Redes neurais em análise de sobrevivência: Uma aplicação na área de relacionamento com clientes

Marcelo Hiroshi Ogava

DISSERTAÇÃO APRESENTADA
AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA
UNIVERSIDADE DE SÃO PAULO
PARA
OBTENÇÃO DO TÍTULO DE MESTRE
EM
CIÊNCIAS

Área de concentração: Estatística
Orientador: Prof. Dr. Antonio Carlos Pedroso de Lima

São Paulo, junho de 2007.

Redes neurais em análise de sobrevivência: Uma aplicação na área de relacionamento com clientes

Este exemplar corresponde à redação final da dissertação devidamente corrigida e defendida por Marcelo Hiroshi Ogava e aprovada pela Comissão Julgadora.

São Paulo, 4 de junho de 2007.

Banca Examinadora:

Prof. Dr. Antonio Carlos Pedroso de Lima (Orientador) – IME-USP

Profa. Dra. Lúcia Pereira Barroso – IME-USP

Prof. Dr. Manoel Raimundo de Sena Junior – UFPE

Perguntaram ao Dalai Lama...

“O que mais te surpreende na Humanidade?”

E ele respondeu:

“Os homens... Porque perdem a saúde para juntar dinheiro, depois perdem dinheiro para recuperá-la.
E por pensarem ansiosamente no futuro, esquecem do presente de tal forma que acabam por não viver nem o presente nem o futuro.
E vivem como se nunca fossem morrer... e morrem como se nunca tivessem vivido.”

Aos meus queridos pais.

Agradecimentos

Agradeço a todos que, de alguma maneira, contribuíram para a elaboração deste trabalho, em especial:

Ao professor **Antonio Carlos**, pela orientação, compreensão e cobranças nos momentos certos.

Aos meus queridos pais **Katsuke** e **Akiko**, pelo carinho, apoio e confiança que sempre depositaram em mim.

À minha irmã **Harumi**, pelo incentivo e por me dar duas sobrinhas maravilhosas: **Gabriela** e a **Daniela**.

À minha namorada **Beatriz**, pela compreensão, carinho e apoio em todos os momentos.

Ao professor **Clóvis de Araújo Peres**, pela amizade, incentivo e ensinamentos que tornaram a Estatística muito mais interessante.

Aos meus amigos da **ETFSP**, do **IME** e da empresa, pela amizade, companheirismo e apoio. Aqui, gostaria fazer um agradecimento especial à minha querida amiga **Tatiana Terabayashi Melhado** que muito me ajudou, sobretudo na fase final deste trabalho.

E, finalmente, aos professores do Departamento de Estatística do IME-USP que muito contribuíram para a minha formação.

Resumo

A medida que as economias modernas tornam-se predominantemente baseadas na prestação de serviços, as companhias aumentam seu valor na criação e na sustentabilidade do relacionamento a longo prazo com seus clientes. O “Customer Lifetime Value (LTV)”, que é uma medida de potencial de geração de lucro, ou valor de um cliente, vem sendo considerado um ponto fundamental para o gerenciamento da relação com os clientes. O principal desafio em prever o LTV é a produção de estimativas para o tempo de duração do contrato de um cliente com um dado provedor de serviços, baseado nas informações contidas no banco de dados da companhia. Neste trabalho, apresentaremos uma alternativa aos modelos estatísticos clássicos, utilizando um modelo de redes neurais para a previsão da taxa de cancelamento a partir do banco de dados de uma empresa de TV por assinatura.

Abstract

As modern economies become predominantly service-based, companies increasingly derive revenue from creation and sustenance of long-term relationships with their customers. The Customer Lifetime Value (LTV), which is a measure of the profit generating potential, or value of a customer, is increasingly being considered as a touchstone for customer relationship management. The central challenge of LTV is the production of estimated customer tenures with a given service provider, based on information contained in the company database. In this study, we consider an alternative to classical statistical models, using a neural network model for hazard prediction based on the database information of a pay TV company.

Índice

Capítulo 1

Introdução

As empresas passaram, na última década, por duas grandes fases na área de relacionamento com o cliente, e algumas delas encontram-se, agora, na terceira fase. A primeira fase foi aquela em que o foco era direcionado apenas na aquisição de novos clientes, não se preocupando com os clientes que estavam saindo da empresa. Com o mercado cada vez mais competitivo como, por exemplo, o mercado de TV por assinatura, é sabido hoje que a aquisição de um novo cliente é muito mais cara do que a manutenção de um cliente que já está na base e, foi neste momento, que a segunda fase se iniciou. A segunda fase é caracterizada pela tentativa de reter o cliente a qualquer custo chegando em alguns casos, a manter clientes que não geram nenhuma receita à empresa. Isso pode ser explicado porque, ainda nos dias de hoje, muitas empresas são avaliadas pela quantidade de clientes na base e não pela "qualidade" dos mesmos.

Algumas empresas, enxergando a possibilidade de aumentar a rentabilidade da base de clientes equilibrando os gastos de aquisição e retenção, entraram na terceira fase do relacionamento com o cliente. Nesta fase a empresa tenta descobrir qual é o ponto até onde vale a pena gastar para manter um cliente na base ou deve-se deixá-lo ir e sair em busca de um novo cliente no mercado. Para este fim, técnicas de Gerenciamento do Relacionamento com o Cliente (conhecido como Customer Relationship Management -

CRM) são desenhadas, desenvolvidas e implementadas. Essas técnicas objetivam auxiliar os administradores a entender as necessidades dos clientes, descobrindo padrões, e ajudando a desenvolver ofertas direcionadas que são, não apenas melhor desenhadas para cada tipo de cliente, mas também mais lucrativas para o negócio a longo prazo.

Entre essas técnicas, o LTV ("Customer Lifetime Value") ou, em uma tradução livre, "Valor Durante o Tempo de Relacionamento com o Cliente", que é uma medida do potencial de geração de receita de um cliente, vem ganhando muita atenção nos últimos anos e grandes companhias como IBM e ING vêm utilizando essa ferramenta rotineiramente para gerenciar e medir o sucesso de seus negócios (Gupta *et al*, 2006). Com ela é possível:

- Criar serviços e ofertas especiais em que, quanto maior o valor do cliente, mais irresistíveis serão esses serviços e essas ofertas, sujeitos ainda a uma satisfatória margem de lucro para os negócios;
- Fazer ações pró-ativas evitando que os clientes troquem de empresa, principalmente em um mercado como o de TV por assinatura que apresentou um baixo crescimento nos últimos anos;
- Atingir e gerenciar clientes que não geram receita;
- Segmentar os clientes para ações de marketing, preços e promoções;
- Estimar e planejar futuras oportunidades baseado no valor acumulado do cliente.

O LTV é, normalmente, composto de dois componentes: o tempo de relacionamento (Tenure) e o potencial de geração de receita de um cliente. Apesar da modelagem do valor (ou lucro), um dos componentes do LTV – que leva em conta o consumo, os gastos fixos e variáveis – já ser um grande desafio, o desafio central na previsão do LTV está na produção de estimativas individuais para o tempo de relacionamento de um cliente com um determinado fornecedor de serviços, baseado no histórico do comportamento e consumo contidos no banco de dados da empresa. Portanto, vamos nos focar exclusivamente na produção dessas estimativas.

No nosso caso vamos estimar o tempo de relacionamento com o cliente a partir do momento em que este entra em contato com a empresa e solicita o cancelamento, ou seja, a partir do momento em que o cliente decidiu cancelar o serviço e foi retido recebendo, ou não, uma oferta de desconto na mensalidade.

Existem diferentes técnicas estatísticas, na área de Análise de Sobrevivência, que podem ser aplicadas para a modelagem do tempo de relacionamento (Tenure) e a nossa intenção é comparar um modelo clássico de Análise de Sobrevivência com um modelo híbrido, que utiliza redes neurais.

Capítulo 2

Contexto da aplicação

Neste capítulo vamos apresentar alguns conceitos e processos utilizados em empresas de TV por assinatura. Em seguida faremos uma breve descrição do banco de dados.

2.1 O mercado de TV por assinatura

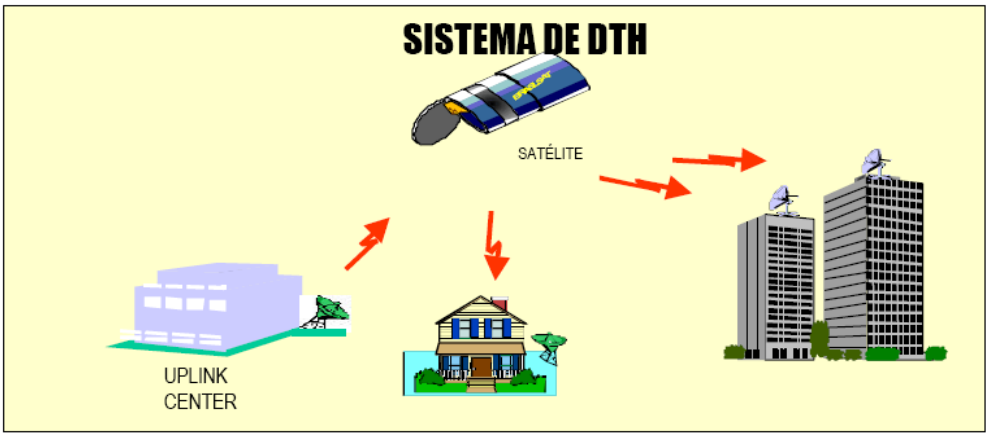
A TV por assinatura surgiu nos Estados Unidos na década de 40 como forma de pequenas comunidades receberem os sinais de TV aberta que não chegavam a suas casas com boa qualidade. As pessoas associavam-se e adquiriam uma antena de alta sensibilidade. Depois, com o uso de cabos coaxiais, os sinais eram distribuídos até as residências.

No Brasil, o processo foi semelhante. Começou há mais de 40 anos em função da necessidade de fazer com que o sinal das emissoras de televisão localizadas na cidade do Rio de Janeiro chegasse às cidades vizinhas com boa qualidade de som e de imagem. Os usuários que desejavam utilizar o serviço pagavam uma taxa mensal da mesma maneira que ocorre hoje no serviço de TV por assinatura.

Desde que se instalaram no Brasil, as empresas do setor enfrentaram três grandes crises econômicas e até meados da década passada ainda tinham uma penetração incipiente. Em 1994 havia apenas 400 mil assinantes e foi atingida em 2006 a marca de 4,5 milhões de assinantes. Apesar desse aumento, a penetração da TV paga no Brasil ainda é umas das menores do mundo atingindo em torno de 8% dos domicílios. Isso pode ser explicado pelo pequeno número de domicílios classe A/B e pela grande cobertura da TV aberta. Com o mercado estagnado, as empresas têm notado que não estão entrando novos clientes no mercado, o que acontece é a troca de uma empresa pela outra, tornando grande a concorrência no setor e fazendo do CRM uma peça chave para evitar que os bons clientes deixem a empresa.

Existe uma grande confusão com relação às empresas de TV por assinatura, pois muitas pessoas acreditam que elas são as responsáveis pela produção do conteúdo dos canais, o que não é verdade. A empresa de TV por assinatura apenas distribui o conteúdo que uma outra empresa, chamada programadora, criou. Esse conteúdo chega via satélite à empresa que irá distribuir o conteúdo através de uma das três tecnologias de TV por assinatura existentes no Brasil:

- DTH – Direct to Home: Nesta tecnologia o cliente recebe os sinais de TV diretamente do satélite, tem cobertura nacional por não depender de uma estrutura de cabos e tem uma participação de 34% do mercado.

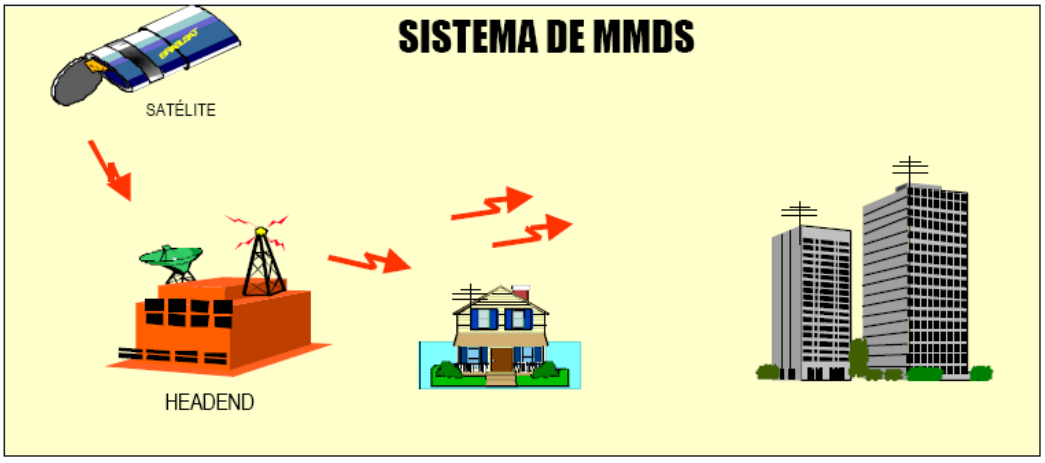


DTH- A PROGRAMADORA TRANSMITE O SINAL POR SATÉLITE DIRETAMENTE ATÉ A CASA DO ASSINANTE.

- Cabo: No Cabo, o assinante recebe o sinal via cabo coaxial ou de fibra ótica e, por depender de uma grande infra-estrutura, só está disponível nos grandes centros, o que não chega a ser um problema, pois são nesses locais que estão os consumidores com maior poder aquisitivo. Como é uma tecnologia mais antiga e mais barata, atinge uma participação de 60% no mercado.

CABO- A PROGRAMADORA TRANSMITE O SINAL POR SATÉLITE ATÉ O "HEADEND" DA OPERADORA, QUE ENVIA A PROGRAMAÇÃO AO ASSINANTE POR MEIO DE CABO COAXIAL OU DE FIBRA ÓPTICA.

- MMDS – Multichannel Multipoint Distribution Service: No MMDS a empresa de TV por assinatura envia os sinais ao assinante através de uma antena de microondas; é a tecnologia menos utilizada e responde por 6% do mercado.



MMDS- A PROGRAMADORA TRANSMITE O SINAL POR SATÉLITE ATÉ O "HEADEND" DA OPERADORA, QUE ENVIA A PROGRAMAÇÃO AO ASSINANTE. ESTE A RECEBE ATRAVÉS DE UMA ANTENA DE MICROONDAS.

2.2 Churn

O cancelamento da assinatura, conhecido como churn ou attrition, pode ser classificado em dois tipos:

- Voluntário: neste tipo de cancelamento o cliente solicita o término do contrato, seja por falta de uso, problemas financeiros ou insatisfação com os serviços. Este será o nosso evento de interesse.
- Involuntário: este cancelamento ocorre quando parte da empresa a decisão em cancelar o contrato do cliente por este estar inadimplente.

2.3 Retenção do cliente

A retenção do cliente ocorre quando este entra em contato com a empresa decidido a cancelar o serviço (no caso do Churn Voluntário) e muda de idéia depois de conversar com o (a) operador (a) e, talvez, receber alguma oferta. O processo se inicia quando ele é atendido pelo Serviço de Atendimento ao Cliente (SAC) e, assim que demonstra sua intenção em cancelar, é transferido para uma célula especial, que possui maior habilidade em negociar e têm disponível um conjunto de ofertas que vão desde brindes até descontos na mensalidade. O processo é finalizado, com a continuidade do contrato, caso o cliente seja retido, ou o cancelamento, caso contrário.

A nossa intenção é estimar a duração do tempo de relacionamento com o cliente a partir do momento que este demonstrar, através de um contato telefônico, a intenção em cancelar o contrato que é revertida. O intuito é conceder a oferta de retenção mais atrativa sem que isso comprometa a rentabilidade da empresa nos próximos meses, pois quanto maior a duração estimada do tempo de relacionamento, maiores serão as opções de ofertas que teremos à disposição.

2.4 Banco de dados

O banco de dados é constituído de informações mensais com histórico de 6 meses ou 12 meses, onde maior parte das informações está relacionada ao comportamento do cliente com relação à empresa, sendo carregadas mensalmente no banco de dados. Essas informações geralmente se mostram mais importantes do que aquelas relacionadas especificamente ao cliente como, por exemplo, o sexo ou a idade, pois nem sempre o titular da assinatura é o maior usuário do serviço.

Como variáveis resposta temos:

- Indicador de censura: assume valor 0 se o cliente cancelou efetivamente o contrato depois de ter sido revertido pela primeira vez, ou valor 1, caso contrário;
- Tempo de acompanhamento: é a duração do contrato (em meses) a partir do momento em que o cliente solicitou o cancelamento pela primeira vez e foi revertido até o término do acompanhamento ou cancelamento efetivo (voluntário) do contrato, o que ocorrer primeiro.

Podemos classificar as informações contidas no banco de dados em alguns grupos (uma lista completa das covariáveis existentes no banco pode ser encontrada no Apêndice A):

- Tempo: Quantidade de meses desde a assinatura do serviço (Tenure);
- Financeiro: Ajustes financeiros, valores devidos pelo assinante, problemas com valores cobrados, créditos concedidos, método de pagamento, etc;
- Contatos: Quantidade de contatos separados por motivo (SAC, assistência técnica, cobrança, solicitação de cancelamento, etc);
- Programação: upgrade, downgrade, canais adultos, canais premium, pay-per-view, etc;
- Outros: e-mail cadastrado, quantidade de receptores, etc.

O período de acompanhamento escolhido foi de 12 meses a partir da data em que o cancelamento foi solicitado e revertido, pois estamos assumindo que, se o cliente não estiver propenso a cancelar durante todo esse período, ele pode receber as ofertas de maior valor sem comprometer a rentabilidade da empresa.

Foram considerados clientes que solicitaram o cancelamento e foram revertidos ao longo do mês de Setembro/05 totalizando 31464 clientes sendo contabilizados 3697 cancelamentos voluntários do contrato durante o período de acompanhamento (o que equivale a 88% de casos censurados).

