NOVA IMS Information Management School

MEGI

Mestrado em Estatística e Gestão da Informação

Master Program in Statistics and Information Management

Implementação do algoritmo PLS-SEM em R

Inês Câncio Reis Pinto

Dissertação apresentada como requisito parcial para obtenção do Grau de Mestre em Estatística e Gestão de Informação

NOVA Information Management School
Instituto Superior de Estatística e Gestão da Informação
Universidade Nova de Lisboa

Prof. Doutor Jorge M. Mendes, Orientador

Implementação do algoritmo
PLS-SEM em R

Este documento foi gerado utilizando o processador (pdf)LaTeX, com base no template "unlthesis" [1] desenvolvido no Dep. Informática da FCT-NOVA [2]. [1] https://github.com/joaomlourenco/unlthesis [2] http://www.di.fct.unl.pt

Resumo

O seguinte trabalho foi desenvolvido no âmbito da estatística computacional com o objetivo de implementar o algoritmo PLS-SEM (*Partial Least Squares Structural Equation Modeling*) na plataforma R. Para programar o algoritmo foram exploradas outras funcionalidades do R em alternativa aos pacotes já existentes.

Quanto à estrutura do trabalho, a primeira parte dedica-se ao enquadramento que teve na base a revisão da principal literatura relacionada com o tema. Em seguida é apresentada a metodologia adotada, nomeadamente os procedimentos envolvidos no cálculo do algoritmo, e uma descrição do desenvolvimento do *software*. Posteriormente foram testados dois subconjuntos de dados para o modelo experimental e modelo ECSI. Utilizando outros programas disponíveis, foi possível estabelecer uma análise comparativa dos resultados obtidos. Estes não apresentaram diferenças relevantes, o que levou a uma avaliação positiva sobre desempenho do código.

Para além dos resultados favoráveis, o novo programa permite adequar facilmente os parâmetros às exigências do utilizador aquando a estimação do método PLS-SEM. Igualmente apresenta a vantagem de ser uma linguagem simples e acessível enquanto código aberto. A importância do uso desta metodologia tem especial destaque no apuramento dos índices de satisfação do cliente, sendo que no atual contexto do mercado deve ser adaptado e revisto para fazer face aos novos paradigmas. É neste sentido que o presente trabalho disponibiliza livremente o seu contributo à comunidade científica em via do desenvolvimento e aperfeiçoamento da técnica.

Palavras-chave: *Structual Equation Models*, *Partial Least Squares*, R, ECSI

Abstract

The work here presented within the field of computational statistics had the goal of implementing PLS-SEM algorithm in the R software. As an alternative to the existing packages, new functionalities were explored.
Concerning the work's structure, the first part lays down a review on the literature related to this theme. The methodology is then presented, namely the procedures that lead to the construction of the algorithm and also a description of the software development. Two datasets where used to test both experimental model and ECSI model. A comparative analysis of the achieved results was possible using other available programmes. The outputs presented no dissimilarities, which led to evaluate positively the code's performance.
Good results aside, the new program foresees the estimation of the PLS-SEM method simply by adjusting the parameters to the user's requirements. As an open source, it also has the advantage of being a simple and manageable language.
This method is important for the calculation of customer satisfaction index. In the market field, this approach can be adapted and reviewed so it can meet up the new circumstances. Due to this fact, the work presented shares its conclusions with the scientific community for the purpose of developing and perfecting this technique.

Keywords: Structual Equation Models, Partial Least Squares, R, ECSI

Índice

Índice **xi**

Lista de Figuras **xiii**

Lista de Tabelas **xv**

Listagens **xvii**

1 Introdução **1**

2 O Método PLS-SEM **5**

2.1 Estado da Arte 5

2.1.1 Enquadramento Geral 6

2.1.2 Enquadramento Histórico 7

2.2 Metodologia PLS-SEM 10

2.2.1 Modelo de Medida 11

2.2.2 Modelo Estrutural 16

2.3 Algoritmo PLS-SEM 18

2.3.1 Etapa 1: Estimação dos *scores* das variáveis latentes . . . 19

2.3.2 Etapa 2: Estimação dos pesos finais e *path coefficients* . . 22

2.4 Qualidade e validação do Modelo 24

2.4.1 Métodos de Reamostragem 26

3 Desenvolvimento do *Software* **29**

3.1 O Programa R 29

3.2 Estrutura do código 30

3.3 Carregamento dos dados iniciais 32
3.4 *Outputs* 33

4 Casos Práticos **37**
4.1 Modelo Experimental 37
4.1.1 Resultados experimentais 38
4.2 Modelo ECSI 41
4.2.1 Resultados ECSI 45
4.3 Comparação de Resultados 47
4.3.1 Comparação do Modelo Experimental 48
4.3.2 Comparação do Modelo ECSI 50

5 Conclusão e desenvolvimentos futuros **53**

Bibliografia **55**

A Anexo **59**

B Anexo **63**

Lista de Figuras

2.1 Exemplo de um modelo PLS-SEM simples 10
2.2 Modo Reflexivo 13
2.3 Modo Formativo 15
2.4 Modelo Misto 16
2.5 Fluxograma do algoritmo PLS 18
2.6 Esquema do Método *Bootstrap* 27

4.1 Modelo experimental 38
4.2 Modelo Estrutural ECSI 42
4.3 Modelo de Medida ECSI para os dados da rede móvel 43
4.4 *Loadings* do modelo experimental por programa 48

Lista de Tabelas

4.1 *Loadings* do modelo experimental 39
4.2 *Path coefficients* do modelo experimental 39
4.3 Pesos externos standardizados do modelo experimental 39
4.4 Coeficientes de correlação do modelo experimental 40
4.5 Medidas de validade e fiabilidadde do modelo experimental 40
4.6 *Loadings* do modelo ECSI 45
4.7 *Path coefficients* do modelo ECSI 45
4.8 Pesos externos standardizados do modelo ECSI 46
4.9 Coeficientes de correlação do modelo ECSI 46
4.10 Medidas de validade e fiabilidadde do modelo ECSI 47
4.11 *Path coefficients* do modelo experimental por programa 49
4.12 Correlação entre as variáveis latentes por programa 49
4.13 R^2 por programa do modelo ECSI 50

Listagens

3.1 Ficheiros de importação 33
B.1 Código desenvolvido 63

Capítulo 1

Introdução

A abundância de dados marca o nosso tempo. Esta tendência crescente tem levado os investigadores a desenvolverem novos métodos e técnicas que possibilitam a transformação dos dados em informação pertinente com implicações práticas no mercado e no dia-a-dia da sociedade. Neste contexto, surgem novas oportunidades e desafios que requerem capacidades analíticas avançadas, especialmente em casos mais complexos.

Em resposta a este cenário, os Modelos de Equações Estruturais (*Structural Equation Modeling* ou SEM) são considerados uma das metodologias mais poderosas e avançadas de entre as técnicas de análise de dados multivariados (Hair Jr et al. 2013). Como qualquer modelo estatístico, a sua principal função é analisar a relação entre as variáveis, combinando, neste caso, diversos aspectos da análise fatorial e regressão linear, com a particularidade de incluir variáveis não observáveis (ou latentes) indirectamente medidas por um conjunto de indicadores. A abordagem *Partial Least Squares* (PLS-SEM ou PLS *path modeling*) prevê que as relações entre as variáveis sejam calculadas a partir de um processo iterativo que envolve a utilização utilização do método Mínimos Quadrados Ordinais (*Ordinary Least Squares* ou OLS).

São considerados dois tipos de métodos de estimação SEM. O primeiro método,

baseado na covariância (*Covariance-based SEM* ou CB-SEM), é utilizado para confirmar, ou rejeitar, relações de causa efeito entre variáveis através do método de estimação da máxima verosimilhança (*Maximum Likelihood* ou ML). Em oposição, o PLS-SEM é utilizado para análises exploratórias, uma vez que tem por objectivo explicar a variância das variáveis dependentes (Hair Jr et al. 2013). Na década de 70, Karl Jöreskog introduziu o primeiro *software*, designado por LISREL, para estimar os modelos associados ao SEM-ML (Sanchez 2013). Sucessivamente, Herman Wold considerou ser um *"hard modeling"*, contrapondo a abordagem PLS como *"soft modeling"* precisamente porque requer poucos pressupostos sobre a distribuição dos dados (Jöreskog 1970). Por conseguinte, esta afirmação contribuiu para a difusão generalizada do uso da técnica PLS, tendo a sua prática crescido exponencialmente nas últimas décadas, com especial destaque no campo das ciências socioeconómicas. Atualmente, encontra-se no centro da investigação marcando uma forte presença em *top jounals* no âmbito de gestão de informação (Hair et al. 2012), com o objectivo de melhoria e diversificação da técnica (Johnson et al. 2001).

Pela importância crescente do fenómeno das variáveis latentes no contexto do mercado, como a perceção dos clientes, o seu comportamento e atitudes, torna-se clara a proeminente utilização do PLS-SEM (Costigliola 2009). Um caso muito concreto é a sua utilização em estudos de satisfação do cliente face a produtos e/ou serviços adquiridos, através de índices que, numa escala de 10 pontos, medem esse grau (Fornell et al. 1996). Na última década, um leque de índices nacionais e internacionais foi introduzido, constituindo uma informação preciosa para o *marketing research* entender a experiência dos consumidores (Johnson et al. 2001). Como afirma um estudo publicado pela Revista Lusófona, "uma vez que é o cliente quem verdadeiramente faz juízo da qualidade, a medição e acompanhamento do seu nível de satisfação constituem uma ferramenta de gestão indispensável para o planeamento e implementação de formas de melhoria" para a organização (Soares et al. 2008). O modelo ECSI (*European Customer Satisfaction Index*) surge assim como o último desenvolvimento neste campo, transversal a quatro indústrias e 11 países da União Europeia (Johnson

et al. 2001), tornando-se um indicador da *performance* da economia nacional e europeia (Fornell et al. 1996). Para o apuramento deste índice sempre foi utilizada a abordagem PLS por não requer muitos pressupostos sobre a relação das variáveis latentes e a normalidade dos dados (O'Loughlin, Coenders et al. 2002). De entre os vários argumentos para a utilização de um Modelo de Equações Estruturais está o facto aproveitar variáveis observáveis para estimar variáveis difíceis de medir, através de dois sub-modelos: o Modelo Estrutural (que integra as relações entre as variáveis latentes) e o Modelo de Medida (que relaciona as variáveis latentes com os respetivos indicadores).

Este projeto apresenta uma primeira secção inteiramente teórica dedicada à definição do método de estimação PLS-SEM e suas características, uma segunda parte focada na estrutura e desenvolvimento do algoritmo associado a este método e uma última parte que incide concretamente na aplicação prática a um modelo experimental e ao modelo ECSI.

Para o presente trabalho foi utilizada a plataforma de desenvolvimento, e simultaneamente linguagem de programação, R. Este disponibiliza uma ampla variedade de técnicas estatísticas e gráficas, que permitem potencializar os resultados. De entre vários pacotes de análise e tratamento de dados que proporciona, o `plspm` (Sanchez et al. 2015) e o `semPLS` (Monecke e Leisch 2012) foram especialmente desenhados para criar as especificações do modelo e ajustá-lo segundo o método de estimação. Embora estes pacotes já permitam uma execução otimizada, o objetivo do trabalho consiste em escrever o *core* do algoritmo PLS-SEM utilizando formas alternativas aos mesmos. O motivo de escolha do R prende-se essencialmente com a facilidade de personalizar o código conforme as especificidades do estudo, constituindo a possibilidade de vir a ser potenciado para desenvolvimentos futuros. Esta particularidade deve-se à importância crescente que esta ferramenta estatística tem ganho ultimamente, e por isso o presente trabalho é disponibilizado livremente a toda a comunidade científica.

O seguinte projeto encontra-se organizado da seguinte forma: no segundo capítulo será introduzida a abordagem PLS-SEM, concretamente o modo de estimação dos parâmetros e dos modelos externo e interno. O terceiro capítulo incide na formalização do algoritmo, com especial enfoque no desenvolvimento do *software*. Já no quarto capítulo serão apresentados dois casos práticos para os modelos experimental e ECSI, seguido de uma comparação com os resultados provenientes de outros programas (SmartPLS e XLSTAT). Por fim, a última secção será dedicada às conclusões e possíveis desenvolvimentos futuros.

Capítulo 2

O Método PLS-SEM

2.1 Estado da Arte

A análise de dados sempre se revelou uma ferramenta essencial ao longo do tempo, especialmente no campo das ciências sociais. Pelo emergente desenvolvimento tecnológico e computacional, cada vez mais voltado para a vertente *user friendly*, o recurso a métodos estatísticos cresceu exponencialmente. No princípio, eram utilizadas análises uni e bivariadas para descrever a relação dos dados, mas rapidamente caíram em desuso pela complexidade de novas relações. Neste sentido, foi necessário desenvolver outras práticas mais sofisticadas de análise (Hair Jr et al. 2013).

Como o próprio nome indica, a análise de dados multivariados envolve a aplicação de técnicas que estudam simultaneamente o comportamento de múltiplas variáveis. Associado a estes métodos, Fornel distinguiu duas abordagens (Fornell 1985). O recurso às técnicas de primeira geração, que incidem essencialmente na pesquisa exploratória, e as técnicas de segunda geração, que estão destinadas a confirmar uma teoria (Guarino 2004). Estas últimas não só são consideradas mais poderosas que as primeiras, como permitem ultrapassar algumas das suas limitações. De entre várias técnicas existentes, destacam-se os

Modelos de Equações Estruturais como uma das mais estudadas nas últimas décadas.

2.1.1 Enquadramento Geral

Os Modelos de Equações Estruturais (SEM) permitem estimar as relações causais entre variáveis, definidas por um modelo teórico. A natureza destas relações não é diretamente observável, pelo que são utilizados um ou mais indicadores para as medir. O principal foco desta técnica está na capacidade de poder analisar a complexidade de um sistema, com base num conjunto de conceitos latentes e indicadores, dados pelas Variáveis Latentes e Variáveis Manifestas, respetivamente.

No contexto de outras técnicas de análise de dados, o SEM representa a ponte entre o *Path Analysis* (PA) e *Confirmatory Factor Analysis* (CFA) (Trinchera 2007). Especificamente, o contributo do CFA está subjacente à ideia de que um bloco de variáveis expressa diferentes faces de um mesmo conceito. Por outro lado, o PA constitui um modelo relacional entre as variáveis medidas direta ou indiretamente.

