto SSD de las puntuaciones de las variables latentes.
5. Para obtener los R^2 de WSD y SSD, reemplazar en el paso 4 cada variable latente por la variable manifiesta. El R^2 resultante del uso de los pesos de la ejecución original será el R^2 sumado ponderado (WSD) y el R^2 sumado simple (SSD) es el que representa el nivel de referencia de los pesos unitarios.
6. Calcular el índice de rendimiento relativo (RPI) usando los pesos originales (R^2 WSD) sobre la regresión sumada simple. (R^2 SSD).

$$RPI = \frac{100 * (R^2_{WSD} - R^2_{SSD})}{R^2_{SSD}}$$

7. Calcular el rendimiento PLS optimizado sumado (PFO) examinando cómo el R^2 WSD difiere del R^2 OSD.

$$PFO = \frac{100 * (R^2_{OSD} - R^2_{WSD})}{R^2_{WSD}}$$

2.4.4. Evaluación del modelo mediante validación cruzada

Las técnicas de validación cruzada basadas en el índice *GoF* nos permiten calcular intervalos de confianza y evaluar la significación estadística de los coeficientes de ruta, de sólo uno, de todos o de un subconjunto de ellos.

Intervalos de confianza

Podemos calcular intervalos de confianza bootstrap para *GoF* y GoF_{rel}.

En ambos casos, la función de distribución acumulada inversa de *GoF* ($\boldsymbol{\phi}_{GoF}$) se aproxima mediante un procedimiento bootstrap: B (generalmente > 100) muestras se extraen del conjunto de datos inicial de N unidades que definen la población de bootstrap. Para cada una de las muestras B, se calcula el índice GoF^b, con $b = 1, ..., B$.

Los valores de GoF^b se utilizan luego para calcular la aproximación de Monte Carlo de la función de distribución acumulada inversa, $\boldsymbol{\phi}^B_{GoF}$. Por lo tanto, es posible calcular los límites del intervalo de confianza empírico a partir de la distribución bootstrap a un nivel de confianza de $1 - \alpha$ usando los percentiles.

Con esto, el intervalo de confianza es:

$$[\boldsymbol{\phi}^B_{GoF}(\alpha/2)\ ,\ \boldsymbol{\phi}^B_{GoF}(1-\alpha/2)]$$

Se ha demostrado que la variabilidad de los valores de GoF se debe principalmente al modelo interno, mientras que la contribución del modelo externo a GoF es muy estable en las diferentes muestras de bootstrap.

Contraste de hipótesis sobre un coeficiente de ruta

El objetivo es contrastar la hipótesis nula de que el coeficiente de ruta genérico β_{qj} es distinto de cero, es decir,

$$\begin{cases} H_0 : \beta_{qj} = 0 \\ H_1 : \beta_{qj} \neq 0 \end{cases}$$

Para ello necesitamos definir un estadístico apropiado y conocer su distribución bajo hipótesis nula. En particular, el índice GoF se usará para probar las hipótesis establecidas, mientras que la distribución bajo hipótesis nula se obtendrá utilizando un procedimiento bootstrap.

Sea GoF_{H_0} el valor GoF bajo hipótesis nula, $\boldsymbol{\Phi}$ la inversa de la función de distribución acumulada de GoF_{H_0}, F la función de distribución acumulada de $\boldsymbol{X}$ y $\boldsymbol{\Phi}^{(B)}$ la aproximación muestral de $\boldsymbol{\Phi}$ para la B muestra bootstrap.
Para aproximar $\boldsymbol{\Phi}$ mediante $\boldsymbol{\Phi}^{(B)}$ necesitamos definir una estimación muestral de F bajo hipótesis nula para la B muestra bootstrap ($\widehat{F}_{H_0^{(b)}}$). Las estimaciones muestrales de F se definen sobre la base de $p(\boldsymbol{x}'_n) = \frac{1}{N}$ con $n = 1, ..., N$ siendo $p(\boldsymbol{x}'_n)$ la probabilidad de extraer la n-ésima observación de la matriz $\boldsymbol{X}$.

El contraste se puede realizar por medio del siguiente algoritmo:

1. Estimar el modelo estructural sobre el conjunto de datos original (que será la población bootstrap) y calcular el índice GoF.
2. Reducir el bloque endógeno de la variable manifiesta $\boldsymbol{X}_j$ de la siguiente forma: $\boldsymbol{X}_{j(q)} = \boldsymbol{X}_j - \boldsymbol{X}_q(\boldsymbol{X}'_q\boldsymbol{X}_q)^{-1}\boldsymbol{X}'_q\boldsymbol{X}_j$
3. Damos B, el número de muestras bootstrap, el cual depende del tamaño muestral, el número de variables manifiestas y de la complejidad del modelo estructural aunque se suele tomar $B \geq 1000$.

Ahora, $\boldsymbol{\Phi}^{(B)}$ se obtiene al repetir B veces el siguiente procedimiento:

Para cada b : $\quad b = 1, 2, ..., B$:

a. Obtener una muestra aleatoria de la estimación de F bajo hipótesis nula.
b. Estimar el modelo bajo la hipótesis nula para la muestra obtenida en el paso anterior.
c. Calcular el valor de GoF, $GoF^{(b)}_{H_0}$.

Finalmente, la decisión sobre la hipótesis nula se toma en función de GoF_{H_0}. En particular, el test se realiza con un nivel de significación α, y si $GoF > \boldsymbol{\Phi}^{(B)}_{(1-\alpha)}$ rechazamos la hipótesis nula siendo $\boldsymbol{\Phi}^{(B)}_{(1-\alpha)}$ el percentil $1-\alpha$ de $\boldsymbol{\Phi}^{(B)}$.

Contraste de hipótesis sobre un conjunto de coeficientes de ruta

El procedimiento anterior se puede generalizar fácilmente para el caso de un subconjunto de coeficientes de ruta o todos ellos al mismo tiempo. Si los coeficientes de ruta se contrastan simultáneamente, entonces esta prueba se puede usar para una evaluación general del modelo. Esta prueba se realiza comparando el modelo predeterminado con los llamados *modelos de referencia*, es decir, el *modelo saturado* y el *modelo de independencia*.
El **modelo saturado** es el modelo menos restrictivo en el que se permiten todas las relaciones estructurales (es decir, todos los coeficientes de ruta son parámetros libres).

El **modelo de independencia** o **modelo nulo** es el modelo más restrictivo sin relaciones entre las variables latentes (es decir, todos los coeficientes de ruta están obligados a ser 0).

En este caso, el contraste viene dado por:

$$\begin{cases} H_0 : \beta_{qj} = 0 \quad \forall q, j \\ H_1 : \text{Al menos un } \beta_{qj} \neq 0 \end{cases}$$

Como antes, tenemos que reducir adecuadamente $\boldsymbol{X}$ para estimar $\boldsymbol{\Phi}^{(B)}$. En particular, cada bloque endógeno $\boldsymbol{X}_j$ tiene que ser reducido de acuerdo con las relaciones estructurales por medio de operadores de proyección ortogonal.

Al tratar con un modelo recursivo, siempre es posible construir bloques que verifiquen la hipótesis nula mediante una secuencia adecuada de reducciones.

Finalmente, la decisión sobre la hipótesis nula se toma en función de GoF_{H_0}. En particular, el test se realiza con un nivel de significación α, y si $GoF > \boldsymbol{\Phi}^{(B)}_{(1-\alpha)}$ rechazamos la hipótesis nula siendo $\boldsymbol{\Phi}^{(B)}_{(1-\alpha)}$ el percentil $1-\alpha$ de $\boldsymbol{\Phi}^{(B)}$.

Al comparar el valor de GoF obtenido para el modelo en la población bootstrap con el $GoF^{(b)}_{H_0}$ obtenido de las muestras bootstrap ($b = 1, 2, ..., B$), un valor empírico del p-valor viene dado por:

$$p - value = \frac{\sum_{b=1}^{B} I_b}{B}$$

donde B es el número de muestras bootstrap e

$$I_b = \begin{cases} 1 & si\ GoF^{(b)}_{H_0} \geq GoF \\ 0 & en\ otro\ caso \end{cases}$$

El procedimiento anterior contrasta la hipótesis nula de que todos los coeficientes de trayectoria son iguales a cero contra la hipótesis alternativa de que al menos uno de ellos es distinto de cero. Al definir una estrategia de reducción adecuada, se pueden realizar pruebas en cualquier subconjunto de coeficientes de ruta. También se pueden definir procedimientos paso a paso para identificar un conjunto de coeficientes significativos.

2.5. Heterogeneidad de los datos en PLS-PM

La heterogeneidad entre las unidades muestrales es un tema importante en el análisis estadístico ya que tratar la muestra como homogénea cuando no lo es, puede afectar a la calidad de los resultados además de conducir a una interpretación sesgada.

2.5.1. Heterogeneidad (tipos y cómo combatirlas)

Los datos se ven afectados por dos tipos de heterogeneidad: la heterogeneidad **observada** y la **no observada**. En el primer caso, la composición de las clases se conoce a priori, mientras que en el segundo caso, la información sobre el número de clases o sobre su composición no está disponible.

En un modelo de ecuaciones estructurales, los dos tipos de heterogeneidad coinciden con la presencia de un factor moderador discreto que, en el primer caso es manifiesto (variable observada) mientras que en el segundo es latente (variable no observada).

Debido a que la heterogeneidad difícilmente puede detectarse utilizando información externa, habitualmente, la heterogeneidad en los modelos de ecuaciones estructurales se trata formando primero grupos sobre la base de variables externas o sobre la base de técnicas de agrupamiento estándar aplicadas a variables manifiestas y/o latentes, y luego usando el análisis de grupos múltiples introducido por Jöreskog y Sörbom.

Sin embargo, la heterogeneidad en los modelos puede no ser necesariamente detectada por variables observadas conocidas que desempeñan el papel de variables moderadoras. Además, las técnicas de agrupación post-hoc en variables manifiestas, o en variables latentes, no tienen en cuenta el propio modelo de ahí la necesidad de un método de agrupamiento basado en la respuesta, donde los grupos obtenidos son homogéneos con respecto al modelo postulado.

Hacer frente a la heterogeneidad en los modelos PLS-PM implica buscar modelos locales caracterizados por parámetros del modelo específico del grupo. Recientemente, se han propuesto varios métodos para tratar la heterogeneidad no observada en el marco PLS-PM. Existen cinco enfoques para manejar la heterogeneidad en el modelado de rutas PLS: la mezcla finita PLS, el modelo de ruta tipológica PLS, el PATHMOX, el agrupamiendo basado en PLS-PM (PLS-PMC) y la Segmentación de Unidades Basada en la Respuesta en el PLS-PM (REBUS-PLS).

2.5.2. El algoritmo REBUS-PLS

El núcleo del algoritmo es una llamada **medida de proximidad (CM)** entre unidades y modelos basados en residuos. La idea detrás de esta definición es que si existen grupos latentes, las unidades que pertenecen al mismo grupo tendrán modelos locales similares. Además, si se asigna una unidad al grupo latente correcto, su rendimiento en el modelo local para ese grupo específico será mejor que el rendimiento de la misma unidad en los otros modelos locales.

La CM utilizada en el algoritmo REBUS-PLS tiene en cuenta tanto los modelos de medida como los estructurales en el procedimiento de clustering o agrupamiento. Para obtener modelos locales que se ajusten mejor que el modelo global, la medida de proximidad elegida se define de acuerdo con la estructura del índice GoF, la única medida disponible de modelo global.

Para la n-ésima unidad del k-ésimo modelo local, es decir, para el modelo latente correspondiente al k-ésimo grupo latente, la medida de proximidad viene dada por:

$$CM_{nk} = \sqrt{\frac{\sum_{q=1}^{Q}\sum_{p=1}^{P_q}\left[\frac{e_{npqk}^2}{Com(\widehat{\xi}_{qk},x_{pq})}\right]}{\frac{\sum_{n}^{N}\sum_{q=1}^{Q}\sum_{p=1}^{P_q}\left[\frac{e_{npqk}^2}{Com(\widehat{\xi}_{qk},x_{pq})}\right]}{(N-t_k-1)}} \times \frac{\sum_{j=1}^{J}\left[\frac{f_{njk}^2}{R^2(\widehat{\xi}_j,\widehat{\xi}_{q:\xi_q\to\xi_j}}\right]}{\frac{\sum_{n}^{N}\sum_{j=1}^{J}\left[\frac{f_{njk}^2}{R^2(\widehat{\xi}_j,\widehat{\xi}_{q:\xi_q\to\xi_j}}\right]}{(N-t_k-1)}}}$$

donde $Com(\widehat{\xi}_{qk}, x_{pq})$ es el índice de comunalidad para la p-ésima variable manifiesta del q-ésimo bloque en el k-ésimo grupo latente, e_{npqk} es el residuo del modelo de medida para la n-ésima unidad en el k-ésimo grupo latente correspondiente a la p-ésima variable manifiesta en el q-ésimo bloque, f_{njk} es el residuo del modelo estructural para la n-ésima unidad en el k-ésimo grupo latente, correspondiente al j-ésimo bloque endógeno, N es el número total de unidades y t_k es el número de componentes extraídas que será siempre igual a 1 porque todos los bloques se suponen que son reflexivos.

El término del lado izquierdo del producto de la definición de CM se refiere a los modelos de medida para los Q bloques en el modelo, mientras que el término del lado derecho se refiere al modelo estructural. Cabe destacar que los residuos de ambos modelos se calculan para cada unidad con respecto a cada modelo local independientemente de la pertenencia de las unidades al grupo latente específico. Al calcular el residuo del k-ésimo modelo latente, se espera que las unidades que pertenecen al k-ésimo grupo latente tengan residuos más pequeños que las unidades que pertenecen a los otros $K-1$ grupos.

La elección de CM como criterio para asignar unidades a los grupos tiene dos ventajas, una es que se puede detectar la heterogeneidad no observada tanto en los modelos de medida como estructurales. Si dos modelos muestran los mismos coeficientes estructurales pero difieren con respecto a uno o más pesos externos en los bloques exógenos, el algoritmo REBUS-PLS es capaz de identificar esta fuente de heterogeneidad, que podría ser de gran importancia en aplicaciones prácticas.Y además, dado que la medida de proximidad se define de acuerdo con la estructura del índice GoF, la otra ventaja es que los modelos locales identificados mostrarán un mejor rendimiento predictivo. El CM es solo el núcleo de un algoritmo iterativo que nos permite obtener un agrupamiento basado en respuestas de las unidades.

De hecho, REBUS-PLS es un algoritmo iterativo que consiste en:

- **Paso 1**: Estimación del modelo global sobre todas las unidades observadas, mediante la realización de un análisis PLS-PM.
- **Paso 2**: Calcular la comunalidad y los residuos estructurales para cada unidad del modelo global.
- **Paso 3**: Calcular el número de grupos (K) y la composición inicial de los mismos mediante un análisis de conglomerados jerárquico sobre los residuos calculados (tanto de los modelos de medida como los estructurales).

- **Paso 4**: Estimar los K modelos locales provisionales con la realización de un análisis PLS-PM en cada grupo. Los parámetros específicos de cada grupo calculados en el paso anterior se utilizan para calcular la comunalidad y los residuos estructurales.
- **Paso 5**: Calcular el CM de cada unidad para cada modelo local.
- **Paso 6**: Asignar cada unidad al modelo local que muestre el menor valor de CM.

Una vez que se actualiza la composición de los grupos, se estiman K nuevos modelos locales por lo que el algoritmo se itera desde el paso 4 al 6 hasta la convergencia, es decir, hasta que se alcanza el umbral de un criterio de parada. Dicho umbral se toma como que menos del 5 % de las unidades cambian de un grupo a otro en el cambio de iteración. De hecho, REBUS-PLS generalmente asegura la convergencia en un pequeño número de iteraciones (menos de 15). También es posible no definir un umbral como un criterio de parada y ejecutar el algoritmo hasta que se formen los mismos grupos en sucesivas iteraciones aunque si el tamaño de muestra es grande, es posible tener unidades que cambien de grupo en sucesivas iteraciones. Esto lleva a obtener una serie de estimaciones de modelos locales que se repiten en iteraciones sucesivas. Para evitar este problema, los autores sugieren siempre definir un criterio de parada.

Una vez que se alcanza la estabilidad en la composición del grupo, se estiman los modelos locales finales. Después, los coeficientes e índices específicos de los grupos se comparan para explicar las diferencias entre los grupos latentes detectados.

La calidad del modelo local obtenido puede evaluarse a través de un nuevo índice llamado **índice de calidad del grupo (GQI)**. Este índice es una reformulación del índice GoF en una perspectiva multigrupo, y también se basa en los residuos. En el caso de que exista un único grupo ($K = 1, \quad n_1 = N$), el índice GQI es igual al GoF pero si se detectan modelos locales con un rendimiento mejor que el global, el índice GQI será más alto que el índice GoF calculado para el modelo global.

La calidad del modelo local también puede evaluarse mediante un **test de permutación** que implica T replicaciones aleatorias del modelo local (manteniendo constantes las proporciones del grupo detectadas por REBUS-PLS) para producir una distribución empírica del índice GQI. El GQI obtenido para el modelo REBUS-PLS se compara con los percentiles de la distribución empírica para decidir si los modelos locales tienen un rendimiento significativamente mejor que el global. Se ha demostrado que, en caso de heterogeneidad no observada y excepto las soluciones outlier, el índice GQI calculado es el valor mínimo obtenido para la distribución empírica del GQI.

Si se dispone de covariantes externas, se puede realizar un análisis ex-post de los grupos detectadas para caracterizar los grupos latentes detectados y mejorar la interpretabilidad de su composición.

Hasta ahora, REBUS-PLS está limitado a los modelos de medida reflexivos porque los residuos del modelo de medida provienen de las regresiones simples entre cada variable manifiesta de un bloque y su correspondiente variable latente.