Nos próximos capítulos apresentaremos as técnicas que serão utilizadas para modelar a estimativa do tempo de relacionamento com o cliente. Primeiramente serão apresentados os modelos usuais de Análise de Sobrevivência como, por exemplo, o modelo de riscos proporcionais de Cox e os modelos para dados agrupados. Em seguidas apresentaremos a técnica de redes neurais e alguns métodos propostos para adaptar essa técnica aos dados censurados.

Capítulo 3

Análise de sobrevivência

Neste capítulo vamos rever algumas técnicas clássicas para a análise de dados de sobrevivência para os tempos de vida contínuos e discretos. Maiores detalhes podem ser encontrados em Kalbfleish e Prentice (1980) e Klein e Moeschberger (2003).

3.1 Definições

Seja T o tempo de ocorrência de um determinado evento, no nosso caso o cancelamento voluntário do contrato, uma variável aleatória não-negativa. A distribuição de probabilidade de T pode ser especificada em diferentes maneiras, três delas são particularmente úteis na análise de sobrevivência: a função de sobrevivência, a função de densidade de probabilidade e a função da taxa de cancelamento. As relações entre essas representações são apresentadas abaixo para distribuições contínuas e discretas de T.

A função de sobrevivência, ou de continuidade do contrato com a empresa, pode ser interpretada como a probabilidade de um cliente não cancelar o contrato até o instante t, é defina por:

$$S(t) = \mathrm{P}(T > t), \qquad 0 < \mathrm{t} < \infty. \tag{3.1}$$

Para *T* (absolutamente) contínuo:

Sendo *T*, o tempo até o cancelamento voluntário do contrato, uma variável aleatória contínua, *S(t)* é uma função contínua e não crescente. A função de sobrevivência é o complemento da função de distribuição acumulada, *S(t) = 1 – F(t)*, em que *F(t) = P(T ≤ t)*. Além disso, a função de sobrevivência é a integral da função densidade de probabilidade, *f(t)*, isto é,

$$S(t) = P(T > t) = \int_t^{\infty} f(u)du \tag{3.2}$$

e,

$$f(t) = -\frac{dS(t)}{dt}. \tag{3.3}$$

A função da taxa de cancelamento voluntário do contrato, que pode ser interpretada como a força de um cliente, ativo num dado instante, cancelar voluntariamente o contrato no próximo instante de tempo infinitesimal, é definida por:

$$\alpha(t) = \lim_{h \to 0} \frac{P(t \le T < t + h | T \ge t)}{h}. \tag{3.4}$$

Sendo *T* uma variável aleatória contínua, então,

$$\alpha(t) = \frac{f(t)}{S(t)} = -\frac{d \ln(S(t))}{dt}. \tag{3.5}$$

A função da taxa de cancelamento voluntário do contrato acumulada *A(t)*, é definida por:

$$A(t) = \int_0^t \alpha(u)du = -\ln(S(t)) . \tag{3.6}$$

Então, para tempos de permanência contínuos,

$$S(t) = \exp(-A(t)) = \exp\left(-\int_0^t \alpha(u)du \right). \tag{3.7}$$

Para ***T*** discreto:

Se *T*, o tempo até o cancelamento voluntário do contrato, é uma variável aleatória discreta assumindo valores $a_1 < a_2 < ...$ com função de probabilidade:

$$f(a_i) = P(T = a_i) , \; i = 1, 2, ... \tag{3.8}$$

Então a função de sobrevivência é,

$$S(t) = \sum_{i | a_i \geq t} f(a_i) . \tag{3.9}$$

A taxa de cancelamento voluntário do contrato em a_i é definida como uma probabilidade condicional de falha em a_i,

$$\alpha(a_i) = P(T = a_i | T \geq a_i) = \frac{f(a_i)}{S(a_i^-)} , \; i = 1, 2, ... \tag{3.10}$$

em que,

$$S(a_i^-) = P(T \geq a_i) . \tag{3.11}$$

Segue que,

$$S(t) = \prod_{i|a_i \le t} (1 - \alpha(a_i)) \tag{3.12}$$

e

$$f(a_i) = \alpha(a_i) \prod_{k=1}^{i-1} (1 - \alpha(a_k)) . \tag{3.13}$$

3.2 Dado incompleto

O principal diferencial da análise de sobrevivência com relação a outras técnicas estatísticas é a censura. A censura ocorre quando, por exemplo, um cliente não é observado durante todo o tempo de duração do seu contrato e sabemos apenas que o tempo de duração é maior que tempo de acompanhamento. Existem vários padrões de censura, que vão desde o conhecimento preciso do tempo de duração do contrato até a informação de que o cancelamento ocorreu em um intervalo (que pode se estender ao infinito, como no exemplo anterior).

No nosso caso, existem basicamente três tipos de censura: se o cliente continua com o contrato depois dos 12 meses de acompanhamento, se ele solicitou o cancelamento novamente e foi revertido ou se o cliente foi cancelado involuntariamente. A partir de agora vamos assumir que o termo "cancelamento" refere-se apenas ao nosso evento de interesse que é o cancelamento voluntário do contrato.

Denote por T_j o tempo até o cancelamento do j-ésimo cliente. Em muitos casos não se tem a informação exata do valor de T_j, mas sabe-se que é maior que um determinado um valor C_j, tempo de censura, o que caracteriza a censura à direita. Normalmente, o tempo que conhecemos é o $min\ (T_j,\ C_j)$ e, além disso, uma variável indicadora de cancelamento $\delta_j = I(T_j \le C_j)$, ou seja,

$$\begin{cases} 1, & se \quad T_j \le C_j \\ 0, & caso \quad contrário. \end{cases}$$

3.3 Estimador de Kaplan-Meier

Um método não-paramétrico, muito conhecido na Análise de Sobrevivência, para estimar as funções de taxa de cancelamento e sobrevivência é o método desenvolvido por Kaplan e Meier (1958). Suponha que os cancelamentos ocorram nos tempos $t_1 < t_2 < \cdots < t_k$, d_t representa o número de cancelamentos no instante $t = t_i$, *i = 1, 2, ..., k*. Seja n_t o número de clientes em risco de cancelamento, este número é calculado subtraindo do número de clientes em risco de cancelamento no início do período $t-1$, o número total de cancelamentos e o número total de censuras nesse mesmo período. Assim, para o instante t_i, a taxa de cancelamento é estimada por:

$$\hat{\alpha}(t_i) = \frac{d_{t_i}}{n_{t_i}}, \; i = 1, \ldots, k \tag{3.14}$$

e a função de sobrevivência é estimada por:

$$\hat{S}(t_i) = \hat{S}(t_i - 1)(1 - \hat{\alpha}(t_i)), \; i = 1, \ldots, k \tag{3.15}$$

com $\hat{S}(0) = 1$.

Propriedades do estimador Kaplan-Meier podem ser vistas em Kalbfleish e Prentice (1980).

3.4 Modelos de regressão para dados de sobrevivência

Até agora nos concentramos apenas em uma amostra univariada de uma única distribuição do tempo de duração do contrato. Na prática, muitas situações envolvem populações heterogêneas e é importante considerar a relação entre o tempo de duração do contrato e os outros fatores. Uma maneira de se fazer isso é através dos modelos de regressão, que podem ser paramétricos ou não-paramétricos (ou semiparamétricos), que envolvem a especificação de um modelo para a distribuição do tempo T, dado Z, onde T representa o tempo de duração do contrato e Z o vetor de covariáveis de um mesmo cliente. Entre os modelos de regressão temos duas grandes classes: os modelos de locação-escala e os modelos de riscos proporcionais, sendo o segundo, a classe de interesse para nosso estudo.

3.4.1 Modelo de regressão semiparamétricos com riscos proporcionais.

A família de modelos de riscos proporcionais é uma classe de modelos com a propriedade que diferentes clientes têm funções de taxas de cancelamento proporcionais entre si. Isto é, a taxa $\alpha(t|z_1)/\alpha(t|z_2)$ para dois clientes com vetores de variáveis z_1 e z_2 não varia ao longo de t. Isto implica que a função de risco de T, dado z, pode ser escrita da seguinte forma:

$$\alpha(t;z) = \alpha_0(t)c(z\beta), \tag{3.16}$$

em que $\alpha_0(t)$ pode ter uma forma paramétrica específica ou uma função não-negativa arbitrária e $c(z\beta)$ qualquer função não-negativa com $c(0)=1$; $\alpha_0(t)$ pode ser interpretada com uma função de taxa de cancelamento base, pois representa a função de taxa de cancelamento para um cliente com $c(z\beta)=1$. Entre esses modelos encontramos o modelo semiparamétrico de riscos proporcionais sugerido por Cox (1972).

Os modelos de riscos proporcionais de Cox introduziram uma nova dimensão de flexibilidade na análise de dados censurados com covariáveis. A função da taxa de cancelamento no tempo t e vetor de regressão z assume a forma:

$$\alpha(t;z) = \alpha_0(t)\exp(z\beta) \tag{3.17}$$

em que $\alpha_0(t)$ é uma função não-negativa arbitrária de t, $z = (z_1, z_2, ..., z_p)$ é um vetor de covariáveis e β é um vetor de parâmetros desconhecido, a ser estimado. O fator $\alpha_0(t)$ é a taxa de cancelamento associado a um cliente com covariável de regressão $z = 0$ que chamaremos de taxa de cancelamento basal.

A função de sobrevivência condicional de T dado z é:

$$\begin{aligned} S(t;z) = P(T > t) &= \exp\left\{-\int_0^t \alpha(u)du\right\} = \exp\left\{-\int_0^t \alpha_0(u)\exp(z\beta)du\right\} \\ &= \exp\left\{-\exp(z\beta)\int_0^t \alpha_0(u)du\right\} \\ &= \exp\left\{-\int_0^t \alpha_0(u)du\right\}^{\exp(z\beta)} \\ &= [S_0(t)]^{\exp(z\beta)} \end{aligned} \tag{3.18}$$

com,

$$S_0(t) = \exp\left[-\int_0^t \alpha_0(u)du\right] \tag{3.19}$$

a função de sobrevivência basal.

A função de densidade condicional de T dado z é:

$$f(t;z) = \alpha(t;z)S(t;z)$$

$$= \alpha_0(t)e^{z\beta} \exp\left[-\int_0^t \alpha_0(u)d(u) \right]^{\exp(z\beta)}$$

$$= \alpha_0(t)e^{z\beta} \left[S_0(t)\right]^{\exp(z\beta)}. \qquad (3.20)$$

Sendo $\alpha_0(.)$ arbitrário, temos que este modelo é suficientemente flexível para muitas aplicações. Entretanto podemos citar duas importantes generalizações que não tornam a estimativa do β muito complicada:

- Estratificação: neste caso permitimos que o $\alpha_0(.)$ varie entre específicos subgrupos do banco de dados. Suponha que os clientes sejam separados em r estratos e que o risco $\alpha_k(t; z)$ do k-ésimo estrato depende da função arbitrária $\alpha_{0k}(t)$ e pode ser escrita como:

$$\alpha_k(t;z) = \alpha_{0k}(t)exp(z\,\beta) \qquad (3.21)$$

para $k=1, ..., r$. Essa generalização é útil nos casos em que uma ou mais covariáveis não aparentam possuir efeito multiplicativo na função de risco.

- Covariáveis dependentes do tempo: esta segunda importante generalização permite que a covariável de regressão z varie com o tempo; neste caso o modelo deixa de ser de risco proporcional.

3.4.2 Modelos de regressão para dados de sobrevivência agrupados

Existem casos em que os tempos de duração do contrato não são mais precisos que um intervalo onde o cancelamento ocorreu; na realidade o tempo observado é sempre discreto, entretanto, se o intervalo for pequeno relativamente à taxa de ocorrência dos cancelamentos é razoável assumirmos que o tempo é contínuo. Quando as unidades de tempo são grandes, meses, anos ou décadas, o tratamento desses dados se torna

problemático. Os dados de sobrevivência deste tipo são chamados de agrupados ou de censura intervalar.

Existem, de maneira geral, duas alternativas para este problema que, na realidade, possuem resultados muito similares. A mais simples é tratar o tempo como se fosse realmente discreto. Uma segunda alternativa é iniciarmos com um modelo de tempo contínuo, normalmente o modelo de riscos proporcionais, e então calcular os estimadores desse modelo que são apropriados para dados agrupados em intervalos. Este foi o procedimento utilizado por Prentice e Gloeckler (1978) e Allison (1982) que apresentaremos a seguir.

Supondo que os tempos de cancelamento sejam agrupados em intervalos $I_i = [\xi_{i-1}, \xi_i)$, $i = 1, ..., k$ com $\xi_0 = 0$ e $\xi_k = \infty$, os tempos de cancelamento em I_i são indicados por a_i. A probabilidade de observarmos um cancelamento de um cliente j no intervalo a_i com vetor de regressão z é:

$$P(T_j = a_i; z) = \alpha_j(a_i; z)\prod_{k=1}^{i-1}(1-\alpha_j(a_k; z)). \tag{3.22}$$

E a probabilidade de um indivíduo não cancelar no intervalo a_i é:

$$P(T_j > a_i; z) = \prod_{k=1}^{i}(1-\alpha_j(a_k; z)). \tag{3.23}$$

Considere, t_j como sendo o tempo de duração do contrato observado a_i, $i = 1, 2, ...,$ do j-ésimo cliente. Suponha que $\delta_j = 1$ se em $T_j = t_j$ ocorreu um cancelamento e 0 caso contrário. Com isso, a verossimilhança dos dados de sobrevivência agrupados é dada por:

$$L = \prod_j \left[P(T_j = t_j; z)\right]^{\delta_j}\left[P(T_j > t_j; z)\right]^{1-\delta_j}$$

$$= \prod_j \left[\frac{P(T_j = t_j; z)}{P(T_j > t_j; z)} \right]^{\delta_j} P(T_j > t_j; z)$$

$$= \prod_j \left\{ \frac{\alpha_j(t_j; z)}{(1 - \alpha_j(t_j; z))} \right\}^{\delta_j} \prod_{k=1}^{t_j} (1 - \alpha_j(k; z))$$

$$= \prod_j \prod_{k=1}^{t_j} \left\{ \frac{\alpha_j(k; z)}{(1 - \alpha_j(k; z))} \right\}^{y_{kj}} (1 - \alpha_j(k; z)) \qquad (3.24)$$

em que $y_{kj} = 1$ se o *j*-ésimo cliente cancelou o serviço no tempo $T_j = k$ e 0 caso contrário. Note que a função de verossimilhança para os dados de sobrevivência agrupados é a mesma verossimilhança de um modelo de resposta binária com probabilidade de evento $\alpha_j(k; z)$. Aplicando-se o logaritmo, a função de verossimilhança pode ser reescrita como:

$$\log L = \sum_{j=1}^{n} \sum_{k=1}^{t_j} \left\{ y_{kj} \log(\alpha_j(k; z)) + (1 - y_{kj}) \log(1 - \alpha_j(k; z)) \right\}. \qquad (3.25)$$

Se assumirmos que os dados foram gerados por um modelo de riscos proporcionais com tempos absolutamente contínuos que foram, posteriormente, agrupados temos:

$$P(T_j = a_i; z) = [1 - \lambda_i^{\exp(z_j \beta)}] \prod_{k=1}^{i-1} \lambda_i^{\exp(z_j \beta)} \qquad (3.26)$$

em que,

$$\lambda_i = \exp\left[- \int_{a_i - 1}^{a_i} \alpha_0(u) du \right] \qquad (3.27)$$

é a probabilidade de sobrevivência condicional em I_i para um cliente com vetor de covariáveis $z = 0$. A probabilidade do cliente não cancelar o contrato no início do intervalo I_i é:

$$P(T_j > a_i; z) = \prod_{k=1}^{i} \lambda_k^{\exp(z_j\beta)}. \tag{3.28}$$

Para o cálculo da função de verossimilhança, Prentice e Gloeckler (1978) substituem λ_k, $0 < \lambda_k < 1$, por $\gamma_k = \log(-\log(\lambda_k))$ para que a restrição sobre este parâmetro seja retirada, pois além de ser um dos passos para que as aproximações assintóticas da verossimilhança sejam adequadas, melhora a convergência do método Newton-Raphson para o cálculo dos estimadores de máxima verossimilhança. Com isso a função de verossimilhança pode ser reescrita como:

$$\begin{aligned} \log L = \sum_{j=1}^{n}\sum_{k=1}^{t_j} \{ & y_{kj} \log\{1 - \exp[-\exp(\gamma_k + z_j\beta)]\} \\ & - (1 - y_{kj}) \exp(\gamma_k + z_j\beta)\} \end{aligned} \tag{3.29}$$

pois,

$$\begin{aligned} \alpha_j(k; z) &= 1 - \lambda_k^{\exp(z_j\beta)} \\ &= 1 - \exp(-\exp(\gamma_k))^{\exp(z_j\beta)} \\ &= 1 - \exp(-\exp(\gamma_k + z_j\beta)) \end{aligned} \tag{3.30}$$

sendo que o vetor de coeficientes β é idêntico ao do modelo de riscos proporcionais com tempo absolutamente contínuo e γ_k é uma constante relacionada à probabilidade de sobrevivência condicional no intervalo definido por $T_j = k$ com $z_j = 0$. O modelo de sobrevivência com dados agrupados torna-se, portanto, equivalente ao modelo de resposta binária com função de ligação complementar log-log (Agresti, 1990).