Para estimar este modelo, pode ser escolhida uma de duas possíveis abordagens: *Covariance-based* ou *Variance-based*. A primeira, baseada na covariância, enquadra-se no âmbito da pesquisa confirmatória e é utilizada para confirmar relações entre múltiplas variáveis testadas empiricamente. Esta procura minimizar a diferença entre a matriz de covariâncias do modelo e a matriz de covariâncias da amostra. Alguns dos métodos utilizados para a estimação CB-SEM são: *Maximum Likelihood*, *Generalized Least Squares*, entre outros. Em contraste, a segunda abordagem, conhecida por *Partial Least Squares* (PLS-SEM) ou *Path Modeling* (PLS-PM), é utilizada para desenvolver teorias no contexto de uma pesquisa exploratória. O seu propósito é maximizar a variância explicada entre as variáveis dependentes do modelo, i.e. o valor do R^2.

A escolha entre estes métodos estatísticos difere nas características e nos objetivos de cada estudo em particular. Wold ao introduzir o PLS-SEM como *soft modeling*, por superar algumas hipóteses estritas do CB-SEM, contribuiu para a

difusão e popularização desta técnica. Contudo, ambas podem ser vistas como complementares ao SEM (Hair et al. 2013).

2.1.2 Enquadramento Histórico

Herman Wold é considerado por muitos o pai do PLS (Sanchez 2013). A origem do estudo data de 1966 com a publicação do livro *Research Papers in Statistics* (Neyman et al. 1966), que posteriormente introduziu o novo procedimento *Nonlinear Estimation by Iterative Least Squares Procedures*, ficando conhecido pelo acrónimo NILES. Três anos mais tarde, as palavras *Partial Least Squares* foram referenciadas pela primeira vez na publicação do artigo *Nonlinear Iterative Partial Least Squares Estimation Procedure* (Wold e Lyttkens 1969), substituindo o acrónimo anterior por NIPALS. Este trabalho sobre algoritmos iterativos foi sendo desenvolvido, até que surge em 1973 outra publicação semelhante denominada *NIPALS Modelling: Some Current Development* (Wold 1973).

Paralelamente, no início dos anos 70, Karl Jöreskog foi pioneiro a utilizar o computador para a estimação *Maximum Likelihood* (Jöreskog e Sorbom 1993). Foi introduzido, assim, o *software* LISREL (*Linear Structural Relations*), capaz de aplicar os métodos *Covariance-based* (Trinchera 2007). Motivado por este tipo de modelos, Wold adaptou o seu trabalho NIPALS ao SEM, permitindo que as variáveis latentes fossem indiretamente medidas por múltiplos indicadores. Alguns dos trabalhos que refletem o desenvolvimento e transformação desta prática podem ser encontrados nas seguintes referências bibliográficas: Wold (1974), Noonan e Wold (1977), Wold (1980).

A versão finalizada da abordagem PLS surge no final da década, após um exercício conjunto de Wold e Jöreskog, que resultou na publicação de *Systems Under Indirect Observation: causality, structure, prediction* em dois volumes (Friedrich et al. 1984). A Parte I é dedicada ao LISREL e a Parte II dedicada ao PLS. Dado por terminado o seu contributo para o aperfeiçoamento da técnica, em 1982 Wold afirmou ter chegado ao que considera a "estação final: o projeto para a estimação PLS *Path Models* através de variáveis latentes ou simplesmente PLS *soft modeling*" (Sanchez 2013).

Nos anos seguintes, os métodos PLS viveram uma época de expansão e consolidação, tendo sido maioritariamente desenvolvidos na perspectiva da regressão PLS. Em particular, esta técnica resultou num enorme sucesso no campo quimiométrico. De forma a evitar a confusão entre PLS *Regression Models* e PLS *Path Modeling* foi acordado que a segunda seria aplicada no contexto dos Modelos de Equações Estruturais (Tenenhaus et al. 2005), assumindo um papel importante para o *Marketing*, Estudos de Mercado, Gestão Estratégica e Sistemas de Informação (Hair et al. 2012). Contudo, ambos os métodos fazem parte da abordagem PLS.

O primeiro programa de *software* aplicado ao PLS-SEM foi desenvolvido por Jan-Bernd Lohmöller na década de 80, com o nome LVPLS. Durante muitos anos, este foi o único programa disponível. No seu livro *Latent Variable Path Modeling with Partial Least Squares* (Lohmöller 1989), o autor deu a entender que numa questão de tempo o método seria aplicado em diversas áreas no âmbito das ciências sociais. De facto, durante os anos 90, o método sofreu com a falta de popularidade, tendo sido apenas restabelecida com o trabalho de Wynne Chin através do programa PLS-Graph (Chin 2001). Este tornou-se o primeiro *software* capaz de permitir ao utilizador criar uma representação gráfica do modelo PLS-SEM.

A situação inverteu-se com a chegada do novo milénio. Um grande impulso para a difusão da técnica deveu-se à Escola Francesa de Análise de Dados e principalmente ao projecto *European Satisfaction Index System* (ESIS). Além disso, uma série de simpósios internacionais exclusivamente dedicados aos métodos PLS contribuíram para a sua utilização generalizada em todo o mundo (Sanchez 2013).

Hoje em dia, os utilizadores desta área encontram um vasto leque de possibilidades para a escolha do *software* que pretendem usar. Os desenvolvimentos progressivos de ferramentas *user friendly* não só permitiram melhorar a facilidade de utilização, como contribuíram para a disseminação desta técnica entre as mais diversas áreas. De entre a variedade de programas disponíveis no mercado, destacam-se pela sua popularidade: LVPLS, VirtualPLS, PLS-Graph,

SPAD e SmartPLS (Temme et al. 2010). Outros desenvolvimentos recentes incluem o XLSTAT-PLSPM e os pacotes `semPLS` e `plspm` disponibilizados pela plataforma R (Monecke e Leisch 2012). Para utilizadores que pretendam uma interface mais gráfica, é aconselhado o SmartPLS ou o XLSTAT-PLSPM, sendo que este último tem a desvantagem de apenas ser distribuído comercialmente, ao contrário do primeiro que se distingue por ser uma "fonte aberta".

2.2 Metodologia PLS-SEM

A abordagem *Partial Least Squares* dos Modelos de Equações Lineares estuda ligações complexas entre variáveis latentes e observáveis através de relações lineares. De forma a simplificar a análise comportamental destas variáveis é possível estabelecer uma representação gráfica a partir de um diagrama – os chamados *path models*. Os objectos que constituem este modelo são dados por:

- *Elipse* ou *círculo*: representam as variáveis latentes ou não observáveis (Y_1, Y_2 e Y_3 no exemplo da Figura 2.1);

- *Rectângulo* ou *quadrado*: são utilizados para distinguir as variáveis manifestas (x_1 a x_8 no exemplo da Figura 2.1);

- *Setas*: apresentam as relações entre as variáveis latentes e manifestas e vice-versa. No contexto PLS-SEM, tratam-se apenas de relações unidirecionais que suportam a teoria da causalidade. Ou seja, contrariamente à abordagem CB-SEM, a cada variável manifesta só é permitida a conexão com uma variável latente.

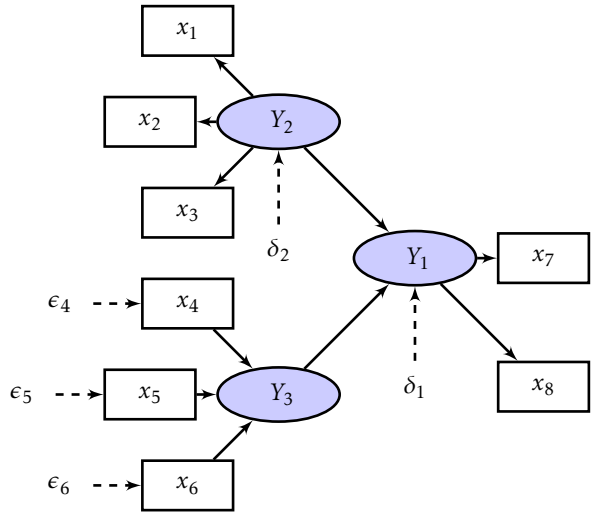

Figura 2.1: Exemplo de um modelo PLS-SEM simples

Os *path models* admitem apenas relações recursivas expressas a partir de um diagrama simples. Este facto implica por um lado a inexistência de *loops* e, por outro, que existe pelo menos uma relação entre um par de variáveis. Consequentemente, nenhuma variável se encontra isolada das restantes.

Para aplicar este método são necessárias algumas considerações prévias (Hair Jr et al. 2013): a dimensão da amostra deve ser pelo menos 10 vezes o número de relações (setas) de qualquer variável latente presente no modelo (*10 times rule*); não é requerida a normalidade dos dados; e a escala de medida utilizada é geralmente métrica, sendo que também é aceite a escala ordinal.

A metodologia PLS-SEM integra dois elementos: o Modelo Estrutural (*inner model*) e o Modelo de Medida (*outer model*). O primeiro diz respeito às ligações entre as variáveis latentes (os *paths*), enquanto que o segundo é relativo às relações entre estas e os indicadores. Os termos de erro também podem estar representados no diagrama. Por definição, dizem respeito à variância não explicada aquando a estimação do modelo, estando associados tanto às variáveis latentes (δ_1 e δ_2 na Figura 2.1), como aos indicadores (ϵ_4, ϵ_5 e ϵ_6 na Figura 2.1).

2.2.1 Modelo de Medida

A teoria assente no Modelo de Medida especifica como os conceitos latentes são medidos. Uma variável não observável, dada por y, é descrita com base num bloco de outras variáveis observadas, definidas como variáveis manifestas (ou indicadores) x_g.

Segundo a abordagem PLS, uma variável manifesta apenas pode estar relacionada com uma única variável latente e um bloco deve conter pelo menos um indicador. A forma como o bloco se relaciona com a respetiva latente pode ser dada por três modos distintos: Reflexivo (ver Figura 2.2), Formativo (ver Figura 2.3) ou Misto (ver Figura 2.4).

Aplicando qualquer um destes modos têm de ser respeitadas as seguintes hipóteses para todas as etapas do algoritmo:

1. Todas as variáveis manifestas contidas na matrix X devem ser standardizadas, ou seja, ter média zero e variância igual a 1;

2. Cada bloco de variáveis manifestas X_g deve ser previamente transformado para garantir correlações positivas com todas as variáveis latentes y_g, $g = 1, ..., G$.

Modo Reflexivo

Optando pela forma reflexiva (ou Modo A), cada bloco de variáveis manifestas reflete a sua variável latente a partir de uma regressão multivariada:

$$X_g = y_g \beta_g^\top + \epsilon_g \tag{2.1}$$

onde os pesos, dados por $\beta_g^\top$, podem ser estimados por mínimos quadrados:

$$\begin{aligned} \hat{w}_g^\top &= (y_g^\top y_g)^{-1} y_g^\top X_g \\ &= var(y_g)^{-1} cov(y_g, X_g) \\ &= cov(y_g, X_g) \\ &= cor(y_g, X_g) \end{aligned} \tag{2.2}$$

Seguindo a abordagem PLS, todas as variáveis latentes são estimadas como combinações lineares das variáveis manifestas a que correspondem, tendo em consideração as restrições impostas por Wold. Ou seja, a equação 2.1 tem de obedecer às hipóteses de variância unitária e média igual a zero:

$$E[\epsilon_g | y_g] = 0 \tag{2.3}$$

Esta restrição impõe que o resíduo ϵ_g tenha média nula e não se encontre relacionado com as variáveis latentes y_g.
O diagrama da Figura 2.2 apresenta um caso simples do modo reflexivo, onde

a variável latente y_g é medida em função do bloco de variáveis X_g, que consiste em três variáveis observáveis x_1, x_2 e x_3.

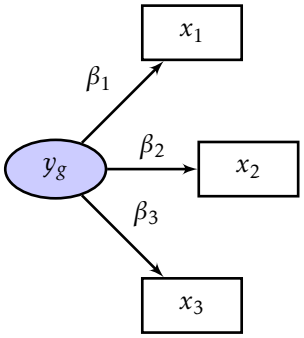

Figura 2.2: Modo Reflexivo

O modo reflexivo assume que o bloco de variáveis manifestas é reflexo de apenas um único conceito latente. Torna-se necessário, portanto, avaliar a presença de unidimensionalidade no bloco. Assim, um dos primeiros critérios a ser testado é a consistência interna. Tipicamente são utilizadas três técnicas:

- α *de Cronbach*: fornece uma estimativa sobre a fiabilidade entre um bloco de indicadores observados e a variável latente correspondente. Este bloco será unidimensional se as variáveis manifestas que o constituem forem altamente correlacionadas (geralmente α superior a 0.7). Consequentemente, espera-se que a média da correlação inter-variáveis também seja elevada. O cálculo deste coeficiente requer que as variáveis observadas sejam standardizadas e positivamente correlacionadas. A fórmula utilizada é a seguinte:

$$\alpha = \frac{\sum_{p \neq p'} cor(x_{pq}, x_{p'q})}{P_q + \sum_{p \neq p'} cor(x_{pq}, x_{p'q})} \times \frac{P_q}{P_q - 1} \tag{2.4}$$

  onde P_q é o número de variáveis manifestas presentes no bloco q.

- ρ *de Dillon-Goldstein*: (mais conhecido por *composite reliability*) foca-se na variância da soma das variáveis que constituem o bloco. Este é considerado multidimensional quando o ρ é superior a 0.7. Esta estatística é dada por:

$$\rho = \frac{(\sum_{p=1}^{P_q} \lambda_{pq})^2}{(\sum_{p=1}^{P_q} \lambda_{pq})^2 + (\sum_{p=1}^{P_q} (1 - \lambda_{pq}^2)} \tag{2.5}$$

sendo λ_{pq} o *loading* da variável manifesta p no bloco q correspondente (Vinzi et al. 2010).

- *Análise de Componentes Principais*: esta técnica envolve a análise dos valores próprios da matriz de correlações dos indicadores. Um bloco é considerado multidimensional se o primeiro valor próprio for superior a 1 e os restantes inferiores, ou bastante afastados deste valor. Posteriormente, é necessário verificar se todas as variáveis presentes no bloco são positivamente correlacionadas com o primeiro factor. Caso contrário, a variável é considerada imprópria para medir o conceito latente, sendo aconselhado a sua remoção do bloco.