Capítulo 3

Comparación de modelos PLS-PM

3.1. Diferencias entre modelos PLS-PM

Cuando obtenemos un modelo PLS-PM, podemos plantearnos la cuestión de cómo compararlo con otro modelo diferente en caso de que exista. Para comparar dos modelos PLS-PM tenemos que atender a cuatro tipos de diferencias que pueden surgir entre ambos:

- **Diferencias a nivel de red causal**: Diferencias en la relación causa-efecto entre las variables latentes. Por ejemplo, dos variables latentes pueden estar correlacionadas en un conjunto de datos dados en función de una categoría de una variable, pero pueden no estar correlacionadas en los datos sobre otra de las categorías.
- **Diferencias a nivel estructural**: Diferencias en magnitud de los coeficientes estructurales (coeficientes de ruta). Por ejemplo, dadas dos categorías de una variable, si el objetivo del modelo es obtener el índice de satisfacción puede que en una categoría la satisfacción sea impulsada por una variable mientras que en la otra categoría la satisfacción sea impulsada por otra variable distinta.
- **Diferencias a nivel de medida**: Diferencias en la forma en la que las variables latentes son definidas por sus variables manifiestas. Mientras que una de las variables manifiestas puede ser apropiada para alguna variable latente en un modelo, la misma variable manifiesta puede no ser apropiada en otro modelo distinto.
- **Diferencias a nivel de variables latentes**: Esto implica que el valor de cualquier función de la distribución de las variables latentes a través de distintos modelos puedan ser diferentes.

En principio, ninguno de los niveles anteriores es mejor o peor que los demás. Dado que la principal característica de los modelos PLS-PM se encuentra en la parte del modelo estructural con la estimación de los coeficientes de ruta, se pone énfasis en la comparación de modelos teniendo en cuenta diferencias sólo a nivel estructural. La razón fundamental para centrarse en los coeficientes de ruta se debe al objetivo de los modelos PLS-PM con variables latentes: estimar las relaciones lineales entre dichas variables.

La limitación principal de este punto de vista es que el resto de diferencias están prácticamente ignoradas y además, al centrarnos en diferencias entre coeficientes de ruta, la red estructural debe ser la misma en todos los grupos, de lo contrario estaríamos comparando modelos completamente distintos. Además, para tener una buena comparación de coeficientes de rutas entre los grupos, las variables manifiestas tienen que ser las mismas entre los modelos.

3.2. Comparación de grupos

Como nos basaremos en las diferencias a nivel estructural, tendremos que centrar la atención en los coeficientes de ruta aunque también en los pesos externos, las cargas, el R^2 y el índice GoF. Además supondremos que en el conjunto de datos hay una variable (cualitativa o cuantitativa) que permite definir los grupos.

Aunque la comparación también se denomina *comparación de grupos múltiples* o *análisis de grupos múltiple* se realiza mediante análisis multi-grupo aunque realmente gran parte de las veces el análisis se concentra en sólo dos grupos. Realmente debería ser así en el caso en que la variable que los define sólo lo haga en base a dos categorías. En caso de sean más de dos, habría que actuar realizando varios análisis en grupos de dos cubriendo todas las posibles combinaciones. Por ejemplo, para hacer un análisis de grupos múltiple cuando la variable puede agruparse en tres categorías tendremos que comparar:

- Categoría 1 -vs- Categoría 2 y Categoría 3
- Categoría 2 -vs- Categoría 1 y Categoría 3
- Categoría 3 -vs- Categoría 1 y Categoría 2

Dentro del PLS-PM, la comparación de grupos puede dividirse en dos categorías dependiendo de la naturaleza de la variable que permite dividir en grupos el conjunto de datos: los **métodos de remuestreo** y los **efectos moderadores**.

3.3. Métodos de remuestreo

Estos métodos consisten en la realización de técnicas de remuestreo para comprobar si existe o no diferencia entre grupos que son diferenciados mediante una *variable cualitativa* o categórica. Las técnicas más usuales son el **t-test bootstrap** y **el procedimiento de permutación**.

3.3.1. El t-test bootstrap

Este enfoque de remuestreo implica el uso de un t-test basado en el remuestreo bootstrap. Consiste en separar los datos en grupos en función de la variable categórica y luego tomar muestras bootstrap con reemplazamiento para cada grupo. Los coeficientes de ruta se calculan en cada muestra bootstrap y las estimaciones de los errores estándar son tratados en un sentido paramétrica a través del t-test.

Supongamos que tenemos dos grupos G_1 y G_2 con muestras de tamaño n_1 y n_2 respectivamente. Para saber si podemos considerar iguales los coeficientes de ruta de ambos grupos ($\beta_{ji}^{G_1}$ y $\beta_{ji}^{G_2}$) realizamos el siguiente procedimiento:

1. Calcular un modelo PLS-PM para cada grupo y obtener los coeficientes de ruta $\beta_{ji}^{G_1}$ y $\beta_{ji}^{G_2}$.

2. Separar los datos en dos grupos y obtener para cada uno de ellos las muestras bootstrap (aproximadamente unas 200).

3. Para cada muestra, calcular un modelo PLS-PM para obtener los coeficientes de ruta del remuestreo.

4. Calcular la estimación del error estándar.

5. Usar dicha estimación en un sentido paramétrico por medio de un t-test.
   El estadístico del t-test sigue una t-Student con $n_1 + n_2 - 2$ grados de libertad y viene dado por:

$$t = \frac{\hat{\beta}_{ji}^{G_1} - \hat{\beta}_{ji}^{G_2}}{(\sqrt{\frac{1}{n_1} + \frac{1}{n_2}})Sp}$$

   donde Sp es el estimador de varianza global y se obtiene de la siguiente forma:

$$Sp = \sqrt{\frac{(n_1 - 1)^2}{(n_1 + n_2 - 2)} SE_{G_1}^2 + \frac{(n_2 - 1)^2}{(n_1 + n_2 - 2)} SE_{G_2}^2}$$

   siendo SE_{G_1} y SE_{G_2} los errores estándar bootstrap de cada grupo.

Hay que tener en cuenta que este procedimiento depende de las hipótesis del t-test, es decir, que datos sigan una distribución normal y que el tamaño de la muestra sea el mismo para cada grupo. Estos procedimientos son robustos pero cuando los datos no tienen distribuciones simétricas o el tamaño de los grupos son muy diferentes, presentan aplicaciones limitadas. Suele ocurrir cuando tenemos datos con la distribución de las variables fuertemente sesgadas. A menos que las muestras sean bastante grandes, el rendimiento de la prueba será escaso.

3.3.2. El procedimiento de permutación

Esta técnica se basa en los procedimientos de aleatorización o de permutación.
En comparación con el remuestreo bootstrap (en el que las muestras se obtienen con reemplazamiento), las muestras de permutación se obtienen sin reemplazamiento.
La premisa básica de este tipo de técnica es utilizar el supuesto de que es posible que todos los grupos sean equivalentes, y que cada miembro del grupo es el mismo antes de que comience el muestreo. A partir de esto, se puede calcular un estadístico y luego observar el grado en que este estadístico es especial viendo qué tan probable sería que las asignaciones de los grupos habían sido mezcladas.

Supongamos que tenemos dos grupos G_1 y G_2 con muestras de tamaño n_1 y n_2 respectivamente. El test de permutación determina si la diferencia observada entre los coeficientes de ruta ($\beta_{ji}^{G_1}$ y $\beta_{ji}^{G_2}$) es lo suficientemente grande como para rechazar la hipótesis nula H_0 de que los dos grupos pueden considerarse los mismos.

El test consiste en lo siguiente:

1. Calcular el estadístico del test para los datos. En nuestro caso es la diferencia de los coeficientes de ruta entre los dos grupos, es decir, $\beta_{ji}^{G_1} - \beta_{ji}^{G_2}$

2. Unir las observaciones de los grupos en un único grupo.

3. Permutar repetidamente los datos de una forma aleatoria consistente. Cada permutación implica: dividir los datos en dos grupos de tamaño n_1 y n_2, estimar los modelos PLS para cada grupo y calcular el estadístico.

4. Ordenar las diferencias y comprobar si el estadístico original está contenido en el 95 % de los valores ordenados. Si no es así, rechazamos la hipótesis nula de que los grupos son iguales a un nivel de significación del 5 %.

Las ventajas de este procedimiento son:

- Es una prueba de distribución libre que no requiere supuestos paramétricos

- Se aplican a una variedad de estadísticas, no sólo a las estadísticas que tienen una distribución sencilla bajo la hipótesis nula

- Pueden dar p-valores muy exactos, independientemente de la distribución y tamaño de la población (si se usan suficientes permutaciones).

3.4. Efectos moderadores

Los efectos moderadores, también llamados efectos de interacción, representan la influencia que una tercera variable (M) (cuantitativa, que permite dividir el conjunto de datos), tiene sobre la relación entre una variable independiente (X) y una dependiente (Y). Una representación gráfica de este concepto aparece en la figura 3.1.

En el marco del PLS-PM, para estudiar este tipo de efectos se tratan las variables moderadoras como variables latentes. Éstas deben considerarse sólo en el modelo estructural del modelo PLS-PM.

Dentro de este método, podemos estudiar los efectos moderadores mediante el uso de los siguientes cuatro enfoques: el **de indicadores**, el del **modelo de ruta en dos etapas**, el de la **regresión en dos etapas** y el de la **variable categórica**.

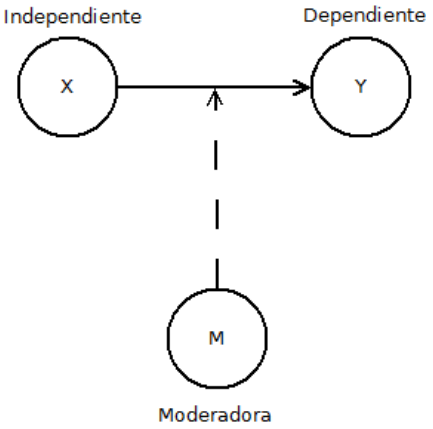

Figura 3.1: Diagrama del efecto moderador

3.4.1. Enfoque del producto de indicadores

En este enfoque se busca crear una nueva variable latente que representa la interacción entre la variable exógena y la variable moderadora. La nueva variable latente se construye mediante productos de las variables manifiestas de la variable latente independiente y las variables manifiestas de la variable latente moderadora. Estos términos de productos son las variables manifiestas que utilizamos para el término de interacción latente en el modelo estructural. Esta interacción latente se toma como una variable latente reflexiva. Para entender este enfoque, usaremos el ejemplo que se muestra en la figura 3.2. En este ejemplo tenemos una variable exógena X, una moderadora M (también exógena) y una endógena Y. Cada una de las variables exógenas tiene dos variables manifiestas mientras que la endógena tiene tres. Para estudiar el efecto moderador creamos el término de interacción latente XM cuyas variables manifiestas serán los productos de las variables manifiestas entre X y M.

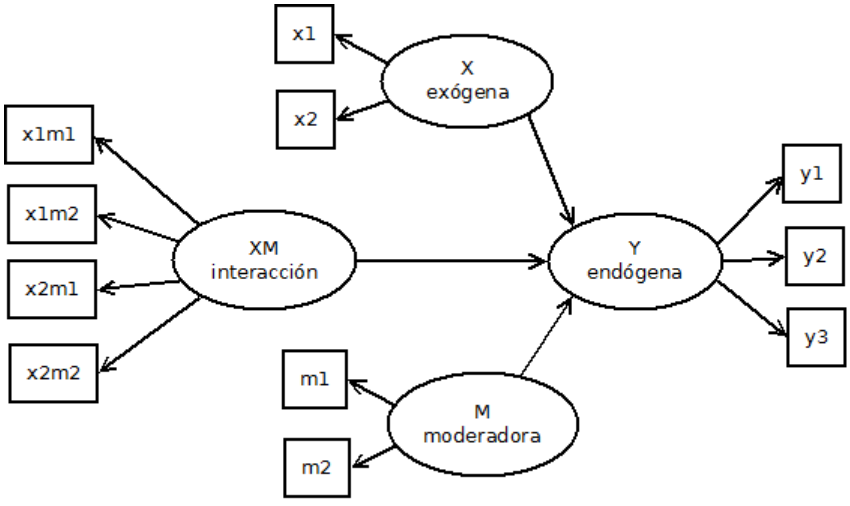

Figura 3.2: Diagrama del enfoque del producto de indicadores

3.4.2. Enfoque del modelo de ruta en dos etapas

Este enfoque consta de dos etapas:

- **Etapa 1**: Consiste en aplicar un análisis PLS-PM sin el término de interacción.

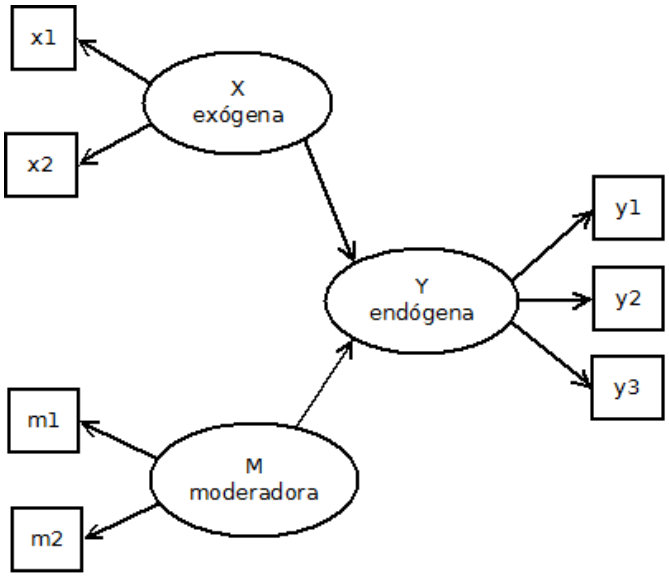

Figura 3.3: Primera etapa del enfoque del modelo de ruta en dos etapas

- **Etapa 2**: Consiste en tomar las puntuaciones obtenidas en la primera etapa para crear el término de interacción, y a continuación, realizar un segundo análisis PLS-PM incluyendo las puntuaciones como variables manifiestas de las variables latentes.

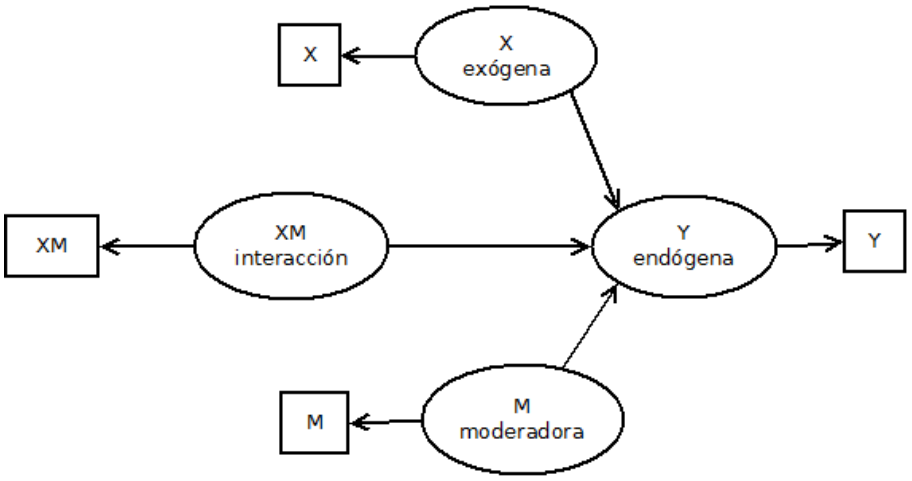

Figura 3.4: Segunda etapa del enfoque del modelo de ruta en dos etapas

3.4.3. Enfoque de la regresión en dos etapas

Este enfoque se realiza en dos etapas:

- **Etapa 1**: Aplicar un análisis PLS-PM sin el término de interacción.
- **Etapa 2**: Tomar las puntuaciones obtenidas en la primera etapa y aplicar un análisis de regresión con las puntuaciones de la primera etapa.

3.4.4. Enfoque de la variable categórica

Este enfoque tiene lugar cuando la variable moderadora es una variable categórica aunque desde un punto de vista teórico, es cuestionable decir que sea una variable latente. Sin embargo, por razones prácticas, podemos tratarlas como tal. La idea es utilizar tantas variables ficticias como las dados por el número de categorías que tenga la variable moderadora menos 1 y con dichas variables ficticias obtener productos de interacción. Para estimar el efecto moderador, tenemos que crear nuevos términos de interacción latentes cuyas variables manifiestas sean los productos de los indicadores de las variables exógenas y moderadoras. Gráficamente puede verse en la figura 3.5:

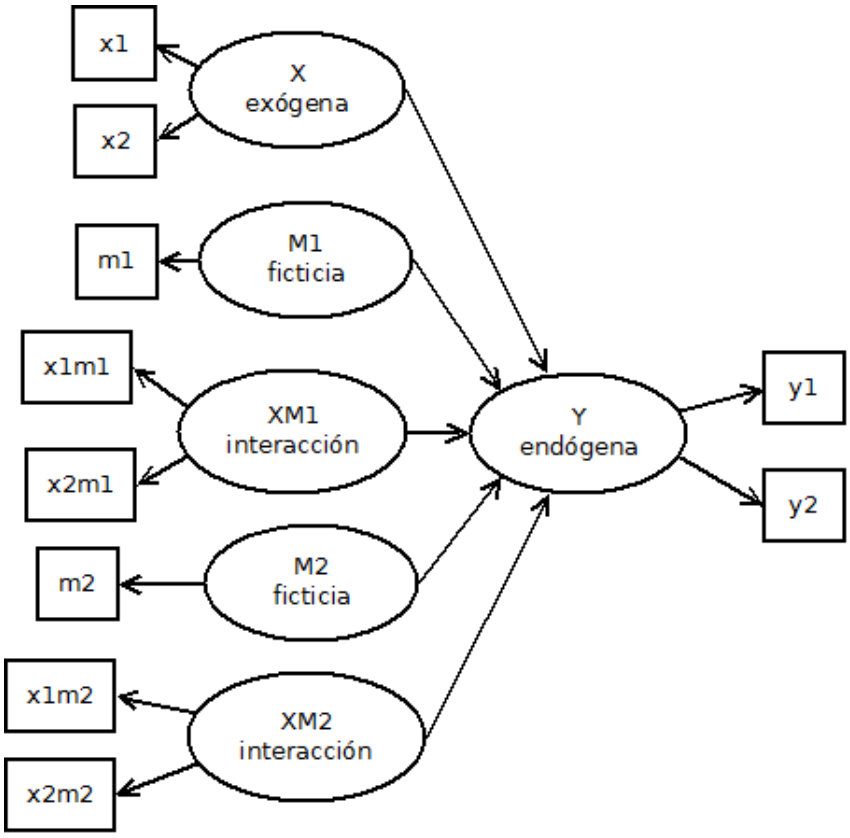

Figura 3.5: Diagrama del enfoque de la variable categórica

Capítulo 4

Aplicación del modelo PLS-PM en R

4.1. Herramientas

4.1.1. Paquete plspm

plspm es un paquete R para realizar un análisis PLS-PM. Está disponible de forma gratuita desde la Red Integral de Archivos R, más conocida como CRAN. Toda la información en relación al mismo se puede consultar en https://cran.r-project.org/web/packages/plspm/plspm.pdf. La versión principal del paquete es la disponible en CRAN. Puede instalarlo utilizando la función *install.packages()*:

```
install.packages("plspm")
```

Una vez que se haya instalado el paquete *plspm*, habría que ejecutar la función *library()* para cargar el paquete en la sesión de trabajo:

```
library(plspm)
```

El paquete `plspm` viene con una serie de funciones para realizar una serie de diferentes tipos de análisis. La función principal es la función `plspm()` que está diseñada para ejecutar un análisis completo de PLS-PM. Una versión modificada es la función `plspm.fit()` que está destinada a realizar un análisis PLS-PM con resultados limitados.