O processo iterativo de Newton-Raphson para a obtenção da estimativa de máxima verossimilhança $(\hat{\gamma}, \hat{\beta})$ é definido expandindo-se a função escore em torno de um valor inicial. A função escore pode ser escrita como:

$$\frac{\partial \log L}{\partial \gamma_l} = \frac{\partial\left[\sum_{j=1}^{n}\sum_{k=1}^{t_j}\left\{y_{kj}\log\{1-\exp[-\exp(\gamma_k+z_j\beta)]\}-(1-y_{kj})\exp(\gamma_k+z_j\beta)\right\}\right]}{\partial\gamma_l}$$

$$= \sum_{j=1}^{n}\left\{y_{lj}\left[\frac{1}{1-\exp[-\exp(\gamma_l+z_j\beta)]}\left[-\exp[-\exp(\gamma_l+z_j\beta)]\right]\left[-\exp(\gamma_l+z_j\beta)\right]\right]\right.$$

$$\left.-(1-y_{lj})\exp(\gamma_l+z_j\beta)\right\}$$

$$= \sum_{j=1}^{n}\left\{y_{lj}\left[\frac{\exp[-\exp(\gamma_l+z_j\beta)]\exp(\gamma_l+z_j\beta)}{1-\exp[-\exp(\gamma_l+z_j\beta)]}\right]\right\}$$

$$-(1-y_{lj})\exp(\gamma_l+z_j\beta)\Big\}. \qquad (3.31)$$

e,

$$\frac{\partial \log L}{\partial \beta_m} = \frac{\partial\left[\sum_{j=1}^{n}\sum_{k=1}^{t_j}\left\{y_{kj}\log\{1-\exp[-\exp(\gamma_k+z_j\beta)]\}-(1-y_{kj})\exp(\gamma_k+z_j\beta)\right\}\right]}{\partial\beta_m}$$

$$= \sum_{j=1}^{n}\sum_{k=1}^{t_j}\left\{y_{kj}\left[\frac{1}{1-\exp[-\exp(\gamma_k+z_j\beta)]}\left[-\exp[-\exp(\gamma_k+z_j\beta)]\right]\left[-\exp(\gamma_k+z_j\beta)\right]z_{jm}\right]\right.$$

$$\left.-(1-y_{kj})\exp(\gamma_k+z_j\beta)z_{jm}\right\}$$

$$= \sum_{j=1}^{n}\sum_{k=1}^{t_j}\left\{y_{kj}\left[\frac{\exp[-\exp(\gamma_k+z_j\beta)]\exp(\gamma_k+z_j\beta)z_{jm}}{1-\exp[-\exp(\gamma_k+z_j\beta)]}\right]\right.$$

$$\left.-(1-y_{kj})\exp(\gamma_k+z_j\beta)z_{jm}\right\}. \qquad (3.32)$$

Os estimadores de máxima verossimilhança $(\hat{\gamma}, \hat{\beta})$ são as soluções para:

$$c' = \left(\frac{\partial \log L}{\partial \gamma_1}, \ldots, \frac{\partial \log L}{\partial \gamma_k}, \frac{\partial \log L}{\partial \beta_1}, \ldots, \frac{\partial \log L}{\partial \beta_s} \right) = (0, \quad ..., \quad 0). \tag{3.33}$$

O método de Newton-Raphson para o cálculo de $(\hat{\gamma}, \hat{\beta})$ necessita ainda da segunda derivada do log L. A informação "observada" de Fisher pode ser escrita como:

$$I = \begin{bmatrix} I_{11} & I_{12} \\ I_{21} & I_{22} \end{bmatrix} = \begin{bmatrix} \dfrac{-\partial^2 \log L}{\partial \gamma \partial \gamma} & \dfrac{-\partial^2 \log L}{\partial \gamma \partial \beta} \\ \dfrac{-\partial^2 \log L}{\partial \beta \partial \gamma} & \dfrac{-\partial^2 \log L}{\partial \beta \partial \beta} \end{bmatrix} \tag{3.34}$$

Os componentes de *I* podem ser escritos como (ver apêndice B):

$$-\frac{\partial^2 \log L}{\partial \gamma_l \partial \gamma_l} = \sum_{j=1}^{n} \left\{ y_{lj} \left[\frac{\exp[-\exp(\gamma_l + z_j \beta)] \exp(\gamma_l + z_j \beta)}{[1 - \exp[-\exp(\gamma_l + z_j \beta)]]^2} \right] \right.$$

$$\left. \cdot [\exp[-\exp(\gamma_l + z_j \beta)] + \exp(\gamma_l + z_j \beta) - 1] + (1 - y_{lj}) \exp(\gamma_l + z_j \beta) \right\} \tag{3.35}$$

$$-\frac{\partial^2 \log L}{\partial \gamma_l \partial \gamma_h} = 0, \; l \neq h \tag{3.36}$$

$$-\frac{\partial^2 \log L}{\partial \gamma_l \partial \beta_m} = z_{jm} \left(-\frac{\partial^2 \log L}{\partial \gamma_l \partial \gamma_l} \right) \tag{3.37}$$

$$-\frac{\partial^2 \log L}{\partial \beta_m \partial \beta_n} = z_{jm} z_{jn} \sum_{j=1}^{n} \sum_{k=1}^{t_j} \left\{ y_{kj} \left[\frac{\exp[-\exp(\gamma_k + z_j \beta)] \exp(\gamma_k + z_j \beta)}{[1 - \exp[-\exp(\gamma_k + z_j \beta)]]^2} \right] \right.$$

$$\left. \cdot [\exp[-\exp(\gamma_k + z_j \beta)] + \exp(\gamma_k + z_j \beta) - 1] + (1 - y_{kj}) \exp(\gamma_k + z_j \beta) \right\} \tag{3.38}$$

As estimativas de máxima verossimilhança de $(\hat{\gamma}, \hat{\beta})$ são obtidas através da atualização dos valores iniciais (chute) via:

$$\begin{bmatrix} \gamma_1 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} \gamma_0 \\ \beta_0 \end{bmatrix} + I_0^{-1} c_0 \tag{3.39}$$

onde I_0 e c_0 representam I e c assumindo γ_0, β_0 (chutes iniciais). O procedimento iterativo continua até que o desejável nível de precisão seja alcançado. Um chute inicial simples é assumirmos $\beta_0 = 0$ e $\gamma_0 = \hat{\gamma}(0)$, o estimador de máxima verossimilhança quando $\beta = 0$. Seja n_i o número de clientes em risco de cancelamento, este número é calculado subtraindo do número de clientes em risco de cancelamento no início do período I_{i-1}, o número total de cancelamentos e o número total de censuras nesse mesmo período, e d_i o número de cancelamentos no intervalo I_i. Assim, o *i*-ésimo componente de $\hat{\gamma}(0)$ é:

$$\hat{\gamma}_i(0) = \log\left[-\log\left(1 - \frac{d_i}{n_i} \right) \right]. \tag{3.40}$$

Além de fornecer os mesmos coeficientes β de um modelo de riscos proporcionais, esse modelo também fornece o efeito basal de cada intervalo de tempo γ_k que pode ser traduzido para o componente de risco basal via:

$$\alpha_0(t) = 1 - \exp(-\exp(\gamma_k)) \tag{3.41}$$

A estimação direta da função de risco basal é muito útil, por ela mesma, e por facilitar a estimativa da função de risco individual. Uma vez que a função de risco $\alpha_j(t; z)$ é estimada para o cliente *j*, a função de sobrevivência usual é estimada por:

$$\hat{S}_j(t_k) = \prod_{t=0}^{t_k} (1 - \hat{\alpha}_j(t_k; z)) \tag{3.42}$$

em que,

$$\hat{\alpha}_j = (t_k; z) = \hat{\alpha}_0(t_k) \exp(z\hat{\beta}) \tag{3.43}$$

e

$$\hat{\alpha}_0(t_k) = 1 - \exp(-\exp(\hat{\gamma}_k)) \tag{3.44}$$

e a mediana do tempo de duração do contrato pode ser estimada como sendo o valor interpolado de *t* quando $S_j(t_k) = 0{,}5$.

Propriedades assintóticas para os estimadores obtidos em (3.39) foram derivados por Andersen e Gill (1982).

Capítulo 4

Redes neurais artificiais

A pesquisa em Redes Neurais Artificiais (RNA), mais comumente conhecidas apenas como "Redes Neurais", foi motivada em seu início pelo reconhecimento de que o cérebro humano trabalha de uma maneira totalmente diferente dos computadores convencionais. O cérebro é altamente complexo, não-linear e com processamento paralelo (sistema de processamento de informações). Ele tem a capacidade de organizar seus componentes estruturais, conhecidos como neurônios, para executar certas tarefas (como por exemplo, reconhecimento de padrões, percepção e controle motor) mais rápido que muitos computadores existentes na atualidade. Neste capítulo vamos apresentar uma visão geral das Redes Neurais Artificiais. Maiores detalhes podem ser encontrados em Bishop (1995), Haykin (1999) e Ohtoshi (2003).

4.1 Definições

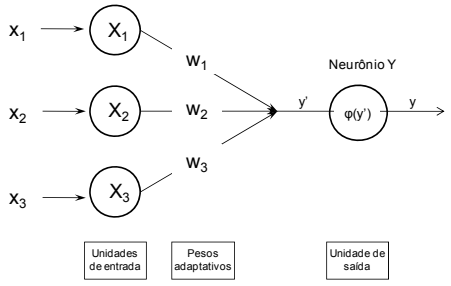
A Rede Neural Artificial (RNA) é um sistema de processamento paralelo de informações inspirado no funcionamento do cérebro humano. Ela é constituída de unidades simples de processamento interligadas por um sistema de conexões que

armazenam o conhecimento experimental e respondem da mesma maneira sempre que recebem sinais similares. As Redes Neurais Artificiais têm as seguintes características:

1.) os sinais são transmitidos através de elementos de processamento simples chamados de neurônios, nós ou unidades;
2.) os sinais trafegam entre os neurônios através de conexões;
3.) cada conexão tem um fator multiplicativo associado (peso) que é aplicado ao sinal transmitido;
4.) a cada neurônio se aplica uma função de ativação (usualmente não linear). A função é aplicada ao valor de entrada do neurônio e limita a amplitude da saída do mesmo. Essa função é escolhida de acordo com o intervalo de valores desejado na saída do neurônio.

A rede neural é formada por um conjunto de neurônios interligados através de conexões, que têm um fator multiplicativo associado. Cada neurônio aplica uma função às entradas recebidas, chamada de função de ativação. Como no cérebro, o conhecimento do ambiente é adquirido pela rede através de um processo de aprendizado e os pesos (associados a cada conexão) são utilizados para armazenar o conhecimento adquirido.

Figura 4.1: Neurônio artificial simples



Como exemplo, considere uma rede neural artificial simples, conforme Figura 4.1, constituída pelos neurônios (unidades) X_1, X_2 e X_3 na camada de entrada com valores (sinais) x_1, x_2 e x_3 e pesos w_1, w_2 e w_3 associados, respectivamente, a esses neurônios. O

valor de entrada, *y'*, do neurônio *Y* é a soma ponderada dos valores de entrada x_1, x_2 e x_3 pelos pesos w_1, w_2 e w_3, ou seja,

$$y' = w_1 x_1 + w_2 x_2 + w_3 x_3 \tag{4.1}$$

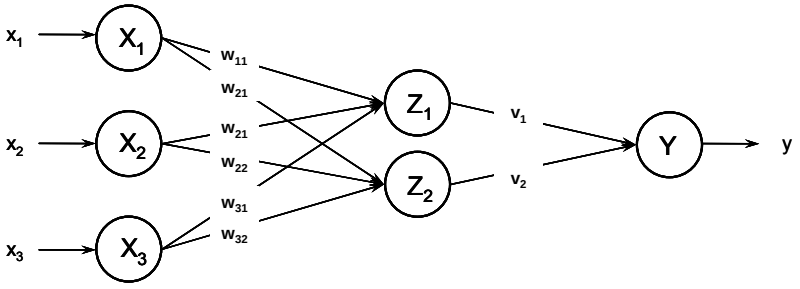
A ativação do neurônio *Y* é dada por alguma função da entrada da rede, $y = \varphi(y')$, em que φ é em geral uma função não linear como, por exemplo, a função logística. Logo, o valor da saída da rede neural será:

$$y = \frac{1}{1 + \exp\left(- \sum_{i=1}^{3} w_i x_i \right)}. \tag{4.2}$$

Note que existe uma grande semelhança com os modelos estatísticos usuais: com as unidades de entradas equivalentes às covariáveis da regressão, os pesos equivalentes aos coeficientes das covariáveis e o valor alvo equivalente à variável resposta ou dependente.

Tipicamente, uma RNA é formada por unidades de entrada, unidades escondidas e unidades de saída ponderadas por pesos chamados pesos adaptativos ou sinápticos. Como exemplo, veja a Figura 4.2 a seguir referente a uma rede neural com duas camadas de pesos adaptativos.

Figura 4.2: Rede Neural com duas camadas de pesos adaptativos



Para a rede neural representada na Figura 4.2, o valor da saída da rede neural pode ser escrito como:

$$y = \varphi(y') = \varphi(z_1 v_1 + z_2 v_2) \tag{4.3}$$

em que:

$$z_i = \phi(z_i') = \phi(\sum_{k=1}^{3} x_k w_{ki}),\ i = 1,2 \tag{4.4}$$

sendo φ e ϕ funções não lineares quaisquer.

Uma RNA é caracterizada principalmente pelos seguintes fatores:

1.) a arquitetura de rede – a forma como os neurônios estão distribuídos pela rede;
2.) o algoritmo de treinamento (ou aprendizado) – método utilizado para determinar os pesos nas conexões da rede;
3.) a função de ativação – função aplicada ao valor de entrada do neurônio e que limita a amplitude da saída do mesmo.

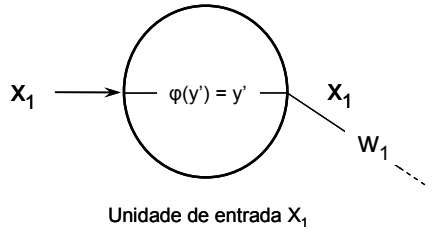
4.2 Tipos de arquitetura

A forma mais comum para se organizar os neurônios é arranjá-los em camadas, onde os neurônios de uma mesma camada, geralmente, possuem a mesma função de ativação e estão conectados aos mesmos conjuntos de neurônios das outras camadas. Veja, por exemplo, a Figura 4.2 em que a camada formada pelos neurônios Z_1 e Z_2 estão conectados aos neurônios X_1, X_2 e X_3 da camada de entrada e ao neurônio Y da camada de saída.

Em geral, na camada de entrada, a ativação de cada unidade é igual ao próprio valor de entrada externo, como se aplicássemos a função identidade ao valor de entrada

externo, na Figura 4.3 está apresentada a unidade de entrada *X1* da rede neural da Figura 4.1.

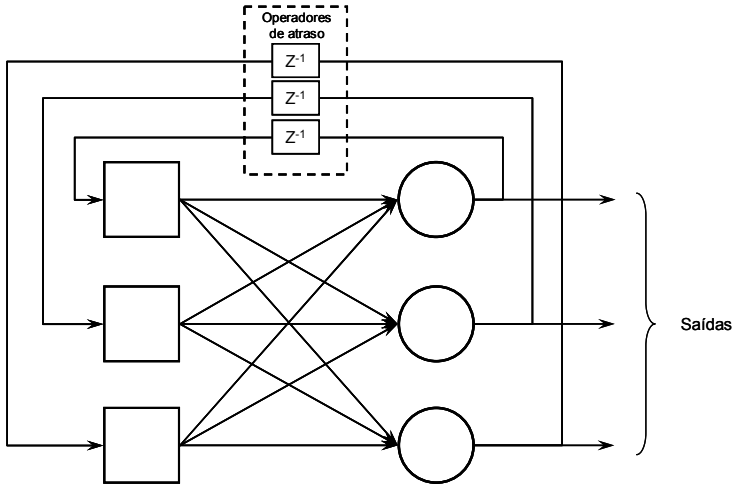
Figura 4.3: Unidade de entrada X1



Quando os sinais de uma rede neural fluem das unidades de entrada para as unidades de saída em uma direção para frente, temos o que são chamadas redes *feed forward.* São exemplos de redes *feed forward* as Figuras 4.1 e 4.2.

Além das redes *feed forward* encontramos outro tipo de rede, que é a rede recorrente, onde os neurônios não dependem somente dos valores de entrada, mas também dos seus próprios valores defasados. Este tipo de rede é semelhante ao processo utilizado na regressão com médias móveis da análise de séries temporais. Um exemplo é a rede de Hopfield (Hopfield, 1984) que consiste em um conjunto de neurônios e um conjunto de unidades de defasagem, formando um sistema recorrente com múltiplos laços, em que um caso particular é o da Figura 4.4.

Figura 4.4: Rede Neural de Hopfield (Recorrente)



Podemos classificar as RNA em redes com uma camada ou multicamada, dependendo da quantidade de camadas de pesos adaptativos.

4.2.1 Redes com uma camada

Uma RNA com uma única camada é aquela que tem apenas uma camada de pesos adaptativos. Como exemplo veja a Figura 4.1.

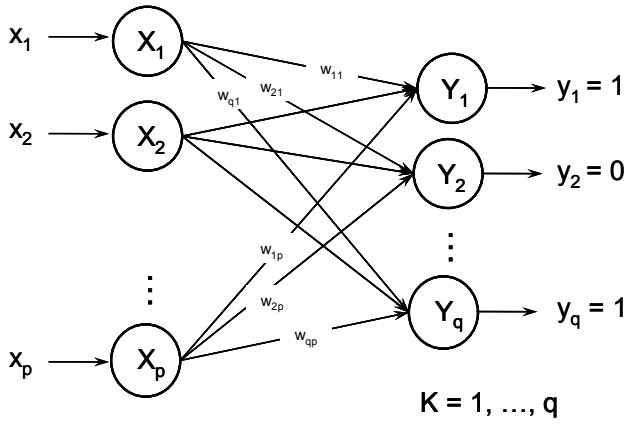
Note que se considerarmos a rede neural com apenas uma camada e assumirmos como função de ativação a função logística temos um modelo usual de Regressão Logística e se assumirmos como função de ativação a função identidade, teremos uma Regressão Linear Múltipla.

Como exemplos de redes com apenas uma camada podemos citar:

Neurônio de McCulloch-Pitts

O modelo de McCulloch-Pitts (McCulloch e Pitts, 1943) é considerado como a primeira Rede Neural Artificial. Neste modelo, se a soma ponderada dos valores provenientes das unidades de entrada for maior que um determinado valor limiar a saída tem valor 1, caso contrário, o valor de saída é 0. Veja a Figura 4.5 a seguir.