Dada a importância de validar a consistência interna, a prática da análise fatorial determina que indicadores podem ser considerados causais (Bollen 1984), ou seja, até que ponto a variável latente explica o bloco de indicadores. Uma vez que esta cláusula encontra-se contemplada na estatística ρ de Dillon-Goldstein, deduz-se que a sua utilização seja mais adequada que o α de Cronbach.

Modo Formativo

No Modo Formativo (ou simplesmente Modo B) a variável latente é formada pelo conjunto de variáveis manifestas que lhe estão associadas. Contrariamente ao Modo Reflexivo, torna-se possível admitir um bloco multidimensional. Por conseguinte, o modelo de medida é então expresso por múltiplas regressões:

$$y_g = X_g \beta_g^\top + \delta_g \tag{2.6}$$

sendo que os pesos β_g são estimados por mínimos quadrados:

$$\begin{aligned} \hat{\beta}_g &= (X_g^\top X_g)^{-1} X_g^\top y_g \\ &= var(X_g)^{-1} cov(X_g, y_g) \\ &= cor(X_g)^{-1} cor(X_g, y_g) \end{aligned} \tag{2.7}$$

Seguindo novamente as especificações de Wold, a equação 2.6 tem presente a seguinte hipótese:

$$E[\delta_g|X_g] = 0 \tag{2.8}$$

A Figura 2.3 apresenta um esquema onde a variável latente y_g resulta da combinação linear das variáveis manifestas x_1, x_2 e x_3 que fazem parte do seu bloco e do respetivo peso β_1, β_2 e β_3. Geralmente, este peso é calculado com base no sinal da correlação entre a variável manifesta x_g e a variável latente y_g.

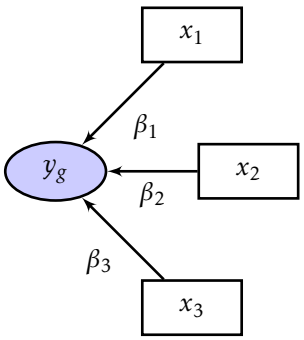

Figura 2.3: Modo Formativo

Modelos Mistos

Quando todos os conceitos latentes são medidos reflexivamente, trata-se de um modelo reflexivo. Por outro lado, se os mesmos conceitos forem todos medidos formativamente, trata-se de um modelo formativo. Uma fusão destes dois modos, resulta no esquema MIMIC (*Multiple Indicators for Multiple Causes*) ou Modelos Mistos (Bollen 1984).

Considere-se $k_g = \{k \in \{1,...,K\} \mid x_k \sim y_g\}$ um conjunto de índices para as variáveis manifestas relacionadas com a variável latente y_g. Os pesos β_g, $g = 1,...,G$ determinam um vector coluna de dimensão $\mid k_g \mid$, que pode ser escrito pela matriz dos pesos externos:

$$W = \begin{pmatrix} \beta_1 & 0 & \cdots & 0 \\ 0 & \beta_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \beta_G \end{pmatrix}$$

As hipóteses estabelecidas por Wold (média nula e variância unitária) mantém-se as mesmas, uma vez que integram os esquemas reflexivo e formativo.

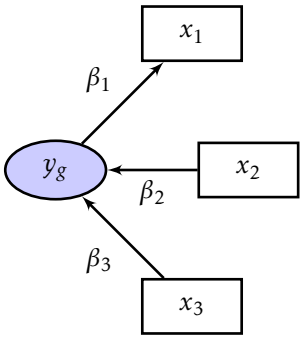

Figura 2.4: Modelo Misto

2.2.2 Modelo Estrutural

Ao ser desenvolvido o Modelo Estrutural devem ser considerados dois princípios: a sequência das variáveis latentes e as suas relações (Hair Jr et al. 2013). O primeiro aspeto tem por base a lógica e as práticas previamente observadas, que suportam a teoria da causalidade do modelo. Quanto à direcção das relações, são identificadas duas classes de variáveis: as exógenas, que não têm qualquer antecessor, e as endógenas, que são explicadas por outras variáveis do modelo.

Em termos globais, a formalização deste modelo pode ser dado pela seguinte expressão:

$$Y = YB + \delta \tag{2.9}$$

onde Y diz respeito à matriz das variáveis latentes, quer sejam endógenas ou

exógenas, e B aos elementos da matriz de coefficientes. O termo de erro ϵ assume-se que seja centrado, i.e., $E[\delta] = 0$.

O Modelo Estrutural pode ser definido por uma matriz triangular de dimensão igual ao número de variáveis latentes presentes no modelo. Trata-se da matriz adjacente D. Quando a entrada $d_{ij} = 1$, a variável latente i explica a variável latente j, caso contrário, a matriz será preenchida com 0. Consequentemente, os elementos de B serão zero quando os elementos da matriz adjacente D forem igualmente nulos.

A abordagem PLS requer ainda que o Modelo Estrutural seja recursivo, o que exclui a utilização de relações causais cíclicas entre as variáveis latentes (Henseler et al. 2012).

2.3 Algoritmo PLS-SEM

O algoritmo PLS-SEM segue duas grandes etapas (Henseler et al. 2012). A primeira procura estimar os *scores* das variáveis latentes através da iteração de quatro passos, enquanto a segunda estima os pesos finais (ou *loadings*) e os *path coefficients*.A Figura 2.5 apresenta um fluxograma da sequência do algoritmo.

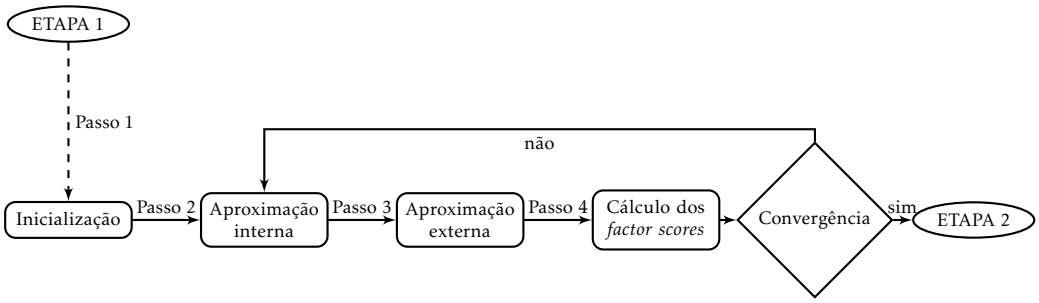

Figura 2.5: Fluxograma do algoritmo PLS

Muito sumariamente, este procedimento tem início com a estimação dos *scores* das variáveis latentes a partir da soma das variáves manifestas correspondentes. Em seguida, a aproximação interna procura reconstruir a variável latente como combinação linear das outras variáveis latentes com ela diretamente relacionadas. Para esta fase estão disponíveis três métodos: centróide, factorial e o esquema estrutural. Na aproximação externa procura-se a melhor combinação linear que expresse cada variável latente em função dos seus indicadores. No passo 4, os pesos externos são calculados por uma de duas formas dependendo do modelo de medida: pela covariância entre os pesos internos de cada variável latente e os seus indicadores (nos modelos reflexivos), ou como regressões ponderadas por OLS (nos modelos formativos). Note-se ainda que, no fim de cada passo, as variáveis latentes são standardizadas. Esta primeira parte termina quando a diferença relativa de todos os pesos externos for inferior, ou igual, a uma tolerância predefinida.

Finalmente, na etapa 2, os pesos externos são utilizados para calcular os *scores* finais das variáveis latentes. Estes integram a regressão OLS para estimar a relação entre as variáveis no modelo estrutural.

2.3.1 Etapa 1: Estimação dos *scores* das variáveis latentes

Passo 1.1: Inicialização

A primeira fase tem por objetivo a estimação dos *scores* das variáveis latentes a partir da soma ponderada dos indicadores. Como mencionado, assume-se que as variáveis manifestas $x_1,...,x_k$ sejam centradas (média $(x_i) = 0$ e $\text{Var}(x_i) = 1$).

Para a primeira iteração, os pesos (*outer weights*) são iguais a um. Consequentemente, pela soma de variáveis centradas, as variáveis latentes têm média nula. Não obstante, é ainda necessário tornar a variância unitária. A expressão generalizada para este passo é a seguinte:

$$\hat{Y} = XM \tag{2.10}$$

sendo M a matriz adjacente para o modelo de medida. Quando a entrada $m_g = 1$, a variável manifesta k é um indicador da variável latente g. Contudo, esta matriz não apresenta qualquer informação sobre a direção ou o modo dos blocos.

As variáveis latentes são então inicializadas como: $\hat{Y} = \hat{y}_1,....\hat{y}_G$. As iterações seguintes utilizam os pesos obtidos posteriormente no passo 1.4.

$$\hat{y}_g = \frac{\hat{y}_g}{\sqrt{var(\hat{y}_g)}}, \quad g = 1,...G \tag{2.11}$$

Passo 1.2: Aproximação interna

Neste passo o algoritmo procede à estimação dos pesos internos (*inner weights*):

$$\tilde{Y} = \tilde{Y}E \tag{2.12}$$

Cada variável latente é estimada através da combinação linear das variáveis latentes adjacentes, segundo a técnica escolhida (*weigting scheme*). Estão disponíveis três cenários:

1. *Esquema do centróide (centroid weigting scheme)*

   Esta técnica estima a matriz dos pesos internos E da seguinte forma:

$$e_{ij} = \begin{cases} sign(r_{ij}) & \text{, se } c_{ij} = 1 \qquad i,j = 1,...,G \\ 0 & \text{, c.c} \end{cases} \tag{2.13}$$

Neste caso os pesos são iguais aos sinais da correlação entre as variáveis latentes i e j.

2. *Esquema fatorial (factorial weigting scheme)*

$$e_{ij} = \begin{cases} r_{ij} & \text{, se } c_{ij} = 1 \qquad i,j = 1,...,G \\ 0 & \text{, c.c} \end{cases} \tag{2.14}$$

Este equema é bastante semelhante ao método do centróide excepto no sinal das correlações entre as variáveis latentes mais próximas, sendo a correlação utilizada diretamente. Quando os valores são próximos de zero, este método é o mais aconselhado (Costigliola 2009).

3. *Esquema estrutural (path weigting scheme)*
Para este esquema, a relação de uma variável latente é determinante. A relação com a variável sucessora é dada pela sua correlação com os pesos internos da matriz E:

$$e_{ij} = \begin{cases} \gamma_j & \text{, para } j \in y_i^{ant} \\ cor(y_i, y_j) & \text{, para } j \in y_i^{suc} \\ 0 & \text{, c.c} \end{cases} \tag{2.15}$$

onde y_i^{suc} define o conjunto de variáveis sucessoras da variável latente y_i. No caso da ligação com a variável antecedente, é determinada por múltiplas relações:

$$y_i = y_i^{ant}\gamma + z_i \qquad i = 1,...,G \tag{2.16}$$

onde $E[z_i] = 0$ e y_i^{ant} define a variável antecedente.

Inicialmente Wold propôs a utilização do método do centróide e mais tarde Lohmöller introduziu os outros dois esquemas, factorial e estrutural (Monecke e Leisch 2012). Quando se verificam correlações próximas de zero, o primeiro

esquema apresenta alguns inconvenientes, mas na prática não representa problemas para estimação dos pesos. Segundo Henserler & Ringle (Henseler et al. 2012), é geralmente preferível utilizar o esquema estrutural por dois motivos: pode ser aplicado em qualquer Modelo de Equações Estruturais e o valor do R^2 para a estimação dos *scores* das variáveis latentes é maximizado.

Para esta etapa é também necessário escalar as variáveis para obter variância unitária. De uma forma genérica, os pesos internos são obtidos pela estimação $\tilde{Y} = \tilde{y}_1, ...\tilde{y}_G$, onde:

$$\tilde{y}_g = \frac{\tilde{y}_g}{\sqrt{var(\tilde{y}_g)}}, \quad g = 1, ...G \tag{2.17}$$

Passo 1.3: Aproximação externa

Este passo procura recalcular os pesos externos, inicialmente iguais a um, com base nos valores das variáveis latentes obtidos na aproximação interna (passo 1.2). Dependendo do modelo de medida adotado, os pesos externos podem ser estimados como:

- *Modo Reflexivo*: coeficiente da regressão multivariada, resultando na covariância entre a estimativa da variável latente e o bloco de variáveis manifestas:

$$\begin{aligned} \hat{w}_g^\top &= (\tilde{y}_g^\top \tilde{y}_g)^{-1} \tilde{y}_g^\top X_g \\ &= cor(\tilde{y}_g, X_g) \end{aligned} \tag{2.18}$$

- *Modo Formativo*: múltiplas regressões a partir da regressão OLS, dada a estimativa da variável latente como termo independente e o bloco de variáveis manifestas como dependente:

$$\begin{aligned} \hat{w}_g &= (X_g^\top X_g)^{-1} X_g^\top \tilde{y}_g \\ &= var(X_g)^{-1} cor(X_g, \tilde{y}_g) \end{aligned} \tag{2.19}$$

Passo 1.4: Cálculo dos *factor scores*

A partir da matriz W, que contém os vectores dos pesos externos $\beta_1, ..., \beta_G$, é possível estimar os *factor scores* através média das variáveis manifestas:

$$\hat{Y}_g = XW \tag{2.20}$$

Resulta assim a estimação dos pesos externos $\hat{Y} = (\hat{y_1}, ..., \hat{y_G})$.

Convergência

Os passos 1 a 4 da primeira etapa sofrem um processo iterativo até chegar à convergência dos pesos. Esta repetição dá-se até que a diferença relativa dos pesos externos entre uma iteração e a seguinte seja inferior a uma tolerância predefinida:

$$\left| \frac{\hat{w}_{kg}^{old} - \hat{w}_{kg}^{new}}{\hat{w}_{kg}^{new}} \right| < \text{tolerância} \qquad \forall k = 1, ..., K \wedge g = 1, ..., G \tag{2.21}$$

Caso se verifique, avança-se para o etapa seguinte.

Regra geral, a tolerância recomendada é 10^{-5}. Este limite assegura a convergência do algoritmo PLS para valores razoavelmente baixos, quando calculadas as diferenças dos *scores* das variáveis latentes em cada iteração (Henseler et al. 2012).

