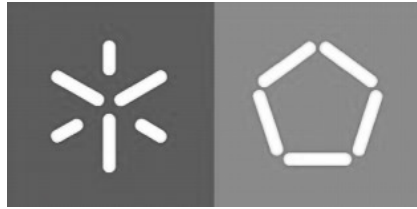


Universidade do Minho
Escola de Engenharia

Paulo Sérgio de Almeida Festa

**Detecção e validação de comportamento
desviante no combate à fraude em
modelos de publicidade *pay-per-click***

Outubro de 2012



Universidade do Minho
Escola de Engenharia

Paulo Sérgio de Almeida Festa

**Detecção e validação de comportamento
desviante no combate à fraude em
modelos de publicidade *pay-per-click***

Dissertação de Mestrado
Mestrado em Engenharia Informática

Trabalho efetuado sob a orientação do
Professor Doutor Paulo Jorge de Sousa Azevedo

Outubro de 2012



DECLARAÇÃO

Nome: Paulo Sérgio de Almeida Festa

Endereço electrónico: paulo.s.festa@gmail.com

Telefone: 913482835

Número do Bilhete de Identidade: 12970860

Título dissertação: Detecção e validação de comportamento desviante no combate à fraude em
modelos de publicidade *pay-per-click*

Orientador(es): Professor Doutor Paulo Jorge de Sousa Azevedo

Ano de conclusão: 2012

Designação do Mestrado: Mestrado Engenharia Informática

É AUTORIZADA A REPRODUÇÃO PARCIAL DESTA TESE/TRABALHO, APENAS PARA EFEITOS DE
INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE
COMPROMETE;

Universidade do Minho, 12 / 12 /2012

Paulo Sérgio de Almeida Festa



Agradecimentos

A presente dissertação representa o culminar de um longo percurso académico e é a demonstração do conhecimento e da experiência adquirida durante uma vida. A sua realização não é um mérito exclusivamente pessoal. É antes o resultado de conselhos, exemplos e orientações de pessoas e instituições que de forma directa ou indirecta contribuíram para o desfecho desta dissertação.

Quero, como tal, agradecer em primeiro lugar ao Departamento de Informática da Universidade do Minho e a todo o seu corpo docente pelos conhecimentos de excelência que me transmitiram no decorrer da licenciatura e do mestrado em engenharia informática.

Agradeço ao professor Paulo Azevedo a confiança demonstrada, as críticas e sugestões partilhadas, a compreensão pelas minhas questões, a disponibilidade, a competência científica e, principalmente, os valores transmitidos. A sua orientação foi indispensável para o desenvolvimento desta dissertação.

Agradeço, igualmente, à AdClip Portugal pela disponibilização de dados que nos permitiu enquadrar a problemática em questão e conceber a solução apresentada.

Aos meus pais expreso a minha gratidão pelos princípios transmitidos e pelo suporte que me concederam, fundamental para alcançar esta fase. Aos amigos, em geral, por me terem acompanhado nesta longa jornada, dando-me apoio e confiança.

Agradeço, em particular, ao Tiago Mendes pelas intermináveis e cordiais discussões que mantivemos, pelo espírito de camaradagem, pela presença nos momentos mais delicados e, acima de tudo, pela sua honestidade e generosidade.

Por último, reservo um agradecimento especial à pessoa que me encorajou nos momentos capitais do meu percurso académico. Com quem partilhei as minhas dúvidas de modo a alcançar as melhores decisões. Com quem partilhei as minhas derrotas de modo a obter ânimo para novas vitórias. Com quem partilhei os melhores e inesquecíveis momentos durante todo o percurso universitário. Na sua companhia, esta caminhada tornou-se muitíssimo mais agradável e gratificante. Obrigado Bernardete Martins.



Resumo

O modelo *pay-per-click* apresenta-se como um dos principais modelos de negócio que acautela a sustentabilidade financeira de vários serviços *online* (e.g. motores de busca e redes sociais) através dos proveitos gerados pela impressão de publicidade. Assim, os anunciantes interessados em divulgar e lucrar com os seus produtos ou serviços pagam comissões dos seus anúncios aos editores (e.g. *sites*) que os difundam. O montante pago pelo anunciante é calculado com base no número de cliques realizado nos anúncios. Infelizmente, a veracidade deste modelo é colocado frequentemente em causa pela existência de um conjunto de incentivos desonestos (i.e. situações de fraude) que favorecem parte dos intervenientes.

Nesta dissertação é proposta uma solução para detecção e validação em tempo real, sem intervenção humana, de situações fraudulentas que envolvam a inflação de cliques em anúncios classificados. Com principal foco na postura do utilizador, são definidas variáveis numéricas para caracterizar o seu comportamento geral. São, igualmente, derivadas regras de associação que permitam detectar padrões de utilização e estimadas as distribuições de valores de cada uma das variáveis consideradas. As suspeitas de fraude surgem se o valor dessas variáveis, para um dado utilizador, se desviar de forma significativa dos valores esperados (i.e. obtidos pelos restantes utilizadores).

A proposta para validação destas suspeitas baseia-se na criação e implementação de cenários que visam dificultar, propositadamente, o comportamento suspeito até aí demonstrado pelo utilizador. Se mesmo nestas circunstâncias o comportamento se mantiver inalterado, as suspeitas são consideradas fundamentadas, o utilizador considerado fraudulento e os seus cliques invalidados.

O método proposto foi testado através de um protótipo desenvolvido especialmente para esta problemática. Os resultados permitem concluir que a combinação de padrões de utilização obtidos por técnicas de mineração de dados e uma análise estatística criteriosa são uma mais valia para identificar situações suspeitas em modelos *pay-per-click*. Evidencia, igualmente, que as situações suspeitas podem ser fundamentadas com base no comportamento e postura do utilizador perante os cenários adversos que lhe são colocados.

Palavras-chave: Análise estatística, Comportamento desviante, Detecção de fraude, Mineração de dados, Pagamento por clique, Publicidade;



Abstract

The pay-per-click advertising model appears as one of the main business models that guarantees the financial sustainability of several online services (e.g. search engines and social networks) through the profits generated by advertising impression. For that reason, the advertisers interested in disclosing and profit from their own products or services pay commissions of their own advertisements to the publishers that disseminate them (e.g. sites). The amount paid by each advertiser is calculated in regard to the number of mouse clicks done in each advertisement. Unfortunately, the integrity of this model is usually questioned by the existence of a set of dishonest incentives (known as fraudulent situations) that favour part of the stakeholders.

In this dissertation, a solution is proposed to detect and validate in real time, with no human intervention, fraudulent situations that involve the inflation of mouse clicks in classified ads. The main focus relies on the user attitude by defining a number of numerical variables that describe his general behaviour. The fraud suspicions arise if the value of those variables for a given user deviates significantly from the expected values (i.e. obtained by the remaining users).

The validation of these suspicions is based on the creation and development of scenarios that want to hamper the suspected behaviours revealed so far. If, even under these circumstances, the behaviour remains unchanged, then those suspicions are deemed as legit, considering that same user as fraudulent and his own clicks are therefore invalidated.

The proposal method as been tested through a prototype specially developed for this purpose. The results allow to conclude that the pattern combination obtained by data mining techniques and a careful statistical analysis are indeed accurate and able to identify the suspicious situations in pay-per-click advertising model. It also shows that the suspicious situations can be grounded in regard to the user behaviour and posture for the adverse scenarios that he have to face with the objective of detecting frauds.

Keywords: Advertising, Data Mining, Fraud detection, Pay-per-click, Outlier detection, Statistical analysis;



“A diferença entre o possível e o impossível
está na vontade humana”

- Louis Pasteur -



Índice

1. Introdução	1
1.1. Expansão Sustentável da Internet	2
1.2. Evolução do Modelo PPC (<i>Pay-Per-Click</i>)	4
1.2.1. Implementação adoptada pela <i>Google</i>	9
1.2.2. Implementação adoptada pela <i>AdClip</i>	12
1.3. Sistemas Fraudulentos no modelo PPC	12
1.4. Motivações	14
1.5. Objectivos	15
1.6. Organização do documento	17
2. Revisão Bibliográfica	18
2.1. <i>Web Mining</i>	19
2.1.1. Identificação do utilizador	22
2.1.2. Perfis de utilizador	24
2.1.3. Padrões de navegação	29
2.2. Detecção de fraude	34
2.2.1. Fraude de Tipo I	34
2.2.2. Fraude de Tipo II	37
2.2.3. Fraude de Tipo III	47
2.3. Contributo para a Solução Desenvolvida	49
3. Concepção da Solução	53
3.1. Perspectiva Geral	54
3.2. Variáveis de análise	58
3.2.1. Variável de análise $V_{NAV}(x, y)$	61
3.2.2. Variável de análise $V_{REL}(x, y)$	65
3.3. Extracção de dados	67



3.4.	Obtenção de regras de associação.....	68
3.5.	Estimativa das distribuições de pontuações.....	71
3.6.	Cálculo de <i>p-values</i>	76
3.7.	Armadilhas.....	77
4.	Resultados	81
4.1.	Protótipo desenvolvido.....	82
4.2.	Visualizações Duplicadas.....	85
4.3.	Visualização aleatória e em quantidade moderada	87
4.4.	Visualização não aleatória e em quantidade elevada.....	88
5.	Conclusões	90
5.1.	Discussão	91
5.1.1.	Limitações	92
5.1.2.	Contribuições.....	94
5.2.	Trabalho Futuro.....	95
Bibliografia	96
Anexos	106
A.1	- Exemplo de regras de associação obtidas pelo CAREN.....	107
A.2	- Exemplo de matrizes de transição	108
A.3	- Exemplo do <i>script</i> R utilizado para a estimativa das distribuições de pontuações	109
A.4	- Estimativa das distribuições de pontuações (Tabela sumária).....	110
A.5	- Estimativa das distribuições de pontuações (Histogramas e CDFs).....	111
A.6	- Comparação visual entre o <i>site</i> modelo e o <i>site</i> AdClip.....	116



Lista de Tabelas

Tabela 2.1 - Contribuição individual de cada autor para a solução desenvolvida.....	50
Tabela 3.1 – Variáveis de análise idealizadas para a detecção e prevenção de fraude.....	59
Tabela 3.2 - Cálculo da V_{NAV} sem incorporação de teste binomial	62
Tabela 3.3 - Cálculo da V_{NAV} com incorporação de teste binomial	65
Tabela 3.4 - Cenários para cálculo da V_{REL}	66
Tabela 3.5 – Resultados obtidos para a estimativa da distribuição de pontuações de V_{NAV} (automóveis, distrito)	75
Tabela 3.6 – Descrição das armadilhas idealizadas e do comportamento esperado	78
Tabela T.1 - Tabela resultante da documentação produzida pela solução.....	110



Lista de Figuras

Figura 1.1 – Impacto da internet na população mundial (ITU - International Telecommunication Union, 2011).....	2
Figura 1.2 – Primeiro anúncio patrocinado disponibilizado na internet.....	5
Figura 1.3 – Sequência de actividades para o modelo PPC de três e quatro intervenientes	7
Figura 1.4 – Evolução (esquerda) e distribuição (direita) de lucros gerados pelos modelos de publicidade <i>online</i>	8
Figura 1.5 – Áreas de negócio (esquerda) e tipo de pagamento (direita) com mais influência nos modelos de publicidade <i>online</i>	8
Figura 1.6 – Distribuição de lucros por entre as 50 empresas que lideram o mercado publicitário.	9
Figura 1.7 – Método de débito aplicado pela <i>Google</i> aos seus anunciantes.....	10
Figura 1.8 – Anúncios impressos através do <i>AdWords</i> (modelo PPC de 3 intervenientes).....	10
Figura 1.9 - Anúncios impressos num editor através do <i>AdSense</i> (modelo PPC de 4 intervenientes).....	11
Figura 2.1 - Etapas do processo de extração de conhecimento numa base de dados (Fayyad <i>et al.</i> , 1996).....	20
Figura 2.2 - Representação de perfis de utilizador usando vectores de palavras-chave (Gauch <i>et al.</i> 2007).....	26
Figura 2.3 - Representação de perfis de utilizador usando redes semânticas de conceitos (Gentili <i>et al.</i> , 2003).....	28
Figura 2.4 - Representação de perfis de utilizador usando hierarquias de conceitos (Sieg, Mobasher, & Burke, 2010)	29
Figura 2.5 - Representação de perfis de navegação usando cadeias de <i>Markov</i> (Sadagopan & Li, 2008)	31
Figura 2.6 - Representação de perfis de navegação usando cadeias de <i>markov</i> de ordem <i>N</i> (Borges & Levene, 2008).....	32
Figura 2.7 - Representação de dados por distribuição Gaussiana e por EPD (Böhm <i>et al.</i> , 2009)	33
Figura 2.8 - Obtenção do número de utilizadores reencaminhados com base nos métodos <i>redirect (HTML)</i> e <i>onload (Javascript)</i> (Reiter, Anupam, & Mayer, 1998)	35



Figura 2.9 - Obtenção do número de utilizadores reencaminhados com base em cadeias de <i>hash</i> (Blundo & Cimato, 2002).....	36
Figura 2.10 - Distribuição <i>Zipf</i> : Frequência de cliques perante a complexidade do ataque (Tuzhilin, 2006).....	37
Figura 2.11 - Solução baseada em cupões que atestam comportamento positivo dos utilizadores (Juels <i>et al.</i> , 2007)	39
Figura 2.12 - Uso de histogramas para medir o desvio para o comportamento actual (direita) e do comportamento esperado (esquerda) (Kantardzic <i>et al.</i> , 2009).....	40
Figura 2.13 - Representação da frequência absoluta de um atributo-valor e do limite máximo aceitável (Kantardzic <i>et al.</i> , 2009).....	41
Figura 2.14 - Proposta de Walgampaya, Kantardzic e Yamolskiy (2010) para detecção e prevenção de fraude em tempo real.....	43
Figura 2.15 - Estrutura <i>Bloom Filter</i> e respectivo preenchimento (Knuth, 1998).....	44
Figura 2.16 - O processo de detecção de cliques inválidos implementado pela <i>Google</i> (Tuzhilin, 2006)	46
Figura 3.1 – Visão global da arquitectura da solução proposta	57
Figura 3.2 – Informação resultante do processo de actualização de dados	58
Figura 3.3 – Esquerda: Número de anúncios visualizados por cliente (inicia em 5 cliques); Centro: Número de categorias visualizadas por cliente; Direita: Número de duplicados por cliente (<i>Histogramas referentes aos dados da AdClip</i>)	60
Figura 3.4 - Representação gráfica do teste binomial para o cálculo da V_{NAV}	65
Figura 3.5 - Representação gráfica do cálculo da V_{REL}	67
Figura 3.6 – Formato do ficheiro <i>distrito.basket</i> (esquerda) e formato do ficheiro <i>ctr.scores</i> (direita)	68
Figura 3.7 – Regras de associação derivadas pelo CAREN para os distritos visualizados	70
Figura 3.8 – Representação parcial da matriz de transição produzida para V_{NAV} (automóveis, marca)	70
Figura 3.9 – Distribuições obtidas pela variação dos parâmetros da distribuição beta	72
Figura 3.10 – Estimativa da distribuição de pontuações de V_{NAV} (automóveis, distrito).....	75
Figura 3.11 – Função de densidade de probabilidade dos dados reais e da estimativa de V_{NAV} (automóveis, distrito)	76
Figura 4.1 – <i>Site</i> modelo concebido para testar e validar a proposta apresentada.....	82



Figura 4.2 - Informação detalhada sobre as variáveis de análise de um utilizador suspeito	84
Figura 4.3 – Modo de operar da Arm_{VID} perante um cenário suspeito.....	86
Figura 4.4 – Modo de operar da Arm_{NAV} perante um cenário suspeito.....	87
Figura 4.5 – Variáveis de análise de um utilizador com navegação criteriosa	88
Figura 4.6 - Modo de operar da Arm_{CTR} perante um cenário suspeito	89
Figura A.1 - Extracto parcial das regras de associação derivadas para o distrito dos anúncios visualizados.....	107
Figura A.2 - Matriz de transição que representa os dados das regras de associação provenientes da figura A.1	108
Figura A.3 - <i>Script</i> utilizado para a estimativa da distribuição de pontuações de V_{CTR}	109
Figura A.4 - Histograma, CDF e estimativa de V_{CAT}	111
Figura A.5 - Histograma, CDF e estimativa de V_{CTR}	111
Figura A.6 - Histograma, CDF e estimativa de V_{VID}	112
Figura A.7 - Histograma, CDF e estimativa de V_{IMD}	112
Figura A.8 - Histograma, CDF e estimativa de V_{PED}	113
Figura A.9 - Histograma, CDF e estimativa de V_{IMV}	113
Figura A.10 - Histograma, CDF e estimativa de V_{TEU}	114
Figura A.11 - Histograma, CDF e estimativa de V_{DIV} (automóveis, marca).....	114
Figura A.12 - Histograma, CDF e estimativa de V_{NAV} (automóveis, marca).....	115
Figura A.13 - Histograma, CDF e estimativa de V_{REL} (automóveis, marca)	115
Figura A.14 - <i>Site</i> modelo concebido para testar e validar a solução desenvolvida	116
Figura A.15 - Zona de classificados produzidos pelo <i>script</i> da AdClip	117
Figura A.16 - Integração dos classificados na página do Jornal Público.....	118



Lista de abreviaturas e siglas

Abreviatura	Descrição
API	<i>Application Programming Interface</i>
AUC	<i>Area Under Curve</i>
CAREN	<i>Class Project Association Rule Engine</i>
CDF	<i>Cumulative Distribution Function</i>
CGI	<i>Common Gateway Interface</i>
CPA	<i>Cost Per Action</i>
CPI	<i>Cost Per Impression</i>
CPM	<i>Cost Per Mille</i>
CPC	<i>Cost Per Click</i>
CTR	<i>Click-Through Rate</i>
DNS	<i>Domain Name System</i>
HTML	<i>HyperText Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
IP	<i>Internet Protocol</i>
ISP	<i>Internet Service Provider</i>
K-S	<i>Kolmogorov Smirnov</i>
PPC	<i>Pay Per Click</i>
ROC	<i>Receiver Operating Characteristic</i>
SVM	<i>Support Vector Machine</i>
TCP	<i>Transmission Control Protocol</i>
URI	<i>Uniform Resource Identifier</i>
URL	<i>Uniform Resource Locator</i>
WWW	<i>World Wide Web</i>



Capítulo

1. Introdução



1.1. Expansão Sustentável da Internet

A expansão tecnológica das últimas décadas, patentes na crescente quantidade e qualidade de produtos e serviços disponibilizados, transformou o modo como as entidades colectivas ou individuais se relacionam e se apresentam à restante sociedade. O auge dessa expansão é, seguramente, a internet que ocupa um espaço fulcral no contexto pessoal e profissional de cada indivíduo. Actualmente com uma população mundial acima dos 7 mil milhões de pessoas, estima-se que cerca de 2.45 mil milhões use a internet ($\cong 35\%$) e cerca de 6.3 biliões ($\cong 90\%$) estejam já cobertos com, pelo menos, banda larga móvel 2G (Figura 1.1). A velocidade de conexão também tem registado melhorias progressivas, encontrando-se em média nos 35 Mb/s por utilizador (ITU - International Telecommunication Union, 2011).

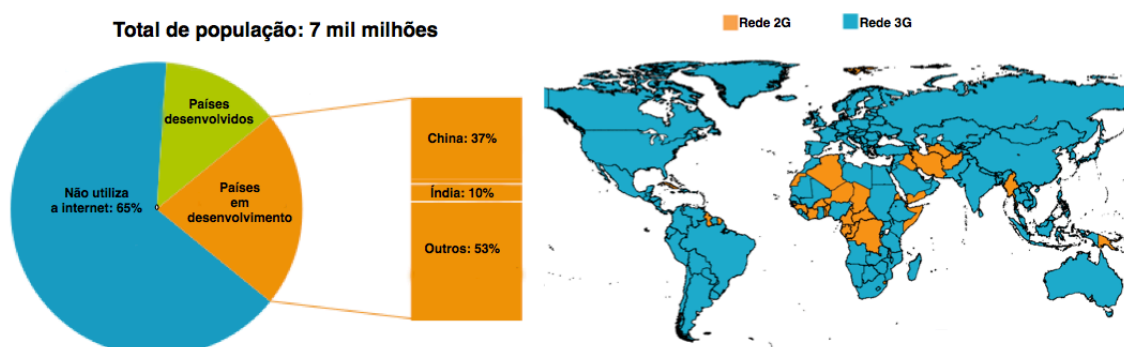


Figura 1.1 – Impacto da internet na população mundial (ITU - International Telecommunication Union, 2011)

Esta popularidade crescente e sustentada da internet foi, principalmente na última década, alimentada pela informação, curiosidade e partilha de conteúdo dos utilizadores. Foi deste modo que os *sites* de sucesso dos mais variados sectores emergiram: motores de busca (*Google* e *Yahoo!*), comércio electrónico (*eBay* e *Amazon*), informativos (*CNN* e *BBC*), software de comunicação (*Microsoft MSN* e *Skype*), reprodutores de vídeo (*YouTube* e *Dailymotion*) e mais recentemente as redes sociais, tais como *Twitter* e *Facebook* (BBC, 2010). No entanto, nem sempre a internet se apresentou com esta aparência. Entender a sua progressão ou o modo como é sustentada é compreender as diferentes necessidades da sociedade ao longo dos anos e os modelos de negócios adjacentes à internet (e.g. modelo de publicidade *pay-per-click*).

Decorria a década de 60, auge da Guerra Fria, quando os primeiros alicerces da internet se começaram a desenvolver. Licklider (1962) reconheceu potencial, para fins científicos e militares, na partilha de informação através de computadores interligados. Kleinrock (1961), no



âmbito do seu doutoramento, desenvolveu uma teoria matemática inovadora para a troca de informação entre computadores: informação dividida em pacotes arbitrários e com diferentes decisões de roteamento. Com base nos conceitos anteriores e com o objectivo de interligar bases militares ou departamentos de pesquisa a agência americana DARPA (*Defense Advanced Research Projects Agency*) lançou a ARPANet: primeira rede de computadores baseada na comutação de pacotes e sem uma rota fixa. Iniciou-se apenas com os nodos de UCLA (*University of California, Los Angeles*) e SRI (*Stanford Research Institute*) em 1969, alastrando-se de forma crescente até a década de 80, com mais de duas centenas de universidades e pontos estratégicos conectadas.

Surgiu ainda o protocolo TCP/IP para permitir a portabilidade entre redes e a capacidade de unificar os diferentes sistemas, bem como o DNS para simplificar a utilização da internet por meio da resolução de *hostnames* e IPs. Os primeiros domínios foram registados em 1985 e 1986, com intuítos comerciais, através de nomes sonantes da tecnologia: *Xerox, HP, IBM, Sun, Intel, AMD e Siemens*. Ainda assim, a generalidade das operações na rede implicavam um custo temporal e conhecimentos técnicos elevados mantendo-a longe da maioria da população.

Seria já no início da década de 90 que se deu, provavelmente, o maior passo na aproximação à internet actual. Berners-Lee e Cailliau (1990) apresentaram uma proposta denominada de *WorldWideWeb* com o objectivo de facilitar e uniformizar a utilização da rede, bem como automatizar alguns processos (e.g. notificações para o utilizador). Amplamente reconhecidos como os fundadores da internet, a proposta baseou-se numa arquitectura cliente-servidor onde toda a informação era visualizada através de um único software: o *browser*. No âmbito desse projecto viriam a ser desenvolvidas tecnologias que perduram até hoje: URI e URL para gerir identificadores únicos para os recursos de internet, linguagem HTML e protocolo HTTP (Berners-Lee & Fischetti, 1999). Em 1993 a CERN (*Organisation Européenne pour la Recherche Nucléaire*), organismo que financiou o projecto, anunciou que a *WorldWideWeb* seria livre e com possibilidade de integração com outros sistemas e extensões sem custos de licenciamento. Não tardou a surgirem outras iniciativas e propostas que despertaram a plataforma *web* para os números hoje conhecidos.

A visualização de informação de forma simples e sem grande custo técnico passou a ser um facto, estimulando a criação de alguns dos navegadores reconhecidos: *Opera, Netscape* e



Internet Explorer. Nos anos seguintes, segunda metade da década de 90, a prioridade focou-se na necessidade de encontrar com facilidade a informação desejada sem conhecer antecipadamente a sua localização. Com esse objectivo em mente, assistiu-se à criação e proliferação de alguns dos mais conceituados motores de busca até ao final da década: *Lycos*, *Altavista* (grupo *Yahoo*), *Sapo*, *Google* e *MSN Search* (actual *Bing* do grupo *Microsoft*).

A visibilidade incomparável que desde de então o universo *online* alcançou foi a alavanca necessária para a centralização dos mais variados serviços em plataformas *web*. Assim, a realização de algumas operações do quotidiano, tais como operações bancárias, compra de produtos ou conversação com amigos podem-se encontrar à distância de um clique e de forma gratuita. No comércio electrónico os interesses de entidades e clientes convergem. Por um lado as entidades, que procuram a maior e melhor visibilidade, a automação de processo, alta disponibilidade e redução de custos. Por outro lado, o cliente que procura uma diminuição do tempo necessário para realizar uma operação, i.e. várias alternativas, comparação fácil, escolha rápida e tudo conjugado num só espaço.

No entanto, existe um paradoxo interessante e inerente a muitos dos *sites* que disponibilizam estes serviços. Apesar do fluxo de clientes, que acarreta encargos de manutenção elevados (e.g. relativo à infraestrutura ou ao pessoal), são gratuitos para os seus utilizadores. Levanta-se então a questão da sustentabilidade financeira destes *sites* e dos modelos de negócio utilizados para gerar receitas. Tal como já acontecia com outros meios de comunicação (e.g. jornais e televisão), também na internet se viria a vender espaços reservados à publicidade de forma a gerar proveitos.

Na próxima secção introduzimos a evolução do modelo de publicidade *pay-per-click*, bem como as implementações adoptadas pela *Google* e pela *AdClip*. As situações de fraude existentes nos modelos de publicidade são expostas na secção 1.3, enquanto que as nossas motivações (secção 1.4) e objectivos (secção 1.5) para a proposta apresentada encerram o capítulo.

1.2. Evolução do Modelo PPC (*Pay-Per-Click*)

Tal como já evidenciado, os motores de busca revolucionaram não só o uso individual da internet mas também o uso comercial e o contacto com os utilizadores. Apesar de as pesquisas



realizadas por utilizadores serem maioritariamente curtas, revelam muitas das suas preferências dando a oportunidade de difundir publicidade relacionada com o seu perfil.

Os modelos de publicidade *online* apresentam-se como uma estratégia de divulgação que tira partido dessas mesmas preferências para ganhar visibilidade perante determinado público alvo. Por esse motivo, a sua evolução está intrinsecamente ligada aos principais motores de busca e às empresas que os sustentam através da impressão de anúncios patrocinados (i.e. pagos pelo anunciante) em cada pesquisa realizada. O lucro gerado com as operações de publicidade é a razão pela qual parte dos serviços dessas empresas se mantêm gratuitos.

Os primeiros esforços para associar conteúdos segundo as características dos utilizadores remonta ao início da década de 90, alguns anos antes do desenvolvimento dos motores de busca. A *DoubleClick*, integrada actualmente na *Google*, propôs o desenvolvimento de uma plataforma que apresentasse novos documentos aos utilizadores com base nas páginas visitadas. Iniciavam-se assim os primeiros estudos para relacionar os conteúdos da internet e os utilizadores.

Em 1994 a *HotWired*, primeira revista com suporte *online* e posteriormente adquirida pela *Lycos*, vendia no seu *site* os primeiros espaços reservados à publicidade de terceiros. Nessa época, a divulgação de serviços ou de empresas era apenas possível em meios de comunicação alternativos (e.g. jornal ou televisão) ou através dos *sites* dos próprios anunciantes. Face à necessidade de debitar monetariamente os anunciantes surgiu o primeiro método de pagamento: CPM (*cost-per-mille*) ou, alternativamente, CPI (*Cost-per-impression*). Assim, os anunciantes pagavam à *HotWired* com base no número de impressões de cada anúncio ou por cada milhar de impressões. A gigante de telecomunicações norte americana *AT&T* acabou por ser a responsável pelo primeiro anúncio impresso na história da internet (Figura 1.2).



Figura 1.2 – Primeiro anúncio patrocinado disponibilizado na internet

A impressão de anúncios realizada pela *HotWired* era aleatória e não considerava o utilizador e as suas preferências. Consequentemente, o método de pagamento CPM (pagamento por impressão) revelou-se extremamente negativo para os anunciantes que não recebiam visitas frequentes de utilizadores. Com base nestes argumentos, em 1996, empresas como a *Amazon*



e a *CDNow* forçaram a alteração do método de pagamento junto dos editores. Surgiu assim um segundo tipo de pagamento denominado de CPA (*Cost-per-Action*), onde o pagamento apenas se realiza se o utilizador executar uma determinada acção (e.g. finalizar compra ou entrar em contracto com o anunciante). Nestes moldes, o cenário inverte-se quando comparado com o pagamento CPM uma vez que o risco do anunciante é praticamente nulo e a margem de lucro dos editores reduz significativamente.

Apenas em 1998 a *GoTo*, mais tarde renomeada de *Overture* e posteriormente adquirida pela *Yahoo*, tira proveito dos termos utilizados nas pesquisas dos motores de busca para realizar a impressão de anúncios. O método de pagamento implementado previa que cada anunciante fosse debitado sempre que um utilizador realizasse um clique num dos seus anúncios. Surgia assim o primeiro modelo de publicidade baseado no clique do utilizador e denominado de *pay-per-click* (PPC).

O número de intervenientes no modelo PPC depende do tipo de implementação escolhida (Figura 1.3). Na versão de três intervenientes, implementada por exemplo pela *GoTo*, os anunciantes interessados em divulgar e lucrar com a venda dos seus produtos ou serviços disponibilizam os seus anúncios ao editor. Indicam, igualmente, os montantes que estão dispostos a oferecer por cada clique nos seus anúncios. Por sua vez, o editor difunde os anúncios junto dos utilizadores sempre que considere adequado (e.g. existe uma relação entre anúncio e o perfil do utilizador) e tendo em conta o montante oferecido, por ordem decrescente. Um clique no anúncio impresso implica que o editor reencaminhe o utilizador para o anunciante em questão e lhe execute o respectivo débito monetário.

A escolha dos anúncios a apresentar a cada utilizador é uma operação complexa que ao longo dos anos tem sofrido muitas alterações e optimizações, sendo da responsabilidade do sistema de recomendação implementado pelo editor. Os detalhes sobre os sistemas de recomendação não serão abrangidos no âmbito deste documento, salientando-se apenas que o seu objectivo é alcançar um equilíbrio entre o interesse de um anúncio para o utilizador e o lucro que o mesmo gera para o editor. A fórmula de cálculo mais simplista é $Pr_{anuncio} * V_{anuncio}$, i.e. a probabilidade de clique no anúncio por parte de um utilizador e o valor que o mesmo irá gerar para o editor. Assim, para n posições, serão escolhidos e ordenados decrescentemente os n anúncios com maior potencial de lucro (Ricci, Rokach, Shapira, & Kantor, 2011)(Jannach,



Zanker, Felfernig, & Friedrich, 2010).

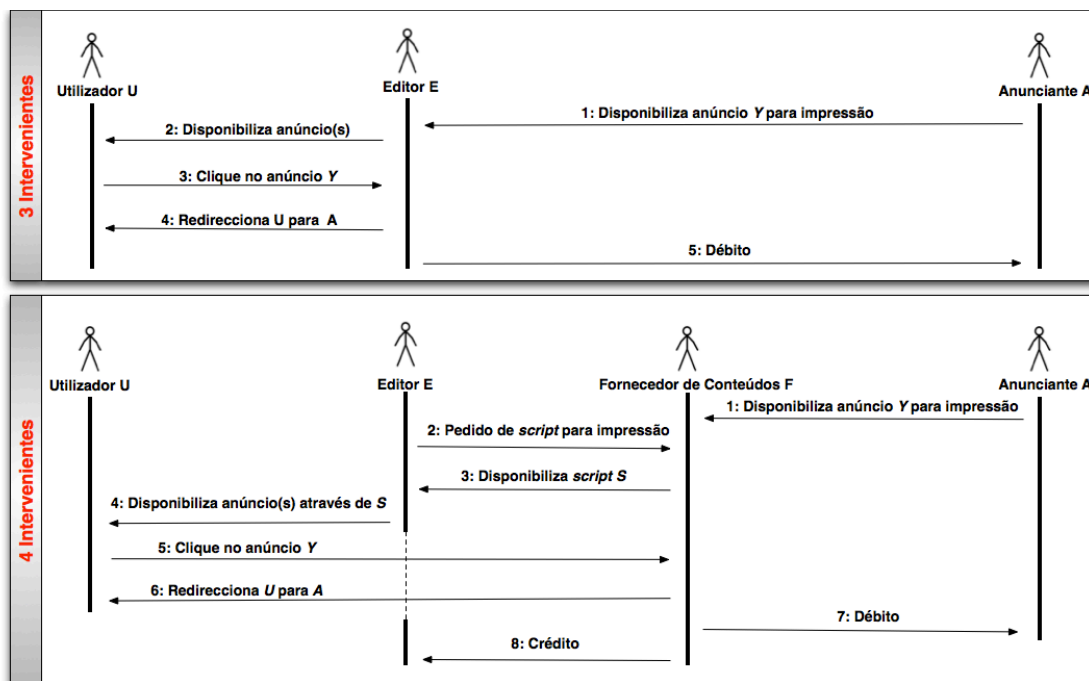


Figura 1.3 – Sequência de actividades para o modelo PPC de três e quatro intervenientes

O modelo PPC de quatro intervenientes é recorrente precisamente nos casos onde o editor pretende lucrar com o sistema de publicidade, mesmo não possuindo um sistema de recomendação próprio. Assim, o fornecedor de conteúdo é o responsável por disponibilizar os melhores anúncios para cada utilizador (e.g. via *script* ou API), garantindo a troca de anúncios entre as partes e a veracidade do modelo (Metwally, Agrawal, & El Abbadi, 2005). Veremos mais tarde que nos anos subsequentes, por iniciativa da *Google*, seria implementada uma segunda versão do modelo PPC mais justa e eficiente.

O modelo PPC é aceite pelos intervenientes como a solução mais equilibrada, visto que o clique representa a vontade do utilizador em ver o anúncio e é um compromisso intermédio entre a impressão e a conversão de um anúncio (e.g. compra de um produto). Segundo o estudo da *Interactive Advertising Bureau* (2012), especialista no mercado publicitário, as empresas que implementam modelos de publicidade *online* nos Estados Unidos geraram receitas que ascendem aos 17 mil milhões de dólares apenas no primeiro semestre de 2012 (aumento de 14% em comparação com período homólogo em 2011).

Com um crescimento progressivo e estimado de mais de 20% ao ano, prevê-se que em 2012 se facture mais de 8 mil milhões de dólares em cada trimestre, atingindo novo recorde em termos



de receitas absolutas e provando a rentabilidade da publicidade na internet (Figura 1.4). Os tipos de anúncios que mais contribuem para estes valores são: impressões em motores de busca (47%), impressões na rede de editores (21%) e anúncios classificados (7%).

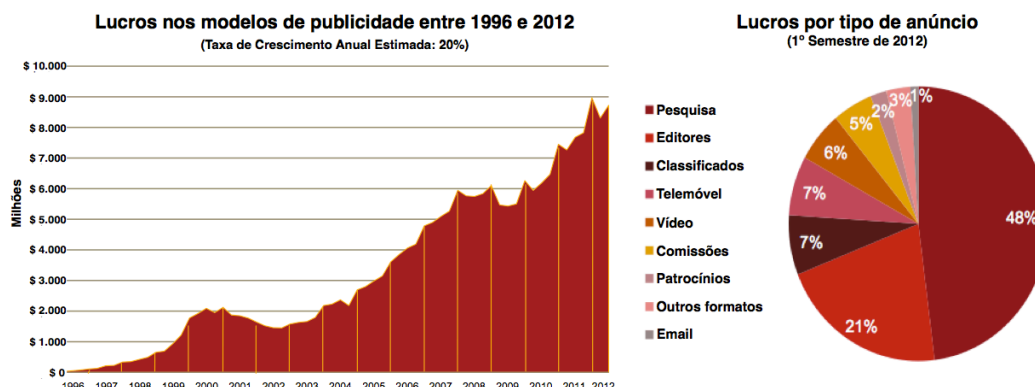


Figura 1.4 – Evolução (esquerda) e distribuição (direita) de lucros gerados pelos modelos de publicidade *online*

Mais de metade do valor transaccionado é relativo às áreas de negócio de retalho (20%), serviços financeiros (13%), automóvel (13%) e telecomunicações (12%). Pela figura 1.5 observa-se que 67% dos anunciantes preferem optar por pagamento CPA ao invés do CPM (31%) ou de soluções híbridas (2%). Neste estudo o pagamento CPC não foi considerado.

De referir igualmente que desde de 2003 estamos perante um oligopólio com 75% dos lucros a serem distribuídos pelas 10 maiores empresas a explorar a rede de publicidade (e.g. *Google*, *Yahoo* e *Microsoft*) e 90% dos lucros pelas primeiras 50 empresas (Figura 1.6).

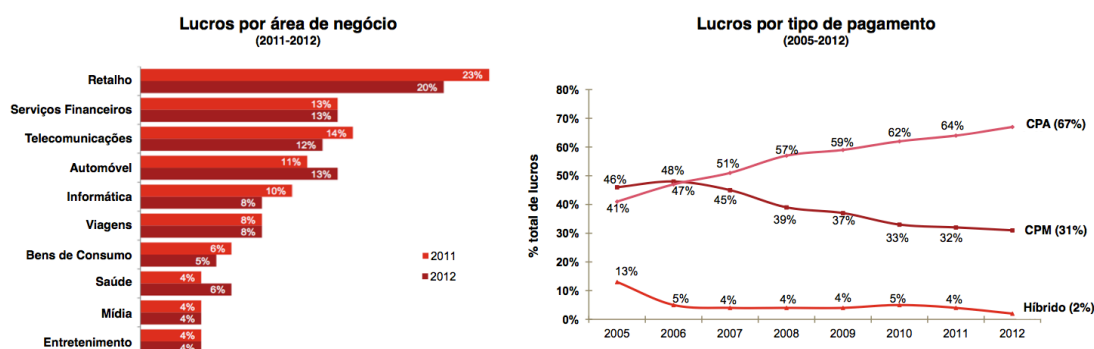


Figura 1.5 – Áreas de negócio (esquerda) e tipo de pagamento (direita) com mais influência nos modelos de publicidade *online*

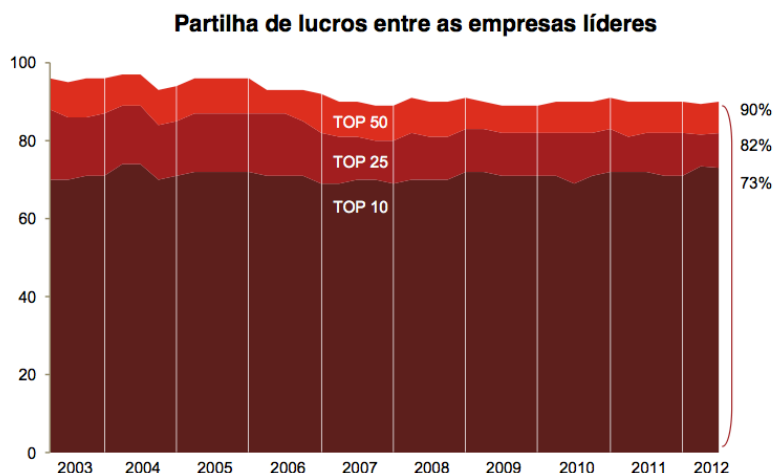


Figura 1.6 – Distribuição de lucros por entre as 50 empresas que lideram o mercado publicitário.

1.2.1. Implementação adoptada pela *Google*

Precisamente em 1998, ano em que a *GoTo* lançou a primeira versão do modelo PPC, nasceu aquela que é actualmente uma das empresas mais valiosas do mundo, a *Google*. Com a imagem de marca focada no seu motor de busca, actual líder de mercado, bastou apenas dois anos até ser implementado o seu primeiro serviço de publicidade. Esse serviço, denominado de *AdWords*, é ainda hoje o responsável pela impressão de anúncios no motor de busca da *Google*. A primeira versão do *AdWords* utilizava a estratégia de pagamento CPM, sendo este o valor responsável pela ordenação ou escolha dos anúncios filtrados.

Decorridos dois anos, fevereiro de 2002, a *Google* concluiu que o modelo de negócio adoptado era pouco eficiente e impedia lucros mais significativos. À semelhança da *GoTo*, alterou o modo de pagamento do *AdWords* para CPC. Concluiu, igualmente, que a ordenação segundo o montante oferecido (i.e. custo por clique) era desapropriado uma vez que os anúncios mais caros estavam sempre no topo e, independentemente da sua popularidade, perdiam interesse a longo prazo. Consequentemente, havia uma redução de cliques e menos circulação de dinheiro.