Figura 4.5: Exemplo de neurônio com uma camada



O valor de saída do *k*-ésimo neurônio é:

$$y_k = \varphi\left(\sum_{j=1}^{p} w_{kj} x_j\right), \tag{4.5}$$

em que y_k é o valor de saída do neurônio Y_k, $k = 1, ..., q$, w_{kj} é o peso associado à conexão entre o neurônio X_j e o neurônio Y_k, x_j é o valor da j-ésima unidade (neurônio) de entrada e φ é a função de ativação definida por:

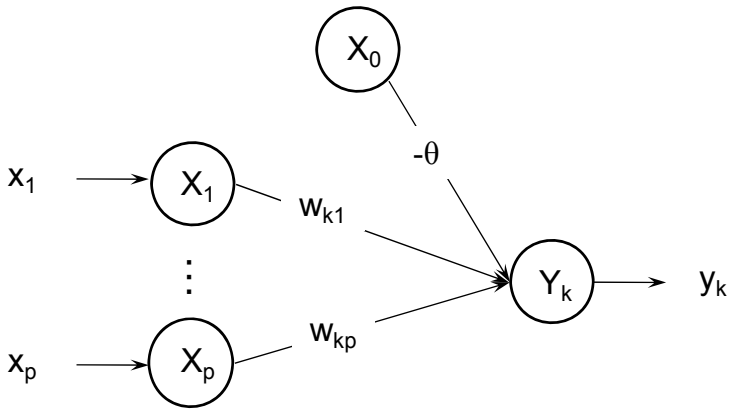
$$\varphi(y') = \begin{cases} 1, & se \quad y' > \theta \\ 0, & se \quad y' \leq \theta \end{cases}$$

em que o θ é o valor limiar.

Perceptron de Rosenblatt

O perceptron (Rosenblatt, 1958) é construído a partir do modelo de neurônio de McCulloch-Pitts onde é incorporado um viés externo. Geralmente essas redes têm uma única camada de pesos fixos e função de ativação limiar.

Figura 4.6: Perceptron



O perceptron é equivalente a uma análise discriminante linear, sendo que a função discriminante é dada por:

$$y_k = \varphi\left(\sum_{j=1}^{p} w_{kj} x_j - \theta\right) = \varphi\left(\sum_{j=0}^{p} w_{kj} x_j\right), \tag{4.6}$$

em que $w_{k0} = -\theta$ e $x_0 = 1$. Se a análise discriminante é aplicada a duas populações, φ é definida como:

$$\varphi(y') = \begin{cases} 1, & se \quad y' > 0 \\ -1, & se \quad y' \leq 0 \end{cases}$$

A constante $w_{k0} = -\theta$ é referida como limiar ou viés, fazendo referência a que a soma ponderada das entradas deve exceder θ.

4.2.2 Redes multicamadas

Quando uma rede neural possui mais de uma camada de pesos adaptativos, ela é chamada de multicamada. Vale lembrar que esta não é uma terminologia padronizada; por exemplo, veja a rede neural da Figura 4.2. Alguns autores podem classificá-la como uma rede com três camadas (entrada, escondida e saída) ou como uma rede com uma camada (uma camada escondida).

O perceptron multicamada (Rumelhart and McClelland,1986) é a arquitetura de rede multicamada mais comumente utilizada, fazendo dessa a nossa escolha para este trabalho. Ela é formada por perceptrons, apresentados na seção anterior, arranjados em camadas. O treinamento da rede perceptron multicamada (PML) utiliza o algoritmo de aprendizado conhecido como algoritmo de retro propagação ou "Back-propagation" (Haykin, 1999), que será descrito a seguir.

4.3 Algoritmos de treinamento

Os algoritmos de treinamento são utilizados para ajustar os pesos com o objetivo de diminuir a diferença entre a resposta desejada, caso esta seja conhecida, e o valor fornecido pela rede. Do ponto de vista estatístico, o treinamento corresponde à fase de estimação dos parâmetros de um modelo usando as informações de uma amostra.

O treinamento pode ser classificado de duas formas:

- Supervisionado: é aquele em que para cada vetor de entrada se conhece a saída, e assim conhecemos a diferença entre a resposta desejada e o valor fornecido pela rede. Por exemplo, uma regressão linear ou logística em que ajustamos um modelo a partir de valores conhecidos para as covariáveis e a variável resposta.

- Não-supervisionado: neste caso não conhecemos o valor desejado de saída para cada um dos vetores de entrada. Isso equivale aos métodos estatísticos de análise de conglomerados e de componentes principais.

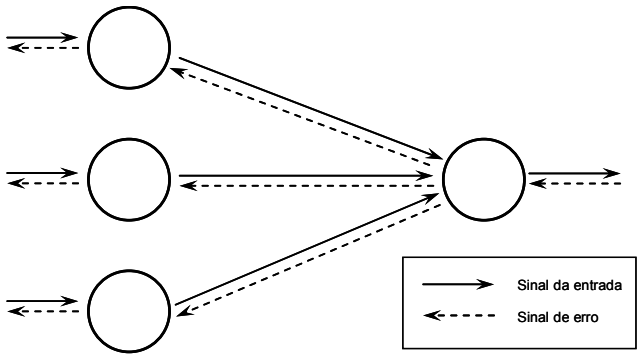
Existem diversos algoritmos utilizados no treinamento supervisionado, dentre os quais destacamos:

- Retro propagação (“Back-propagation”);
- Gradiente Descendente Conjugado;
- Levenberg-Marquardt;
- Delta-bar-Delta;
- Quick-Propagation.

Algoritmo Back-propagation

O algoritmo mais conhecido e utilizado para treinamento de redes neurais é o Back-propagation (Fausett, 1994). Ele é baseado na regra delta (Rumelhart e McClelland, 1986) que define que o valor da correção a ser aplicado aos pesos das conexões é igual a multiplicação de uma taxa de aprendizagem, o gradiente local e o sinal de entrada do neurônio. Para cada registro inserido na rede durante o treinamento, a informação é alimentada através da rede para gerar a previsão na unidade de saída. Esta previsão é comparada com o valor real, e a diferença entre os dois (Erro) é retro propagada pela rede para ajustar os pesos adaptativos para melhorar a predição para padrões similares.

Figura 4.7: Fluxo dos Sinais de um neurônio

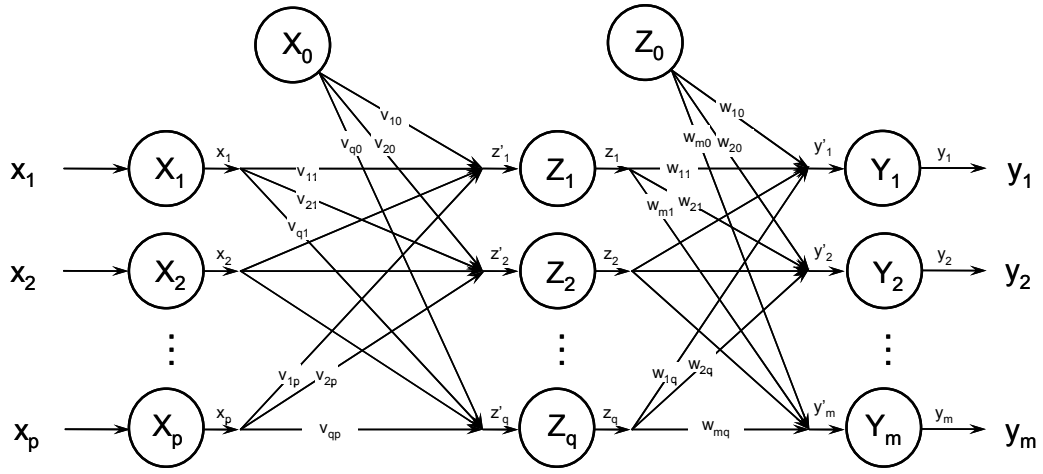


Neste método o vetor gradiente da superfície de erros é calculado. Este vetor aponta na direção da inclinação mais íngreme a partir do ponto atual; movendo-se uma curta distância nessa linha, obtém-se uma diminuição do erro. Com uma seqüência de tais movimentos, eventualmente se chegará a um mínimo. Na prática, o tamanho do passo (distância) que queremos nos mover é proporcional à inclinação e uma constante especial, chamada de taxa de aprendizagem. Quanto menor for a taxa de aprendizagem, menores serão as mudanças dos pesos das conexões na rede neural de uma iteração para próxima, e mais suaves serão as trajetórias no espaço dos pesos. Isso, entretanto, poderá exigir um grande número de iterações até alcançarmos o erro desejado. Em contrapartida, se escolhermos uma taxa de aprendizagem muito grande, isso resulta em grandes mudanças nos pesos das conexões o que poderia tornar a rede neural instável, com movimentos

oscilatórios. A escolha dos pesos iniciais influenciará a chegada a um mínimo global ou local. Usualmente iniciam-se os pesos e vieses com valores entre -0,5 e 0,5 e taxa de aprendizagem η com valor entre 0 e 1.

O treinamento por Back-propagation tem três estágios: o *feed forward* das unidades de entrada, o Back-propagation do erro associado e o ajuste dos pesos e vieses. Como exemplo, considere uma rede com uma camada escondida, como na Figura 4.8. Fixamos os pesos e vieses (v_{hj} e w_{kh}) e a taxa de aprendizagem (η) no passo inicial.

Figura 4.8: Rede neural com uma camada escondida (duas camadas de pesos adaptativos)



Estágio: Feed forward

Nesta etapa, para cada unidade da amostra de treinamento, as unidades de entrada *(x_j, j = 1, 2, ..., p)* transmitem o seu valor para as unidades da camada escondida. Calcula-se o valor para cada camada escondida *(z_h, h = 1, 2, ..., q)*:

$$z'_h = v_{h0} + \sum_{j=1}^{p} x_j v_{hj} \tag{4.7}$$

e aplica-se a função de ativação:

$$z_h = \varphi\left(z'_h\right). \tag{4.8}$$

Este processo é repetido para as unidades de saída *(y_k, k = 1, 2, ..., m)*, ou seja,

$$y'_k = w_{k0} + \sum_{h=1}^{q} z_h w_{kh} \tag{4.9}$$

e, novamente, aplica-se a função de ativação:

$$y_k = \varphi\left(y'_k\right). \tag{4.10}$$

Estágio: Back-propagation

Para cada unidade de saída, o valor alvo que chamaremos de t_k, é conhecido (equivalentemente, como no caso da regressão onde a variável resposta é conhecida na amostra) e pode-se calcular o erro:

$$e_k = \left(t_k - y_k\right). \tag{4.11}$$

O algoritmo foi desenvolvido com o objetivo de obter os pesos que minimizem a função de erros, dada por:

$$E = \frac{1}{2}\sum_{j=1}^{m}\left(t_j - y_j\right)^2. \tag{4.12}$$

Para isso é necessário calcular as derivadas de *E* com relação aos pesos e vieses ou gradiente das superfícies de erros.

A base matemática para o algoritmo Back-propagation é a técnica de otimização conhecida como gradiente descendente. O gradiente de uma função (neste caso, a função é a função de erros e as variáveis são os pesos da rede), quando assume valor positivo, dá a direção em que a função cresce mais rapidamente; o valor negativo dá a direção em que a função decresce mais rapidamente.

Através da aplicação sucessiva da regra da cadeia, temos:

$$\frac{\partial E}{\partial w_{kh}} = \frac{\partial E}{\partial e_k}\frac{\partial e_k}{\partial y_k}\frac{\partial y_k}{\partial y_k^*}\frac{\partial y_k^*}{\partial w_{kh}}, \tag{4.13}$$

diferenciando a Eq. (4.12) em função de e_k:

$$\frac{\partial E}{\partial e_k} = \frac{\partial\left[\frac{1}{2}\sum_{j=1}^{m}\left(t_j - y_j\right)^2\right]}{\partial e_k} = \frac{\partial\left[\frac{1}{2}\sum_{j=1}^{m} e_j{}^2\right]}{\partial e_k} = e_k, \tag{4.14}$$

diferenciando a Eq. (4.11) em função de y_k:

$$\frac{\partial e_k}{\partial y_k} = \frac{\partial\left(t_k - y_k\right)}{\partial y_k} = -1, \tag{4.15}$$

diferenciando a Eq. (4.10) em função de y_k^*:

$$\frac{\partial y_k}{\partial y_k^*} = \frac{\partial\varphi(y_k^*)}{\partial y_k^*} = \varphi'(y_k^*), \tag{4.16}$$

e, finalmente, diferenciando a Eq. (4.9) em função de w_{kh}:

$$\frac{\partial y_k^*}{\partial w_{kh}} = \frac{\partial\left[w_{k0} + \sum_{h=1}^{q} z_h w_{kh} \right]}{\partial w_{kh}} = z_h \,. \tag{4.17}$$

Substituindo as Eqs. (4.14) a (4.17) na Eq. (4.13) temos:

$$\frac{\partial E}{\partial w_{kh}} = -e_k \varphi'(y_k^*) z_h \,. \tag{4.18}$$

A correção Δw_{kh} aplicada em w_{kh} é definida pela *regra delta*:

$$\Delta w_{kh} = -\eta \frac{\partial E}{\partial w_{kh}} \tag{4.19}$$

sendo que η é a taxa de aprendizagem do algoritmo de Back-propagation. O uso do sinal negativo na Eq. (4.19) é justificado pelo gradiente descendente no espaço dos pesos, isto é, a procura da direção da mudança dos pesos que reduz o valor de *E*. Substituindo a Eq. (4.18) na Eq. (4.19) temos:

$$\Delta w_{kh} = \eta \delta_k^{(2)} z_h \tag{4.20}$$

sendo que o gradiente local $\delta_k^{(2)}$ é definido por:

$$\begin{aligned} \delta_k^{(2)} &= -\frac{\partial E}{\partial y_k^*} \\ &= -\frac{\partial E}{\partial e_k} \frac{\partial e_k}{\partial y_k} \frac{\partial y_k}{\partial y_k^*} \\ &= e_k \varphi'(y_k^*) \,. \end{aligned} \tag{4.21}$$

Os termos para correção dos pesos e vieses das conexões posteriores à camada escondida são:

$$\Delta w_{k0} = \eta \delta_k^{(2)} \tag{4.22}$$

e

$$\Delta w_{kh} = \eta \delta_k^{(2)} z_h . \tag{4.23}$$

A partir das Eq. (4.22) e (4.23) nota-se que o fator chave envolvido no cálculo do ajuste dos pesos Δw_{kh} é o sinal do erro e_k da unidade de saída k. Nesse contexto nós podemos identificar dois casos distintos, dependendo de onde a unidade k está localizada. No caso anterior, a unidade k é uma unidade de saída. Este caso é simples porque cada unidade de saída tem um valor alvo, tornando o cálculo do sinal do erro associado direto. Já nas unidades escondidas, elas dividem a responsabilidade sobre o erro nas unidades de saída da rede. A questão, entretanto, é como penalizar ou premiar as unidades escondidas pela sua participação. Isso é resolvido pela retro propagação dos erros através da rede.

Quando um neurônio h está em uma camada escondida da rede, não existe um valor alvo específico esperado. Portanto, o sinal do erro deve ser determinado recursivamente em termos dos erros de todos os neurônios nos quais este neurônio está diretamente conectado. Definimos agora o gradiente local $\delta_h^{(1)}$ para o neurônio escondido h:

$$\begin{aligned} \delta_h^{(1)} &= -\frac{\partial E}{\partial z_h^*} \\ &= -\frac{\partial E}{\partial z_h}\frac{\partial z_h}{\partial z_h^*} \\ &= -\frac{\partial E}{\partial z_h}\varphi'(z_h^*) . \end{aligned} \tag{4.24}$$

Diferenciando a Eq. (4.12) em função de z_h, temos:

$$\frac{\partial E}{\partial z_h} = \sum_k e_k \frac{\partial e_k}{\partial z_h}$$

$$= \sum_k e_k \frac{\partial e_k}{\partial y_k^*} \frac{\partial y_k^*}{\partial z_h} . \tag{4.25}$$

Das Eqs. (4.10) e (4.11),

$$e_k = t_k - \varphi(y_k^*) . \tag{4.26}$$

Então,

$$\frac{\partial e_k}{\partial y_k^*} = -\varphi'(y_k^*) \tag{4.27}$$

diferenciando a Eq. (4.9) em função de z_h:

$$\frac{\partial y_k^*}{\partial z_h} = w_{kh} \tag{4.28}$$

e, utilizando os resultados das Eqs. (4.27) e (4.28) na Eq. (4.25) temos:

$$\frac{\partial E}{\partial z_h} = -\sum_k e_k \varphi'(y_k^*) w_{kh}$$

$$= -\sum_k \delta_k^{(2)} w_{kh} \tag{4.29}$$

sendo que na segunda linha utilizamos a definição do gradiente local dado pela Eq. (4.21). Finalmente substituindo a Eq. (4.29) na Eq. (4.24) encontramos:

$$\delta_h^{(1)} = \varphi'(z_h^*)\sum_k \delta_k^{(2)} w_{kh} \ . \tag{4.30}$$

Os termos para correção dos pesos e vieses das conexões anteriores à camada escondida são:

$$\Delta v_{h0} = \eta \delta_h^{(1)} \tag{4.31}$$

e

$$\Delta v_{hj} = \eta \delta_h^{(1)} x_j \, . \tag{4.32}$$

Estágio: Ajuste dos pesos e vieses

Para os pesos e vieses das conexões posteriores à camada escondida temos que:

$$w_{kh}(atual) = w_{kh}(anterior) + \Delta w_{kh} \tag{4.33}$$

para $h = 0, 1, 2, ..., q$ e $k = 1, 2, ..., m$.

Para os pesos e vieses das conexões anteriores à camada escondida, temos que:

$$v_{hj}(atual) = v_{hj}(anterior) + \Delta v_{hj} \, . \tag{4.34}$$

Em cada iteração do algoritmo, todos os casos da amostra de treinamento (unidades amostrais) são submetidos um a um à rede e os valores de saída real e previsto são comparados através dos erros calculados. Estes erros, juntamente com o gradiente da superfície de erros, são usados para ajustar os pesos e então o processo se repete. Ao final de cada iteração é verificado se um número estipulado de iterações foi atingido ou se o erro total alcançou um nível aceitável; em caso positivo, o processo é interrompido.

