Rodrigo Pinto Moreira

Modelo de Regressão Logística com Transição Suave Estruturado por Árvore (STLR-Tree)

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós–graduação em Métodos de apoio à decisão do Departamento de Engenharia Elétrica da PUC-Rio

Orientador: Prof. Dr. Álvaro Veiga

Rio de Janeiro
Abril de 2008

Rodrigo Pinto Moreira

Modelo de Regressão Logística com Transição Suave Estruturado por Árvore (STLR-Tree)

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós–graduação em Métodos de apoio à decisão do Departamento de Engenharia Elétrica do Centro Técnico Científico da PUC-Rio. Aprovada pela Comissão examinadora abaixo assinada.

Prof. Dr. Álvaro Veiga
Orientador
Departamento de Engenharia Elétrica — PUC-Rio

Prof. Dr. Marcelo C. Medeiros
Departamento de Economia - PUC-Rio

Prof. Dr. Joel Maurício Corrêa da Rosa
Departamento de Estatística - UFPR

Prof. José Eugênio Leal
Coordenador Setorial do Centro Técnico Científico — PUC-Rio

Rio de Janeiro, 11 de Abril de 2008

Rodrigo Pinto Moreira

Graduou-se em Ciência Estatísticas na Escola Nacional de Ciências Estatísticas - ENCE (Rio de Janeiro, Brasil). Durante o mestrado em Engenharia Elétrica trabalhou com técnicas de análise estatítica multivariada, geoestatística, modelagem linear e não-linear e em projetos na área de seguros colaborando com seu orientador no desenvolvimento de modelos internos para seguradoras.

Ficha Catalográfica

Moreira, Rodrigo Pinto

Modelo de Regressão Logística com Transição Suave Estruturado por Árvore (STLR-Tree) / Rodrigo Pinto Moreira; orientador: Dr. Álvaro Veiga. — Rio de Janeiro : PUC-Rio, Departamento de Engenharia Elétrica, 2008.

v., 82 f: il. ; 29,7 cm

1. Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica.

Inclui referncias bibliogrficas.

1. Engenharia Elétrica – Tese. 2. Modelos não-lineares estruturados por árvore. 3. Classificação. 4. Regressão Logística. 5. Árvores de Classificação e Regressão (CART). I. Veiga, Álvaro. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. III. Ttulo.

CDD: 6213

Agradecimentos

Ao meu orientador Álvaro Veiga Lima Filho, pelo apoio e incentivo a este trabalho.

À FAPERJ, CNPq, CAPES e à PUC-Rio, pelos auxílios financeiros concedidos, sem os quais este trabalho não poderia ter sido realizado.

Aos meus pais e minha irmã. Família que me atura e ajuda diariamente.

A todos os meus outros familiares, principalmente minhas avós e meu avô, que por muitos anos ainda me darão força.

Ao meu avô João (*in memoriam*), que certamente está me ajudando de um bom lugar.

À minha futura esposa, Suene, e à sua família. Ela foi a pessoa que mais me aturou no decorrer deste trabalho.

Aos meus queridos amigos da TDP, ENCE e todos os demais.

Aos meus companheiros da PUC-Rio, principalmente aos freqüentadores da sala L604, na favelinha.

Aos professores da ENCE, Kaizô Beltrão e Sandra Canton.

Aos professores Cristiano Fernandes, Marcelo Medeiros e Joel Corrêa da Rosa.

Ao mestrando do ICA, Gustavo Victor C. Ortega, pela ajuda com a aplicação de Redes Neurais e na obtenção dos dados para a mesma.

Ao pessoal da secretaria e do suporte do departamento de Engenharia Elétrica.

Enfim, a todos aqueles que contribuíram de forma direta ou indireta na realização deste feito.

Resumo

Moreira, Rodrigo Pinto; Veiga, Álvaro. **Modelo de Regressão Logística com Transição Suave Estruturado por Árvore (STLR-Tree)**. Rio de Janeiro, 2008. 82p. Dissertação de Mestrado — Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Este trabalho tem como objetivo principal adaptar o modelo STR-Tree, o qual é a combinação de um modelo *Smooth Transition Regression* com *Classification and Regression Tree (CART)*, a fim de utilizá-lo em Classificação. Para isto algumas alterações foram realizadas em sua forma estrutural e na estimação. Devido ao fato de estarmos fazendo classificação de variáveis dependentes binárias, se faz necessária a utilização das técnicas empregadas em Regressão Logística, dessa forma a estimação dos parâmetros da parte linear passa a ser feita por Máxima Verossimilhança. Assim o modelo, que é paramétrico não-linear e estruturado por árvore de decisão, onde cada nó terminal representa um regime os quais têm seus parâmetros estimados da mesma forma que em uma Regressão Logística, é denominado *Smooth Transition Logistic Regression Tree (STLR-Tree)*. A inclusão dos regimes, determinada pela divisão dos nós da árvore, é feita baseada em testes do tipo Multiplicadores de Lagrange, que em sua forma para o caso Gaussiano utiliza a Soma dos Quadrados dos Resíduos em suas estatísticas de teste, aqui é substituída pela Função Desvio (*Deviance*), que é equivalente para o caso dos modelos não Gaussianos, cuja distribuição da variável dependente pertença à família exponencial. Na aplicação a dados reais selecionou-se dois conjuntos das variáveis explicativas de cada uma das duas bases utilizadas, que resultaram nas melhores taxas de acerto, verificadas através de Tabelas de Classificação (Matrizes de Confusão). Esses conjuntos de variáveis foram usados com outros métodos de classificação existentes, são eles: *Generalized Additive Models (GAM), Regressão Logística, Redes Neurais, Análise Discriminante, k-Nearest Neighbor (k-NN)* e *Classification and Regression Trees (CART)*.

Palavras–chave

Modelos não-lineares estruturados por árvore. Classificação. Regressão Logística. Árvores de Classificação e Regressão (CART).

Abstract

Moreira, Rodrigo Pinto; Veiga, Álvaro. **Smooth Transition Logistic Regression Model Tree**. Rio de Janeiro, 2008. 82p. MsC Thesis — Departament of Electric Engineering, Pontifícia Universidade Católica do Rio de Janeiro.

The main goal of this work is to adapt the STR-Tree model, which is the combination of a *Smooth Transition with Regression* model with *Classification and Regression Tree (CART)*, in order to use it in Classification. Some changes were made in its structural form and in the estimation. Due to the fact we are doing binary dependent variables classification, is necessary to use the techniques employed in Logistic Regression, so the estimation of the linear part will be made by Maximum Likelihood. Thus the model, which is nonlinear parametric and structured by a decision tree, where each terminal node represents a regime that have their parameters estimated in the same way as in a Logistic Regression, is called *Smooth Transition Logistic Regression Tree (STLR-Tree)*. The inclusion of the regimes, determined by the splitting of the tree´s nodes, is based on Lagrange Multipliers tests, which for the Gaussian cases uses the Residual Sum-of-squares in their test statistic, here are replaced by the *Deviance function*, which is equivalent to the case of non-Gaussian models, that has the distribution of the dependent variable in the exponential family. After applying the model in two datasets chosen from the bibliography comparing with other methods of classification such as: *Generalized Additive Models (GAM), Logistic Regression, Neural Networks, Discriminant Analyses, k-Nearest Neighbor (k-NN)* and *Classification and Regression Trees (CART)*. It can be seen, verifying in the Classification Tables (Confusion Matrices) that STLR-Tree showed the second best result for the overall rate of correct classification in three of the four applications shown, being in all of them, behind only from GAM.