2.3.2 Etapa 2: Estimação dos pesos finais e *path coefficients*

A partir do apuramento dos *factor scores* podem ser estimadas as relações do modelo estrutural - os *path coefficients* - a partir do método OLS. Para cada variável latente $\hat{y}_g$, $g = 1, ..., G$, os coeficientes são dados pela regressão das suas variáveis antecedentes $\hat{y}_g^{ant}$:

$$\begin{aligned} \hat{\beta}_g &= (\hat{y}_g^{ant\ \top} \hat{y}_g^{ant})^{-1} \hat{y}_g^{ant\ \top} \hat{y}_g \\ &= cor(\hat{y}_g^{ant}, \hat{y}_g^{ant})^{-1} cor(\hat{y}_g^{ant}, \hat{y}_g) \end{aligned} \tag{2.22}$$

Desta forma são estimados os elementos $\hat{b}_{ij}, i, j = 1, ..., G$ que compõem a matriz $\hat{B}$ dos *path coefficients*:

$$\hat{\beta}_{ij} = \begin{cases} \hat{\beta}_{ij} & \text{, para } j \in y_i^{ant} \\ 0 & \text{, para } j \in y_i^{suc} \end{cases} \tag{2.23}$$

Esta matriz pode ser interpretada como transitória no modelo estrutural (Monecke e Leisch 2012). Para obter a matriz dos efeitos totais $\hat{T}$ basta calcular a soma de todas estas matrizes intermédias desde o passo 1 ao G:

$$\hat{T} = \sum_{g=1}^{G} \hat{B}^g \tag{2.24}$$

2.4 Qualidade e validação do Modelo

Medir a qualidade do modelo PLS-SEM implica analisar a discrepância entre os valores das variáveis dependentes, quer sejam observadas (caso das variáveis manifestas) ou aproximadas (no caso das variáveis latentes), e o valor previsto pelo modelo. Consequentemente, a qualidade global do modelo é dado pela sua capacidade preditiva.

De forma a testar a qualidade do modelo PLS devem ser validados os modelos de medida e estrutural que o compõe. Concretamente, para o primeiro importa verificar a unidimensionalidade dos indicadores e calculadas as comunalidades. Quanto ao segundo, são frequentemente utilizados três indicadores: os coeficientes de determinação R^2, o índice de redundância e o teste *Goodness-of-Fit* (GoF).

Comunalidade

Este índice tem o propósito de verificar se os indicadores de cada bloco explicam devidamente a sua variável latente. As comunalidades são simplesmente obtidas pelo quadrado dos *loadings* e medem a parte da variância que é partilhada por ambas as variáveis manifestas e latente que constituem o bloco:

$$comunalidade(y_g, X_{gj}) = cor^2(y_g, X_g) = loading^2_{gj} \tag{2.25}$$

Quando o valor da comunalidade é reduzido, o modelo aprepresenta uma ineficiência. Esta informação poderá resultar na eliminação das variáveis em análise de forma a melhorar a sua qualidade.

A comunalidade média de uma variável latente y_g é obtida pelo valor médio de todas as comunalidades j pertencentes ao seu bloco:

$$\overline{comunalidade(y_g)} = \frac{\sum_{j=1}^{J} comunalidade_{gj}}{J} \tag{2.26}$$

Coeficiente de determinação R^2

Para cada regressão no modelo estrutural é possível calcular o R^2. A sua interpretação é semelhante à da regressão linear, ou seja, avalia a capacidade de

ajustamento do modelo em relação aos dados observados. Neste caso concreto, indica a quantidade de variância da variável latente endógena explicada pelas suas variáveis latentes independentes.

De uma forma geral, os valores do R^2 podem ser classificados em três categorias (Sanchez 2013):

1. $R^2 \leq 30$ - Baixo
2. $30 < R^2 < 60$ - Moderado
3. $R^2 \geq 60$ - Elevado

Logicamente, quanto maior o valor do R^2 melhor a capacidade explicativa do modelo, ou neste caso, das variáveis.

Redundância

Este índice mede a variação das variáveis manifestas associadas à variável latente endógena, explicada pelas variáveis latentes não diretamente relacionadas com esta.

O índice de redundância para a variável manifesta j, pertencente ao bloco g é dado por:

$$redundância(y_g, X_{gj}) = loading_{gj}^2 \times R_{gj}^2 \tag{2.27}$$

Assim como a comunalidade, é possível obter a redundância média de cada variável latente y_g:

$$\overline{redundância(y_g)} = \frac{\sum_{j=1}^{J} redundância_{gj}}{J} \tag{2.28}$$

Valores elevados da redundância significam maior capacidade preditiva.

***Goodness-of-Fit* (GoF)**

Uma vez que não existe um único critério para avaliar a qualidade global do modelo PLS, não é possível introduzir testes de inferência estatística para medir a precisão do ajustamento (*goodness of fit*). Em alternativa, podem ser aplicados testes não-paramétricos.

O índice GoF é calculado como a média geométrica do índice de comunalidade médio e o valor do R^2 médio:

$$GoF = \sqrt{\overline{comunalidade} \times \overline{R^2}} \tag{2.29}$$

Este resultado pode ser utilizado para avaliar o desempenho global do modelo, abrangendo o modelo estrutural e de medida. As suas únicas desvantagens prendem-se com o facto de não apresentar tanto um limite que permita avaliar a significância estatística, como orientações para o seu valor aceitável. No entanto, pode ser considerado um indicador da capacidade preditiva do modelo. Neste sentido, a conclusão mais intuitiva é de que maiores valores conduzem a uma apreciação mais positiva, sendo que a comunidade PLS-SEM considera um "bom valor" GoF > 0.7 (Sanchez 2013).

2.4.1 Métodos de Reamostragem

Os métodos de reamostragem são utilizados para validar o modelo quanto à variabilidade dos parâmetros estimados. De uma forma genérica, permitem calcular estimativas a partir de sucessivos conjuntos de dados retirados da mesma amostra. Algumas das técnicas mais conhecidas e utilizadas são o *bootstrapping*, *jack-knife* e *blindfolding*. No âmbito deste projeto foi utilizado o primeiro método.

Método *Bootstrap*

Esta técnica não-paramétrica permite analisar a precisão das estimativas dos parâmetros PLS. O seu procedimento consiste em obter um número determinado de sub-amostras com a mesma dimensão que a amostra original. A selecção das observações é dada através de uma amostragem com reposição, sendo que para obter estimações mais razoáveis é aconselhado um número de sub-amostras superior a 100. O seguinte diagrama ilustra a lógica subjacente a esta técnica.

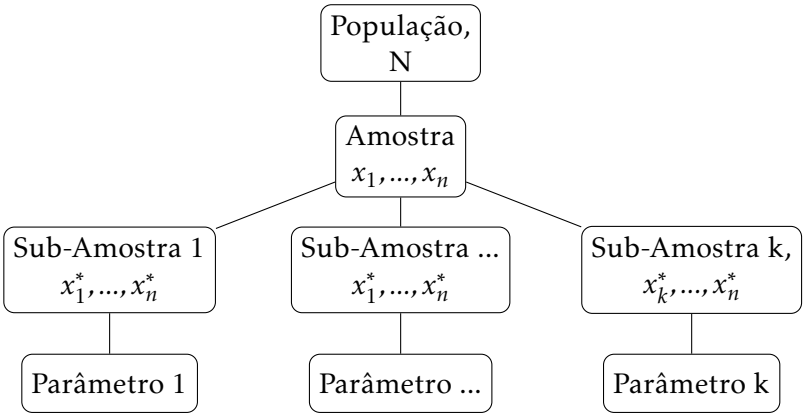

Figura 2.6: Esquema do Método *Bootstrap*

De uma forma genérica, este método engloba três fases, sendo as duas primeiras repetidas k vezes:

1. *Reamostragem*: esta fase baseia-se na extração com reposição de uma sub-amostra de dimensão semelhante à amostra original e estimam-se os parâmetros desejados;

2. *Distribuição da amostra*: neste fase são calculadas algumas estatísticas sobre os parâmetros previamente estimados. Para efeitos deste projeto, foi utilizada a média (2.30) e o desvio-padrão (2.31) em cada sub-amostra:

$$E[\beta^*]_{bootstrap} = \frac{1}{n}\sum_{i=1}^{k}\beta_{ki}^*\frac{1}{k} \tag{2.30}$$

$$Var[\beta^*]_{bootstrap} = \frac{\sum_{i=1}^{k}(\beta_k^* - E[\beta^*]_{bootstrap})^2}{k-1} \tag{2.31}$$

   onde β_k^* diz respeito ao valor do parâmetro estimado na iteração k do procedimento *bootstrapping*.

3. *Teste*: nesta fase pode ser utilizada a estatística pseudo-t para testar a significância de cada coeficiente (*path coefficient*). Utilizando o valor esperado e a variância calculados anteriormente, é possível testar as hipóteses $H_0 : \hat{\beta}_k = 0$ vs $H_1 : \hat{\beta}_k \neq 0$ a partir da seguinte expressão:

$$t_{\tilde{\beta}} = \frac{\hat{\beta}^*}{\sqrt{Var[\hat{\beta}^*]}} \tag{2.32}$$

Capítulo 3

Desenvolvimento do *Software*

3.1 O Programa R

Tendo em vista o objetivo de escrever o *core* do algoritmo PLS-SEM em código aberto, foi utilizada a plataforma estatística R (R Core Team 2014). Uma vez que se encontra em período de expansão, a escolha deste programa contribui de forma positiva para o desenvolvimento do trabalho entre a comunidade científica.

Uma das principais razões para a eleição deste programa deve-se à sua poderosa capacidade de manipulação, exploração e interpretação de dados. O segundo maior motivo relaciona-se com o facto de ser gratuito e independente. Ou seja, não requer contratos, licenças nem *updates* regulares, e é compatível com Windows, MacOS X, Linux ou outro sistema operativo.

Contudo, o que torna o R uma ferramenta realmente útil, e ajuda a explicar o seu forte crescimento e aceitação por parte da comunidade, é a facilidade com que o utilizador pode melhorar e personalizar o código para o seu estudo. A disponibilização de pacotes integrados, que permitem implementar técnicas e algoritmos mais avançados, constitui a sua maior vantagem. Na verdade, nenhum outro *software* comercial oferece em tão pouco tempo a versão mais atual

dos métodos que constituem o estado da arte (Sanchez 2013).

A contribuir para a popularidade desta ferramenta, destacam-se os inúmeros recursos de ajuda, quer em suporte *online* ou físico. Concretamente, estão disponíveis muitos fóruns *online*, grupos de interesse, *blogs*, sites e livros ricos em informação sobre o R. De momento estão disponíveis cerca de 8200 pacotes diferentes em constante atualização e progressivos desenvolvimentos. Em suma, o grande valor acrescentado do R é ser uma fonte aberta de partilha de informação de qualidade.

Para implemetar o algoritmo PLS-SEM, o R disponibiliza um pacote chamado `semPLS` que utiliza principalmente duas funções: o `plsm` e o `semplss`. O primeiro serve para criar as especificações do modelo, e o segundo ajusta o modelo especificado anteriormente através de um conjunto de funções que descrevem os procedimentos do algoritmo (Monecke e Leisch 2012). Todavia, uma vez que o desafio do projeto está em constituir uma fonte de código aberto, este pacote específico não foi utilizado.

3.2 Estrutura do código

Os procedimentos utilizados através do código R para o cálculo do algoritmo PLS-SEM são descritos em seguida (referências das linhas no Anexo B).

A primeira tarefa prende-se com a importação dos dados através da definição dos diretórios dos respetivos ficheiros base (ver secção 3.3). Estes são devidamente transformados em formato matricial para garantir o bom funcionamento do programa (linhas 10 a 37). É então definida uma função que abrange todo o processo calculatório do algoritmo através da definição dos parâmetros que a constituem (linha 40).

A informação sobre o modelo que está a ser estudado é dado pelas variáveis latente (linha 43), manifesta (linha 44) e pelas matrizes das relações e dos pesos iniciais (linhas 45 a 49). Ainda numa fase preparatória, foram estabelecidas algumas funções, designadas auxiliares, que são transversais a várias fases do processo de cálculo. De entre elas, a variância (linhas 53 a 55) e o desvio padrão

(linhas 62 a 64) que estão na base da função de standardização dos dados (linhas 62 a 64). Outras funções servem para a multiplicação de matrizes (linhas 67 a 72) e visualização de variáveis e tabelas (linhas 75 a 81).

O cálculo do algoritmo encontra-se subdividido pelas duas etapas. A primeira, que prevê a estimação dos *scores* das variáveis latentes, começa com a normalização dos dados (linha 89). Relembra-se o facto de que em cada passo os dados são standardizados. É então introduzida a matriz dos pesos que será recalculada ao longo do algoritmo (linhas 92 a 97). A partir desta fase arranca o processo iterativo com delimitação da tolerância recomendada de 10^{-5} (linhas 100 a 102). O segundo passo procede à estimação dos pesos internos através do esquema do centróide (linhas 109 a 118). Posteriormente no passo 3 são recalculados os pesos externos, no começo iguais a 1, com base nos valores obtidos no passo anterior (linhas 121 a 129). O quarto passo procura obter os *factor scores* (linhas 166 a 174) a partir da estimação dos pesos externos (linhas 133 a 163). Estes são obtidos dependendo do modelo de medida que foi adotado. Por fim, é possível visualizar o resultado do processo iterativo do algoritmo, que termina até chegar à convergência dos pesos (linhas 177 a 185)

A segunda etapa procura estimar os pesos finais - os *loadings* (linhas 187 a 198) e as relações do modelo estrutural - os *path coefficients* (linhas 205 a 218), variando novamente consoante o modo formativo ou reflexivo. Também é possível obter a matriz de correlações entre as variáveis latentes do modelo (linhas 201 a 202).

Quase a terminar, são calculados e apresentados os índices que avaliam a qualidade e testam a coerência do modelo estudado. Esses índices são: a comunalidade (linhas 233 a 235), o R^2, α de Cronbach e ρ de Dillon-Goldstein (linhas 238 a 250), redundância (linhas 253 a 258), e GoF (linhas 261 a 262).

Em última instância é apresentado o resultado do método *bootstrap* para a média e o desvio padrão calculados para 100 subamostras (linhas 278 a 291).

3.3 Carregamento dos dados iniciais

Para garantir o bom funcionamento do programa é necessário assegurar que os ficheiros de entrada são carregados corretamente. Uma vez que o código está preparado para receber os dados como tabelas, o procedimento mais fácil é utilizando o formato excel ".xls" (Microsoft Office ®). Assim, a estrutura de qualquer ficheiro de *input* deverá fazer corresponder a cada coluna uma variável. Em seguida é selecionada toda a área que contém os dados base e copiada integralmente para um documento de texto ".txt".