Desde então o *AdWords* permite aos anunciantes submeterem um conjunto de palavras-chave para cada anúncio, bem como o montante máximo que pretendem desembolsar por cada clique e para cada palavra-chave. A ordenação dos anúncios passou a ter em consideração (para além do custo por clique oferecido) a relevância do anúncio para o utilizador. Desta forma, um anúncio com baixa oferta mas com elevada procura poderá ser impresso mais vezes que a situação inversa. Adicionalmente, o valor debitado ao anunciante é apenas o mínimo necessário



para manter a sua posição, i.e. o mesmo montante que o valor oferecido pela posição imediatamente inferior. Tomando o exemplo da figura 1.7, um clique no anúncio com mais destaque (i.e. Renault) gera um débito de 0,72€ em vez dos 0,79€ oferecidos.






Posição do anúncio	1	2	3	4	5
Anúncio	 RENAULT	 CITROËN	 BMW	 Mercedes-Benz	 SEAT
Preço por clique oferecido	0.79€	0.72€	0.61€	0.49€	0.31€
Valor cobrado pela Google em caso de clique	0.72€	0.61€	0.49€	0.31€	(...)

Figura 1.7 – Método de débito aplicado pela Google aos seus anunciantes

Desta forma, o valor é normalmente inferior ao custo por clique definido pelo anunciante tornando o modelo PPC mais justo. Apesar de o procedimento ser conhecido, o anunciante não tem acesso aos montantes oferecidos pelos seus rivais e não lhe é possível antecipar os custos de uma campanha publicitária. Para tal, o anunciante tem a possibilidade de definir o montante máximo disponível para cada campanha publicitária, período (e.g. dia, semana, mês) ou palavra-chave. Quando este valor é atingido os anúncios em questão deixam de ser considerados para impressão ou aparecem de forma muito espaçada não sendo cobrado qualquer valor ao anunciante.

Figura 1.8 – Anúncios impressos através do AdWords (modelo PPC de 3 intervenientes)



Estas alterações impulsionaram os proveitos da *Google* e mudaram definitivamente o conceito PPC. A implementação da *GoTo* tornou-se obsoleta e o modelo de publicidade da *Google* nunca mais se alterou do ponto de vista estrutural, sofrendo apenas ajustes de optimização. A figura 1.8 ilustra o modo como os anúncios do *AdWords* são impressos.

Embora este modelo PPC de 3 intervenientes implementado pelo *AdWords* seja um sucesso reconhecido, não permite explorar a rede de parceiros e editores da *Google* (e.g. *AOL* e *EarthLink*). Para suprimir esta lacuna foi criado, em 2003, o *AdSense*. O *AdSense* é um modelo PPC de 4 intervenientes onde a *Google* age como fornecedor de conteúdo e onde a incorporação de anúncios nos seus parceiros pode ser realizada por API e de duas formas distintas:

- AFS (*AdSense For Search*): Pedido de anúncios baseados nos termos de pesquisa realizada no *site* do editor;
- AFC (*AdSense for Content*): Pedido de anúncios baseados no conteúdo da página do editor, na localização do cliente e noutros atributos conhecidos antecipadamente pela *Google*.

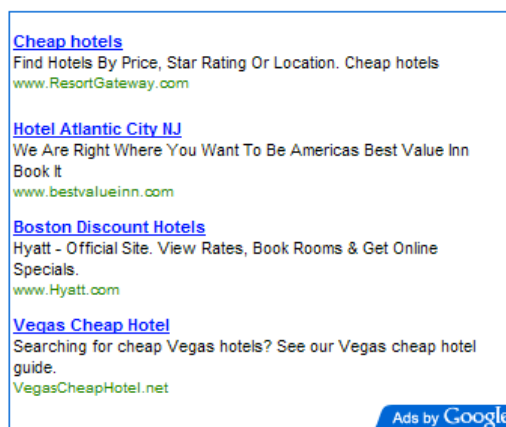


Figura 1.9 - Anúncios impressos num editor através do *AdSense* (modelo PPC de 4 intervenientes)

Tal como já referenciado, o fornecedor de conteúdo é o responsável pela atribuição dos anúncios e pela partilha de proveitos. Enquanto a primeira tarefa é visível e pode ser avaliada (Figura 1.9), a segunda é mantida sobre a máxima discrição evitando conflito de interesses entre editores e a própria *Google*. Neste contexto, apenas é disponibilizado aos editores relatórios com estatísticas sobre actividades ou cliques de utilizadores.

Embora esses dados permitam estimar os ganhos, não possibilitam a análise individual de cada clique salvaguardando a *Google* de queixas de utilizadores sobre situações anómalas (e.g.



situações de fraude). Por um lado temos o cliente, que paga e que tem o direito a perceber quais os cliques aceites ou rejeitados, por outro a *Google* que evita essas informações para não dar indicações sobre o seu modo de operar (e.g. processos de detecção de fraude).

1.2.2. Implementação adoptada pela *AdClip*

O modo de operar da *Google* viria a cativar diferentes empresas a utilizar ou adaptar o modelo PPC. Uma dessas abordagens é a organização de anúncios em forma de classificados, a terceira mais lucrativa nos Estados Unidos (Figura 1.4).

A *AdClip*, fundada em 2008, apresenta-se como a primeira rede de classificados *online* na internet. Com presença sólida em Portugal, conta com mais de 180 parceiros (i.e. editores) com destaques para *portal MSN*, *Jornal Público* e os *Diários de Coimbra*, *Minho* e *Aveiro*.

O seu principal objectivo é, através da sua rede de editores, permitir a rápida difusão de anúncios dos seus clientes. Tal como na *Google*, basta o anunciante inserir o anúncio uma única vez para que o mesmo seja instantaneamente difundido em centenas de outros *sites* (i.e. editores). A secção de classificados utilizada pela *AdClip* tende a gerar uma maior visibilidade e fluxo de utilizadores nos editores.

A incorporação de anúncios pode ser realizada por intermédio de uma API ou por *scripts*. A primeira implica um elevado tempo de implementação mas permite que cada editor personalize a sua área de classificados. A segunda garante uma implementação instantânea ao embeber os classificados na página do editor, mas não permite personalização (Anexo A.16).

O âmbito do projecto irá focar-se na detecção de fraude em modelos *pay-per-click* implementados em classificados *online*, precisamente a implementação utilizada pela *AdClip*.

1.3. Sistemas Fraudulentos no modelo PPC

O princípio de fraude está intrinsecamente associado a esquemas ilícitos, praticado com fins lucrativos ou para obtenção de algum tipo de vantagens ou regalias e punidos por lei como crime. A sua aplicação está espalhada pelas mais variadas áreas, aumentando drasticamente com a expansão tecnológica que permitiu à comunidade uma rápida comunicação e o acesso a mais e melhor informação.



A extensão das situações fraudulentas é difícil de quantificar com rigor devido a dois factores primordiais: complexidade em apurar se as acções foram realizadas com intuito criminoso e a relutância das empresas em divulgar valores devido a falta de segurança manifestada perante terceiros (e.g. clientes). No contexto de fraude, o sector financeiro é um dos mais carismáticos e reconhecido devido à contrafacção ou roubo de informação de cartões bancários (Financial Fraud Action, 2012). Outro, apesar de não haver estudos conclusivos, é o sector de publicidade *online* onde se estima que no mínimo 10% a 15% dos cliques realizados sejam fraudulentos (Haddadi, 2010).

A fraude nos modelos de publicidade orientados ao clique é um tipo de crime que ocorre quando uma pessoa, *script* ou programa produz um clique num anúncio sem qualquer propósito de compra ou sem interesse no produto. Sendo este um modelo que circula grandes montantes de dinheiro, a veracidade do mesmo é posto em causa perante a existência de incentivos desonestos para todos os intervenientes. As diversas formas de fraude reconhecidas são: deflação do número de cliques (denominada neste documento por fraude do tipo I), inflação do número de cliques de forma individualizada (fraude de tipo II) e inflação de cliques de forma coligada (fraude de tipo III).

A primeira situação de fraude, acredita-se, não está activa nos modelos PPC implementados actualmente. A sua aplicação implica que a gestão do número de cliques (ou acções) seja controlada pelo anunciante e não pelo editor ou pelo fornecedor de conteúdo. Na teoria, o número de cliques produzidos é igual ao número de utilizadores reencaminhados do editor para os respectivos anunciantes. No entanto, veremos, existem circunstâncias que impossibilitam tal conclusão (e.g. problemas técnicos). Sendo a gestão realizada pelo anunciante e tirando partido desta situação, o valor creditado aos editores ou aos fornecedores de conteúdo é menor do que o real. Este esquema fraudulento atenuou após a introdução do modelo PPC implementado pela Google.

Por outro lado, a fraude de tipo II e III beneficia os editores, fornecedores de conteúdo e, inclusive, alguns anunciantes. O lucro das empresas mantém uma proporcionalidade directa com o número de cliques realizados, pelo que é inequívoco o seu favorecimento. O caso mais comum é o dos editores que aliciam utilizadores para visualizar anúncios no seu *site* de modo a aumentar os seus proveitos. Neste caso, os utilizadores que compactuam com o esquema



fraudulento recebem parte dos lucros gerados. Recentemente temos assistido também a proliferação de serviços que obrigam à realização de múltiplos cliques em publicidade antes do utilizador aceder ao conteúdo pretendido. Ao contrário do cenário anterior, aqui o utilizador não recebe qualquer compensação.

Os anunciantes também podem beneficiar com a fraude no modelo PPC de duas formas distintas: melhoria no posicionamento de anúncios sem aumentar os valores oferecidos por clique e insatisfação dos seus rivais. Qualquer das situações ocorre quando um anunciante obriga, na sequência de múltiplos cliques, o editor a debitar o valor máximo definido pelos anunciantes rivais para determinado período ou campanha publicitária. Relembre-se que uma vez atingido esse valor, os anúncios deixam de ser considerados para impressão favorecendo dessa forma os restantes anunciantes.

O que diferencia o tipo de fraude II e III é o modo como estes cliques fraudulentos são produzidos. Assim, se forem produzidos de forma individualizada estamos perante fraude de tipo II. Esta vertente é a mais simples de identificar por ser recorrente e de implementação pouco rebuscada. Se, por outro lado, o tráfego responsável pelos cliques é partilhado por múltiplos utilizadores estamos perante fraude do tipo III. Este é o caso mais complexo de identificar e de implementar, uma vez que obriga à organização e sincronização de múltiplos utilizadores.

Com estes esquemas fraudulentos as companhias mundiais perdem centenas de milhões de euros para compensar os prejuízos causados aos seus anunciantes e reduzem a sua credibilidade junto do mercado.

1.4. Motivações

A detecção de fraude é uma actividade de extrema importância para os intervenientes e para a sobrevivência do modelo PPC. Com o crescimento progressivo da internet a quantidade de informação disponível aumenta, tornando a análise de dados uma tarefa complexa e temporalmente dispendiosa. São necessários sistemas automatizados para realizar estas tarefas e reduzir os níveis de fraude.

O foco da nossa proposta é a detecção de fraude de tipo II em modelos PPC que utilizem uma estrutura de classificados para divulgar os seus anúncios. A generalidade das soluções existentes do mercado é executada em *offline* e baseia-se na detecção de situações anómalas para atribuir



uma de duas classificações aos cliques: válido ou fraudulento. Uma situação que se desvia do esperado apenas pode ser considerada como suspeita (e nunca como fraudulenta) uma vez que a intenção do utilizador não é apurada ou mesurada em nenhum momento. Nesse sentido, as execuções em *offline* também não permitem atestar a intenção do utilizador na generalidade dos casos. Para tal, é preferível uma solução em tempo real que permita a prevenção para além de detecção e que fundamente as suspeitas que recaem sobre o utilizador.

Normalmente as variáveis analisadas são a taxa de cliques¹ e as características da máquina do utilizador. Para o primeiro caso assume-se a inexistência de valores elevados, enquanto para o segundo é expectável a ausência de configurações anómalas ou alterações sistemáticas nas características recolhidas. Infelizmente, este tipo de abordagem apresenta duas limitações evidentes e restritivas.

A primeira limitação está relacionada com o próprio objectivo do modelo PPC. Idealmente o sistema de recomendação apresenta ao utilizador anúncios que se relacionam com as suas preferências. Nestas circunstâncias, existem duas interpretações distintas para as taxas de clique elevadas: é uma consequência dos objectivos traçados e demonstra eficiência do sistema de recomendação ou, alternativamente, é uma situação suspeita por ser estatisticamente incomum. Como tal, utilizadores com taxas elevadas podem não ser fraudulentos.

A segunda limitação decorre do facto de ser impossível identificar um utilizador de forma inequívoca na internet. Sem essa identificação, o rastreio completo da actividade de um utilizador não é trivial, dificultando a análise de anomalias ou de alterações nas características da máquina. Adicionalmente, veremos que este é um tema com elevados contornos legais e sobre o qual a comunidade científica continua a interessar-se.

Fica evidente que tais limitações podem contribuir de forma considerável para o aumento do número de falsos positivos, i.e. cliques classificados como fraudulentos de forma errada.

¹ Rácio entre o número de anúncios impressos e o número de anúncios visualizados.

² Um *proxy* actua como intermediário na ligação entre o utilizador e o servidor destino permitindo, entre outras



1.5. Objectivos

O principal objectivo da nossa proposta é a criação e teste de um protótipo capaz de detectar e validar situações fraudulentas no modelo PPC, em particular numa estrutura de classificados *online* de 3 intervenientes. Para tal pretende-se que esse protótipo esteja capacitado de um processo de aprendizagem automática de modo a dispensar a intervenção humana no decorrer das suas execuções.

Sabemos que ignorando variáveis comportamentais e históricas estaremos a inviabilizar a análise do clique segundo a atitude dos diferentes utilizadores ao longo do tempo. Como tal, serão idealizadas variáveis numéricas (i.e. classificação quantitativa em detrimento da classificação qualitativa) que nos permitam representar – do melhor modo possível - o comportamento de um utilizador. Nessa tarefa, espera-se o auxílio dos dados fornecidos pela *AdClip Portugal*. Entre outras possibilidades, as variáveis devem ser capazes de medir e avaliar a qualidade e aleatoriedade na navegação, o tempo empregue na visualização de anúncios e a quantidade de duplicados gerados. Recorre-se à utilização de regras de associação de modo a derivar relações que nos fundamentem alguns comportamentos normais.

A identificação de um comportamento desviante é baseada nos valores das variáveis obtidos pelos restantes utilizadores. Assim, pretende-se estimar as distribuições desses valores e posteriormente, por teste de hipóteses, avaliar se o valor obtido pelo utilizador em questão é significativo ou não. Se for significativo, o utilizador é considerado suspeito.

Nestes casos, pretende-se que o sistema de recomendações se possa adaptar ao próprio utilizador, colocando-lhes cenários adversos de modo a apurar o seu interesse ou intenção. Se daí resultar evidências de fraude (e.g. reacções não esperadas), o utilizador é considerado fraudulento.

Uma vez que o protótipo será concebido para executar em tempo real, dando oportunidade de fundamentar as suspeitas existentes, parte dos dados devem ser materializados e actualizados periodicamente. Deste modo reduz-se a carga computacional, executando apenas o estritamente necessário (e.g. cálculo de variáveis e teste de hipóteses).

Não é de mais salientar que, no âmbito desta temática, qualquer proposta obterá sempre resultados que serão apenas uma aproximação à realidade. A subjectividade na avaliação de



comportamentos humanos nos modelos PPC não permite formalizar ou definir as situações fraudulentas de forma exacta. Pretende-se validar a proposta e o respectivo protótipo em contexto experimental (e.g. cenários pré concebidos) e em contexto real (e.g. integrado numa rede de classificados *online*).

1.6. Organização do documento

No capítulo seguinte serão abordadas propostas que, de forma directa ou indirecta, contribuíram para o desenvolvimento da solução apresentada. Sempre com as técnicas de mineração de dados como fundo, serão abordadas as problemáticas inerentes à identificação de utilizadores, à recolha de perfis de utilizador e à recolha de padrões de navegação. O capítulo encerra com um levantamento bibliográfico de propostas para a detecção de fraude de tipo I, II e III.

No terceiro capítulo apresenta-se detalhadamente a proposta desenvolvida. Inicia com uma perspectiva geral da solução que serve fundamentalmente como contextualização. Prossegue com informações sobre as variáveis implementadas, extracção de dados, extracção de conhecimento por via de regras de associação, estimativa de distribuições e parâmetros para os valores das variáveis, cálculo de *p-values* e, por fim, os cenários adversos criados para testar os utilizadores suspeitos.

Os resultados experimentais são expostos no quarto capítulo por demonstração do protótipo desenvolvido, bem como as reacções a 3 cenários (pré-concebidos) de fraude. No capítulo seguinte, quinto, surgem as conclusões com uma apreciação geral da proposta, as suas limitações e as ideias para trabalho futuro.

O documento encerra com a bibliografia consultada e com o conjunto de anexos.



Capítulo

2. Revisão Bibliográfica



2.1. *Web Mining*

A quantidade de utilizadores e de dados que circula diariamente na internet tem crescido de forma progressiva. Srivastava, Cooley, Deshpande e Tan (2000) classificam esses dados, que denominaremos de dados *web*, em quatro categorias distintas: dados de contexto, dados de estrutura, dados de utilização e dados de utilizador. Os primeiros relacionam-se com a informação apresentada ao utilizador (e.g. texto ou imagens), enquanto os dados de estrutura representam o modo como a informação está organizada (e.g. hiperligações entre páginas). Os dados de utilização estão relacionados com o modo como o *site* é acedido e o modo como a navegação é realizada (e.g. IP origem ou sequência de páginas visitas) e os dados de utilizador representam informação sobre cada um dos visitantes do *site* (e.g. nome, idade ou preferências).

A análise eficiente destes dados tornou-se, nos últimos anos, uma operação humanamente impraticável. Assim, a utilização de técnicas de mineração de dados (em contexto *web* denominadas de *web mining*) tornou-se indispensável para obter conhecimento subjacente a este tipo de dados que, doutro modo, tendem a não ser perceptíveis. Este conhecimento traduz-se, fundamentalmente, em informações ou associações que representam um papel importante na compressão das necessidades individuais e colectivas dos utilizadores. Por essa razão, constitui um suporte à estratégia das organizações que vivem essencialmente do negócio *online*. No modelo PPC, em particular, assume-se como uma mais-valia na previsão dos melhores anúncios para cada utilizador e no aumento de sensibilidade do sistema para situações de fraude ou de ataque iminente. Veremos que as duas categorias de dados com mais importância, das quatro anteriormente mencionadas, são os dados de utilização e os dados de utilizador.

As etapas necessárias para a descoberta de conhecimento em dados *web* não são significativamente diferentes daquelas que são realizadas, por exemplo, numa base de dados tradicional. Considere-se o processo de Fayyad, Piatetsky-Shapiro e Smyth (1996) apresentado na Figura 2.1. A primeira fase é orientada à selecção e extracção de dados relevantes, tanto do lado do cliente como do lado do servidor. A fase seguinte é reservada ao pré-processamento, onde se incluem tarefas para eliminação de todos os dados que não reflitam actividade humana, tal como *web crawlers* ou outros agentes semelhantes (Tan & Kumar, 2002). Na terceira fase executa-se a transformação de dados de modo a que sua representação seja reconhecida e



interpretada correctamente (na quarta fase) pelas técnicas de mineração de dados que melhor se adaptarem aos objectivos pretendidos. A quinta e última fase é reservada à interpretação e filtragem de resultados mediante os objectivos traçados.

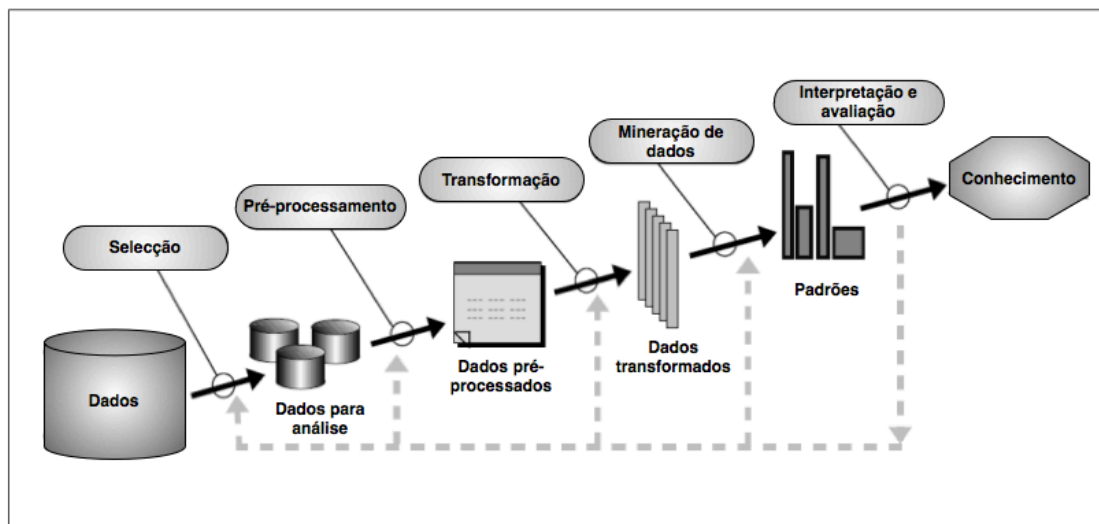


Figura 2.1 - Etapas do processo de extração de conhecimento numa base de dados (Fayyad *et al.*, 1996)

No entanto, para um contexto *web*, existe uma nuance na etapa de transformação de dados: a identificação inequívoca do utilizador que gerou os dados para análise. Se nos restringirmos ao modelo PPC, esta identificação é essencial porque tal como já referenciado o modelo foi concebido para se comportar de modo distinto perante diferentes utilizadores e respectivas preferências. O processo de detecção de fraude é comprometido se não for possível identificar, ainda que de forma aproximada, o utilizador que gera cada clique.

A escolha das técnicas de mineração de dados a utilizar varia consoante os objectivos propostos, sendo as mais utilizadas as regras de associação e modelos de classificação (Witten, Frank, & Hall, 2005), a detecção de anomalias (Chandola, Banerjee, & Kumar, 2009) e detecção de acontecimentos frequentes e temporais (Laxman & Sastry, 2006). Os principais objectivos tendem a relacionar-se com conhecimento sobre o conjunto de páginas e termos de pesquisa utilizados frequentemente e o modo como se relacionam, padrões de navegação, detecção de comportamentos ou de características atípicas e a ordem pela qual as acções são executadas (Pei, Han, Mortazavi-Asl, & Zhu, 2000).

Este conhecimento será denominado, no contexto deste documento, como padrões individuais se forem obtidos tendo em conta apenas um utilizador ou como padrões colectivos se tiverem



como base um conjunto finito de utilizadores. Se estes padrões forem adquiridos com base em dados de utilizador são denominados de perfis de utilizador, i.e. um conjunto de preferências ou características (e.g. gosta de música). Por outro lado, se forem adquiridos com base em dados de utilização são denominados de perfis de navegação uma vez que representam um conjunto de acções realizados por um utilizador, por determinada ordem e intervalo temporal (e.g. ouviu música *pop* e, minutos depois, música *rock*).

Uma vez adquirido o conhecimento sobre este tipo de padrões pode-se aplica-lo das mais variadas formas, entre eles:

- No auxílio à criação de *sites* que se adaptem dinamicamente a cada visitante e que melhorem a sua experiência, técnica muito comum em comércio electrónico e denominado de *adaptive web* (Perkowitz & Etzioni, 2000);
- No auxílio à construção e organização da estrutura de *sites* (Eirinaki & Vazirgiannis, 2003);
- Na previsão de próximos acessos de modo a diminuir a latência nos sistemas de *cache* (Chen & Zhang, 2003);
- Na escolha dos resultados devolvidos para uma determina pesquisa num motor de busca (Speretta, 2005).

A sua aplicação alastra-se, igualmente, aos modelos PPC uma vez que os perfis de utilizador aumentam a eficiência na correspondência utilizador-anúncio e descrevem as preferências de um utilizador. Por outro lado, o comportamento com fins ilícitos tende a desmarcar-se do comportamento dos restantes utilizadores (e.g. sequência de acções pouco frequentes). Consequentemente os perfis de navegação tornam-se úteis na detecção de fraude.

É evidente que a problemática de fraude nos modelos PPC é composta por um conjunto de subproblemas que devem ser individualmente resolvidos de modo a alcançar uma solução eficiente. Nas próximas secções, embora não se debrucem directamente sobre a questão da detecção de fraude em sistemas PPC, são analisados estudos nas três áreas que consideramos fulcrais para a compreensão de todo o problema: identificação de utilizadores, perfis de utilizador e perfis de navegação. O capítulo encerra com uma análise às propostas que de uma forma objectiva propuseram soluções para resolver a problemática debatida neste documento.



2.1.1. Identificação do utilizador

A questão da identificação dos utilizadores na internet representa ainda hoje uma problemática em aberto na comunidade científica, muito em parte pelos contornos éticos e legais que a mesma levanta (Caudill & Murphy, 2000). Os resultados obtidos pelas técnicas mais usuais, sessões baseadas em IP e *cookies*, tendem a ser pouco conclusivos por se fundamentar na máquina que realiza o acesso e não no humano que está no controle do mesmo. Nestes termos, um utilizador que use múltiplos computadores (e.g. acesso no telemóvel e no computador pessoal) poderá ser considerado como sendo dois indivíduos e múltiplos utilizadores que partilham o mesmo computador (e.g. espaços públicos que disponibilizam serviço de internet) poderão ser considerados como sendo o mesmo indivíduo.

A identificação por IP é baseada na informação cedida pelo fornecedor de serviço de internet (ISP), não podendo ser controlada pelo utilizador. Por norma assume-se tratar-se do mesmo utilizador se, para o mesmo IP, dois acessos forem realizados num espaço temporal suficientemente curto e/ou onde a localização desse IP se manteve inalterável. A localização é obtida interrogando directamente o próprio dispositivo (e.g. caso dos equipamentos que possuem geolocalização por triangulação) ou organizações responsáveis por identificar geograficamente um utilizador com base nas informações que cada ISP regista sobre os seus IPs (e.g. RIPE). No entanto, por uma questão de privacidade estas informações não reflectem a localização exacta do utilizador. Esta situação, bem como o uso de *proxies*² ou o dinamismo do IP são apontados como os pontos de falha na identificação por IP (Search Engine Land, 2007).

Por outro lado, os *cookies* apresentam-se actualmente como a técnica mais utilizada (Gauch, Speretta, Chandramouli, & Micarelli, 2007). A sua criação e manipulação é exclusiva de cada *site*, embora possam ser eliminados pelos utilizadores uma vez que as informações são armazenadas em ficheiros que se alojam no computador do mesmo. Na verdade, este facto acaba por constituir, tal como no IP, uma limitação para a identificação de utilizadores.

Se excluirmos os locais onde a autenticação é obrigatória, a sessão de um utilizador é normalmente definida em função destes dois parâmetros. Se o *cookie* é removido mas o utilizador acedeu com o mesmo IP instantes antes e com a mesma localização, então a

² Um *proxy* actua como intermediário na ligação entre o utilizador e o servidor destino permitindo, entre outras funcionalidades, bloquear, alterar e omitir informações do utilizador.



identificação do utilizador não foi perdida. A mesma situação ocorre se o IP alterar mas o *cookie* não é removido pelo utilizador. Por outro lado, o rasto do utilizador é perdido no caso onde ocorre alteração de IP e os *cookies* são removidos, marcando o fim do ciclo de vida de um utilizador *web*.

Consciente destas limitações, a comunidade científica tem-se debruçado sobre técnicas que permitam identificar um utilizador com uma precisão superior. As soluções propostas recaem, sobretudo, em técnicas baseadas em *third-party cookies* e em técnicas de mineração de dados que permitam reconstruir sessões através do comportamento individual de cada utilizador.

Os *third-party cookies* são *cookies* guardados com um domínio diferente do visitado pelo utilizador, possibilitando a uma terceira entidade realizar a sua leitura (Tappenden & Miller, 2009). Por exemplo, se um utilizador visitar o *site* www.exemploA.com ou o www.exemploB.com, ambos iram editar um *cookie* com o nome www.terceiros.com. Tal operação permitirá ao *site* www.terceiros.com saber quando e em que contexto o utilizador visitou os *sites* anteriormente referenciados. Para além de colectar mais informação sobre cada utilizador, bastará um dos *sites* identificar o utilizador (e.g. por autenticação ou porque o seu *cookie* não foi removido) para que este passe a ser reconhecido nos restantes *sites*.

Ivancsy e Juhasz (2007) basearam-se nesta técnica para identificar o utilizador que está no controle de uma máquina. A solução é suportada por uma coligação entre *sites* onde todos eles criam os *third-party cookies* e alguns requerem autenticação do utilizador de modo a aumentar a probabilidade de identificação. Assim que a identificação seja conhecida é divulgada por todos os *sites* que se encontram coligados. Juels, Jakobsson e Jagatic (2006) sugeriram uma aplicação equiparada, mas com níveis de privacidade superiores para ultrapassar as preocupações éticas e legais. Com este tipo de soluções, a identificação do utilizador continua a ser garantida no caso de eliminação parcial dos *cookies* e é mais rápida e eficiente, devido a comunicação entre múltiplos *sites*, no caso em que todos os *cookies* são removidos. As principais limitações desta abordagem encontram-se na relutância dos *sites* contribuírem para esta partilha de informação e na probabilidade de o utilizador visitar um subconjunto desses *sites* sem realizar qualquer autenticação e sem que lhe sejam identificados *cookies* válidos.

Para contrariar essas limitações, Spiliopoulou, Mobasher, Berendt e Nakagawa (2003) sugeriram a reconstrução de sessões baseando-se no comportamento individual de cada utilizador. A



proposta sustenta-se em dois tipos de técnicas: proactivas e reactivas. As técnicas proactivas tentam diferenciar os utilizadores antes ou durante a sua navegação através de informações fornecidas de forma directa (e.g. autenticação) ou indirecta (e.g. *cookies*). As técnicas reactivas baseiam-se nas características individuais de cada utilizador (e.g. tempo de sessão, navegação ou *referrer*³) e são identificadas com base em regras de associação. O principal objectivo destas técnicas é para cada utilizador desconhecido, i.e. sem *cookie*, com IP distinto ou que não tenha realizado qualquer autenticação, associar-lhe a identificação do utilizador que possui um comportamento mais semelhante ao seu. Se não forem encontrados comportamentos semelhantes, ser-lhe-á associada uma nova identificação. O recurso a técnicas de mineração de dados para a reconstrução de sessões foi igualmente abordado por Zhang e Ghorbani (2004), Gao e Sheng (2004) e Yang (2010). Infelizmente, a precisão desta solução está directamente relacionada com a heterogeneidade entre utilizadores que, em contexto real, nem sempre se verifica.

A identificação pode ser igualmente obtida por sessões de *browser* e, embora de forma menos comum e com maiores limitações, através de *software* instalado no computador do utilizador ou através do registo das máquinas numa *proxy* (Gauch *et al.*, 2007). A verdade é que, embora o conjunto de soluções e técnicas propostas sejam variadas, a problemática associada à identificação do utilizador está por solucionar e continuará seguramente a ser um dos alvos de investigação nos próximos anos.

2.1.2. Perfis de utilizador

Os perfis de utilizador são um conjunto de preferências ou interesses (e.g. tecnologia e desporto), características (e.g. idade e nacionalidade) e actividades (e.g. data da última visita) que definem um ou mais utilizadores. Tal como referido, este tipo de conhecimento é obtido por técnicas de mineração de dados que evidenciam padrões individuais ou colectivos e que darão origem a, respectivamente, perfis individuais e perfis colectivos. O seu processo de construção define-se em três fases essenciais: recolha de informação relevante sobre os utilizadores, definição do tipo de perfil e definição da representação.

³ O *referrer* é uma variável opcional do pedido HTTP enviado pelos *browsers* aos servidores *web*, possibilitando a identificação do servidor que gerou o pedido.



O processo de colecta de informação é realizado de forma explícita, implícita ou utilizando ambas as técnicas. A recolha explícita requer a participação ou intervenção dos próprios utilizadores para declarar os seus interesses e, conseqüentemente, representa um custo temporal para os mesmos. Pazzani, Muramatsu e Billsus (1996) utilizaram-na de forma pioneira para recomendar aos seus utilizadores novas páginas de interesse, com base no *feedback* anteriormente atribuído pelos mesmos. No entanto, esta técnica tende a ser mais eficiente em situações onde o utilizador desfruta da sua participação, tal como avaliações ou classificações de filmes (e.g. www.imdb.com e www.netflix.com).

Por outro lado, a recolha implícita é realizada sem a intervenção do utilizador e é baseada em análise de actividades ou navegação (Kelly & Teevan, 2003). Embora corrija a limitação da técnica anterior, a recolha implícita levanta a questão da privacidade de um utilizador. Assim, um utilizador que suspeite ou que tenha conhecimento que um determinado *site* está recolher informação poderá demonstrar relutância em visita-lo. Esta técnica foi já implementada para suportar a personalização de *sites* (Mobasher, 2007) e para apoiar na inferência de contexto de uma pesquisa (Sieg, Mobasher, & Burke, 2004).

Qual das duas técnicas é a mais eficiente dependerá sempre do contexto de aplicação, embora os estudos demonstrem que tendem a não ser significativamente diferentes (Quiroga & Mostafa, 1999; White, Jose, & Ruthven, 2001). No entanto, um dado adquirido é que a quantidade e diversidade de informação disponível garante melhor qualidade nos resultados finais (Teevan & Dumais, 2005).

A definição do tipo de perfil (segunda fase) inicia-se com a escolha do objectivo principal, i.e. se pretendemos obter perfis individuais, perfis colectivos ou ambos. A identificação correcta dos utilizadores, abordada da secção anterior, é nesta fase crucial se o objectivo passar pela criação de perfis individuais. Assim, podemos concluir de um modo geral que um interesse em B normalmente é precedido de um interesse em A, i.e. $A \rightarrow B$. No entanto, sem a identificação dos utilizadores não conseguiremos entender em quais deles esta implicação é mais evidente.

Os perfis que não permitem actualizações ao longo do tempo denominam-se de estáticos, enquanto os que permitem designam-se de dinâmicos e podem ser de dois tipos: curto prazo ou longo prazo. Os perfis de curto prazo estão relacionados com interesses recentes ou que se alteram frequentemente e, pelo facto de haver menos informação, são mais difíceis de identificar



e gerir. Os perfis de longo prazo referem-se a interesses que praticamente se mantêm inalteráveis ao longo do tempo e tendem a representar melhor o utilizador (Gauch *et al.*, 2007).

Quanto à representação, independentemente do tipo de perfil, baseiam-se em vectores de palavras-chave, redes semânticas de conceitos ou hierarquias de conceitos. Em comum têm o facto de utilizar termos (i.e. palavras) contidas em anúncios, *e-mails*, *sites* visualizados ou pesquisas realizadas para construir a representação do perfil de um utilizador. Por essa razão, os vectores de palavras-chave foram os primeiros a surgir, sendo os mais fáceis de implementar mas os menos eficientes.

Nos vectores de palavras-chave cada posição simboliza um interesse do utilizador e tem a si associado um peso numérico que permite revelar a importância do mesmo no seu perfil. Para esta representação, qualquer elemento (e.g. documento, resultado de pesquisa ou anúncio) que se pretenda comparar com o perfil do utilizador deve manter vectores semelhantes aos do utilizador. Assuma-se, no contexto dos modelos PPC, um utilizador U , um conjunto de α anúncios e β posições disponíveis para impressão de publicidade. Considere-se, igualmente, o vector VU para representar o perfil do utilizador, os vectores VA_j com $j \in [0, \alpha]$ para representar as características de cada anúncio, D_j para representar a semelhança entre vectores VU e VA_j e, por último, P_w com $w \in [0, \beta]$ para representar as posições disponíveis. Neste cenário, os anúncios escolhidos correspondem aos β vectores com maior D_j (Figura 2.2).

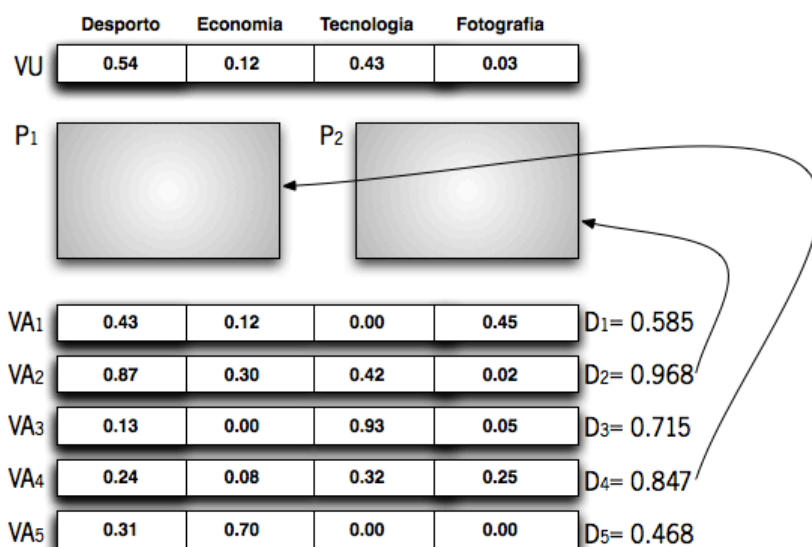


Figura 2.2 - Representação de perfis de utilizador usando vectores de palavras-chave (Gauch *et al.* 2007)



O modo como é obtido o vector de cada elemento varia entre implementações. Pode-se utilizar medidas referentes à frequência relativa de cada palavra no elemento, como por exemplo a medida *tf-idf* (Manning, Raghavan, & Schtze, 2008) ou medidas de ganho informativo como a divergência de *Kullback-Leibler* (Shmueli-Scheuer, Roitman, Carmel, Mass, & Konopnicki, 2010). O fundamental é que o vector consiga representar de forma correcta a importância de cada termo no elemento em análise. Em igual situação está a noção de proximidade entre dois vectores, por exemplo calculada com base no cosseno do ângulo entre vectores (Salton & McGill, 1986).

As limitações desta representação relacionam-se com o tamanho que o vector pode atingir – solucionado com a utilização de um vector por cada área de interesse - e a polissemia das palavras. Por esse motivo, embora sejam a base de outras representações, é necessário recuar alguns anos na literatura para encontrar sistemas famosos que tenham implementado vectores de palavras-chave: *Letizia* (Lieberman, 1995) e *Syskill & Webert* (Pazzani, Muramatsu, & Billsus, 1996) constroem perfis individuais de forma implícita e *Amalthea* (Moukas, 1997) constrói perfis individuais de forma implícita e explícita.

Outra abordagem, que resolve o problema da polissemia das palavras, é a representação baseada em redes semânticas de conceitos (grafos). Denomina-se de *conceito* termos equivalentes a categorias ou temáticas que sejam suficientemente abrangentes e não ambíguos. A principal barreira desta representação é encontrar os conceitos distintos que melhor definem as palavras reunidas aquando da colecta de dados.

A alternativa mais comum é a utilização de um dicionário externo onde substantivos, verbos, adjectivos e advérbios estão organizados dentro de sinónimos, cada um representando um conceito léxico fundamentado (e.g. <http://wordnet.princeton.edu>). Os conceitos pouco frequentes são eliminados e os pesos associados aos vértices e às arestas representam, respectivamente, a tendência dos mesmos existirem e coexistirem nas preferências do utilizador. De modo a lidar com perfis de curto prazo os pesos associados aos nodos ou às arestas baixam conforme os conceitos vão deixando de ter relevância.

Gentili, Micarelli e Sciarrone (2003) desenvolveram um sistema para, numa biblioteca *online*, filtrar os melhores documentos para um determinado utilizador com base no seu perfil individual de longo prazo e com recolha de informação explícita. Adicionalmente enriqueceu a



representação incluindo palavras-chave (denominadas de *satélites*) em cada conceito (denominados de *planetas*), tal como ilustra a Figura 2.3. Micarelli e Sciarrone (2004) aplicaram igualmente redes semânticas para criar o mesmo tipo de perfis, de modo a filtrar resultados do motor de busca *Altavista*.

A última representação, hierarquia de conceitos representada por árvores *n*-árias, é similar à anterior uma vez que utiliza igualmente conceitos em vez de palavras-chave. No entanto os vértices conectam-se de modo a estabelecer uma hierarquia, muitas vezes equiparada à estrutura de conteúdo de um *site* (e.g. <http://dir.yahoo.com>). O interesse dos utilizadores em cada conceito, i.e. vértice, é evidenciado pela variável numérica que lhe está associado no seu perfil (Figura 2.4).