Keywords

Tree structured nonlinear models. Classification. Logistic Regression. Classifications and Regression Trees (CART).

Sumário

Lista de figuras

Lista de tabelas

Jamais considere seus estudos como uma obrigação, mas como uma oportunidade invejável para aprender a conhecer a influência libertadora da beleza do reino do espírito, para seu próprio prazer pessoal e para proveito da comunidade à qual seu futuro trabalho pertencer.

Albert Einstein, *Notas Autobiográficas Albert Einstein.*

1
Introdução

Dado um modelo STR-Tree (*Tree-Structured Smooth Transition Regression*) proposto em (7), que é composto pela fusão do algoritmo CART (*Classification and Regression Tree*), apresentado em (2), por sua praticidade e interpretabilidade; e o modelo STR (*Smooth Transition Regression*) encontrado e amplamente debatido em (10), que garante a utilização dos métodos da Inferência Estatística clássica, permitindo-nos lançar mão de testes de hipóteses, intervalos de confiança e garantindo a significância estatística dos parâmetros.

Contudo, no trabalho citado inicialmente, o STR-Tree é utilizado para fazer regressão e não, como será proposto neste estudo, para classificação, sendo confrontado com alguns dos usuais métodos existentes como: *Regressão Logística, Análise Discriminante, CART, k-Vizinhos e GAM.*

O processo de Classificação nada mais é do que, por meio do conjunto de dados previamente classificados, gerar classificadores que descrevam ou distingam classes de dados ou conceitos mediante um rótulo, que não é mais do que um valor de um atributo. Em posse do classificador, poderemos testar seu poder classificatório através de dados que saibamos previamente a qual classe estes pertencem e, com isso possamos confrontar com o resultado obtido por aquele. Após tais testes, podemos utilizar o método para prever qual será a classificação de outros dados dos quais a classe é desconhecida.

Iremos propor uma adaptação feita ao modelo STR-Tree, inserindo o mesmo em um contexto de Regressão Logística para a aplicação em dados reais. Levando em consideração que algumas mudanças na estrutura e na forma de estimação foram feitas, passaremos a chamá-lo de STLR-Tree (*Smooth Transition Logistic Regression-Tree*). As mudanças mencionadas são: não considerar mais a possibilidade de que as variáveis sejam correlacionadas no tempo, como no caso das séries temporais, não admitindo a utilização de variáveis defasadas dentro do conjunto de variáveis explicativas tão pouco dentro das variáveis de transição. Além disso, a estimação dos parâmetros lineares não será mais feita através do usual método de Mínimos Quadrados Não Lineares (MQNL), dado o tipo de variáveis dependentes em que se aplica a Regressão Logística, variáveis categóricas. Assim será desenvolvida a função de

Máxima Verossimilhança para o modelo adaptado, a fim de se fazer a estimação dos parâmetros. A parte não-linear é estimada por *grid* e, os testes para a inclusão de novos regimes não são mais aqueles utilizados com a estatística F, onde medimos a soma dos quadrados dos resíduos (SQR) dos modelos testados, pois em Regressão Logística não se tem uma estrutura sistemática de resíduos e os testes, que utilizamos para a divisão dos nós, são os testes de Razão de Verossimilhança.

A capacidade de classificação do modelo STLR-Tree será comparada com outros métodos tais como: *Generalized Additive Models (GAM), Regressão Logística, Redes Neurais, Análise Discriminante, k-Nearest Neighbor (k-NN)* e *Classification and Regression Trees (CART).*

2
Regressão Logística

Este capítulo foi dedicado à Regressão Logística, pois se trata de um método base para o entendimento do modelo proposto. Porém antes da apresentação dos conceitos daquela, faz-se necessário uma introdução aos Modelos Lineares Generalizados para um bom entendimento deste capítulo.

2.1
Revisão de Modelos Lineares Generalizados (MLG)

Os Modelos Lineares Generalizados foram propostos em (24), com a finalidade de permitir a modelagem, não apenas utilizando os modelos lineares clássicos, os quais assumem, dentre outras coisas, que a variável dependente (Y_i) segue uma distribuição Normal (ou Gaussiana). Assim os MLG´s admitem que Y_i possa seguir outras distribuições pertencentes à família exponencial. No mesmo trabalho é introduzido o conceito de *Deviance*, que é uma medida utilizada para comparar os modelos.

2.1.1
Componentes de um MLG

Assim como nos modelos lineares clássicos, o objetivo dos modelos lineares generalizados é descrever a relação entre y_i, que são as realizações da variável aleatória Y_i, e outras variáveis chamadas regressores (também conhecidas como variáveis explicativas, preditores ou covariáveis). A realização de uma variável explicativa, X_i, será representada por x_i e, é descrita por meio de um conjunto de parâmetros representado por $\beta_1, ..., \beta_p$, que ponderam a combinação linear dos valores de X_i, bem como a um erro aleatório (ou perturbação) ϵ_i, consegue descrever o comportamento da variável dependente através da seguinte expressão

$$y_i = \sum_{j=1}^{p} x_{ji}\beta_j + \epsilon_i, \quad i = 1, ..., n \tag{2-1}$$

ou ainda $y_i = \mathbb{E}(Y_i|\mathbf{x}_i) + \epsilon_i$, onde a principal suposição sob o erro no caso de modelos lineares é que o mesmo segue a distribuição Normal com média zero e variância constante e, por conseqüência, a distribuição da variável dependente condicional as variáveis explicativas será Normal com média $\mathbb{E}(Y_i|\mathbf{x}_i)$ e variância constante.

Descrevendo tal modelo na forma matricial teremos

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \beta_1 \begin{pmatrix} x_{11}=1 \\ x_{12}=1 \\ \vdots \\ x_{1n}=1 \end{pmatrix} + \beta_2 \begin{pmatrix} x_{21} \\ x_{22} \\ \vdots \\ x_{2n} \end{pmatrix} + ... + \beta_p \begin{pmatrix} x_{p1} \\ x_{p2} \\ \vdots \\ x_{pn} \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_{21} & ... & x_{p1} \\ 1 & x_{22} & ... & x_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{2n} & ... & x_{pn} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

que pode ser expressa simplesmente por

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

em que $\mathbf{y}$ e $\boldsymbol{\epsilon}$ são vetores $n \times 1$, $\mathbf{X}$ uma matriz $n \times p$ e $\boldsymbol{\beta}$ um vetor $p \times 1$.

Um elemento do vetor $\mathbf{y}$ é dado pela expressão

$$y_i = \mathbf{x}_i\beta + \epsilon_i, \quad i = 1, ..., n$$

que corresponde a forma matricial de (2-1), onde $\mathbf{x}_i = (1, x_{1i}, x_{2i}, ..., x_{pi})'$.

A fim de identificar os componentes de um MLG iremos assumir, neste primeiro momento, que $Y_1, ..., Y_n$ são variáveis independentes e normalmente distribuídas, as quais, também por suposição, são independentes, com distribuição, não necessariamente, mas usualmente, Normal e têm variância constante (σ^2_ϵ), um *Ruído Branco*.

Tal variância também é um parâmetro desconhecido e, desta maneira, além dos p parâmetros representados pelos β´s, teremos σ^2_ϵ totalizando $p + 1$ parâmetros.