Las funciones accesorias de plspm () son las funciones de trazado y resumen. El método `plot()` es un wraper de las funciones `innerplot()` y `outerplot()` que le permiten visualizar los resultados del modelo interno y externo, respectivamente. A su vez, la función `summary()` mostrará los resultados en un formato similar a otro software estándar para PLS-PM.

En tercer lugar, tenemos la función `plspm.groups()` que le permite comparar dos modelos. Esta función ofrece dos opciones para hacer la comparación: una prueba t de bootstrap y una prueba de permutación no paramétrica.

En cuarto lugar, está el conjunto de funciones dedicadas a la detección de clases latentes mediante el uso de REBUS-PLS.

Por último, el paquete `plspm` contiene conjuntos de datos: `satisfaction`, `mobile`, `spainfoot`, `soccer`, `offense`, `technology`, `oranges`, `wines`, `arizona`, `russett`, `russa`, `russb`, y `sim.data`.

4.1.2. Función plspm()

La función que realiza un análisis PLS-PM completo es `plspm()`. Para ver la información relevante a esta función o bien ejecutamos la siguiente instrucción:

```
help(plspm)
```

o bien, accedemos al siguiente enlace http://cran.r-project.org/web/packages/plspm/plspm.pdf.

La función `plspm()` tiene 14 argumentos los cuales podemos dividirlos en tres conjuntos dependiendo de su función: el primer conjunto de argumentos son los parámetros usados para definir el modelo PLS, el segundo conjunto de parámetros se corresponde con los relacionados con el algoritmo PLS y el tercer conjunto son opciones adicionales para métodos de remuestreo. Los argumentos son los siguientes:

- **Parámetros para definir el PLS-PM**
  `Data` → Contiene el conjunto de datos que queremos analizar
  `pathmatrix` → Define el modelo interno
  `blocks` → Lista que define los bloques de variables del modelo externo
  `scaling` → Lista que define la escala de medida de las variables para datos no métricos
  `modes` → Vector que define la forma de medida de cada bloque
- **Parámetros relacionados con el algoritmo PLS-PM**
  `scheme` → Esquema de cálculo de los pesos internos
  `scaled` → Indica si los datos deben ser estandarizados
  `tol` → Umbral de tolerancia para comprobar la convergencia en el proceso iterativo del algoritmo
  `maxiter` → Máximo número de iteraciones
  `plscomp` → Incida el número de componentes PLS por bloque cuando trabajamos con datos no métricos
- **Parámetros adicionales**
  `boot.val` → Indica si debemos realizar validación bootstrap
  `br` → Número de muestras bootstrap
  `plsr` → Indica si los coeficientes de ruta deben ser calculados por regresión PLS
  `dataset` → Indica si la matriz de datos debe ser recuperada

Aunque la función tenga 14 argumentos, en la práctica sólo se suelen usar pocos de ellos siendo los más importantes los tres primeros (que son los que definen el modelo y no tiene valores por defecto): `Data, pathmatrix` *y* `blocks`. Además, el orden en el que aparecen no puede ser alterado.

4.2. Análisis PLS-PM básico

4.2.1. Datos

Los datos que usaremos como ejemplo son los correspondientes a 20 equipos de la Liga Española de Fútbol Profesional (LPF) sobre los que se han medido 14 variables que son las siguientes:

Variable	Descripción
GSH	Nº total de goles marcados como local
GSA	Nº total de goles marcados como visitante
SSH	Porcentaje de partidos en los que se han marcado goles como local
SSA	Porcentaje de partidos en los que se han marcado goles como visitante
GCH	Nº total de goles recibidos como local
GCA	Nº total de goles recibidos como visitante
CSH	Porcentaje de partidos sin recibir goles como local
CSA	Porcentaje de partidos sin recibir goles como visitante
WMH	Nº total de partidos ganados como local
WMA	Nº total de partidos ganados como visitante
LWR	Racha más larga de partidos ganados
LWRL	Racha más larga de partidos sin perder
YC	Nº total de tarjetas amarillas
RC	Nº total de tarjetas rojas

La lectura de datos en R la realizamos con la siguiente instrucción:

```
data(spainfoot) ; datos=spainfoot
```

4.2.2. Descripción del modelo

En nuestro caso, el objetivo que buscamos sobre estos datos es obtener el **índice de éxito** de cada uno de los equipos. Para ello, primero tomaremos el siguiente modelo:

*A mayor calidad en el **Ataque** así como en la **Defensa**, mayor será el **Éxito**.*

De forma matemática, podemos expresarlo como la siguiente combinación lineal:

$$\text{Éxito} = b_1 Ataque + b_2 Defensa$$

En este modelo, las variables **Ataque**, **Defensa** y **Éxito** son variables latentes. Las variables manifiestas asociadas a cada una de ellas se muestran en el **diagrama de rutas**, en el cual las variables manifiestas se representan con un rectángulo, las latentes con una elipse y las relaciones entre ambas con flechas. Además, se señalan los dos submodelos que lo componen: el modelo estructural (o interno) y el modelo de medida (o externo).

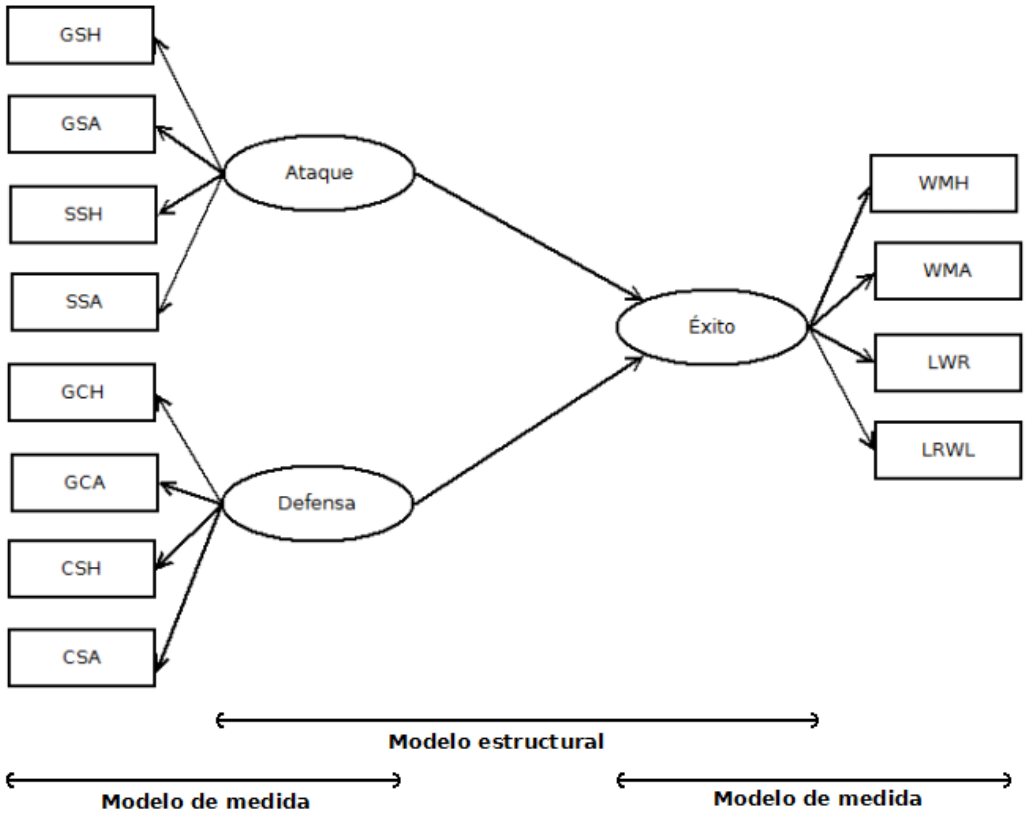

Figura 4.1: Diagrama de rutas (LFP). Análisis básico.

4.2.3. Algoritmo PLS-PM

Ahora, pasamos a realizar la estimación de las puntuaciones de las variables latentes y la cuantificación de las relaciones en el modelo. Realmente no sólo obtendremos el índice de éxito que era nuestro objetivo principal sino también un índice de ataque y de defensa. Como ya disponemos de los datos, pasamos a construir el modelo interno y el modelo externo.

Matriz de ruta o matriz del modelo interno

El modelo interno se representa mediante una matriz que debe ser una matriz booleana triangular inferior, es decir, una matriz cuadrada donde la diagonal y los elementos superiores son cero y el resto ceros o unos.
La interpretación de esa matriz es la siguiente: los ceros en la diagonal significan que una variable latente no se afecta a sí misma, los ceros de encima de la diagonal implican que el PLS-PM sólo usa modelos recursivos (sin lazos) y debajo de la diagonal, un 1 en la celda (i,j) significa que la columna j afecta a la fila i.

Filas de la matriz

```
Ataque=c(0,0,0)
Defensa=c(0,0,0)
Exito=c(1,1,0)
```

Matriz de ruta

```
matriz=rbind(Ataque,Defensa,Exito) ; colnames(matriz)=rownames(matriz)
knitr::kable(matriz, booktabs = TRUE,
             caption = "Matriz de ruta. Análisis básico.")
```

Cuadro 4.2: Matriz de ruta. Análisis básico.

	Ataque	Defensa	Exito
Ataque	0	0	0
Defensa	0	0	0
Exito	1	1	0

Representación de la matriz de ruta

```
innerplot(matriz)
```

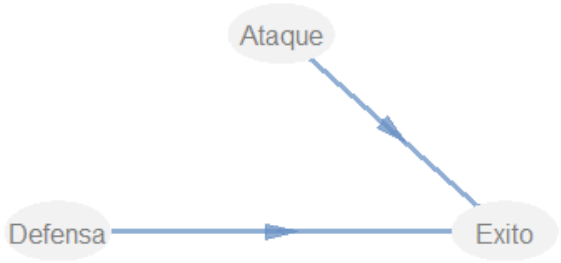

Figura 4.2: Modelo interno. Análisis básico.

Lista del modelo externo

El modelo externo se define mediante una lista con tantos elementos como variables latentes hay siendo cada uno de los elementos un vector de índices que indica las variables asociadas a cada variable latente. De forma alternativa, también podemos utilizar los nombres de las variables en lugar de los índices.

```
bloques=list(1:4,5:8,9:12)
```

Vector de tipos

Por defecto, la función *plspm()* asume que la forma de medir las variables latentes es la forma reflexiva conocida como *modo A*, el modo formativo se conoce como *modo B*.

El vector de tipos de medida que usa la función `plspm()` como argumento es un vector de caracteres (`"A"` o `"B"`) con tantos elementos como variables latentes haya en el modelo.

```
modos=c("A","A","A")
```

Resultados

Ahora, ya tenemos todo lo necesario para ejecutar la función *plspm*.

```
(pls=plspm(datos,matriz,bloques,modos))
```

```
Partial Least Squares Path Modeling (PLS-PM)
---------------------------------------------
   NAME             DESCRIPTION
1  $outer_model     outer model
2  $inner_model     inner model
3  $path_coefs      path coefficients matrix
4  $scores          latent variable scores
5  $crossloadings   cross-loadings
6  $inner_summary   summary inner model
7  $effects         total effects
8  $unidim          unidimensionality
9  $gof             goodness-of-fit
10 $boot            bootstrap results
11 $data            data matrix
---------------------------------------------
You can also use the function 'summary'
```

Los argumentos de salida de esta función son los que aparecen en la tabla anterior y haciendo `pls$` se pueden visualizar así como un resumen, si se ejecuta la siguiente instrucción:

```
summary(pls)
```

4.2.4. Interpretación de resultados

La interpretación de los resultados obtenidos podemos realizarla en dos etapas: la evaluación del modelo de medida y la evaluación del modelo estructural, en ese orden.

Valoración del modelo de medida

1. Unidimensionalidad de las variables manifiestas

En PLS-PM hay tres índices para comprobar la unidimensionalidad: el alpha de Cronbach, el rho de Dillon-Goldstein y el primer autovalor de la matriz de correlaciones de las variables manifiestas. Todos ellos se pueden obtener ejecutando la siguiente instrucción:

```
pls$unidim
```

Cuadro 4.3: Tabla para comprobar la unidimensionalidad. Análisis básico.

	Mode	MVs	C.alpha	DG.rho	eig.1st	eig.2nd
Ataque	A	4	0.8905919	0.9245608	3.017160	0.7923055
Defensa	A	4	0.0000000	0.0260168	2.393442	1.1752781
Exito	A	4	0.9165491	0.9423287	3.217294	0.5370492

En la tabla de la salida anterior, se obtienen estos índices para cada variable latente y en la primera columna aparece el tipo de medida, en la segunda el número de variables manifiestas asociadas a cada variable latente, en la tercera aparece el alpha de Cronbach, en la cuarta el rho de Dillon-Goldstein y en la quinta y sexta el primer y segundo autovalor de la matriz de correlaciones de las variables manifiestas, respectivamente.

Ahora, obtendremos cada uno de los índices de unidimensionalidad:

- **Alpha de Cronbach**

```
pls$unidim[,3,drop=FALSE]
```

Cuadro 4.4: Alpha de Cronbach. Análisis básico.

	C.alpha
Ataque	0.8905919
Defensa	0.0000000
Exito	0.9165491

En este caso, `Ataque` y `Éxito` se consideran unidimensionales mientras que `Defensa` no.

· **Rho de Dillon-Goldstein**

```
pls$unidim[,4,drop=FALSE]
```

Cuadro 4.5: Rho de Dillon-Goldstein. Análisis básico.

	DG.rho
Ataque	0.9245608
Defensa	0.0260168
Exito	0.9423287

Podemos considerar que `Ataque` y `Éxito` son unidimensionales mientras que `Defensa` no lo es.

· **Primer autovalor de la matriz de correlaciones de las variables manifiestas**
En el ejemplo, obtenemos primer y segundo autovalor con la siguiente instrucción:

```
pls$unidim[,5:6]
```

Cuadro 4.6: Primer autovalor. Análisis básico.

	eig.1st	eig.2nd
Ataque	3.017160	0.7923055
Defensa	2.393442	1.1752781
Exito	3.217294	0.5370492

Por tanto, `Ataque` y `Éxito` los consideramos unidimensionales mientras que `Defensa` no.

Nota
El bloque de *Defensa* no es unidimensional ya que las cargas de las variables manifiestas `CSH` y `CSA` son negativas, es decir, que las correlaciones son negativas. Veamos las cargas de este bloque en forma de tabla:

```
subset(pls$outer_model,block=="Defensa")
```

Cuadro 4.7: Cargas bloque defensa. Análisis básico.

	name	block	weight	loading	communality	redundancy
5	GCH	Defensa	-0.1087380	0.4836561	0.2339232	0
6	GCA	Defensa	-0.3914625	0.8759007	0.7672021	0
7	CSH	Defensa	0.3273741	-0.7463736	0.5570735	0
8	CSA	Defensa	0.4035138	-0.8926150	0.7967615	0

Podemos ver que los pesos y las cargas de las variables de este bloque tienen signos contrarios de ahí que no sean adecuados ni el alpha de Cronbach (porque las correlaciones deben ser positivas) y el rho de Dillon-Goldstein.
Además, es contradictorio que los coeficientes de ruta entre *Éxito* y *Defensa* sean negativos pero eso es consecuencia de que los signos de los pesos y las cargas sean diferentes.

La solución a este problema es cambiar el signo de las variables `GCH` y `GCA` (por la definición) pero lo haremos creando nuevas variables que se corresponden con las opuestas (en signo) de las anteriores.
A continuación, realizaremos el estudio anterior pero con las nuevas variables:

```
datos$NGCH=-1*datos$GCH
datos$NGCA=-1*datos$GCA
bloques_new=list(1:4,c(15,16,7,8),9:12)
```

```
pls_new=plspm(datos,matriz,bloques_new,modes=modos)
```

```
pls_new$unidim
```

Cuadro 4.8: Tabla para comprobar la unidimensionalidad. Análisis básico (modificado).

	Mode	MVs	C.alpha	DG.rho	eig.1st	eig.2nd
Ataque	A	4	0.8905919	0.9245608	3.017160	0.7923055
Defensa	A	4	0.7717552	0.8548914	2.393442	1.1752781
Exito	A	4	0.9165491	0.9423287	3.217294	0.5370492

Ahora, todas las variables manifiestas son unidimensionales.