Os ficheiros de texto a serem importados para o R são os seguintes:

- **Dados** - Este ficheiro contém a amostra de dados sobre os quais é utilizado o método PLS-SEM. Cada coluna representa uma variável manifesta (sendo que se tratam das variáveis observáveis) e a primeira linha de cada coluna contém a respetiva identificação;
- **Modelo estrutural** - Neste ficheiro são indicadas as variáveis latentes e as relações que constituem o modelo estrutural, sendo que a ordem importa para estabelecer a direção da relação. Mais concretamente, a primeira coluna (*source*) faz corresponder a variável antecessora, e a segunda coluna (*target*) a variável a que se destina;
- **Modelo de medida** - Este ficheiro diz respeito aos blocos de variáveis, em que a primeira coluna corresponde à variável latente e a segunda coluna às variáveis manifestas que lhe estão associadas;
- **Relações do modelo de medida** - Este ficheiro segue a mesma lógica que o ficheiro do modelo de medida mas com a diferença de a ordem das variáveis determinar a direção da relação. Ou seja, contém indiretamente a informação sobre o tipo de relação, se segue o modo formativo ou reflexivo.

A forma como estes ficheiros devem ser importados está estabelecida no código (nas linhas 10 a 24) e basta especificar a diretoria da pasta onde se encontram.

```
## carregamento dos ficheiros de importacao

# dados
dados <- read.table(paste(base_dir, file_dados, sep=''), header=T)
# modelo estrutural
SMteste <- read.table(paste(base_dir, file_SM, sep=''), header=T)
# modelo de medida
MMteste <- read.table(paste(base_dir, file_MM, sep=''), header=T,
    stringsAsFactors=F)
# matriz auxiliar que  indica a direccao das relacoes (reflexivo ou
    formativo)
AUXteste <- read.table(paste(base_dir, file_aux, sep=''), header=T)
```

Listagem 3.1: Ficheiros de importação

3.4 *Outputs*

Segundo as especificidades do R, para se observar o resultado de qualquer operação é necessário apenas citar uma dada expressão. Assim, em qualquer parte do código desenvolvido é possível referenciar a variável que guardou o resultado pretendido. Contudo, alguns dos *outputs* importantes que vão sendo gerados ao longo do algoritmo são:

Dados standardizados A partir da variável "dadostand" é possível obter os dados iniciais padronizados;

Critério de convergência A variável "tolerance" define o limite de convergência que leva à paragem da parte iterativa do algoritmo;

Iterações Para se obter os resultados de cada iteração a condição deve ser "print_iterations = TRUE";

***Scores* das variáveis latentes** A variável "factor_score" apresenta os valores das variáveis latentes para cada observação;

***Loadings* do modelo de medida** Através desta variável é possível obter os *loadings* que dizem respeito às variáveis manifestas. Para melhor visualização da matriz de resultados;

Path coefficients Os coeficientes que estão associados às relações entre as variáveis latentes são apresentados pela variável "path";

Outer weigths Os pesos externos são disponibilizados pela "matrizR" ou simplesmente os valores através da variável "outerweigths", que foram calculados a partir da regressão entre as variáveis que constituem um bloco;

Correlações As correlações entre todas as variáveis latentes presentes no modelo são apresentadas pela variável "LVs_cor";

$\mathbf{R}^2$ O coeficiente de determinação das regressões do modelo estrutural é obtido facilmente a partir da "matrizR2";

Comunalidade A variável "communality" apresenta o índice de comunalidade que serve para medir a qualidade do modelo externo relativo a cada boco de variáveis latentes;

Redundância A variável "redundancy" apresenta o índice de redundância que permite avaliar a qualidade do modelo estrutural para cada bloco de variáveis endógenas;

α **de Cronbach** A primeira coluna da tabela "testes" contém este índice para cada variável manifesta. A sua função é testar a unidimensionalidade de cada bloco de variáveis manifestas;

ρ **de Dillon-Goldstein** Na segunda coluna da tabela "testes" é possível observar o resultado deste índice que serve igualmente para avaliar a unidimensionalidade;

Goodness of fit A variável "GoF" apresenta o resultado deste índice que permite avaliar a qualidade global do modelo adotado.;

Médias das estimativas A função "arrayMeans" apresenta as médias dos parâmetros que foram estimados para as iterações do método *bootstrap*;

Desvios padrão das estimativas A função "arraySDs" devolve os valores dos desvios padrão calculados pelo método *bootstrap*;

Testes Os valores de teste estão associados aos parâmetros estimados no *bootstrap* e foram calculados pelo rácio entre o valor médio e o desvio padrão, anteriormente referidos.

Capítulo 4

Casos Práticos

4.1 Modelo Experimental

Por forma a testar o algoritmo PLS-SEM desenvolvido, foi utilizado um modelo simples que ilustra a aplicação prática desta metodologia. O pensamento que esteve na sua génese integra ambas as relações entre as variáveis latentes e manifestas, ou seja, os modos reflexivo e formativo.

O seguinte exemplo prefigura a satisfação dos clientes face a um determinado Produto e Serviço que relacionados prevêm a sua Lealdade. Cada uma das seguintes variáveis latentes é dada pelos respetivos indicadores, PROD1, PROD2 e PROD3 para a variável Produto; SERV1, SERV2 e SERV3 para a variável Serviço; e LOYA1 e LOYA2 para a variável Lealdade (Figura 4.1). Este tipo de modelos é muitas vezes utilizado para estudar a *performance* das empresas, a sua posição no mercado e até mesmo para estratégias de *marketing* (Tenenhaus et al. 2005).

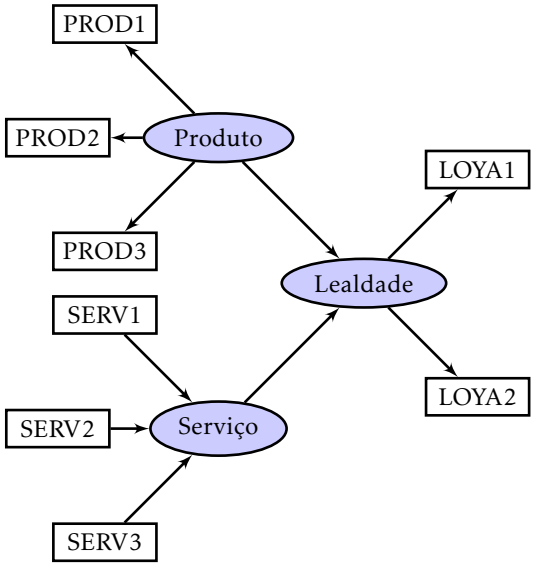

Figura 4.1: Modelo experimental

Formalmente, é possível escrever a equação para o seguinte modelo que contém apenas uma variável endógena:

$$\text{Lealdade} = \beta_{11}\text{Produto} + \beta_{12}\text{Serviço} + \delta_1 \tag{4.1}$$

Os dados utilizados para analisar e comparar os resultados provém de uma amostra de 30 observações aleatórias, que variam na escala entre 1 e 7. Adicionalmente, com vista a conseguir *outputs* com algum significado, os dados que correspondem ao mesmo bloco de variáveis (por exemplo PROD1, PROD2 e PROD3) foram ajustados de forma a terem alguma correlação entre si e com a respetiva variável latente.

4.1.1 Resultados experimentais

Seguindo os procedimentos do algoritmo e a estrutura do código (na secção 3.2), o primeiro *output* gerado apresenta os *loadings* estimados para o Modelo de Medida. Estes foram obtidos através da correlação entre as variáveis latentes e os seus indicadores.

	Loyalty	Produto	Serviço
LOYA1	0.94	0	0
LOYA2	0.96	0	0
PROD1	0	0.90	0
PROD2	0	0.78	0
PROD3	0	0.89	0
SERV1	0	0	0.19
SERV2	0	0	0.44
SERV3	0	0	0.52

Tabela 4.1: *Loadings* do modelo experimental

Posteriormente surgem os coeficientes das relações entre as variáveis latentes do modelo estrutural que representam os chamados *path coefficients*:

	Loyalty
Produto	0.50
Serviço	0.45

Tabela 4.2: *Path coefficients* do modelo experimental

Por fim, são calculados os pesos externos (ou *outer weights*) dependendo do modo como cada bloco se relaciona com a respetiva variável latente.

	Produto	Serviço	Loyalty
LOYA1	0.45	0	0
LOYA2	0.55	0	0
PROD1	0	0.37	0
PROD2	0	0.30	0
PROD3	0	0.33	0
SERV1	0	0	0.17
SERV2	0	0	0.38
SERV3	0	0	0.45

Tabela 4.3: Pesos externos standardizados do modelo experimental

No modo reflexivo os pesos são dados pela regressão dos indicadores e a respetiva variável latente, enquanto no modo formativo, esta é formada pelo conjunto dos indicadores que lhe estão associados.

A análise dos coeficientes de correlação, após a estimação do algoritmo, permite ter uma perceção da relação entre as variáveis latentes do modelo:

	Loyalty	Produto	Serviço
Loyalty	1	0.83	0.82
Produto	-	1	0.75
Serviço	-	-	1

Tabela 4.4: Coeficientes de correlação do modelo experimental

Uma vez apresentados os resultados do algoritmo, são consideradas medidas para validar e testar a fiabilidade do modelo estrutural e de medida (ver secção 2.4). Estes indicadores de qualidade das variáveis latentes encontra-se na seguinte tabela:

	R^2	Comunalidade	Redundância	α de Cronbach	ρ Dillon-Goldstein
Produto	0.78	0.52	0.40	0.82	0.89
Serviço	0	0.66	0	0.82	0.90
Loyalty	0	0.51	0	0.89	0.95

Tabela 4.5: Medidas de validade e fiabilidadde do modelo experimental

Adicionalmente o teste *Goodness-of-Fit* (GoF) resulta em 0.383.

4.2 Modelo ECSI

Após um modelo simples e experimental foi testado um conjunto de dados no âmbito do modelo ECSI. Este segue igualmente a abordagem PLS-SEM baseado em modelos de equações simultâneas e variáveis latentes (Soares et al. 2008).

Face a uma sociedade em constante mudança torna-se necessário adquirir novas formas de análise para melhor acompanhar as suas variações e tendências. Neste contexto, os índices de satisfação do cliente surgem com o objetivo de medir a qualidade dos bens e serviços disponíveis no mercado. No fundo, permitem avaliar o estado financeiro das empresas através da perceção dos seus clientes, tornando-se indicadores de medida da *performance* de uma economia (Fornell et al. 1996).

Nas últimas décadas, um número considerável de barómetros e índices nacionais e internacionais de satisfação do cliente têm vindo a ser introduzidos. Destacam-se, entre outros, o da Suécia (*Swedish Customer Satisfaction Barometer* ou SCSB), Estados Unidos (*American Customer Satisfaction Index* ou ACSI), Noruega (*Norwegian Customer Satisfaction Barometer* ou SCSB) e União Europeia (*European Customer Satisfaction Index* ou ECSI) (Johnson et al. 2001). A maioria destes modelos procura explicar as relações de causalidade entre as dimensões em análise. Concretamente, procura-se estudar conceitos latentes como a qualidade, satisfação, imagem ou lealdade do cliente e para tal são utilizadas *proxies* que medem indiretamente estas variáveis. Ora, o método PLS encaixa nestes requisitos.

A seguinte imagem (Figura 4.2) apresenta as relações causais entre as vaiáveis não observadas que constituem o modelo estrutural do ECSI.

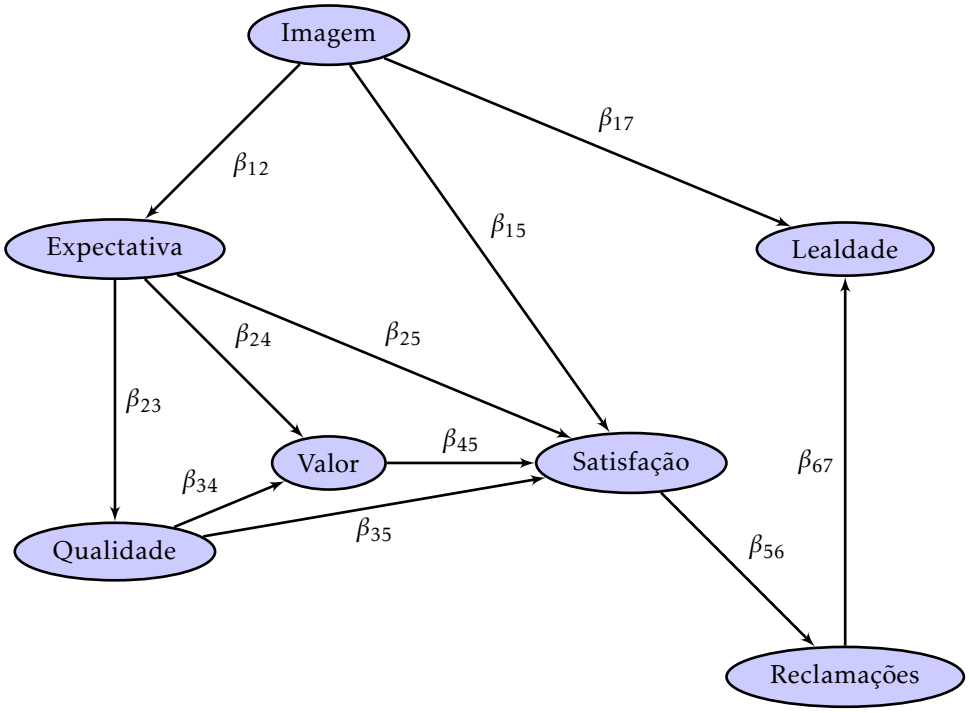

Figura 4.2: Modelo Estrutural ECSI

Formalmente, é possível escrever as seguintes equações para este modelo:

$$\begin{aligned}
\text{Imagem} &= \text{Imagem} + 0 \\
\text{Expectativa} &= \beta_{12}\text{Imagem} + \delta_2 \\
\text{Qualidade} &= \beta_{23}\text{Expectativa} + \delta_3 \\
\text{Valor} &= \beta_{24}\text{Expectativa} + \beta_{34}\text{Qualidade} + \delta_4 \\
\text{Satisfação} &= \beta_{15}\text{Imagem} + \beta_{25}\text{Expectativa} + \beta_{35}\text{Qualidade} + \beta_{45}\text{Valor} + \delta_5 \\
\text{Reclamações} &= \beta_{56}\text{Satisfação} + \delta_6 \\
\text{Lealdade} &= \beta_{17}\text{Imagem} + \beta_{57}\text{Satisfação} + \beta_{67}\text{Reclamações} + \delta_7
\end{aligned} \tag{4.2}$$

Tomado o modelo ECSI, os dados utilizados dizem respeito a um inquérito realizado sobre a satisfação do cliente face aos operadores de rede móvel (Tenenhaus et al. 2005). A descrição dos instrumentos utilizados, bem como os próprios dados, encontram-se disponíveis na página `help > ECSImobi` do pacote `semPLS` do R (ver Anexo A).