Porém diferentemente do caso linear Gaussiano aqui $y_i = \pi(\mathbf{x}_i) + \epsilon_i$, onde ϵ_i assume apenas dois valores dependendo daquele assumido por y_i. Se $y_i = 1$ então $\epsilon_i = 1 - \pi(\mathbf{x}_i)$ com probabilidade $\pi(\mathbf{x}_i)$. Caso $y_i = 0$, $\epsilon_i = -\pi(\mathbf{x}_i)$ com

probabilidade $1 - \pi(\mathbf{x}_i)$.

Com isso reescrevemos o modelo como

$$\mathbb{E}[Y_i|\mathbf{x}_i] = \mu_i = \sum_{j=1}^{p} x_{ji}\beta_j, \quad i = 1, ..., n \tag{2-2}$$

Expressão que, na forma matricial será tal que

$$\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$$

Após a estimação dos β´s podemos encontrar os valores de $\hat{\mu_1}, ..., \hat{\mu_n}$ escrevendo assim o modelo estimado da seguinte maneira

$$\hat{\mu_i} = \sum_{j=1}^{p} x_{ji}\hat{\beta}_j, \quad i = 1, ..., n$$

O modelo pode ser dividido em três partes:

– Componente aleatória: componente da variável aleatória Y_i, $i = 1, ...n$, admitindo que a mesma tenha distribuição pertencente à *família exponencial*;

– Preditor linear: representado por η e denominado por

$$\eta_i = \sum_{j=1}^{p} x_{ji}\beta_j, \quad i = 1, ..., n \tag{2-3}$$

– Função de ligação: função $g(\cdot)$ monotônica diferenciável que liga o preditor linear ao valor esperado de Y_i, ou seja, $g(\mu_i) = \eta_i$, $i = 1, ...n$

2.1.2 Ligações Canônicas

No modelo linear clássico, a função que ligava o preditor linear ao valor esperado de Y_i era a identidade, pois sendo aquela uma variável aleatória com distribuição normal, a média e o preditor linear são idênticos. Em se tratando de variáveis dependentes que tenham uma distribuição pertencente à família exponencial, porém diferente da Normal, temos disponíveis outras funções de ligação clássicas como, por exemplo, para o caso de uma distribuição binomial, em que $\mu \in (0, 1)$

1. Logito: $g(\mu_i) = \log\left(\frac{\mu}{1-\mu}\right)$
2. Probito: $g(\mu_i) = \Phi^{-1}(\mu)$
   onde $\Phi(\cdot)$ é uma função de distribuição acumulada Normal padrão
3. Complemento Log-log: $g(\mu_i) = \log[-\log(1-\mu)]$

Temos também o caso clássico para contagens, que seguem uma distribuição de Poisson, cuja função de ligação é a logarítmica, $g(\mu_i) = \log(\mu)$. Além das distribuições citadas anteriormente, também fazem parte da família exponencial a distribuição gamma e a binomial negativa.

Utilizaremos algumas dessas funções na próxima seção, onde abordaremos a *Regressão Logística*, a qual em seu caso particular mais simples tem uma variável dependente dicotômica e possui uma distribuição binomial.

2.2 Dados binários (Regressão Logística)

Como mencionado, o caso mais simples de uma Regressão Logística ocorre quando a variável aleatória Y_i assume apenas dois valores, 0 ou 1. O primeiro é a ocorrência de um determinado evento *fracasso* e o segundo *sucesso*. Para isso, teremos que definir a probabilidade de interesse, ou *probabilidade de sucesso*, $\mathbb{P}(Y_i = 1) = \pi_i$ e a *probabilidade de fracasso* $\mathbb{P}(Y_i = 0) = 1 - \pi_i$.

Para investigar a relação entre a probabilidade de sucesso π_i e o vetor de covariáveis $\mathbf{X} = (X_1, X_2, ..., X_p)'$ escrevemos o modelo

$$\mathbb{E}(Y_i|\mathbf{x}_i) = \pi_i = \sum_{j=1}^{p} x_{ji}\beta_j, \quad i = 1, ..., n$$

Entretanto, como $0 < \pi < 1$, dificilmente a igualdade acima é verdadeira. Desta maneira usaremos uma transformação $g(\pi)$ para, corretamente, poder escrever o modelo. Nosso próximo passo é escolher a transformação, que será chamada função de ligação e assim formalizar a relação como segue

$$g(\pi_i) = \eta_i$$

$$g(\pi_i) = \sum_{j=1}^{p} x_{ji}\beta_j, \quad i = 1, ..., n$$

Optamos pela logito (ou função logística) por ser a ligação canônica.

$$\log\left[\frac{\mathbb{P}(Y_i=1|\mathbf{x}_i)}{\mathbb{P}(Y_i=0|\mathbf{x}_i)}\right]=\log\left(\frac{\pi_i}{1-\pi_i}\right)=\sum_{j=1}^{p}x_{ji}\beta_j,\quad i=1,...,n \tag{2-4}$$

A probabilidade π_i pode ser escrita em função do preditor linear conforme:

$$\frac{\pi_i}{1-\pi_i}=e^{\sum_{j=1}^{p}x_{ji}\beta_j},\quad i=1,...,n \tag{2-5}$$

$$\pi_i=\frac{e^{\eta_i}}{1+e^{\eta_i}} \tag{2-6}$$

2.2.1 Especificação do modelo de Regressão Logística

Antes de descrever os métodos de especificação do modelo de Regressão Logística, será deduzida a expressão da função de *Máxima Verossimilhança* e introduzido o conceito de *Deviance* (ou *Função Desvio*).

Se olharmos apenas para o caso em que π é um escalar temos a função de *Máxima Verossimilhança* para $y_1,...,y_n$, realizações da distribuição *Bernoulli* (π), dada por:

$$L(\boldsymbol{\beta})=\prod_{i=1}^{n}\pi^{y_i}(1-\pi)^{1-y_i}=\pi^{\sum_{i=1}^{n}y_i}(1-\pi)^{n-\sum_{i=1}^{n}y_i},\quad 0\leq\pi\leq 1.$$

A expressão do $\log L(\boldsymbol{\beta})$, também chamada *log-verossimilhança* é

$$l(\boldsymbol{\beta})=\sum_{i=1}^{n}y_i\log(\pi)+\left(n-\sum_{i=1}^{n}y_i\right)\log(1-\pi),\ 0\leq\pi\leq 1.$$

Na Regressão Logística π é função de outras covariáveis, $x_1,...,x_n$, assim temos:

$$L(\boldsymbol{\beta})=\prod_{i=1}^{n}\pi(\mathbf{x}_i)^{y_i}[1-\pi(\mathbf{x}_i)]^{1-y_i}=\pi(\mathbf{x}_i)^{\sum_{i=1}^{n}y_i}[1-\pi(\mathbf{x}_i)]^{n-\sum_{i=1}^{n}y_i} \tag{2-7}$$

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} y_i \log[\pi(\mathbf{x}_i)] + \left(n - \sum_{i=1}^{n} y_i\right) \log[1 - \pi(\mathbf{x}_i))] \tag{2-8}$$

$$\begin{aligned} l(\boldsymbol{\beta}) = & \quad \sum_{i=1}^{n} y_i \log[\pi(\mathbf{x}_i)] + \left(n - \sum_{i=1}^{n} y_i\right) \log[1 - \pi(\mathbf{x}_i)] \\ = & \quad \sum_{i=1}^{n} y_i \log[\pi(\mathbf{x}_i)] + n \log[1 - \pi(\mathbf{x}_i)] - \sum_{i=1}^{n} y_i \log[1 - \pi(\mathbf{x}_i)] \end{aligned}$$

$$l(\boldsymbol{\beta}) = \left\{ \sum_{i=1}^{n} y_i \log \left[\frac{\pi(\mathbf{x}_i)}{1 - \pi(\mathbf{x}_i)} \right] + n \log[1 - \pi(\mathbf{x}_i)] \right\}. \tag{2-9}$$

Podemos notar em (2-9) o aparecimento da função logito, $\log \left[\frac{\pi(\mathbf{x}_i)}{1-\pi(\mathbf{x}_i)} \right]$, que é a função de ligação entre o preditor linear e o valor esperado de Y_i.