2. Comprobar que las variables manifiestas están bien explicadas por su variable latente

Para ello usaremos las cargas y las comunalidades que se obtienen en la tercera y cuarta columna del resultado de la ejecución de este comando:

```
pls_new$outer_model
```

Cuadro 4.9: Modelo externo. Análisis básico.

name	block	weight	loading	communality	redundancy
GSH	Ataque	0.3366288	0.9379508	0.8797516	0.0000000
GSA	Ataque	0.2819426	0.8620921	0.7432028	0.0000000
SSH	Ataque	0.2892600	0.8408383	0.7070091	0.0000000
SSA	Ataque	0.2395953	0.8262994	0.6827707	0.0000000
NGCH	Defensa	0.1087511	0.4836811	0.2339474	0.0000000
NGCA	Defensa	0.3914458	0.8758874	0.7671787	0.0000000
CSH	Defensa	0.3273984	0.7463952	0.5571058	0.0000000
CSA	Defensa	0.4035029	0.8926037	0.7967413	0.0000000
WMH	Exito	0.2308947	0.7755066	0.6014105	0.5145463
WMA	Exito	0.3029557	0.8863664	0.7856455	0.6721715
LWR	Exito	0.2821408	0.9686187	0.9382222	0.8027110
LRWL	Exito	0.2957720	0.9437100	0.8905885	0.7619572

3.Evaluar el grado en que una variable latente es diferente a otras

Para ello, tenemos que evaluar las cargas cruzadas que para obtenerlas en R hacemos:

```
pls_new$crossloadings
```

Cuadro 4.10: Cargas cruzadas. Análisis básico.

name	block	Ataque	Defensa	Exito
GSH	Ataque	0.9379508	0.5159459	0.8977255
GSA	Ataque	0.8620921	0.3390753	0.7519205
SSH	Ataque	0.8408383	0.4139246	0.7713852
SSA	Ataque	0.8262994	0.3361441	0.6390027
NGCH	Defensa	0.1305182	0.4836811	0.1597542
NGCA	Defensa	0.4621555	0.8758874	0.5751234
CSH	Defensa	0.3188118	0.7463952	0.4809669
CSA	Defensa	0.4214867	0.8926037	0.5928288
WMH	Exito	0.7085904	0.4226212	0.7755066
WMA	Exito	0.7730504	0.7114645	0.8863664
LWR	Exito	0.8444037	0.5380136	0.9686187
LRWL	Exito	0.8600578	0.5891707	0.9437100

Valoración del modelo estructural

Una vez evaluada la calidad del modelo de medida, pasamos a evaluar la calidad del modelo estructural estudiando los resultados obtenidos en cada regresión de las ecuaciones estructurales. Esto lo hacemos mirando:

```
pls_new$inner_model
```

Cuadro 4.11: Modelo estructural. Análisis básico.

	Estimate	Std. Error	t value	Pr(>|t|)
Intercept	0.0000000	0.0921744	0.000000	1.0000000
Ataque	0.7572649	0.1043991	7.253561	0.0000013
Defensa	0.2836035	0.1043991	2.716534	0.0146599

La calidad del modelo estructural se mide en base al coeficiente de determinación R^2 y el índice de redundancia. Estos índices se pueden obtener en R de la siguiente forma:

```
pls_new$inner_summary
```

En esta tabla, para cada variable latente tenemos el tipo de variable que es (endógena o exógena), el coeficiente de determinación (sólo para variables endógenas), la comunalidad media (cuánta variabilidad es reproducible por la variable latente) y la redundandia media (sólo para variables endógenas).

Cuadro 4.12: Tabla para evaluar la calidad del modelo estructural. Análisis básico.

	Type	R2	Block_Communality	Mean_Redundancy	AVE
Ataque	Exogenous	0.0000000	0.7531835	0.0000000	0.7531835
Defensa	Exogenous	0.0000000	0.5887433	0.0000000	0.5887433
Exito	Endogenous	0.8555659	0.8039667	0.6878465	0.8039667

Valoración del modelo global

Se utiliza el coeficiente de bondad de ajuste (GoF) y se considera una pseudo-medida de bondad de ajuste tanto en el modelo estructural como en el de medida.
En R, podemos obtenerlo como:

```
pls_new$gof
```

```
[1] 0.7822944
```

Bootstrap

El procedimiento de remuestreo bootstrap en PLS-PM se usa para estimar la precisión de los parámetros estimados. En R por defecto usa 100 muestras bootstrap pero podemos especificar cuántas muestras queremos cambiando el argumento `br`

```
plsboot=plspm(datos,matriz,bloques_new,modes=modos,boot.val = TRUE,br=200)
plsboot$boot
```

```
$weights
              Original  Mean.Boot  Std.Error     perc.025  perc.975
Ataque-GSH   0.3366288 0.34093259 0.04619775  0.281354438 0.4522748
Ataque-GSA   0.2819426 0.27181737 0.03365943  0.183624967 0.3177575
Ataque-SSH   0.2892600 0.29892422 0.05258311  0.226510605 0.4246844
Ataque-SSA   0.2395953 0.23546540 0.04531520  0.117179243 0.2947361
Defensa-NGCH 0.1087511 0.07182956 0.18976866 -0.360364401 0.3120766
Defensa-NGCA 0.3914458 0.37441444 0.06913751  0.254363718 0.5180486
Defensa-CSH  0.3273984 0.29353797 0.09420586  0.007821648 0.4355119
Defensa-CSA  0.4035029 0.40496054 0.08627954  0.254627965 0.5555926
Exito-WMH    0.2308947 0.22888160 0.03924463  0.156107604 0.2895486
Exito-WMA    0.3029557 0.30289902 0.03231743  0.259748573 0.3648260
Exito-LWR    0.2821408 0.28378927 0.01663817  0.259537015 0.3203770
Exito-LRWL   0.2957720 0.30070473 0.03065894  0.262128590 0.3885354

$loadings
              Original Mean.Boot  Std.Error   perc.025  perc.975
Ataque-GSH   0.9379508 0.9417165 0.02357484  0.8895945 0.9759766
Ataque-GSA   0.8620921 0.8310363 0.12448338  0.4925898 0.9517100
Ataque-SSH   0.8408383 0.8539010 0.04901127  0.7515134 0.9316986
Ataque-SSA   0.8262994 0.8022941 0.12202912  0.4464451 0.9451410
Defensa-NGCH 0.4836811 0.4069006 0.34483569 -0.4381025 0.8344928
Defensa-NGCA 0.8758874 0.8135662 0.31286866 -0.8766916 0.9590623
Defensa-CSH  0.7463952 0.6888774 0.20790036  0.1258990 0.9338453
```

```
Defensa-CSA  0.8926037 0.8360119 0.31440759 -0.7361238 0.9663238
Exito-WMH    0.7755066 0.7679673 0.12911056  0.4136910 0.9127930
Exito-WMA    0.8863664 0.8832127 0.04890583  0.7738323 0.9498485
Exito-LWR    0.9686187 0.9583582 0.03781008  0.8179229 0.9873346
Exito-LRWL   0.9437100 0.9380537 0.03257175  0.8494914 0.9768278

$paths
                  Original Mean.Boot  Std.Error   perc.025  perc.975
Ataque -> Exito  0.7572649 0.7426234 0.08875876  0.5410022 0.9047589
Defensa -> Exito 0.2836035 0.2807979 0.13961701 -0.2898210 0.4771405

$rsq
       Original Mean.Boot  Std.Error  perc.025  perc.975
Exito 0.8555659  0.876415 0.06141963 0.7312587 0.9691953

$total.efs
                   Original Mean.Boot  Std.Error   perc.025  perc.975
Ataque -> Defensa 0.0000000 0.0000000 0.00000000  0.0000000 0.0000000
Ataque -> Exito   0.7572649 0.7426234 0.08875876  0.5410022 0.9047589
Defensa -> Exito  0.2836035 0.2807979 0.13961701 -0.2898210 0.4771405
```

Esto nos proporciona la siguiente lista de resultados:
- los pesos externos (`plsboot$boot$weights`)
- las cargas (`plsboot$boot$loadings`)
- los coeficientes de ruta (`plsboot$boot$paths`)
- el coeficiente de determinación R^2 (`plsboot$boot$rsq`)
- los efectos totales (`plsboot$boot$total.efs`)

Cada uno de ellos es una matriz con cinco columnas que contienen: los valores originales de los parámetros, los valores medios del bootstrap, el error bootstrap estándar, el extermo inferior y superior del intervalo de confianza bootstrap al 95 %.

4.3. Análisis PLS-PM completo

Este ejemplo está basado en un caso particular de obtención de una medida de satisfacción del cliente siendo esta una de las principales aplicaciones del análisis PLS-PM.

Entre todas las medidas disponibles de satisfacción del cliente, podemos encontrar el Índice Europeo de Satisfacción del Cliente (ECSI). El modelo ECSI está diseñado para medir las relaciones causa-efecto desde los antecedentes de Satisfacción del Cliente hasta sus consecuencias y se ilustra en la figura 4.3.

Los antecedentes o factores que afectan a la Satisfacción del Cliente son: Reputación, Expectativas, Calidad Percibida y Valor Percibido mientras que las consecuencias son Lealtad y Quejas.

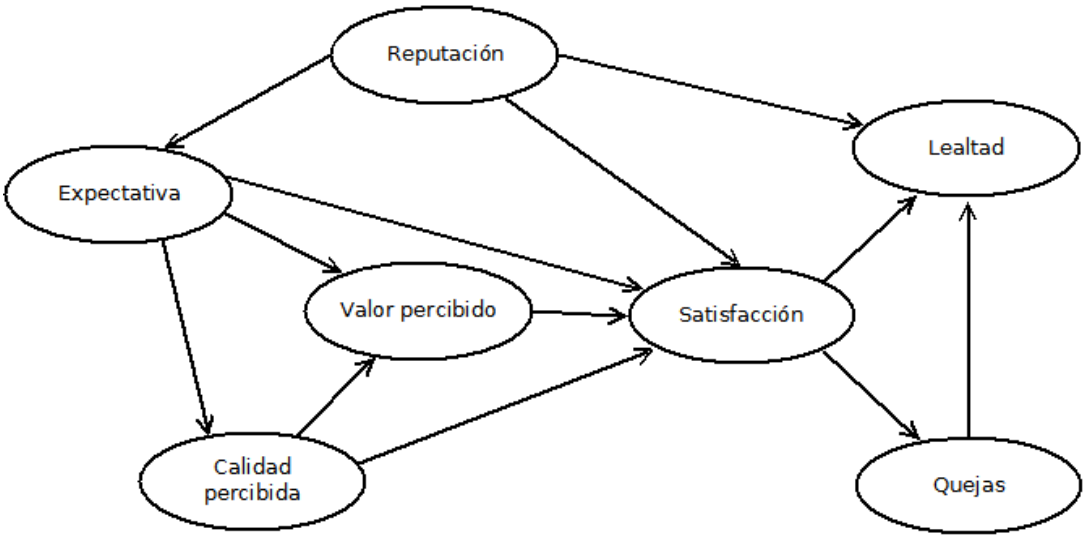

Figura 4.3: Diagrama de rutas (modelo ECSI). Análisis completo.

Descripción general del modelo ECSI

- **Reputación** se refiere a la marca y el tipo de asociaciones que los clientes obtienen del producto/marca/compañía. Se espera que la imagen tenga un efecto positivo en la satisfacción y lealtad del cliente. Además, también se espera que la imagen tenga un efecto directo sobre las expectativas.
- **Expectativas** es la información basada en la información acumulada sobre la calidad por parte de fuentes externas como son la publicidad, el "boca a boca" y los medios de comunicación.
- **Calidad percibida** comprende la calidad del producto y la calidad del servicio. La calidad del producto percibida es la evaluación de la experiencia de consumo reciente de los productos. La calidad del servicio percibida es la evaluación de la experiencia de consumo reciente de servicios asociados, como el servicio al cliente, las condiciones de exhibición del producto, la gama de servicios y productos, etc. Se espera que la calidad percibida afecte la satisfacción.
- **Valor percibido** es el nivel percibido de la calidad del producto en relación con el precio pagadopor él según la experiencia del cliente.
- **Satisfacción** se define como una evaluación general del rendimiento posterior a la compra o a la utilización de un servicio.
- **Quejas** implica las quejas o reclamaciones de los clientes.
- **Lealtad** se refiere a la intención de volver a comprar y a la aceptación de precios por parte de los clientes. Es la última variable dependiente en el modelo y se espera que a mayor reputación y satisfacción del cliente la lealtad del cliente sea mayor.

4.3.1. Descripción del modelo

El caso que vamos a tratar en este ejemplo consiste en una aplicación del modelo ECSI a los Servicios Educativos y consideraremos una versión simplificada y modificada. El modelo adaptado tiene como objetivo medir qué tan satisfechos están los miembros del programa académico teniendo en cuenta la calidad del apoyo brindado, la calidad de la asesoría y la calidad de la tutoría. También se supone que estos tres factores de calidad tienen un impacto en el valor percibido. Además de medir la satisfacción, el modelo también presta atención a la Lealtad entendida como el compromiso y la participación de los estudiantes con el programa. Como era de esperar, cuanto más satisfechos estén los miembros del programa, más leales serán al mismo.

En el conjunto de datos, cada fila es una respuesta del cuestionario aplicado a 181 estudiantes miembros del programa académico. También hay tres variables categóricas (las últimas tres columnas): género, becas y trabajo. Indican el género de los encuestados, si tienen una beca y si tienen un trabajo. El resto de las preguntas se miden en una escala de 7 puntos.

Según la pregunta, se dieron las siguientes opciones a los encuestados:

$$\text{A:} \begin{cases} 1 = \text{Totalmente en desacuerdo} \\ 4 = \text{Ni de acuerdo ni en desacuerdo} \\ 7 = \text{Completamente de acuerdo} \end{cases} \qquad \text{B:} \begin{cases} 1 = \text{Nada} \\ 7 = \text{Mucho} \end{cases}$$

A continuación se presentan las variables del conjunto de datos agrupadas en bloques ya que serían las variables manifiestas correspondientes a cada variable latente además de la pregunta planteada correspondiente.

- **Apoyo**
  - `sup.help` Comodidad al pedir ayuda al personal del programa
  - `sup.under` Sensación de ser subestimado en el programa
  - `sup.safe` Capacidad de encontrar un lugar donde sentirse seguro en el programa
  - `sup.conc` Asistencia al programa cuando surgen dudas sobre la educación

- **Asesoría**
  - `adv.comp` Competencia de asesores
  - `adv.acces` Acceso a los asesores
  - `adv.comm` Habilidades de comunicación de los asesores
  - `adv.qual` Calidad general de asesoramiento

- **Tutoría**
  - `tut.prof` Habilidad de los tutores
  - `tut.sched` Horarios de tutorías
  - `tut.stud` Diversidad de grupos de estudio
  - `tut.qual` Calidad general de las tutorías

- **Valor**
  - `val.devel` Utilidad en el desarrollo personal
  - `val.deci` Utilidad en la toma de decisiones
  - `val.meet` Facilidad para conocer gente
  - `val.info` Accesibilidad al soporte y a la información

- **Satisfacción**
  - `sat.glad` Satisfacción por ser miembro del programa
  - `sat.expe` El programa cumple las expectativas
  - `sat.over` Satisfacción general con el programa

- **Lealtad**
  - `loy.proud` Sensación de orgullo al decir que forma parte del programa
  - `loy.recom` Recomendaría el programa a sus conocidos
  - `loy.asha` Sensación de vergüenza alguna vez por ser miembro del programa
  - `loy.back` Interés en aportar algo al programa

El conjunto de datos que trataremos puede descargarse en formato texto desde el siguiente enlace: http://www.gastonsanchez.com/education.txt o en formato `.csv` separado por comas en este otro enlace http://www.gastonsanchez.com/education.csv.

El modelo correspondiente puede verse en la figura 4.4.

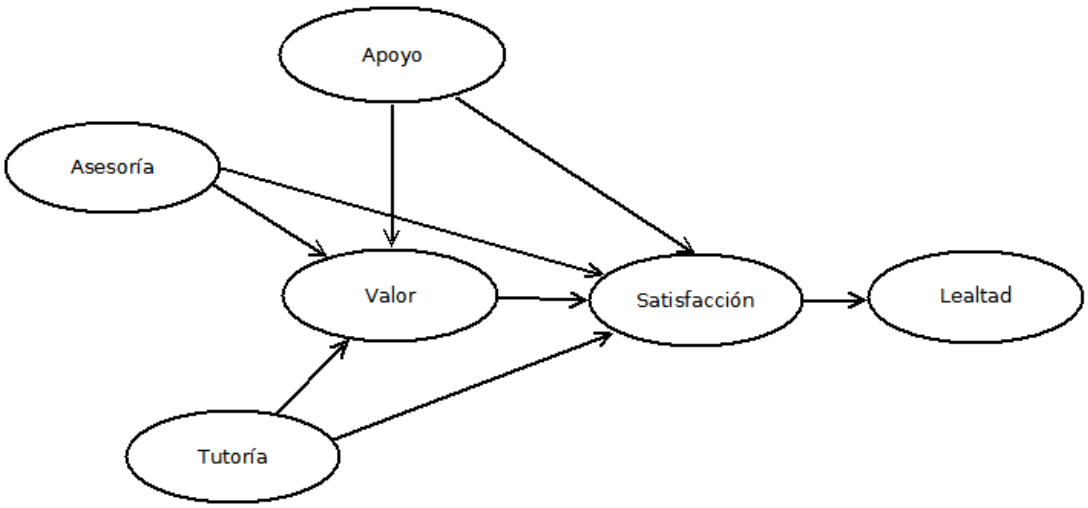

Figura 4.4: Diagrama de rutas (ECSI educación). Análisis completo.

4.3.2. Algoritmo PLS-PM

4.3.2.1. Iteración 1

Una vez que se haya descargado el archivo, el primer paso es importar los datos en R con algunas de las siguientes funciones: `read.table()` para leer archivos `.txt`, `read.csv()` para leer archivos `.csv` y `read.delim()` para leer archivos con otros tipos de delimitadores.

Lectura de datos tipo txt

```
education=read.table("education.txt",header=TRUE,row.names=1)
```

Lectura de datos tipo csv

```
education=read.csv("education.csv",header=TRUE,row.names=1)
```

Suponiendo que la limpieza de los datos, el tratamiento de los valores perdidos y los valores atípicos y el formato de las variables que necesitan algunas transformaciones están hechos, pasamos a la construcción del modelo PLS-PM. Lo primero es construir la matriz de ruta, la lista de bloques y el vector de tipos de medida.

Matriz de ruta o matriz del modelo interno

El modelo interno se representa mediante una matriz que debe ser una matriz booleana triangular inferior, es decir, una matriz cuadrada donde la diagonal y los elementos superiores son cero y el resto ceros o unos.