Note que, para o desenvolvimento acima, a função de ativação deve ser contínua, diferenciável e monotonicamente não-decrescente.

O algoritmo também pode ser modificado pela inclusão de um termo de momento, o qual reforça o movimento em uma determinada direção, de modo que se muitos passos são tomados em uma mesma direção, o algoritmo aumenta a velocidade para se mover mais rapidamente sobre platôs.

Outros algoritmos de treinamento

Além do Back-propagation, outros algoritmos de treinamento mais sofisticados para otimização da função não linear podem ser utilizados. Algoritmos mais avançados considerando a abordagem de redução do gradiente são: o Gradiente Descendente Conjugado (Bishop, 1995) e Levenberg-Marquardt (Shepherd, 1997), que são mais rápidos do que o Back-propagation em muitos problemas. E ainda temos os algoritmos Delta-bar-Delta (Jacobs, 1988) e Quick-Propagation (Fahlman, 1988) são variações do algoritmo Back-propagation.

4.4 Funções de ativação

A função de ativação, denotada por $\varphi(.)$, define a saída de um neurônio em termos de valores que podem ser uma soma ponderada para o caso de neurônios na camada escondida ou um valor vindo da amostra, caso seja uma unidade de entrada. Note que para as unidades de entrada a função de ativação é a função identidade. Lembre que a função de ativação é a mesma para todos os neurônios de uma camada, mas podem diferir de uma camada para outra.

Entre as funções de ativação mais comuns podemos citar:

Função identidade:

$$\varphi(y') = y' .$$

Função indicadora:

$$\varphi(y') = \begin{cases} 1, & se \quad y' > \theta \\ 0, & se \quad y' \leq \theta \end{cases}$$

em que θ é o valor limiar.

Função logística:

$$\varphi(y') = \frac{1}{1 + \exp(-y')}, \quad y' \in (0,1) .$$

Função tangente hiperbólica:

$$\varphi(y') = \tanh\left(\frac{y'}{2}\right) = \frac{1 - \exp(-y')}{1 + \exp(-y')}, \qquad y' \in (-1,1) .$$

Função normal:

$$\varphi(y') = \phi(y')$$

em que ϕ é a função de distribuição acumulada da Normal padrão.

A escolha da função de ativação define o intervalo dos valores de saída do neurônio, se utilizarmos a função logística o intervalo irá variar entre 0 a 1; se necessitarmos que os valores variem entre -1 a 1 podemos escolher, por exemplo, a função tangente hiperbólica.

4.5 Redes neurais em problemas de análise de sobrevivência

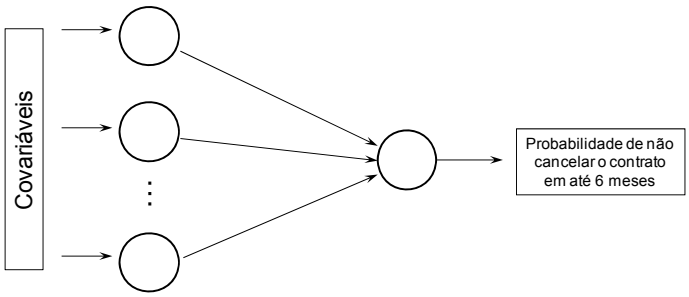
Diferentemente das situações em que o interesse é a ocorrência ou não de um determinado evento, na Análise de Sobrevivência o interesse é o tempo até um determinado evento, que no nosso caso é o cancelamento voluntário do contrato, que pode ou não ocorrer. Nesses casos o tratamento correto das censuras é essencial. A simples exclusão das observações censuradas pode introduzir vieses na predição dos eventos.

Muitas estratégias têm sido desenvolvidas a fim de adaptar as redes neurais aos dados censurados (censura à direita). Vamos apresentar a seguir alguns métodos que foram desenvolvidos para a estimação do tempo de vida.

Classificação direta

Este é o método mais simples, que considera a sobrevivência em um período fixo de tempo e, conseqüentemente, resulta em um problema de classificação binária (veja Figura 4.9). As observações censuradas são removidas, introduzindo assim, os vieses. A unidade de saída da rede neural fornece uma estimativa da probabilidade de sobrevivência de um cliente, ou seja, a probabilidade de não cancelar o contrato durante aquele período de tempo. Se o valor for acima de 50% assumimos que o cliente não irá cancelar nesse período. Está claro que este método além de básico, não produz as curvas de sobrevivência ou risco individuais. Além do mais, não lida com o problema da censura e das covariáveis dependentes do tempo. As probabilidades de não cancelamento são assim subestimadas.

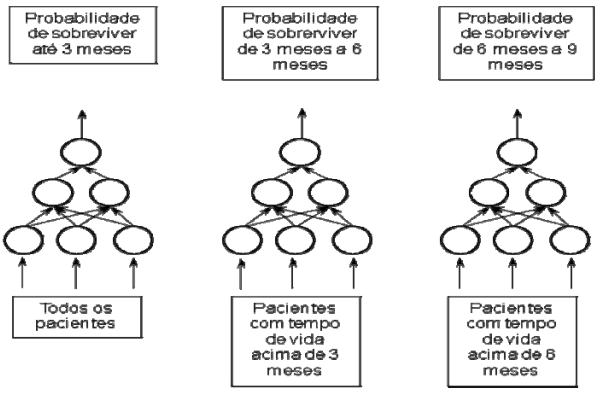
Figura 4.9: Classificação direta



Ohno-Machado

Ohno-Machado (1996) utiliza múltiplas redes neurais para resolver o problema da análise de sobrevivência. Cada rede neural tem uma única unidade de saída que estima a sobrevivência em um determinado período de tempo (Figura 4.10).

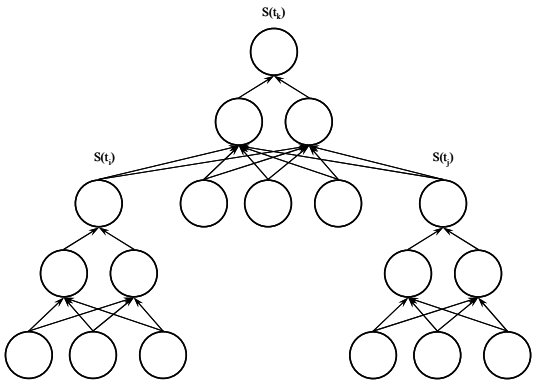
Figura 4.10: Estrutura utilizada por Ohno-Machado (1996)



O banco de dados é particionado de maneira que cada parte corresponda a um período de tempo. Então, cada uma das redes neurais é treinada utilizando somente a parte do banco de dados que lhe corresponde. As observações censuradas são incluídas em todos os períodos de tempo até a sua censura. Note que o número de unidades de treinamento diminui à medida que o período de tempo se distancia do período inicial, o que torna essas estimativas menos confiáveis. A autora alerta que quando utilizamos estas

redes de maneira isolada, podemos obter curvas de sobrevivência não-monotônicas. Como resultado, a probabilidade de um indivíduo sobreviver a dois períodos pode ser maior do que a probabilidade de sobreviver a um período, pois as interdependências das probabilidades de sobrevivência ao longo do tempo não são propriamente levadas em conta quando as redes neurais são treinadas isoladamente. A autora descreve uma maneira de reduzir a freqüência de curvas não-monotônicas combinando as redes neurais. As estimativas de sobrevivência de uma rede são usadas como uma entrada (covariável) adicional para outra rede neural, como ilustrado na Figura 4.11. Entretanto, ainda é possível obter curvas de sobrevivência não-monotônicas apesar do número de ocorrências serem menores do que quando as redes não estão conectadas entre elas. Além disso, a maneira como as redes neurais devem ser combinadas ainda é uma questão em aberto. A necessidade de utilizar múltiplas redes neurais e a maneira como combiná-las representa um problema de escala, que torna este método inapropriado para lidar com grandes bancos de dados.

Figura 4.11: Um exemplo de rede neural modular

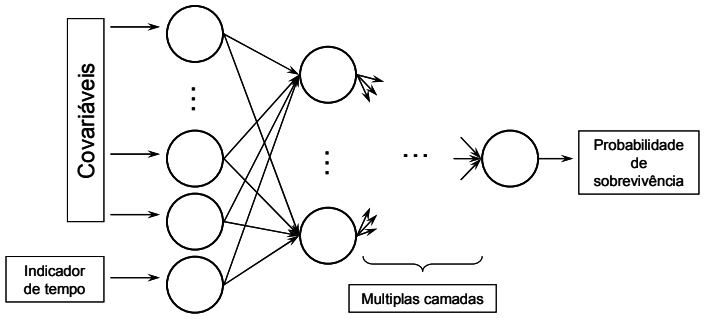


Ravdin e Clark

Ravdin e Clark (1992) utilizam uma rede neural de múltiplas camadas com alimentação para frente ("multi-layer feed forward neural network") com uma única unidade de saída para estimar a probabilidade de sobrevivência. Nesse método, um indicador do tempo e um indicador de status de sobrevivência são incluídos a cada

unidade experimental o que resulta em replicar as observações para todo o intervalo de tempo que está sendo considerado (veja Figura 4.12). O indicador do tempo mostra os sucessivos períodos de tempo *[1, T]* para os quais a predição será feita, com *T* representando o tempo máximo de acompanhamento. Uma unidade experimental não-censurada será replicada *T* vezes enquanto uma unidade censurada será replicada *t* vezes em que *t* é o tempo de censura. O status de sobrevivência é o alvo da rede e tem valor 0 enquanto o indivíduo não sofreu o evento de interesse e 1 caso contrário. Apesar das entradas (covariáveis) dependentes do tempo não serem discutidas no estudo original, elas podem ser facilmente incluídas nos intervalos de tempo correspondentes. Os autores colocam que a saída da rede neural é aproximadamente proporcional à estimativa de Kaplan-Meier da probabilidade de sobrevivência. Entretanto, eles não garantem que as curvas de sobrevivência geradas serão monotonicamente decrescentes. Além disso, replicar as unidades experimentais gera dois problemas. Primeiro, ela resulta em grandes vieses porque o número de eventos nos intervalos mais tardios será superestimado e, portanto, precisam ser corrigidos por amostras seletivas. Segundo, enquanto este método não necessita o treinamento de várias redes neurais, ele gera bancos de dados muito grandes o que causa sérios problemas de escala.

Figura 4.12: Rede neural utilizada por Ravdin e Clark (1992)



Biganzoli *et al*

Uma variação do método proposto por Ravdin e Clark (1992) foi sugerida por Biganzoli *et al* (1998). Neste método, eles também utilizam uma rede neural com uma saída e um indicador do tempo como uma entrada adicional. Entretanto, diferente de

Ravdin e Clark, as unidades experimentais não-censuradas são replicadas até o intervalo de tempo onde elas são realmente observadas, ou seja, as unidades experimentais que sofreram o evento de interesse não são incluídas nos intervalos de tempo posteriores ao intervalo do evento. Novamente, entradas (covariáveis) dependentes do tempo podem ser facilmente incluídas, pois cada unidade experimental tem múltiplos vetores de entrada que podem se alterados entre os intervalos de observação. A rede neural fornece taxas de falhas que podem ser convertidas em probabilidades de sobrevivência monotônicas. Entretanto, este método também encontra problemas de escala, pois, ao replicar os dados, ele gera grandes bancos de dados.

Liestol *et al*

Liestol *et al* (1994) propõem dois métodos distintos, um para tempos de sobrevivência contínuos e outro para tempos de sobrevivência agrupados.

Para tempos de sobrevivência contínuos eles sugerem uma partição do tempo em *m* intervalos e uma função de erros específica. Neste caso a rede neural a ser implementada possui as mesmas restrições do método proposto para tempos de sobrevivência agrupados e uma função de ativação $\varphi(y') = \exp(y')$. Então são definidas *m* + 1 valores alvo onde os *m* primeiros componentes são os tempos de sobrevivência nos intervalos e o último componente indica se a unidade experimental é observada até o evento. Os autores citam que é possível incluir entradas dependentes do tempo replicando uma observação em duas (ou mais) observações censuradas.

Para tempos de sobrevivência agrupados eles utilizam uma rede com apenas uma camada e *T* unidades de saída, sendo que *T* representa o tempo máximo de acompanhamento. É imposta a restrição de que todas as conexões vindas de uma mesma unidade de entrada têm o mesmo peso, isto é, $w_{1j} = w_{2j} = \cdots = w_{Tj} = \beta_j, j = 1, ..., p$. O valor alvo das unidades de saída tem valor 0 enquanto a unidade experimental não sofreu o evento de interesse até esse período e tem valor 1, caso contrário. A rede neural fornece, dependendo da função de ativação escolhida, a probabilidade condicional de

falha para dados agrupados sugerida por Cox (1972), caso a função de ativação seja a função logística, ou a probabilidade condicional de falha sugerida por Prentice e Gloeckler (1978) caso a função de ativação escolhida seja $\varphi(y') = 1 - \exp(-\exp(y'))$. A probabilidade condicional de falha pode ser convertida em uma curva de sobrevivência monotônica, entretanto este método não permite entradas dependentes do tempo e não pode ser facilmente implementada nos programas de redes neurais comuns.

Lapuerta *et al*

Lapuerta *et al* (1995) sugerem uma estratégia de múltiplas redes para imputar tempos de sobrevivência para dados censurados. Para cada período de tempo considerado, uma rede neural separada é construída. Estas redes são treinadas utilizando somente as observações para as quais o status de sobrevivência no período de tempo correspondente é conhecido. Em seguida, as redes treinadas são utilizadas para prever o status dos casos censurados. As observações não-censuradas e censuradas (imputadas) são então utilizadas para treinar a rede neural principal que estima a probabilidade de sobrevivência para cada intervalo de tempo considerado. O método proposto ainda não garante que as probabilidades de sobrevivência serão monotonicamente decrescentes e também não permite entradas (covariáveis) dependentes do tempo. Além disso, fica claro que este método não é apropriado para aplicações em larga escala por exigir o treinamento um número de redes neurais igual ao número de intervalos de tempo considerados.

Faraggi e Simon

Faraggi e Simon (1995) propuseram uma variante do modelo de riscos proporcionais de Cox, descrito na Eq. (3.16), a função linear $z\beta$ é substituída pela saída $g(z;\theta)$ de uma rede neural com uma camada escondida com função de ativação logística e uma unidade de saída com função de ativação linear:

$$\alpha(t;z) = \alpha_0(t)\exp(g(z,\theta)). \qquad (4.35)$$

Analogamente ao modelo de Cox, nenhum viés é considerado na unidade de saída, pois ele está implicitamente incorporado ao risco basal $\alpha_0(t)$. Os parâmetros θ são estimados utilizando o princípio de verossimilhança parcial e a otimização de Newton-Raphson. Este método preserva todas as vantagens do modelo de riscos proporcionais clássicos. Entretanto, ele assume que os riscos são proporcionais. Embora covariáveis dependentes do tempo e/ou estratificações podem lidar com os problemas de não proporcionalidade, essas generalizações podem não ser a melhor maneira para modelar a variação basal porque podem aumentar significativamente a complexidade da rede neural a ser construída.

Street

Street (1998) utiliza uma rede Perceptron Multicamada (PML) com *T* unidades de saída, em que *T* representa o tempo máximo de acompanhamento (veja Figura 4.13). Uma função de ativação tangente hiperbólica é utilizada na camada de saída de maneira que todas as unidades de saída têm valores entre -1 e 1. O primeiro neurônio de saída com valor menor que 0 é considerado como o neurônio que prediz o tempo até o evento. Se todos os neurônios de saída têm valores maiores que 1, então consideramos que a unidade experimental irá sobreviver a todo o período de acompanhamento. As unidades de saída, portanto, representam a probabilidade de sobrevivência para o correspondente período de tempo.

Para os casos não-censurados, o valor alvo das unidades de saída tem valor 1 enquanto a unidade experimental não sofrer o evento de interesse e valor -1 após isso. Para os casos censurados, o valor alvo das unidades de saída tem valor 1 até o tempo de censura. Após esse período, o autor utiliza as estimativas de Kaplan-Meier das equações (3.14) e (3.15) para estimar a probabilidade de sobrevivência das observações censuradas após o tempo de censura das mesmas. Então, essas probabilidades são padronizadas para o intervalo da função tangente hiperbólica utilizando a fórmula:

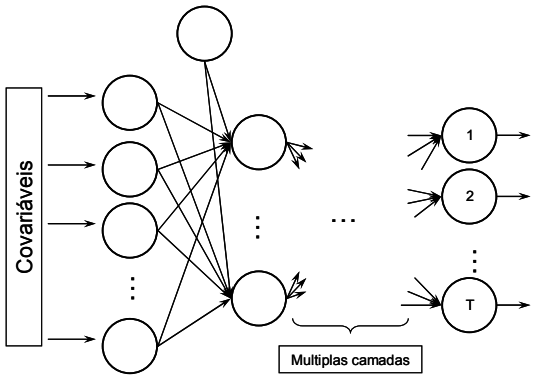
$$ativação = 2 \times probabilidade - 1. \tag{4.36}$$

O valor alvo das unidades de saída é definido por:

$$S(t) = \begin{cases} 1, & se \quad 1 \le t \le L \\ -1, & se \quad \delta = 1 \quad e \quad L < t \le T \\ S(t-1) \times (1 - \alpha(t)), & se \quad \delta = 0 \quad e \quad L < t \le T \end{cases}$$

em que T representa o tempo máximo de acompanhamento, L é o mínimo entre o tempo do evento e da censura e δ indica se a unidade experimental é censurada ($\delta=0$) ou não ($\delta=1$). A curva de sobrevivência individual pode ser obtida baseada nos valores de ativação das unidades de saída. Como não é possível forçar a rede neural a gerar unidades de saída monotonicamente decrescentes, uma curva de sobrevivência não-monotônica ainda é possível, o que complica a interpretação. Além disso, nenhuma extensão é apresentada para trabalhar com entradas dependentes do tempo.