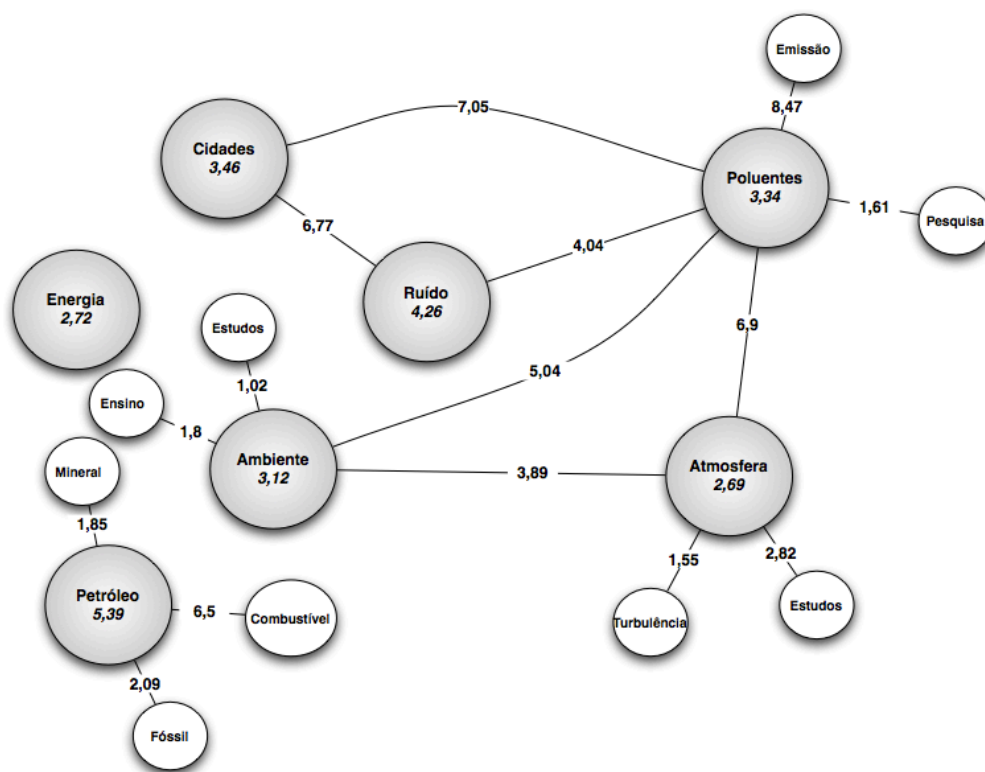


Figura 2.3 - Representação de perfis de utilizador usando redes semânticas de conceitos (Gentili *et al.*, 2003)

Esta representação foi já utilizada para construir perfis com base em páginas visitadas ou pesquisas realizadas (Liu, Yu, & Meng, 2002; Sieg, Mobasher, & Burke, 2004) e com base em artigos de investigação consultados (Middleton, Shadbolt, & De Roure, 2003). No entanto, Chin Chen, Chang Chen e Sun (2001) propuseram uma hierarquia dinâmica de modo a que os perfis sejam suficientemente ajustados aos interesses do utilizador.



Para tal, consideram-se os primeiros Ω níveis como fixos e os restantes como dinâmicos. Sempre que o peso de um nó ultrapassar um limite superior β^+ será dividido em sub-nós. Por outro lado, sempre que o peso de um nodo atinja um limite inferior β^- será fundido com o seu nodo superior. A principal limitação desta abordagem está em identificar correctamente o valor para Ω e para os limites β^- e β^+ .

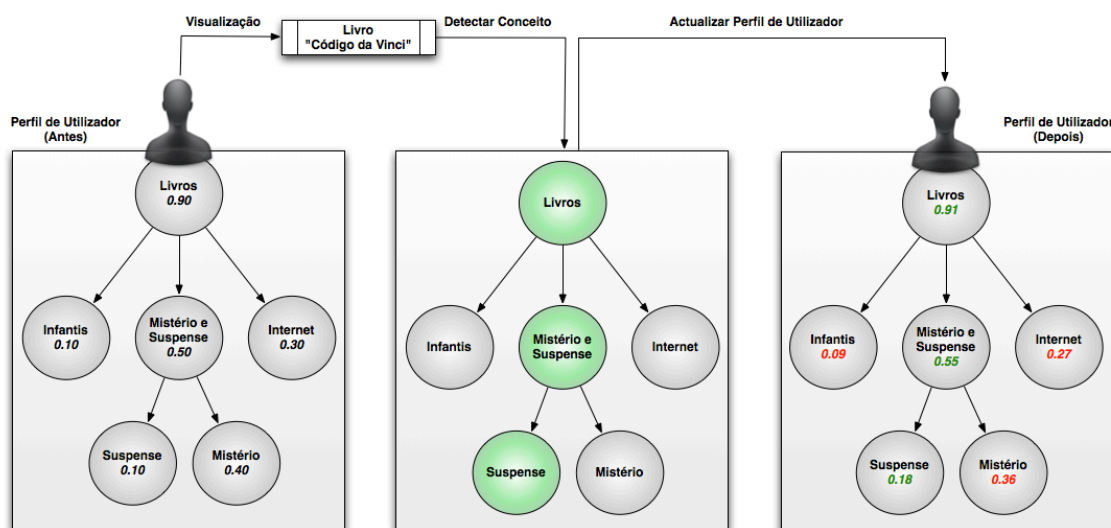


Figura 2.4 - Representação de perfis de utilizador usando hierarquias de conceitos (Sieg, Mobasher, & Burke, 2010)

2.1.3. Padrões de navegação

Na literatura os padrões de navegação surgem variadas vezes associados aos perfis de utilizador. No entanto, o objectivo e o tipo de dados de onde se extrai conhecimento são distintos. Na secção anterior abordamos os perfis de utilizador, responsáveis por analisar os dados do utilizador e identificar preferências ou interesses. Por seu turno, os padrões de navegação são responsáveis por extrair conhecimento dos dados de utilização, identificando actividades ou sequências de acções relevantes.

Para detectar eficazmente este tipo padrões é necessário definir acertadamente o objectivo primário e o tipo de técnicas a utilizar. Se o objectivo é identificar navegações comuns, considerando-as normais, estaremos perante casos de detecção de padrões frequentes e/ou sequenciais, com ou sem restrições temporais. Por outro lado, pode-se pretender identificar navegações incomuns ou anormais através do seu desvio para as restantes navegações, estando perante uma situação de detecção de comportamento desviante. A variedade de técnicas a utilizar é extensa e os resultados variam em função da sua escolha.



As cadeias de *markov* têm sido amplamente utilizadas pela comunidade científica para representar comportamentos de utilizadores. Ye (2000) utilizou-as para, com base nos dados históricos, modelar o comportamento habitual dos utilizadores num computador ou numa rede, i.e. padrões colectivos. No entanto, as acções ou opções dos utilizadores são baseadas em factos de curto-prazo e de longo-prazo não respeitando uma ordem específica. Uma vez que as cadeias de *markov* obedecem à propriedade de que os estados anteriores são irrelevantes para a predição dos estados seguintes desde que o estado actual seja conhecido, foi concluído pelos autores que não se adaptavam por serem pouco abrangentes. Apesar do custo computacional, as cadeias de *markov* de ordem n (onde n indica o número de antecedentes considerados para predição) foram apontadas como a solução natural desde que se conheça o n ideal e suficientemente curto para descrever o comportamento dos utilizadores.

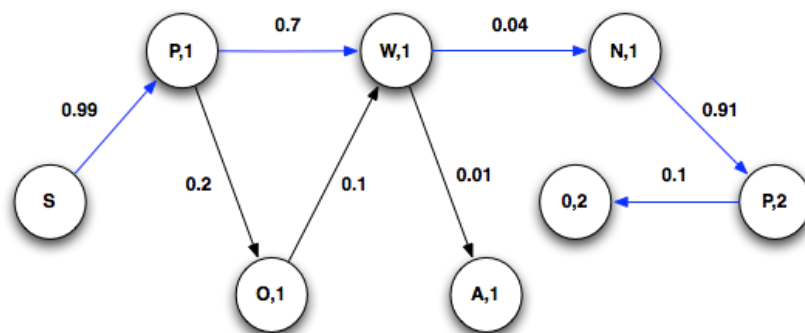
Igualmente com cadeias de *markov*, Sadagopan e Li (2008) propuseram-se a classificar sessões de utilizadores como típicas ou atípicas em motores de busca que implementassem modelos PPC. Para tal, abordaram a problemática medindo a raridade de uma sessão através da distância *mahalanobis* de diversas variáveis e considerando as sessões com maior valor como atípicas.

Os eventos considerados para a construção do grafo que descreve a navegação geral dos utilizadores no motor de busca foram: pedido de página (P), clique em anúncio (O), clique num dos resultados apresentados (W), clique na opção “Seguinte” para requerer a próxima página de resultados (N) e qualquer outro evento diferente dos enumerados (A). Existe ainda associado um indicador da página onde ocorre o evento. A pontuação obtida resulta da multiplicação das probabilidades condicionadas, i.e. do custo das arestas do grafo (Figura 2.5). No entanto, uma vez que um utilizador com elevado número de eventos tende a ter uma pontuação muito reduzida, é obtido o logaritmo desse valor e normalizado pelo número de eventos. Nestes moldes, quanto mais uma pontuação se distanciar de zero mas atípica será considerada a sua navegação.

Infelizmente, esta pontuação não é suficiente uma vez que dois utilizadores com a mesma pontuação podem ter comportamentos completamente distintos. Como tal são consideradas outras variáveis, tais como quantidade de páginas requisitadas ou quantidade de cliques em anúncios. Usando todas essas variáveis, foram consideradas como sessões atípicas aquelas que



possuam uma distância *mahalanobis* superior a um limite predefinido. A definição correcta deste limite constitui uma restrição nesta abordagem.



$$Y = P((P,1) | S) + P((W,1) | (P,1)) + P((N,1) | (W,1)) + P((P,2) | (N,1)) + P((0,2) | (P,2)) = 0.00252$$

$$\text{Pontuação} = \ln(Y) / 5 = -1.197$$

Figura 2.5 - Representação de perfis de navegação usando cadeias de *Markov* (Sadagopan & Li, 2008)

Os resultados práticos demonstraram que os utilizadores com apenas pedidos de páginas, com elevado número de pedidos “Seguinte” e com múltiplos cliques em resultados de pesquisa ou anúncios tendem a ser considerados atípicos.

Borges e Levene (2008) propuseram uma solução para prever o próximo passo (i.e. próxima acção ou evento) de um utilizador. Conscientes da limitação subjacente à propriedade markoviana e do impacto que esta criaria na sua solução, recorreram as cadeias de *markov* de ordem n . O método de predição utilizado foi o *maximum likelihood* (máxima verosimilhança), i.e. o caminho com maior probabilidade em cada instante.

A figura 2.6 ilustra que para o mesmo conjunto de dados as cadeias de *markov* de ordem superior obtém melhores resultados, descrevendo melhor o comportamento dos utilizadores. Considerando o caso onde se pretende prever a probabilidade condicionada $p(A_3 | A_1, A_2)$, a cadeia de primeira e segunda ordem devolve – respectivamente - a probabilidade de 0.38 ($6/16$) e de 0.45 ($5/11$), sendo a probabilidade correcta 0.5 ($3/6$). O modo como as cadeias de *markov* são transformadas é descrito em Borges e Levene (2005).

Apesar de a precisão das cadeias de *markov* de ordem n serem inversamente proporcionais ao seu tamanho e ao seu tempo de construção, os resultados práticos demonstraram que para um n relativamente baixo a precisão da solução pode-se aproximar de 1. Para tal, é necessário duas premissas: os acontecimentos raros serem classificados de imediato como inesperados não



sofrendo previsão uma vez que tendem a dar origem a erros e, por outro lado, o conjunto de dados de treino tem de ser suficientemente representativo.

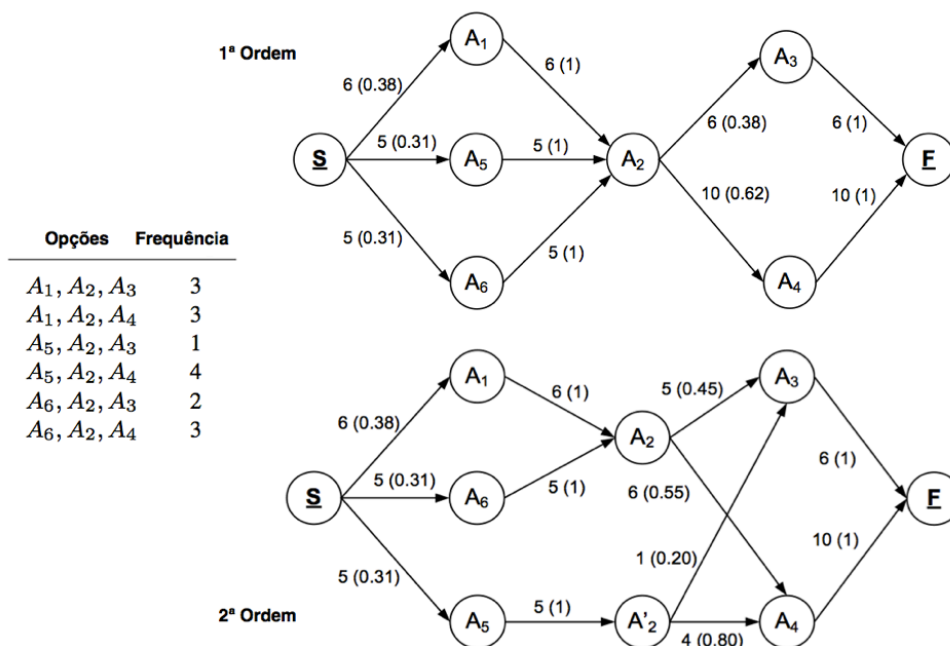


Figura 2.6 - Representação de perfis de navegação usando cadeias de *markov* de ordem N (Borges & Levene, 2008)

Outra das abordagens utilizada são as técnicas baseadas em regras de associação, onde as regras com elevada confiança descrevem comportamentos normais e usuais. A ordem pela qual os eventos se sucedem ou a distância temporal que os separa são normalmente fundamentais para descrever os comportamentos dos utilizadores. Considerando apenas a ordem dos eventos estamos perante padrões sequenciais. Se considerarmos a diferença temporal que separa os eventos estamos perante episódios frequentes.

Os episódios frequentes, introduzidas pela primeira vez por Mannila, Toivonen e Verkamo (1997), possuem duas implementações base. A primeira, denominada de *winepi*, utiliza uma janela deslizante onde um episódio é frequente quando a sua frequência é superior a um dado limite. A segunda, denominada de *minepi*, utiliza apenas um número mínimo de ocorrências. A sua aplicação foi testada na identificação de correlação entre alarmes, i.e. se ocorrer um alarme do tipo A e um do tipo B com 5 segundos de diferença, então um alarme do tipo E irá ocorrer com uma probabilidade de 0.7 (Mannila, Toivonen, & Verkamo, 1999). Este tipo de regras foi igualmente aplicado por Laxman, Sastry e Unnikrishnan (2004) no sentido de encontrar associações entre falhas de uma linha de montagem.



Existem ainda variantes destas técnicas, nomeadamente os *unbounded episodes*, que se propõem a eliminar a limitação da janela deslizante ao definir automaticamente o seu tamanho com base no número de episódios e na sua distância (Casas-Garriga, 2003).

A detecção de comportamentos atípicos pode ainda ser alcançada por técnicas de *clustering* que detectam comportamentos desviantes. Neste âmbito, os estudos recentes têm-se debruçado sobre soluções que permitam evitar as limitações comuns dos algoritmos de *clustering*, tais como a necessidade de definir parâmetros de entrada e a suposição de que os dados seguem distribuições uniformes ou gaussianas. Com base nestas limitações, Böhm, Haegler e Müller (2009) implementaram uma solução para *clustering* que não requisita quaisquer parâmetros de entrada, definindo um comportamento desviante como sendo um objecto que não pode ser eficientemente encaixado em nenhuma das funções de distribuição pertencente aos objectos vizinhos. No entanto, a principal contribuição é a utilização de uma família de distribuições denominadas de EPD (*Exponential Power Distribution*) para obter uma melhor representação dos objectos (Figura 2.7).

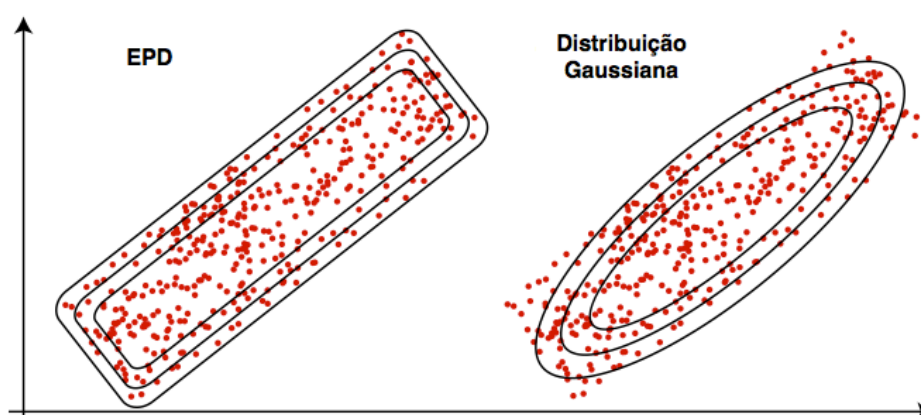


Figura 2.7 - Representação de dados por distribuição Gaussiana e por EPD (Böhm *et al.*, 2009)

O custo de incorporar um dado objecto na distribuição que representa os objectos vizinhos é calculado com base no princípio de compressão de informação MDL - *Minimum Description Length*. A aplicação da solução foi testada com um conjunto de dados referente ao comportamento de jogadores de basquetebol (e.g. assistências, tempo de jogo, pontos, jogos realizados), demonstrando-se mais eficiente na detecção de comportamentos atípicos.



2.2. Detecção de fraude

A quantidade e diversidade dos esquemas fraudulentos nos modelos PPC tem crescido de forma contínua nos últimos anos. O expoente máximo destes ataques é a inflação do número de cliques de forma coligada e sem conhecimento dos próprios utilizadores. No entanto, não existe uma concordância global sobre o melhor mecanismo para a detecção de fraude. Deste modo, as empresas que implementam modelos PPC utilizam soluções próprias que dificilmente são divulgadas, enquanto a comunidade científica continua a propor as mais variadas soluções para a problemática em análise.

Na secção anterior foi abordado o modo como as técnicas de mineração de dados são aplicadas em contexto *web*, nomeadamente para a construção de perfis de utilizador e de perfis de navegação. Nesta secção iremos rever estudos que, usando ou assumindo o conhecimento presente nesses mesmos perfis, implementaram soluções que visam objectivamente a detecção dos três tipos de fraude conhecidos.

2.2.1. Fraude de Tipo I

Tal como visto anteriormente, a fraude de tipo I refere-se ao caso em que o anunciante declara um número de utilizadores redireccionados inferior ao real. No entanto, prevê-se que a sua aplicação nos modelos PPC se tenha extinto quando a entidade que gere os pagamentos se alterou. Actualmente é o editor o responsável por debitar automaticamente o anunciante - ao invés de ser o anunciante a creditar o editor - provocando a existência de fraude de tipo II e III. Assim, a tentativa de calcular de forma precisa o número de utilizadores que são redireccionados no modelo PPC iniciou-se ainda nos finais da década de 90.

Reiter, Anupam e Mayer (1998) propuseram o uso da opção *redirect* disponível no protocolo HTTP ou o uso da opção *onload* disponível na linguagem *Javascript* para aproximar o número n de utilizadores redireccionados, sem a necessidade de interacção do anunciante ou do utilizador. A análise da solução requer o devido enquadramento com a tecnologia existente à época, considerando como ponto de partida o redireccionamento de utilizadores sem qualquer medição de n (Figura 2.8 - Esquerda).



A primeira proposta baseia-se no *redirect* e é utilizada numa página secundária do editor, criada para o efeito, com o objectivo de reencaminhar o utilizador para o anunciante (Figura 2.8 - Centro). Assim, um utilizador *U* começa por carregar a página *pagA.html* do editor *A* que contém o anúncio. Quando o utilizador executa o clique não é reencaminhado para o anunciante *B* mas sim para a página secundária *sec.html* que, através de um código de estado (e.g. 301 para indicar que a página foi movida permanentemente) irá redirecciona-lo para a página *pagB.html* do anunciante *B*. Só nesta altura o utilizador irá visualizar o anúncio. No entanto, a solução apresenta uma limitação óbvia: *n* indica-nos apenas o limite superior dos redireccionamentos, uma vez que a página secundária é carregada antes do *redirect*, não sendo sensível aos casos em que o utilizador não chega a visualizar o anúncio (e.g. problemas de conexão ou desistência do utilizador).

De modo a lidar com esta limitação, é apresentada a proposta baseada na opção *onload* que apenas notifica o editor depois de o utilizador visualizar a página do anunciante (Figura 2.8 - Direita). Nesta abordagem, o utilizador *U* começa por carregar a página *pagA.html* do editor que contém o anúncio e, após o clique, é reencaminhado numa nova janela para a página do anunciante *pagB.html*. Só nesta altura, a página do anunciante irá notificar através de uma CGI (*Common Gateway Interface*) a página do editor. A limitação nesta abordagem é contrária à anterior: *n* indica-nos apenas o limite inferior dos redireccionamentos (e.g. o utilizador fecha a página do editor *pagA.html* antes desta receber a notificação).

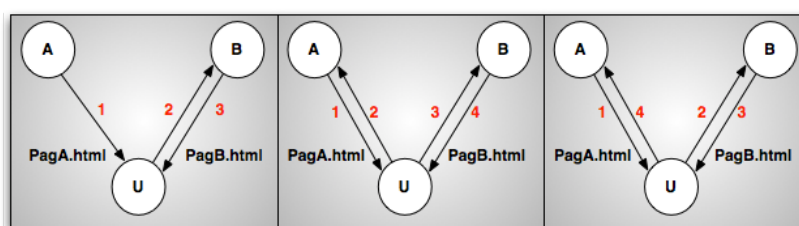


Figura 2.8 - Obtenção do número de utilizadores reencaminhados com base nos métodos *redirect (HTML)* e *onload (Javascript)* (Reiter, Anupam, & Mayer, 1998)

Sem resultados experimentais, Reiter *et al.* (1998) concluem que a aplicação em simultâneo do *redirect* e do *onload* tende a obter um resultado aceitável e assumem que as questões de fraude continuam abertas. No entanto, de forma visionária, deixam antever o uso de assinaturas digitais ou de cadeias de *hash* para a resolução desta problemática.



Seria já mais tarde, na tentativa de medir com precisão o número de utilizadores que visita determinado *site*, que Blundo & Cimato (2002) propuseram uma solução baseada em primitivas criptográficas onde se incluiu as cadeias de hash mencionadas por Reiter *et al.* (1998). Nesse sentido, são considerados três intervenientes: utilizador U que visita os anúncios, servidor S que disponibiliza os anúncios e uma agência A de auditoria. A solução arquitectada baseia-se em três fases distintas: inicialização, interacção e verificação.

Na fase de inicialização (Figura 2.9 - Esquerda) o utilizador deve registar-se na agência para requerer uma tuplo composto por: identificador i , a sua função de *hash* h e um valor inicial aleatório ω (i.e. semente para a função de *hash*). Por sua vez, o servidor irá receber apenas o identificador do utilizador.

A fase de interacção (Figura 2.9 - Centro) é referente ao acesso do utilizador ao servidor. Em cada acesso o utilizador cede o tuplo $[i, h_j(\omega)]$, sendo $h_j(\omega)$ a j -ésima aplicação da função de *hash* h sobre a semente ω . O servidor, por sua vez, guarda para cada utilizador um único tuplo $[i, v_h, c]$, onde c é o número de visitas de i que o servidor contabilizou e v_h o último valor de $h_j(\omega)$ cedido por i .

A fase de verificação (Figura 2.9 - Direita) é referente ao pedido de pagamento. O servidor deverá ceder o tuplo referente a um utilizador i e aguardar a verificação da agência. O tuplo será validado se $v_h = h_c(\omega)$, sendo ω e h , respectivamente, a semente e função de *hash* do utilizador i .

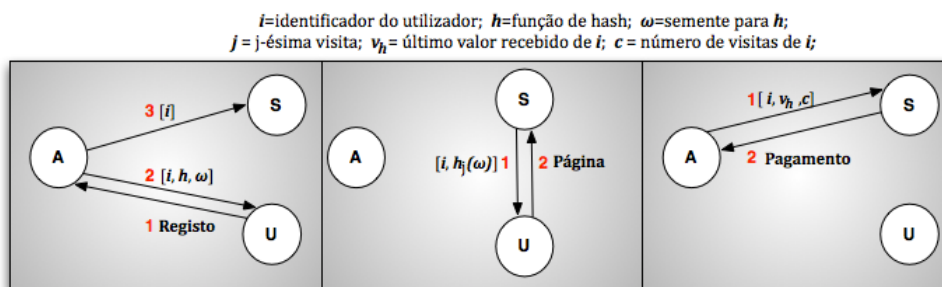


Figura 2.9 - Obtenção do número de utilizadores reencaminhados com base em cadeias de *hash* (Blundo & Cimato, 2002)

Recentemente, Majumdar, Kulkarni e Ravishankar (2007) aplicaram uma abordagem semelhante no contexto dos sistemas de distribuição de conteúdos. No entanto, ao invés de



cadeias de *hash*, foram utilizadas assinaturas digitais e *bloom filters* (veremos a sua definição no âmbito da fraude de tipo II).

2.2.2. Fraude de Tipo II

Devido aos custos, suporte e conhecimento necessário para conceber ataques sofisticados é expectável que a maioria seja simplista (Metwally, Agrawal, & El Abbadi, 2007), seguindo uma distribuição *Zipf* - ou distribuição *zeta* - onde a frequência de um clique fraudulento é inversamente proporcional à complexidade necessária para o gerar (Figura 2.10). A fraude de tipo II, inflação de cliques de forma não coligada, é a mais simples de implementar e consequentemente tornou-se a mais comum nos modelos PPC.

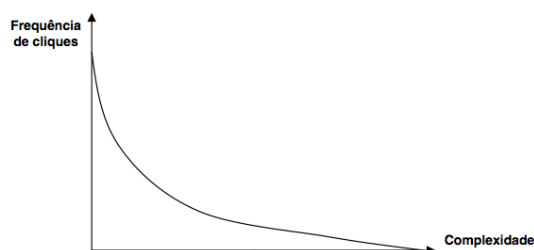


Figura 2.10 - Distribuição *Zipf*: Frequência de cliques perante a complexidade do ataque (Tuzhilin, 2006)

O seu impacto nos editores e anunciantes foi já amplamente examinado, tanto para níveis de fraude determinísticos como para níveis de fraude estocásticos. Wilbur e Zhu (2009) demonstraram que quando os anunciantes conhecem esse nível podem adaptar o valor da oferta por clique até ao montante em que o impacto de fraude é nulo na sua despesa. Assim, para um nível de fraude de $\lambda\%$ o anunciante pagará em média mais $\lambda\%$ por cada clique, impondo-se uma redução em igual proporção no valor da oferta por clique. No entanto, em contexto real os valores de fraude não são totalmente conhecidos e o anunciante não sabe exactamente como reagir.

Nestes casos, o editor irá tradicionalmente lucrar mais com a presença de fraude quando um anúncio é pouco competitivo (e.g. palavras-chave incomuns) e terá menos lucro no caso contrário. Do ponto de vista do anunciante, considerando um anúncio com pouca concorrência, as posições cimeiras tendem a ser baratas e extremamente rentáveis. Assim, o anunciante tende a não limitar a sua oferta por clique, independentemente do nível de fraude, uma vez que o valor despendido eventualmente em fraude é relativamente baixo. No caso de o anúncio possuir



extrema concorrência, as posições cimeiras tendem a ser custosas e com rentabilidade relativa levando o anunciante a tentar reduzir a oferta por clique para controlar o valor desembolsado em casos fraudulentos.

Wilbur e Zhu (2009) concluíram desta forma que os editores podem potencializar o seu lucro detectando fraude apenas nos casos de concorrência elevada. Para que a veracidade do modelo PPC seja mantido, é defendido o uso de uma entidade externa e independente para realizar auditoria às técnicas de detecção de fraude dos editores. Outras abordagens similares foram aplicadas, tendo obtido conclusões equiparadas (Chen, Jacob, Radhakrishnan, & Ryu, 2012).

Uma abordagem alternativa é a criação de entidades que forneçam credenciais devidamente encriptadas aos utilizadores que aparentem ter comportamentos isento de suspeitas, tais como compras de produtos ou qualquer outra acção de conversão (Juels, Stamm, & Jakobsson, 2007). Estas entidades são denominadas de certificadores e as credenciais de cupões. Assim, os cliques provenientes de utilizadores que possuam cupões seriam imediatamente considerados válidos. Para todos os outros cliques podem ser aplicadas técnicas tradicionais, i.e. que procuram existência de padrões fraudulentos, ou simplesmente serem considerados inválidos. Esta abordagem é denominada pelos autores de “abordagem positiva”.

Os certificadores são *sites* que podem fazer parte da rede de anúncios ou serem externos à mesma, desde que consigam atestar o comportamento positivo do utilizador. O exemplo prático apontado pelos autores é um *site* de retalho que emite cupões aos utilizadores que gastem mais do que um determinado valor em compras. O pagamento aos certificadores é realizado de forma fixa ou mediante o número de cupões emitido, sendo apontadas técnicas de auditoria para controlar as entidades desonestas e um novo conceito de fraude.

A atribuição de cupões e o modo como estes são acedidos é tido como o possível ponto de estrangulamento da proposta. A instanciação de cupões por *cookies* é uma solução vulnerável devido às tradicionais exclusões ou bloqueios. Por outro lado, o redireccionamento para os fornecedores de conteúdos com URLs personalizados torna-se inviável com o crescimento abusivo de ligações entre estes e os certificadores. Assim, surge a proposta para uso de *cache cookies* (Juels, Jakobsson, & Jagatic, 2006), i.e. *cookies* que não são bloqueados normalmente e que tem as mesmas características dos *third-party cookies*.

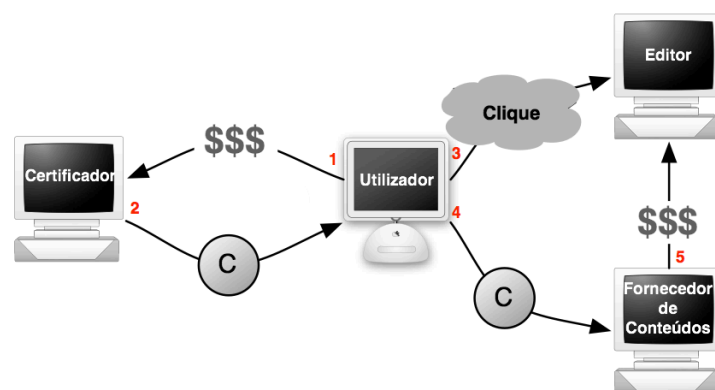


Figura 2.11 - Solução baseada em cupões que atestam comportamento positivo dos utilizadores (Juels *et al.*, 2007)

Em termos práticos um utilizador, sempre que execute um clique no editor, será reencaminhado para o fornecedor de conteúdo (e posteriormente para o anunciante) onde apresentará os cupões que possui. Se os mesmos forem validos, o fornecedor debita o anunciante e credita o editor (Figura 2.11). Infelizmente, embora não assumido pelos autores, a utilização excessiva e abusiva de cupões poderá conduzir-nos a novos tipos de fraude.

Até agora os estudos referenciados abordam a problemática com propostas que alteram o modo como o modelo PPC funciona ou que necessitam de mais intervenientes (e.g. entidades de auditoria). Tal situação pode não ser exequível em termos reais, pelo que é necessário propostas que não coloquem esses obstáculos.

Haddadi (2010) sugere a impressão de anúncios reais de reduzido interesse para um utilizador como forma de medir a aleatoriedade de um clique ou da navegação, técnica denominada de *bluff ads*. Como descrito anteriormente, os modelos de publicidade *online* tem por objectivo associar a cada cliente os melhores anúncios, tendo por base os seus perfis. Para que a implementação não seja reconhecida pelos utilizadores é assumido que os anunciantes dos *bluff ads* não serão debitados e que a experiência do utilizador ou a qualidade do editor não é colocada em causa. Assim, a título de exemplo, não será racional realizar impressões de anúncios relativos a bebidas ou comidas em tempo de Ramadão para um utilizador proveniente de um país islâmico. Em contraste, a impressão moderada de anúncios relativos a automóveis não irá lesar as partes e será válida desde que o utilizador, na sua informação de perfil, não possua nenhuma preferência por automóveis.



Nesta abordagem a suspeita de fraude é sustentada com base no rácio entre cliques em anúncios normais e em *bluff ads*. A incorporação de *bluff ads* durante 3 semanas na rede de anúncios da *Google* demonstrou um desinteresse geral dos utilizadores pelos mesmos. Reitera-se, desta forma, a ideia de que um utilizador ou um *bot* que navegue de forma aleatória e sem interesse irá com alguma probabilidade visualizar os *bluff ads*, tornando-se suspeito.

A análise de tráfego é outra abordagem que contribui de forma significativa para a detecção e classificação de comportamentos desviantes nos modelos PPC. Kantardzic, Wenerstrom e Walgampaya (2009) desenvolveram uma solução baseada em histogramas para detectar esses desvios através da frequência absoluta de atributos pré-seleccionados e a sua distância para o esperado. É defendido o uso de múltiplos atributos (i.e. variáveis de análise) provenientes do lado do cliente e do servidor - de modo a aumentar a qualidade geral da utilização - e uma classificação quantitativa para o clique em vez de qualitativa. Esta classificação é baseada num contexto de espaço uma vez que os atributos são analisados individualmente e num contexto temporal pois dependem do histórico dos utilizadores. Se, por exemplo, o *referrer* de um clique é o *www.google.com* diremos que o atributo (representado por um histograma) é o *referrer* e o valor é o *www.google.com*, construindo um par atributo-valor. Deste modo, para cada atributo, o comportamento normal é definido com base no histórico da frequência absoluta do atributo-valor (Figura 2.12 – Esquerda) e o comportamento actual é definido com base na frequência absoluta do atributo-valor no período em análise (Figura 2.12 – Direita).

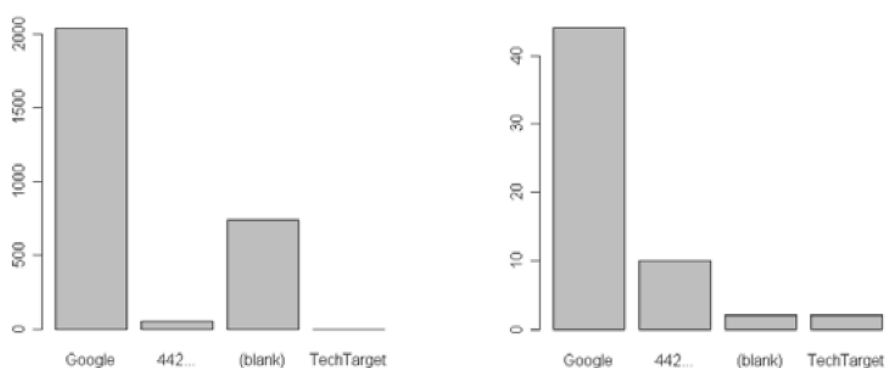


Figura 2.12 - Uso de histogramas para medir o desvio para o comportamento actual (direita) e do comportamento esperado (esquerda) (Kantardzic *et al.*, 2009)

Assumindo que a probabilidade de um dado par atributo-valor λ segue uma distribuição normal com média μ e desvio padrão σ , um clique é definido como anormal sempre que a frequência absoluta de λ ultrapassar o limite pré-estabelecido de $(\mu + 1,645 * \sigma) * \omega$. De notar que



$(\mu + 1,645 * \sigma)$ representa o início da zona de rejeição num teste direccional para uma distribuição normal e com um grau de certeza de 95%, enquanto ω representa a frequência absoluta do atributo-valor em termos históricos. Na figura 2.13 encontra-se ilustrado uma representação do valor limite (cinza escuro) e do valor actual (cinza claro) para o atributo *referrer*.

A pontuação é definida em função do rácio entre o excesso e o total de cliques, estando limitado a uma valor máximo de 0.2. Os resultados experimentais demonstraram a detecção de algumas situações anormais, nomeadamente alguns *referrers* até então desconhecidos.

No entanto, mais do que a detecção, é necessário prevenir e validar as situações de fraude nos modelos de publicidade *online*. Infelizmente, o volume de dados e a carga computacional apresentam-se como características da generalidade das soluções desenvolvidas não permitindo a sua aplicação em tempo real.

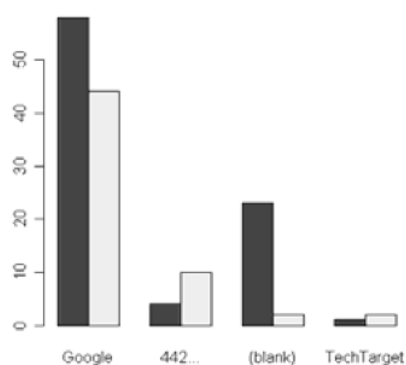


Figura 2.13 - Representação da frequência absoluta de um atributo-valor e do limite máximo aceitável (Kantardzic *et al.*, 2009)

Na sequência do estudo anterior, Walgampaya, Kantardzic e Yamolskiy (2010) apresentaram uma solução para executar em tempo real denominada de CCFDP (*Collaborative Click Fraud Detection and Prevention*). Os mecanismos de prevenção visam o bloqueio instantâneo de tráfego suspeito, defendendo os anunciantes e a veracidade do modelo PPC. Para tal, o clique é quantificado segundo a evidência de fraude apresentada e sempre que obtiver valores acima do limite pré-estabelecido pelos histogramas serão retiradas conclusões e aplicadas medidas (e.g. bloqueio do *IP* ou do *referrer*, emissão de alerta para o editor, etc). A solução desenvolvida é constituída por três componentes principais: recolha e fusão de dados, classificação de tráfego e monitorização.



A recolha de dados é realizada de forma assíncrona e implícita do lado do cliente e do lado do servidor, garantindo as informações relevantes ao nível da máquina e do utilizador. Adicionalmente, num processo denominado de fusão de informação, estes dados são ainda enriquecidos com informações como o *referrer*, local de origem do utilizador e *timezone* (Kantardzic M. , Walgampaya, Wenerstrom, Lozitskiy, Higgins, & King, 2008).

O componente responsável pela classificação de tráfego tem como objectivo atribuir uma pontuação a cada clique em análise, variando entre 0 (sem evidência de fraude) e 1 (claramente fraudulento). Para esse fim foram desenvolvidos três módulos baseados em: regras, detecção de comportamentos anómalos e análise de navegação. As regras visam a análise de eventos que tenham ocorrido numa janela temporal curta (e.g. minutos). Por sua vez, a análise de eventos com elevada janela temporal (e.g. dias) é realizado pelo segundo módulo. O último módulo analisa o deslocamento e posição final do rato em relação ao anúncio. Assim, para cada clique, cada um dos módulos analisa os dados disponíveis e emite a sua apreciação final sobre a evidência de fraude. Tendo em conta as três pontuações disponibilizadas será calculado o grau de credibilidade para o qual se acredita que o clique seja suspeito, utilizando a teoria de *Dempster-Shafer* (Wu, Siegel, Yang, & Stiefelhagen, 2002). A arquitectura desta solução é apresentada na figura 2.14.

Walgampaya *et al.* (2010) assumiram que um clique com pontuação superior a 0.9 deve ser considerado fraudulento. Nestes termos, os resultados experimentais revelam que 64% do tráfego analisado foi classificado como fraudulento. No entanto, se restringirmos a análise apenas ao primeiro e segundo módulo a percentagem de tráfego classificado como fraudulento reduz, respectivamente, para 53% e 35% demonstrando a eficiência dos módulos combinados pela teoria *Dempster-Shafer*.

Ao nível da prevenção de fraude deve-se referir que 4% do tráfego fraudulento foi bloqueado em tempo real. A solução desenvolvida foi ainda capaz de evidenciar algumas associações interessantes, entre elas:

- Relação *referrer* e *javascript*, i.e. se parte significativa dos utilizadores provenientes de um *referrer* φ possuírem o *javascript* desactivado então φ é suspeito;
- Origens geográficas suspeitas, tais como Turquia, Gana e Vietnam.

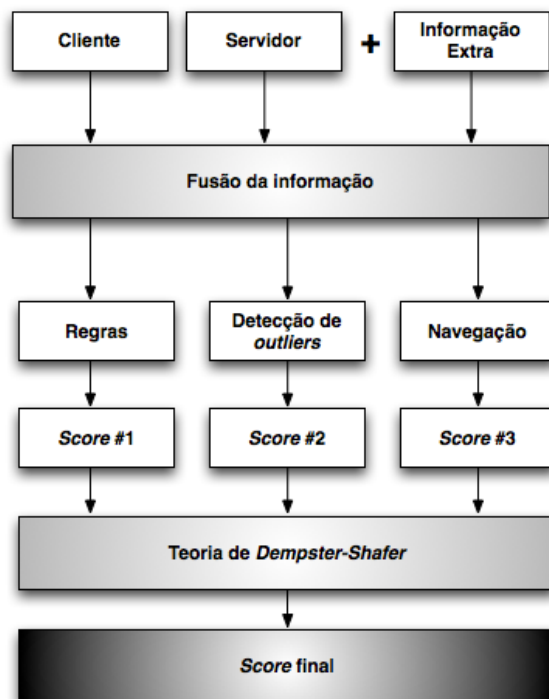


Figura 2.14 - Proposta de Walgampaya, Kantardzic e Yamolskiy (2010) para detecção e prevenção de fraude em tempo real

A detecção de duplicados em fluxo de dados é outra das técnicas utilizadas na detecção de fraude. A ocorrência de dois cliques no mesmo anúncio num reduzido espaço temporal é sinónimo de clique inválido. Por outro lado, a presença de vários duplicados para um mesmo utilizador ou anúncio é uma suspeita de fraude bastante fundamentada. Nestes casos deve-se proceder a uma análise mais detalhada das actividades realizadas pelo utilizador em causa. De salientar, que a procura de duplicados é semelhante à procura de elementos frequentes. No entanto, o contexto de aplicação das duas técnicas é distinto: na procura de duplicados assume-se que a diversidade dos elementos é elevada e a sua frequência reduzida, sendo o contrário na procura de elementos frequentes.