La interpretación de esa matriz es la siguiente: los ceros en la diagonal significan que una variable latente no se afecta a sí misma, los ceros de encima de la diagonal implican que el PLS-PM sólo usa modelos recursivos (sin lazos) y debajo de la diagonal, un 1 en la celda (i,j) significa que la columna j afecta a la fila i.

Filas de la matriz

```
Apoyo=c(0,0,0,0,0,0)
Asesoria=c(0,0,0,0,0,0)
Tutoria=c(0,0,0,0,0,0)
Valor=c(1,1,1,0,0,0)
Satisfaccion=c(1,1,1,1,0,0)
Lealtad=c(0,0,0,0,1,0)
```

Matriz de ruta

```
edu_path=rbind(Apoyo,Asesoria,Tutoria,Valor,Satisfaccion,Lealtad)
colnames(edu_path)=rownames(edu_path)
```

```
edu_path
```

Cuadro 4.13: Matriz de ruta. Análisis completo.

	Apoyo	Asesoria	Tutoria	Valor	Satisfaccion	Lealtad
Apoyo	0	0	0	0	0	0
Asesoria	0	0	0	0	0	0
Tutoria	0	0	0	0	0	0
Valor	1	1	1	0	0	0
Satisfaccion	1	1	1	1	0	0
Lealtad	0	0	0	0	1	0

Representación de la matriz de ruta

```
innerplot(edu_path,box.size = 0.1)
```

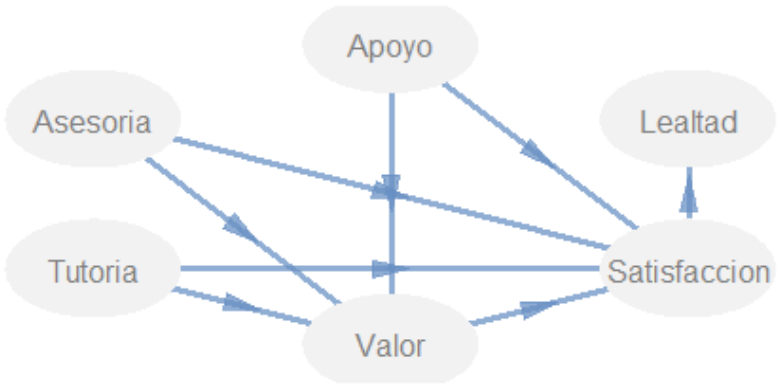

Figura 4.5: Modelo interno. Análisis completo.

Lista del modelo externo

El modelo externo se define mediante una lista con tantos elementos como variables latentes hay siendo cada uno de los elementos un vector de índices que indica las variables asociadas a cada variable latente. De forma alternativa, también podemos utilizar los nombres de las variables en lugar de los índices.

```
edu_bloques=list(1:4,5:8,9:12,13:16,17:19,20:23)
```

Vector de tipos

Por defecto, la función `plspm()` asume que la forma de medir las variables latentes es la forma reflexiva conocida como *modo A*, el modo formativo se conoce como *modo B*.

El vector de tipos de medida que usa la función `plspm()` como argumento es un vector de caracteres (`"A"` o `"B"`) con tantos elementos como variables latentes haya en el modelo.

```
edu_modos=rep("A",6)
```

Resultados

Ahora, ya tenemos todo lo necesario para ejecutar la función `plspm()`.

```
edu_pls=plspm(education,edu_path,edu_bloques,edu_modos)
```

Los argumentos de salida de esta función son los que aparecen en la tabla anterior y haciendo `pls$` se pueden visualizar así como un resumen, si se ejecuta la siguiente instrucción:

```
summary(edu_pls)
```

Interpretación de resultados

La interpretación de los resultados obtenidos podemos realizarla en dos etapas: la evaluación del modelo de medida y la evaluación del modelo estructural, en ese orden.

Valoración del modelo de medida

1. Unidimensionalidad de las variables manifiestas

En PLS-PM hay tres índices para comprobar la unidimensionalidad: el alpha de Cronbach, el rho de Dillon-Goldstein y el primer autovalor de la matriz de correlaciones de las variables manifiestas. Todos ellos se pueden obtener ejecutando la siguiente instrucción:

```
edu_pls$unidim
```

Cuadro 4.14: Tabla para comprobar la unidimensionalidad. Análisis completo (It.1).

	Mode	MVs	C.alpha	DG.rho	eig.1st	eig.2nd
Apoyo	A	4	0.1966819	0.6158067	2.270830	0.7291746
Asesoria	A	4	0.9282665	0.9492069	3.295520	0.3429636
Tutoria	A	4	0.8545382	0.9020758	2.791439	0.5376909
Valor	A	4	0.9103054	0.9371317	3.154300	0.5090336
Satisfaccion	A	3	0.9024956	0.9389807	2.510583	0.2699962
Lealtad	A	4	0.3383147	0.7222804	2.623427	0.6791109

En la tabla de la salida anterior, se obtienen estos índices para cada variable latente y en la primera columna aparece el tipo de medida, en la segunda el número de variables manifiestas asociadas a cada variable latente, en la tercera aparece el alpha de Cronbach, en la cuarta el rho de Dillon-Goldstein y en la quinta y sexta el primer y segundo autovalor de la matriz de correlaciones de las variables manifiestas, respectivamente.

· **Alpha de Cronbach**

```
edu_pls$unidim[,3,drop=FALSE]
```

Cuadro 4.15: Alpha de Cronbach. Análisis completo (It.1).

	C.alpha
Apoyo	0.1966819
Asesoria	0.9282665
Tutoria	0.8545382
Valor	0.9103054
Satisfaccion	0.9024956
Lealtad	0.3383147

Salvo `Apoyo` y `Lealtad` el resto de bloques presentan valores aceptables de alpha.

· **Rho de Dillon-Goldstein**

```
edu_pls$unidim[,4,drop=FALSE]
```

Cuadro 4.16: Rho de Dillon-Goldstein. A.completo (It.1).

	DG.rho
Apoyo	0.6158067
Asesoria	0.9492069
Tutoria	0.9020758
Valor	0.9371317
Satisfaccion	0.9389807
Lealtad	0.7222804

Podemos considerar que todos los bloques son unidimensionales según este índice.

· **Primer autovalor de la matriz de correlaciones de las variables manifiestas**

```
edu_pls$unidim[,5:6]
```

Cuadro 4.17: Primer autovalor. Análisis completo (It.1).

	eig.1st	eig.2nd
Apoyo	2.270830	0.7291746
Asesoria	3.295520	0.3429636
Tutoria	2.791439	0.5376909
Valor	3.154300	0.5090336
Satisfaccion	2.510583	0.2699962
Lealtad	2.623427	0.6791109

Por tanto, según este criterio consideramos todos los bloques unidimensionales.

Nota
Veamos qué ocurre con los bloques `Apoyo` y `Lealtad` con lo que respecta a la unidimensionalidad.
Para ello, primero representamos las cargas:

```
plot(edu_pls,what="loadings")
```

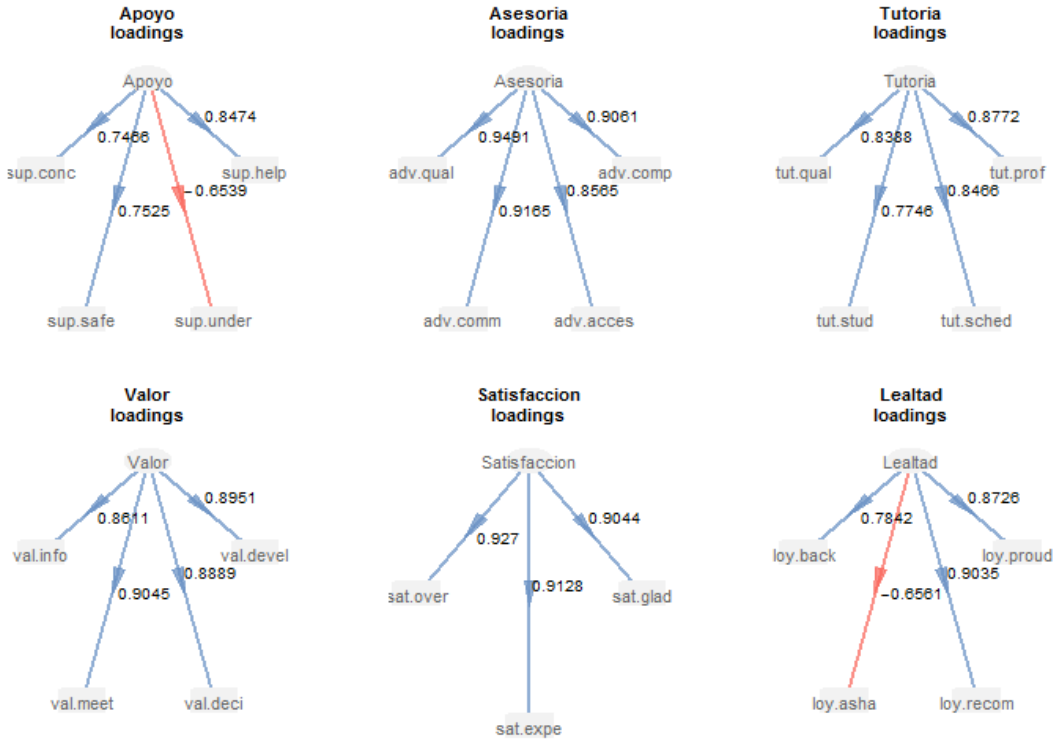

Figura 4.6: Diagrama de cargas. Análisis completo (iteración 1).

El alpha de Cronbach en estos bloques no es adecuado ya que como podemos ver, las cargas de las variables de estos bloques tienen signos contrarios: `sup.under` tiene una carga negativa con `Apoyo` y `loy.asha` tiene carga negativa con `Lealtad`.

La solución a este problema es transformar las variables `sup.under` y `loy.asha` de acuerdo a su definición. Como la pregunta del cuestionario asociada a `sup.under` es: "Sensación de ser subestimado en el programa" en su lugar deberíamos tener: "Sensación de ser valorado en el programa". Del mismo modo, como la pregunta correspondiente a `loy.asha` es: "Sensación de vergüenza alguna vez por ser miembro del programa" en su lugar deberíamos tener "Buena sensación por ser miembro del programa". Computacionalmente, la solución está en invertir la escala de medida de las variables manifiestas correspondientes. Realmente, crearemos nuevas variables que llamaremos `sup.appre` y `loy.pleas`.

4.3.2.2. Iteración 2

A continuación, realizaremos el estudio anterior pero con las nuevas variables. Como las nuevas variables se añaden al final, los índices de los bloques cambian:

```
education$sup.appre=8-education$sup.under
education$loy.pleas=8-education$loy.asha
edu_bloques2=list(c(1,27,3,4),5:8,9:12,13:16,17:19,c(20,21,28,23))
```

```
edu_pls2=plspm(education,edu_path,edu_bloques2,modes=edu_modos)
```

```
plot(edu_pls2,"loadings")
```

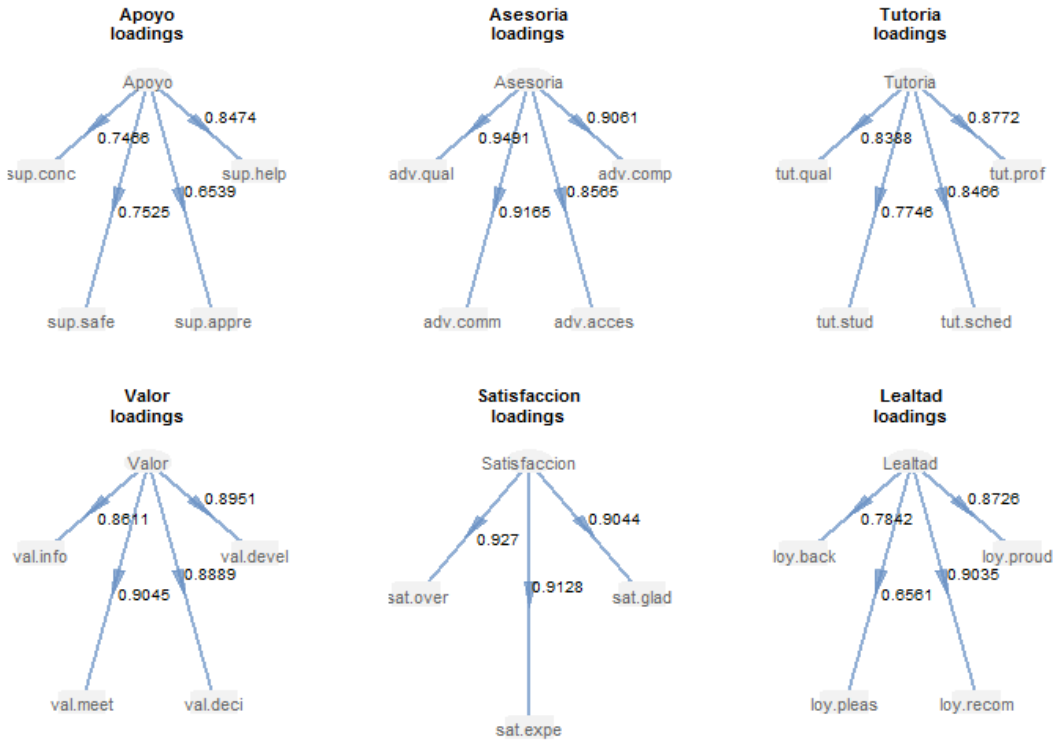

Figura 4.7: Diagrama de cargas. Análisis completo (iteración 2).

```
edu_pls2$unidim
```

Cuadro 4.18: Tabla para comprobar la unidimensionalidad. Análisis completo (It.2).

	Mode	MVs	C.alpha	DG.rho	eig.1st	eig.2nd
Apoyo	A	4	0.7432695	0.8391975	2.270830	0.7291746
Asesoria	A	4	0.9282665	0.9492069	3.295520	0.3429636
Tutoria	A	4	0.8545382	0.9020758	2.791439	0.5376909
Valor	A	4	0.9103054	0.9371317	3.154300	0.5090336
Satisfaccion	A	3	0.9024956	0.9389807	2.510583	0.2699962
Lealtad	A	4	0.8199850	0.8826899	2.623427	0.6791109

Ahora, todas las variables manifiestas son unidimensionales. Además de comprobar la unidimensionalidad es necesario comprobar que las variables latentes están bien explicadas por sus variables manifiestas. Para ello usaremos las cargas y las comunalidades que se obtienen en la tercera y cuarta columna del resultado de la ejecución del siguiente comando:

```
edu_pls2$outer_model
```

Cuadro 4.19: Modelo externo. Análisis completo (It.2).

name	block	weight	loading	communality	redundancy
sup.help	Apoyo	0.3869676	0.8474158	0.7181136	0.0000000
sup.appre	Apoyo	0.2740855	0.6538637	0.4275378	0.0000000
sup.safe	Apoyo	0.3172804	0.7525010	0.5662578	0.0000000
sup.conc	Apoyo	0.3403551	0.7465998	0.5574113	0.0000000
adv.comp	Asesoria	0.2652023	0.9060801	0.8209812	0.0000000
adv.acces	Asesoria	0.2572178	0.8565128	0.7336142	0.0000000
adv.comm	Asesoria	0.2879199	0.9165159	0.8400013	0.0000000
adv.qual	Asesoria	0.2902939	0.9490795	0.9007519	0.0000000
tut.prof	Tutoria	0.3023281	0.8772088	0.7694953	0.0000000
tut.sched	Tutoria	0.3153917	0.8466293	0.7167811	0.0000000
tut.stud	Tutoria	0.2974617	0.7745923	0.5999933	0.0000000
tut.qual	Tutoria	0.2829958	0.8387538	0.7035080	0.0000000
val.devel	Valor	0.2628094	0.8951159	0.8012325	0.5195390
val.deci	Valor	0.2759146	0.8889187	0.7901765	0.5123700
val.meet	Valor	0.2739089	0.9044643	0.8180556	0.5304476
val.info	Valor	0.3155918	0.8610767	0.7414531	0.4807765
sat.glad	Satisfaccion	0.3543066	0.9043571	0.8178618	0.5144359
sat.expe	Satisfaccion	0.3550445	0.9127603	0.8331314	0.5240406
sat.over	Satisfaccion	0.3834896	0.9270390	0.8594012	0.5405643
loy.proud	Lealtad	0.3333923	0.8725976	0.7614265	0.4590993
loy.recom	Lealtad	0.3512786	0.9034974	0.8163076	0.4921896
loy.pleas	Lealtad	0.2422710	0.6561207	0.4304944	0.2595650
loy.back	Lealtad	0.2967867	0.7842142	0.6149919	0.3708071

Para visualizarlo mejor, representaremos las cargas de las variables en el diagrama de barras de la figura 4.8.

Podemos considerar que casi todas las variables manifiestas explican bien su variable latente correspondiente ya que las cargas son mayores que 0.7 y, por tanto, las comunalidades mayores a 0.49. Sólo las variables transformadas presentan cargas un poco menores que 0.7 por lo que las eliminaremos del modelo y veremos si así obtenemos mejores resultados.