A amostra é constituída por 250 observações para 24 variáveis manifestas medidas numa escala de 1 a 10 (onde 1 representa a conotação mais negativa e 10 a conotação mais positiva). Estas variáveis servem para medir os conceitos

latentes: Imagem, Expectativa, Qualidade apercebida, Valor apercebido, Satisfação do Cliente, Lealdade e Reclamações. O modelo de medida (Figura 4.3) pode ser obtido apartir das relações entre as variáveis latentes e de medida.

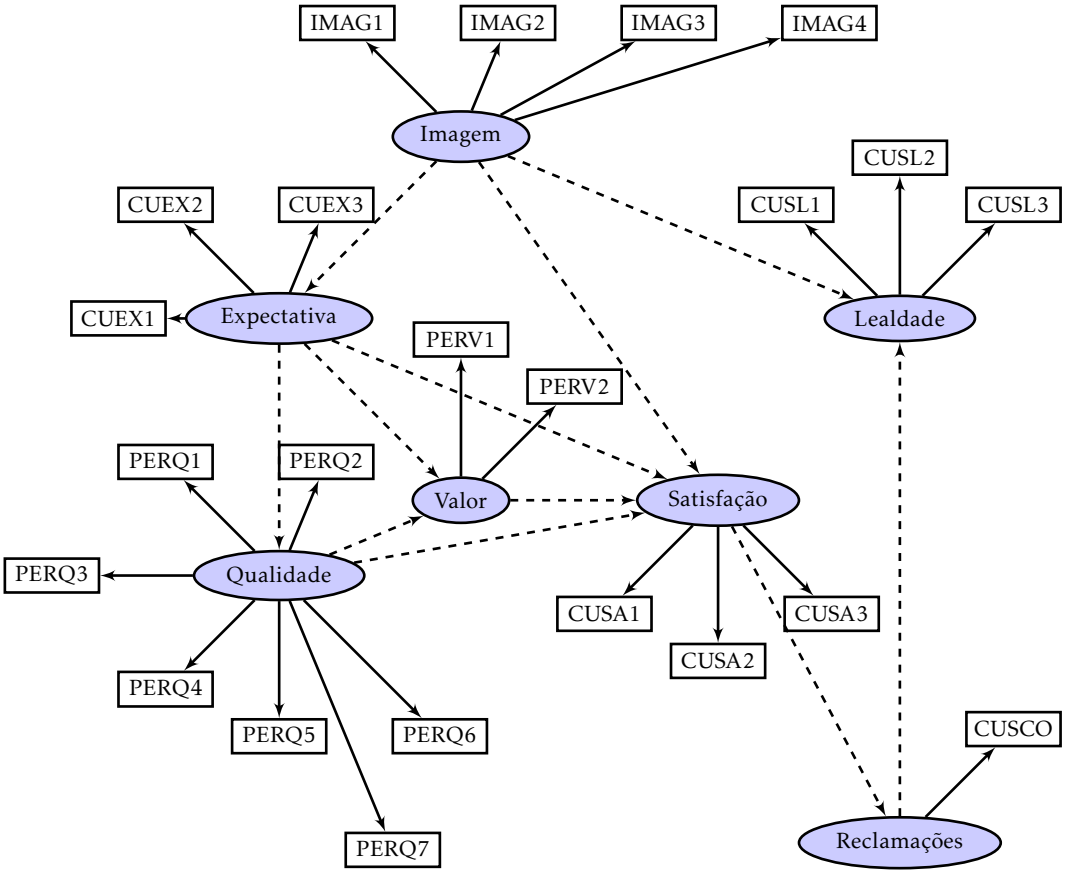

Figura 4.3: Modelo de Medida ECSI para os dados da rede móvel

Dada a vizualização gráfica de ambos os modelos estrutural e de medida, facilmente conseguimos obter as matrizes adjacentes D e M, respetivamente, para o modelo ECSI:

$D =$

	Imagem	Expectativa	Qualidade	Valor	Satisfação	Reclamações	Lealdade
Imagem	**1**	**1**	0	0	**1**	0	**1**
Expectativa	0	0	**1**	**1**	**1**	0	0
Qualidade	0	0	0	**1**	**1**	0	0
Valor	0	0	0	0	**1**	0	0
Satisfação	0	0	0	0	0	**1**	**1**
Reclamações	0	0	0	0	0	0	**1**
Lealdade	0	0	0	0	0	0	0

$M =$

	Imagem	Expectativa	Qualidade	Valor	Satisfação	Reclamações	Lealdade
IMAG1	**1**	0	0	0	0	0	0
IMAG2	**1**	0	0	0	0	0	0
IMAG3	**1**	0	0	0	0	0	0
IMAG4	**1**	0	0	0	0	0	0
IMAG5	**1**	0	0	0	0	0	0
CUEX1	0	**1**	0	0	0	0	0
CUEX2	0	**1**	0	0	0	0	0
CUEX3	0	**1**	0	0	0	0	0
PERQ1	0	0	**1**	0	0	0	0
PERQ2	0	0	**1**	0	0	0	0
PERQ3	0	0	**1**	0	0	0	0
PERQ4	0	0	**1**	0	0	0	0
PERQ5	0	0	**1**	0	0	0	0
PERQ6	0	0	**1**	0	0	0	0
PERQ7	0	0	**1**	0	0	0	0
PERV1	0	0	0	**1**	0	0	0
PERV2	0	0	0	**1**	0	0	0
CUSA1	0	0	0	0	**1**	0	0
CUSA2	0	0	0	0	**1**	0	0
CUSA3	0	0	0	0	**1**	0	0
CUSCO	0	0	0	0	0	**1**	0
CUSL1	0	0	0	0	0	0	**1**
CUSL2	0	0	0	0	0	0	**1**
CUSL3	0	0	0	0	0	0	**1**

4.2.1 Resultados ECSI

Novamente, seguindo os procedimentos do algoritmo e a estrutura do código previamente apresentados, o primeiro *output* gerado são os *loadings*:

	Imagem	Expectativa	Qualidade	Valor	Satisfação	Reclamações	Lealdade
IMAG1	0.90	0	0	0	0	0	0
IMAG2	0.78	0	0	0	0	0	0
IMAG3	0.89	0	0	0	0	0	0
IMAG4	0.59	0	0	0	0	0	0
IMAG5	0.64	0	0	0	0	0	0
CUEX1	0	0.94	0	0	0	0	0
CUEX2	0	0.96	0	0	0	0	0
CUEX3	0	0.78	0	0	0	0	0
PERQ1	0	0	0.68	0	0	0	0
PERQ2	0	0	0.70	0	0	0	0
PERQ3	0	0	0.84	0	0	0	0
PERQ4	0	0	0.70	0	0	0	0
PERQ5	0	0	0.52	0	0	0	0
PERQ6	0	0	0.68	0	0	0	0
PERQ7	0	0	0.77	0	0	0	0
PERV1	0	0	0	0.89	0	0	0
PERV2	0	0	0	0.89	0	0	0
CUSA1	0	0	0	0	0.64	0	0
CUSA2	0	0	0	0	0.70	0	0
CUSA3	0	0	0	0	0.63	0	0
CUSCO	0	0	0	0	0	0.73	0
CUSL1	0	0	0	0	0	0	0.73
CUSL2	0	0	0	0	0	0	0.68
CUSL3	0	0	0	0	0	0	0.88

Tabela 4.6: *Loadings* do modelo ECSI

Em seguida surgem os coeficientes das relações entre as variáveis latentes do modelo estrutural:

	Reclamações	Expectativa	Lealdade	Qualidade	Satisfação	Valor
Reclamações	0	0	0.07	0	0	0
Expectativa	0	0	0	0.56	0.06	0.05
Imagem	0	0.50	0.20	0	0.18	0
Qualidade	0	0	0	0	0.51	0.56
Satisfação	0.53	0	0.48	0	0	0
Valor	0	0	0	0	0.19	0

Tabela 4.7: *Path coefficients* do modelo ECSI

Finalmente são calculados os pesos externos dependendo do modo como cada bloco se relaciona com a respetiva variável latente, neste caso, todas as relações são do tipo reflexivo.

	Imagem	Expectativa	Qualidade	Valor	Satisfação	Reclamações	Lealdade
IMAG1	0.21	0	0	0	0	0	0
IMAG2	0.18	0	0	0	0	0	0
IMAG3	0.15	0	0	0	0	0	0
IMAG4	0.23	0	0	0	0	0	0
IMAG5	0.23	0	0	0	0	0	0
CUEX1	0	0.36	0	0	0	0	0
CUEX2	0	0.33	0	0	0	0	0
CUEX3	0	0.31	0	0	0	0	0
PERQ1	0	0	0.16	0	0	0	0
PERQ2	0	0	0.11	0	0	0	0
PERQ3	0	0	0.15	0	0	0	0
PERQ4	0	0	0.14	0	0	0	0
PERQ5	0	0	0.14	0	0	0	0
PERQ6	0	0	0.14	0	0	0	0
PERQ7	0	0	0.16	0	0	0	0
PERV1	0	0	0	0.45	0	0	0
PERV2	0	0	0	0.55	0	0	0
CUSA1	0	0	0	0	0.31	0	0
CUSA2	0	0	0	0	0.10	0	0
CUSA3	0	0	0	0	0.37	0	0
CUSCO	0	0	0	0	0	1.00	0
CUSL1	0	0	0	0	0	0	0.37
CUSL2	0	0	0	0	0	0	0.10
CUSL3	0	0	0	0	0	0	0.53

Tabela 4.8: Pesos externos standardizados do modelo ECSI

Após os resultados provenientes do algoritmo é possível observar a matriz dos coeficientes de correlação entre as variáveis latentes do modelo:

	Reclamações	Expectativa	Imagem	Lealdade	Qualidade	Satisfação	Valor
Reclamações	1	0.26	0.48	0.42	0.53	0.53	0.35
Expectativa	-	1	0.50	0.38	0.56	0.51	0.36
Imagem	-	-	1	0.56	0.75	0.69	0.51
Lealdade	-	-	-	1	0.54	0.66	0.53
Qualidade	-	-	-	-	1	0.79	0.59
Satisfação	-	-	-	-	-	1	0.61
Valor	-	-	-	-	-	-	1

Tabela 4.9: Coeficientes de correlação do modelo ECSI

Para colmatar a análise dos resultados, os seguintes indicadores ajudam a testar a qualidade e coerência do modelo estudado (Tabela 4.10). Adicionalmente, o teste GoF é de 0.467.

	R^2	Comunalidade	Redundância	α de Cronbach	ρ Dillon-Goldstein
Imagem	0.28	0.48	0.13	0.72	0.82
Expectativa	0.25	0.48	0.12	0.45	0.73
Qualidade	0.46	0.58	0.26	0.88	0.90
Valor	0.31	0.85	0.26	0.82	0.92
Satisfalção	0.68	0.69	0.47	0.78	0.87
Reclamações	0.34	1.00	0.34	1.00	1.00
Lealdade	0.00	0.52	0.00	0.47	0.73

Tabela 4.10: Medidas de validade e fiabilidadde do modelo ECSI

4.3 Comparação de Resultados

O processo de comparação dos resultados tem como objetivo validar as estimações produzidas pelo algoritmo utilizando diferentes ferramentas. De forma a ser possível comparar os *outputs* gerados foram testados os mesmos dados e garantidos os mesmos pressupostos. Para além do código desenvolvido em R, os programas utilizados foram os seguintes:

- **SmartPLS** (Ringle et al. 2014): Trata-se de um programa especializado nos modelos que seguem a abordagem PLS. A sua utilização é bastante amigável e intuitiva por apresentar uma interface gráfica bastante avançada que permite ao utilizador especificar o modelo estrutural via *drag & drop*. Os resultados obtidos podem ser disponibilizados em Excel, HTML e Latex, sendo também possível exportar o diagrama que representa o modelo para o formato PNG. Para além dos diversos *outups* (quer tabelas quer gráficos), disponibiliza ainda algumas técnicas de reamostragem.

- **XLSTAT** (Addinsoft 2011): É um suplemento estatístico do programa MS Excel ® que permite trabalhar os dados no mesmo formato e apresenta componentes autónomas para o cálculo de diferentes análises. Este *add-on* é compatível para ambos os sistemas operativos Windows e MacOS. Para efeitos do estudo, integra o módulo XLSAT-PLSPM que prevê a estimação dos modelos PLS-SEM.

- **Pacote** `semPLS` (Monecke e Leisch 2012): Como o próprio nome indica, é um pacote de funções específicas para o modelar equações estruturais

utilizando a abordagem PLS no R[1]. Para além de ser disponibilizado um código aberto e facilmente percetível, oferece métodos que permitem modelar e ajustar os dados, calcular índices de qualidade e obter uma diversidade de gráficos úteis para analisar os resultados. Internamente, este pacote subdivide-se em duas funções centrais: `plsm` e `semplsm`. A primeira é utilizada para criar as especificações do modelo, enquanto a segunda ajusta o modelo criado anteriormente.

Seguindo a ordem do trabalho, foram efetuadas comparações aos resultados produzidos pelo código R e os referidos programas para ambos os modelos experimental e ECSI.

4.3.1 Comparação do Modelo Experimental

Os resultados do modelo experimental foram comparados com os resultados desenvolvidos nos programas XLSTAT e SmartPLS. De forma a tornar possível este exercício, foram garantidas as mesmas condições para os *softwares* utilizados: 1) as relações entre as variáveis latentes Produto e Lealdade são do tipo reflexivo e a para a variável latente Serviço são do tipo formativo; 2) para a estimação dos *path coefficients* foi utilizado o método do centróide; e 3) o critério de convergência foi de 10^{-5}.