A procura de duplicados, dependendo do modo como se pretende analisar o fluxo de dados, tem três variantes: *landmark window* que considera todos os elementos desde de um determinado ponto referência, *sliding window* que considera os últimos n elementos e *jumping window* que é um compromisso entre as duas anteriores.

Metwally, Agrawal, & El Abbadi (2005) desenvolveram uma solução para a detecção de cliques duplicados baseada em *bloom filters* (Knuth, 1998). *Bloom filters* são estruturas de dados que conseguem rapidamente e sem grande consumo de memória indicar se um elemento está



presente ou ausente num determinado conjunto. A eficiência temporal e espacial da estrutura é inversamente proporcional à certeza da resposta tornando-a numa estrutura de dados probabilística sem falsos negativos mas com possibilidade de ocorrência de falsos positivos. A estrutura original é suportada por um *array* de *bits* A com tamanho M (inicializado a zero) e H funções de *hash*. Para cada elemento E do conjunto C , preenche-se com o valor 1 os indexes de A que foram devolvidos pelas funções de *hash*. Se não ocorrer nenhuma alteração de *bit* em todos os indexes considerados, então E é considerado duplicado. A figura 2.15 demonstra o processo de preenchimento, bem como os falsos positivos (elemento S) e os duplicados efectivamente detectados (elemento X).

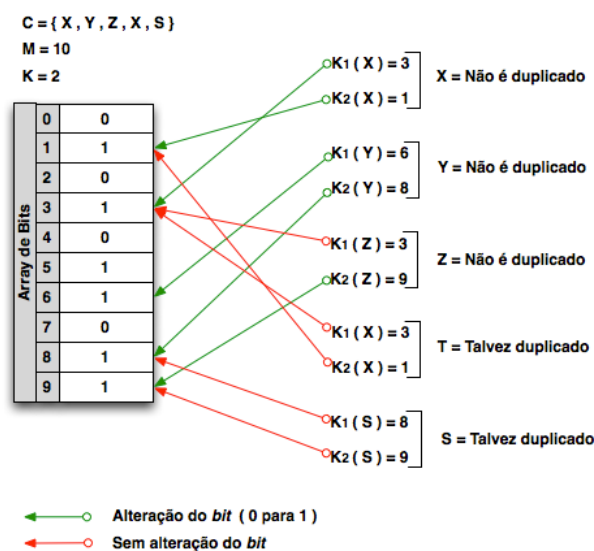


Figura 2.15 - Estrutura *Bloom Filter* e respectivo preenchimento (Knuth, 1998)

Para permitir a detecção de cliques duplicados Metwally *et al.* (2005) alteraram a estrutura original, usando um *array* de *bits* por cada função de hash. Esta alteração resulta num aumento do paralelismo do servidor que executa as operações uma vez que a latência da memória diminui. Para testar a precisão da detecção de cliques duplicados baseado em *bloom filters* foram testadas as variantes *landmark window* e *jumping window*, em dados sintéticos e reais, para diferentes quantidades de funções de hash.

Na *sliding window*, para o cálculo da taxa de erro, é necessário medir o número exacto de duplicados para cada janela e compara-lo com a abordagem aproximada pelos *bloom filters*. Tal operação foi considerada inviável uma vez que o número de janelas tende a ser proibitivo. A *landmark window* obteve taxas de erro de quatro a oito vezes inferior ao erro teórico, situando-se entre os 0.01% e os 1.55% quer em dados sintéticos, quer em dados reais. Por outro lado, a



jumping window apresentou taxas de erro entre 0.01% e 1.56% para dados sintéticos (i.e. tal como a *landmark window*) e entre 0.46% e 5% em dados reais (i.e. duas a quatro vezes pior que o erro teórico). De salientar que o elemento mais duplicado ocorreu 10.781 vezes num dia, usando uma técnica primitiva para a geração de cliques.

Todas as técnicas até aqui descritas são, em alguns pontos, similares às que algumas empresas multinacionais aplicam para proteger os seus serviços de publicidade. Apesar da confidencialidade por detrás das soluções desenvolvidas, Tuzhilin (2006) divulgou algumas informações sobre o modo como a *Google* desenvolveu, evoluiu e mantém a sua rede de publicidade (*AdWords* e *AdSense*) e, fundamentalmente, um resumo das técnicas utilizadas para detectar fraude nos modelos PPC.

Os relatórios facultados pela *Google* aos seus anunciantes e editores são diários e sumariados, evitando que os mesmos possam realizar uma análise individual dos cliques. É utilizado o termo *inválido* - ao invés do termo *fraudulento* - e a definição de fraude assume um ponto fulcral: a validação do clique deve ter em conta se o utilizador demonstra intenção em converter o clique (e.g. compra) ou se o fez apenas para lucrar e prejudicar terceiros. No entanto, esta decisão envolve comportamentos humanos que são complexos de modelar tornando a validação do clique complicada e não determinística.

Segundo o autor, são utilizadas essencialmente soluções baseadas em técnicas de mineração de dados para identificar os cliques inválidos: regras de associação, detecção de anomalias e classificação. As regras são especificadas com base no conhecimento adquirido e têm que ser conceptualmente correctas, i.e. se uma regra classifica um clique como inválido devido a determinadas características então não existe qualquer probabilidade de um outro clique com essas mesmas características ser considerado válido. Estas regras são analisadas e validadas apenas por engenheiros, uma vez que os gestores são influenciados pela implicação monetária que estas representam. A detecção de anomalias visa encontrar as actividades infrequentes e com um desvio estatístico significativo em relação ao normal. A principal dificuldade desta abordagem é a definição correcta das actividades normais. Por último, os modelos de classificação utilizam conhecimento resultante de dados históricos para classificar novos cliques. Nesta técnica a principal restrição é a assunção de que o comportamento passado é indício do



comportamento futuro e a necessidade de pré-identificar casos em que existe evidência de fraude para que o modelo possa ser construído.

O processo de detecção de cliques inválidos é realizado em quatro etapas distintas (Figura 2.16). A primeira etapa é denominada de pré-filtragem, onde são removidos os cliques de teste ou inconsistências relacionados com problemas de comunicação. A segunda etapa é a análise *online*, onde são aplicadas as técnicas acima enunciadas. Segundo o autor, as novas técnicas desenvolvidas pela Google para esta etapa tem demonstrado baixa melhoria, o que tende a provar que as existentes já são suficientemente capazes. A terceira etapa refere-se à análise *offline*, útil para detectar padrões mais complexos e subtis que evidenciem igualmente fraude (e.g. utilização de múltiplos IPs). Por último, a detecção manual que é realizada por funcionários e que visa a análise do comportamento de editores e anunciantes ou para análise de queixas de fraude.

Em conclusão, Tuzhilin (2006) assume que o sistema de filtragem *online* utilizado pela *Google* resulta pela combinação de diferentes tipos de filtros e pela simplicidade da maior parte dos ataques. Por outro lado, a generalidade dos anunciantes e editores bloqueados por gerar cliques inválidos não apresentam queixa ou contra-provas, o que simboliza que o sistema está a detectar situações ilícitas de forma assertiva.

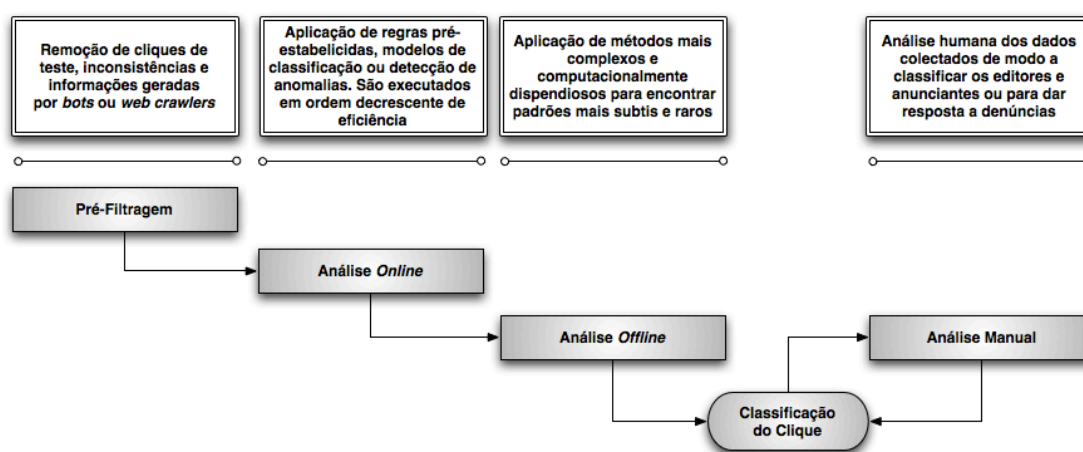


Figura 2.16 - O processo de detecção de cliques inválidos implementado pela *Google* (Tuzhilin, 2006)

Por análise de patentes registadas é igualmente possível reconhecer alguns métodos utilizados por outros gigantes do sector da publicidade *online*. Quer a *Microsoft* (Immorlica, Jain, Mahdian, & Talwar, 2007) como a *Yahoo* (Sadagopan & Li, 2010) possuem soluções que consideram o



comportamento anterior a um clique (e.g. eventos ou acções) para calcular a probabilidade de fraude nos seus modelos PPC.

Existe ainda técnicas que foram aplicadas em áreas ou contextos similares e que podem ser facilmente transportadas para a detecção de fraude em sistemas PPC. Krause, Schmitz, Hotho e Stumme (2008) testaram a aplicação de técnicas de aprendizagem automática para detecção de *spam* em sistemas de partilha de *sites online* (denominado de *social bookmarks*). O passo primário foi a definição de *spam*, que no contexto do estudo, é todo o conteúdo sem interesse de partilha ou mal identificado (e.g. palavras-chave mal associadas). De seguida, apesar de ser um processo extremamente subjectivo tal e qual nos sistemas PPC, foram identificados todos os casos possíveis de *spam* no conjunto de dados em análise. Desta forma, o processo de aprendizagem tornou-se supervisionado e foram utilizados algoritmos de classificação, nomeadamente SVM, *Naive Bayes* e J48. A escolha de atributos foi realizada e categorizada de forma distinta: informações individuais (e.g. nome do utilizador), email (e.g. domínio de *email*), interacção com o sistema (e.g. tempo entre o registo e a primeira actividade) e o uso de palavras-chave (e.g. número média de palavras utilizadas). Como medidas de precisão utilizou-se a medida-F (equilíbrio harmónico entre a precisão e *recall*) e as curvas ROC (balanço entre os custos e os benefícios).

Os resultados práticos demonstraram que o uso de SVMs garante a melhor precisão de entre os algoritmos testados. Foi igualmente possível identificar padrões comuns nas situações suspeitas, tais como: são utilizados demasiados números e caracteres especiais no nome escolhido pelos utilizadores, os domínios de *email* são claramente desmarcados (i.e. altos níveis de *spam* ou ausência) e o tempo de espera entre o registo e a primeira actividade é mais elevada nos *spammers*, presumivelmente para não levantar suspeitas.

2.2.3. Fraude de Tipo III

Tal como visto, a coligação de utilizadores que pretendem cometer fraude em sistemas de publicidade *online* apresenta-se como o caso mais sofisticado e complexo de detectar, uma vez que o tráfego fraudulento é partilhado por mais que uma máquina. Este tipo de coligação pode ser realizado de forma explícita, onde os utilizadores conhecem o esquema fraudulento e realizam cliques deliberadamente, ou de forma implícita, onde os cliques ou a requisição de anúncios é realizada sem consentimento dos utilizadores. Está última situação é cada vez mais



usual com, por exemplo, a abertura de publicidade em janelas secundárias sem que tenha sido o utilizador a realizar o pedido de página. Independentemente de ser coligação implícita ou explícita, a resolução deste tipo de fraude reside em identificar associações ou padrões de acesso comuns a um conjunto de utilizadores.

Metwally, Agrawal e El Abbadi (2005) foram os únicos autores a abordar esta problemática de forma específica e com duas soluções distintas. Primeiramente, desenvolveram uma nova abordagem baseada em regras de associação para encontrar relações entre pares de elementos. Na formalização do problema são enumeradas as seguintes suposições:

- Todos os pedidos são recebidos num único *stream*, evitando a questão da identificação do utilizador;
- Os eventos com alguma associação podem ocorrer de forma não consecutiva e com uma distância máxima pré-definida, permitindo o intercalamento de pedidos resultantes da navegação de cada utilizador e da latência da comunicação;
- Os elementos duplicados são independentes, i.e. a ocorrência de duas visualizações de um determinado anúncio é associado a diferentes utilizadores;
- A existência de um tamanho máximo pré-definido para cada *stream*, onde é suficientemente expectável o aparecimento de associação entre elementos.

O problema é apresentado com duas variantes: *forward association* onde o objectivo é descobrir os elementos que normalmente sucedem-se a outros elementos interessantes ou frequentes e a situação contrária denominada de *backward association* onde o objectivo a descoberta de elementos que precedem outros elementos frequentes. Enquanto a primeira variante tem utilidade para contextos de *cache*, a segunda ganha importância para a detecção de sequência de eventos atípicos, sendo a única com interesse no âmbito deste documento. Assim, é definido o suporte mínimo ϕ do consequente da regra (i.e. elementos frequentes), a confiança mínima ψ da regra e uma distância máxima σ entre o antecedente e o consequente. Assim, para o conjunto de eventos $\{x, x, u, u, c, g, d, c, x, f, x, u\}$, $\phi = 0.2$, $\psi = 0.3$ e $\sigma = 3$ os elementos frequentes (i.e. consequentes) são $\{x, u\}$ e as regras obtidas são:

- $x \rightarrow u$ [$S = 0.25$; $C = 0.33$];
- $f \rightarrow u$ [$S = 0.25$; $C = 0.33$].



Os testes experimentais propuseram-se a medir a escalabilidade e a eficiência da solução num conjunto de dados que incluía cerca de 678 cliques. Em termos de recursos, apesar de corresponder às expectativas dos autores com ótimos rácios entre espaço necessário, tempo de execução e o tamanho do conjunto de eventos, demonstrou ser inviável como solução em tempo real. Quanto à eficiência, apenas uma entidade externa que monitorize todos os pedidos HTML realizados (e.g. ISPs) pode medir a eficácia das regras derivadas pela solução. Apesar desta limitação, foi identificado um caso onde um conjunto de *sites* suspeitos A de frequência reduzida foi sempre requisitado antes de outro conjunto de *sites* B com $\psi \geq 0.5$ e $\sigma \geq 10$.

Metwally, Agrawal e El Abbadi (2007) voltariam a abordar o tema, desta vez com mais sucesso, propondo a resolução da problemática com recurso à análise de tráfego. O objectivo central é a descoberta de coligações, i.e. conjunto de *IPs*, de tamanho arbitrário cuja semelhança exceda um determinado limite pré-estabelecido. Foram utilizadas várias medidas de similaridade como o coeficiente *Dice*, *Jaccard* ou *Cosine* (Charikar, 2002). Neste estudo, a semelhança ω entre duas coligações é o número de editores em comum e o limite τ é o número de editores a partir do qual um *IP* é ignorado. A utilização da variável τ visa a redução de ruído nos dados eliminando, por exemplo, os indexadores automáticos mais relevantes (e.g. *Google* e *Yahoo*).

Os resultados experimentais da solução desenvolvida, denominada de *Similarity-Seeker*, são apresentados em função de diferentes τ e demonstram que 98.94% (ou 99.98%) dos pares possuem um $\omega < 1\%$ para um $\tau = 1000$ (ou $\tau = 10$). Assim, concluiu-se que a similaridade entre *sites* tende a ser negligente, pelo que a sua ocorrência deve ser considerada suspeita. Com um $\tau = 10/\tau = 30/\tau = 40$ e para um $\omega > 10\%$ foram encontrados 81/189/647 pares. O caso mais evidente de fraude foi a detecção de uma coligação de tamanho 29 que partilhava 15 editores ($\omega = 15$) com uma outra coligação de tamanho 22.

Conclui-se, após investigação, que todos os 51 *IPs* identificados possuíam tráfego moderado, localizações espalhadas pelo mundo e um *referrer* nos pedidos HTTP que apontava para páginas que não continham qualquer referência aos anúncios visitados.

2.3. Contributo para a Solução Desenvolvida

Antes de apresentar a solução proposta nesta dissertação analisaremos as propostas até aqui apresentadas, mencionando as suas limitações e contribuições. Não serão abordados os



padrões de navegação uma vez que para a solução desenvolvida foram preteridos em função dos perfis de utilizador. Esta decisão é justificada pelo facto de a generalidade das empresas já possuírem perfis de utilizador, o que não sucede em relação aos perfis de navegação que teriam ainda que ser colectados. Do mesmo modo, a detecção de fraude de tipo I e III não será mencionada por, respectivamente, já não se enquadrar actualmente do modelo PPC e por ser um caso muito particular que não é o âmbito desta dissertação. Assim, analisaremos as restantes com particular foco na detecção de fraude de tipo II.

A tabela 2.1 resume as propostas dos autores que mais contribuíram de forma efectiva para a solução que apresentamos, bem como a secção onde os assuntos voltaram a ser mencionados. Todos os restantes estudos contribuíram essencialmente para o enquadramento do tema e planeamento da solução, não apresentando semelhanças evidentes com a proposta que apresentamos no próximo capítulo.

Para a identificação do utilizador fica patente a falta de consenso por parte da comunidade nas estratégias a utilizar. As divergências resultam da tentativa em encontrar um compromisso aceitável entre a privacidade do utilizador - a principal limitação em todas as propostas - e a sua identificação inequivocamente. As opções mais utilizadas e respeitadas são os *cookies*, a autenticação e as sessões de *browser*. Uma vez que esta problemática não é o nosso objectivo primário, foram utilizadas sessões de *browser* por serem as mais simples de implementar, deixando a proposta de Spiliopoulou *et al.* (2003) para a reconstrução de sessões como uma referência para trabalho futuro.

Tabela 2.1 - Contribuição individual de cada autor para a solução desenvolvida

Secção	Autor	Contribuição
3.2 - Variáveis de Análise	Gentili <i>et al.</i> (2003)	Implementação de redes semânticas para determinar e filtrar os melhores documentos de uma biblioteca <i>online</i> para cada utilizador
	Sadagopan e Li (2008)	Classificação de sessões de utilizadores como típicas ou atípicas em motores de busca que implementassem modelos PPC
	Borges e Levene (2008)	Previsão da próxima acção ou evento de um utilizador com base em cadeias de <i>markov</i> de ordem n
3.6 - Cálculo de <i>p-values</i>	Haddadi (2010)	Impressão de anúncios de baixo interesse para medir a aleatoriedade do clique de um utilizador
3.7 - Armadilhas	Kantardzic <i>et al.</i> (2009) Kantardzic <i>et al.</i> (2010)	Utilização de <i>p-values</i> para medir os desvios significativos nas variáveis em análise (assumindo distribuições gaussianas)



Nos perfis de utilizador a estrutura com vectores de palavras-chave é a que apresenta maiores limitações: polissemia das palavras, dimensões proibitivas e a incerteza de que garanta a real importância de cada termo num perfil. As hierarquias de conceitos apresentam-se como alternativa mas, conseqüentemente, são mais complexas de implementar e não representam uma mais-valia para o contexto em causa. Por outro lado, as redes semânticas contribuíram de forma concreta para a identificação e compreensão de dependências condicionais, i.e. as relações entre os diversos anúncios e os padrões de utilização. Para o mesmo fim contribuíram as propostas que utilizam cadeias de *markov*, sugeridas por Sadagopan e Li (2008) e Borges e Levene (2008).

Da análise às propostas para a detecção de fraude de tipo II resultam duas particularidades comuns, que limitam a classificação final de um utilizador ou dos seus cliques:

- Utilização de apenas duas etiquetas de classificação para cliques ou utilizadores: válido e inválido;
- Maior propensão para análise de variáveis relacionadas com a máquina de acesso e das suas características (e.g. IP ou *browser*), descartando variáveis comportamentais ou que afirmam o interesse geral do utilizador (e.g. coerência na escolha de anúncios).

Na classificação de um clique a transição de *válido* para *inválido* é imediata e sem possibilidade de retorno quando é excedido um limite que representa a normalidade (e.g. quantidade de duplicados ou a taxa de cliques).

Infelizmente esta opção pode acarretar erros grosseiros na análise da real actividade de um utilizador. Façamos um paralelismo com o mundo criminal para descortinar alguns factos interessantes. Tal como num julgamento de um crime, os sistemas PPC padecem da mesma dificuldade e subjectividade em avaliar ou ajuizar acontecimentos. Em ambos os cenários, para condenação, não basta a disposição (ou as suspeitas) de um réu para cometer um crime. É necessário indícios anteriores ou posteriores que nos permitam uma conclusão assertiva (e.g. premeditação do crime ou persistência de acções). É na presença de tais indícios que, normalmente, é realizada prova e respectiva condenação. Também o comportamento das entidades policiais segue o mesmo raciocínio: na presença de suspeitas tendem a não realizar apreensões imediatas mas sim investigação, por vezes emboscadas, que lhes permitam obter uma prova real, testemunhal e irrefutável.



Com esta analogia fica patente que considerar um conjunto de cliques inválidos e fundamentá-lo apenas na anormalidade ou na infrequência pode constituir um aumento de falsos positivos. Um facto anormal ou infrequente deverá ser apenas alvo de suspeita, tendo ainda que haver mais provas para o considerar fraudulento. Um exemplo elucidativo é a rejeição de cliques de utilizadores com um CTR (*click-through rate*) perto dos 100%, sem antes entender as suas motivações. Tal como já indicado, o próprio sistema de recomendação poderá conduzir o utilizador a esse estado uma vez que tende a direccionar-lhe os anúncios com maior probabilidade de visualização.

Apesar destas limitações, transversais a quase todas as propostas, Kantardzic *et al.* (2009) e Kantardzic *et al.* (2010) foram referências para definir o momento em que a solução apresentada transita um utilizador do estado *normal* para *suspeito*. A sua proposta assenta no cálculo de *p-values* para as suas variáveis de análise assumindo distribuições gaussianas (o que constitui outra limitação e que na nossa solução foi ultrapassada).

O segundo ponto refere-se, fundamentalmente, ao descarte de variáveis comportamentais. É evidente que a análise e monitorização da máquina de acesso deve existir, mas sem nunca rejeitar uma análise rigorosa às opções que o utilizador vai tomando. Sem a mesma, a título de exemplo, grande parte dos utilizadores poderá passar incólume à monitorização se não alterar qualquer característica da máquina (e.g. renovação sistemática de IP e *cookies*) e se mantiver o CTR e número de visualizações minimamente reduzido.

Por esse motivo, é imperial levantar uma nova questão em paralelo: “As visualizações de anúncios relacionam-se de alguma forma ou demonstram intenção de conversão?”. A resposta a esta questão é o principal objectivo da nossa solução e, segundo Tuzhilin (2006), da própria solução implementada pela *Google*. Neste sentido, Juels *et al.* (2006) propôs uma abordagem que valida apenas cliques a utilizadores credenciados, i.e. com conversões efectuadas. No entanto, existe limitações evidentes nesta solução: redução inconcebível de lucros (e.g. só uma percentagem reduzida de utilizadores converte os seus cliques), aparecimento de novos tipos de fraude e alteração da estrutura do modelo PPC para suportar as credenciais. Em sentido oposto, a proposta de Haddadi (2010) – impressão de anúncios de reduzido interesse – contribuiu para o modo como avaliamos o interesse e a atenção do utilizador nas suas escolhas, definindo o momento em que transitamos um utilizador do estado *suspeito* para *fraudulento*.



Capítulo

3. Concepção da Solução



3.1. Perspectiva Geral

A solução desenvolvida visa, objectivamente, detectar e validar situações suspeitas de fraude em tempo real. Nesse sentido, a nossa proposta é que o *site* de classificados se altere perante as situações de suspeita, avaliando o comportamento posterior do utilizador. Perante essa avaliação será possível fundamentar com algum nível de precisão a decisão de classificar um utilizador como sendo fraudulento.

Para uma implementação correcta e eficiente é necessário que o motor de publicidade implementado e o seu sistema de recomendação estejam preparados para executar em dois modos distintos: normal e em segurança. O modo normal corresponde ao que se encontra implementado nos motores de publicidade actuais, onde o objectivo é canalizar para cada utilizador os anúncios relacionados ou com maior probabilidade de visualização. No entanto, tal como já mencionado, este modo de operar pode em alguns casos conduzir o próprio utilizador para situações suspeitas (e.g. CTR elevados) e, conseqüentemente, para falsos positivos na detecção de fraude. Para ser conciliável com o nosso objectivo o motor de publicidade deve ser capaz de operar em modo de segurança. Assim, perante suspeitas suspende-se a impressão de anúncios relacionados ou lucrativos para se iniciar um teste ao comportamento do utilizador. Veremos mais tarde como e quando são executados estes testes, denominados de *armadilhas*.

Embora as próximas secções detalhem com rigor a solução, é necessário introduzir de imediato a principal terminologia e o conjunto de premissas utilizadas, bem como uma visão geral sobre a arquitectura concebida. A terminologia visa uniformizar a linguagem empregue ao longo deste documento, clarificando a proposta e o âmbito da mesma. Os termos mais relevantes são:

- **Visualização:** Corresponde ao acto de clicar num anúncio e simboliza o interesse do utilizador no mesmo. Como tal, o termo clique deve ser interpretado de modo semelhante. É a única acção que gera circulação de dinheiro no modelo PPC;
- **Impressão:** Corresponde ao acto de disponibilizar um anúncio na página requisitada. Poderá resultar numa visualização;
- **CTR:** Rácio entre as visualizações e as impressões realizadas pelo utilizador;
- **Utilizador Normal:** Classificação atribuída a um utilizador que não demonstre qualquer indicio de actividade ilícita;



- **Utilizador Suspeito:** Classificação atribuída a um utilizador que demonstre indício de actividade ilícita, mas que não tenha despoletado nenhuma armadilha;
- **Utilizador Fraudulento:** Classificação atribuída a um utilizador em que as suspeitas de fraude tenham sido fundamentadas através das armadilhas;
- **Modo Normal:** Refere-se ao estado do motor de publicidade. Visa a obtenção do maior lucro possível e está activo quando se trata de um utilizador normal;
- **Modo Segurança:** Refere-se ao estado do motor de publicidade. Visa testar o comportamento do utilizador e está activo quando se trata de um utilizador suspeito;
- **Armadilha:** Cenários previamente concebidos que visam a confirmação das suspeitas de fraude pela análise do comportamento e reacção do utilizador. Só aplicáveis quando o motor de publicidade está em modo segurança, i.e. estamos na presença de um utilizador suspeito;
- **Armadilha despoletada:** Quando um utilizador suspeito mantém o seu comportamento mesmo perante cenários adversos (i.e. as armadilhas). Uma vez despoletada, um utilizador transita do estado suspeito para o estado fraudulento.
- **Variáveis de Análise:** Variáveis da solução desenvolvida que actualizam a cada clique do utilizador e que representam o seu comportamento;
- **Transição:** Acção que decorre da visualização de novo anúncio. Uma visualização de um anúncio com característica x seguido de um anúncio de característica y resulta numa transição $x \rightarrow y$;
- **Pontuação:** Valor atribuído a cada variável de análise. Está compreendida entre 0 e 1.

Nos termos apresentados, salientam-se as armadilhas, as variáveis de análise e as diferentes classificações do utilizador que terão um papel fundamental na compreensão da solução. Por sua vez, as premissas apoiam na tarefa de definir uma solução e de não desviar o foco do objectivo para temáticas que, embora relacionadas e essenciais no contexto PPC, não serão abordadas neste documento. O conjunto de premissas é:

- 1.** As questões de privacidade relacionadas com a recolha de informação para identificação ou análise de comportamento de utilizadores não inviabilizam a aplicação da solução desenvolvida;



2. A solução foi desenvolvida para implementação em *sites* de classificados onde o número de utilizadores com comportamentos normal supera consideravelmente a quantidade de utilizadores suspeitos de fraude;
3. O comportamento histórico dos utilizadores é um indício do comportamento a curto prazo, sendo as alterações comportamentais progressivas e de médio ou longo prazo;
4. A ordem pela qual o utilizador realiza uma acção é relevante e contém indicações da sua intenção;
5. Um comportamento é consciente e motivado se se mantiver inalterável perante cenários adversos a esse mesmo comportamento;
6. Os anúncios impressos por consequência de armadilhas não geram qualquer débito aos seus anunciantes e não colocam em causa a qualidade dos classificados do editor.

Quanto à arquitectura, ilustrada na figura 3.1, podemos subdividi-la em três partes: utilizador, motor de publicidade (Servidor #1) e o servidor para detecção e prevenção de fraude (Servidor #2). Todas os pedidos de página de classificados são submetidas no motor de publicidade. Nesse instante, a informação do utilizador recolhida de forma implícita é salvaguardada no segundo servidor e é obtida uma avaliação do comportamento desse mesmo utilizador.

O servidor #2 fornece informações através de três operações distintas: obtenção de padrões de utilização, obtenção das distribuições das pontuações e criação de documentação.

A informação referente aos pedidos dos utilizadores é materializada num ficheiro de transacções que será interpretado pelo CAREN (<http://www.di.uminho.pt/~pja/class/caren.html>), resultando num conjunto de regras de associação que evidenciam padrões de utilização. Igualmente materializada são as pontuações obtidas pelos utilizadores nas diversas variáveis de análise. Desta feita, estes ficheiros serão interpretados por scripts de R (<http://www.r-project.org>) que irão identificar as distribuições que melhor se adaptam a esses dados. Ambos os softwares são enquadrados em secções posteriores.

Destas duas acções resultam dois executáveis *Java* que fornecem as pontuações para determinado comportamento com base nos padrões identificados e os *p-values* associados a essas pontuações segundo as distribuições estimadas. É produzido, igualmente, um conjunto de ficheiros de documentação para que todo o processo possa ser posteriormente analisado.

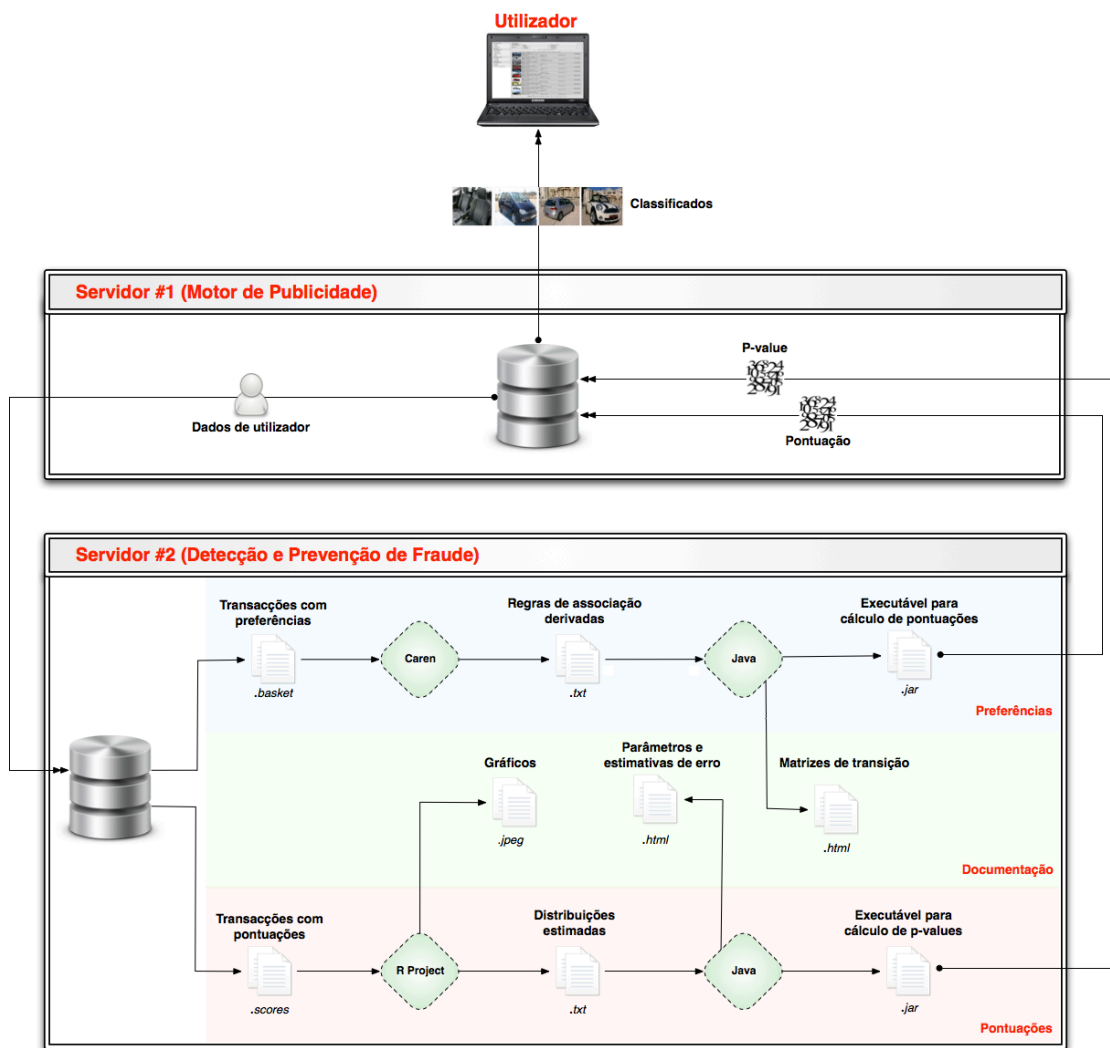


Figura 3.1 – Visão global da arquitectura da solução proposta

É esta a informação que será continuamente entregue ao motor de publicidade. Sempre que um utilizador possua uma pontuação com *p-value* inferior a um determinado limite, o motor de publicidade irá entrar em modo de segurança e os classificados atribuídos a esse utilizador serão no sentido de testar a sua intenção e não de gerar lucro. O servidor retorna ao modo normal se o utilizador ultrapassar todos esses testes com sucesso.

Salienta-se que o processo ilustrado para o servidor de detecção e prevenção de fraude executa de forma periódica evitando um custo computacional permanente. Em cada execução é produzida informação sobre o modo como o processo se comportou, nomeadamente tempo parcial e total necessário, ficheiros de entrada analisados e ficheiros de saída produzidos (Figura 3.2).



```
(Caren) Determinar as regras de associação (3,28 segundos)
> Leitura de transacções: Caren/Basket/<variável>.basket
> Escrita de regras de associação: Caren/Basket/<variável>.txt

(JAVA) Criar classes com resultados materializados das regras de associação (4,94 segundos)
> Leitura de regras de associação: Caren/Basket/<variável>.txt
> Escrita de classes java: FraudeReal/src/Main/<variável>.java
> Escrita de executável java: FraudeReal/dist/real.jar
> Escrita de documentação: Documentação/<variável>.html

(R-Project) Identificar distribuições para as pontuações (68,44 segundos)
> Leitura de scores: R/Scores/<variável>.html
> Escrita de resultados: R/Resultados/<variável>.txt
> Escrita de gráficos de apoio: R/Gráficos/<variável>.jpeg
> Escrita de documentação: WebSite/P-Values/Distribuições.html

# Tempo total de execução:76,72 segundos
```

Figura 3.2 – Informação resultante do processo de actualização de dados

Desta forma, assume-se que os dados produzidos (i.e. *p-values* e pontuações) tem uma validade temporal definida e finita. Este processo representa um processo de aprendizagem automática uma vez que se adapta de forma periódica e sem intervenção humana ao comportamento dos utilizadores.

Na próxima secção introduzimos de forma detalhada as variáveis de análise, descrevendo o modo como são calculadas e o que pretendem avaliar. O processo de extracção e armazenamento dos dados será abordado na secção 3.3 e a obtenção de regras de associação e a sua materialização na secção 3.4. A estimativa das distribuições de pontuações e o cálculo dos *p-values* ficam reservados para, respectivamente, secção 3.5 e secção 3.6. O capítulo encerra na secção 3.7 com a enumeração das armadilhas idealizadas.

3.2. Variáveis de análise

As variáveis de análise assumem um valor numérico compreendido no intervalo [0,1] e definem o comportamento do utilizador segundo diversas perspectivas. Tendo por base maioritariamente o número de visualizações e de impressões foram definidas 10 variáveis que são apresentadas na tabela 3.1. As suas expressões algébricas são de compreensão simples e resultam em grande parte de rácios. A direcção dos valores suspeitos, essencial para o cálculo dos *p-values*, é definido com base nos dados fornecidos pela AdClip e, naturalmente, na convicção e percepção que temos dos modelos PPC. Ao longo do documento iremo-nos referir às variáveis de análise pelas respectivas siglas.

A primeira das variáveis, V_{CAT} , representa o rácio entre o número de categorias visitadas e o número de categorias disponibilizadas. Tratando-se de classificados *online*, espera-se que a



generalidade dos utilizadores visite um número reduzido de categorias quando comparado com as categorias disponíveis, não atingindo pontuações elevadas (Figura 3.3 - centro).

Tabela 3.1 – Variáveis de análise idealizadas para a detecção e prevenção de fraude

Sigla	Expressão Algébrica	Pontuação Suspeita
V_{CAT}	$\frac{\sum \text{Categorias visualizadas}}{\sum \text{Categorias}}$	Demasiado elevada
V_{CTR}	$\frac{\sum \text{Visualizações}}{\sum \text{Impressões}}$	Demasiado elevada
V_{VID}	$\frac{\sum \text{Visualizações duplicadas}}{\sum \text{Visualizações}}$	Demasiado elevada
V_{IMD}	$\frac{\sum \text{Impressões duplicadas}}{\sum \text{Impressões}}$	Demasiado elevada
V_{PEC}	$\frac{\sum \text{Pedidos de contacto}}{\sum \text{Visualizações}}$	Demasiado elevada
V_{IMV}	$\frac{\sum \text{Visualizações c/ abertura de imagem}}{\sum \text{Visualizações}}$	Demasiado elevada ou demasiado reduzida
V_{TEU}	$\frac{\sum \text{Tempo utilizado}}{\sum \text{Tempo previsto}}$	Demasiado reduzida
$V_{DIV}(x, y)$	$\frac{\sum \text{Atributos de } (x, y) \text{ visualizados}}{\sum \text{Atributos de } (x, y) \text{ disponíveis}}$	Demasiado elevada
$V_{NAV}(x, y)$	Análise na Secção 3.2.1	Demasiado reduzida
$V_{REL}(x, y)$	Análise na Secção 3.2.2	Demasiado reduzida

As variáveis V_{CTR} e V_{VID} seguem o mesmo princípio mudando apenas o tipo de comportamento que representam. Em qualquer uma delas se espera a inexistência de pontuações elevadas e é de realçar, desde de já, que desempenham um papel especialmente importante na detecção de fraude. A primeira identificará os utilizadores que se predispõem a visualizar a grande maioria dos anúncios impressos, sendo natural a baixa quantidade de visualizações (Figura 3.3 - esquerda). A segunda identificará utilizadores que visualizem permanente e repetidamente determinado anúncios (e.g. anunciante a prejudicar rivais), um comportamento que não é habitual (Figura 3.3 - direita).

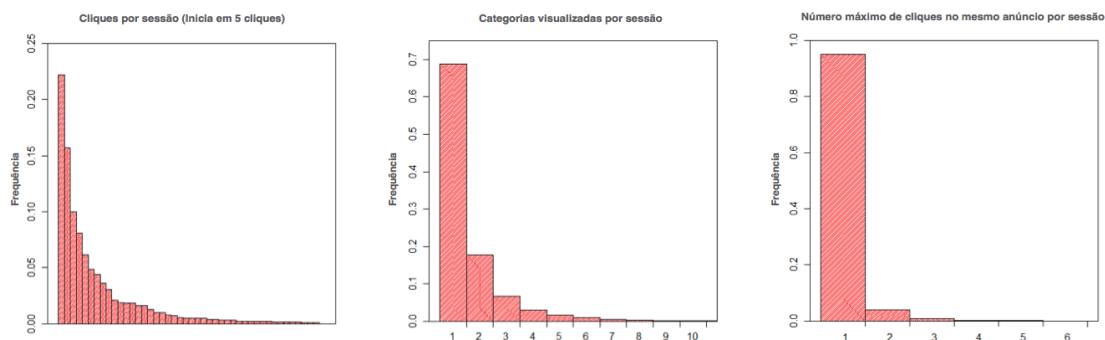


Figura 3.3 – Esquerda: Número de anúncios visualizados por cliente (inicia em 5 cliques); Centro: Número de categorias visualizadas por cliente; Direita: Número de duplicados por cliente (*Histogramas referentes aos dados da AdClip*)

As variáveis V_{IMD} e V_{PEC} focam-se na identificação de utilizadores que conhecendo melhor o sistema PPC tentam, respectivamente, gerar impressões no sentido de baixar o CTR ou contactar um número elevado de anunciantes no sentido de baixar suspeitas. O rácio entre anúncios onde se visualizou imagens e o total de anúncios visitados é controlado pela variável V_{IMV} , esperando-se a inexistência de uma pontuação demasiado reduzida ou demasiado elevada.