Da qual sabemos que

$$\log \left[\frac{\pi(\mathbf{x}_i)}{1 - \pi(\mathbf{x}_i)} \right] = \beta' \mathbf{x}_i$$

e de forma análoga

$$\pi(\mathbf{x}_i) = \frac{e^{\beta' \mathbf{x}_i}}{1 + e^{\beta' \mathbf{x}_i}}$$

logo

$$1 - \pi(\mathbf{x}_i) = \frac{1}{1 + e^{\beta' \mathbf{x}_i}}$$

então

$$l(\beta) = \sum_{i=1}^{n} \left[y_i \beta' \mathbf{x}_i - \log(1 + e^{\beta' \mathbf{x}_i}) \right]. \tag{2-10}$$

A *função Desvio*, também conhecida como *Deviance* (ver (19)) é basicamente a distância entre a log-verossimilhança do modelo contendo um parâmetro para cada uma das n observações (modelo saturado) e o modelo ajustado para p parâmetros, medindo assim a qualidade do ajuste. Se o seu valor for pequeno, indica que o ajuste do modelo com p parâmetros é próximo daquele com n e, segundo o princípio da parcimônia, escolhe-se o primeiro.

No caso Binomial a *Deviance* toma a forma

$$D = -2\sum_{i=1}^{n}\left[y_i \log\left(\frac{y_i}{\hat{\pi}_i}\right) + (m_i - y_i)\log\left(\frac{m_i - \hat{\pi}_i}{m_i - y_i}\right)\right], \tag{2-11}$$

onde $0 \leq y_i \leq m_i$ e, no caso, $m_i = 1, \forall t$.

Segundo o apresentado em (15), quando tal expressão é computada para regressão linear simples é equivalente a soma dos quadrados dos resíduos. Porém, se tratando de uma Regressão Logística em que $y = 0$ ou 1, tal medida não pode ser utilizada como sinalizadora de um bom ajuste. Desenvolvendo a equação acima temos

$$\begin{aligned} D &= -2\sum_{i=1}^{n}\left[y_i \log\left(\frac{y_i}{\hat{\pi}_i}\right) + (1 - y_i)\log\left(\frac{1 - \hat{\pi}_i}{1 - y_i}\right)\right] \\ &= -2\sum_{i=1}^{n}\left[y_i \log(y_i) + (1 - y_i)\log(1 - y_i) - y_i \log\left(\frac{\hat{\pi}_i}{1 - \hat{\pi}_i}\right) - \log(1 - \hat{\pi}_i)\right] \end{aligned}$$

como y assume apenas os valores 0 e 1 temos que

$$y_i \log(y_i) = (1 - y_i)\log(1 - y_i) = 0.$$

Além disso, $\log\left(\frac{\hat{\pi}_i}{1-\hat{\pi}_i}\right) = \hat{\beta}'\mathbf{x}_i$, e desta maneira

$$\begin{aligned} D &= -2\boldsymbol{\beta}'\mathbf{X}'\mathbf{Y} - 2\sum_{i=1}^{n}\log(1 - \hat{\pi}_i) \\ &= -2\hat{\boldsymbol{\eta}}'\hat{\boldsymbol{\pi}} - 2\sum_{i=1}^{n}\log(1 - \hat{\pi}_i). \end{aligned}$$

A *Deviance* não pode ser utilizada como medida da qualidade do ajuste quando m_i é um número pequeno, geralmente para $m_i \leq 5$. Neste caso D passa a ter uma distribuição condicional degenerada, dado os valores de $\hat{\boldsymbol{\beta}}$ (ver (19)). A frente serão apresentadas as medidas da qualidade do ajuste para regressão logística. A função desvio poderá ser usada em testes de hipótese de nulidade dos parâmetros através da estatística F.

A seleção dos regressores pode ser feita utilizando-se uma metodologia proposta em (15), a qual é uma variante do método *Stepwise*, ou através dos critérios de informação, AIC (*Akaike Information Criterion*) e BIC (*Bayesian Information Criterion*).

Tais critérios penalizam a função de log-verossimilhança pela inclusão de

novas variáveis, respeitando o princípio da parcimônia. Escolhe-se o modelo que minimiza o AIC ou BIC, que estão descritos em (2-12) e (2-13)

$$AIC = -2\frac{l(\boldsymbol{\beta})}{n} + 2\frac{p}{n} \tag{2-12}$$

$$BIC = -2\frac{l(\boldsymbol{\beta})}{n} + p\frac{\log(n)}{n} \tag{2-13}$$

onde p é o número de parâmetros, n a quantidade de observações e $l(\boldsymbol{\beta})$ é o log da função de verossimilhança.

Já a metodologia proposta em (15) considerando um modelo com p variáveis explicativas segue alguns passos mostrados a seguir

1. Ajusta-se o modelo nulo somente com intercepto e $(p-1)$ modelos cada um contendo o intercepto mais uma das variáveis explicativas $\mathbf{x}_i$. Confronta-se cada um desses $(p-1)$ modelos, com o modelo nulo através da estatística de *Razão de Verossimilhança* dada por

$$\xi_{RV}^{(0)} = 2\ln\left[\frac{L(\hat{\boldsymbol{\beta}};\mathbf{y})}{L(\boldsymbol{\beta}^0;\mathbf{y})}\right] = 2\left[l(\hat{\boldsymbol{\beta}};\mathbf{y}) - l(\boldsymbol{\beta}^0;\mathbf{y})\right] \xrightarrow{a} \chi^2_{(p)}, \tag{2-14}$$

onde $l(\hat{\boldsymbol{\beta}};\mathbf{y})$ é a log verossimilhança do modelo com intercepto mais uma das variáveis explicativas e $l(\boldsymbol{\beta}^0;\mathbf{y})$ é a log verossimilhança do modelo apenas com intercepto. Tendo conhecimento do parâmetro de dispersão, no contexto de MLG denotado por ϕ, podemos expressar a razão de verossimilhança através da diferença entre as funções desvio e assim

$$\xi_{RV}^{(0)} = \phi\left[D(\mathbf{y};\hat{\boldsymbol{\mu}}^0) - D(\mathbf{y};\hat{\boldsymbol{\mu}})\right] \xrightarrow{a} \chi^2_{(p)}, \tag{2-15}$$

em que $\hat{\boldsymbol{\mu}}^0 = \mathbf{g}^{-1}(\hat{\boldsymbol{\eta}}^0)$, $\hat{\boldsymbol{\eta}}^0 = \mathbf{X}\boldsymbol{\beta}^0$. De maneira análoga, a estatística F, a seguir, pode ser utilizada como alternativa de teste das hipóteses, apresentando ainda a vantagem de não depender do parâmetro de dispersão, ϕ,

$$F = \frac{\left[D(\mathbf{y};\hat{\boldsymbol{\mu}}^0) - D(\mathbf{y};\hat{\boldsymbol{\mu}})\right]/q}{D(\mathbf{y};\hat{\boldsymbol{\mu}})/(N-p)} \xrightarrow{a} F_{q,(N-p)}. \tag{2-16}$$