Figura 4.13: Rede neural utilizada por Street (1998)



Mani *et al*

Uma variação do método de Street (1998) foi desenvolvida por Mani *et al* (1999). Novamente, para cada observação do banco de dados de treinamento, T unidades de saídas são computadas. Mas, neste método, essas unidades de saída representam a taxa de falha ao invés da probabilidade de sobrevivência que foi utilizada por Street (1998). O valor alvo das unidades de saída é definido por:

$$\alpha(t) = \begin{cases} 0, & se \quad 1 \le t \le L \\ 1, & se \quad \delta = 1 \quad e \quad L < t \le T \\ \dfrac{d_t}{n_t}, & se \quad \delta = 0 \quad e \quad L < t \le T \end{cases}$$

Novamente, T representa o número máximo de períodos acompanhados pelo estudo, L é o mínimo entre o tempo do evento e da censura e δ indica se a unidade experimental é censurada ($\delta=0$) ou não ($\delta=1$). Para observações não-censuradas, o risco é 0 até o tempo do evento e 1 depois disso. Para observações censuradas, o risco é 0 até o tempo de censura e é substituído pelo estimador de Kaplan-Meier depois disso. As curvas de sobrevivência geradas serão monotonicamente decrescentes o que facilita a interpretação. Entretanto, não existe uma extensão para as entradas (covariáveis) dependentes do tempo.

Brown *et al*

Analogamente a Mani *et al* (1999), Brown *et al* (1997) sugerem uma única rede neural com múltiplas unidades de saída para estimar a taxa de falha. Para as observações não-censuradas, o alvo da rede neural é 0 enquanto a unidade experimental estiver em risco e 1 depois que sofreu o evento. Para os intervalos de tempo seguintes ao evento, o risco não é definido. O valor alvo para as observações censuradas é 0 até o tempo de censura e também não são definidos para os períodos subseqüentes. Os autores então sugerem treinar uma rede neural que minimize a soma quadrática dos erros e não façam correções nos pesos e vieses nos casos onde o risco não estiver definido, assumindo os erros correspondentes como sendo 0. Este método apresenta escala e resulta em curvas de sobrevivência monotonicamente decrescentes. Novamente, nenhuma extensão é apresentada para as entradas dependentes do tempo.

Pela revisão da literatura apresentada anteriormente, fica claro que para grandes bancos de dados, os métodos apresentados por Faraggi e Simon, Mani *et al* e Brown *et al*

são os mais interessantes. Os três métodos permitem que sejam geradas curvas de sobrevivência monotonicamente decrescentes e necessitam apenas que uma rede neural seja treinada. Apesar do método proposto por Faraggi e Simon também permitir entradas dependentes do tempo, ele é menos flexível na modelagem da variação basal. Por outro lado, enquanto os métodos propostos por Mani *et al* e Brown *et al* têm essa flexibilidade, eles não resolvem o problema das entradas dependentes do tempo. Entretanto, o método proposto por Mani *et al* possui, comparado aos outros dois métodos, uma vantagem adicional que é a facilidade de aplicação pois não precisamos alterar as equações para o cálculo dos erros para os ajustes dos pesos e vieses. Logo, esse método pode ser aplicado utilizando-se qualquer pacote computacional que contenha os métodos usuais de redes neurais, fazendo dele a nossa escolha para este trabalho.

Capítulo 5

Resultados

Neste capítulo apresentamos o resultado das análises dos dados apresentados no Capítulo 2. O principal objetivo é avaliar fatores que influenciam o tempo de cancelamento efetivo de contratos, a partir da primeira solicitação de cancelamento que é revertida pelos operadores da companhia de TV por assinatura. São consideradas análises baseadas em técnica de análise de sobrevivência para dados agrupados e em técnica de redes neurais, seguindo o método proposto por Mani *et al* (1999) para a análise de dados com censura.

Inicialmente será apresentada a forma como o banco de dados foi particionado, seguido pelo tratamento e a seleção das covariáveis. A partição do banco de dados em amostra de treinamento, teste e validação é especialmente importante quando utilizamos a técnica de redes neurais porque podemos ter um problema de superestimação ("overfitting") que ocorre quando a rede neural é treinada por um tempo excessivo e, assim, apresenta um alto acerto na amostra de treinamento e um resultado ruim na amostra de teste e/ou validação do modelo. Na seqüência são apresentadas as técnicas descritas anteriormente utilizando essas covariáveis e os resultados finais.

Para a técnica de análise de sobrevivência para dados agrupados foi utilizado o pacote SAS 9.1 (proc logistic) e para a técnica de redes neurais foi utilizado o pacote Clementine 10.0.

5.1 Partição do banco de dados

Para compor a base de modelagem foram selecionados, do banco de dados descrito no Capítulo 2, todos os clientes que cancelaram o contrato e uma amostra aleatória com 3697 casos dos 27767 censurados, isto é, que não cancelaram seus contratos até o fim do período de acompanhamento. Esta base foi particionada em: base de treinamento, base de teste e base de validação, utilizando o método de partição disponível no software Clementine (versão 10). Este método mantém as mesmas proporções das covariáveis e das variáveis resposta encontradas na base original nas três bases geradas. Foram selecionados, aproximadamente, 50% dos clientes para a base de treinamento (3663 clientes), 25% para a base de teste (1835 clientes) e 25% para a base de validação dos resultados (1896 clientes).

5.2 Tratamento das covariáveis

A grande parte do banco de dados é composta por covariáveis categóricas, sendo que as covariáveis contínuas presentes em nosso banco de dados foram categorizadas de modo que fizesse sentido às regras de negócio da empresa.

Quando utilizamos a técnica de rede neural com apenas uma única variável resposta, usualmente, utilizamos as técnicas de árvores de decisão como uma pré-seleção das covariáveis que devem ser utilizadas na entrada da rede. Isso é necessário porque quanto maior o número de entradas maior será o tempo de treinamento, que pode chegar a dias dependendo do tamanho do banco de dados. Como no nosso caso além das 12 variáveis resposta, temos o problema das censuras, essa técnica de pré-seleção não se torna viável. Optou-se, portanto, em pré-selecionar as covariáveis através das estatísticas

usuais em análise de sobrevivência: os testes de Log-Rank e Breslow (Klein and Moeschberger, 2003). Iniciamos o processo com 94 covariáveis e finalizamos a pré-seleção com 33 covariáveis, reduzindo assim, significativamente, o tempo de treinamento da rede neural. Foram consideradas para a modelagem as covariáveis que apresentaram níveis descritivos menores que 5%, e suas categorias foram transformadas em variáveis dicotômicas (Hosmer and Lemeshow, 1989).

Como exemplo podemos citar a variável tempo de contrato, cujos valores foram categorizados em 7 grupos. Esses grupos foram separados de acordo com o ciclo de vida do cliente definido pela área de relacionamento com clientes da empresa e foram transformados em variáveis *dummy* (veja Tabela 5.1).

Tabela 5.1: Transformação das variáveis categóricas

	Variáveis dummy					
Tempo de contrato (meses)	Tenure1	Tenure2	Tenure3	Tenure4	Tenure5	Tenure6
0 ├── 8	0	0	0	0	0	0
8 ├── 12	1	0	0	0	0	0
12 ├── 15	0	1	0	0	0	0
15 ├── 18	0	0	1	0	0	0
18 ├── 20	0	0	0	1	0	0
20 ├── 24	0	0	0	0	1	0
Acima de 24	0	0	0	0	0	1

Note que não é necessário criarmos uma variável para o Tenure entre 0 e 8 meses, pois ele está representado quando todas as variáveis *dummy* criadas para a variável tempo de contrato são iguais a 0 e é a referência para o tempo de contrato.

5.3 Aplicação das técnicas

5.3.1 Análise de sobrevivência para dados agrupados

Para utilizarmos esta técnica, o banco de dados precisa ser modificado da seguinte maneira: Cada unidade de tempo de cada cliente é tratada como uma observação. Para

cada uma dessas observações a variável dependente (alvo) é codificada com 1 se o cliente cancelou o contrato naquela unidade de tempo e 0 caso contrário. Portanto se o cancelamento ocorreu no tempo 5 para um determinado cliente, 5 diferentes observações são criadas. Para a quinta observação a variável dependente será codificada como 1, para as demais quatro observações a variável dependente deverá ser codificada como 0 (veja Tabela 5.2). Para estimar as constantes $\alpha_0(t)$, t = 0, 1, 2, ..., uma variável *dummy* deve ser criada para cada unidade de tempo possível menos 1, ou seja, foram criadas 11 novas variáveis (M1 a M11).

Tabela 5.2: Exemplo de banco de dados para dados agrupados

Identificação	**…**	**Tempo**	**Churn**	**Alvo**	**M1**	**M2**	**M3**	**M4**	**…**	**M11**
123	…	3	1	0	1	0	0	0	…	0
123	…	3	1	0	0	1	0	0	…	0
123	…	3	1	1	0	0	1	0	…	0
432	…	4	0	0	1	0	0	0	…	0
432	…	4	0	0	0	1	0	0	…	0
432	…	4	0	0	0	0	1	0	…	0
432	…	4	0	0	0	0	0	1	…	0

Feito isso foi utilizado o "proc logistic" do pacote SAS 9.1 para ajustarmos um modelo complementar log-log para dados binários. Foi escolhido o método de Newton-Raphson para a estimação de máxima verossimilhança dos parâmetros. O modelo final foi obtido através do método Stepwise. As estimativas dos parâmetros, os erros padrão, a estatística de Wald e os níveis descritivos (p-valor) encontram-se na Tabela 5.3.

Tabela 5.3: Modelo complementar log-log para dados binários

Covariável	g.l.	Estimativa	Erro padrão	Wald	p-valor
M1	1	-1,994	0,200	99,895	<0,001
M2	1	-0,355	0,131	7,332	0,007
M3	1	0,118	0,124	0,910	0,340
M4	1	0,104	0,126	0,679	0,410
M5	1	0,379	0,124	9,375	0,002
M6	1	0,520	0,124	17,692	<0,001
M7	1	0,479	0,127	14,283	0,000
M8	1	0,298	0,134	4,925	0,027
M9	1	0,222	0,140	2,529	0,112
M10	1	-0,070	0,153	0,212	0,645
M11	1	-0,295	0,166	3,168	0,075
ARN	1	-0,365	0,124	8,744	0,003
EMAIL	1	-0,285	0,048	35,005	<0,001
AJCOMB (3 a 6 meses)	1	-0,276	0,122	5,132	0,024
CAMPANHA (Adesão Zero)	1	-0,141	0,056	6,420	0,011
CEP (Grupo 1)	-	0,000	-	-	-
CEP (Grupo 2)	1	0,370	0,100	13,821	0,000
CEP (Grupo 3)	1	0,541	0,098	30,756	<0,001
CEP(Grupo 4)	1	0,846	0,102	68,787	<0,001
CHVOLCOMB (7 a 12 meses)	1	0,580	0,170	11,651	0,001
OFERTA (Sem ofertas)	-	0,000	-	-	-
OFERTA (Valor baixo)	1	-0,393	0,082	22,938	<0,001
OFERTA (Valor alto)	1	-0,158	0,066	5,786	0,016
OFERTA (Pacote especial)	1	0,158	0,063	6,344	0,012
RCCOMB (0 a 3 meses)	1	0,324	0,072	20,306	<0,001
STIULT (acima de 10 meses)	1	-0,383	0,191	4,012	0,045
STTA	-	0,000	-	-	-
STTA (1 a 6 meses)	1	-1,623	0,146	123,006	<0,001
STTA (6 a 12 meses)	1	-1,392	0,127	120,256	<0,001
TENURE (até 7 meses)	-	0,000	-	-	-
TENURE (8 a 11 meses)	1	-1,641	0,227	52,314	<0,001
TENURE (12 a 14 meses)	1	-0,938	0,245	14,612	0,000
TENURE (15 a 17 meses)	1	-0,571	0,246	5,402	0,020
TENURE (18 a 20 meses)	1	-0,932	0,246	14,405	0,000
TENURE (20 a 23meses)	1	-1,051	0,249	17,775	<0,001
TENURE (acima de 24 meses)	1	-1,224	0,235	27,051	<0,001
TERMOTEMP (9 meses ou mais)	1	-1,956	0,205	91,025	<0,001

5.3.2 Redes Neurais

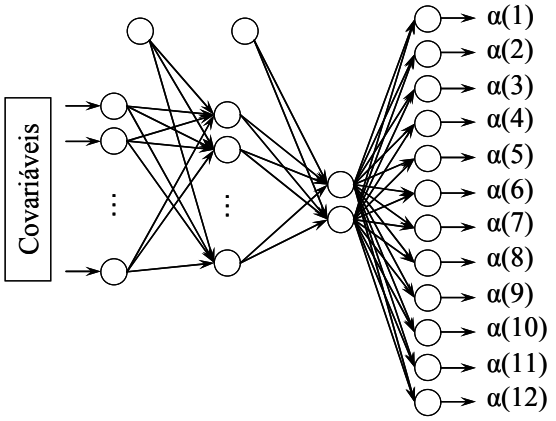
Seguindo o método proposto por Mani *et al* (1999) foram criadas 12 novas variáveis representando a taxa de cancelamento em cada um dos meses de acompanhamento, conforme descrito no Capítulo 4 (ver Tabela 5.4).

Tabela 5.4: Exemplo do banco de dados assumindo que as estimativas de Kaplan-Meier para as taxas de cancelamento para t = 3, 4, 5 e 12 sejam 0,06, 0,05, 0,03 e 0,1, respectivamente.

Identificação	**…**	**Tempo**	**Churn**	**α(1)**	**α(2)**	**α(3)**	**α(4)**	**α(5)**	**…**	**α(12)**
123	…	3	1	0	0	0	1	1	…	1
134	…	2	0	0	0	0,06	0,05	0,03	…	0,1
412	…	4	0	0	0	0	0	0,03	…	0,1
432	…	12	1	0	0	0	0	0	…	1

Utilizando o pacote Clementine 10.0, foi ajustada uma rede Perceptron Multicamada (PML) com o algoritmo de treinamento Back-propagation com função de ativação logística. A rede neural final, após um período de processamento de aproximadamente 1 hora em um computador comum, apresentou 44 covariáveis na entrada e 39 nós primeira camada intermediária de 2 nós na segunda camada intermediária (veja Figura 5.1).

Figura 5.1: Rede neural final



As covariáveis utilizadas foram:

- AJCOMB
- ARN
- BILLN
- CEP
- CSN
- CAMPANHA
- CHVOLCOMB
- CREDM
- CREDTOTAL
- CREDVR
- EMAIL
- MOPCB02
- OFERTA
- PPVACOMB
- PACKDOWN02
- RCCOMB
- SATINCULT
- STIULT
- STTA
- TENURE
- TERMOTEMP

Note que temos 44 unidades de entradas, isso ocorre porque as covariáveis citadas acima foram transformadas em variáveis *dummy*. Os resultados serão apresentados na próxima seção.

5.4 Comparação dos modelos

Para a comparação dos modelos serão utilizadas duas técnicas bem conhecidas: Área sob a curva ROC (Hanley and McNeil, 1982) e a Estatística de Kolmogorov-Smirnov (Conover, 1999) e uma nova técnica que consiste na área entre as curvas (AEC) de distribuição acumulada utilizadas para o cálculo da estatística de Kolmogorov-Smirnov (KS) proposta por Tomazela (2007). Essas técnicas serão aplicadas em dois pontos do tempo distintos 8 e 12 meses.

A área sob a curva ROC (Receiver Operating Characteristic) está associada ao poder discriminante de um modelo ajustado. Com ele é possível estudar a variação da sensibilidade (probabilidade do modelo prever que o cliente irá cancelar o contrato dado que ele cancelou efetivamente o mesmo) e especificidade (probabilidade do modelo prever que o cliente irá continuar com o contrato dado que o cliente não cancelou o mesmo) para diferentes pontos de corte; não sendo necessário assumir um ponto de corte arbitrário para, no nosso caso, a probabilidade de sobrevivência. A curva é obtida calculando, para cada ponto de corte, o valor da especificidade e sensibilidade e, em seguida, os valores obtidos são colocados em um gráfico bidimensional. A área sob a curva é obtida por integração e quanto maior o valor da área, maiores serão os valores de sensibilidade e especificidade alcançados, ou seja, o modelo fornece altas probabilidades de sobrevivência aos clientes que não cancelaram e baixas probabilidades aos que cancelaram efetivamente o contrato.

A estatística de Kolmogorov-Smirnov (KS) é uma estatística não paramétrica que tem como objetivo testar se as funções de distribuição de dois grupos são iguais. Logo, esta estatística pode ser utilizada para verificar se a probabilidade de sobrevivência predita tem a mesma distribuição entre os que cancelaram e os que ainda não cancelaram o contrato. Então, esperamos que os clientes que cancelaram o contrato estejam concentrados nos valores de probabilidade de sobrevivência mais baixos enquanto que para os clientes que não cancelaram o contrato haja uma predominância de valores de

probabilidades de sobrevivência mais altos. O valor KS do modelo é a maior diferença entre as distribuições acumuladas dos clientes que cancelaram e que não cancelaram o contrato. Esse valor pode variar de 0% a 100% e, quanto maior a diferença maior será o poder discriminatório do modelo ajustado. Tão importante quanto o valor desta medida é também o seu posicionamento na distribuição. Os melhores modelos apresentam alto KS nos primeiros decis da distribuição do escore.

A área entre as curvas de distribuição acumulada dos escores (AEC) é uma medida proposta a fim de avaliar inteiramente as duas funções de distribuição acumulada utilizadas no cálculo do KS, e não apenas a maior diferença entre elas. O cálculo da área é feito através da soma das áreas dos retângulos, em que a base de cada retângulo é formada pela diferença entre dois escores consecutivos e a altura pela função de distribuição acumulada empírica dos clientes que não cancelaram o contrato até cada escore. Analogamente, esse método é utilizado para o cálculo da área dos clientes que cancelaram o contrato. A estimativa da área entre as curvas é obtida pela diferença absoluta entre as duas áreas e, de maneira similar ao KS, quanto maior a área melhor e discriminação entre os clientes que irão e os que não irão cancelar o contrato.

Nas Tabelas 5.5 e 5.6 estão dispostos os valores da área sob a curva ROC e seus respectivos intervalos de confiança para cada um dos tempos de previsão fixados.