Com a variável V_{TEU} pretende-se compreender se o tempo de visualização que o utilizador utiliza é adequado ao anúncio em causa, afigurando-se como uma medida de interesse. É necessário definir duas variáveis extras para cada anúncio: tempo previsto para visualização T_{prev} e tempo máximo admissível T_{max} . É esperado, naturalmente, que os utilizadores que pretendem lucrar com esquemas fraudulentos não estejam receptivos a despendem muito tempo em cada anúncio, sendo facilmente identificados porque se distanciarem de T_{prev} . Por outro lado, com a inserção de T_{max} , evitamos (i.e. não consideramos) tempos irrealistas de utilizadores que deixam os anúncios abertos por tempo indeterminado, o que aumentaria V_{TEU} até uma pontuação próxima de 1.

No protótipo desenvolvido, T_{prev} foi definido com base no tamanho da descrição do anúncio e no número de fotos, enquanto T_{max} corresponde a $2 * T_{prev}$. No entanto outras medidas (e.g. média ou mediana) podem ser utilizadas, desde que representem de forma adequada o comportamento habitual dos utilizadores. Tomemos como exemplo um anúncio com T_{max} de 10 segundos e um utilizador com V_{TEU} actual de $\frac{132}{197}$. Se um dado utilizador despende ϕ tempo na visualização desse anúncio a pontuação final de V_{TEU} será calculado do seguinte modo:



$$f(\phi) = \begin{cases} \frac{132 + \phi}{197 + T_{Prev}}, & 0 < \phi \leq T_{Prev} \\ \frac{132 + T_{Prev}}{197 + T_{Prev}}, & T_{Prev} < \phi \leq T_{Max} \\ V_{TEU}, & \phi > T_{Max} \end{cases}$$

Por último, a variável $V_{DIV}(x, y)$ foi criada com o intuito de avaliar a diversidade das escolhas do utilizador, sendo x a secção do anúncio e y o atributo do anúncio que queremos analisar. Para melhor compreensão, considere-se $x = \text{automóveis}$ e $y = \text{marca}$. Neste caso, estaríamos a calcular o rácio entre as marcas visualizadas pelo utilizador e as marcas totais existentes na zona de classificados. É expectável que a gama de valores visualizados por um utilizador para cada um dos atributos seja consideravelmente inferior à gama de valores disponíveis.

3.2.1. Variável de análise $V_{NAV}(x, y)$

A navegação entre anúncios é um dos factores mais relevantes no que concerne à análise da coerência e da relação existente nas opções de cada utilizador. Em classificados *online* é comum as visualizações de um utilizador, para determinada sessão de *browser*, partilharem características semelhantes. No entanto, para um ataque fraudulento, o utilizador tem por objectivo gerar o maior lucro possível através do aumento do número de cliques, independentemente dos anúncios que visualiza. Nestas circunstâncias é previsível que esta semelhança ou relação entre anúncios seja consideravelmente menor, conduzindo-o para um cenário suspeito.

O intuito da variável $V_{NAV}(x, y)$ é medir essa relação entre anúncios considerando uma secção de classificados x e o atributo do anúncio y . O cálculo da pontuação de V_{NAV} está directamente relacionado com o conhecimento que é extraído dos dados por via de regras de associação. Quanto maior a relação entre os anúncios visualizados pelo utilizador, mais relevantes serão os padrões obtidos nas regras. Consequentemente, maior será a pontuação de V_{NAV} e menor serão as suspeitas.

Os detalhes sobre a obtenção e escolha das regras de associação serão abordados na secção 3.4. Para já, concentremo-nos no modo como é obtida a pontuação de V_{NAV} . Assuma-se um exemplo simplista de anúncios automóveis com a sequência de visualizações $S = \{\text{Seat}, \text{Audi}, \text{Seat}, \text{Seat}, \text{Seat}, \text{Audi}, \text{Volvo}, \text{Mercedes}, \text{Mercedes}, \text{Mercedes}\}$ para melhor entender o modo



como as regras de associação sustentam o cálculo. Em cada momento, neste exemplo, são consideradas três variáveis distintas:

- Conjunto A (Antecedentes) para representar as marcas de automóveis já visualizadas;
- Elemento C (Consequente) para a marca do automóvel em visualização;
- Lista L com a confiança das regras de associação seleccionadas.

Em cada visualização, sendo A' os subconjuntos resultantes de A , será escolhida a regra de associação com maior confiança de entre as regras $C \leftarrow \beta$, $\beta \in A'$. A confiança pode ser interpretada como a probabilidade condicionada $P(C|\beta)$, pelo que estaremos em cada instante a escolher a maior probabilidade de visitar o elemento C sabendo que foi visitado um subconjunto de elementos de A . Maximizando a confiança estaremos a maximizar igualmente a variável V_{NAV} e a escolher os padrões que melhor definem o comportamento ou acção do utilizador naquele instante.

Será de salvaguardar que este processo só decorre para situações onde o consequente nunca foi visitado (i.e. não pertence ainda aos antecedentes). Deste modo a sequência S é equivalente a $S' = \{Seat, Audi, Volvo, Mercedes\}$, a ordem pela qual os anúncios são visitados. A tabela 3.2 ilustra as etapas que definem o cálculo de V_{NAV} para a sequência S' .

Tabela 3.2 - Cálculo da V_{NAV} sem incorporação de teste binomial

Etapa	Consequente	Antecedentes	Regras aplicáveis	Confiança
1	<i>Seat</i>	\emptyset	[Cf: 0.10250] Seat \leftarrow	0.10250
2	<i>Audi</i>	\emptyset <i>Seat</i>	[Cf=0.49268] Audi \leftarrow Seat [Cf=0.17450] Audi \leftarrow	0.49268
3	<i>Volvo</i>	\emptyset <i>Seat</i> <i>Audi</i>	[Cf=0.06350] Volvo \leftarrow	0.06350
4	<i>Mercedes</i>	\emptyset <i>Seat</i> <i>Audi</i> <i>Volvo</i>	[Cf=0.40792] Mercedes \leftarrow Seat & Audi [Cf=0.39024] Mercedes \leftarrow Seat [Cf=0.29150] Mercedes \leftarrow	0.40792
V_{NAV}				0.19018



A primeira etapa é sempre constituída pela regra $C \leftarrow \emptyset$, pelo que a confiança corresponde ao suporte do consequente. Na segunda etapa, para este exemplo, é possível apurar que a probabilidade de visualizar veículos da marca *Audi* aumenta se houver visualizações anteriores da marca *Seat*. Pela terceira etapa pode-se concluir que a marca *Volvo* não está relacionada com nenhum dos veículos anteriormente visualizados, pelo que se volta a seleccionar a regra base, i.e. suporte de *Volvo*. Por último, quarta etapa, assume-se que a visualização do veículo *Mercedes* advém do facto do utilizador ter visualizado igualmente as marcas *Seat* e *Audi*. Mesmo que tal afirmação não seja correcta, estamos a beneficiar a pontuação de V_{NAV} e naturalmente o próprio utilizador.

Tal como introduzido na terminologia adoptada, cada uma das etapas mencionadas é denominada de transição, i.e. a segunda transição é a que possui antecedente $\{\emptyset, Seat\}$ e consequente $\{Audi\}$. Assim, transições com confiança elevada são tidas como normais e aumentam o valor de V_{NAV} , acontecendo a situação inversa para confianças reduzidas.

De modo semelhante a Sadagopan e Li (2008), o cálculo de V_{NAV} é baseado na confiança das transições segundo a equação 1, sendo C_i a confiança da i -ésima transição. Este valor pode igualmente ser interpretado como o produto das probabilidades condicionadas que maximiza V_{NAV} .

No entanto, para evitar pontuações reduzidas inerentes a utilizadores com elevada quantidade de transições utiliza-se a função logarítmica e respectiva normalização pelo número de transições. A função exponencial é igualmente aplicada para que o resultado retome ao intervalo $[0,1]$, tal como todas as outras pontuações. O cálculo final para V_{NAV} é obtido pela expressão algébrica apresentada na equação 2.

$$\gamma = \prod_{1}^i C_i = P(Mercedes|Seat, Audi) * P(Volvo|\emptyset) * P(Audi|Seat) * P(Seat|\emptyset) \quad \text{(Eq.1)}$$

$$V_{NAV} = \exp\left(\frac{\log(\gamma)}{i}\right) \quad \text{(Eq.2)}$$

Embora haja semelhanças, a abordagem escolhida é distinta das cadeias markovianas e das redes bayesianas. Considere-se A os antecedentes, A' os subconjuntos de A , i o número de transições, X_n o consequente da transição n e m a ordem das cadeias de markov. Nestas



condições, as equações 4, 5 e 6 representam, respectiva e formalmente, as cadeias de markov, as redes bayesianas e a abordagem por nós escolhida.

$$\begin{cases} \prod_{n=1}^i P(X_n|X_{n-1}, X_{n-2}, \dots, X_1) & , \text{sendo } n \leq m \\ \prod_{n=1}^i P(X_n|X_{n-1}, X_{n-2}, \dots, X_{n-m}), & \text{sendo } n > m \end{cases} \quad \text{(Eq.4)}$$

$$\prod_{n=1}^i P(X_n|X_{n-1}, X_{n-2}, \dots, X_1) \quad \text{(Eq.5)}$$

$$\prod_{n=1}^i \max (P(X_n|\omega)), \forall \omega \in A' \quad \text{(Eq.6)}$$

Enquanto que as cadeias de *markov* calculam a probabilidade de um acontecimento com base num tamanho fixo de antecedentes (denominadas cadeias de ordem m), as redes bayesianas calculam-na considerando todos os antecedentes. Na nossa abordagem, relembramos, pretendemos objectivamente a maximização da probabilidade de um acontecimento de modo a considerar o melhor dos comportamentos possíveis (i.e. padrões) para cada utilizador. Assim, se a probabilidade for abaixo do esperado restará ainda menos dúvidas de que o utilizador é de facto suspeito.

Por último, é necessário garantir que todas transições consideradas são significativamente representativas, i.e. um clique accidental ou a curiosidade em determinado anúncio relacionado ou não relacionado interfere de forma mínima (ou até nula) na pontuação obtida. Para esse fim são realizados testes binomiais $x \sim Bin(n, p)$ a cada conseqente. Estes testes são executados de forma direccional (à direita) e com um grau de confiança de 95%. Neste contexto x representa o número de visualizações do conseqente (casos de sucesso), n o número de impressões do conseqente (número de tentativas) e p a probabilidade de o utilizador visualizar um anúncio (probabilidade de sucesso em cada tentativa). O valor de p é, em cada instante, igual à pontuação de V_{CTR} .

Os valores apurados após incorporação do teste, para o exemplo em análise, são apresentados na tabela 3.3 e, graficamente, na figura 3.4. Salienta-se que pelo facto de a marca *Volvo* ter sido ignorada com um *p-value* inferior a 0.05, a pontuação de V_{NAV} subiu de 0.19018 para 0.37885.



Tabela 3.3 - Cálculo da V_{NAV} com incorporação de teste binomial

Transição	Marca	Visualizações	Impressões	P-Value $x \sim Bin(n, p)$	Confiança
1	<i>Seat</i>	15	19	0.99674	0.10250
2	<i>Audi</i>	27	45	0.90227	0.49268
3	<i>Volvo</i>	3	14	0.02214	0.06350
4	<i>Mercedes</i>	9	23	0.16437	0.40792
CTR Geral		0.51485	V_{NAV}		0.37885

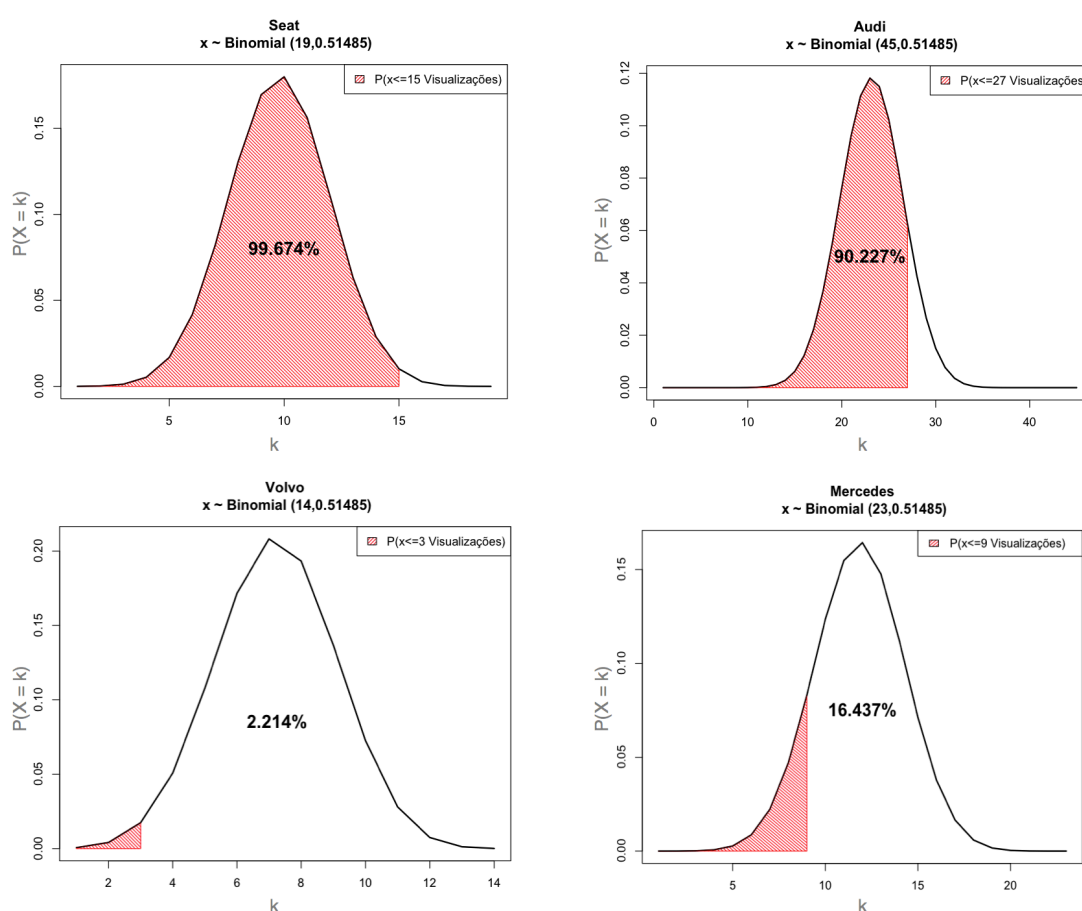


Figura 3.4 - Representação gráfica do teste binomial para o cálculo da V_{NAV}

3.2.2. Variável de análise $V_{REL}(x, y)$

A utilização da variável $V_{REL}(x, y)$ visa identificar os casos onde não existe uma ou mais preferências óbvias nas escolhas dos utilizadores segundo o número de visualizações. A dificuldade em extrair preferências é mais perceptível nos casos onde as escolhas dos



utilizadores é afectada de alguma aleatoriedade. Para o cálculo de $V_{REL}(x, y)$, onde $x =$ *automóveis* e $y =$ *distrito*, considere-se os dois cenários ilustrados na tabela 3.4.

Tabela 3.4 - Cenários para cálculo da V_{REL}

Cenário #1				Cenário #2			
Distrito	V	FR	FRA	Distrito	V	FR	FRA
<i>Braga</i>	14	0.350	0.350	<i>Porto</i>	7	0.175	0.175
<i>Porto</i>	7	0.175	0.525	<i>Lisboa</i>	6	0.150	0.325
<i>Aveiro</i>	5	0.125	0.650	<i>Faro</i>	6	0.150	0.475
<i>Coimbra</i>	4	0.100	0.750	<i>Viseu</i>	5	0.125	0.600
<i>Viseu</i>	3	0.075	0.825	<i>Aveiro</i>	4	0.100	0.700
<i>V. Real</i>	2	0.050	0.875	<i>Braga</i>	4	0.100	0.800
<i>Lisboa</i>	2	0.050	0.925	<i>V. Real</i>	3	0.075	0.875
<i>Faro</i>	1	0.025	0.950	<i>Coimbra</i>	2	0.050	0.925
<i>V. Castelo</i>	1	0.025	0.975	<i>Guarda</i>	2	0.050	0.975
<i>Guarda</i>	1	0.025	1.000	<i>V. Castelo</i>	1	0.025	1.000

(*) V=Número de visualizações; FR=Frequência relativa; FRA=Frequência relativa acumulada;

Apesar de ambos possuírem um total de 40 cliques distribuídos por 10 distritos, o primeiro cenário deixa transparecer uma preferência do utilizador mais notória. Para o primeiro caso 75% dos cliques do utilizador são realizados em apenas 4 distritos (*Braga*, *Porto*, *Aveiro* e *Coimbra*) e o distrito com maior número de visualizações contribuí com cerca de metade desses cliques. No segundo caso, para obter a mesma percentagem, é necessário considerar os primeiros 6 distritos, sendo que o distrito principal apenas contribuí com 17.5%.

As visualizações aleatórias tendem a gerar no pior caso uma frequência relativa uniforme para todos os distritos. À semelhança das curvas ROC⁴, onde se pode utilizar a AUC (*Area Under Curve*) para mesurar a qualidade de um modelo de classificação face à aleatoriedade, optou-se por medir a diferença da área da frequência relativa acumulada para uma situação puramente aleatória de modo a medir a evidência de preferências (Figura 3.5). Os valores apurados ($V_{REL} = 0.465$ e $V_{REL} = 0.166$) representam o acréscimo de área em relação ao pior caso e confirmam a existência de uma maior preferência no primeiro caso.

⁴ *Receiver Operating Characteristics*: Método gráfico para avaliação e selecção de modelos de classificação com base no seu desempenho.



Deste modo, existe uma proporcionalidade directa entre o peso (i.e. frequência relativa) dos principais valores do atributo y que são visualizados na secção x e a pontuação de $V_{REL}(x, y)$. É expectável que os utilizadores que naveguem de forma aleatória ou sem critério possuam pontuações de V_{REL} reduzidas.

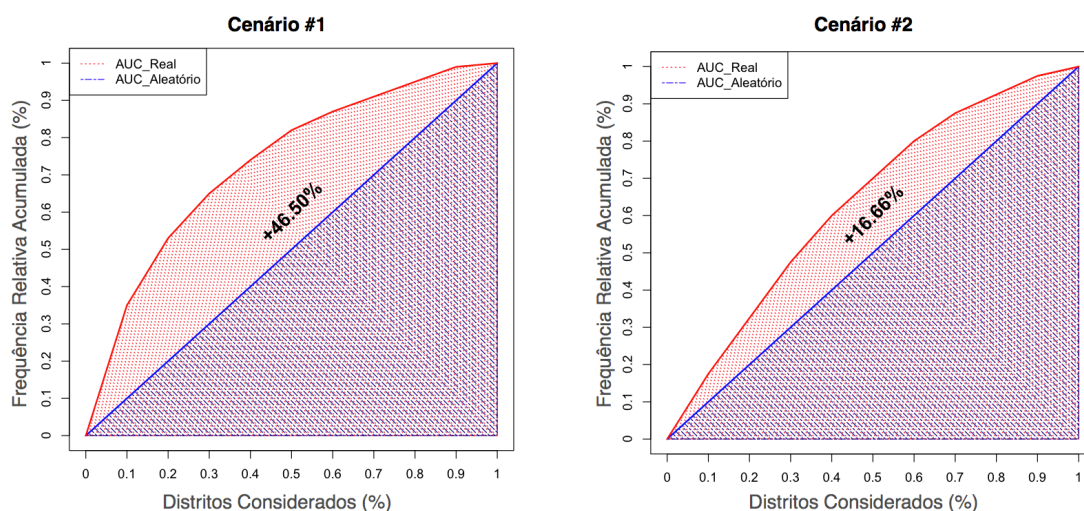


Figura 3.5 - Representação gráfica do cálculo da V_{REL}

3.3. Extração de dados

A extração e análise de dados referentes à actividade do utilizador e às pontuações obtidas em cada variável de análise é a primeira das etapas a realizar. Uma vez que são utilizados programas externos que necessitam de um *input* com um formato pré-estabelecido, toda a informação é extraída, manipulada e salva em ficheiros de transacções com extensão *.basket* e *.scores*.

Os ficheiros *.basket* são a base para o cálculo da variável de análise $V_{NAV}(x, y)$ e referem-se a uma representação de dados em formato *cesto de compras*⁵. São criados $x * y$ ficheiros, um por cada atributo de cada secção que pretendemos analisar. Possuem uma representação horizontal onde cada linha representa um utilizador e a gama de valores de y visualizado pelo mesmo. Uma vez criados, são utilizados para detectar padrões de visualização entre os valores de y . A figura 3.6 (esquerda) ilustra parte desse ficheiro para o atributo *distrito*, da secção *automóvel*.

⁵ Formato tipicamente utilizado para identificar hábitos e relações nas compras dos clientes através de uma técnica de mineração de dados denominada de *Market Basket Analysis* (MBA).



Assim, a transacção número dois ilustra um utilizador que visualizou anúncios de automóveis dos distritos de Aveiro, Coimbra e Porto.

1, Beja, Setúbal, Évora, Faro, Lisboa, Portalegre	0.3959079872983065
2, Aveiro, Coimbra, Porto	0.366033513807344
3, Bragança, Vila Real, Viseu, Guarda, Porto	0.17937878311364877
4, Beja, Évora, Setúbal, Faro	0.4199635575838867
5, Braga, Porto, Viana Do Castelo	0.6298160176831537
6, Beja, Faro, Évora	0.1715583352893732
7, Coimbra, Viseu, Aveiro	0.5308208924572916
8, Braga, Viana Do Castelo, Porto, Vila Real	0.21728507126143373
9, Bragança, Vila Real	0.4977426709484439

Figura 3.6 – Formato do ficheiro *distrito.basket* (esquerda) e formato do ficheiro *ctr.scores* (direita)

Por sua vez, os ficheiros *.scores* (Figura 3.6 – direita) armazenam as pontuações obtidas pelos utilizadores nas variáveis de análise. Cada ficheiro corresponde a uma única variável de análise e cada linha representa um utilizador e a sua pontuação.

De salientar que no decorrer do desenvolvimento foi testado o formato atributo-valor, semelhante uma tabela relacional, para representar os dados. No entanto para esse formato o processo de extracção passou a ser um ponto de estrangulamento devido ao tamanho e tempo necessário para a manipulação dos ficheiros. Pela mesma razão, os dados contidos nestes ficheiros não devem representar toda a população mas apenas uma amostra suficientemente representativa.

Uma vez que a extracção de informação irá desencadear um conjunto de acções – analisadas nas próximas secções – é necessário definir de forma minuciosa o momento e a periodicidade com que se executa esta actividade. Aconselha-se um período de baixa actividade no *site* para atenuar a redução no tempo de resposta ao motor de publicidade e uma periodicidade tal que permita aos dados representarem de forma fiel e actual a actividade dos utilizadores.

3.4. Obtenção de regras de associação

Uma vez obtidos os ficheiros de transacções *.basket* é utilizado o CAREN. Este software, desenvolvido em Java por Azevedo (2003), foi implementado com o propósito de derivar regras de associação e construir modelos de classificação. A simplicidade de processos, a celeridade no cálculo das mesmas, o facto de as exportar para vários formatos e, principalmente, a variedade das medidas de interesse para filtrar as regras de associação - 13 no total - fez do CAREN a nossa escolha. No entanto, a potencialidade do *software* estende-se a outras funcionalidades disponíveis e que não foram utilizadas no âmbito deste projecto.



Parte das regras de associação derivadas possui informação não relevante, sendo necessário filtra-las através das diferentes medidas de interesse que a problemática exige. Deste modo, reduz-se a complexidade, o número de regras a aplicar e as regras que não estão de facto correlacionadas.

Uma vez que utilizamos a confiança como pontuação da variável V_{NAV} deve-se manter regras para a generalizada dos casos. Assim, o suporte mínimo é fixado em 1% para permitir a identificação de padrões raros e a confiança mínima fixada em 5% para obter praticamente todas as regras. Para evitar as regras em que, independentemente da confiança, o consequente e o(s) antecedente(s) são negativamente dependentes ou independentes é necessário aplicar outras medidas de interesse. Das várias existentes e implementadas pelo CAREN (*Conviction, Lift, Leverage*, etc) optou-se pela noção de melhoria (*improvement*), i.e. uma regra mais específica tem que produzir uma mais-valia em termos de conhecimento. Para o protótipo desenvolvido o seu valor esta fixada a um mínimo de 10%. O modo como a confiança e a melhoria são obtidas são expressos pelas equações 7 e 8, onde $s(x)$ representa o suporte de x .

$$conf(A \rightarrow B) = \frac{s(A \cup B)}{s(A)} \quad \text{(Eq.7)}$$

$$imp(A \rightarrow B) = \min(conf(A \rightarrow B) - conf(A' \rightarrow B)), \forall A' \in A \quad \text{(Eq.8)}$$

Com estas três restrições o número de regras produzidas reduziu, em média, 40%. A figura 3.7 ilustra algumas das regras obtidas para os consequentes Porto, Braga e Faro. Além de todas as regras estarem nativamente ordenadas por confiança, o CAREN permite ainda derivar apenas regras com determinado consequente e antecedentes (opção -h e *a).

Apesar de esta funcionalidade facilitar na escolha da regra que maximiza a variável V_{NAV} , é, infelizmente, inviável a sua utilização em tempo real visto que a cada clique de um utilizador teríamos que derivar novamente todas as regras. Uma vez que se pretende uma solução em tempo real, optou-se pela materialização das regras num executável JAVA através de uma estrutura denominada de matrizes de transição.

As matrizes de transição apresentam-se como sendo tabelas bidimensionais onde as colunas, linhas e valores representam, respectivamente, consequentes, antecedentes e confiança. O número de colunas corresponde à gama de valores do atributo em análise. No entanto, o



número de linhas é significativamente inferior ao número de regras derivadas uma vez que a quantia de antecedentes distintos é, igualmente, inferior à quantidade de regras.

Sup = 0.07350	Conf = 0.97351	Porto	<--	Braga & Coimbra
Sup = 0.05850	Conf = 0.93600	Porto	<--	Vila Real & Aveiro
Sup = 0.13600	Conf = 0.92517	Porto	<--	Viana Do Castelo
Sup = 0.03350	Conf = 0.91781	Porto	<--	Vila Real & Coimbra
Sup = 0.20400	Conf = 0.87179	Porto	<--	Braga
Sup = 0.26800	Conf = 0.71754	Porto	<--	Aveiro
Sup = 0.22100	Conf = 0.68210	Porto	<--	Coimbra
Sup = 0.14050	Conf = 0.62584	Porto	<--	Vila Real
Sup = 0.17900	Conf = 0.61407	Porto	<--	Viseu
Sup = 0.39200	Conf = 0.39200	Porto	<--	
Sup = 0.05800	Conf = 0.92800	Braga	<--	Vila Real & Aveiro
Sup = 0.03300	Conf = 0.90411	Braga	<--	Vila Real & Coimbra
Sup = 0.11150	Conf = 0.79359	Braga	<--	Vila Real & Porto
Sup = 0.12400	Conf = 0.55234	Braga	<--	Vila Real
Sup = 0.20400	Conf = 0.52041	Braga	<--	Porto
Sup = 0.23400	Conf = 0.23400	Braga	<--	
Sup = 0.07300	Conf = 0.64035	Faro	<--	Lisboa
Sup = 0.11950	Conf = 0.57314	Faro	<--	Setúbal
Sup = 0.13650	Conf = 0.54382	Faro	<--	Beja
Sup = 0.11550	Conf = 0.49677	Faro	<--	Évora
Sup = 0.14000	Conf = 0.14000	Faro	<--	

Figura 3.7 – Regras de associação derivadas pelo CAREN para os distritos visualizados

A figura 3.8 ilustra uma parte da representação das matrizes de transição para o atributo *marca* da secção *automóvel*. Pela sua análise é possível afirmar que estamos perante matrizes esparsas. Tal facto justifica-se com a inexistência de regras que relacionem antecedente e consequente ou pelas restrições anteriormente enumeradas (suporte, confiança e *improvement* mínimo).

Antecedente \ Consequente	Audi	BMW	Cadillac	Chevrolet	Chrysler	Dodge	Ferrari	Ford	GMC	Honda	Hyundai	Isuzu	Jaguar	Jeep
∅	0.1575	0.059	0.2305	-	0.17	0.197	0.13	-	0.172	0.135	0.151	0.157	0.0705	0.065
Audi	-	-	0.3746	-	0.40317	-	0.25079	-	-	-	-	-	-	-
Audi,Cadillac	-	-	-	-	0.64407	-	-	-	-	-	-	-	-	-
Audi,Chrysler	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Audi,Chrysler,Hyundai	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Audi,Chrysler,Toyota	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Audi,Dodge	-	-	-	-	-	-	0.45333	-	-	-	-	-	-	-
Audi,Ferrari	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Audi,Ferrari,Kia	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Audi,GMC	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Audi,Hyundai	-	-	0.75714	-	0.62857	-	-	-	-	-	-	-	-	-
Audi,Hyundai,Kia	-	-	-	-	0.73684	-	-	-	-	-	-	-	-	-
Audi,Isuzu	-	-	-	-	0.625	-	-	-	-	-	-	-	-	-
Audi,Mitsubishi	-	-	0.725	-	0.925	-	-	-	-	-	-	-	-	-
Audi,Nissan	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Audi,Toyota	-	-	-	-	-	0.43056	0.38889	-	-	-	-	-	-	-
Audi,Volkswagen	-	-	0.82143	-	-	-	-	-	-	-	-	-	-	-
BMW	-	-	-	-	-	0.44915	-	-	0.52542	-	-	-	-	-
BMW,GMC	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BMW,Honda	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BMW,Kia	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BMW,Smart	-	-	-	-	-	-	-	-	0.74138	-	-	-	-	-

Figura 3.8 – Representação parcial da matriz de transição produzida para V_{NAV} (automóveis, marca)



Com esta alteração, o CAREN apenas executa uma única vez imediatamente a seguir à extracção dos dados. As regras derivadas são posteriormente materializadas num executável JAVA responsável por devolver o valor da célula que une um dado antecedente e um dado consequente (i.e. a confiança da regra). São estes os valores utilizados no cálculo da pontuação da variável V_{NAV} .

3.5. Estimativa das distribuições de pontuações

Até ao momento foram já abordadas as questões de extracção de dados, extracção de conhecimento por meio das regras de associação e as variáveis de análise utilizadas. Com estas operações é já possível ter pontuações em concordância com o comportamento ou atitude de cada utilizador. No sentido de considera-lo suspeito ou fraudulento é necessário entender e distinguir o que são pontuações normais ou anormais e idealmente identificar o local de fronteira entre estas duas classificações.

Para o contexto PPC não se conhecem definições consensuais para definir normalidade. Como tal, um acontecimento anormal é tipicamente tido como algo que se desvia de forma significativa ou extrema dos restantes casos observados. Desta forma, para um grau de confiança α e uma distribuição de pontuações ω , optamos por classificar cada pontuação segundo um teste de significância estatística assumindo as tradicionais duas hipóteses:

- H_0 (hipótese nula): Pontuação normal, i.e. não é significativamente diferente do valor esperado quando considerada a aleatoriedade em ω ;
- H_1 (hipótese alternativa): Pontuação suspeita, i.e. significativamente diferente do valor esperado.

A rejeição da hipótese nula ocorre se a pontuação actual estiver na zona de rejeição, i.e. obtiver um *p-value* inferior a $1 - \alpha$.

Para dispensar a intervenção humana na definição de ω , é necessário que a solução esteja dotada de um processo de aprendizagem automática capaz de estimar, para cada variável, a distribuição ω e os respectivos parâmetros. Sabendo que as pontuações variam entre 0 e 1, são poucas as distribuições que sem normalização e respectiva reformulação no cálculo dos *p-values* podem-se ajustar correctamente aos dados. Como tal, foram consideradas três distribuições: beta, exponencial e normal.



A distribuição beta varia exclusivamente no intervalo $[0,1]$ e apresenta uma enorme versatilidade (Figura 3.9). Caracteriza-se por dois parâmetros (denominaremos de A e B) que controlam a sua forma e escala, podendo assemelhar-se facilmente pela curvatura a outras distribuições como a exponencial, *fisher*, gaussiana, X^2 ou uniforme.

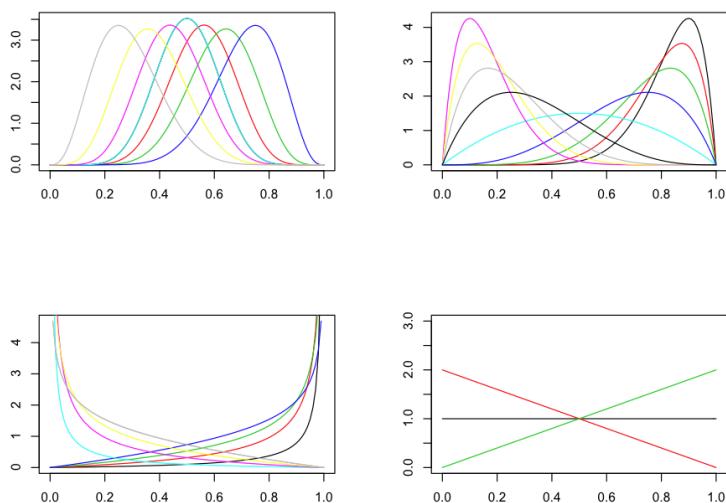


Figura 3.9 – Distribuições obtidas pela variação dos parâmetros da distribuição beta

Por outro lado, no contexto PPC, é expectável que as pontuações de algumas variáveis de análise possam apresentar uma natureza exponencial ou gaussiana. Nestes casos, as duas restantes distribuições (i.e. exponencial e normal) descrevem os dados com maior precisão que a distribuição beta. Consequentemente foram igualmente consideradas desde que os valores possíveis nas distribuições estimadas estejam totalmente compreendidos entre 0 e 1, de modo a não interferir no cálculo dos p-values. Por exemplo, uma distribuição $x \sim Norm(0.5,1)$ será rejeitada uma vez que contém valores inferiores a 0 e superiores a 1.

Uma vez escolhidas as distribuições, é necessário encontrar o valor dos parâmetros que permitem o melhor ajuste aos dados. Para este objectivo foi utilizada as potencialidades do R (<http://www.r-project.org>). O R faz parte do projecto GNU e é um software gratuito que se assume como uma linguagem de programação para computação estatística e gráfica. A familiaridade com o R, o facto de permitir a execução de *scripts* e de apresentar uma rapidez apreciável nos cálculos pretendidos foram os factores predominantes para a nossa escolha.

A primeira funcionalidade aplicada é a *fitdistr* da biblioteca MASS (Venables & Ripley, 2002) que visa a estimativa de parâmetros para um determinado modelo estatístico pela maximização da função de verosimilhança (Myung, 2003). O valor que maximiza uma função de verosimilhança



$V(x; \theta)$, onde θ representa os parâmetros a estimar e x os valores obtidos por amostra, é denominado de estimador de máxima verosimilhança (equação 9). Por sua vez, este valor é igual ao valor que maximiza o logaritmo da função de verosimilhança $L(x; \theta)$, ilustrada na equação 10 e denominado de estimador log-verosimilhança. Como tal, opta-se pela derivada de $L(x; \theta)$ para a obtenção do estimador de máxima verosimilhança (equação 11) ao invés da derivada de $V(x; \theta)$, uma vez que é computacionalmente menos dispendiosa a derivação de somas quando comparado com a derivação de multiplicações.

$$V(x_1 \dots x_n; \theta) = f(x_1; \theta) * \dots * f(x_n; \theta) = \prod_{i=1}^n f(x_i; \theta) \quad \text{(Eq.9)}$$

$$L(x_1 \dots x_n; \theta) = \ln (V(x_1 \dots x_n; \theta)) = \ln (f(x_1; \theta) * \dots * f(x_n; \theta))$$

$$= \sum_{i=1}^n \ln (f(x_i; \theta)) \quad \text{(Eq.10)}$$

$$\frac{d (L(x_1 \dots x_n; \theta))}{d(\theta)} = 0 \quad \text{(Eq.11)}$$

Em suma, para uma distribuição D com parâmetros D_p e um conjunto de dados amostrais A , a funcionalidade *fitdistr* fornece a estimativa de D_p - e respectivos erros - que maximizam o ajuste de D a A .

Obtidos os parâmetros de cada uma das três distribuições é necessário identificar a que melhor se ajusta aos dados. Embora o estimador log-verosimilhança já deixe transparecer o melhor, optou-se por validar a escolha com duas medidas de ajuste ao modelo estatístico: AIC (*Akaike Information Criterion*) e o majorante da diferença das funções de densidade de probabilidades proveniente do teste de K-S (*Kolmogorov-Smirnov*). Ambas podem ser encontradas na biblioteca *STATS* do R.

O critério de informação AIC desenvolvido por Akaike (1974) é definido pelo número de parâmetros P estimados e o estimador log-verosimilhança M (equação 9). Baseia-se na teoria de informação e visa penalizar a utilização de parâmetros extras que não reduzam a variância. Quanto menor o seu valor, melhor será o ajuste da distribuição estimada.

$$AIC = 2 * P - 2 * \ln (M) \quad \text{(Eq.9)}$$

Por outro lado, o teste K-S compara a proximidade entre as funções de densidade de probabilidades dos nossos dados e do modelo estimado, respectivamente $F_{Dados}(x)$



e $F_{Estimada}(x)$. O objectivo é identificar e mesurar a distância máxima, D_{KS} , existente entre as duas funções. Considera-se as equações 10, 11 e 12 na interpretação desta métrica, onde valores menores simbolizam melhores ajustes.

$$F_{Dados}(x) = \frac{1}{n} * \sum_{i=1}^n F'_{]-\infty, x]}(x_i)$$

$$n = \sum x_i \quad \text{(Eq.10)}$$

$$F'_{Intervalo}(x) = \begin{cases} 1; & \text{se } x \in \text{intervalo} \\ 0; & \text{caso contrário} \end{cases}$$

$$F_{Estimada}(x) = P(X \leq x) \quad \text{(Eq.11)}$$

$$D_{KS} = \max (|F_{Estimada}(x_i) - F_{Dados}(x_i)|) \quad \text{(Eq.12)}$$

Ao contrário do AIC, o K-S fornece igualmente um teste ao modelo aceitando ou rejeitando a hipótese nula com base nos valores críticos da tabela da distribuição *Kolmogorov*. Infelizmente, para o tamanho das amostra utilizadas (e.g. tipicamente superior a 10000) os modelos testados são sempre rejeitados de forma categórica, mesmo estando perante ajustes muito aceitáveis para o nosso contexto. Esta situação justifica-se pelo facto de os valores críticos serem calculados em função do tamanho da amostra n . Assim, por exemplo, para um grau de confiança de 95% qualquer modelo com distância máxima superior a $\frac{1.36}{\sqrt{n}}$ é rejeitado, apresentando-se como um teste demasiado relaxado.

Vejamos um exemplo prático para melhor entender todo o processo de identificação de distribuições. O histograma da figura 3.10 ilustra a distribuição de pontuações dos utilizadores para a variável $V_{NAV}(\text{automóveis, distrito})$.

Da estimativa de parâmetros resultaram as três distribuições representadas: beta (vermelho), exponencial (azul) e gaussiana (verde). A falta de ajuste da distribuição exponencial não surpreende e é óbvia antes de qualquer análise estatística. No entanto, a distribuição beta e normal tiveram um ajuste muito significativo e semelhante. Uma vez que a solução gera documentação no final de cada análise, é possível entender não só qual a distribuição escolhida como os fundamentos dessa decisão.