Escolhe-se o modelo com o menor *p-valor*, ou seja, sendo $p_{e_i}^{(0)} = P\left(\chi^2_{(\nu)} > \xi_{RV}^{(0)}\right)$ o *p-valor* associado a $\mathbf{x}_i$, $\forall t$, onde $\nu = 1$ se $\mathbf{x}_i$ é contínua e $\nu = k-1$ se $\mathbf{x}_i$ é categórica com $k-1$ níveis. Assim, o modelo escolhido

é aquele que apresenta $p_{e_1} = \min\left[p_{e_i}^{(0)}\right]$, e a variável escolhida é denominada $\mathbf{x}_{e_1}$. Além disso, é determinada uma probabilidade de entrada, P_E, a partir da qual verifica-se a significância da variável escolhida, onde a seqüência da modelagem se dá caso $p_{e_1} < P_E$, quando se prossegue para o passo seguinte, e caso o contrário aconteça, o modelo especificado é escolhido. Geralmente $0,15 < P_E < 0,25$ (ver (15)).

2. Ajustam-se agora $(p-2)$ modelos incluindo mais uma variável explicativa, das que restaram em relação ao modelo que foi selecionado no passo anterior. Cada um desses modelos é avaliado em relação ao modelo do passo 1 como se segue

$$\xi_{RV_i}^{(1)} = 2\left[l_{e_1e_i}(\hat{\boldsymbol{\beta}};\mathbf{y}) - l_{e_1}(\hat{\boldsymbol{\beta}};\mathbf{y})\right], \quad i = 2,3,...,p.$$

A escolha da variável $\mathbf{x}_{e_2}$ se dá para $p_{e_2} = \min\left[p_{e_i}^{(1)}\right]$, onde $p_{e_i}^{(1)} = P\left(\chi^2_{(\nu)} > \xi_{RV_i}^{(1)}\right)$. Se $p_{e_2} < P_E$ siga para o passo 3, caso contrário pare.

3. Com o modelo formado pelo intercepto mais $\mathbf{x}_{e_1}$ e $\mathbf{x}_{e_2}$ deve-se testar se ao incluir esta última, a variável $\mathbf{x}_{e_1}$ deixa de ser significativa. Desta forma, calcula-se

$$\xi_{RV_i}^{(2)} = 2\left[l_{e_1e_2}(\hat{\boldsymbol{\beta}};\mathbf{y}) - l_{e_i}(\hat{\boldsymbol{\beta}};\mathbf{y})\right], \quad i = 1,2$$

$$p_{e_i}^{(2)} = P\left(\chi^2_{(\nu)} > \xi_{RV_i}^{(2)}\right).$$

Nesta situação, observa-se a variável que possui o maior *p-valor*, a fim de testar se ela irá ser retirada ou não do modelo. A variável escolhida é denominada $\mathbf{x}_{r_2}$. Além disso bem como a probabilidade de entrada é determinada uma probabilidade da variável ser retirada, P_R, onde $0,15 < P_R < 0,25$. Se $p_{r_2} > P_R$ então a variável é retirada, caso contrário a variável permanece e deve-se verificar a entrada de outra variável no modelo escolhido. Ainda neste passo ajusta-se $(p-3)$ modelos (supondo que $\mathbf{x}_{e_1}$ e $\mathbf{x}_{e_2}$ tenham permanecido) e calcula-se

$$\xi_{RV_i}^{(2^*)} = 2\left[l_{e_1e_2e_i}(\hat{\boldsymbol{\beta}};\mathbf{y}) - l_{e_1e_2}(\hat{\boldsymbol{\beta}};\mathbf{y})\right], \quad i = 3,4,...,p,$$

$$p_{e_i}^{(2^*)} = P\left(\chi^2_{(\nu)} > \xi_{RV_i}^{(2^*)}\right),$$

$$p_{e_3} = \min\left[p_{e_i}^{(2^*)}\right].$$

Se $p_{e_3^*} < P_E$ siga para o próximo passo, caso contrário pare.

Os próximos passos são semelhantes ao passo 3 até que sejam esgotadas as possibilidades de entrada e retirada de variáveis. Com o modelo, após a escolha das variáveis principais (ou efeitos principais), é verificada a significância de cada coeficiente (β) individualmente, através de testes de *Wald*[1], onde $H_0 : \beta = 0$, e aqueles que não forem estatisticamente significativos, ou seja, quando a hipótese nula não é rejeitada, determina-se a retirada de sua respectiva covariável do modelo.

Feito isto, os passos seguintes consistem na inclusão das interações de primeira ordem seguindo-se o mesmo procedimento de entrada e retirada feito anteriormente sem se esquecer de não eliminar os efeitos principais. Se for necessário incluir as interações de segunda e terceira ordem segue-se o mesmo padrão.

2.2.2 Estimação do modelo de Regressão Logística

A estimação dos parâmetros de uma Regressão Logística é feita por Máxima Verossimilhança, utilizando-se o método iterativo de *Newton-Raphson*.

Os cálculos das derivadas de $l(\boldsymbol{\beta})$ (log da verossimilhança) bem como o algoritmo do método são apresentados a seguir.

Derivando (2-16) em relação ao parâmetro β (*Função Escore*) teremos:

$$\begin{aligned}\frac{\partial l(\boldsymbol{\beta})}{\partial \beta} &= \sum_{i=1}^{n} \mathbf{x}_i \left[y_i - \frac{e^{\beta' \mathbf{x}_i}}{(1 + e^{\beta' \mathbf{x}_i})}\right] \\ &= \sum_{i=1}^{n} \mathbf{x}_i \left[y_i - \pi(\mathbf{x}_i)\right].\end{aligned}$$

Na forma matricial,

$$\frac{\partial l(\boldsymbol{\beta})}{\partial \beta} = \mathbf{X}'(\mathbf{y} - \boldsymbol{\pi}). \tag{2-17}$$