Tabela 5.5: Quadro comparativo da área sob a curva ROC para t = 8 meses

Técnica	Amostra					
	Treinamento		Teste		Validação	
	Area	IC(95%)	Area	IC(95%)	Area	IC(95%)
Método clássico	0,643	[0,625; 0,661]	0,656	[0,631; 0,682]	0,636	[0,611; 0,662]
Rede Neural	0,664	[0,646; 0,682]	0,653	[0,627; 0,679]	0,649	[0,623; 0,674]

Tabela 5.6: Quadro comparativo da área sob a curva ROC para t = 12 meses

Técnica	Amostra					
	Treinamento		Teste		Validação	
	Area	IC(95%)	Area	IC(95%)	Area	IC(95%)
Método clássico	0,668	[0,651; 0,685]	0,664	[0,639; 0,689]	0,656	[0,632; 0,681]
Rede Neural	0,654	[0,636; 0,672]	0,639	[0,614; 0,664]	0,627	[0,602; 0,653]

Observando as tabelas acima podemos notar que a rede neural é melhor quanto t = 8 meses e pior para t = 12 meses, de qualquer forma, não encontramos nenhuma diferença significativa entre as duas técnicas. Mas podemos observar que no método clássico a performance melhora à medida que o período de previsão aumenta.

Nas Tabelas 5.7 e 5.8 estão dispostos os valores da estatística KS para cada um dos tempos de previsão fixados.

Tabela 5.7: Quadro comparativo da estatística KS para t = 8 meses

Técnica	Amostra		
	Treinamento	Teste	Validação
Método clássico	20,24%	21,93%	20,24%
Rede Neural	28,61%	23,82%	22,96%

Tabela 5.8: Quadro comparativo da estatística KS para t = 12 meses

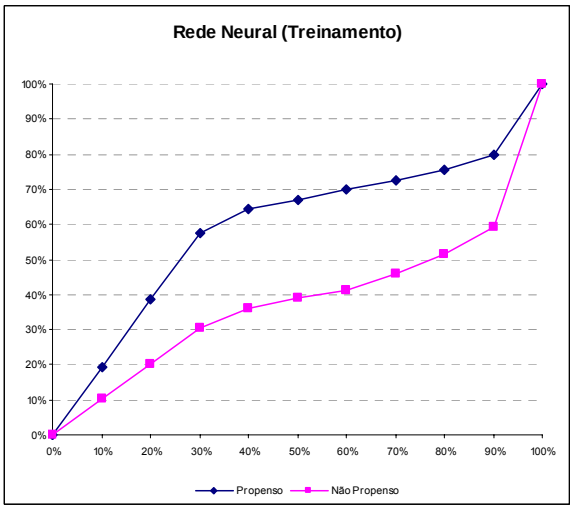
Técnica	Amostra		
	Treinamento	Teste	Validação
Método clássico	23,84%	23,13%	22,53%
Rede Neural	29,36%	24,15%	23,71%

Observando as estatísticas de KS apresentadas acima podemos notar que o modelo de redes neurais possui uma performance superior quando comparado ao método clássico, tanto na base de treinamento quanto na base de teste e também na base de validação. Nota-se também que o desempenho das duas técnicas aumenta a medida que o

período de previsão aumenta, mas de maneira mais acentuada na técnica clássica de análise de sobrevivência para dados agrupados.

Nas Gráficos 5.1 a 5.6 estão dispostos os gráficos da medida KS para cada uma das bases e para cada um dos tempos de previsão fixados.

Gráfico 5.1: Gráficos da medida KS para amostra de treinamento (t = 8 meses)



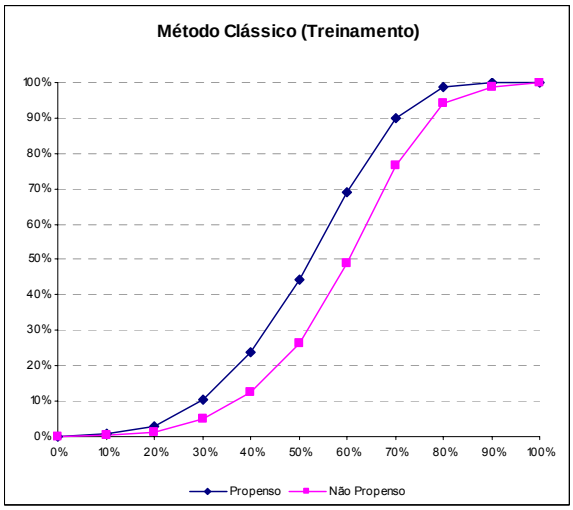
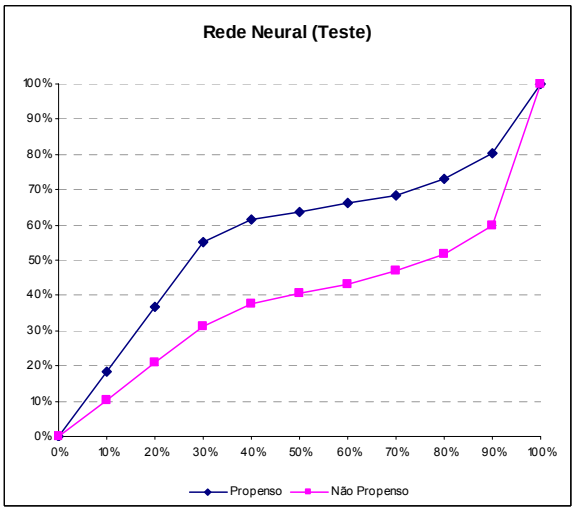


Gráfico 5.2: Gráficos da medida KS para amostra de teste (t = 8 meses)



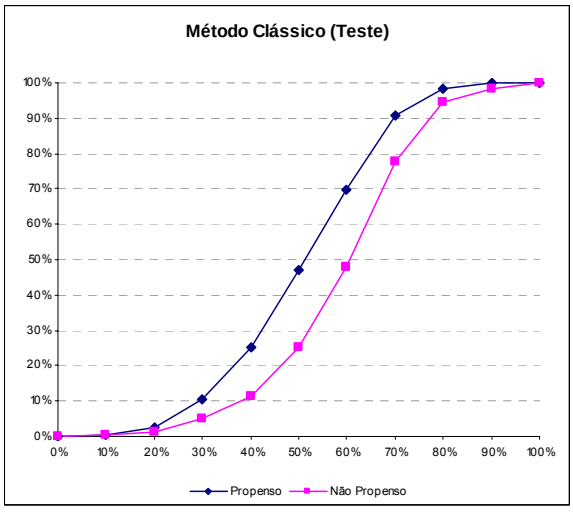


Gráfico 5.3: Gráficos da medida KS para amostra de validação (t = 8 meses)

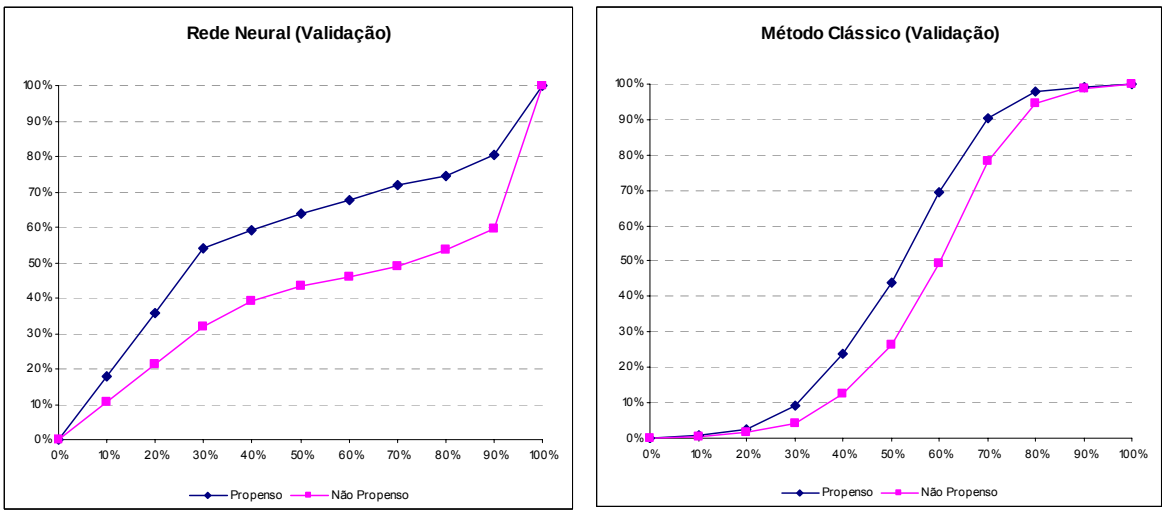


Gráfico 5.4: Gráficos da medida KS para amostra de treinamento (t = 12 meses)

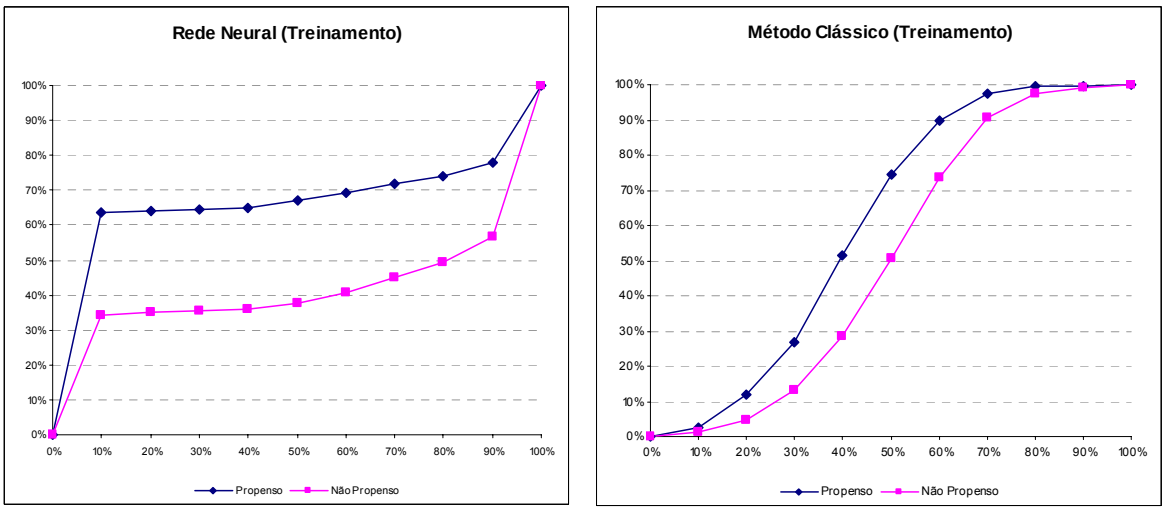


Gráfico 5.5: Gráficos da medida KS para amostra de teste (t = 12 meses)

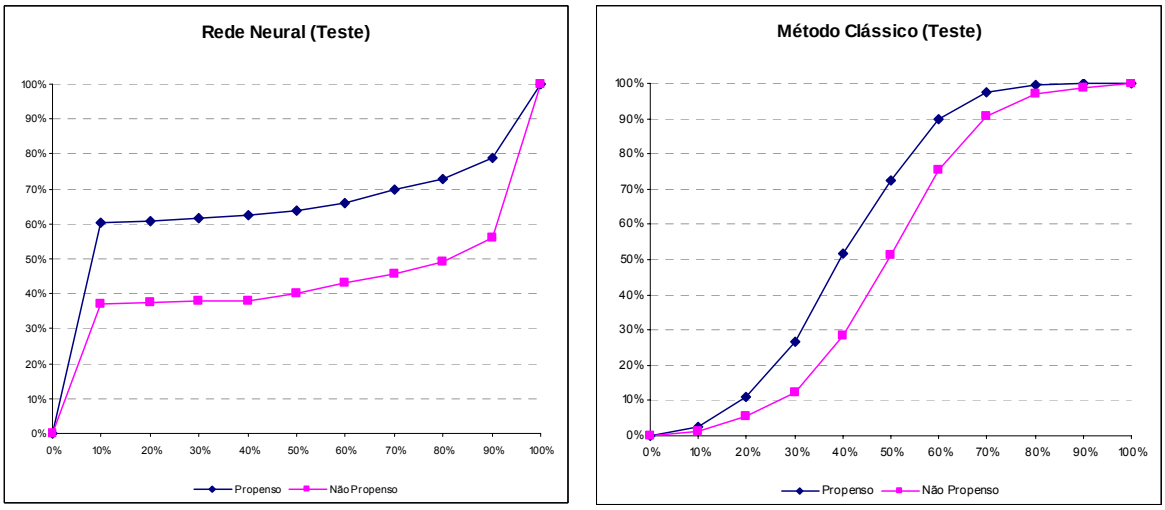
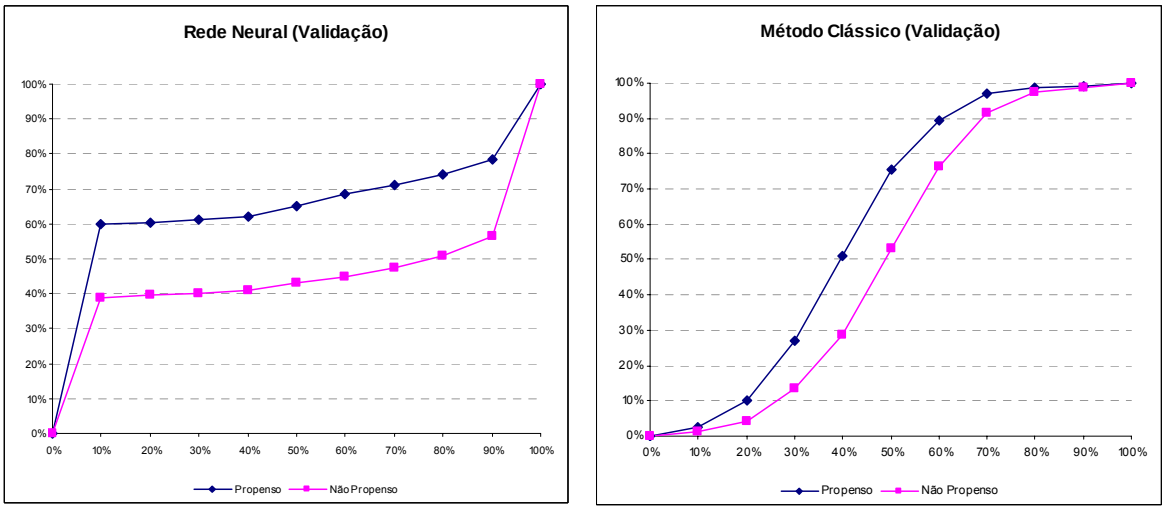


Gráfico 5.6: Gráficos da medida KS para amostra de validação (t = 12 meses)



Observando os gráficos podemos notar que as curvas geradas pela rede neural e pelo método clássico são muito diferentes, apesar da estatística KS não apresentar uma diferença muito significativa. Isso se deve a diferença na distribuição dos clientes ao longo das faixas de escore, a técnica de redes neurais concentrou os clientes nas faixas mais baixas enquanto que na técnica clássica os clientes foram distribuídos em todas as faixas de escore.

Nas Tabelas 5.9 e 5.10 estão dispostos os valores da área entre curvas (AEC) para cada um dos tempos de previsão fixados.

Tabela 5.9: Quadro comparativo da área entre curvas (AEC) para t = 8 meses

Técnica	Amostra		
	Treinamento	Teste	Validação
Método clássico	0,076	0,083	0,071
Rede Neural	0,184	0,157	0,149

Tabela 5.10: Quadro comparativo da área entre curvas (AEC) para t = 12 meses

Técnica	Amostra		
	Treinamento	Teste	Validação
Método clássico	0,095	0,090	0,085
Rede Neural	0,189	0,165	0,156

Como era esperado observando os gráficos de KS, podemos notar que o modelo de rede neural apresenta uma diferença entre a área entre curvas bem superior ao modelo clássico, isto se deve ao fato de que na rede neural a diferença entre as distribuições acumuladas dos clientes que cancelaram e não cancelaram efetivamente o contrato foi alta já nos primeiros decis do escore.

Capítulo 6

Conclusões

Neste trabalho foram desenvolvidos modelos para a estimação do tempo até o cancelamento voluntário do contrato por um cliente em uma empresa de TV por assinatura. Os modelos foram ajustados utilizando duas técnicas: um método clássico de análise de sobrevivência e uma alternativa utilizando redes neurais. O objetivo principal foi a comparação da performance entre as duas técnicas.

Comparando as técnicas através da medida de área sob a curva ROC, apesar da técnica de rede neural apresentar áreas maiores quando t = 8 meses e menores quando t = 12 meses, não encontramos nenhuma diferença significativa, entretanto pudemos observar que a performance da técnica clássica melhora à medida que o período de previsão aumenta. Já nas medidas KS e área entre as curvas (AEC) pudemos notar que o modelo de redes neurais possui uma performance superior quando comparado à técnica clássica, tanto na base de treinamento quanto na base de teste e também na base de validação. Nessas medidas o desempenho das duas técnicas aumenta a medida que o período de previsão aumenta, mas de maneira mais acentuada na técnica clássica. Como era esperado, observando os gráficos de KS, o modelo de rede neural apresenta uma diferença entre a medida de área entre curvas bem superior ao modelo clássico, isto se deve ao fato de que na rede neural a diferença entre as distribuições acumuladas dos

clientes que cancelaram e não cancelaram efetivamente o contrato foi alta já nos primeiros decis do escore. Idealmente, essa diferença nos primeiros decis deveria ocorrer em conjunto com um pequeno percentual de clientes que não cancelaram o contrato, o que de fato não ocorreu. Entretanto, observando as três medidas de comparação podemos dizer que a técnica de rede neural apresenta uma performance superior, ou pelo menos, semelhante à técnica clássica de análise de sobrevivência para dados agrupados.

Na aplicação das técnicas em dados reais, a grande vantagem que a técnica estatística clássica apresenta é a facilidade de interpretação; em contrapartida, a necessidade de replicarmos os dados aumenta a complexidade de uso à medida que o tempo de acompanhamento aumenta. A rede neural, apesar da dificuldade de interpretação, apresentou um resultado mais consistente durante todo o tempo de acompanhamento e, além disso, não necessita que os riscos sejam proporcionais; entretanto, dependendo da quantidade de covariáveis e do tempo de acompanhamento, o treinamento da rede neural pode ser um processo demorado que irá consumir pesados recursos computacionais.