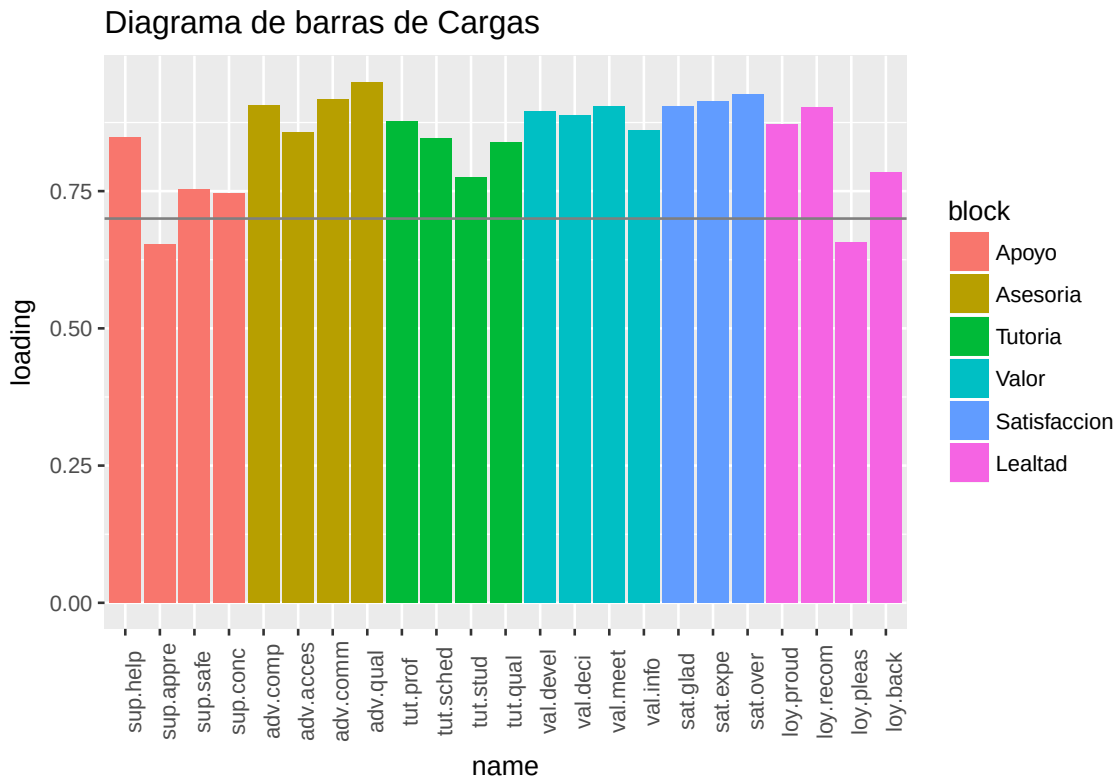

Figura 4.8: Diagrama de barras de las cargas. Análisis completo (iteración 2)

4.3.2.3. Iteración 3

La construcción del nuevo modelo PLS-PM implica re-definir los bloques ya que tenemos dos variables menos que añadir en los mismos: `sup.appre` y `loy.pleas`.

```
edu_bloques3=list(c(1,3,4),5:8,9:12,13:16,17:19,c(20,21,23))
edu_pls3=plspm(education,edu_path,edu_bloques3,modes=edu_modos)
```

Valoración del modelo de medida

1. Unidimensionalidad de las variables manifiestas

Como ya hemos visto, podemos comprobar la unidimensionalidad con tres índices:

· **Alpha de Cronbach**

```
edu_pls3$unidim[,3,drop=FALSE]
```

Cuadro 4.20: Alpha de Cronbach. Análisis completo (It.3).

	C.alpha
Apoyo	0.7327833
Asesoria	0.9282665
Tutoria	0.8545382
Valor	0.9103054
Satisfaccion	0.9024956
Lealtad	0.8454232

· **Rho de Dillon-Goldstein**

```
edu_pls3$unidim[,4,drop=FALSE]
```

Cuadro 4.21: Rho de Dillon Goldstein. Análisis completo (It.3).

	DG.rho
Apoyo	0.8491830
Asesoria	0.9492069
Tutoria	0.9020758
Valor	0.9371317
Satisfaccion	0.9389807
Lealtad	0.9071416

· **Primer autovalor de la matriz de correlaciones de las variables manifiestas**

```
edu_pls3$unidim[,5:6]
```

Cuadro 4.22: Primer autovalor. Análisis completo (It.3).

	eig.1st	eig.2nd
Apoyo	1.958832	0.6293020
Asesoria	3.295520	0.3429636
Tutoria	2.791439	0.5376909
Valor	3.154300	0.5090336
Satisfaccion	2.510583	0.2699962
Lealtad	2.296624	0.4889435

Podemos considerar todos los bloques unidimensionales ya que tienen valores aceptables de cada uno de los índices.

2. Comprobar que las variables latentes están bien explicadas por sus variables manifiestas

Para ello usaremos las cargas y las comunalidades que se obtienen en la tercera y cuarta columna del resultado de la ejecución de este comando. Para visualizarlo mejor, representaremos en un diagrama de barras (figura 4.9) las cargas de cada una de las variables.

```
edu_pls3$outer_model
```

Podemos considerar que todas las variables manifiestas explican bien su variable latente correspondiente ya que las cargas son mayores que 0.7 y, por tanto, las comunalidades mayores a 0.49.

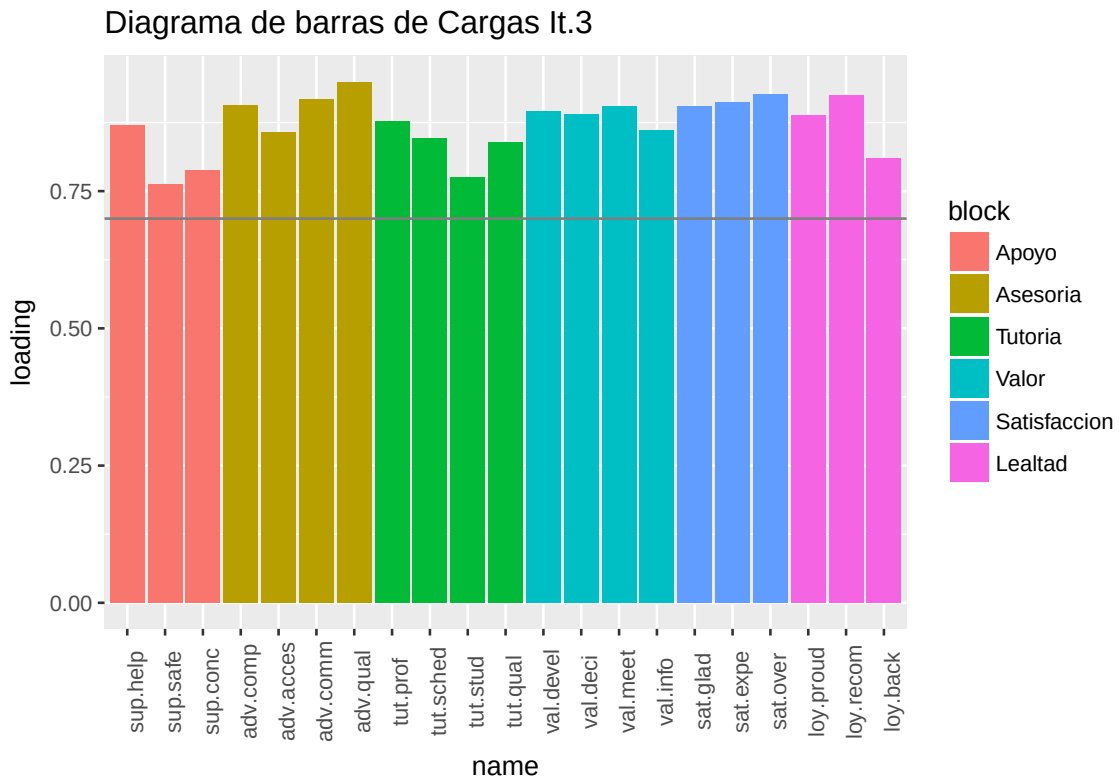

Figura 4.9: Diagrama de barras de las cargas. Análisis completo (iteración 3).

3. Evaluar el grado en que una variable latente es diferente a otras

Para ello, tenemos que evaluar las cargas cruzadas que para obtenerlas en R hacemos:

```
edu_pls3$crossloadings
```

Al ejecutar esta instrucción vemos que las variables presentan la mayor carga cruzada con su variable latente asociada, entonces podemos considerar que dichas asociaciones están realizadas correctamente.

Valoración del modelo estructural

Una vez evaluada la calidad del modelo de medida, pasamos a evaluar la calidad del modelo estructural estudiando los resultados obtenidos en cada regresión de las ecuaciones estructurales. Esto lo hacemos mirando:

```
edu_pls3$inner_model
```

```
$Valor
               Estimate Std. Error       t value     Pr(>|t|)
Intercept -5.805200e-17 0.04375066 -1.326883e-15 1.000000e+00
Apoyo      6.702131e-01 0.05260231  1.274113e+01 8.311305e-27
Asesoria   1.727304e-01 0.05233050  3.300760e+00 1.166175e-03
Tutoria    8.195890e-02 0.05255977  1.559346e+00 1.207006e-01

$Satisfaccion
              Estimate Std. Error      t value     Pr(>|t|)
Intercept 7.878109e-17 0.04617867 1.706006e-15 1.000000e+00
Apoyo     1.004346e-01 0.07687591 1.306451e+00 1.931040e-01
```

```
Asesoria  3.764012e-01 0.05690922 6.614063e+00 4.323511e-10
Tutoria   1.312058e-01 0.05585641 2.348983e+00 1.993448e-02
Valor     3.492695e-01 0.07933598 4.402410e+00 1.852944e-05

$Lealtad
                 Estimate Std. Error      t value     Pr(>|t|)
Intercept    3.867146e-16  0.0482275 8.018549e-15 1.000000e+00
Satisfaccion 7.639799e-01  0.0482275 1.584117e+01 6.764529e-36
```

La calidad del modelo estructural se mide en base al coeficiente de determinación R^2 y el índice de redundancia. Estos índices se pueden obtener en R como sigue:

```
edu_pls3$inner_summary
```

Cuadro 4.23: Tabla para evaluar el modelo estructural. Análisis completo (It.3).

	Type	R2	Block_Communality	Mean_Redundancy	AVE
Apoyo	Exogenous	0.0000000	0.6527765	0.0000000	0.6527765
Asesoria	Exogenous	0.0000000	0.8238370	0.0000000	0.8238370
Tutoria	Exogenous	0.0000000	0.6974447	0.0000000	0.6974447
Valor	Endogenous	0.6612007	0.7877939	0.5208898	0.7877939
Satisfaccion	Endogenous	0.6246853	0.8368150	0.5227460	0.8368150
Lealtad	Endogenous	0.5836654	0.7655063	0.4467995	0.7655063

En esta tabla, para cada variable latente tenemos el tipo de variable que es (endógena o exógena), el coeficiente de determinación (sólo para variables endógenas), la comunalidad media (cuánta variabilidad es reproducible por la variable latente) y la redundandia media (AVE) (sólo para variables endógenas) los cuales presentan valores aceptables.

Como podemos considerar que disponemos de un buen modelo el modelo interno, vamos a visualizar los coeficientes de ruta obtenidos:

```
edu_pls3$path_coefs
```

Cuadro 4.24: Coeficientes de ruta. Análisis completo (It.3).

	Apoyo	Asesoria	Tutoria	Valor	Satisfaccion	Lealtad
Apoyo	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0
Asesoria	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0
Tutoria	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0
Valor	0.6702131	0.1727304	0.0819589	0.0000000	0.0000000	0
Satisfaccion	0.1004346	0.3764012	0.1312058	0.3492695	0.0000000	0
Lealtad	0.0000000	0.0000000	0.0000000	0.0000000	0.7639799	0

Gráficamente:

```
plot(edu_pls3,arr.pos=0.35,arr.lwd=7*round(edu_pls3$path_coefs,2))
```

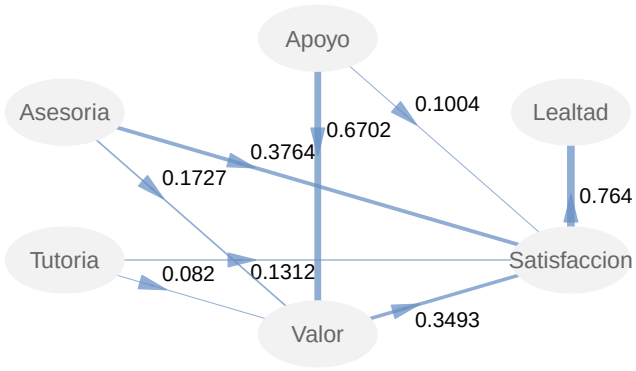

Figura 4.10: Modelo interno. Análisis completo (iteración 3).

Además de ver los coeficientes de ruta, hay que revisar los resultados de la regresión de las variables latentes endógenas. Para ello, tenemos que interpretar el R^2 de la siguiente instrucción:

```
edu_pls3$inner_model
```

```
$Valor
               Estimate Std. Error       t value     Pr(>|t|)
Intercept -5.805200e-17 0.04375066 -1.326883e-15 1.000000e+00
Apoyo      6.702131e-01 0.05260231  1.274113e+01 8.311305e-27
Asesoria   1.727304e-01 0.05233050  3.300760e+00 1.166175e-03
Tutoria    8.195890e-02 0.05255977  1.559346e+00 1.207006e-01

$Satisfaccion
              Estimate Std. Error      t value     Pr(>|t|)
Intercept 7.878109e-17 0.04617867 1.706006e-15 1.000000e+00
Apoyo     1.004346e-01 0.07687591 1.306451e+00 1.931040e-01
Asesoria  3.764012e-01 0.05690922 6.614063e+00 4.323511e-10
Tutoria   1.312058e-01 0.05585641 2.348983e+00 1.993448e-02
Valor     3.492695e-01 0.07933598 4.402410e+00 1.852944e-05

$Lealtad
                Estimate Std. Error      t value     Pr(>|t|)
Intercept    3.867146e-16  0.0482275 8.018549e-15 1.000000e+00
Satisfaccion 7.639799e-01  0.0482275 1.584117e+01 6.764529e-36
```

Otro resultado importante a tener en cuenta es la tabla del argumento `$effects`. Esta tabla contiene los efectos que cada variable latente tiene en el resto teniendo en cuenta el número total de conexiones en el modelo interno. El efectos directos están dados por los coeficientes de ruta. Pero también están los efectos indirectos y los efectos totales. Un efecto indirecto es la influencia de una variable latente sobre otra mediante una ruta indirecta y se obtienen multiplicando los coeficientes de rutas de las dos variables conectadas indirectamente con la variable intermedia que permite dicha conexión. Los efectos totales son la suma de los efectos directos e indirectos.

Se obtienen de la siguiente forma:

```
edu_pls3$effects
```

Cuadro 4.25: Efectos. Análisis completo (It.3).

relationships	direct	indirect	total
Apoyo -> Asesoria	0.0000000	0.0000000	0.0000000
Apoyo -> Tutoria	0.0000000	0.0000000	0.0000000
Apoyo -> Valor	0.6702131	0.0000000	0.6702131
Apoyo -> Satisfaccion	0.1004346	0.2340850	0.3345196
Apoyo -> Lealtad	0.0000000	0.2555663	0.2555663
Asesoria -> Tutoria	0.0000000	0.0000000	0.0000000
Asesoria -> Valor	0.1727304	0.0000000	0.1727304
Asesoria -> Satisfaccion	0.3764012	0.0603295	0.4367307
Asesoria -> Lealtad	0.0000000	0.3336535	0.3336535
Tutoria -> Valor	0.0819589	0.0000000	0.0819589
Tutoria -> Satisfaccion	0.1312058	0.0286257	0.1598315
Tutoria -> Lealtad	0.0000000	0.1221081	0.1221081
Valor -> Satisfaccion	0.3492695	0.0000000	0.3492695
Valor -> Lealtad	0.0000000	0.2668349	0.2668349
Satisfaccion -> Lealtad	0.7639799	0.0000000	0.7639799

Gráficamente:

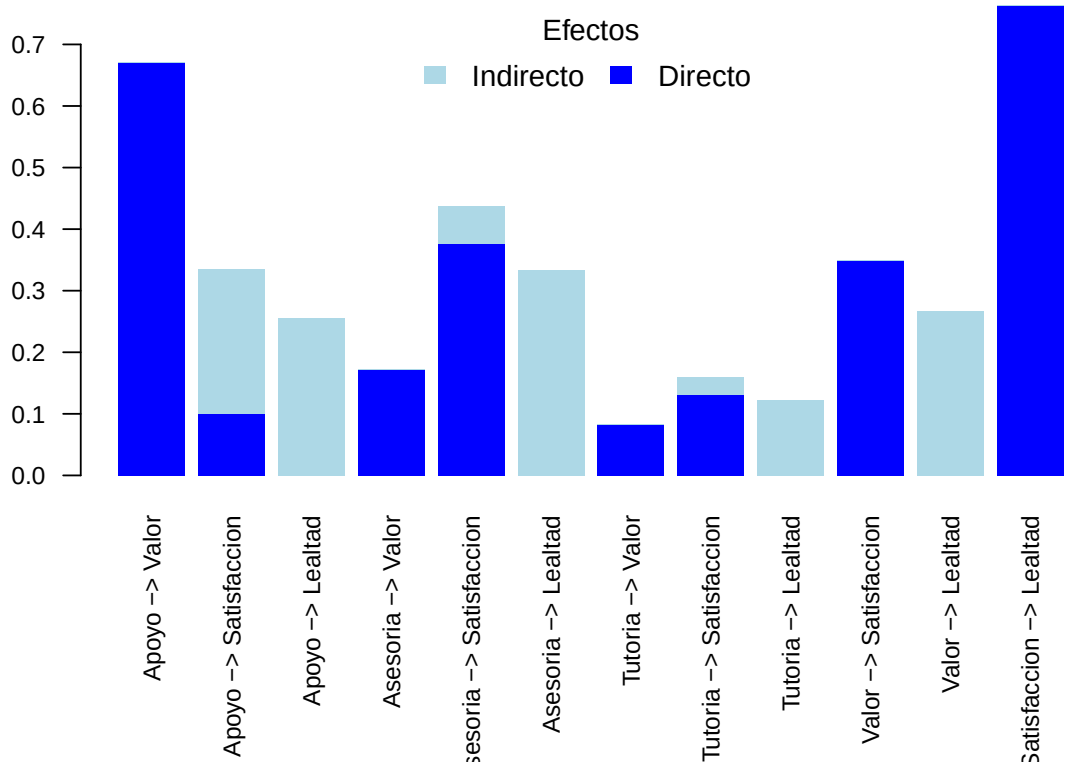

Figura 4.11: Gráfica de efectos. Análisis completo (iteración 3).