O algoritmo também requer a Hessiana, obtida por

[1] A estatística de teste é dada por: $W = \frac{\hat{\beta}}{\sqrt{Var(\hat{\beta})}}$

$$\begin{aligned}
\frac{\partial^2 l(\boldsymbol{\beta})}{\partial\beta\partial\beta'} &= -\sum_{i=1}^{n}\left[\frac{(1+e^{\beta'\mathbf{x}_i})e^{\beta'\mathbf{x}_i}\mathbf{x}_i\mathbf{x}_i' - e^{\beta'\mathbf{x}_i}\mathbf{x}_i' e^{\beta'\mathbf{x}_i}\mathbf{x}_i'}{(1+e^{\beta'\mathbf{x_i}})^2}\right] \\
&= -\sum_{i=1}^{n}\left[\frac{(1+e^{\beta'\mathbf{x}_i})e^{\beta'\mathbf{x}_i}\mathbf{x}_i\mathbf{x}_i' - (e^{\beta'\mathbf{x}_i})^2\mathbf{x}_i\mathbf{x}_i'}{(1+e^{\beta'\mathbf{x_i}})^2}\right] \\
&= -\sum_{i=1}^{n}\left[\frac{e^{\beta'\mathbf{x}_i}\mathbf{x}_i\mathbf{x}_i'}{1+e^{\beta'\mathbf{x}_i}} - \left(\frac{e^{\beta'\mathbf{x}_i}}{1+e^{\beta'\mathbf{x}_i}}\right)^2\mathbf{x}_i\mathbf{x}_i'\right] \\
&= -\sum_{i=1}^{n}\left[\pi(\mathbf{x}_i)\mathbf{x}_i\mathbf{x}_i' - \pi(\mathbf{x}_i)^2\mathbf{x}_i\mathbf{x}_i'\right] \\
&= -\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i'\pi(\mathbf{x}_i)[1-\pi(\mathbf{x}_i)].
\end{aligned}$$

E matricialmente representada por

$$\frac{\partial^2 l(\boldsymbol{\beta})}{\partial\beta\partial\beta'} = -\mathbf{X}'\mathbf{W}\mathbf{X}. \tag{2-18}$$

A *Matriz de Informação de Fisher* para β é conhecida pela expressão $I(\beta) = -E\left[\frac{\partial^2 l(\boldsymbol{\beta})}{\partial\beta\partial\beta'}\right]$.

Tendo esses elementos falta apenas determinar um valor inicial para β, chamaremos de $\beta^{(m)}$, e assim dar início ao algoritmo de Newton-Raphson, que a fim de obter a estimativa de Máxima Verossimilhança do parâmetro em questão, β, expande-se a Função Escore, $U(\beta)$ em torno do valor inicial, um $\beta^{(m)}$ qualquer, de maneira que

$$U(\beta) \cong U(\beta^{(m)}) + \frac{\partial}{\partial\beta'}U(\beta^{(m)})(\beta^{(m+1)} - \beta^{(m)}), \quad m = 0, 1, ...$$

Iterativamente obtem-se

$$\beta^{(m+1)} = \beta^{(m)} + \left[-\frac{\partial}{\partial\beta'}U(\beta^{(m)})\right]^{-1}U(\beta^{(m)}), \quad m = 0, 1, ...$$

A matriz $-\frac{\partial}{\partial\beta'}U(\beta^{(m)})$ deve ser positiva definida, e como não se pode garantir tal hipótese, substitui-se a mesma pelo seu valor esperado $E\left[-\frac{\partial}{\partial\beta'}U(\beta^{(m)})\right]^{-1} = I^{-1}(\beta^{(m)})$ e assim, continuando o processo iterativo

$$\beta^{(m+1)} = \beta^{(m)} + I^{-1}(\beta^{(m)})U(\beta^{(m)}), \quad m = 0, 1, ...$$

$$\beta^{(m+1)} = \beta^{(m)} - \left(\frac{\partial^2 l(\beta)}{\partial\beta\partial\beta'}\right)^{-1} \frac{\partial l(\beta)}{\partial\beta}, \quad m = 0, 1, ... \tag{2-19}$$

Trabalhando mais uma vez com a forma matricial em que $\mathbf{y}$ é o vetor $(n \times 1)$ de valores y_i, $\mathbf{X}$ a matriz $(n \times (p+1))$ de valores x_i, $\boldsymbol{\pi}$ o vetor $(n \times 1)$ das probabilidades ajustadas com o *i-ésimo* elemento igual a $\pi(x_i; \beta^{(m)})$ e $\mathbf{W}$ a matriz diagonal $(n \times n)$ de pesos com o *i-ésimo* elemento igual a $\pi(x_i; \beta^{(m)})(1 - \pi(x_i; \beta^{(m)}))$ tem-se

$$\begin{aligned} \beta^{(m+1)} = & \quad \beta^{(m)} + (\mathbf{X'WX})^{-1}\mathbf{X'}(\mathbf{y} - \boldsymbol{\pi}) \\ = & \quad (\mathbf{X'WX})^{-1}\mathbf{X'W}[\mathbf{X}\beta^{(m)} + \mathbf{W^{-1}}(\mathbf{y} - \boldsymbol{\pi})] \end{aligned}$$

$$\beta^{(m+1)} = (\mathbf{X'WX})^{-1}\mathbf{X'Wz} \tag{2-20}$$

onde $\mathbf{z} = \mathbf{X}\beta^{(m)} + \mathbf{W^{-1}}(\mathbf{y} - \boldsymbol{\pi})$.

Considerando $\mathbf{z}$ como se fosse o vetor de observações de uma variável dependente qualquer, também chamada de variável dependente ajustada, a equação $(\mathbf{X'WX})^{-1}\mathbf{X'Wz}$ seria o cálculo do estimador de *Mínimos Quadrados Ponderados* (ver (33)). A implementação do método pode ser feita considerando $z_i = \beta'\mathbf{x}_i + \frac{y_i - \pi(\mathbf{x}_i)}{\pi(\mathbf{x}_i)[1-\pi(\mathbf{x}_i)]}$, $w_i = \pi(\mathbf{x}_i)[1 - \pi(\mathbf{x}_i)]$ e, conseqüentemente, $z_i = \beta'\mathbf{x}_i + \frac{y_i - \pi(\mathbf{x}_i)}{w_i}$, usando regressão linear ponderada para explicar z_i por $\mathbf{x}_i$. Este procedimento é conhecido como *Iteratively Reweighted Least Squares (IRLS)* e matricialmente é descrito por

$$\mathbf{z} = \mathbf{X}\beta^{(m)} + \mathbf{W^{-1}}(\mathbf{y} - \boldsymbol{\pi})$$

$$\beta^{(m+1)} \longleftarrow \arg\min_{\beta}(\mathbf{z} - \mathbf{X}\beta)^{-1}\mathbf{W}(\mathbf{z} - \mathbf{X}\beta), \quad m = 0, 1, ...$$

O valor inicial usual é $\beta = 0$.

Fazemos a regressão de z_0 em relação as covariáveis $x_1, ..., x_p$ com peso W_0 e assim achamos a estimativa de $\hat{\boldsymbol{\beta}}^{(1)}$, com isto alimento o modelo e faço o cálculo para z_1, acho $\hat{\boldsymbol{\beta}}^{(2)}$ e assim sucessivamente. O procedimento converge em um número finito de passos, podendo falhar apenas se uma ou mais componentes de $\hat{\boldsymbol{\beta}}$ forem infinitas, o que implica em algumas probabilidades ajustadas serem iguais a zero ou um. Caso isso ocorra, poderemos identificar as convergências anormais através da análise de *Deviance*, dado que as probabil-

idades ajustadas serão alteradas; outra forma é verificar a mudança no $\hat{\boldsymbol{\beta}}$ ou no preditor linear, $\hat{\boldsymbol{\eta}}$.

2.2.3 Avaliação do ajuste (Medidas de aderência)

Avaliamos os modelos através de alguns métodos, tais como: Tabela de Classificação (ou Tabela de Previsão ou ainda Matriz de Confusão), Área abaixo da Curva ROC (*Receiver Operating Characteristic*), χ^2 de Pearson e o Teste de Hosmer-Lemeshow. Os dois últimos não serão abordados neste trabalho, mas encontram-se vastamente explicados em (15), onde podem ser vistos de forma bastante aplicada.