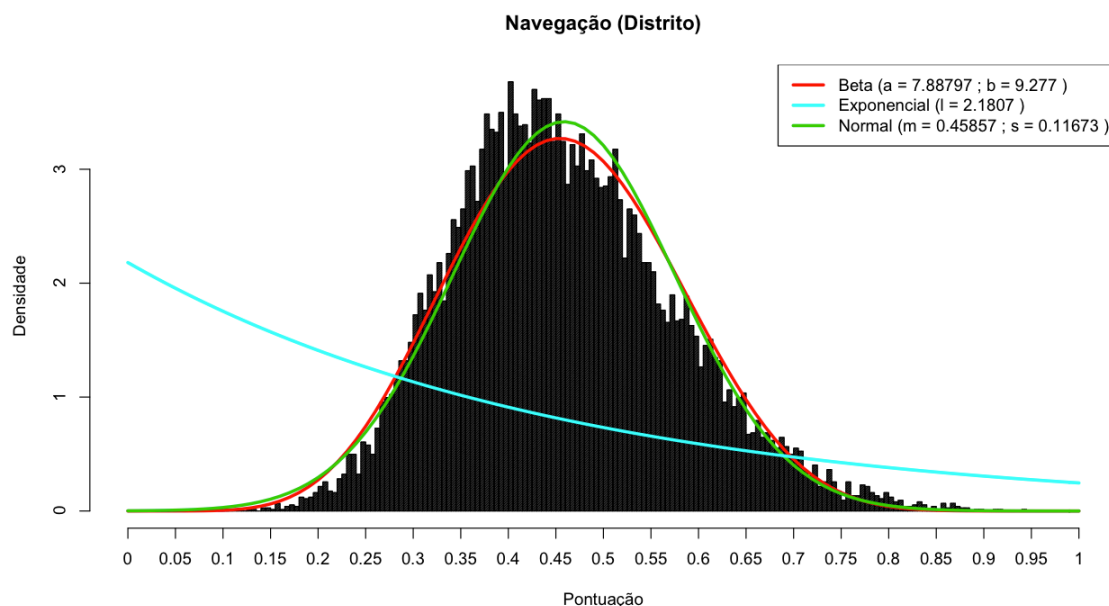


Figura 3.10 – Estimativa da distribuição de pontuações de V_{NAV} (automóveis, distrito)

Pela tabela 3.5 podemos afirmar que a distribuição exponencial foi excluída, primeiramente, por não enquadrar a totalidade dos seus valores no intervalo $[0,1]$. Salienta-se novamente que a distribuição beta é a única que nunca é excluída por esta restrição uma vez que varia unicamente nesse intervalo. As distribuições beta e normal seguiram para análise resultando, respectivamente, um estimador de log-verosimilhança de 10854.37 e 10834.09. De ressaltar que, pese embora a visualização de valores negativos para o estimador seja recursivo, o mesmo pode assumir valores positivos sob determinadas circunstâncias. Se as densidades assumirem maioritariamente valores superiores a 1, o logarítmico do produto resultará num valor positivo. Precisamente o caso em análise, onde cerca de $\frac{2}{3}$ das amostras tem densidade superior a 1 resultando num valor final bastante superior a 0.

Tabela 3.5 – Resultados obtidos para a estimativa da distribuição de pontuações de V_{NAV} (automóveis, distrito)

Distribuição	$x \in [0,1]$?	Escolhido?	$L(x;\Theta)$	D [KS]	AIC	1º Parametro +- Erro estimado	2º Parametro +- Erro estimado
Beta(A,B)	✓	✓	10854.37	0.0336876	-21704.74	a=7.887974 +- 0.09002277	b=9.276996 +- 0.10638123
Exp(λ)	✗	✗	-3274.936	0.4125389	6551.871	$\lambda=2.180695$ +- 0.01788777	
Normal(X, σ)	✓	✗	10834.09	0.03727081	-21664.19	$X=0.4585694$ +- 0.0009574859	$\sigma=0.1167269$ +- 0.0006770448

Esta é, igualmente, a razão pela qual o critério de informação AIC resultou em aproximadamente $-2 * \ln(L(x; \theta))$, uma vez que a penalização de parâmetros não provoca qualquer efeito quando a maioria das densidades é superior a 1. Para o teste *Kolmogorov-Smirnov* resultaram



distâncias de 0.0337 e 0.0373 para, respectivamente, distribuição beta e normal. As funções de densidade de probabilidade consideradas são visíveis na figura 3.11.

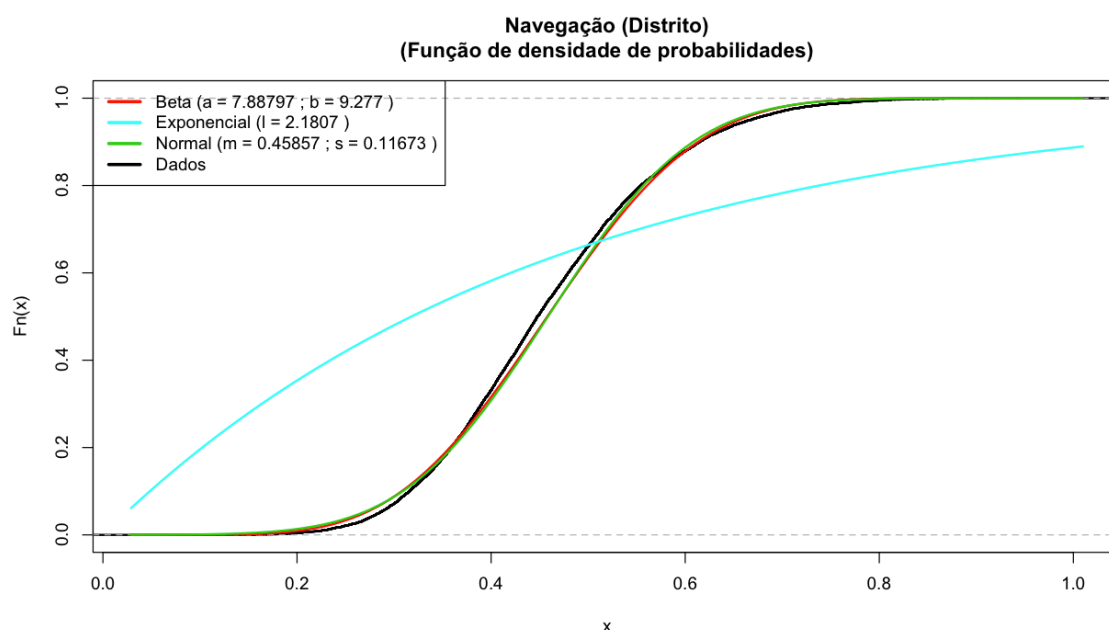


Figura 3.11 – Função de densidade de probabilidade dos dados reais e da estimativa de $V_{NAV}(automóveis, distrito)$

Destes dois testes resulta uma escolha consensual: a distribuição $beta(7.88797, 9.27770)$ é a que representa melhor as pontuações da variável $V_{NAV}(automóveis, distrito)$. Como tal, será a referência para a análise em tempo real.

Infelizmente, nem sempre estes dois testes resultam na mesma escolha. Após inúmeras análises o teste K-S afigura-se como o mais eficiente para o contexto desta solução, pelo que será o escolhido em casa de discórdia. Em anexo pode ser consultada a documentação gerada, onde é visível a eficiência deste processo mesmo perante distribuições mais incomuns.

3.6. Cálculo de *p-values*

Uma vez estimada a distribuição que representa as pontuações habituais das nossas variáveis de análise poderemos, em tempo real, avaliar a normalidade do comportamento de um utilizador segundo o teste de hipóteses anteriormente enunciado.

Para a solução desenvolvida assumiu-se um grau de confiança de 95%. O teste a realizar pode ser direccional ou não direccional, segundo o pré-estabelecido como sendo pontuações suspeitas (tabela 3.1). Por exemplo, para a variável V_{CTR} o teste é direccional à direita, enquanto que para



a V_{IMV} é não direcional.

Para o cálculo dos *p-values* assumiu-se uma implementação semelhante à disponibilizada por Hossein Arsham (<http://home.ubalt.edu/ntsbarsh/Business-stat/otherapplets/pvalues.htm>). Desenvolvido em javascript, utiliza alguns valores previamente computados para garantir brevidade no cálculo e demonstra uma precisão assinalável (erros na ordem dos 10^{-5}).

A hipótese nula é rejeitada se a pontuação em causa tiver um *p-value* inferior a 0.05. Nestas circunstâncias o *site* de classificados irá entrar em modo de segurança e começará a testar o comportamento do utilizador através da utilização de armadilhas.

3.7. Armadilhas

Com os processos descritos somos já capazes de identificar situações de suspeita de fraude com base nos comportamentos normais dos restantes utilizadores. No entanto, o objectivo é validar estas suspeitas com provas mais evidentes de que o comportamento desviante é realizado com intuito.

Cada uma das variáveis de análise utilizadas foi idealizada no sentido de supervisionar o comportamento de um utilizador. Desta forma, quando as pontuações dessas variáveis atingem valores significativos o *site* de classificados entra em modo de segurança. Deve-se lembrar que, até esse instante e em modo normal, o objectivo do motor de publicidade é canalizar para o utilizador os anúncios com mais potencial de lucro. No entanto, em modo de segurança o objectivo é criar cenários adversos à continuidade da actividade até ai demonstrada pelo utilizador. Se mesmo assim persistir, assume-se tratar-se de um comportamento consciente e lesivo para o sistema PPC e considera-se o utilizador como fraudulento. Para tal, todas as variáveis de análise tem a si associado um conjunto de alterações previamente estabelecidas, denominadas de armadilhas.

A notação adoptada para as armadilhas é $Arm_i(n, t)$, onde i é a sigla da variável de análise, n o número de vezes que a armadilha pode ser despoletada antes de o utilizador ser considerado fraudulento e t a duração em segundos da armadilha. Assim, para $n = 0$ e $t = 60$ podemos afirmar que a armadilha irá permanecer activa durante 60 segundos e que ao primeiro clique o utilizador é considerado fraudulento. Em cada instante o utilizador apenas se cruza com um tipo de armadilha. Se a ultrapassar com sucesso, então serão utilizadas as restantes armadilhas que



entretanto tenham sido activadas. A tabela 3.6 resume a informação relativa às armadilhas consideradas.

Tabela 3.6 – Descrição das armadilhas idealizadas e do comportamento esperado

Armadilha	Pontuação Suspeita	Armadilha	Comportamento Esperado
$Arm_{CAT}(n, t)$	Demasiado elevada	Retirar as hiperligações secundárias ou impressão de anúncios referentes a categorias não visitadas (acesso limitado à página principal)	Ausência, durante t segundos, de visualização de anúncios de categorias ainda não visitadas. Pressupõe-se que quando se requisita uma categoria, visualiza-se anúncios dessa categoria, i.e. não se altera de categoria de forma contínua.
$Arm_{CTR}(n, t)$	Demasiado elevada	Impressão de anúncios com baixa relação e sem imagem associada ou de anúncios já visualizados	Ausência de clique em anúncios sem imagem e pouco correlacionados com o utilizador ou nos anúncios já visualizados. Pressupõe-se que uma navegação de um utilizador cuidado e interessado evitará tais anúncios.
$Arm_{VID}(n, t)$	Demasiado elevada ou 3 pedidos duplicados consecutivos	Alteração da posição dos últimos anúncios que produziram cliques duplicados.	Ausência, durante t segundos, de cliques nos anúncios que tem sido alvo de pedidos duplicados. Pressupõe-se que um utilizador normal evitará a sua visualização pela enésima vez.
$Arm_{IMD}(n, t)$	Demasiado elevada	Redução do número de resultados para metade numa primeira fase e posteriormente para zero.	Ausência de duplicação de impressões durante t segundos.
$Arm_{PEC}(n, t)$	Demasiado elevada	Impressão de anúncios sem interesse para o utilizador e que possua baixa (ou nula) taxa de contacto.	Ausência de pedidos de contacto nos anúncios com baixa taxa de contacto.
$Arm_{IMV}(n, t)$	Demasiado elevada ou demasiado reduzida	Introdução de imagens falsas e não relacionadas em anúncios visualizados (para pontuações elevadas) ou impressão de anúncios com alta taxa de visualização de imagens (para pontuações reduzidas).	Ausência de visualização de imagens nos anúncios visados. Pressupõe-se que imagens falsas e correlacionadas não geram interesse e que anúncios com alta taxa de visualização de imagens conduzem o utilizador à abertura de imagens.
$Arm_{TEU}(n, t)$	Demasiado reduzida	Impressão de anúncios com taxas de tempo de utilização elevado e alteração do formato de apresentação dos anúncios (expectável maior tempo de	Aumento do tempo utilizado na observação do anúncio. Pressupõe-se que anúncios em que estatisticamente se dispensa mais tempo e novos formatos de apresentação cativem o utilizador, dispensando mais tempo.



		visualização na leitura no novo formato)	
$Arm_{DIV(x,y)}(n, t)$	Demasiado elevada	Impressão de um número reduzido de anúncios da secção x e com gama de valores de y ainda não visualizados.	Ausência de pesquisa e visualização de anúncios da secção x com valores de y não visualizados. Pressupõe-se que a procura de tais anúncios em t visa aumentar a diversidade dos valores visualizados.
$Arm_{NAV(x,y)}(n, t)$	Demasiado reduzida	Impressão reduzida, mas não nula, de anúncios da secção x e com baixa correlação com os valores de y já visitados.	Ausência de cliques nos anúncios visados. Pressupõe-se que sendo os menos correlacionados não deveriam gerar cliques com facilidade.
$Arm_{REL(x,y)}(n, t)$	Demasiado reduzida	Impressão elevada de anúncios da secção x com valores de y que permitam aumentar $V_{REL(x,y)}$.	Aumento de $V_{REL(x,y)}$ uma vez que aumentamos o número de anúncios que possuem valores de y pela qual o utilizador demonstrou interesse.

Como já referenciado, V_{CAT} é responsável por controlar o rácio de categorias visitadas. Quando a sua pontuação atinge valores suspeitos, o motor de publicidade retira todas as referências a hiperligações que dão acesso a anúncios de categorias não visitadas. A visualização de outras categorias torna-se menos acessível, sendo apenas possível através da página principal. Se nestas condições o utilizador aceder a outras n categorias num espaço de t segundos será considerado fraudulento (Arm_{CAT}).

O controlo da variável V_{CTR} (i.e. Arm_{CTR}) é realizado por impressão de anúncios pouco relacionados e sem qualquer imagem associada ou por anúncios já visualizados, o que estatisticamente reduz consideravelmente a probabilidade de clique. Sendo o comportamento do utilizador o oposto, i.e. taxa de clique elevada, estaremos a testar o seu interesse real na selecção de anúncios. Quanto à Arm_{VID} , sempre que as visualizações duplicadas alcancem valores suspeitos os últimos anúncios visados terão a sua posição na lista de resultados alterada. Neste caso, se o utilizador os visualizar novamente nas novas posições fica patente o seu interesse em gerar novo clique nesse mesmo anúncio.

As variáveis V_{IMD} e V_{PEC} visam a monitorização das situações onde se gera visualizações ou acções de conversão acima de um nível normal. No sentido de testar estas situações as armadilhas Arm_{IMD} e Arm_{PEC} foram concebidas. A primeira reduz o número de resultados para metade quando o nível de impressões duplicadas é suspeita e, momentos depois, se as duplicações prosseguirem, para zero. Atingida esta situação o utilizador não terá interesse em



continuar a fazer pedidos de uma página que não tem resultados. Se o seu intuito for realmente o aumento de impressões, irá procurar novos anúncios e prosseguir com o comportamento anterior, despoletando a Arm_{IMD} . A outra armadilha, Arm_{PEC} , utiliza anúncios não relacionados e com taxas de contacto baixas (ou nulas) para testar a coerência nos pedidos de contacto do utilizador.

As restantes armadilhas Arm_{IMV} , Arm_{TEU} , $Arm_{DIV(x)}$, $Arm_{NAV(x)}$ e $Arm_{REL(x)}$ utilizam abordagens semelhantes às até aqui descritas. No próximo capítulo apresentamos o protótipo e os resultados práticos obtidos perante alguns cenários reais.



Capítulo

4. Resultados



Todas as variáveis de análise foram implementadas com êxito e caracterizam, tal como idealizado, os movimentos do utilizador. É necessário um número mínimo de cliques de modo a colectar alguma informação antes de iniciar a análise. Caso contrário, a pontuação inicial de algumas variáveis activaria as respectivas armadilhas desde do primeiro instante (e.g. V_{IMV} inicia a 0, sendo um valor suspeito). Esse valor mínimo foi definido para 10 cliques e, até que seja atingido, obteremos uma contagem decrescente no painel de análise que nos indica o número de cliques em falta para iniciar o processo de detecção de fraude.

Após esse momento, é obtida a seguinte informação para cada variável: pontuação e os rácios que a originam, a estimativa para a distribuição de pontuação segundo os dados dos restantes utilizadores, o *p-value* da pontuação actual com a indicação do tipo de teste (direccional à esquerda ou à direita) e, por último, o estado das armadilhas. Adicionalmente, para as variáveis $V_{DIV}(x, y)$, $V_{NAV}(x, y)$ e $V_{REL}(x, y)$ são mantidas as tabelas de transição, i.e. a ordem de visita e gama de valores de y visualizados na secção x .

A figura 4.2 apresenta essa informação para um utilizador que se suspeita ser fraudulento devido ao reduzido tempo despendido nos classificados automóveis. De uma forma geral trata-se de um utilizador com baixa taxa de clique (V_{CTR}), sem pedidos de contacto (V_{PEC}) e com pouca apetência para a visualização de fotos nos anúncios (V_{IMV}). Gerou uma visualização duplicada, V_{VID} , não sendo a mesma significativa no conjunto dos 15 cliques realizados. No entanto apresenta uma pontuação de V_{TEU} (alusiva ao tempo despendido nos anúncios) demasiado reduzida, pelo que a armadilha Arm_{TEU} foi activada. Uma vez que a mesma ainda não foi despoletada o utilizador mantém-se apenas como suspeito. O regresso à classificação de utilizador normal (ou a passagem a utilizador fraudulento) é garantido se dispensar mais tempo nos próximos anúncios de modo a elevar a pontuação (ou se despoletar a armadilha).

É igualmente possível afirmar que a marca de automóveis *Toyota* aparenta ser a preferida (8 visualizações em 13 impressões) e que a única visualização da marca *Audi* não foi considerada significativa no conjunto das 20 impressões. Deste modo, V_{NAV} foi calculada apenas com recurso às marcas *Toyota*, *Volkswagen* e *Cadillac*. É igualmente perceptível que não existe preferências evidentes ao nível do distrito ($V_{DIV}(automoveis, distrito)$), o que deixa o utilizador no limite da suspeita apesar de as escolhas - Aveiro, Porto e Braga - estarem altamente correlacionadas ($V_{REL}(automoveis, distrito)$).

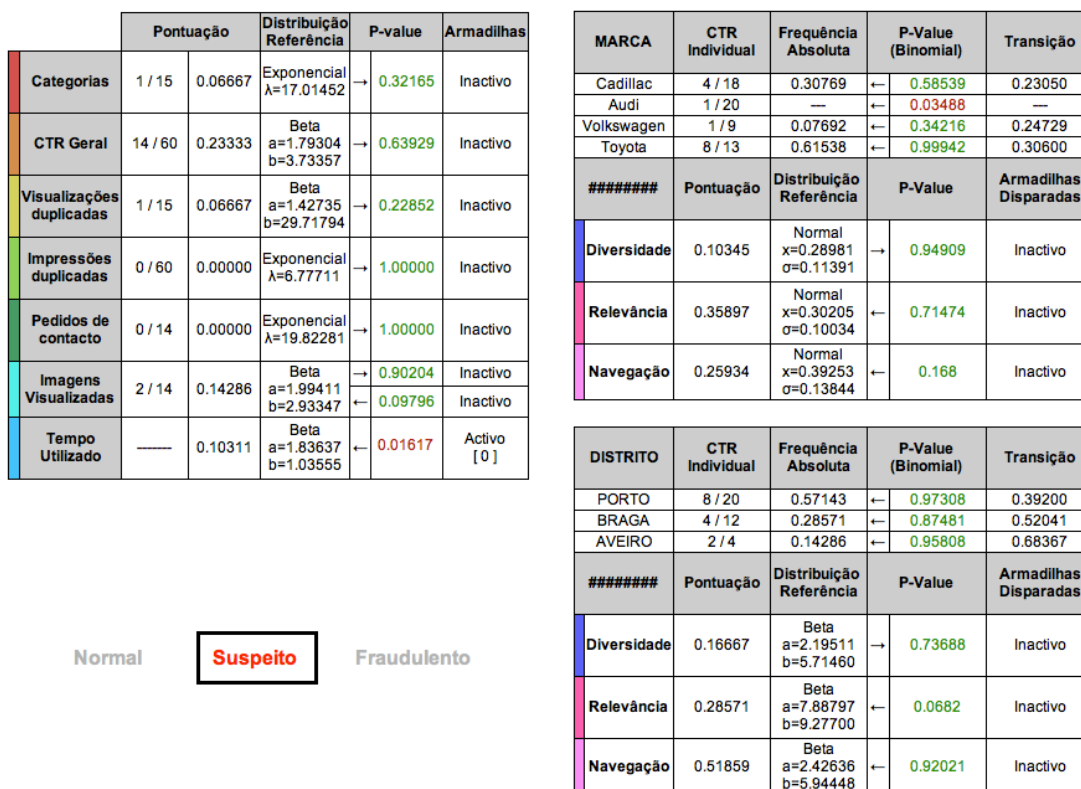


Figura 4.2 - Informação detalhada sobre as variáveis de análise de um utilizador suspeito

Para demonstrar que uma fração das situações fraudulentas existentes bem como o comportamento dos utilizadores podem ser eficientemente identificados reproduzimos cenários tipicamente suspeitos no protótipo desenvolvido. Estes cenários são citados de forma permanente ao longo da bibliografia como sendo padrões de utilização associados a esquemas fraudulentos:

- Cenário 1 – Visualizações duplicadas: O utilizador duplica a visualização de um ou mais anúncios de modo a eliminar dos classificados os anunciantes rivais. Pode fazê-lo de forma constante (e.g. actualizar a página que contém o anúncio) ou de forma aleatória (e.g. intercalando as visualizações). Neste cenário serão assumidos níveis de CTR reduzidos para dificultar a sua detecção;
- Cenário 2 – Visualização aleatória e em quantidade moderada: O utilizador visualiza os anúncios de uma forma aleatória, tendo por objectivo maximizar o maior número de visualizações. No entanto tenta manter um rácio de impressões e visualizações baixo para não levantar suspeitas;



- Cenário 3 – Visualização não aleatória e em quantidade elevada: O utilizador visualiza a generalidade dos anúncios impressos, realizando pesquisas e filtragem de resultados para camuflar a sua actividade e torna-la mais natural.

Devido ao custo temporal associado à programação de cada armadilha, apenas foram implementadas duas das dez armadilhas apresentadas: Arm_{CTR} e Arm_{VID} . Consequentemente, até ao momento, estas serão as únicas armadilhas capazes de validar as suspeitas existentes.

Nas próximas secções analisamos os cenários enumerados passo a passo. O principal objectivo é demonstrar de uma forma mais prática quando, como e porquê as armadilhas implementadas podem ser úteis no contexto PPC. Os valores assumidos para estes cenários foram $x = \text{automóveis}$ e $y = \{\text{marca}, \text{distrito}\}$ para as variáveis de análise e $n = 0$ e $t = 60s$ para as armadilhas.

4.2. Visualizações Duplicadas

Segundo a literatura, tal como a taxa de visualizações, os pedidos em duplicado é um dos principais indícios de fraude em modelos PPC. O exemplo mais sonante é a do anunciante que tenta remover anúncios rivais, esgotando o crédito dos anunciantes que os publicaram. É este o exemplo que utilizamos para demonstrar o modo como a armadilha Arm_{VID} reage a este cenário. Para facilitar a descrição não abordamos as variáveis de análise que tem impacto nulo no desenrolar deste teste. Considere-se a figura 4.3.

O perfil do utilizador que realiza este tipo de fraude é normalmente caracterizado pelo elevado conhecimento que possui do modelo PPC e das estratégias de defesa implementadas. Para enriquecer o teste, camuflamos a actividade do utilizador mantendo a taxa de visualizações reduzida. Tal situação foi alcançada por visualização inicial de anúncios diferentes dos que pretendíamos destronar do topo (i.e. V_{CTR} reduzida), sem visualizações replicadas (i.e. V_{VID} reduzida) e com um tempo de visualização considerável (i.e. V_{TEU} reduzida).

Após 78 impressões e 20 visualizações, o utilizador realiza a primeira duplicação. Para o 21º anúncio são realizados 4 pedidos: um não duplicado (i.e. válido) e três duplicados (i.e. inválidos). Nestas circunstâncias a pontuação de V_{VID} torna-se suspeita segundo a distribuição estimada $x \sim \text{beta}(1.42735, 29.71794)$ e a armadilha Arm_{VID} é accionada. Mesmo que assim não fosse,



recordamos que 3 pedidos consecutivos e duplicados iriam conduzir o utilizador para o mesmo cenário (tabela 3.6).

Uma vez accionada a Arm_{VID} , o utilizador é reencaminhado de forma forçada para a lista de resultados. No entanto o anúncio alvo de duplicação alterou a sua posição de modo a testar o objectivo do utilizador (i.e. passou da 1ª posição para a 4ª posição dos resultados). Uma vez que o utilizador manteve a sua postura, clicando novamente nesse mesmo anúncio, foi considerado fraudulento.

Esta troca de posição do anúncio duplicado pode ser igualmente útil para tentar identificar os anúncios que o utilizador está a tentar elevar no *ranking* de classificados. Nessas circunstâncias, poderíamos tentar apurar quais os anunciantes que estão a ser beneficiados (ou não afectados) por fraude, uma vez que podem estar envolvidos nos esquemas fraudulentos.

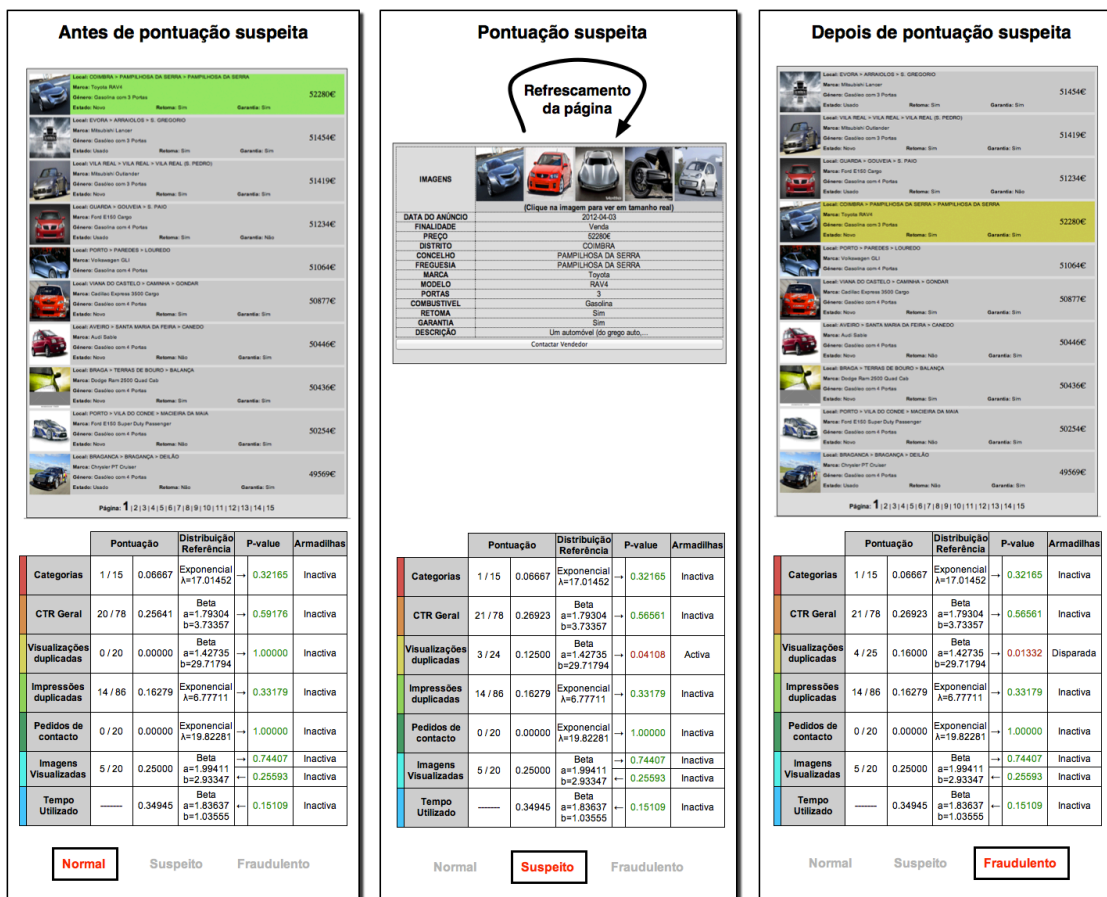


Figura 4.3 – Modo de operar da Arm_{VID} perante um cenário suspeito



4.3. Visualização aleatória e em quantidade moderada

Extremamente comum é, igualmente, a navegação sem critério. Nos utilizadores mais experientes este comportamento é acompanhado da inexistência de duplicados e de taxas de cliques reduzida de modo a não levantar suspeitas (i.e. V_{VID} e V_{CTR} reduzida). Nestes termos, gerar um lucro considerável é temporalmente dispendioso devido ao número de impressões necessárias para atenuar o alto número de visualizações pretendidas. O exemplo ilustrado contém um total de 20 visualizações em 104 impressões (Figura 4.4).

Para este cenário é possível afirmar que o utilizador realizou visualizações em apenas 6 marcas de automóveis das cerca de 30 existentes no protótipo, garantindo-lhe uma pontuação de $V_{DIV(automoveis,marca)}$ normal. No entanto, para os anúncios visualizados não são visíveis preferências evidentes ($V_{REL(automoveis,marca)}$) nem uma relação entre as marcas de automóveis visualizados ($V_{NAV(automoveis,marca)}$). Este último facto é de tal modo discrepante em relação ao comportamento dos restantes utilizadores que a Arm_{NAV} é accionada. Quanto ao distrito, a situação inverte-se ligeiramente. Existe uma correlação entre os distritos visitados, maioritariamente litoral norte, mas mantém-se a dificuldade em identificar uma preferência óbvia ($V_{REL(automoveis,distrito)}$) e uma diversidade excessiva nos anúncios visualizados, ($V_{DIV(automoveis,distrito)}$).

MARCA	CTR Individual	Frequência Absoluta	P-Value (Binomial)	Transição
Audi	3 / 15	0.15000	← 0.36581	0.15750
BMW	6 / 9	0.30000	← 0.99736	0.05900
Mitsubishi	2 / 3	0.10000	← 0.97856	0.20700
Volkswagen	3 / 9	0.15000	← 0.77872	0.13200
Toyota	2 / 3	0.10000	← 0.97856	0.64407
Ford	4 / 8	0.20000	← 0.95762	0.09179
#####	Pontuação	Distribuição Referência	P-Value	Armadilhas Disparadas
Diversidade	0.20690	Normal $\mu=0.28981$ $\sigma=0.11391$	→ 0.76667	Inactiva
Relevância	0.21667	Normal $\mu=0.30205$ $\sigma=0.10034$	← 0.1974	Inactiva
Navegação	0.15706	Normal $\mu=0.39253$ $\sigma=0.13844$	← 0.04448	Activa

DISTRITO	CTR Individual	Frequência Absoluta	P-Value (Binomial)	Transição
AVEIRO	3 / 6	0.15000	← 0.94579	0.37350
VIANA DO CASTELO	1 / 4	0.05000	← 0.69064	0.14700
BRAGA	2 / 13	0.10000	← 0.25511	0.98299
EVORA	1 / 2	0.05000	← 0.92283	0.23250
PORTO	5 / 9	0.25000	← 0.98282	0.92517
LISBOA	1 / 3	0.05000	← 0.81138	0.40430
VILA REAL	4 / 8	0.20000	← 0.95762	0.60544
GUARDA	2 / 8	0.10000	← 0.60838	0.44321
VISEU	1 / 6	0.05000	← 0.46940	0.94444
#####	Pontuação	Distribuição Referência	P-Value	Armadilhas Disparadas
Diversidade	0.50000	Beta $a=2.19511$ $b=6.71460$	→ 0.08858	Inactiva
Relevância	0.33333	Beta $a=7.88797$ $b=9.27700$	← 0.14603	Inactiva
Navegação	0.47319	Beta $a=2.42636$ $b=5.94448$	← 0.87619	Inactiva

Figura 4.4 – Modo de operar da Arm_{NAV} perante um cenário suspeito

Uma vez que o comportamento em classificados *online* é bastante orientada, fica clara a convicção que a falta de critério ou dificuldade em identificar preferências do utilizador pode ser utilizada no sentido de testar o seu comportamento. Se $Arm_{NAV(automoveis,marca)}$ se encontra



se implementada (relembra-se que, actualmente, apenas estão operacionais a Arm_{CTR} e a Arm_{VID}) iniciaria a impressão de alguns anúncios automóveis onde a marca do veículo era pouco relacionada com as marcas já visitadas: *Audi*, *BMW*, *Mitsubishi*, *Volkswagen*, *Toyota* e *Ford*. Assume-se que tratando-se de um utilizador honesto e interessado, não irá visualizar tais anúncios uma vez que não se relacionam com os até aqui visualizados, optando pelos restantes.

4.4. Visualização não aleatória e em quantidade elevada

Introduz-se agora o cenário oposto ao anteriormente apresentado. Neste caso, a navegação é realizada com muito critério de modo a possibilitar um aumento da taxa de cliques, gerando maior lucro em menos tempo. Os utilizadores mais experientes utilizam os filtros de resultados ou as pesquisas para que as visualizações geradas sejam sempre bastante relacionadas.

A figura 4.5 demonstra a tipologia dos anúncios visualizados para este exemplo, confirmando a coerência da navegação do utilizador. Apenas foram visitados 5 marcas de automóveis de 6 distritos distintos, o que garante desde de logo baixa diversidade. Ambos os atributos apresentam sinais de preferência (*Audi* e *Toyota* dos distritos do *Porto* e de *Braga*) e estão claramente relacionados.

Após 23 cliques por entre 5 marcas distintas, o utilizador é classificado como suspeito devido à elevada taxa de visualizações apresentada (24 em 36 impressões), i.e. a pontuação de V_{CTR} .

MARCA	CTR Individual	Frequência Absoluta	P-Value (Binomial)	Transição
Toyota	6 / 10	0.26087	→ 0.51622	0.30600
Kia	4 / 7	0.17391	→ 0.49315	0.54739
Nissan	4 / 8	0.17391	→ 0.31771	0.27394
Audi	8 / 10	0.34783	→ 0.92463	0.59223
Ferrari	1 / 1	0.04348	→ 1.00000	0.26205
#####	Pontuação	Distribuição Referência	P-Value	Armadilhas Disparadas
Diversidade	0.17241	Normal x=0.26981 σ=0.11391	→ 0.84865	Inactiva
Relevância	0.27826	Normal x=0.30205 σ=0.10034	→ 0.40629	Inactiva
Navegação	0.37197	Normal x=0.39253 σ=0.13844	→ 0.44097	Inactiva

DISTRITO	CTR Individual	Frequência Absoluta	P-Value (Binomial)	Transição
LISBOA	2 / 3	0.08696	→ 0.73922	0.11400
PORTO	10 / 19	0.43478	→ 0.21463	0.39200
BRAGA	5 / 6	0.21739	→ 0.93199	0.52041
AVEIRO	3 / 4	0.13043	→ 0.83338	0.68367
VIANA DO CASTELO	2 / 2	0.08696	→ 1.00000	0.61752
GUARDA	1 / 1	0.04348	→ 1.00000	0.21350
#####	Pontuação	Distribuição Referência	P-Value	Armadilhas Disparadas
Diversidade	0.33333	Beta a=2.19511 b=5.71460	→ 0.32765	Inactiva
Relevância	0.39855	Beta a=7.88797 b=9.27700	→ 0.31143	Inactiva
Navegação	0.35774	Beta a=2.42636 b=5.94448	→ 0.69552	Inactiva

Figura 4.5 – Variáveis de análise de um utilizador com navegação criteriosa

Uma vez activada, a armadilha Arm_{CTR} prepara a próxima listagem de resultados para testar o utilizador. São incorporados 2 anúncios sem imagem de apresentação e com baixa correlação ou, alternativamente, anúncios já visualizados. Perante a navegação bastante coerente do



utilizador até então, espera-se que estes anúncios de baixo interesse não sejam visualizados. Se, por outro lado, o utilizador lhes clicar será classificado de fraudulento.

Neste exemplo, onde o utilizador realizou um clique num desses anúncios, deve-se apenas elucidar para o facto de ter sido considerado fraudulento sem possuir nesse instante qualquer pontuação suspeita. No momento em que o utilizador realiza a visualização que coloca a V_{CTR} em patamares suspeitos a armadilha Arm_{CTR} é activada (Figura 4.6 - centro). Assim, independentemente do que possa acontecer nos momentos seguintes o próximo conjunto de 10 anúncios terá inequivocamente dois anúncios associados à armadilha Arm_{CTR} . No momento que é requisitada mais uma página de resultados a quantidade de impressões totais aumenta e, conseqüentemente, a pontuação de V_{CTR} diminuiu (Figura 4.6 -direita). Esta diminuição foi o suficiente para conduzir a pontuação para níveis normais, embora a armadilha se mantenha activa por 60 segundos.

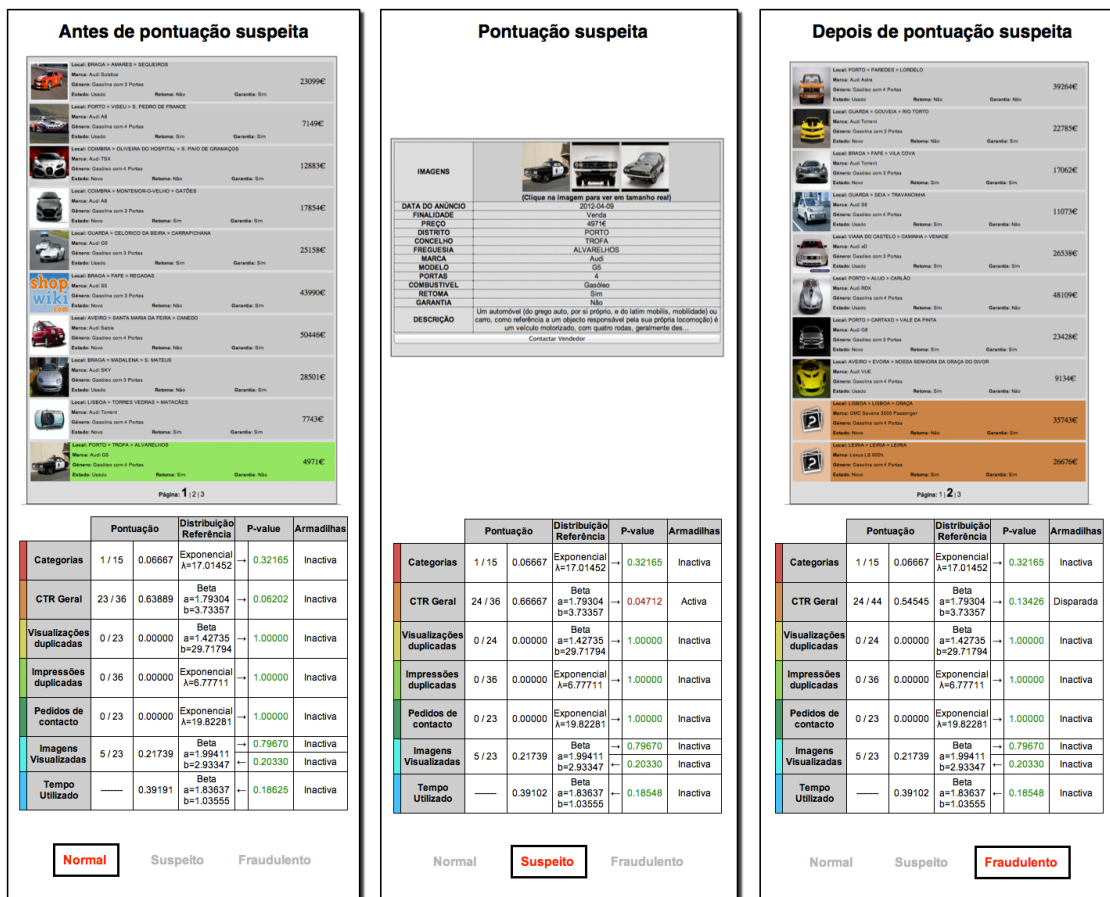


Figura 4.6 - Modo de operar da Arm_{CTR} perante um cenário suspeito



Capítulo

5. Conclusões



5.1. Discussão

Em virtude do que foi mencionado ao longo do documento somos obrigados a concluir que não existe uma solução única e infalível para a questão de fraude em sistemas PPC. Por um lado, o número de cenários passíveis de fraude e a sua complexidade crescem continuamente. Por outro, a subjectividade da avaliação de comportamentos humanos é incontornável. Mesmo perante estes factos, o modelo de publicidade baseado em pagamentos por clique prevê-se que continue a ter o nível de sucesso até aqui demonstrado.

Neste documento apresentamos uma proposta, dotada de aprendizagem automática, que visa detectar e validar em tempo real a generalidade dos casos suspeitos de fraude em anúncios classificados. Para tal, são utilizados dados relativos à navegação ou às preferências de utilizadores. Esses dados são fundamentais para a obtenção de padrões de utilização que, para este contexto, definem o comportamento comum e esperado de um utilizador.

Foram utilizadas duas ferramentas de uso livre para a obtenção e análise destes padrões: CAREN (<http://www.di.uminho.pt/pja/class/caren.html>) e R (<http://www.r-project.org>). O primeiro é responsável pela derivação de regras de associação que relacionem os diversos anúncios visualizados por cada utilizador, enquanto o segundo é usado para estimar as distribuições de pontuações das diversas variáveis de análise. Por variáveis de análise entenda-se formas de representar numericamente - valor no intervalo $[0,1]$ e denominado de pontuação - o comportamento de um utilizador. Ambas as operações são realizadas em segundo plano e de forma periódica de modo a garantir que os padrões mantêm-se actuais e representativos. Através da materialização dos resultados obtidos pelas duas ferramentas reduzimos a carga computacional necessária em tempo real, não colocando em causa o tempo de resposta dos servidores ou do sistema de recomendação.