Además, también debemos comprobar la calidad del modelo interno y de la capacidad predictiva del modelo mediante las puntuaciones de las variables latentes. En R la obtenemos:

```
summary(edu_pls3$scores)
```

```
     Apoyo            Asesoria           Tutoria            Valor        
 Min.   :-3.1481   Min.   :-5.7659   Min.   :-4.0897   Min.   :-3.7810  
 1st Qu.:-0.6540   1st Qu.:-0.3437   1st Qu.:-0.7205   1st Qu.:-0.6298  
 Median : 0.1642   Median : 0.2694   Median : 0.2791   Median : 0.2275  
 Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 0.7603   3rd Qu.: 0.7924   3rd Qu.: 0.8239   3rd Qu.: 0.8904  
 Max.   : 1.2422   Max.   : 0.7924   Max.   : 1.4039   Max.   : 1.0847  
  Satisfaccion        Lealtad        
 Min.   :-6.6441   Min.   :-6.4929  
 1st Qu.:-0.4511   1st Qu.:-0.2469  
 Median : 0.3227   Median : 0.5758  
 Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 0.6646   3rd Qu.: 0.5758  
 Max.   : 0.6646   Max.   : 0.5758  
```

Como no todas las variables están en la misma escala, debemos obtener dichas puntuaciones sobre las variables normalizadas. Para ello:

```
Scores=rescale(edu_pls3)
summary(Scores)
```

```
     Apoyo          Asesoria        Tutoria          Valor
 Min.   :1.607   Min.   :1.234   Min.   :1.768   Min.   :1.278
 1st Qu.:4.674   1st Qu.:6.000   1st Qu.:5.000   1st Qu.:5.000
 Median :5.652   Median :6.525   Median :5.977   Median :6.000
 Mean   :5.471   Mean   :6.301   Mean   :5.671   Mean   :5.710
 3rd Qu.:6.370   3rd Qu.:7.000   3rd Qu.:6.484   3rd Qu.:6.757
 Max.   :7.000   Max.   :7.000   Max.   :7.000   Max.   :7.000
  Satisfaccion      Lealtad
 Min.   :1.000   Min.   :1.000
 1st Qu.:6.003   1st Qu.:6.340
 Median :6.676   Median :7.000
 Mean   :6.438   Mean   :6.506
 3rd Qu.:7.000   3rd Qu.:7.000
 Max.   :7.000   Max.   :7.000
```

Valoración del modelo global

Para ello se utiliza el coeficiente de bondad de ajuste (GoF). En R, podemos obtenerlo como:

```
edu_pls3$gof
```

```
[1] 0.6890965
```

Bootstrap

El procedimiento de remuestreo bootstrap en PLS-PM se usa para estimar la precisión de los parámetros estimados. En R por defecto usa 100 muestras bootstrap pero podemos especificar cuántas muestras queremos cambiando el argumento `br`

```
edu_pls_val=plspm(education,edu_path,edu_bloques3,modes=edu_modos,
                  boot.val = TRUE,br=200)
```

Los resultados de la validación cruzada se obtienen de la siguiente forma:

```
edu_pls_val$boot
```

Esto nos proporciona la siguiente lista de resultados:
- los pesos externos (`plsboot$boot$weights`)
- las cargas (`plsboot$boot$loadings`)
- los coeficientes de ruta (`plsboot$boot$paths`)
- el coeficiente de determinación R^2 (`plsboot$boot$rsq`)
- los efectos totales (`plsboot$boot$total.efs`)

Cada uno de ellos es una matriz con cinco columnas que contienen: los valores originales de los parámetros, los valores medios del bootstrap, el error bootstrap estándar y el extermo inferior y superior del intervalo de confianza bootstrap al 95 %.

4.4. Comparación de modelos PLS-PM

4.4.1. Comparación de grupos

Ahora, vamos a realizar comparaciones de modelos PLS-PM en base a lo descrito en el capítulo anterior. Usando el conjunto de datos anterior, como disponemos de tres variables categóricas que indican el sexo de los encuestados, si tienen trabajo o no y si disfrutan de beca o no, vamos a utilizar, por ejemplo, la variable `gender` para comparar los modelos PLS en hombres y mujeres. De la misma forma que procedemos con esta variable podríamos hacerlo con las variables `scholarships` y `job`.

Primero, dividimos el conjunto de datos según los encuestados sean hombres o mujeres.

```
fem = education[education$gender == "female", ]
fem_pls = plspm(fem, edu_path, edu_bloques3, modes = edu_modos)
masc = education[education$gender == "male", ]
masc_pls = plspm(masc, edu_path, edu_bloques3, modes = edu_modos)
```

Para comparar ambos modelos PLS-PM atendemos a los coeficientes de ruta de cada uno.

```
plot(fem_pls,arr.pos = 0.35)
```

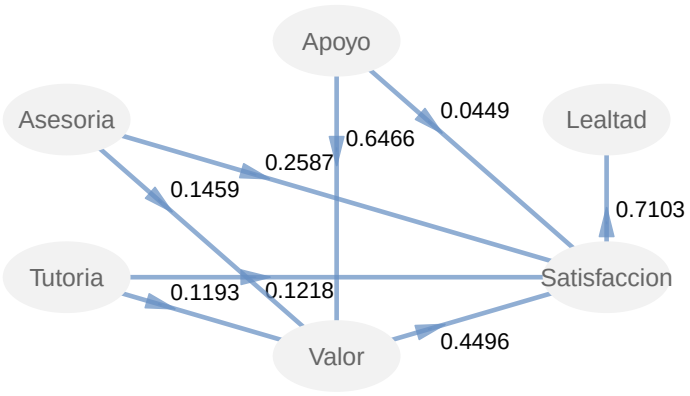

Figura 4.12: Coeficientes de ruta (mujeres). Comparación de grupos.

```
plot(masc_pls,arr.pos=0.35)
```

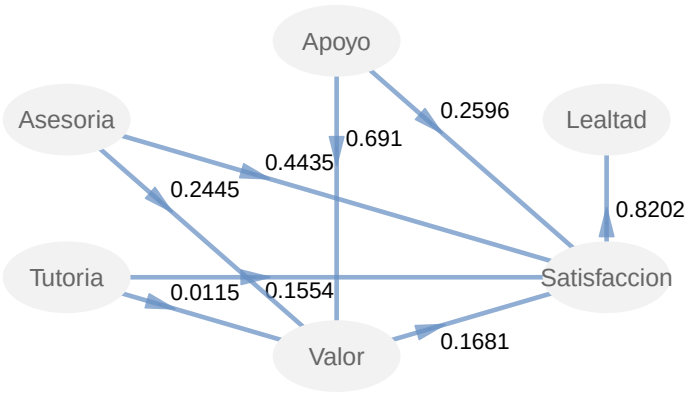

Figura 4.13: Coeficientes de ruta (hombres). Comparación de grupos.

Podemos observar diferencias numéricas entre los coeficientes de ruta pero necesitamos realizar un análisis de grupos.

4.4.1.1. T-test bootstrap

```
(edu_boot = plspm.groups(edu_pls3, education$gender, method ="bootstrap"))
```

```
GROUP COMPARISON IN PLS-PM FOR PATH COEFFICIENTS

Scale of Data:       TRUE
Weighting Scheme:    centroid
Selected method:     bootstrap
Num of replicates:   100

$test
                        global  group.female  group.male  diff.abs  t.stat
Apoyo->Valor            0.6702        0.6466      0.6910    0.0444  0.3656
Apoyo->Satisfaccion     0.1004        0.0449      0.2596    0.2148  1.2217
Asesoria->Valor         0.1727        0.1459      0.2445    0.0986  0.6025
Asesoria->Satisfaccion  0.3764        0.2587      0.4435    0.1848  1.3091
Tutoria->Valor          0.0820        0.1193      0.0115    0.1079  0.6960
Tutoria->Satisfaccion   0.1312        0.1218      0.1554    0.0337  0.0832
Valor->Satisfaccion     0.3493        0.4496      0.1681    0.2815  1.4350
Satisfaccion->Lealtad   0.7640        0.7103      0.8202    0.1099  0.4634
```

```
                        deg.fr  p.value  sig.05
Apoyo->Valor               179   0.3576      no
Apoyo->Satisfaccion        179   0.1117      no
Asesoria->Valor            179   0.2738      no
Asesoria->Satisfaccion     179   0.0961      no
Tutoria->Valor             179   0.2437      no
Tutoria->Satisfaccion      179   0.4669      no
Valor->Satisfaccion        179   0.0765      no
Satisfaccion->Lealtad      179   0.3218      no

Inner models in the following objects:
$global
$group1
$group2
```

La primera parte de la salida es una descripción con los parámetros especificados en `pls` y relativos a la comparación de grupos que se está realizando (método y número de muestras). La segunda parte son los datos contenidos en `$test`. La primera columna, `global`, muestra los coeficientes de ruta del modelo global; la segunda y tercera columna muestran los coeficientes de ruta para cada grupo, respectivamente y la cuarta columna `diff.abs` es la diferencia absoluta entre los coeficientes de ruta de ambos grupos. Las columnas `t.stat`, `deg.fr`, y `p.value` contienen el estadístico del t-test, sus grados de libertad y el p-valor asociado. La última columna, `sig.05`, es sólo una etiqueta auxiliar para indicar si la diferencia de los coeficientes de ruta es significativa a un nivel del 5 %.

En este caso, ninguno de los coeficientes de ruta entre los grupos de hombres y mujeres pueden considerarse significativamente diferentes.

4.4.1.2. El procedimiento de permutación

```
(edu_perm = plspm.groups(edu_pls3, education$gender, method ="permutation"))
```

```
GROUP COMPARISON IN PLS-PM FOR PATH COEFFICIENTS

Scale of Data:       TRUE
Weighting Scheme:    centroid
Selected method:     permutation
Num of replicates:   100

$test
                        global  group.female  group.male  diff.abs
Apoyo->Valor            0.6702        0.6466      0.6910    0.0444
Apoyo->Satisfaccion     0.1004        0.0449      0.2596    0.2148
Asesoria->Valor         0.1727        0.1459      0.2445    0.0986
Asesoria->Satisfaccion  0.3764        0.2587      0.4435    0.1848
Tutoria->Valor          0.0820        0.1193      0.0115    0.1079
Tutoria->Satisfaccion   0.1312        0.1218      0.1554    0.0337
Valor->Satisfaccion     0.3493        0.4496      0.1681    0.2815
Satisfaccion->Lealtad   0.7640        0.7103      0.8202    0.1099
```

```
                        p.value  sig.05
Apoyo->Valor             0.7327      no
Apoyo->Satisfaccion      0.2574      no
Asesoria->Valor          0.3960      no
Asesoria->Satisfaccion   0.4653      no
Tutoria->Valor           0.3960      no
Tutoria->Satisfaccion    0.8119      no
Valor->Satisfaccion      0.2871      no
Satisfaccion->Lealtad    0.5446      no

Inner models in the following objects:
$global
$group1
$group2
```

En este caso la salida es muy smilar a la anterior sólo que la parte correspondiente a `$test` sólo contiene los p-valores. Como en el caso anterior, a un nivel de significación del 5 %, ninguno de los coeficientes de ruta pueden considerarse significativamente distintos.

Las diferencias entre los coeficientes de ruta puede visualizarse mediante el siguiente gráfico.

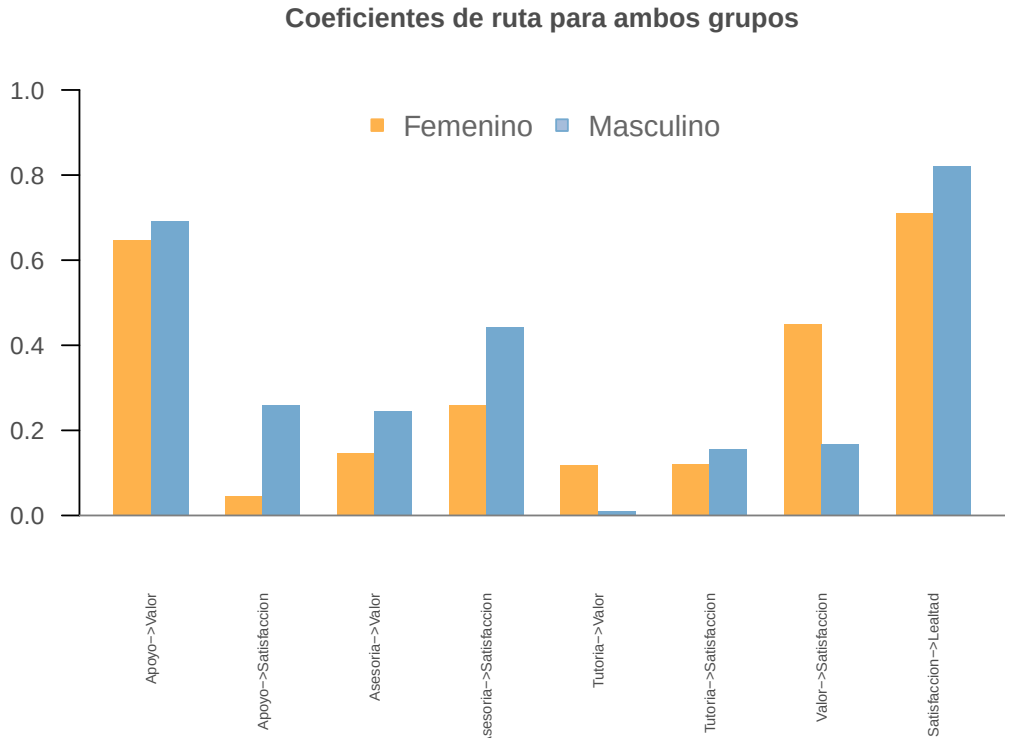

Figura 4.14: Diferencias de coeficientes de ruta. Procedimiento de permutación.

4.4.2. Efectos moderadores

En este caso, usaremos una reducción del modelo para el Índice Europeo de Satisfacción del Cliente tomando sólo las variables **Reputación**, **Lealtad** y **Satisfacción**.

4.4.2.1. Enfoque del producto de indicadores

En nuestro modelo tendríamos tres indicadores (variables manifiestas) que son las variables **Reputación**, **Lealtad** y **Satisfacción** mientras que la variable latente sería la variable que llamaremos **Interacción** que surge de la interacción entre **Reputación** y **Satisfacción**. El diagrama de rutas para este caso es el siguiente:

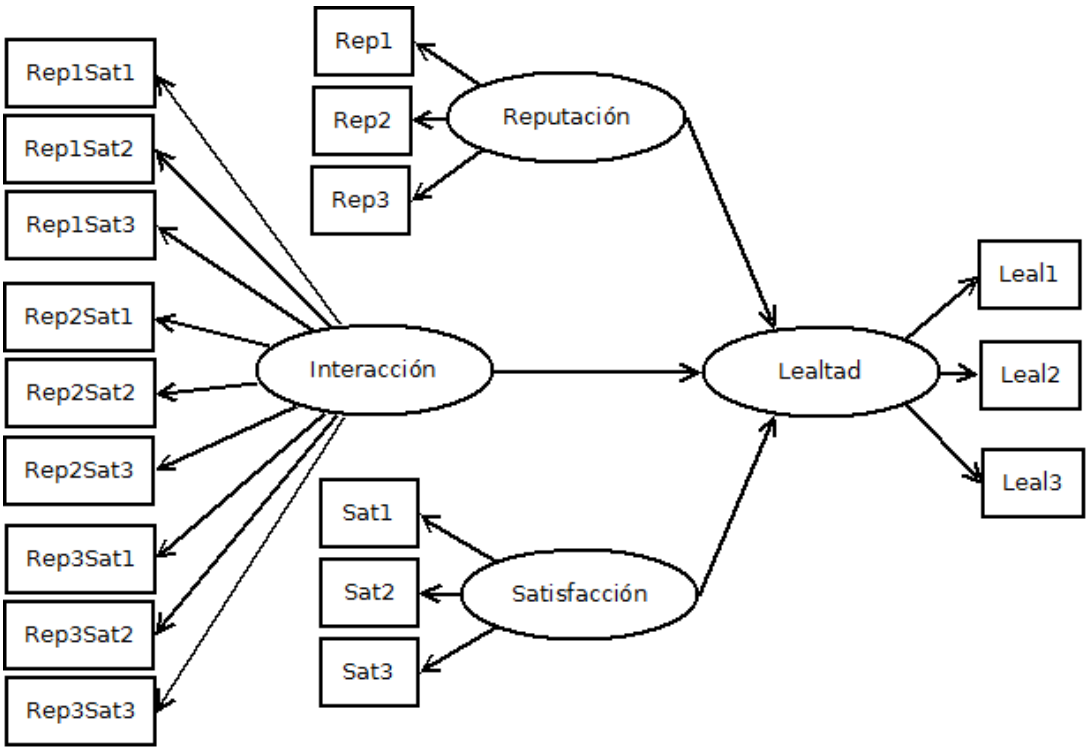

Figura 4.15: Esquema del producto de indicadores

Paso 1

Primero, cargamos los datos que están disponibles en el paquete `plspm`.

```
data(satisfaction)
satisf= satisfaction
```

Una vez que disponemos del conjunto de datos, lo primero que tenemos que hacer es crear los productos de las variables manifiestas que generan las variables manifiestas de la variable latente **Interacción**.

```
satisf$inter1=satisf$imag1*satisf$sat1
satisf$inter2=satisf$imag1*satisf$sat2
satisf$inter3=satisf$imag1*satisf$sat3
satisf$inter4=satisf$imag2*satisf$sat1
```

```
satisf$inter5=satisf$imag2*satisf$sat2
satisf$inter6=satisf$imag2*satisf$sat3
satisf$inter7=satisf$imag3*satisf$sat1
satisf$inter8=satisf$imag3*satisf$sat2
satisf$inter9=satisf$imag3*satisf$sat3
```

Paso 2

Ahora, procedemos a crear los argumentos necesarios para aplicar la función `plspm()` y realizamos el análisis PLS-PM.

```
r1=c(0,0,0,0)
r2=c(0,0,0,0)
r3=c(0,0,0,0)
r4=c(1,1,1,0)
prod_path=rbind(r1,r2,r3,r4)
rownames(prod_path)=c("Reputación","Interacción",
                      "Satisfacción","Lealtad")
colnames(prod_path)=c("Reputación","Interacción",
                      "Satisfacción","Lealtad")
prod_bloques=list(1:3,29:37,20:22,24:26)
prod_modos=rep("A",4)
prod_pls=plspm(satisf,prod_path,prod_bloques,
               modes=prod_modos,boot.val=TRUE,br=200)
```

Paso 3

Observamos los coeficientes de ruta y vemos que la **Interacción** tiene un efecto negativo sobre la **Lealtad**.

```
prod_pls$path_coefs
```

Cuadro 4.26: Coeficientes de ruta. Enfoque del producto de indicadores.