– Tabela de Classificação: Para esta análise deve-se estipular um ponto de corte, c^*, geralmente usa-se o valor 0,5. Este será comparado aos valores estimados do modelo de regressão logística, $\hat{\pi}(\mathbf{x}_i)$, e desta forma obtém-se os valores de $\hat{y}_i$ da seguinte maneira:

$$\text{se } \hat{\pi}(\mathbf{x}_i) > c^* \text{ então } \hat{y}_i = 1$$
$$\text{se } \hat{\pi}(\mathbf{x}_i) \leq c^* \text{ então } \hat{y}_i = 0.$$

Feito isto, se monta uma tabela de contingência cruzando com os valores observados de y_i com os valores encontrados no procedimento anterior, $\hat{y}_i$ e verifica-se quantas classificações foram feitas corretamente.

Na Tabela 2.1 tiramos algumas medidas relevantes para avaliar o ajuste:

– Taxa de acerto total: $\left(\frac{A+D}{A+B+C+D}\right) \times 100\%$

– Taxa de acertos para 0: $\left(\frac{A}{A+B}\right) \times 100\%$ (*Especificidade*)

– Taxa de acertos para 1: $\left(\frac{D}{C+D}\right) \times 100\%$ (*Sensitividade*)

Tabela 2.1: Tabela de Classificação

	Predito (ŷ)		
Observado (y)	0	1	
0	A	B	A + B
1	C	D	C + D
	A + C	B + D	A + B + C + D

A taxa de acertos para 1, ou seja, a probabilidade de estimar o sucesso dado que o valor real observado é realmente 1, também é chamada de

Sensitividade. Da mesma forma, a taxa de acertos para 0 é conhecida como *Especificidade*.

– Área abaixo da Curva ROC: Diferentemente da Tabela de Classificação na qual a Especificidade e a Sensitividade provêm de um único ponto de corte, neste método, pode-se variar o valor do ponto de corte utilizando o maior número possível de opções, a fim de recalcular as duas medidas citadas. Após, é feito um gráfico da Sensibilidade contra (1 - Especificidade). A curva que se forma neste gráfico é chamada de Curva ROC.

  Pelo fato de se esperar que a Sensitividade e a Especificidade sejam complementares, a área abaixo da curva ROC que indica se o modelo discriminou corretamente os fracassos (zeros) e sucessos (uns) deve ser igual a 1.

  Na literatura encontra-se uma regra que descreve a área abaixo da ROC e a qualidade do ajuste ligada a ela (ver (15)) como descrito na Tabela 2.2.

Área abaixo da ROC	Discriminação
$= 0.5$	Sem discriminação
$0.7 \leq ROC < 0.8$	Aceitável
$0.8 \leq ROC < 0.9$	Excelente
≥ 0.9	Excepcional

Tabela 2.2: Qualidade do ajuste - ROC

3
Modelos e metodologias comparadas

Este capítulo tem o propósito de listar algumas das alternativas existentes na literatura que envolve classificação, e serão utilizadas neste trabalho sendo comparadas ao modelo STLR-Tree. A maioria delas está resumida e outras bem detalhadas em (14), que ilustra com muitos exemplos suas aplicações e comparações. A seção referente à Regressão Logística não foi colocada neste capítulo, pois a mesma aparece bem detalhada no capítulo 2.

3.1
Classification and Regression Trees (CART)

Uma breve revisão da estrutura em árvore, seguindo o algoritmo CART (*Classification and Regression Trees*) em (2), onde foram unificados todos os métodos de árvores de regressão e classificação existentes no período, será feita sobre sua formulação matemática, a fim de melhor entender a estrutura do STLR-Tree apresentada posteriormente.

A distinção entre as árvores de classificação e regressão é feita de acordo com o tipo de variável dependente. Quando a variável é contínua, utiliza-se árvores de regressão e no caso de variáveis categóricas, árvores de classificação. Por não fazerem suposições sobre componentes aleatórias e sobre a forma funcional do modelo, tão pouco assumirem a existência de modelos probabilísticos, tal como acontece nos modelos estatísticos de regressão e classificação, as árvores são tidas como métodos não-paramétricos para tais fins.

De fácil entendimento, as árvores particionam de forma recursiva o espaço das covariáveis, $\mathbb{X}$. Sua estrutura é simples e usualmente são representadas e ajustadas em um gráfico que cresce de um nó inicial (ou nó raiz), que é determinado como posição 0, em direção aos nós terminais (ou folhas) passando pelos nós intermediários (ou nós geradores, criadores). Cada nó gerador na posição j dá origem a dois novos nós nas posições $2j + 1$ e $2j + 2$, e assim progressivamente, até que os nós geradores não sejam mais divididos, quando passam a ser chamados de nós terminais.

Formulação Matemática

Seja $\mathbf{x}_i = (x_{1i}, x_{2i}, ..., x_{qi})' \in \mathbb{X} \subseteq \mathbb{R}^q$ o vetor que contém q variáveis explicativas (covariáveis ou preditores) para uma resposta univariada contínua, $y_i \in \mathbb{R}, \quad i = 1, ..., n$.

Suponha que a relação entre y_i e $\mathbf{x}_i$ segue o modelo de regressão

$$y_i = f(\mathbf{x}_i) + \epsilon_i$$

Seguindo (17), como foi citado e (7) um modelo de árvore de regressão com K folhas é um modelo de particionamento recursivo do espaço das covariáveis, $\mathbb{X}$, que aproxima $f(\cdot)$ por uma função geral não-linear, $H(\mathbf{x}_i; \boldsymbol{\psi})$ de $\mathbf{x}_i$ e definida pelo vetor de parâmetros $\boldsymbol{\psi} \in \mathbb{R}^r$ onde r é o número total de parâmetros do modelo.

A partição é usualmente definida por um conjunto de hiperplanos ortogonais aos eixos das variáveis explicativas, chamada de *variável de transição* (em inglês: *split variable*).

No contexto apresentado em (2), $H(\mathbf{x}_i; \boldsymbol{\psi})$ é uma função constante por partes definida por K subregiões $k_j(\boldsymbol{\theta}_j), \quad i = 1, ..., K$ de $\mathbb{K} \subset \mathbb{R}^q$. A determinação dessas subregiões é feita pelo vetor de parâmetros não-lineares $\boldsymbol{\theta}_j, \quad j = 1, ..., K$ onde

$$f(\mathbf{x}_i) \approx H(\mathbf{x}_i; \boldsymbol{\psi}) = \sum_{j=1}^{K} \beta_j I_j(\mathbf{x}_i; \boldsymbol{\theta}_j) \tag{3-1}$$

em que

$$I_j(\mathbf{x}_i; \boldsymbol{\theta}_j) = \begin{cases} 1 & , se\ \mathbf{x}_i \in k_j(\boldsymbol{\theta}_j) \\ 0 & , se\ \mathbf{x}_i \notin k_j(\boldsymbol{\theta}_j) \end{cases} ;$$

e o vetor de parâmetros é $\boldsymbol{\psi} = (\beta_1, ..., \beta_K, \boldsymbol{\theta}'_1, ..., \boldsymbol{\theta}'_K)'$.