Uma vez estimada a distribuição de pontuações dos utilizadores para as diversas variáveis de análise, garantimos o suporte necessário à execução em tempo real. Um utilizador é classificado de suspeito se alguma das suas pontuações, em dado instante, se desviar de forma significativa do comportamento esperado. A determinação deste desvio é obtida por teste de hipóteses, para um definido grau de confiança α e segundo as distribuições estimadas para cada variável de análise.



A proposta contempla ainda a tentativa de validação das suspeitas. Nesse sentido, o motor de publicidade altera o seu modo de operar quando existem suspeitas, colocando à prova o utilizador através de cenários que contrariam o comportamento até ai demonstrado. O principal objectivo é atestar o intuito do utilizador, dificultando-lhe a realização de cliques que o conduzam para um estado ainda mais suspeito. Se, ainda assim, o comportamento do utilizar se mantiver perante estas condições adversas, o utilizador será classificado de fraudulento.

Os resultados obtidos em contexto experimental, através de um protótipo desenvolvido para o efeito, permitem-nos concluir que a combinação de padrões de utilização obtidos por técnicas de mineração de dados e uma análise estatística criteriosa são fundamentais para o sucesso deste tipo de solução. Com a arquitectura apresentada, a viabilidade de execução em tempo real e a obtenção de resultados consideráveis parece-nos encaminhada.

O contexto de aplicação não está restrito apenas aos cenários expostos. Embora vise objectivamente a identificação de fraude do tipo II, a mesma poderá ser competente na identificação de fraude de tipo III. Para tal, basta que o tráfego partilhado por cada utilizador seja em quantidade suficiente de modo a permitir a detecção e validação de comportamentos incomuns de forma individual. Por outro lado, com algumas alterações é um facto, pode-se transportar esta proposta para modelos de publicidade PPC de 4 intervenientes (i.e. obriga que o editor recolha e forneça dados dos utilizadores) e para modelos que não sejam orientados ao clique (i.e. CPM e CPA).

5.1.1. Limitações

A avaliação de comportamentos humanos é por si só uma temática sensível mas torna-se ainda mais susceptível a erros quando avaliada através de dados de utilização na internet. Tal facto contribui, infelizmente, para parte das limitações que a solução apresentada possui. A primeira dessas limitações é transversal a todas as propostas que conhecemos: identificar de forma eficiente um utilizador. Não sendo o âmbito deste projecto encontrar a melhor estratégia para o efeito, optou-se pela utilização da sessão de *browser*. No entanto, não poderemos deixar de assumir que se trata de uma opção demasiado débil apesar de não colocar grandes entraves à privacidade do utilizador.



A suposição de que o comportamento actual de um dado utilizador deve seguir, ainda que com algum desvio, o comportamento histórico dos vários utilizadores é outra das restrições. No entanto, ao contrário da anterior, não é possível ter outra abordagem que nos garanta melhores resultados para a definição de normalidade em modelos PPC.

A utilização de regras de associação para extrair conhecimento dos dados colectados verificou-se acertada mas é necessário considerar mais atributos no momento de analisar o comportamento dos utilizadores. O preço dos artigos visualizados, as pesquisas realizadas ou a alteração brusca de comportamento em relação ao perfil de curto prazo são apenas algumas das vertentes a explorar. Não nos restam dúvidas que, tal como afirmado por Teevan e Dumais (2005), quando maior a variedade de informação potencialmente melhor será a análise e as conclusões obtidas. Adicionalmente, o facto de termos considerado a ordem dos acontecimentos relevante pode em alguns casos conduzir-nos a falsos positivos, uma vez que nem sempre os utilizadores visualizam os anúncios pretendidos do mesmo modo ou na mesma ordem. No entanto, preferimos assumir essa condição (i.e. $A \rightarrow C$ pode ser diferente de $C \rightarrow A$) e utilizar a confiança das regras do que assumir que o contrário (i.e. $A \rightarrow C = C \rightarrow A$) e utilizar, por exemplo, a medida *jaccard* ou *cosine* nas regras derivadas.

A estimativa das distribuições de pontuações também apresenta lacunas. Primeiramente, é essencial considerar um maior número de distribuições e, por necessidade, normaliza-las para o intervalo $[0,1]$ e reformular o cálculo dos *p-value*. Por outro lado, a distribuição com melhor ajuste poderá ainda assim não representar de forma conveniente os dados. Nesse sentido foi utilizado o teste *Kolmogorov-Smirnov*, tendo os resultados práticos demonstrado que para amostras elevadas o teste torna-se demasiado relaxado.

Adicionalmente, a qualidade de ajuste das distribuições consideradas está sempre directamente relacionada com a própria natureza dos dados. Assim, por exemplo, se os dados mantiverem uma distribuição n-modal (e.g. bimodal ou trimodal) a qualidade do ajuste reduzirá significativamente. Embora a complexidade aumente, factor pelo qual não se explorou está temática no âmbito deste documento, poderá ser considerada a combinação de múltiplas distribuições para contornar esta limitação.

Em relação às armadilhas, é necessário obter uma visão global dos dados e alguns comportamentos comuns a vários utilizadores de forma a definir a melhor estratégia para cada



variável de análise. Para esse fim, em contextos reais, deve-se considerar igualmente as técnicas de *clustering* para obter essa mesma perspectiva. Existe ainda trabalho por desenvolver e limitações a suprimir ao nível das armadilhas. O facto de não as termos implementado na totalidade e, principalmente, não termos tido oportunidade de testar o protótipo em cenários reais limita-nos a análise à eficiência da proposta.

A discrição na apresentação dos anúncios armadilhados é igualmente fundamental e deve ser melhorado uma vez que, tal como indicado por Haddadi (2010), o utilizador não deve suspeitar em nenhum momento que o seu comportamento está a ser analisado. Para além de transmitir o nosso modo de operar junto dos utilizadores, provocaria-lhes algum constrangimento saber que o seu comportamento está a ser analisado.

Por fim, falta realizar testes de carga no sentido de validar a solução para situações em que a quantidade de acessos ao motor de publicidade é extremo.

5.1.2. Contribuições

No sentido de lidar com a problemática em discussão foram exploradas e apresentadas as principais propostas na área de detecção de fraude em sistemas PPC ou em áreas directamente relacionadas (e.g. identificação e perfis de utilizador). Os objectivos, os métodos utilizados e as suas limitações foram criteriosamente detalhados permitindo o devido enquadramento do leitor.

Apresentamos uma proposta, conjuntamente com o protótipo desenvolvido, que combina técnicas de mineração de dados e a análise estatística para a recolha de padrões de utilização em *sites* de classificados *online*. As evidências apuradas são renovadas e materializadas de forma periódica e sem qualquer intervenção humana. Desta forma, estamos perante uma proposta de aprendizagem automática e execução em tempo real para detecção e validação de situações suspeitas em modelos PPC.

Segundo o nosso conhecimento, o objectivo de validar – para além de detectar – os comportamentos desviantes é uma opção nunca abordada na literatura. À semelhança da *Google*, segundo Tuzhilin (2006), também consideramos que as visualizações devem ser validadas tendo em consideração a intenção de conversão de cada utilizador. Nesse sentido, estamos convictos que o conceito de armadilha aqui apresentado pode contribuir de forma significativa para atestar o propósito das visualizações geradas pelos utilizadores.



5.2. Trabalho Futuro

Para trabalho futuro pretende-se, de uma forma geral, suprimir as limitações anteriormente apresentadas e transportar o protótipo desenvolvido para um cenário real de modo a atestar e melhorar a sua eficiência.

A solução proposta por Spiliopoulou *et al.* (2003) para a reconstrução de sessões com base em padrões de utilização é a principal referência para substituir as sessões de *browser* utilizadas. Para melhorar os padrões derivados e consequente análise comportamental idealiza-se um estudo às pesquisas realizadas pelos utilizadores na zona de classificados, semelhante à proposta de He, Goker e Harper (2002).

Pretende-se igualmente dotar o protótipo para estimar outras distribuições comuns, tais como X^2 , *Fisher*, *T-Student*, *Poisson* ou *LogNormal*. Deseja-se eliminar a necessidade de definir o limite de *improvement* a partir do qual as regras são consideradas significativas, utilizando para tal o teste de *Fisher* igualmente disponível no CAREN.

Por fim, mas não menos importante, aperfeiçoar o sistema de armadilhas introduzido neste documento.



Bibliografia



1. Azevedo, P. J. (2003). *A Java based Apriori Implementation for Classification Purposes*. Universidade do Minho.
2. Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19, pp. 716-723.
3. BBC. (Jan de 2010). *SuperPower: Visualising the internet*. Obtido em 31 de Out de 2012, de BBC News: <http://news.bbc.co.uk/2/hi/technology/8562801.stm>
4. Berners-Lee, T., & Cailliau, R. (1990). *WorldWideWeb: Proposal for a HyperText Project*. CERN.
5. Berners-Lee, T., & Fischetti, M. (1999). *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web* (Vol. 1). Harper San Francisco.
6. Blundo, C., & Cimato, S. (2002). SAWM: a tool for secure and authenticated web metering. *Proceedings of the 14th international conference on Software engineering and knowledge engineering* (pp. 641-648). Ischia, Italy: ACM.
7. Böhm, C., Haegler, K., & Müller, N. S. (2009). CoCo: coding cost for parameter-free outlier detection. *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 149-158). New York, NY, USA: ACM.
8. Borges, J., & Levene, M. (2005). Generating dynamic higher-order markov models in web usage mining. *Proceedings of the 9th European conference on Principles and Practice of Knowledge Discovery in Databases* (pp. 34-35). Porto, Portugal: Springer-Verlag.
9. Borges, J., & Levene, M. (2008). Mining Users' Web Navigation Patterns and Predicting Their Next Step. *NATO Science for Peace and Security Series: Information and Communication Security*, 15, 45-55.
10. Caudill, E. M., & Murphy, P. E. (2000). Consumer Online Privacy: Legal and Ethical Issues. *Journal of Public Policy & Marketing*, 19 (1), 7-19.



11. Casas-Garriga, G. (2003). Discovering Unbounded Episodes in Sequential Data. *Knowledge Discovery in Databases* , 2838, 83-94.
12. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Comput. Surv.* , 41 (3), 15:1-15:58.
13. Charikar, M. S. (2002). Similarity estimation techniques from rounding algorithms. *Proceedings of the 34th annual ACM symposium on Theory of computing* (pp. 380-388). Montreal, Quebec, Canada: ACM.
14. Chen, X., & Zhang, X. (2003). A Popularity-Based Prediction Model for Web Prefetching. *Computer* , 36 (3), 63-70.
15. Chen, M., Jacob, V., Radhakrishnan, S., & Ryu, Y. (2012). The Effect of Fraud Investigation Cost on Pay-Per-Click Advertising. *WEIS - Workshop on the Economics of Information Security*. Berlin.
16. Chin Chen, C., Chang Chen, M., & Sun, Y. (2001). PVA: a self-adaptive personal view agent system. *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 257-262). San Francisco, California: ACM.
17. Eirinaki, M., & Vazirgiannis, M. (2003). Web mining for web personalization. *ACM Trans. Internet Technol.* 3, pp. 1-27. ACM.
18. El-Sayed, M., Ruiz, C., & Rundensteiner, E. A. (2004). FS-Miner: efficient and incremental mining of frequent sequence patterns in web logs. *Proceedings of the 6th annual ACM international workshop on Web information and data management* (pp. 128-135). Washington DC, USA: ACM.
19. Fayyad, U. M., Piatetsky-Shapiro, G., & Smyth, P. (1996). Advances in knowledge discovery and data mining. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy, *From data mining to knowledge discovery: an overview* (pp. 1-34). American Association for Artificial Intelligence.
20. Financial Fraud Action. (2012). *Financial Fraud Action*. Obtido em 31 de Oct de 2012, de Financial Fraud Action: <http://www.financialfraudaction.org.uk/Publications>



21. Gauch, S., Speretta, M., Chandramouli, A., & Micarelli, A. (2007). User profiles for personalized information access. In P. a. Brusilovsky, *The adaptive web* (pp. 54-89). Berlin, Heidelberg: Springer-Verlag.
22. Gao, W., & Sheng, O. R. (2004). Mining Characteristic Patterns to Identify Web Users.
23. Gentili, G., Micarelli, A., & Sciarrone, F. (2003). Infoweb: An adaptive information filtering system for the cultural heritage domain. *Applied Artificial Intelligence: An International Journal* , 17, 715-744.
24. Ghorbani, A. A., & Zhang, J. (2004). The Reconstruction of User Sessions from a Server Log Using Improved Time-oriented Heuristics. *Second Annual Conference on Communication Networks and Services Research*, (pp. 315-322).
25. Ivancsy, R., & Juhasz, S. (2007). Analysis of Web User Identification Methods. *World Academy of Science, Engineering, and Technology* (34), 34-59.
26. Interactive Advertising Bureau. (Oct de 2012). *IAB internet advertising revenue report (2012 first six months' results)*. Obtido em 31 de Oct de 2012, de IAB internet advertising revenue report (2012 first six months' results):
http://www.iab.net/media/file/IAB_Internet_Advertising_Revenue_Report_HY_2012.pdf
27. Immorlica, N. S., Jain, K., Mahdian, M., & Talwar, K. (2007). *Patente N.º 0073579*. US.
28. ITU - International Telecommunication Union. (2011). *ICT Facts and figures 2011*. Obtido em 31 de Out de 2012, de <http://www.itu.int>: <http://www.itu.int/ITU-D/ict/facts/2011/material/ICTFactsFigures2011.pdf>
29. Haddadi, H. (2010). Fighting online click-fraud using bluff ads. *SIGCOMM Comput. Commun. Rev.* , 40, 21-25.
30. He, D., Goker, A., & Harper, D. J. (2002). Combining evidence for automatic web session identification. *Inf. Process. Manage.* , 38, 727-742.



31. Juels, A., Jakobsson, M., & Jagatic, T. N. (2006). Cache Cookies for Browser Authentication. *Proceedings of the 2006 IEEE Symposium on Security and Privacy* (pp. 301-305). IEEE Computer Society.
32. Juels, A., Stamm, S., & Jakobsson, M. (2007). Combating click fraud via premium clicks. *Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium* (pp. 2:1-2:10). Boston, MA: USENIX Association.
33. Jannach, D., Zanker, M., Felfernig, A., & Friedrich, G. (2010). *Recommender Systems: an Introduction* (Vol. 1). New York: Cambridge University Press.
34. Kantardzic, M., Walgampaya, C., Wenerstrom, B., Lozitskiy, O., Higgins, S., & King, D. (2008). Improving Click Fraud Detection by Real Time Data Fusion. *Signal Processing and Information Technology. IEEE International Symposium*, (pp. 69-74). Sarajevo.
35. Kantardzic, M., Walgampaya, C., & Yamolskiy, R. (2010). Real Time Click Fraud Prevention using multi-level Data Fusion. *WCECS: World Congress on Engineering and Computer Science, 1*.
36. Kantardzic, M., Wenerstrom, B., & Walgampaya, C. (2009). Time and Space Contextual Information Improves Click Quality Estimation. *IADIS International Conference e-Commerce*.
37. Kelly, D., & Teevan, J. (2003). Implicit feedback for inferring user preference: a bibliography. *SIGIR Forum*, 37 (2), 18-28.
38. Kleinrock, L. (1961). *Information Flow in Large Communication Nets, Ph.D. Thesis Proposal*. Massachusetts Institute of Technology.
39. Knuth, D. E. (1998). *The Art of Computer Programming* (Vol. 3). Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc.
40. Krause, B., Schmitz, C., Hotho, A., & Stumme, G. (2008). The anti-social tagger: detecting spam in social bookmarking systems. *Proceedings of the 4th international workshop on Adversarial information retrieval on the web* (pp. 61-68). Beijing, China: ACM.



41. Laxman, S., & Sastry, P. S. (2006). A survey of temporal data mining. *SADHANA, Academy Proceedings in Engineering Sciences*. 31, pp. 173-198. The Indian Academy of Sciences.
42. Laxman, S., Sastry, P. S., & Unnikrishnan, K. P. (2004). Fast algorithms for frequent episode discovery in event sequences. *Proceedings of the 3rd Workshop on Mining Temporal and Sequential Data, SIGKDD*. Seattle, WA: Association for Computing Machinery, Inc.
43. Liu, F., Yu, C., & Meng, W. (2002). Personalized web search by mapping user queries to categories. *Proceedings of the eleventh international conference on Information and knowledge management* (pp. 558-565). McLean, Virginia, USA: ACM.
44. Licklider, J. C. (1962). On-line man-computer communication. *Spring Joint Computer Conference* (pp. 113-128). San Francisco, California: ACM.
45. Lieberman, H. (1995). Letizia: an agent that assists web browsing. *Proceedings of the 14th international joint conference on Artificial intelligence*. 1, pp. 924-929. Montreal, Quebec, Canada: Morgan Kaufmann Publishers Inc.
46. Lops, P., de Gemmis, M., & Semeraro, G. (2011). Recommender Systems Handbook. In P. Lops, M. de Gemmis, & G. Semeraro, *Recommender Systems Handbook* (pp. 73-105). Springer.
47. Myung, I. J. (2003). Tutorial on maximum likelihood estimation. *J. Math. Psychol.* , 47 (1), 90-100.
48. Majumdar, S., Kulkarni, D., & Ravishankar, C. (2007). Addressing Click Fraud in Content Delivery Systems. *26th IEEE International Conference on Computer Communications. IEEE*, (pp. 240-248). Anchorage, AK.
49. Mannila, H., Toivonen, H., & Verkamo, A. (1997). Discovery of Frequent Episodes in Event Sequences. *Data Mining Knowledge Discover* , 1, 259-289.
50. Mannila, H., Toivonen, H., & Verkamo, A. (1999). Rule Discovery in Telecommunication Alarm Data. *Journal of Network and Systems Management* , 7 (4), 395-423.



51. Manning, C. D., Raghavan, P., & Schtze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.
52. Metwally, A., Agrawal, D., & El Abbadi, A. (2005). Using association rules for fraud detection in web advertising networks. *Proceedings of the 31st international conference on Very large data bases* (pp. 169-180). Trondheim, Norway: VLDB Endowment.
53. Metwally, A., Agrawal, D., & El Abbadi, A. (2007). Detectives: detecting coalition hit inflation attacks in advertising networks streams. *Proceedings of the 14th international conference on World Wide Web* (pp. 12-21). New York, USA: ACM.
54. Methods, A. o. (2007). Ivancsy, R.; Juhasz, S. *World Academy of Science, Engineering, and Technology* (34), 34-59.
55. Micarelli, A., & Sciarrone, F. (2004). Anatomy and Empirical Evaluation of an Adaptive Web-Based Information Filtering System. *User Modeling and User-Adapted Interaction* , 14 (2-3), 159-200.
56. Middleton, S. E., Shadbolt, N. R., & De Roure, D. C. (2003). Capturing interest through inference and visualization: ontological user profiling in recommender systems. *Proceedings of the 2nd international conference on Knowledge capture* (pp. 62-69). Sanibel Island, FL, USA: ACM.
57. Moukas, A. G. (1997). Amalthea: Information Filtering and Discovery Using A Multiagent Evolving System. *Journal of Applied Artificial Intelligence* , 437-457.
58. Mobasher, B. (2007). Data mining for web personalization. In P. a. Brusilovsky, *The adaptive web* (pp. 90-135). Berlin: Springer-Verlag.
59. Quiroga, L. M., & Mostafa, J. (1999). Empirical evaluation of explicit versus implicit acquisition of user profiles in information filtering systems. *Proceedings of the fourth ACM conference on Digital libraries* (pp. 238-239). New York, NY, USA: ACM.
60. Pazzani, M., Muramatsu, J., & Billsus, D. (1996). Syskill & webert: Identifying interesting web sites. *Proceedings of the thirteenth national conference on Artificial intelligence. 1*, pp. 54-61. AAAI Press.



61. Pei, J., Han, J., Mortazavi-Asl, B., & Zhu, H. (2000). Mining Access Patterns Efficiently from Web Logs. In *Proceedings of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Current Issues and New Applications* (pp. 396-407). Springer-Verlag.
62. Perkowit, M., & Etzioni, O. (2000). Adaptive Web sites. *Communications of the ACM*, 43 (8), 152-158.
63. Pretschner, A., & Gauch, S. (1999). Ontology Based Personalized Search. *Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence*. Washington, DC, USA: IEEE Computer Society.
64. Sadagopan, N., & Li, J. (2010). *Patente N.º 7860870*. US.
65. Sadagopan, N., & Li, J. (2008). Characterizing typical and atypical user sessions in clickstreams. *Proceedings of the 17th international conference on World Wide Web* (pp. 885-894). Beijing, China: ACM.
66. Salton, G., & McGill, M. J. (1986). *Introduction to Modern Information Retrieval*. NY, USA: McGraw-Hill, Inc.
67. Search Engine Land. (2007). *Geolocation: Core to The Local Space And Key To Click-Fraud Detection*. Search Engine Land.com.
68. Sendhilkumar, S., & Geetha, T. V. (2008). Personalized ontology for web search personalization. *Proceedings of the 1st Bangalore Annual Compute Conference* (pp. 18:1-18:7). Bangalore, India: ACM.
69. Sieg, A., Mobasher, B., & Burke, R. (2004). Inferring user's information context from user profiles and concept hierarchies. *IFCS - International Federation of Classification Societies*, (pp. 563-574).
70. Sieg, A., Mobasher, B., & Burke, R. (2010). Improving the effectiveness of collaborative recommendation with ontology-based user profiles. *Proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems* (pp. 39-46). Barcelona, Spain: ACM.



71. Shmueli-Scheuer, M., Roitman, H., Carmel, D., Mass, Y., & Konopnicki, D. (2010). Extracting user profiles from large scale data. *Proceedings of the 2010 Workshop on Massive Data Analytics on the Cloud* (pp. 4:1-4:6). Raleigh, North Carolina: ACM.
72. Speretta, M. a. (2005). Personalized Search Based on User Search Histories. *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence* (pp. 622-628). Washington, DC, USA: IEEE Computer Society.
73. Spiliopoulou, M., Mobasher, B., Berendt, B., & Nakagawa, M. (2003). A Framework for the Evaluation of Session Reconstruction Heuristics in Web-Usage Analysis. *INFORMS J. on Computing* , 15 (2), 171-190.
74. Srivastava, J., Cooley, R., Deshpande, M., & Tan, P.-N. (2000). Web usage mining: discovery and applications of usage patterns from Web data. *SIGKDD Explor. Newsl.* , 1 (2), 12-23.
75. Reiter, M. K., Anupam, V., & Mayer, A. (1998). Detecting hit shaving in click-through payment schemes. *Proceedings of the 3rd conference on USENIX Workshop on Electronic Commerce*. 3, p. 13. Boston, Massachusetts: USENIX Association.
76. Ricci, F., Rokach, L., Shapira, B., & Kantor, P. B. (2011). *Recommender Systems Handbook* (Vol. 1). New York: Springer.
77. Tuzhilin, A. (2006). *The Lane's Gifts v. Google Report*.
78. Tan, P.-N., & Kumar, V. (2002). Discovery of Web Robot Sessions Based on their Navigational Patterns. *Data Mining Knowledge Discovery* , 6 (1), 9-35.
79. Tappenden, A. F., & Miller, J. (2009). Cookies: A deployment study and the testing implications. *ACM Trans. Web* , 3 (3), 9:1-9:49.
80. Teevan, J., & Dumais, S. T. (2005). Personalizing search via automated analysis of interests and activities. *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 449-456). ACM.
81. Venables, W., & Ripley, B. (2002). *Modern Applied Statistics with S* (Vol. 4). Springer.



82. Wu, H., Siegel, M., Yang, J., & Stiefelhagen, R. (2002). Sensor Fusion Using Dempster-Shafer Theory. *Proceedings of IEEE Instrumentation and Measurement Technology Conference*, (pp. 21-23).
83. Wilbur, K. C., & Zhu, Y. (2009). Click Fraud. *Marketing Science*, 28, 293-308.
84. Witten, I. H., Frank, E., & Hall, M. A. (2005). *Data Mining: Practical Machine Learning Tools and Techniques* (Vol. 2). Morgan Kaufmann.
85. White, R., Jose, J., & Ruthven, I. (2001). Comparing explicit and implicit feedback techniques for web retrieval. *Proceedings of the Tenth Text Retrieval Conference (TREC-10)*, (pp. 534-538).
86. Yang, Y. C. (2010). Web user behavioral profiling for user identification. *Decision Support System*, 49 (3), 261-271.
87. Ye, N. (2000). A Markov chain model of temporal behavior for anomaly detection. *The 2000 IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*. West Point, NY.



Anexos



A.1 - Exemplo de regras de associação obtidas pelo CAREN

```
> caren Basket/Carros_Distrito.basket 0.01 0.05 -s, -d -imp0.1 -ovrBasket/Carros_Distrito

Computation Time = 0 hrs 0 mts 1 secs 279 millis

Mining on dataset Basket/Carros_Distrito.basket with 18 frequent items and with 2000 transactions.
using minsup = 0.01000 minconf = 0.05000 and minimal Improvement = 0.1

Number of derived rules = 216

Sup = 0.07350 Conf = 0.97351 PORTO <- BRAGA & COIMBRA
Sup = 0.05850 Conf = 0.93600 PORTO <- VILA REAL & AVEIRO
Sup = 0.13600 Conf = 0.92517 PORTO <- VIANA DO CASTELO
Sup = 0.03350 Conf = 0.91781 PORTO <- VILA REAL & COIMBRA
Sup = 0.20400 Conf = 0.87179 PORTO <- BRAGA
Sup = 0.26800 Conf = 0.71754 PORTO <- AVEIRO
Sup = 0.22100 Conf = 0.68210 PORTO <- COIMBRA
Sup = 0.14050 Conf = 0.62584 PORTO <- VILA REAL
Sup = 0.17900 Conf = 0.61407 PORTO <- VISEU
Sup = 0.39200 Conf = 0.39200 PORTO <-
Sup = 0.01400 Conf = 1.00000 COIMBRA <- SANTAREM & PORTO
Sup = 0.03250 Conf = 0.98485 COIMBRA <- LEIRIA & BRAGA
Sup = 0.02350 Conf = 0.97917 COIMBRA <- SANTAREM & AVEIRO
Sup = 0.09600 Conf = 0.97462 COIMBRA <- LEIRIA & PORTO
Sup = 0.13700 Conf = 0.96820 COIMBRA <- LEIRIA & AVEIRO
Sup = 0.01400 Conf = 0.96552 COIMBRA <- VIANA DO CASTELO & LEIRIA
Sup = 0.17200 Conf = 0.91733 COIMBRA <- VISEU & AVEIRO
Sup = 0.09800 Conf = 0.85965 COIMBRA <- LEIRIA & VISEU
Sup = 0.30150 Conf = 0.80723 COIMBRA <- AVEIRO
Sup = 0.02250 Conf = 0.80357 COIMBRA <- VIANA DO CASTELO & VISEU
Sup = 0.13550 Conf = 0.75698 COIMBRA <- VISEU & PORTO
Sup = 0.14850 Conf = 0.70714 COIMBRA <- LEIRIA
Sup = 0.17600 Conf = 0.60377 COIMBRA <- VISEU
Sup = 0.22100 Conf = 0.56378 COIMBRA <- PORTO
Sup = 0.32400 Conf = 0.32400 COIMBRA <-
Sup = 0.01400 Conf = 1.00000 AVEIRO <- SANTAREM & PORTO
Sup = 0.09750 Conf = 0.98985 AVEIRO <- LEIRIA & PORTO
Sup = 0.03200 Conf = 0.96970 AVEIRO <- LEIRIA & BRAGA
Sup = 0.02700 Conf = 0.96429 AVEIRO <- VIANA DO CASTELO & VISEU
Sup = 0.01350 Conf = 0.93103 AVEIRO <- VIANA DO CASTELO & LEIRIA
Sup = 0.30150 Conf = 0.93056 AVEIRO <- COIMBRA
Sup = 0.09650 Conf = 0.84649 AVEIRO <- LEIRIA & VISEU
Sup = 0.14450 Conf = 0.80726 AVEIRO <- VISEU & PORTO
Sup = 0.26800 Conf = 0.68367 AVEIRO <- PORTO
Sup = 0.14150 Conf = 0.67381 AVEIRO <- LEIRIA
Sup = 0.18750 Conf = 0.64322 AVEIRO <- VISEU
Sup = 0.07750 Conf = 0.52721 AVEIRO <- VIANA DO CASTELO
Sup = 0.11400 Conf = 0.48718 AVEIRO <- BRAGA
Sup = 0.37350 Conf = 0.37350 AVEIRO <-
Sup = 0.13650 Conf = 0.97500 BEJA <- FARO
Sup = 0.20150 Conf = 0.96643 BEJA <- SETUBAL
Sup = 0.21450 Conf = 0.92258 BEJA <- EVORA
Sup = 0.10250 Conf = 0.89912 BEJA <- LISBOA
Sup = 0.06300 Conf = 0.37725 BEJA <- PORTALEGRE
Sup = 0.25100 Conf = 0.25100 BEJA <-
Sup = 0.01350 Conf = 0.96429 VISEU <- SANTAREM & PORTO
Sup = 0.01300 Conf = 0.96296 VISEU <- C BRANCO & VILA REAL
Sup = 0.03150 Conf = 0.95455 VISEU <- LEIRIA & BRAGA
Sup = 0.02550 Conf = 0.94444 VISEU <- GUARDA & BRAGA
Sup = 0.01400 Conf = 0.93333 VISEU <- GUARDA & COIMBRA
Sup = 0.01350 Conf = 0.93103 VISEU <- VIANA DO CASTELO & LEIRIA
Sup = 0.01900 Conf = 0.92683 VISEU <- GUARDA & AVEIRO
Sup = 0.02500 Conf = 0.89286 VISEU <- BRAGANCA & BRAGA
Sup = 0.01200 Conf = 0.85714 VISEU <- C BRANCO & AVEIRO
Sup = 0.03800 Conf = 0.82609 VISEU <- GUARDA & PORTO
Sup = 0.03950 Conf = 0.79000 VISEU <- BRAGANCA & PORTO
Sup = 0.07650 Conf = 0.77665 VISEU <- LEIRIA & PORTO
Sup = 0.01700 Conf = 0.70833 VISEU <- SANTAREM & AVEIRO
Sup = 0.06850 Conf = 0.68844 VISEU <- GUARDA & VILA REAL
Sup = 0.09650 Conf = 0.68198 VISEU <- LEIRIA & AVEIRO
Sup = 0.09800 Conf = 0.65993 VISEU <- LEIRIA & COIMBRA
Sup = 0.07950 Conf = 0.59551 VISEU <- BRAGANCA
Sup = 0.17600 Conf = 0.54321 VISEU <- COIMBRA
Sup = 0.11400 Conf = 0.54286 VISEU <- LEIRIA
Sup = 0.18750 Conf = 0.50201 VISEU <- AVEIRO
Sup = 0.10600 Conf = 0.49649 VISEU <- GUARDA
Sup = 0.17900 Conf = 0.45663 VISEU <- PORTO

(...)
```

Figura A.1 - Extracto parcial das regras de associação derivadas para o distrito dos anúncios visualizados



A.2 - Exemplo de matrizes de transição

###	AVEIRO	BEJA	BRAGA	BRAGANCA	C BRANCO	COIMBRA	EVORA	FARO	GUARDA	LEIRIA	LISBOA	PORTALEGRE	PORTO	SANTAREM	SETUBAL	VIANA DO CASTELO	VILA REAL	VISEU
	0.3735	0.251	0.234	0.1335	0.1505	0.324	0.2325	0.14	0.2135	0.21	0.114	0.167	0.392	0.1415	0.2085	0.147	0.2245	0.2915
AVEIRO	-	-	-	-	-	0.80723	-	-	-	0.37885	-	-	0.71734	-	-	-	-	0.50201
AVEIRO,C BRANCO	-	-	-	-	-	-	-	-	0.89286	-	-	-	-	-	-	-	-	0.85714
AVEIRO,GUARDA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.92683
AVEIRO,LEIRIA	-	-	-	-	-	0.9682	-	-	-	-	-	-	-	-	-	-	-	0.68198
AVEIRO,SANTAREM	-	-	-	-	-	0.97917	-	-	-	1.0	-	-	-	-	-	-	-	0.70833
AVEIRO,VILA REAL	-	-	0.928	-	-	-	-	-	-	-	-	-	0.936	-	-	0.848	-	-
AVEIRO,VISEU	-	-	-	-	-	0.91733	-	-	-	0.51467	-	-	-	-	-	-	-	-
BEJA	-	-	-	-	-	-	0.85458	0.54382	-	0.40837	-	-	-	-	0.80279	-	-	-
BEJA,PORTALEGRE	-	-	-	-	-	-	0.98413	-	-	0.61905	-	-	-	-	0.9127	-	-	-
BEJA,SANTAREM	-	-	-	-	-	-	-	-	-	0.77419	-	-	-	-	0.93548	-	-	-
BRAGA	0.48718	-	-	-	-	-	-	-	-	-	-	-	0.87179	-	-	0.61752	0.52991	-
BRAGA,BRAGANCA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.89286
BRAGA,COIMBRA	-	-	-	-	-	-	-	-	-	-	-	-	0.97351	-	-	-	-	-
BRAGA,GUARDA	-	-	-	0.85185	-	-	-	-	-	-	-	-	-	-	-	-	0.85185	0.94444
BRAGA,LEIRIA	0.9697	-	-	-	-	0.98485	-	-	-	-	-	-	-	-	-	-	-	0.95455
BRAGA,VILA REAL	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.71774	-	-
BRAGANCA	-	-	-	-	-	-	-	-	0.74157	-	-	-	-	-	-	-	-	0.90637
BRAGANCA,PORTO	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.79
C BRANCO	-	-	-	-	-	-	-	-	0.7608	0.37209	-	0.71096	-	0.55482	-	-	-	-
C BRANCO,EVORA	-	-	-	-	-	-	-	-	0.60606	-	0.9697	-	0.75758	-	-	-	-	-
C BRANCO,LEIRIA	-	-	-	-	-	-	-	-	-	-	0.91964	-	0.8125	-	-	-	-	-
C BRANCO,LEIRIA,VISEU	-	-	-	-	-	-	-	-	-	-	-	-	0.94286	-	-	-	-	-
C BRANCO,PORTALEGRE	-	-	-	-	-	-	-	-	0.48131	-	-	-	0.71028	-	-	-	-	-
C BRANCO,SANTAREM	-	-	-	-	-	-	-	-	-	-	0.91018	-	-	-	-	-	-	-
C BRANCO,VILA REAL	-	-	-	-	-	-	-	-	1.0	-	-	-	-	-	-	-	-	0.96296
C BRANCO,VISEU	-	-	-	-	-	-	-	0.95402	-	-	-	-	-	-	-	-	-	-
COIMBRA	0.93056	-	-	-	-	-	-	-	0.45833	-	-	0.6821	-	-	-	-	-	0.54321
COIMBRA,GUARDA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.93333
COIMBRA,LEIRIA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.65993
COIMBRA,SANTAREM	-	-	-	-	-	-	-	-	0.98413	-	-	-	-	-	-	-	-	-
COIMBRA,VILA REAL	-	-	0.90411	-	-	-	-	-	-	-	-	-	-	-	-	0.84932	-	-
EVORA	-	0.92258	-	-	-	-	-	0.49677	-	-	0.4043	0.32258	-	-	0.7914	-	-	-
EVORA,LEIRIA	-	-	-	-	0.68966	-	-	-	-	-	0.4043	0.32258	-	0.89655	-	-	-	-
EVORA,PORTALEGRE	-	-	-	-	-	-	-	-	-	-	0.52667	-	-	-	-	-	-	-
EVORA,SANTAREM	-	-	-	-	-	-	-	-	-	-	0.6375	0.775	-	-	-	-	-	-
FARO	-	0.975	-	-	-	-	0.825	-	-	0.52143	-	-	-	-	0.85357	-	-	-
FARO,PORTALEGRE	-	-	-	-	-	-	0.97015	-	-	0.8209	-	-	-	-	0.97015	-	-	-
FARO,SANTAREM	-	-	-	-	-	-	-	-	-	0.89474	-	-	-	-	0.97368	-	-	-
GUARDA	-	-	0.4637	0.5363	-	-	-	-	-	-	0.40047	-	0.33724	-	-	-	0.46604	0.49649
GUARDA,LEIRIA	-	-	-	0.88073	-	-	-	-	-	-	0.86239	-	0.83486	-	-	-	-	-
GUARDA,PORTALEGRE	-	-	-	1.0	-	-	-	-	0.54971	-	-	-	0.77193	-	-	-	-	-
GUARDA,PORTO	-	-	0.86957	-	-	-	-	-	-	-	-	-	-	-	-	-	0.88043	0.82609
GUARDA,SANTAREM	-	-	-	0.94444	-	-	-	-	-	-	0.91667	-	-	-	-	-	-	-
GUARDA,VILA REAL	-	-	0.94472	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.68844
GUARDA,VISEU	-	-	0.61321	-	-	-	-	-	-	-	-	-	-	-	-	-	0.64623	-
LEIRIA	0.67381	-	-	-	0.26667	0.70714	-	-	-	-	-	-	-	0.36667	-	-	-	0.54286
LEIRIA,LISBOA	-	-	-	-	-	-	-	-	-	-	-	-	-	1.0	-	-	-	-
LEIRIA,PORTALEGRE	-	-	-	0.92793	-	-	-	0.84685	-	-	-	-	-	0.85586	-	-	-	-
LEIRIA,PORTALEGRE,VISEU	-	-	-	-	-	-	-	-	-	-	-	-	-	0.9697	-	-	-	-
LEIRIA,PORTO	0.98985	-	-	-	-	0.97462	-	-	-	-	-	-	-	-	-	-	-	0.77665
LEIRIA,VIANA DO CASTELO	0.93103	-	-	-	-	0.96552	-	-	-	-	-	-	-	-	-	-	-	0.93103
LEIRIA,VISEU	0.84649	-	-	-	-	0.85965	-	-	-	-	-	-	-	-	-	-	-	-
LISBOA	-	0.89912	-	-	-	-	0.82456	0.64035	-	-	-	0.35526	-	0.28947	0.85965	-	-	-
LISBOA,PORTALEGRE	-	-	-	-	-	-	0.97531	-	-	-	-	-	-	-	-	-	-	-
PORTALEGRE	-	0.37725	-	-	0.64072	-	0.4491	-	0.51198	0.33234	0.24251	-	-	0.56886	0.34731	-	-	-
PORTALEGRE,SANTAREM	-	-	-	-	0.8	-	-	-	0.69474	-	-	-	-	-	-	-	-	-
PORTALEGRE,SETUBAL	-	-	-	-	-	-	1.0	-	-	-	0.66379	-	-	-	-	-	-	-
PORTALEGRE,VISEU	-	-	-	-	1.0	-	-	-	0.93182	0.75	-	-	-	0.84091	-	-	-	-
PORTO	0.68367	-	0.52041	-	-	0.56378	-	-	-	-	-	-	-	-	-	0.34694	0.35842	0.45663
PORTO,SANTAREM	1.0	-	-	-	-	1.0	-	-	-	1.0	-	-	-	-	-	-	-	0.96429
PORTO,VILA REAL	-	-	0.79359	-	-	-	-	-	-	-	-	-	-	-	-	0.60498	-	-
PORTO,VISEU	0.80726	-	-	-	-	0.75698	-	-	-	-	-	-	-	-	-	-	-	-
SANTAREM	-	-	-	-	0.59011	-	-	-	0.50883	0.54417	0.23322	0.67138	-	-	-	-	-	-
SANTAREM,SETUBAL	-	-	-	-	-	-	-	-	0.6338	0.92958	0.84127	-	-	-	-	-	-	-
SANTAREM,VISEU	-	-	-	-	-	-	-	-	0.6338	0.92958	0.84127	-	-	-	-	-	-	-
SETUBAL	-	0.96643	-	-	-	-	0.88249	0.57314	-	-	0.47002	0.27818	-	-	-	-	-	-
VIANA DO CASTELO	0.52721	-	0.98299	-	-	-	-	-	-	-	-	-	0.92517	-	-	-	-	0.60544
VIANA DO CASTELO,VISEU	0.96429	-	-	-	-	0.80357	-	-	-	-	-	-	-	-	-	-	-	-
VILA REAL	-	-	0.55234	0.53898	-	-	-	-	0.44321	-	-	-	0.62584	-	-	0.39644	-	0.42094
VILA REAL,VISEU	-	-	-	0.79894	-	-	-	-	0.72487	-	-	-	-	-	-	-	-	-
VISEU	0.64322	-	-	0.27273	-	0.60377	-	-	0.36364	0.39108	-	-	0.61407	-	-	-	-	-

Figura A.2 - Matriz de transição que representa os dados das regras de associação provenientes da figura A.1