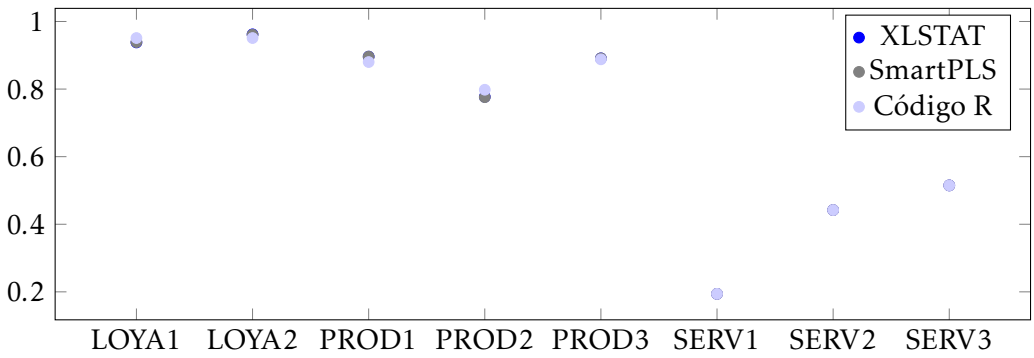

Figura 4.4: *Loadings* do modelo experimental por programa

[1]Informação relativa a este pacote encontra-se disponível em: https://cran.r-project.org/web/packages/semPLS/index.html

Os resultados obtidos na estimação dos *loadings*, visíveis na Figura 4.4, são semelhantes utilizando o XLSTAT e SmartPLS. Para as variáveis manifestas PROD1 e PROD2 notam-se umas ligeiras discrepâncias com os valores obtidos no R em cerca de 0.02 e -0.02, respetivamente.

	Loyalty		
	Código R	XLSTAT	SmartPLS
Produto	0.5022	0.4971	0.4971
Serviço	0.4292	0.4473	0.4473

Tabela 4.11: *Path coefficients* do modelo experimental por programa

No cálculo dos *path coefficients* (Tabela 4.11), os programas XLSTAT e SmartPLS apresentaram os mesmos resultados e diferem do o código R em 0.005 e 0.05 nas variáveis latentes Produto e Serviço.

Quanto aos pesos externos e ao R^2 os *outputs* gerados nos dois programas não verificam diferenças face aos obtidos pelo código R.

Para a correlação entre as variáveis latentes, os resultados dos três programas são ligeiramente diferentes, como se pode observar na seguinte tabela:

	Código R		XLSTAT		SmartPLS	
	Lealdade	Produto	Serviço	Lealdade	Produto	Serviço
Lealdade	1		1		1	
Produto	0.818	1	0.831	1	0.879	1
Serviço	0.799	0.736	0.818	0.746	0.849	0.743

Tabela 4.12: Correlação entre as variáveis latentes por programa

As medidas de qualidade e fiabilidade do modelo resultantes do XLSTAT e SmartPLS, nomeadamente o α de Cronbach e o ρ Dillon-Goldstein, encontram pequenas diferenças, na ordem das centésimas, face aos valores do código R. Já para o GoF, estas diferenças rondam as milésimas.

4.3.2 Comparação do Modelo ECSI

Para comparar os resultados obtidos para o modelo ECSI foram utilizados os programas XLSTAT e o pacote semPLS do R. Estão na base os mesmos dados e as mesmas condições: 1) todas as relações entre as variáveis latentes e a manifestas são do tipo reflexivo; 2) para a estimação dos *path coefficients* foi utilizado o método do centróide; e 3) o critério de convergência foi 10^{-5}.

Começando novamente pelos *loadings*, os resultados obtidos através do XLSTAT e do pacote semPLS são semelhantes e não apresentam diferenças relevantes face aos resultados do código R (encontram-se na ordem dos 10^{-4}).

Da mesma forma, os resultados da estimação dos *path coefficients* no XLSTAT e no pacote semPLS são iguais. As maiores diferenças face ao código R encontra-se nas relações entre as variáveis latentes Lealdade vs Reclamações (-0.00092), Satisfação vs Expectativa (0.0005) e Valor vs Expectativa e Qualidade (-0.0006 e 0.0005, respetivamente).

Em relação aos pesos externos, os valores produzidos pelos mesmos dois programas são idênticos e o maior diferença encontrada face ao R foi de 0.005 entre a variável latente Espectativa e as suas variáveis manifestas.

Quanto aos valores do R^2, apenas se verificam diferenças face aos resultados do código R nas variáveis Qualidade e Satisfação, como se pode observar pela seguinte quadro:

	Código R	XLSTAT	SmartPLS
Expectativa	0.255	0.255	0.255
Qualidade	0.310	0.311	0.311
Valor	0.345	0.345	0.345
Satisfação	0.681	0.680	0.680
Reclamações	0.277	0.277	0.277
Lealdade	0.457	0.457	0.457

Tabela 4.13: R^2 por programa do modelo ECSI

As correlações entre as variáveis latentes resultam nos mesmos valores para o XLSTAT e pacote semPLS, sendo que as maiores discrepâncias face ao código R encontram-se entre as correlações das variáveis latentes Lealdade vs Reclamações (-0.00091), Satisfação vs Lealdade (-0.00056) e Valor vs Expectativa

(-0.00057).

Para as medidas de qualidade e fiabilidade do modelo, o α de Cronbach foi apenas comparado com os resultados do XLSTAT, uma vez que o pacote semPLS não apresenta essa funcionalidade. Os valores encontrados não apresentam diferenças. Quanto ao ρ Dillon-Goldstein, existem apenas algumas dissemelhanças entre o código R e os dados produzidos no pacote semPLS na ordem das milésimas.

Finalmente para os valores do GoF, os resultados são semelhantes entre o código R e o XLSTAT mas diferem em -0.016 com o resultado obtido no semPLS.

Capítulo 5

Conclusão e desenvolvimentos futuros

O trabalho desenvolvido enquadra-se no âmbito da estatística computacional. Como objetivo fundamental do estudo esteve a implementação do *core* do algoritmo PLS-SEM no programa R, em via do progresso da técnica enquanto código aberto.

Sendo uma metodologia bastante utilizada nas disciplinas socioeconómicas, a importância de testar a sua coerência para fenómenos reais é inquestionável. Vivendo no contexto de uma sociedade que dispõe de muita oferta, os índices de qualidade de um produto ou serviço marcam a diferença na procura. Concretamente, assistimos ao papel crescente dos índices de satisfação do cliente que avaliam a posição no mercado das empresas e setores de atividade.

A importância de escrever o algoritmo subjacente ao cálculo destes índices teve por base a simplificação dos procedimentos a partir de uma linguagem acessível. Graças à versatilidade do programa R, este propósito foi alcançado. Contudo, mesmo que adaptado às exigências do utilizador, algumas funcionalidades relacionadas com a compilação do código podem ainda ser melhoradas, nomeadamente:

- criação de uma componente gráfica para visualização dos modelos estrutural e de medida;
- exploração dos dados através de gráficos de análise, em especial para os *outputs* das técnicas de reamostragem;
- otimização do método *bootstrap* enquanto medida de validação do modelo e desenvolvimento de outras técnicas de reamostragem (*jack-knife* ou *blindfolding*);

Segundo a análise comparativa dos resultados, o código desenvolvido apresenta diferenças insignificantes face aos outros programas testados.

Em busca do aperfeiçoamento do método para melhor corresponder à realidade, alguns trabalhos têm vindo a ser desenvolvidos nesta área. Concretamente, observa-se muitas vezes que as relações entre as variáveis latentes não são lineares. Por exemplo, analisando o comportamento dos consumidores, a relação entre a satisfação e a lealdade não é suposto ser linear. Neste caso é então sugerido a utilização de uma função definida por ramos para descrever a não linearidade. O trabalho desenvolvido por Jakobowicz (2007) prevê a inclusão das relações não lineares do modelo estrutural do algoritmo PLS-SEM utilizando métodos de otimização da escala (*optimal scaling methods*). Neste caso, para obter transformações das variáveis latentes são aplicados *B-splines* [1].

Assim, o seguinte trabalho disponibiliza a toda a comunidade científica o seu contributo, de forma aberta e livre, para potenciais desenvolvimentos nesta matéria.

[1] É uma técnica de aproximação que consiste na divisão de um intervalo de interesse em vários subintervalos para interpolar polinóminos de menor dimensão, da forma mais suave possível.

Bibliografia

Addinsoft (2011). "XLSTAT - Statistics Package for Excel". URL: https://www.xlstat.com.

Bollen, K. A. (1984). "Note Multiple Indicators : Internal Consistency or No Necessary Relationship?" *Elsevier* 18, pp. 377–385.

Chin, W. W. (2001). "PLS - Graph User's Guide". *Soft Modeling Inc.* Version 3.0.

Costigliola, F. (2009). "Partial Least Square – Path Modeling: metodologia, software e aplicação". URL: http://run.unl.pt///handle/10362/8818.

Fornell, C. (1985). "A second Generation of Multivariate Analysis: Classification of Methods and Implications for Marketing Research". 414ª sér.

Fornell, C., M. D. Johnson, E. W. Anderson, J. Cha e B. E. Bryant (1996). "The American customer satisfaction index: nature, purpose, and findings". *JSTOR* 60, pp. 7–18.

Friedrich, D., K. Jöreskog e H. Wold (1984). "Systems Under Indirect Observation. Causality - Structure - Prediction. 2 Bde. Part I. Part II". *JSTOR*.

Guarino, A. J. (2004). "A Comparison of First and Second Generation Multivariate Analyses: Canonical Correlation Analysis and Structural Equation Modeling". *Florida Journal* 42, pp. 22–40.

Hair, J. F., M. Sarstedt, T. M. Pieper e C. M. Ringle (2012). "The Use of Partial Least Squares Structural Equation Modeling in Strategic Management Research : A Review of Past Practices and Recommendations for Future Applications". *Elsevier* 45, pp. 320–340. ISSN: 0024-6301.

Hair, J. F., C. M. Ringle e M. Sarstedt (2013). "Editorial Partial Least Squares: The Better Approach to Structural Equation Modeling?" *Elsevier* 45, pp. 312–319. DOI: 10.1016/j.lrp.2012.09.011.

Hair Jr, J. F., G. T. Hult, C. M. Ringle e M. Sarstedt (2013). "A primer on partial least squares structural equation modeling (PLS-SEM)". *Sage Publications*.

Henseler, J., C. M. Ringle e M. Sarstedt (2012). "Using Partial Least Squares Path Modeling in International Advertising Research : Basic Concepts and Recent Issues". *Partial least squares path modeling in advertising research*, pp. 252–270. DOI: 10.4337/9781848448582.00023.

Jakobowicz, E. (2007). "Latent variable transformation using monotonic B-splines in PLS path modeling".

Johnson, M. D., A. Gustafsson, T. W. Andreassen, L. Lervik e J. Cha (2001). "The evolution and future of national customer satisfaction index models". *Elsevier* 22.2, pp. 217–245.

Jöreskog, K. G. (1970). "A general method for analysis of covariance structures". *Biometrika Trust* 57.2, pp. 239–251.

Jöreskog, K. G. e D. Sorbom (1993). "LISREL 8: Structural equation modeling with the SIMPLIS command language". *Scientific Software International*.

Lohmöller, J.-B. (1989). "Latent Variable Path Modeling with Partial Least Squares". *Physica-Verlag*.

Monecke, A. e F. Leisch (2012). "semPLS : Structural Equation Modeling Using Partial Least Squares". *Journal of Statistical Software* 48.3, pp. 1–32. ISSN: 1548-7660. URL: http://www.jstatsoft.org/v48/i03/.

Neyman, J., E. Fix e F. N. David (1966). "Research papers in statistics : festschrift for J. Neyman". *John Wiley & Sons*.

Noonan, R. e H. Wold (1977). "NIPALS Path Modelling with Latent Variables". *Scandinavian Journal of Educational Research* 21, pp. 33–61. URL: http://dx.doi.org/10.1080/0031383770210103.

O'Loughlin, C., G. Coenders et al. (2002). "Application of the European Customer Satisfaction Index to Postal Services. Structural Equation Models versus Partial Least Squares".

R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: http://www.R-project.org/.

Ringle, C. M., S. Wende e J.-M. Becker (2014). "SmartPLS 3". *Hamburg: SmartPLS*. URL: http://www.smartpls.com.

Sanchez, G. (2013). "PLS Path Modeling with R". *Trowchez Editions*. DOI: http://gastonsanchez.com/PLS_Path_Modeling_with_R.pdf.

Sanchez, G., L. Trinchera e G. Russolillo (2015). *plspm: Tools for Partial Least Squares Path Modeling (PLS-PM)*. R package version 0.4.7. URL: http://CRAN.R-project.org/package=plspm.

Soares, A., A. Vaz, P. S. Coelho e S. P. Esteves (2008). "Aplicação do European Customer Satisfaction Index (ECSI) ao sector das águas". *Revista Lusófona*

de Humanidades e Tecnologias, pp. 69–74. URL: http://hdl.handle.net/10437/2662.

Temme, D., H. Kreis e L. Hildebrandt (2010). "A comparison of current PLS path modeling software: Features, ease-of-use, and performance". *Handbook of Partial Least Squares*, pp. 737–756.

Tenenhaus, M., V. E. Vinzi, Y.-M. Chatelinc e C. Lauro (2005). "PLS path modelling PLS path modeling". *Elsevier*. DOI: 10.1016/j.csda.2004.03.005.

Trinchera, L. (2007). "Unobserved Heterogeneity in Structural Equation Models: a new approach to latent class detection in PLS Path Modeling". *Università degli Studi di Napoli Federico II*, p. 338.

Vinzi, V. E., L. Trinchera e S. Amato (2010). "PLS path modeling: from foundations to recent developments and open issues for model assessment and improvement". *Handbook of partial least squares*. Springer.

Wold, H. (1973). "Nonlinear Iterative Partial Least Squares (NIPALS) modelling: Some current developments". *Academic Press*, pp. 383–407.

— (1974). "Causal flows with latent variables". *European Economic Review* 5, pp. 67–86. ISSN: 0014-2921.

— (1980). "Model Construction and Evaluation When Theoretical Knowledge is Scarce". *National Bureau of Economic Research*, pp. 47–74. URL: http://www.nber.org/chapters/c11693.

Wold, H. e E Lyttkens (1969). "Nonlinear Iterative Partial Least Squares (NIPALS) estimation procedures". *Bulletin of the International Statistical Institute* 43.1.