	Reputación	Interacción	Satisfacción	Lealtad
Reputación	0.000000	0.0000000	0.0000000	0
Interacción	0.000000	0.0000000	0.0000000	0
Satisfacción	0.000000	0.0000000	0.0000000	0
Lealtad	0.299946	-0.0457624	0.5225669	0

```
plot(prod_pls)
```

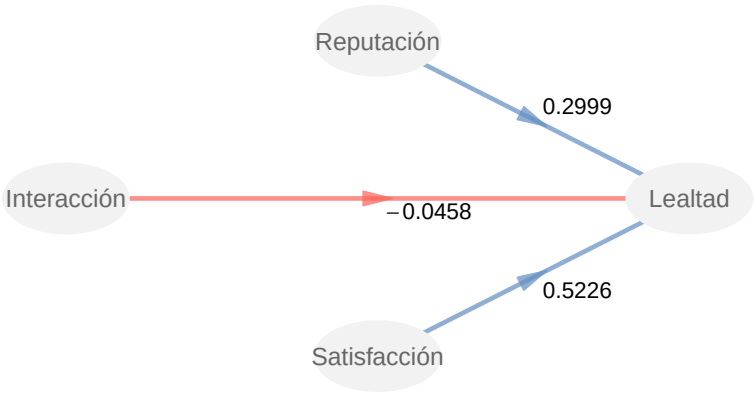

Figura 4.16: Coeficientes de ruta. Enfoque del producto de indicadores.

Comprobamos ahora la significación de los coeficientes de ruta:

```
prod_pls$boot$paths
```

Cuadro 4.27: Significación de los coeficientes de ruta. Enfoque del producto de indicadores.

	Original	Mean.Boot	Std.Error	perc.025	perc.975
Reputación -> Lealtad	0.2999460	0.3187543	0.1437827	0.0452736	0.6063720
Interacción -> Lealtad	-0.0457624	-0.0748047	0.2394528	-0.5201231	0.3867848
Satisfacción -> Lealtad	0.5225669	0.5368102	0.1534816	0.2224987	0.8010296

Aunque **Interacción** tiene un efecto negativo sobre **Lealtad**, su intervalo de confianza bootstrap contiene al cero, por lo que tiene un efecto no significativo.

4.4.2.2. Enfoque del modelo de ruta en dos etapas

Este enfoque involucra dos etapas:

- La primera etapa, que consiste en aplicar un análisis PLS-PM sin el término de interacción.

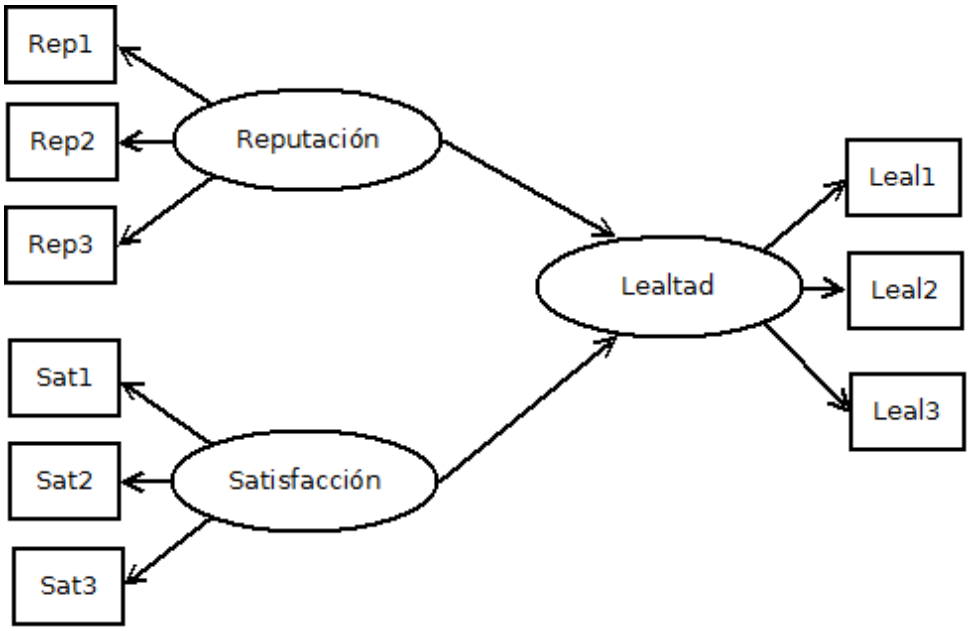

Figura 4.17: Esquema de la primera etapa. Enfoque del modelo de ruta en dos etapas.

En R se haría de la siguiente forma:

```
f1=c(0,0,0)
f2=c(0,0,0)
f3=c(1,1,0)
et1_path=rbind(f1,f2,f3)
rownames(et1_path)=c("Reputacion","Satisfaccion","Lealtad")
colnames(et1_path)=c("Reputacion","Satisfaccion","Lealtad")
et1_bloques=list(1:3,20:22,24:26)
et1_modos=rep("A",3)
et1_pls=plspm(satisfaction,et1_path,et1_bloques,modes=et1_modos)
```

- La segunda etapa, que consiste en tomar las puntuaciones obtenidas del análisis anterior para crear un término de interacción con el que se realizará un análisis PLS-PM incluyendo las puntuaciones como variables manifiestas de las variables latentes.

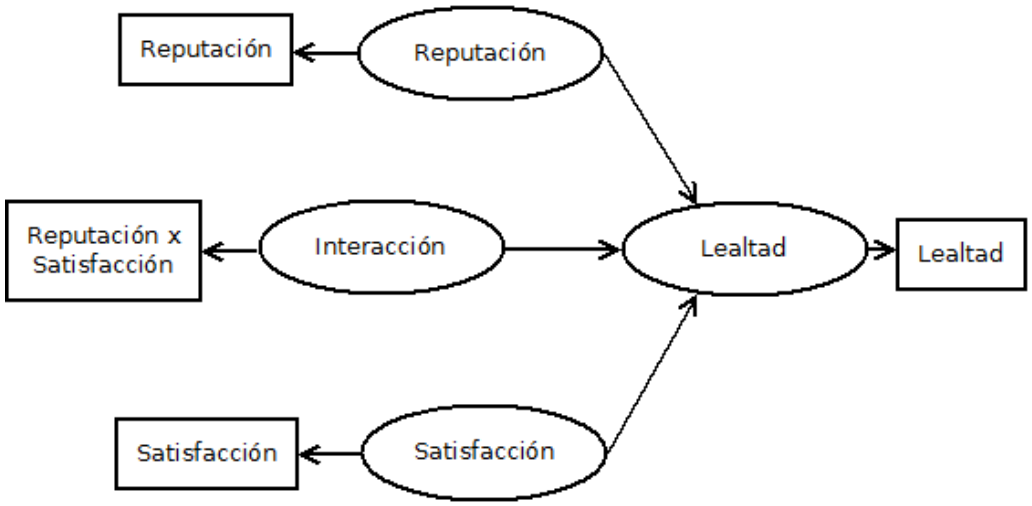

Figura 4.18: Esquema de la segunda etapa. Enfoque del modelo de ruta en dos etapas.

En R se realiza como sigue:

```
Punt=as.data.frame(et1_pls$scores)
Punt$Interaccion= Punt$Reputacion * Punt$Satisfaccion
et2_path=matrix(c(0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0),
                nrow=4,ncol=4,byrow=TRUE)
rownames(et2_path)=c("Reputacion","Interaccion","Satisfaccion","Lealtad")
colnames(et2_path)=c("Reputacion","Interaccion","Satisfaccion","Lealtad")
et2_bloques=list(1,4,2,3)
et2_modos=rep("A",4)
et2_pls=plspm(Punt,et2_path,et2_bloques,modes=et2_modos,
              boot.val=TRUE,br=200)
```

Visualicemos ahora los resultados:

```
round(et2_pls$boot$paths,4)
```

Cuadro 4.28: Significación de los coefs. de ruta. Enfoque del modelo de ruta en dos etapas.

	Original	Mean.Boot	Std.Error	perc.025	perc.975
Reputacion -> Lealtad	0.2769	0.2809	0.0615	0.1514	0.4048
Interaccion -> Lealtad	-0.0005	-0.0076	0.0494	-0.1038	0.0843
Satisfaccion -> Lealtad	0.4957	0.4929	0.0710	0.3477	0.6275

```
plot(et2_pls)
```

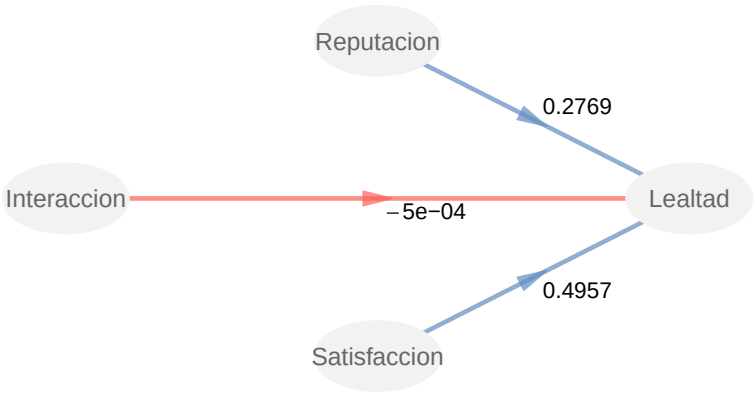

Figura 4.19: Coeficientes de ruta. Enfoque del modelo de ruta en dos etapas.

Ahora, **Interaccion** tiene un efecto negativo muy pequeño sobre **Lealtad** aunque es no significativo ya que su intervalo de confianza bootstrap contiene al cero. Esto quiere decir que el efecto moderador de **Reputación** sobre la relación entre **Satisfacción** y **Lealtad** no es significativo.

4.4.2.3. Enfoque de la regresión en dos etapas

Este enfoque también consta de dos etapas:

- La primera etapa es exactamente igual que la del enfoque del modelo de rutas en dos etapas.

```
reg_pls=et1_pls
PuntR=Punt
```

- La segunda etapa consiste en tomar las puntuaciones obtenidas en la primera etapa y aplicar un análisis de regresión dado por:

$$Lealtad = b_1 Reputación + b_2 Satisfaccion + b_3 Interaccion$$

```
reg=lm(Lealtad~Reputacion+Interaccion+Satisfaccion-1,data=PuntR)
```

Vamos a comprobar ahora los coeficientes de regresión:

```
reg$coefficients
```

Cuadro 4.29: Coeficientes de regresión. Enfoque de la regresión en dos etapas.

Reputacion	0.2769506
Interaccion	-0.0002456
Satisfaccion	0.4957667

El modelo interno obtenido en este caso es el siguiente:

```
c1=c(0,0,0,0)
c2=c(0,0,0,0)
c3=c(0,0,0,0)
c4=c(reg$coefficients,0)
reg_path=rbind(c1,c2,c3,c4)
rownames(reg_path)=c("Reputacion","Interaccion","Satisfaccion","Lealtad")
colnames(reg_path)=c("Reputacion","Interaccion","Satisfaccion","Lealtad")
innerplot(reg_path,show.values=TRUE)
```

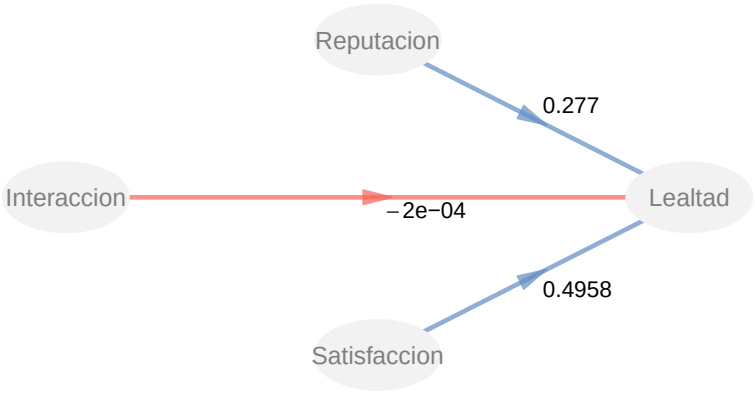

Figura 4.20: Modelo interno. Enfoque de la regresión en dos etapas.

Al igual que en el enfoque anterior, el efecto de la **Interacción** sobre **Lealtad** es muy pequeño. El problema que surge en este enfoque es que no podemos aplicar una validación tipo bootstrap en R y no podemos saber si dicho efecto es o no significativo.

4.4.2.4. Enfoque de la variable categórica

Este enfoque se utiliza cuando la variable moderadora es una variable categórica.

Paso 1 Lo primero que debemos hacer es crear una variable categórica, que en este caso, tendrá tres categorías.

```
categorica=gl(n=3,k=80,length=250)
```

Paso 2 Una vez que tengamos la variable categórica definida, necesitamos definir las variables artificiales que en nuestro caso serían 2 (número de categorías de la variable -1)

```
artif1=rep(0,250) ; artif2=rep(0,250)
artif1[categorica==1]=1
artif2[categorica==2]=1
```

Paso 3 Una vez que las variables artificiales han sido definidas podemos crear los productos indicadores auna vez que hayan sido incluidas en el conjunto de datos.

```
satisfVC=satisfaction[,c(20:22,24:26)]
satisfVC$artif1=artif1
satisfVC$artif2=artif2
satisfVC$sat1m1=satisfVC$sat1*artif1
satisfVC$sat2m1=satisfVC$sat2*artif1
satisfVC$sat3m1=satisfVC$sat3*artif1
satisfVC$sat1m2=satisfVC$sat1*artif2
satisfVC$sat2m2=satisfVC$sat2*artif2
satisfVC$sat3m2=satisfVC$sat3*artif2
```

Paso 4 Una vez que ya tenemos todas las variables necesarias, podemos aplicar el análisis PLS-PM.

```
c1=c(0,0,0,0,0,0)
c2=c(0,0,0,0,0,0)
c3=c(0,0,0,0,0,0)
c4=c(0,0,0,0,0,0)
c5=c(0,0,0,0,0,0)
c6=c(1,1,1,1,1,0)
cat_path=rbind(c1,c2,c3,c4,c5,c6)
rownames(cat_path)=c("Satisf","M1","SatifM1",
                     "M2","SatisfM2","Lealtad")
colnames(cat_path)=c("Satisf","M1","SatifM1",
                     "M2","SatisfM2","Lealtad")
cat_bloques=list(1:3,7,9:11,8,12:14,4:6)
cat_modos=rep("A",6)
cat_pls=plspm(satisfVC,cat_path,cat_bloques,
              modes=cat_modos,boot.val=TRUE)
```

Veamos los resultados del procedimiento bootstrap que obtenemos:

```
round(cat_pls$boot$paths,4)
```

Cuadro 4.30: Significación de los coeficientes de ruta. Enfoque de la variable categórica.

	Original	Mean.Boot	Std.Error	perc.025	perc.975
Satisf -> Lealtad	0.6378	0.5975	0.1067	0.3995	0.8193
M1 -> Lealtad	-0.0196	-0.2420	0.3135	-0.8114	0.3990
SatifM1 -> Lealtad	0.0030	0.2275	0.3013	-0.3840	0.7876
M2 -> Lealtad	0.0363	-0.0112	0.2756	-0.5749	0.5565
SatisfM2 -> Lealtad	0.1179	0.1687	0.2514	-0.3413	0.6407

Podemos observar que todos los intervalos de confianza bootstrap contienen al cero salvo el que corresponde a la relación entre **Satisfacción** y **Lealtad**.

Ahora, veamos el modelo interno correspondiente (en un caso real en el que la variable categórica no sea artificial, debería estudiarse también el modelo externo para hacer una evaluación completa del modelo).

```
plot(cat_pls)
```

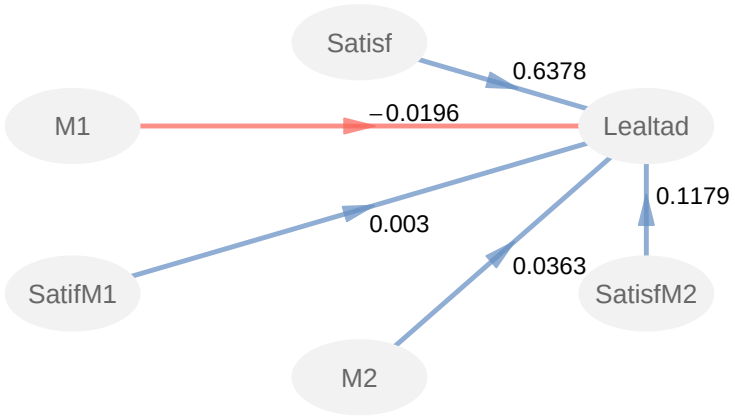

Figura 4.21: Modelo interno. Enfoque de la variable categórica.

Bibliografía

[1] Duarte, P., Raposo, M., Vinzi, V., Chin, W., Henseler, J. and Wang, H. 2010. Handbook of partial least squares: Concepts, methods and applications. Springer Heidelberg, Germany.

[2] Ravand, H. and Baghaei, P. 2016. Partial least squares structural equation modeling with r. *Practical Assessment, Research & Evaluation*. 21, 11 (2016), 1–16.

[3] Russolillo, G. "An introduction to partial least squares path modeling". Disponible en http://maths.cnam.fr/IMG/pdf/pls_pm_cle4bee88.pdf.

[4] Sanchez, G. 2013. PLS path modeling with r. *Berkeley: Trowchez Editions*. 383, (2013).

[5] Sanchez, G., Trinchera, L. and Russolillo, G. 2017. *Plspm: Tools for partial least squares path modeling (pls-pm)*.

[6] Shmueli, G., Ray, S., Estrada, J.M.V. and Chatla, S.B. 2016. The elephant in the room: Predictive performance of pls models. *Journal of Business Research*. 69, 10 (2016), 4552–4564.

[7] Tenenhaus, M., Vinzi, V.E., Chatelin, Y.-M. and Lauro, C. 2005. PLS path modeling. *Computational statistics & data analysis*. 48, 1 (2005), 159–205.