A.3 - Exemplo do *script* R utilizado para a estimativa das distribuições de pontuações

```
# ctr
ctr = read.table("Sites/R/Scores/ctr.score")
Score <- ctr$V1
sink("Sites/R/Resultados/ctr.txt")
b <- fitdistr(ctr$V1,"beta",start=list(shape1=0.1,shape2=0.1))
e <- fitdistr(ctr$V1,"exponential")
n <- fitdistr(ctr$V1,"normal")
b$sn
b$loglik
b$estimate
b$sd
e$loglik
e$estimate
e$sd
n$loglik
n$estimate
n$sd
b_info <- if (pbeta(0,b$estimate[1],b$estimate[2])<0.01 && pbeta(1,b$estimate[1],b$estimate[2])>0.99) 1 else 0
e_info <- if (pexp(1,e$estimate[1])>0.99) 1 else 0
n_info <- if (pnorm(0,n$estimate[1],n$estimate[2])<0.01 && pnorm(1,n$estimate[1],n$estimate[2])>0.99) 1 else 0
b_info
e_info
n_info
ks_b <- ks.test(Score, "pbeta", shape1=b$estimate[1], shape2=b$estimate[2])
ks_b$statistic
ks_e <- ks.test(Score, "pexp", rate=e$estimate[1])
ks_e$statistic
ks_n <- ks.test(Score, "pnorm", mean=n$estimate[1], sd=n$estimate[2])
ks_n$statistic
AIC(b)
AIC(e)
AIC(n)
jpeg(filename="Sites/R/Gráficos/ctr.jpeg",width=1280,height=1280,quality=100)
h1 <- hist(Score,breaks = 200,xlim=c(0,1),density=100,main="CTR Geral",freq=FALSE,xaxt='n',ylab="Densidade")
axis(side=1, at=seq(0,1, 0.05), labels=seq(0,1,0.05))
curve(dbeta(x,b$estimate[1],b$estimate[2]), col = 2, lwd=3, add = TRUE)
curve(dnorm(x,n$estimate[1],n$estimate[2]), col = 3, lwd=3, add = TRUE)
curve(dexp(x,e$estimate[1]), col = 5, lwd=3, add = TRUE)
legend("top", legend=c(paste("Beta (a =",round(b$estimate[1],5),"; b =",round(b$estimate[2],5),")"), paste("Exponencial (l =",round(e$estimate[1],5),")"),paste("Normal (m =",round(n$estimate[1],5),"; s =",round(n$estimate[2],5),")"),lty=c(1, col=c(2,5,3), lwd=3, bty="o",cex=1)
dev.off()
jpeg(filename="Sites/R/Gráficos/ctr[CDF].jpeg",width=1280,height=1280,quality=100)
plot(ecdf(Score),col=1,verticals=TRUE,lwd=2)
curve(pbeta(x,b$estimate[1],b$estimate[2]),add=TRUE,col=2,lwd=2)
curve(pnorm(x,n$estimate[1],n$estimate[2]),add=TRUE,col=3,lwd=2)
curve(pexp(x,e$estimate[1]),add=TRUE,col=5,lwd=2)
legend("topleft", legend=c(paste("Beta (a =",round(b$estimate[1],5),"; b =",round(b$estimate[2],5),")"), paste("Exponencial (l =",round(e$estimate[1],5),")"),paste("Normal (m =",round(n$estimate[1],5),"; s =",round(n$estimate[2],5),")"),paste("Dados")),lty=c(1, col=c(2,5,3,1), lwd=3, bty="o",cex=1)
dev.off()
sink()
```

Figura A.3 - *Script* utilizado para a estimativa da distribuição de pontuações de V_{CTR}



A.4 - Estimativa das distribuições de pontuações (tabela sumária)

Tabela T.1 - Tabela resultante da documentação produzida pela solução

	Distribuição	$x \in [0,1]$?	Escolhido?	$L(x;\theta)$	D [KS]	AIC	1º Parametro +- Erro estimado	2º Parametro +- Erro estimado
categorias (n=9033)	Beta(A,B)	✓	✗	16332.1	0.08283649	-32660.19	a=1.036231 + 0.01362249	b=16.083189 + 0.26404226
	Exp(λ)	✓	✓	16567.13	0.08165051	-33132.25	$\lambda=17.01452 + 0.1790209$	
	Normal(X, σ)	✗	✗	11019.8	0.2258413	-22035.6	X=0.05877334 + 0.0007516744	$\sigma=0.07144071 + 0.0005315140$
contactos (n=8971)	Beta(A,B)	✓	✓	17729.88	0.07637368	-35455.76	a=1.164799 + 0.01551931	b=21.409356 + 0.34908920
	Exp(λ)	✓	✗	17823.88	0.09447405	-35645.77	$\lambda=19.82281 + 0.2092883$	
	Normal(X, σ)	✗	✗	12831.82	0.2162203	-25659.65	X=0.05044693 + 0.0006111415	$\sigma=0.05788449 + 0.0004321423$
ctr (n=9948)	Beta(A,B)	✓	✓	3449.523	0.04015392	-6895.047	a=1.793037 + 0.02361920	b=3.733570 + 0.05278867
	Exp(λ)	✗	✗	1155.396	0.1987074	-2308.792	$\lambda=3.053058 + 0.03061027$	
	Normal(X, σ)	✗	✗	3156.469	0.03357817	-6308.937	X=0.3275405 + 0.001766423	$\sigma=0.1761824 + 0.001249050$
diversidade_1 (n=10000)	Beta(A,B)	✓	✗	7114.553	0.1257039	-14225.11	a=3.632182 + 0.04932223	b=9.003467 + 0.12741192
	Exp(λ)	✗	✗	2385.286	0.3242711	-4768.571	$\lambda=3.450532 + 0.03450532$	
	Normal(X, σ)	✓	✓	7534.496	0.07726932	-15064.99	X=0.2898103 + 0.0011390527	$\sigma=0.1139053 + 0.0008054319$
diversidade_2 (n=10000)	Beta(A,B)	✓	✓	5412.808	0.0821879	-10821.62	a=2.195110 + 0.02911325	b=5.714604 + 0.08126848
	Exp(λ)	✗	✗	2802.541	0.2470646	-5603.082	$\lambda=3.597554 + 0.03597554$	
	Normal(X, σ)	✗	✗	4772.914	0.1158926	-9541.828	X=0.2779667 + 0.001501336	$\sigma=0.1501336 + 0.001061605$
imagens (n=9967)	Beta(A,B)	✓	✓	2258.959	0.03888358	-4513.919	a=1.994111 + 0.02656032	b=2.933471 + 0.04062643
	Exp(λ)	✗	✗	-1001.104	0.2354942	2004.209	$\lambda=2.458516 + 0.02462583$	
	Normal(X, σ)	✗	✗	2206.587	0.0191588	-4409.174	X=0.4067494 + 0.001942372	$\sigma=0.1939165 + 0.001373465$
impressoes_duplicadas (n=9709)	Beta(A,B)	✓	✗	8291.972	0.08203399	-16579.94	a=0.7838972 + 0.009716982	b=4.1352538 + 0.064958517
	Exp(λ)	✓	✓	8869.659	0.04767214	-17737.32	$\lambda=6.777106 + 0.06877919$	
	Normal(X, σ)	✗	✗	3797.718	0.1918384	-7591.435	X=0.1475556 + 0.001660731	$\sigma=0.1636389 + 0.001174314$
navegacao_1 (n=9996)	Beta(A,B)	✓	✗	8495.577	0.04355292	-16987.15	a=5.411856 + 0.07446054	b=12.571255 + 0.17760559
	Exp(λ)	✗	✗	1970.78	0.3340846	-3939.56	$\lambda=3.310691 + 0.03311353$	
	Normal(X, σ)	✓	✓	8798.963	0.005477097	-17593.93	X=0.3020518 + 0.0010036047	$\sigma=0.1003404 + 0.0007096557$
navegacao_2 (n=14862)	Beta(A,B)	✓	✓	10854.37	0.0336876	-21704.74	a=7.887974 + 0.09002277	b=9.276996 + 0.10638123
	Exp(λ)	✗	✗	-3274.936	0.4125389	6551.871	$\lambda=2.180695 + 0.01788777$	
	Normal(X, σ)	✓	✗	10834.09	0.03727081	-21664.19	X=0.4585694 + 0.0009574859	$\sigma=0.1167269 + 0.0006770448$
relevancia_1 (n=6046)	Beta(A,B)	✓	✗	2938.559	0.1490199	-5873.118	a=3.596188 + 0.06298018	b=5.726267 + 0.10291336
	Exp(λ)	✗	✗	-392.1684	0.3020445	786.3368	$\lambda=2.54756 + 0.0327635$	
	Normal(X, σ)	✓	✓	3375.936	0.1290935	-6747.871	X=0.3925325 + 0.001780449	$\sigma=0.1384406 + 0.001258968$
relevancia_2 (n=5216)	Beta(A,B)	✓	✓	2828.833	0.04585974	-5653.666	a=2.426363 + 0.04481645	b=5.944483 + 0.11668224
	Exp(λ)	✗	✗	1237.807	0.237454	-2473.613	$\lambda=3.446329 + 0.04771863$	
	Normal(X, σ)	✗	✗	2748.894	0.04316022	-5493.789	X=0.2901638 + 0.001977952	$\sigma=0.1428514 + 0.001398623$
tempo (n=15265)	Beta(A,B)	✓	✓	2146.768	0.06159588	-4289.536	a=1.836374 + 0.02075185	b=1.035553 + 0.01065717
	Exp(λ)	✗	✗	-8111.376	0.2821988	16224.75	$\lambda=1.597802 + 0.01293227$	
	Normal(X, σ)	✗	✗	133.2431	0.06435865	-262.4862	X=0.6258597 + 0.001941438	$\sigma=0.2398678 + 0.001372804$
visualizacoes_duplicadas (n=9301)	Beta(A,B)	✓	✓	19770.76	0.05007532	-39537.52	a=1.427348 + 0.01898472	b=29.717935 + 0.46748971
	Exp(λ)	✓	✗	19422.44	0.103999	-38842.88	$\lambda=21.93777 + 0.2274719$	
	Normal(X, σ)	✗	✗	16322.37	0.1660794	-32640.73	X=0.04558348 + 0.0004338630	$\sigma=0.04184249 + 0.0003067875$

(*) $L(x; \theta)$ = Estimador de máxima verosimilhança; $D[KS]$ = Majorante da diferença das CDFs resultante do teste Kolmogorov-Smirnov; AIC = An Information Criterion



A.5 - Estimativa das distribuições de pontuações (Histogramas e CDFs)

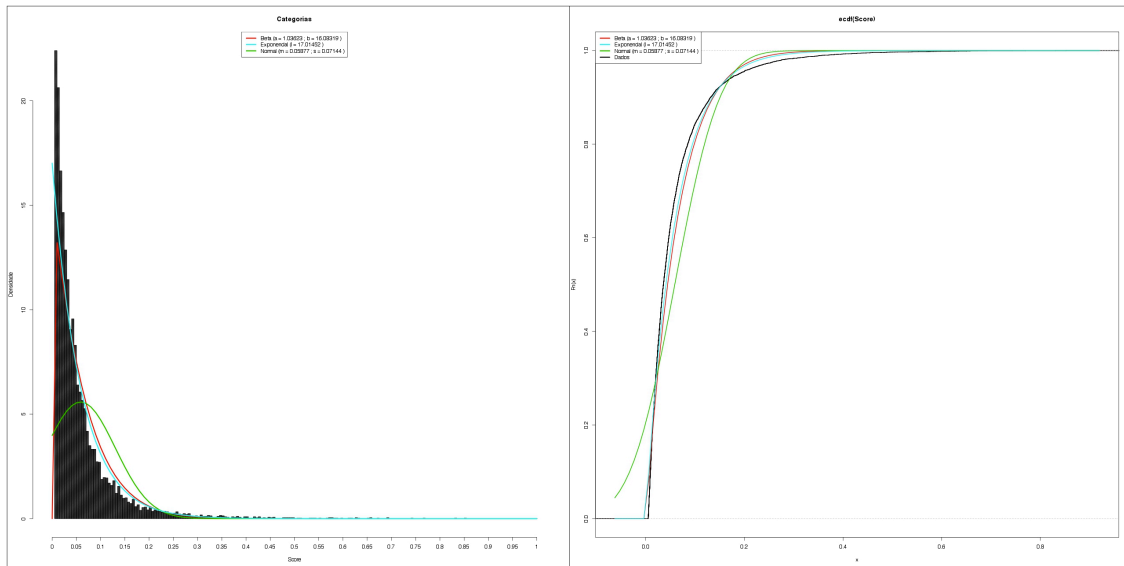


Figura A.4 - Histograma, CDF e estimativa de V_{CAT}

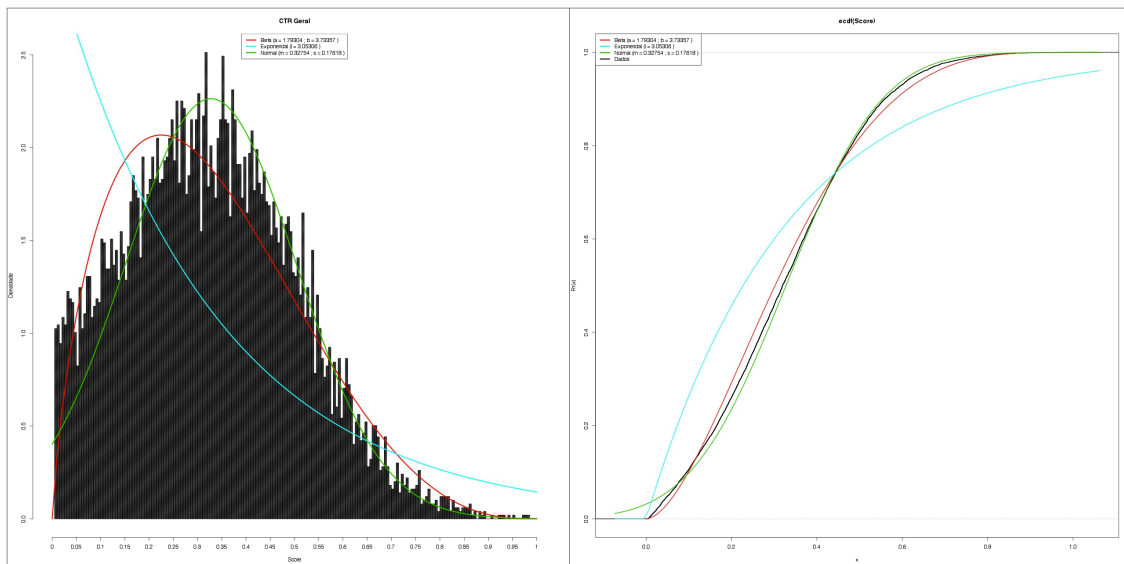


Figura A.5 - Histograma, CDF e estimativa de V_{CTR}

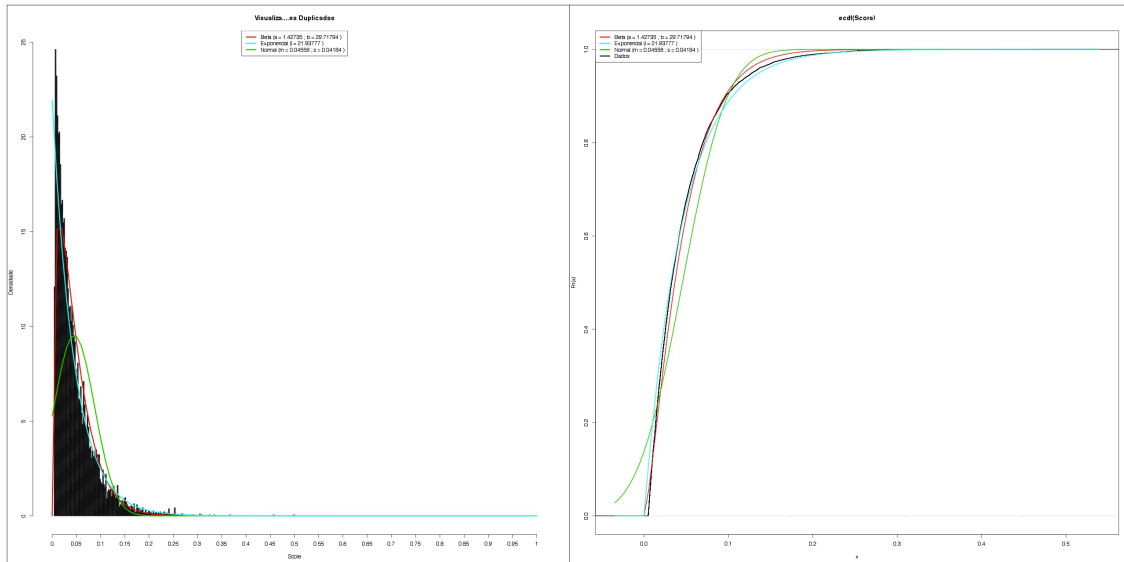


Figura A.6 - Histograma, CDF e estimativa de V_{VID}

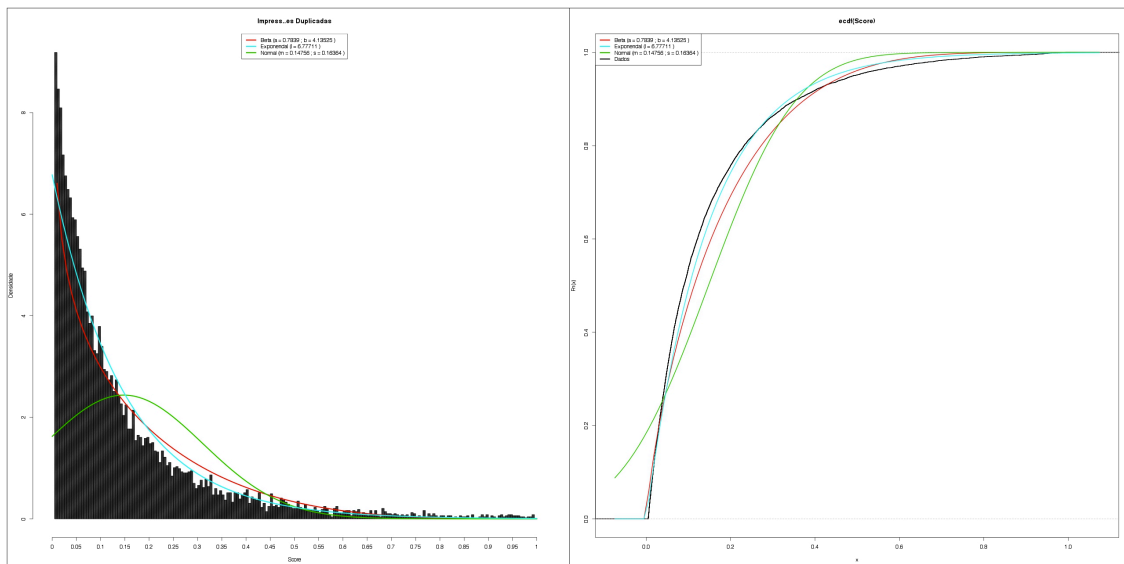


Figura A.7 - Histograma, CDF e estimativa de V_{IMD}

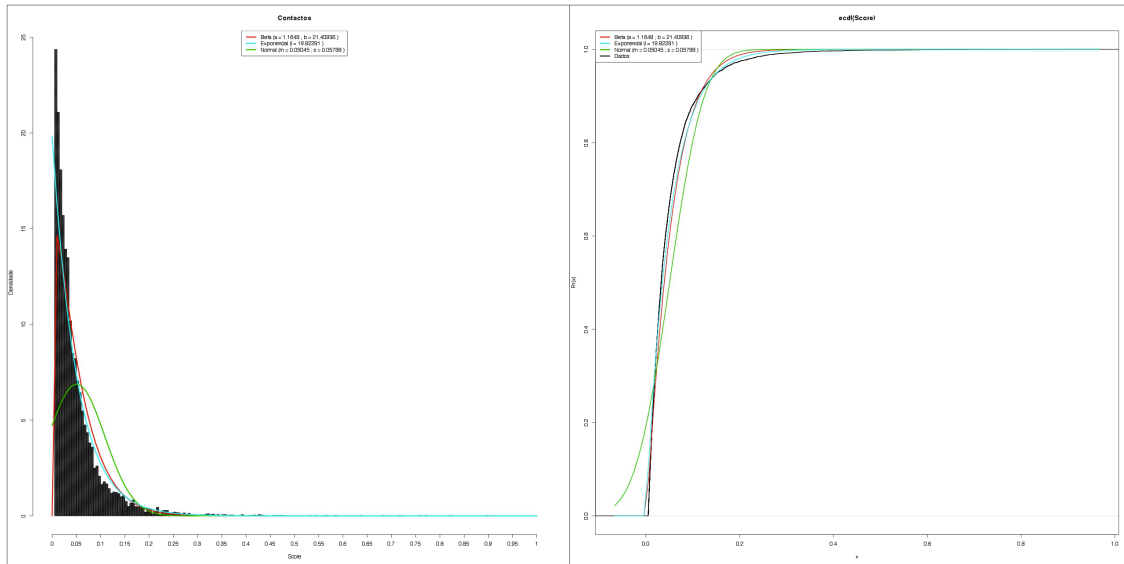


Figura A.8 - Histograma, CDF e estimativa de V_{PEC}

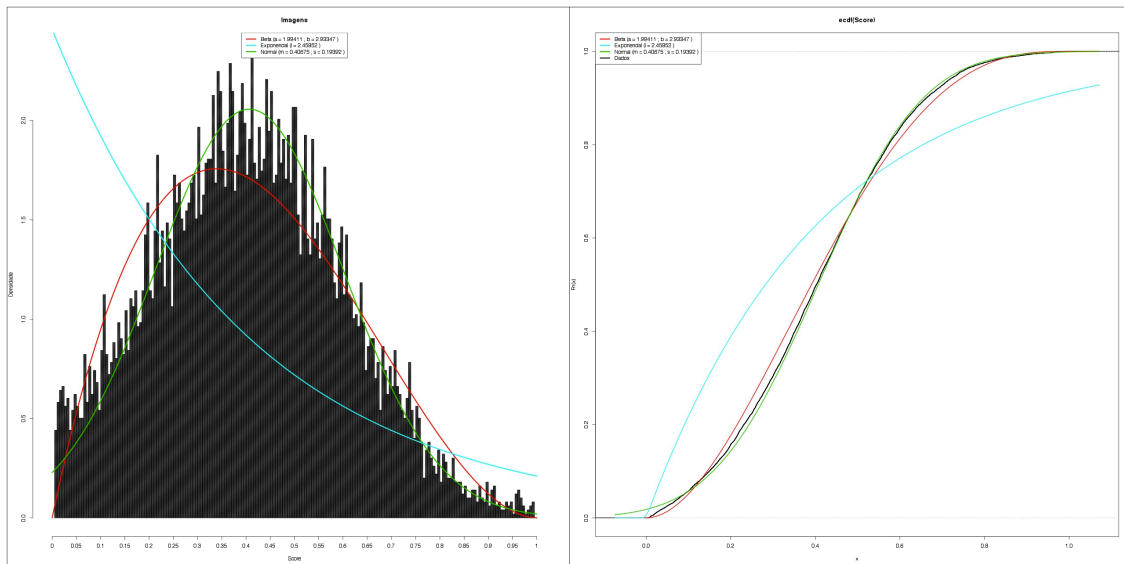


Figura A.9 - Histograma, CDF e estimativa de V_{IMV}

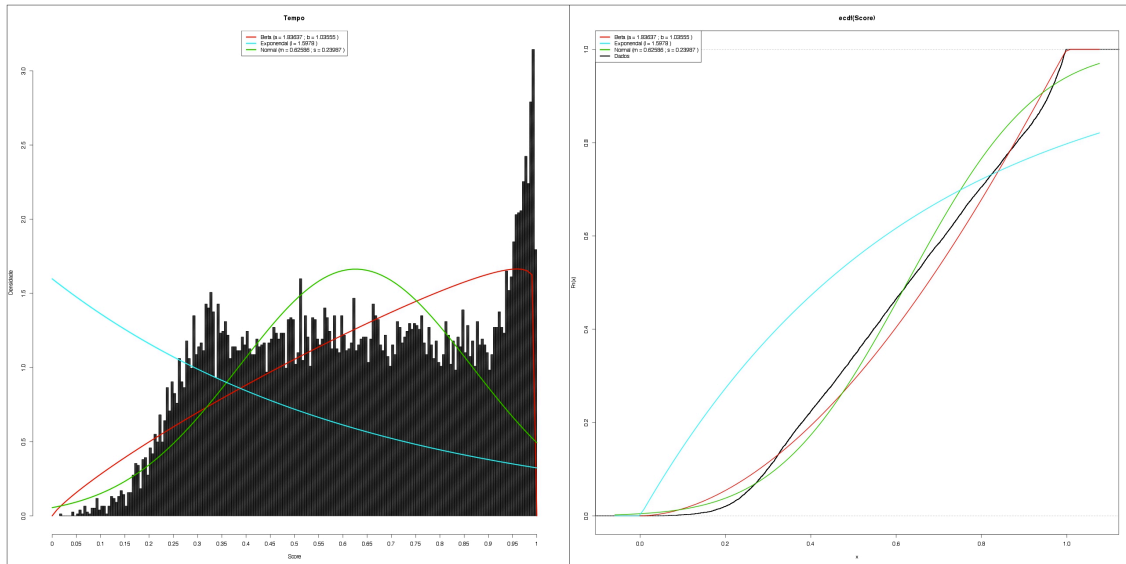


Figura A.10 - Histograma, CDF e estimativa de V_{TEU}

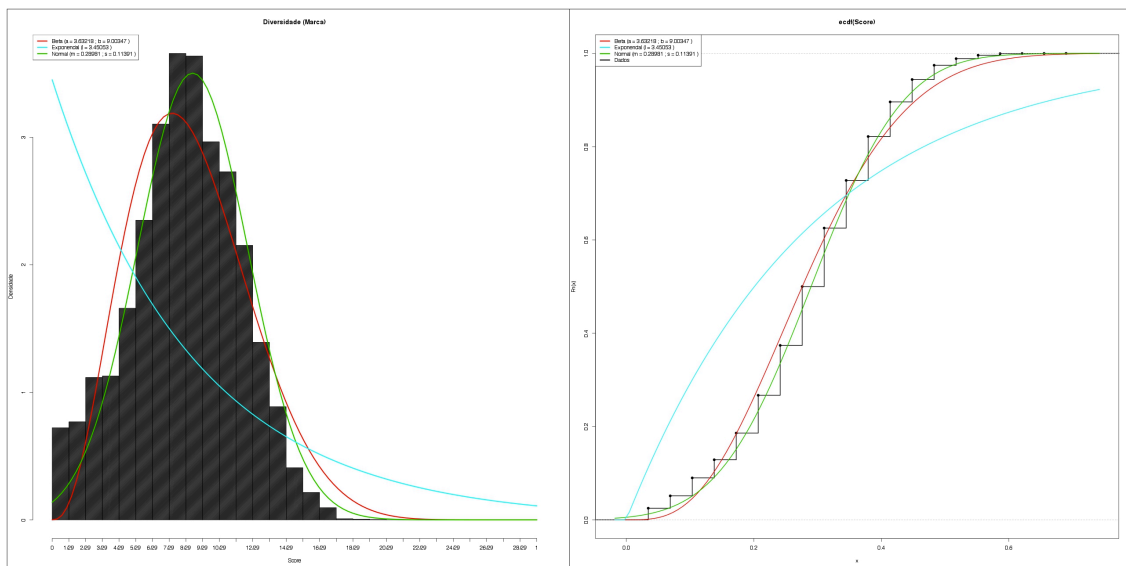


Figura A.11 - Histograma, CDF e estimativa de $V_{DIV}(aumov\acute{o}eis, marca)$

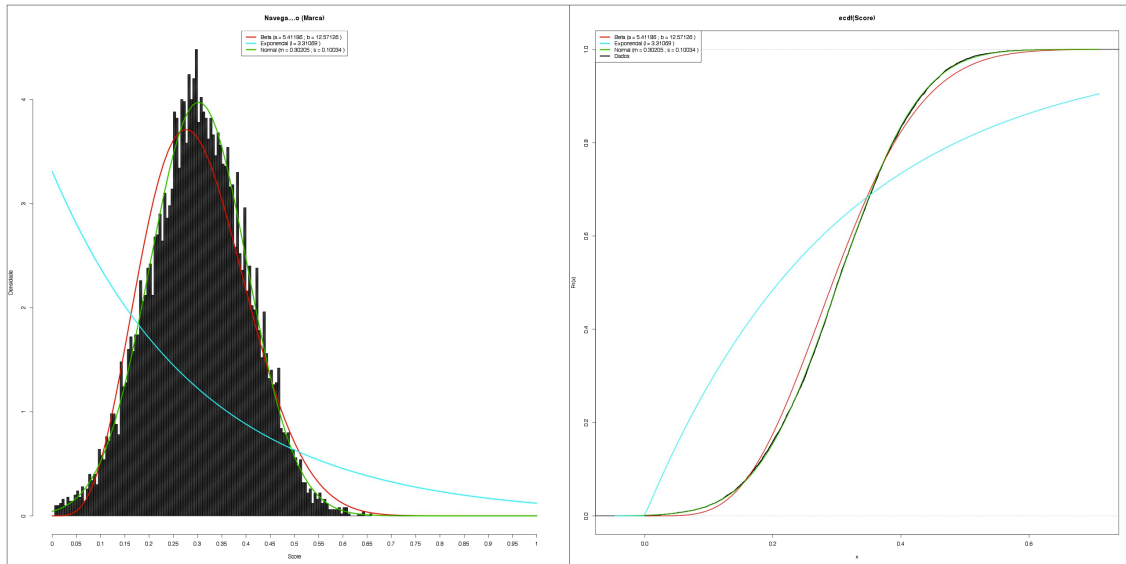


Figura A.12 - Histograma, CDF e estimativa de $V_{NAV}(aumovóveis, marca)$

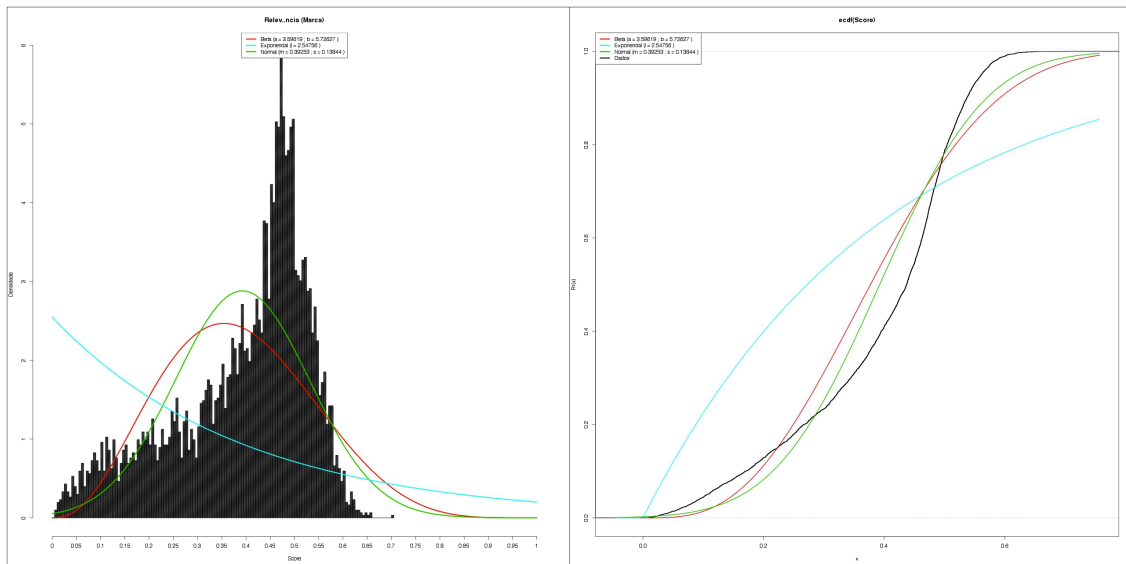


Figura A.13 - Histograma, CDF e estimativa de $V_{REL}(aumovóveis, marca)$



A.6 - Comparação visual entre *site* modelo e *site* AdClip











<ul style="list-style-type: none"> ▼ Animais <ul style="list-style-type: none"> Cães Gatos Pássaros Peixes ▼ Consolas <ul style="list-style-type: none"> Gameboy Nintendo Playstation PSP Xbox ▼ Vestuário <ul style="list-style-type: none"> Calças T-Shirts Sapatos ▼ Viaturas <ul style="list-style-type: none"> Carros Motos Veiculos Pesados 	Pesquisa <input type="text"/> Ordenação PREÇO <input type="text"/> DESCENDENTE <input type="text"/> Preço <input type="text"/> Distrito <input type="text"/> Filtros Marca <input type="text"/> Concelho <input type="text"/> Condição <input type="text"/> Género <input type="text"/> <input type="button" value="Ver resultados da pesquisa"/>
	 Local: COIMBRA > PAMPILHOSA DA SERRA > PAMPILHOSA DA SERRA Marca: Toyota RAV4 Género: Gasolina com 3 Portas Estado: Novo Retoma: Sim Garantia: Sim 52280€
	 Local: EVORA > ARRAIOLOS > S. GREGORIO Marca: Mitsubishi Lancer Género: Gasóleo com 3 Portas Estado: Usado Retoma: Sim Garantia: Sim 51454€
	 Local: VILA REAL > VILA REAL > VILA REAL (S. PEDRO) Marca: Mitsubishi Outlander Género: Gasóleo com 3 Portas Estado: Novo Retoma: Sim Garantia: Sim 51419€
	 Local: GUARDA > GOUVEIA > S. PAIO Marca: Ford E150 Cargo Género: Gasolina com 4 Portas Estado: Usado Retoma: Sim Garantia: Não 51234€
	 Local: PORTO > PAREDES > LOUREDO Marca: Volkswagen GLI Género: Gasolina com 4 Portas Estado: Usado Retoma: Sim Garantia: Não 51064€
	 Local: VIANA DO CASTELO > CAMINHA > GONDAR Marca: Cadillac Express 3500 Cargo Género: Gasóleo com 4 Portas Estado: Novo Retoma: Sim Garantia: Sim 50877€
	 Local: AVEIRO > SANTA MARIA DA FEIRA > CANEDO Marca: Audi Sable Género: Gasóleo com 4 Portas Estado: Novo Retoma: Não Garantia: Sim 50446€
	 Local: BRAGA > TERRAS DE BOURO > BALANÇA Marca: Dodge Ram 2500 Quad Cab Género: Gasóleo com 4 Portas Estado: Novo Retoma: Sim Garantia: Sim 50436€
	 Local: PORTO > VILA DO CONDE > MACIEIRA DA MAIA Marca: Ford E150 Super Duty Passenger Género: Gasóleo com 4 Portas Estado: Novo Retoma: Não Garantia: Sim 50254€
	 Local: BRAGANCA > BRAGANÇA > DEILÃO Marca: Chrysler PT Cruiser Género: Gasóleo com 4 Portas Estado: Usado Retoma: Não Garantia: Sim 49569€
	Página: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Figura A.14 - Site modelo concebido para testar e validar a solução desenvolvida




23.555 resultados da pesquisa


Procuo **Categoria** Carros (23.555)


Marca Seleccione **Modelo** Nenhum **Combustível** Seleccione


[Mais opções de pesquisa](#) [Nova pesquisa](#)


1-10 de 23.555 Ordenar por: [Sugestões](#) Ver em: [EUR \(€\)](#)


 **Volkswagen Golf 2.0 TDI Confortline 5p**
Diesel , 80000 , 2009
Lisboa, Lisboa
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)


 **Honda Jazz 1.4 i V-TEC Exclusive Navigation**
Gasolina , 9000 , 2011
Lisboa, Lisboa
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)


 **Land Rover Freelander 2.2 Td4 SE** **21.500,00 €**
Diesel , 103000 , 2007
Porto, Valongo
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)


 **Audi A6 Avant 2.0 Tdi Sport** **18.500,00 €**
Diesel , 170000 , 2006
Porto, Valongo
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)


 **Lexus GS Hybrid** **25.500,00 €**
Gasolina , 57000 , 2007
Porto, Matosinhos
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)

 **Citroen C5 Exclusive HDI FAP 4p** **18.850,00 €**
Diesel , 138 CV, 58102 km, 2008
Lisboa, Oeiras
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)

 **BMW Série 1 120d Limited Edition** **22.900,00 €**
Diesel , 177 CV, 95900 km, 2007
Braga, Braga
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)

 **Ford Galaxy 1.9 TDi 7 Lugares** **10.900,00 €**
Diesel , 187000 , 2004
Aveiro, Águeda
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)

 **Mini Cooper 1.6 Cabrio**
Gasolina , 21000 , 2010
Lisboa, Lisboa
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)

 **Volkswagen Polo Cross 1.2 i**
Gasolina , 24000 , 2011
Lisboa, Lisboa
[Viaturas > Carros](#) [Favorito](#)
[Ver detalhes](#)

[< Anterior](#) [1](#) [2](#) [3](#) [4](#) [5](#) ... [2.356](#) [Próxima >](#)

Figura A.15 - Zona de classificados produzidos pelo *script* da AdClip



P Classificados

Meus classificados | Classificados favoritos (0)

Português (Portugal)

Anúncios Classificados

Portugal

Inserir classificado grátis

8.519 resultados da pesquisa

Procuo Categoria

Distrito Concelho Freguesia

Mais opções de pesquisa Nova pesquisa Pesquisar

1-10 de 8.519

Ordenar por: Sugestões Ver em: EUR (€)

- Arrendamento Apartamento T1 Lisboa** 700,00 €
Novo, 55.00 m²
Lisboa, Lisboa, São João de Deus
Imóveis > Apartamentos
- Arrendamento Apartamento T3 Lisboa** 1.200,00 €
Usado
Lisboa, Lisboa, Lapa
Imóveis > Apartamentos
- Arrendamento Apartamento T3 Lisboa** 750,00 €
Usado
Lisboa, Lisboa, Lumiar
Imóveis > Apartamentos
- Arrendamento Apartamento T2 Lisboa** 2.500,00 €
Novo, 206.54 m²
Lisboa, Lisboa, Santa Maria dos Olivais
Imóveis > Apartamentos
- Arrendamento Apartamento T4 Lisboa** 1.800,00 €
Usado, 157.00 m²
Lisboa, Lisboa, São Sebastião da Pedreira
Imóveis > Apartamentos
- Arrendamento Apartamento T5 Lisboa** 2.500,00 €
Novo
Lisboa, Lisboa, Santa Isabel
Imóveis > Apartamentos
- Venda Apartamento T4 Lisboa** 825.000,00 €
Novo
Lisboa, Lisboa, São Jorge de Arroios
Imóveis > Apartamentos
- Arrendamento Apartamento T1 Lisboa** 900,00 €
Usado, 70.00 m²
Lisboa, Lisboa, Santa Catarina
Imóveis > Apartamentos
- Arrendamento Apartamento T5 Lisboa** 1.000,00 €
Usado
Lisboa, Lisboa, Alvalade
Imóveis > Apartamentos
- Venda Apartamento T3 Lisboa** 420.000,00 €
Novo
Lisboa, Lisboa, São João de Deus
Imóveis > Apartamentos

< Anterior 1 2 3 4 5 ... 852 Próxima >

Não encontrou? Coloque um novo anúncio "Procura-se"!

Procure por anúncios classificados em:

- Brasil | Irlanda | Bélgica | Países Baixos | Polónia | Dinamarca | Angola
- Cabo Verde | Espanha | França

Powered by AdClip - Anúncios Classificados | Termos Legais | Ajuda | Fraude
Saiba mais sobre as soluções AdClip: Anunciantes | Editores | Parceiros

Foto do dia



"Gangnam Style"

Philippe Lopez/AFP

+ Lidas + Comentadas + Partilhadas Últimas

1. Relvas ajudou empresa ligada a Passos a ter monopólio de formação em aeródromos do Centro
2. Uma viagem ao delirante mundo dos carteiristas
3. FMI reconhece que calculou mal o impacto da austeridade na economia
4. Comunicado: Sonaecon anuncia reestruturação no PÚBLICO
5. Comentário no Facebook vale processo a funcionária do Hospital de Braga
6. Função pública perde "subsídio" nos três primeiros dias de faltas por doença
7. O elemento 113 existe (quase de certeza). Por agora, chama-se ununtrio
8. FMI pede travão à austeridade na Europa
9. Comunicado da Direcção Editorial do PÚBLICO
10. CGD vende casas para arrendamento com inquilino já incluído

Jornal do dia



Campanhas Web Marketing

Complete a divulgação da sua marca com recurso à comunicação online
www.wco.pt

Proseguir Alarmes Casa

Proteja a Família com menos 1€ dia Faça já a sua Simulação Grátis!
Proseguir-alarmes-casa.com

Oferta Maldivas 2012

Grande oferta de uma viagem às Maldivas com tudo incluído!
www.viagem-maldivas.com

Nº1 em Carros Usados

Standvirtual é Nº1 em Carros Usados Comprar/vender só no Standvirtual®!
www.standvirtual.com/carros-usados

Anúncios PÚBLICO

Loja Público loja.publico.pt

Figura A.16 - Integração dos classificados na página do Jornal Público