APÊNDICE A

ANEXO

Inquérito sobre satisfação dos clientes com operadores da rede móvel
Versão Original

Latent variables	**Manifest variables**	**Description**
Image	IMAG1	It can be trusted in what it says and does
	IMAG2	It is stable and firmly established
	IMAG3	It has a social contribution for the society
	IMAG4	It is concerned with customers
	IMAG5	It is innovative and forward looking
Expectation	CUEX1	Expectations for the overall quality of "your mobile phone provider" at the moment you became customer of this provider
	CUEX2	Expectations for "your mobile phone provider" to provide products and services to meet your personal need
	CUEX3	How often did you expect that things could go wrong at "your mobile phone provider"

Quality	PERQ1	Overall perceived quality
	PERQ2	Technical quality of the network
	PERQ3	Customer service and personal advice offered
	PERQ4	Quality of the services you use
	PERQ5	Range of services and products offered
	PERQ6	Reliability and accuracy of the products and services provided
	PERQ7	Clarity and transparency of information provided
Value	PERV1	Given the quality of the products and services offered by "your mobile phone provider" how would you rate the fees and prices that you pay for them?
	PERV2	Given the fees and prices that you pay for "your mobile phone provider" how would you rate the quality of the products and services offered by "your mobile phone provider"?
Satisfaction	CUSA1	Overall satisfaction
	CUSA2	Fulfillment of expectations
	CUSA3	How well do you think "your mobile phone provider" compares with your ideal mobile phone provider?
Complaints	CUSCO	You complained about "your mobile phone provider" last year. How well, or poorly, was your most recent complaint handled. or You did not complain about "your mobile phone provider" last year. Imagine you have to complain to "your mobile phone provider" because of a bad quality of service or product. To what extent do you think that "your mobile phone provider" will care about your complaint?

Loyalty	CUSL1	If you would need to choose a new mobile phone provider how likely is it that you would choose “your provider” again?
	CUSL2	Let us now suppose that other mobile phone providers decide to lowertheirfees and prices, but “yourmobile phone provider” stays at the same level as today. At which level of diQerence (in %) would you choose another mobile phone provider?
	CUSL3	If a friend orcolleague asks you foradvice, how likely is it that you would recommend “your mobile phone provider”?

APÊNDICE B

ANEXO

```
###   ALGORITMO PLS-SEM   ###

## Bibliotecas a importar
library(semPLS)
library(plspm)     # for alpha and rho calculations
library(utils)     # for the function is.Zero()
library(abind)     # for arrays

## Definir directorio base
ines <- "D:/TESE/codigo/FINAL/"
base_dir <- ines

modelo = 'experimental' # 'escsi'

if (modelo == 'experimental') {
file_dados = 'modelo_experimental/teste.txt'
file_SM = 'modelo_experimental/teste_sm.txt'
file_MM = 'modelo_experimental/teste_mm.txt'
file_aux = 'modelo_experimental/teste_mm_aux.txt'
} else {
file_dados = 'modelo_ECSI/MobiData.txt'
file_SM = 'modelo_ECSI/ECSI_sm.txt'
```

```
23  file_MM = 'modelo_ECSI/ECSI_mm.txt'
24  file_aux = 'modelo_ECSI/ECSI_mm_aux.txt' }
25
26  ## Carregamento dos ficheiros de importacao
27  # dados
28  dados <- read.table(paste(base_dir, file_dados, sep=''), header=T)
29
30  # modelo estrutural
31  SMteste <- read.table(paste(base_dir, file_SM, sep=''), header=T)
32
33  # modelo de medida
34  MMteste <- read.table(paste(base_dir, file_MM, sep=''), header=T,
        stringsAsFactors=F)
35
36  # matriz auxiliar que  indica a direccao das relacoes (reflexivo ou
        formativo)
37  AUXteste <- read.table(paste(base_dir, file_aux, sep=''), header=T)
38
39  # funcao que define o algoritmo
40  metodoPLS = function (dados, SM, MM, AUX, print_iterations = FALSE,
        print_results=TRUE) {
41
42  ## Informacao sobre o modelo
43  latente <- unique(MMteste['source']) # variaveis latentes (VL)
44  manifesta <- t(MMteste['target'])    # variaveis manifestas (VM)
45  matrizA <- table(AUXteste) # matriz das relacoes entre VL e VM
46  matrizD <- table(SMteste)  # matriz das relacoes entre VL
47
48  # matriz inicial dos pesos = 1
49  matrizM <- t(table(MMteste))
50
51  ### FUNCOES AUXILIARES ###
52  # variancia
53  varpop <- function (x) {
54  n <- length(x)
55  var(x) * (n-1)/n }
56
```

```
57  # desvio padrao
58  sdpop <- function (x) {
59  sqrt(varpop(x)) }
60
61  # normalizacao
62  normalizepop <- function(data) {
63  dp = apply(data, 2, sdpop)
64  scale(data, scale=dp) }
65
66  # multiplicacao da matriz com os dados e a matriz dos pesos
67  rowProd <- function(row, matr) {
68  rnames <- rownames(matr)
69  rowSums(sapply(rnames, function(i) row[i] * matr[i, ])) }
70
71  matrixProd <- function(data, weights) {
72  t(apply(data, 1, rowProd, weights))  }
73
74  # imprimir variaveis
75  printVar <- function(var) {
76  cat(deparse(substitute(var)), '=', var, '\n') }
77
78  # imprimir tabelas com pontos em vez dos zeros
79  printTable <- function(t, na.print='.') {
80  t <- replace(t, isZero(t), NA)
81  print.table(t, na.print=na.print, digits=3)  }
82
83  ######################
84  ##### ALGORITMO #####
85  ######################
86
87  ##### STAGE 1 #####
88  # normalizar os dados
89  dadostand <- normalizepop(dados)
90
91  ###  STEP 1.1 - Inicializacao dos pesos  ###
92  matriz0 <- matrizM    # matriz dos pesos =1
93
```

```
94  # inicializacao da matriz de correlacoes
95  correlacoes <- matrizD
96  source_names = rownames(matrizD)
97  target_names = colnames(matrizD)
98
99  ###############  ITERACAO  ###############
100 tolerance <- 10^(-7)
101 sumdif <- 1
102 iteration <- 0
103
104 while (sumdif > tolerance) {
105 iteration <- iteration + 1
106
107 ###  STEP 1.2 - Aproximacao interna  ###
108 # dados nao-normalizados
109 step1 <- matrixProd(dadostand, matriz0)
110
111 # dados normalizados
112 step1st <- normalizepop(step1)
113
114 # Centroid weigting scheme (correlacao de pearson)
115 for (i in 1:nrow(matrizD)) {
116 for (j in 1:ncol(matrizD)) {
117 if (matrizD[i, j] == 1) {
118 correlacoes[i, j] <- cor(step1st[, source_names[i]], step1st[,
      target_names[j]]) } } }
119
120 ###  STEP 1.3 - Aproximacao externa  ###
121 step3 <- step1 * 0
122
123 for (i in source_names) {
124 for (j in target_names) {
125 step3[, i] <- step3[, i] + step1st[, j] * correlacoes[i, j]
126 step3[, j] <-  step3[, j] + step1st[, i] * correlacoes[i, j]  } }
127
128 # normalizar step3
129 step3st <- normalizepop(step3)
```

```
130
131 ###  STEP 1.4 - Calculo dos factor scores  ###
132 # calculo auxiliar -> nao-normalizados
133 step4 <- rep(0, length(manifesta))
134 names(step4) <- manifesta
135 matrizaux <- step3 * 0
136
137 for (target in unique(AUXteste[, 'target'])) {
138 if (target %in% manifesta) {
139 lat <- MMteste[which(MMteste[,'target'] == target), 'source']
140 step4[target] <- cor(dadostand[, target], step3st[, lat])
141 matrizaux[, lat] <- matrizaux[, lat] + step4[target] * dadostand[,
    target]
142 } else {
143 sources <- which(AUXteste[, 'target'] == target)
144 y = step3st[, target]
145 x = dadostand[, sources]
146 ols <- lm(y~x)
147 step4[sources] <- ols$coefficients[-1]
148 matrizaux[, target] <- matrizaux[, target] + colSums(step4[sources]
     * t(dadostand[, sources])) } }
149
150 # calculo auxiliar -> normalizados
151 despad4 <- apply(matrizaux, 2, sdpop)
152
153 ## Calculo dos pesos externos
154 outerweigths <- c()
155 matrizR <- matriz0
156 sumdif <- 0
157 for (manif in manifesta) {
158 lat <- MMteste[which(MMteste[, 'target'] == manif), 'source']
159 outerweigths[manif] <- step4[manif] / despad4[as.character(lat)]
160 matrizR[manif, lat] <- outerweigths[manif]
161 sumdif <- sumdif + abs(matriz0[manif, lat] - matrizR[manif, lat]) }
162
163 matriz0 <- matrizR
164
```

```
# factor scores
FS <- matrix(nrow=nrow(dadostand), ncol=nrow(latente))
colnames(FS) <- colnames(matrizR)

for(i in 1:ncol(matrizR)){
manif <- names(which(matrizM[,i]==1))
calculo <-  outerweigths[manif] %*% t(dadostand[,manif])
FS [,i] <- calculo }

factor_score <- normalizepop(FS)

### output
if (print_iterations) {
printVar(iteration)
printVar(outerweigths)
printVar(sumdif) } }

##### STAGE 2 #####
sumdif < tolerance

## Calculo dos loadings
# para o caso de ser unicamente reflexivo
ref <- cor(dadostand,step1st)
reflexivo <- ref[order(row.names(ref)), ]
loadings <- ifelse(matrizM, reflexivo,0)

# para o caso de ser formativo
for (i in unique(AUXteste[, 'source'])) {
if (i %in% manifesta) {
lat <- MMteste[which(MMteste['target'] == i), 'source']
manif <- which(rownames(loadings)== i)
sources <- which(AUXteste[, 'source'] == i)
formativo <- outerweigths [sources]
loadings [manif,which(colnames(loadings) == lat)] <- formativo }  }

## Correlacoes entre as variaveis latentes
LVs_cor <- cor(step1st,step1st)
```

```
202 LVs_cor[!upper.tri(LVs_cor,diag=TRUE)] <- NA
203
204 ## Estimacao dos path coefficients por OLS
205 path = matrizD*0
206 matrizR2 <- matrix(0,nrow=nrow(latente))
207 rownames(matrizR2) <- unlist(latente)
208 colnames(matrizR2) <- "R2"
209
210 for (i in 1:ncol(matrizD)) {
211 y = step1st[, target_names[i]]
212 x = step1st[, names(which(matrizD[,i]==1))]
213 ols <- lm(y~x)
214 R2 <- summary(ols)$r.squared
215 coeff <- round(summary(ols)$coefficients[,1])
216 teste <- coef(ols)
217 path[names(which(matrizD[,i]==1)), target_names[i]] <- ols$
      coefficients[2:length(ols$coefficients)]
218 matrizR2[i,1] <- R2  }
219
220 #### FINAL OUTPUT ####
221 if (print_results) {
222
223 factor_score   # scores das variaveis latentes
224 printTable(loadings[manifesta, unlist(latente)])   # loadings
225 printTable(path)   # path coefficients
226 printTable(LVs_cor)   # latent variables correlations
227 printTable(matrizR[manifesta, unlist(latente)])   # outer wheights
      }
228
229 return (list(loadings=loadings, path=path, LVs_cor=LVs_cor, matrizR
    =matrizR))
230
231 ######### TESTES DE VALIDACAO E CONSISTENCIA #########
232 # Communalidade
233 communality <- matrix(0, ncol=1, nrow=nrow(latente))
234 colnames(communality) <- "Communality"
235 rownames(communality) <- unlist(latente)
```

```
236
237 ## Alpha de Cronbach e Rho Dillon-Goldstein
238 testes <- matrix(0, ncol=2, nrow=nrow(latente))
239 colnames(testes) <- list("Alpha", "Rho")
240 rownames(testes) <- unlist(latente)
241
242 for(i in 1:ncol(matrizM)){
243 nvar <- sum(matrizM[,i])
244 x <- which(colnames(matrizM)[i]==latente)
245 com <- sum(cor(dadostand[,names(which(matrizM[,i]==1))],step1st[,i
    ])^2)/nvar
246 communality[x,1] <- com
247 a <- alpha(dadostand[,names(which(matrizM[,i]==1))])
248 r <- rho(dadostand[,names(which(matrizM[,i]==1))])
249 testes[x,1] <- round(a, digits=3)
250 testes[x,2] <- round(r, digits=3) }
251
252 ## Redundancia
253 validacao <- cbind(matrizR2,communality,0)
254 colnames(validacao)[3] <- "Redundancy"
255
256 for(i in 1:nrow(validacao)){
257 red <- validacao[i,1]*validacao[i,2]
258 validacao[i,3] <- red }
259
260 ## GoF
261 GoF <- sqrt((mean(validacao[,1])) %*% mean(validacao[,2]))
262 colnames(GoF) <- "GoF"
263
264 ### RESULTADOS ###
265
266 validacao      # R2 + communality + redundancy
267 testes         # alpha and rho
268 GoF
269
270 }
271
```

```
272 r = metodoPLS(dados, SMteste, MMteste, AUXteste, print_results = T)
273
274 ######################
275 ####  BOOTSTRAP  ####
276 ######################
277
278 amostra <- function (d) {
279 return (apply(d, 2, function (x) sample(x, rep=T))) }
280
281 n_rep = 100
282 arrLoadings <- c()
283 arrPath <- c()
284 arrMatrizR <- c()
285
286 for (i in 1:n_rep) {
287 d <- amostra(dados)
288 pls <- metodoPLS(d, SMteste, MMteste, AUXteste, print_results =
      FALSE)
289 arrLoadings <- abind(arrLoadings, pls$loadings, along=3)
290 arrPath <- abind(arrPath, pls$path, along=3)
291 arrMatrizR <- abind(arrMatrizR, pls$matrizR, along=3) }
292
293 # calcular as medias e desvio padrao por elementos da matriz
294 arrayMeans <- function (arr) {
295 apply(arr, c(1, 2), mean) }
296
297 arraySDs <- function (arr) {
298 apply(arr, c(1, 2), sd)  }
299
300 arrayMeans(arrLoadings)
301 arrayMeans(arrPath)
302 arrayMeans(arrMatrizR)
303
304 arraySDs(arrLoadings)
305 arraySDs(arrPath)
306 arraySDs(arrMatrizR)
```

Listagem B.1: Código desenvolvido