Como sugestão para trabalhos futuros, além da inclusão de covariáveis dependentes do tempo, modelos estratificados e outras formas para o tratamento dos dados censurados podem ser estudadas para as técnicas de redes neurais, como o treinamento de uma rede neural que minimize a soma quadrática dos erros e não façam correções nos pesos e vieses nos casos onde o risco não estiver definido. Além disso, o desenvolvimento de métodos estatísticos para a seleção de variáveis e avaliação da qualidade de ajuste do modelo que ainda não foram bem desenvolvidos na literatura.

Apêndice A

Lista de covariáveis

Apresentamos a seguir a lista de covariáveis contidas no banco de dados:

Nome	Descrição
AJComb	Indica, caso o cliente tenha recebido alguma correção na fatura, o período em que isso ocorreu.
AJ2S	Indica se o cliente recebeu correções na fatura em meses seguidos.
AJUlt	Indica quando o cliente recebeu a última correção na fatura.
ARComb	Indica, caso o cliente tenha ficado inadimplente, o período em que isso ocorreu.
ARMP	Maior quantidade de dias em que o cliente ficou inadimplente.
ARN	Indica se o cliente já ficou inadimplente.
ARUlt	Indica quando o cliente ficou inadimplente pela última vez.
ARUP	Quantidade de dias em que o cliente ficou inadimplente pela última vez.
AT	Indica se o cliente teve problemas técnicos no último mês.
ATComb	Indica, caso o cliente tenha tido um problema técnico, o período em que isso ocorreu.
ATN	Quantidade de meses com problemas técnicos.
AT02	Indica se o cliente teve problemas técnicos nos últimos 3 meses.
AT36	Indica se o cliente teve problemas técnicos de 4 a 3 meses atrás.
AT2MS	Indica se o cliente teve problemas técnicos em dois meses seguidos.
AT3OM	Indica se o cliente teve três ou mais problemas técnicos em um mesmo mês.
ATUlt	Indica há quantos meses o cliente teve o último problema técnico.
ATT	Quantidade total de problemas técnicos nos últimos 6 meses
BILL	Indica se o cliente teve problemas com o faturamento.
BILLComb	Indica, caso o cliente tenha tido um problema com o faturamento, o período em que isso ocorreu.
BILLN	Quantidade de meses com problemas de faturamento.
BILL02	Indica se o cliente teve problemas de faturamento nos últimos 3 meses.
BILL36	Indica se o cliente teve problemas de faturamento de 4 a 6 meses atrás.
BILL2MS	Indica se o cliente teve problemas de faturamento em dois meses seguidos.
BILL3OM	Indica se o cliente teve três ou mais problemas de faturamento em um mesmo mês.

BILLUlt	Indica há quantos meses o cliente teve o último problema com o faturamento.
BILLTotal	Quantidade total de problemas de faturamento.
CallBack	Indica se o receptor está conectado na linha telefônica.
Campanha	Indica a campanha de venda que o cliente recebeu.
CEP	Indica a qual grupo de CEP o cliente pertence.
ChInvComb	Indica, caso o cliente tenha tido um cancelamento involuntário, o período em que isso ocorreu.
ChInvUlt	Indica há quantos meses o cliente teve o último cancelamento involuntário.
ChVolComb	Indica, caso o cliente tenha tido um cancelamento voluntário, o período em que isso ocorreu.
ChVolUlt	Indica há quantos meses o cliente teve o último cancelamento voluntário.
CredMVC	Indica qual foi o maior valor de desconto recebido pelo cliente nos últimos 12 meses.
CredM	Quantidade de meses que o cliente recebeu algum desconto.
CredUlt	Indica há quantos meses o cliente recebeu o ultimo desconto.
CredVr	Valor do desconto recebido pelo cliente no último mês.
CredTotal	Valor total de descontos recebidos pelo cliente nos últimos 12 meses.
CSComb	Indica, caso o cliente tenha tido um contato com o SAC, o período em que isso ocorreu.
CSN	Quantidade de meses que o cliente entrou em contato no SAC
CS2MS	Indica se o cliente entrou em contato com o SAC em dois meses seguidos.
CS3OM	Indica se o cliente entrou em contato com o SAC em um mesmo mês.
CSUlt	Indica há quantos meses o cliente entrou em contato com o SAC pela última vez.
CSTotal	Quantidade total de contatos no SAC nos últimos 6 meses.
Email	Indica se o cliente tem email cadastrado.
SETUPBOX	Indica se o cliente possui mais de um receptor
MOP	Indica o método de pagamento.
MOPCB02	Indica se o cliente alterou o método de pagamento para boleto nos últimos 3 meses.
MOPCB36	Indica se o cliente alterou o método de pagamento para boleto 4 a 6 meses atrás.
MOPCBU	Indica há quantos meses o cliente alterou o método de pagamento para boleto.
Oferta	Indica a oferta de desconto que o cliente recebeu no momento que foi revertido.
Pack	Indica o pacote de programação
PackDown	Indica se o cliente fez um downgrade do pacote de programação
PackDown02	Indica se o cliente fez um downgrade do pacote de programação nos últimos 3 meses
PackDown36	Indica se o cliente fez um downgrade do pacote de programação 4 a 6 meses atrás
PackDown_Comb	Indica, caso o cliente tenha feito um downgrade, o período em que isso ocorreu.
PackDownUlt	Indica há quantos meses o cliente fez um downgrade pela última vez.
PackUp	Indica se o cliente fez um upgrade do pacote de programação
PackUp_Comb	Indica, caso o cliente tenha feito um upgrade, o período em que isso ocorreu.
PackUpUlt	Indica há quantos meses o cliente fez um upgrade pela última vez.
PPVComb	Indica, caso o cliente tenha assistido a um pay-per-view, o período em que isso ocorreu.
PPVAComb	Indica, caso o cliente tenha assistido a um pay-per-view adulto, o período em que isso ocorreu.
PPVAAum	Indica se houve aumento no consumo de pay-per-view adulto.
PPVARed	Indica se houve redução no consumo de pay-per-view adulto.
PPVAUlt	Indica há quantos meses o cliente assistiu a um pay-per-view adulto pela última vez.
Prem	Indica se o cliente possui canais premium
PremEx02	Indica se o cliente cancelou canais premium de sua programação nos últimos 3 meses.
PremEx36	Indica se o cliente cancelou canais premium de sua programação 4 a 6 meses atrás.
PremExUlt	Indica há quantos meses o cliente cancelou canais premium pela última vez.
PremInc02	Indica se o cliente incluiu canais premium de sua programação nos últimos 3 meses.
PremInc36	Indica se o cliente incluiu canais premium de sua programação 4 a 6 meses atrás.
PremIncUlt	Indica há quantos meses o cliente incluiu canais premium pela última vez.

Repack	Indica se o cliente recebeu alguma alteração no pacote
Repack02	Indica se o cliente recebeu alguma alteração no pacote nos últimos 3 meses.
Repack36	Indica se o cliente recebeu alguma alteração no pacote 4 a 6 meses atrás.
RepackUlt	Indica há quantos meses o cliente recebeu alguma alteração no pacote pela última vez.
RC	Indica se o cliente já solicitou cancelamento.
RCComb	Indica, caso o cliente tenha solicitado cancelamento, o período em que isso ocorreu.
RCN	Quantidade de meses que o cliente solicitou cancelamento.
RC2S	Indica se o cliente solicitou cancelamento em dois meses seguidos.
RCPC	Indica se esse foi a primeira solicitação de cancelamento.
RCUlt	Indica há quantos meses o cliente solicitou cancelamento pela última vez.
RCTotal	Quantidade total de solicitações de cancelamento nos últimos 6 meses.
SAT	Indica se o cliente possui suporte técnico.
SATEx02	Indica se o cliente cancelou o suporte técnico nos últimos 3 meses.
SATEx36	Indica se o cliente cancelou o suporte técnico de 4 a 6 meses atrás.
SATExUlt	Indica há quantos meses o cliente cancelou o suporte técnico pela última vez.
SATInc02	Indica se o cliente incluiu o suporte técnico nos últimos 3 meses.
SATInc36	Indica se o cliente incluiu o suporte técnico de 4 a 6 meses atrás.
SATIncUlt	Indica há quantos meses o cliente incluiu o suporte técnico pela última vez.
STIUlt	Indica há quantos meses o cliente estava sem TV por assinatura pela última vez.
STTA	Indica a quantidade de meses que o cliente está ativo
STTI	Indica a quantidade de meses que o cliente estava sem TV por assinatura.
Tenure	Tempo de contrato
Termo	Indica se o cliente possui termo de fidelidade com a empresa.
TermoTemp	Indica o tempo até o término do termo de fidelidade.

Apêndice B

Derivadas de 2ª ordem

Apresentamos aqui os cálculos da derivada de 2ª ordem da Eq. (3.31):

$$\frac{\partial^2 l}{\partial\gamma_l\partial\gamma_l} = \frac{\partial\left[\sum_{j=1}^{n}\left\{y_{lj}\left[\frac{\exp[-\exp(\gamma_l+z_j\beta)]\exp(\gamma_l+z_j\beta)}{1-\exp[-\exp(\gamma_l+z_j\beta)]}\right]-(1-y_{lj})\exp(\gamma_l+z_j\beta)\right\}\right]}{\partial\gamma_l}$$

$$= \sum_{j=1}^{n}\left\{y_{lj}\left[\frac{-1}{[1-\exp(-\exp(\gamma_l+z_j\beta))]^2}[-\exp(-\exp(\gamma_l+z_j\beta))][-\exp(\gamma_l+z_j\beta)]\right.\right.$$

$$\cdot\exp(-\exp(\gamma_l+z_j\beta))\exp(\gamma_l+z_j\beta)$$

$$+\frac{\exp(-\exp(\gamma_l+z_j\beta))[-\exp(\gamma_l+z_j\beta)]}{1-\exp(-\exp(\gamma_l+z_j\beta))}$$

$$\left.\cdot\exp(\gamma_l+z_j\beta)+\frac{\exp(-\exp(\gamma_l+z_j\beta))\exp(\gamma_l+z_j\beta)}{1-\exp(-\exp(\gamma_l+z_j\beta))}\right]$$

$$\left.-(1-y_{lj})\exp(\gamma_l+z_j\beta)\right\}$$

$$= \sum_{j=1}^{n} \left\{ y_{lj} \left[- \left[\frac{\exp(-\exp(\gamma_l + z_j\beta))\exp(\gamma_l + z_j\beta)}{1 - \exp(-\exp(\gamma_l + z_j\beta))} \right]^2 \right.\right.$$

$$- \frac{\exp(-\exp(\gamma_l + z_j\beta))[\exp(\gamma_l + z_j\beta)]^2}{1 - \exp(-\exp(\gamma_l + z_j\beta))}$$

$$\left.\left. + \frac{\exp(-\exp(\gamma_l + z_j\beta))\exp(\gamma_l + z_j\beta)}{1 - \exp(-\exp(\gamma_l + z_j\beta))} \right] - (1 - y_{lj})\exp(\gamma_l + z_j\beta) \right\}$$

$$= \sum_{j=1}^{n} \left\{ y_{lj} \left[- \left[\frac{\exp(-\exp(\gamma_l + z_j\beta))\exp(\gamma_l + z_j\beta)}{1 - \exp(-\exp(\gamma_l + z_j\beta))} \right]^2 \right.\right.$$

$$- \frac{\exp(-\exp(\gamma_l + z_j\beta))[\exp(\gamma_l + z_j\beta)]^2}{[1 - \exp(-\exp(\gamma_l + z_j\beta))]^2}$$

$$+ \frac{[\exp(-\exp(\gamma_l + z_j\beta))\exp(\gamma_l + z_j\beta)]^2}{[1 - \exp(-\exp(\gamma_l + z_j\beta))]^2}$$

$$+ \frac{\exp(-\exp(\gamma_l + z_j\beta))\exp(\gamma_l + z_j\beta)}{[1 - \exp(-\exp(\gamma_l + z_j\beta))]^2}$$

$$\left.\left. - \frac{[\exp(-\exp(\gamma_l + z_j\beta))]^2 \exp(\gamma_l + z_j\beta)}{[1 - \exp(-\exp(\gamma_l + z_j\beta))]^2} \right] - (1 - y_{lj})\exp(\gamma_l + z_j\beta) \right\}$$

$$= \sum_{j=1}^{n} \left\{ y_{lj} \left[\frac{\exp[-\exp(\gamma_l + z_j\beta)]\exp(\gamma_l + z_j\beta)\left[\exp[-\exp(\gamma_l + z_j\beta)] + \exp(\gamma_l + z_j\beta) - 1\right]}{\left[1 - \exp[-\exp(\gamma_l + z_j\beta)]\right]^2} \right] \right.$$

$$\left. - (1 - y_{lj})\exp(\gamma_l + z_j\beta) \right\}$$

Referências bibliográficas

Agresti, A. (1990), *Categorical data analysis*, New York: John Wiley and Sons.

Allison, P. D. (1982), 'Discrete-time methods for the analysis of event histories', *Sociological Methodology* **13**, 61-98.

Andersen, P. K. and Gill, R. D. (1982), 'Cox's regression model for counting process: a large sample study', *Annals of Statistics* **10**, 1100-1120.

Biganzoli, E., Boracchi, P., Mariani, L. and Marubini, E. (1998), 'Feed forward neural networks for the analysis of censored survival data', *Statistics in Medicine* **17**, 1169-1186.

Bishop, C. M. (1995), *Neural networks for pattern recognition*, Oxford: Oxford University Press.

Brown, S. F., Branford, A. J. and Moran, W. (1997), 'On the use of artificial neural networks for the analysis of survival data', *IEEE Transactions on Neural Networks* **8**, 1071-1077.

Conover, W. (1999), *Practical nonparametric statistics*, New York: John Wiley and Sons.

Cox, D. R. (1972), 'Regression models and life-tables', *Journal of the Royal Statistical Society*, Ser. **B 39**, 1-38.

Fahlman, S. E. (1988), 'Faster-learning variations on back-propagation: An empirical study', *Proceedings of the 1988 Connectionist Models Summer School*, Morgan Kaufmann, San Mateo, 38-51.

Faraggi, D. and Simon, R. (1995), 'A neural network model for survival data', *Statistics in Medicine* **14**, 73-82.

Fausett, L. (1994), *Fundamentals of neural networks*, New York: Prentice Hall.

Gupta, S., Hanssens, D., Hardie, B., Kahn, W., Kumar, V., Lin, N. and Sriram, N. R. S. (2006), 'Modeling customer lifetime value', *Journal of Service Research* **9**, 139-155.

Hanley, J. A. and McNeil, R. J. (1982), 'The meaning and use of the area under receiver operating characteristic (ROC) curve', *Radiology* **143**, 29-36.

Haykin, S. (1999), *Neural networks: A comprehensive foundation*, Upper Saddle River, New Jersey: Prentice Hall.

Hopfield, J. J. (1984), 'Neurons with graded response have collective computational properties like those of two-state neurons', *Proceedings of the National Academy of Sciences* **81**, 3088-3092.

Hosmer, J. D. and Lemeshow, S. (1989), *Applied Logistic Regression*, New York: John Wiley and Sons,

Jacobs, R. A. (1988), 'Increased rates of convergence through learning rate adaptation', *Neural Networks* **1(4)**, 295-307.

Kalbfleish, J. D. and Prentice, R. L. (1980), *The statistical Analysis of failure time data*, New York: John Wiley and Sons.

Kaplan, E. L. and Meier, P. (1958), 'Nonparametric estimation from incomplete observations', *Journal of American Statistical Society* **53**, 457-481.

Klein, J. P. and Moeschberger, M. L. (2003), *Survival analysis: Techniques for censored and truncated data*, New York: Springer.

Lapuerta, P., Azen, S. P. and LaBree, L. (1995), 'Use of neural networks in predicting the risk of coronary artery disease', *Computers and Biomedical Research* **28**, 38-52.

Liestol, K., Andersen, P. K. and Andersen, U. (1994), 'Survival Analysis and Neural Nets', *Statistics in Medicine* **13**, 1189-1200.

Mani, D. R., Drew, J., Betz, A. and Datta, P. (1999), 'Statistics and data mining techniques for lifetime value modeling', *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, ACM, San Diego, CA, USA, 94-103.

McCulloch, W. S. and Pitts, W. (1943), 'A logical calculus of the ideas immanent in nervous activity', *Bulletin of Mathematical Biophysics* **5**, 115-133.

Ohno-Machado, L. (1996), 'Sequential use of neural networks for survival prediction in Aids', *Journal of the American Medical Informatics Association* **3**, 170-174.

Ohtoshi, C. (2003), *Uma comparação de regressão logística, árvores de decisão e redes neurais: Analisando dados de crédito*, Dissertação de Mestrado, Universidade de São Paulo

Prentice, R. L. and Gloeckler, L. A. (1978), 'Regression analysis of grouped survival data with application to breast cancer data', *Biometrics* **34**, 57-67.

Ravdin, P. M. and Clark, G. M. (1992), 'A practical application of neural network analysis for predicting outcome of individual breast cancer patients', *Breast Cancer Research and Treatment* **22**, 285-293.

Rosenblatt, F. (1958), 'The perceptron: A probabilistic model for information storage and organization in the brain', *Psychological Review* **65**, 386-408.

Rumelhart, D. E. and McClelland, J. (1986), *Parallel distributed processing: Explorations in the microstructure on cognition, Vol. 1,* Cambridge: MIT Press.

Shepherd, A. J. (1997), *Second-order methods for neural networks*, New York: Springer.

Street, W. N. (1998), 'A neural network model for prognostic prediction', *Proceedings of the Fifteenth International Conference on Machine Learning (ICML).* Morgan Kaufman, Madison, Wisconsin, USA, 540-546.

Tomazela, S. M. O. (2007), *Avaliação de desempenho de modelos de Credit Score ajustados por Análise de Sobrevivência*, Dissertação de Mestrado, Universidade de São Paulo