Neste trabalho, será considerada uma regressão logística linear por partes em cada folha, que representa um novo regime, e a transição entre os regimes é feita de forma suave. Nessa linha e dentro do contexto dos modelos lineares generalizados destacam-se os trabalhos de (5) e (4) que discutem as função $H(\mathbf{x}_i; \boldsymbol{\psi})$ para uma árvore de regressão Poisson e em Regressão Logística. Já o primeiro propõe a diferença entre funções desvio para a divisão dos nós e crescimento da árvore.

Cada nó gerador tem uma variável de transição $x_{s_j i} \in \mathbf{x}_i$ associada, onde $s_j \in \mathbb{S} = \{1, 2, ..., m\}$. Temos ainda os conjuntos de índices dos nós geradores e nós terminais que estão contidos, respectivamente, nos conjuntos $\mathbb{J}$ e $\mathbb{T}$.

No exemplo mais simples que se pode apresentar de uma árvore em que temos apenas uma profundidade ($d = 1$) e $K = 2$ nós terminais, a equação que explica a relação entre y_i e $\mathbf{x}_i$ é dada por

$$y_i = \beta_1 I_0(\mathbf{x}_i; s_0, c_0) + \beta_2[1 - I_0(\mathbf{x}_i; s_0, c_0)] + \epsilon_i$$

onde

$$I_0(\mathbf{x}_i; s_0, c_0) = \begin{cases} 1 & , se\ \mathbf{x}_{s_0 i} \leq c_0 \\ 0 & , se\ \mathbf{x}_{s_0 i} > c_0 \end{cases}$$

e $s_0 \in \mathbb{S} = 1, 2, ..., m$.

Um exemplo numérico apresentado em (7) é mostrado na figura 3.1, a seguir. Logo após, na figura 3.2, apresentamos a divisão no espaço das covariáveis, $\mathbb{X} \subseteq \mathbb{R}^2$ e a tabela 3.1 com as sentenças lógicas e o correspondente valor da variável dependente estimada, $\hat{y}$.

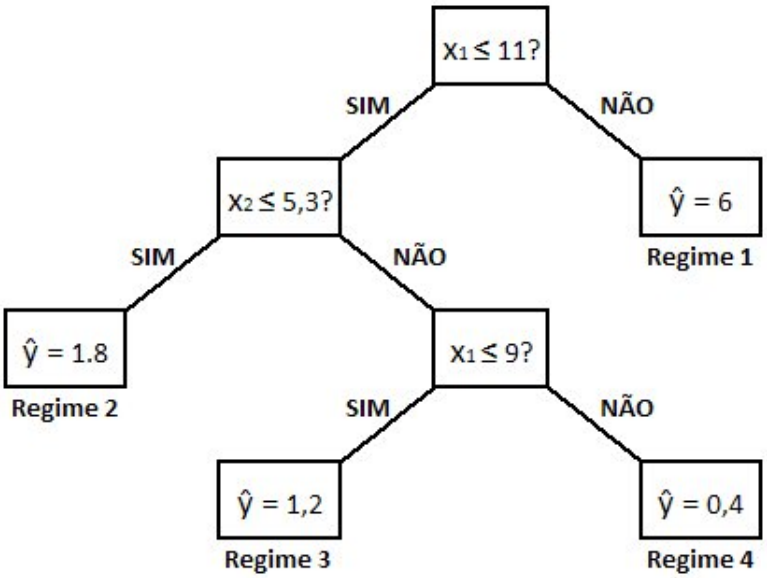

Figura 3.1: Estrutura do modelo. Exemplo em (7)

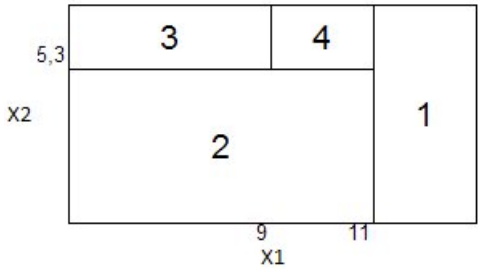

Figura 3.2: Divisão do espaço das covariáveis

Divisões de $\mathbb{X}$	$\hat{y}$
se $x_1 \geq 11$	6
se $x_1 < 11$ e $x_2 < 5.3$	1.8
se $x_1 < 9$ e $x_2 \geq 5.3$	1.2
se $9 < x_1 < 11$ e $x_2 \geq 5.3$	0.4

Tabela 3.1: Divisão do espaço das covariáveis

Algoritmo de Crescimento

Consiste na escolha de um nó a ser dividido, conseqüentemente uma variável de transição (x_{s_j}) e um limiar (c_j), e, de forma iterativa, estima-se os parâmetros dos modelos contidos em cada nó gerador. A seleção dos elementos citados e a estimação dos parâmetros são feitos simultaneamente.

Basicamente busca-se, a partir do nó raiz, x_{s_0} e c_0 que minimizam a soma dos erros quadráticos:

$$SQ^{Arv1} = \sum_{i=1}^{n} \{y_i - \beta_1 I_0(\mathbf{x}_i; s_0, c_0) - \beta_2[1 - I_0(\mathbf{x}_i; s_0, c_0)]\}^2$$

A estimação dos parâmetros β_1 e β_2 é dada por

$$\hat{\beta}_1^{MQ} = \frac{\sum_{i=1}^{n} y_i I_0(\mathbf{x}_i; s_0, c_0)}{\sum_{i=1}^{n} I_0(\mathbf{x}_i; s_0, c_0)} \tag{3-2}$$

$$\hat{\beta}_2{}^{MQ} = \frac{\sum_{i=1}^{n} y_i [1 - I_0(\mathbf{x}_i; s_0, c_0)]}{\sum_{i=1}^{n} [1 - I_0(\mathbf{x}_i; s_0, c_0)]} \tag{3-3}$$

A divisão do nó gerado na posição 1 é feita da mesma maneira, através da busca por x_{s_1} e c_1 que minimizam

$$\begin{aligned} SQ^{Arv2} \quad = \sum_{i=1}^{n} \{y_i - \beta_2[1 - I_0(\mathbf{x}_i; s_0, c_0)] - [\beta_3 I_1(\mathbf{x}_i; s_1, c_1) \\ + \beta_4[1 - I_1(\mathbf{x}_i; s_1, c_1)] I_0(\mathbf{x}_i; s_0, c_0)\}^2 \end{aligned}$$

e assim sucessivamente até que não se tenha ganhos com a divisão. Em (2) é proposto um critério de parada, declarando como nó terminal aquele que contenha 5 observações ou menos.

Além disso, a fim de diminuir a complexidade da árvore que pode crescer mais que o necessário,mesmo utilizando-se o critério de parada, existe uma técnica que determina o corte de algumas folhas e por essa razão é chamada de *Podagem*.

Uma função, sugerida em (2), e apresentada em (7), cumpre o papel de

avaliar a necessidade de se reespecificar o modelo na tentativa de melhorar seu poder de previsão é

$$R^*(N, \alpha) = \sum_{i=1}^{N} R_i + |\alpha| \, N, \tag{3-4}$$

onde R_i é uma medida da qualidade do ajuste na i-ésima folha em uma árvore com N folhas, em que α é o parâmetro que penaliza a árvore pelo seu tamanho. Como um exemplo da R_i, suponha que ela seja calculada após a divisão do nó gerador da posição 1, assim o melhor modelo é o que maximiza

$$R(Arv_2) = SQ(Arv_1) - SQ(Arv_2).$$

3.2 Generalized Additive Models (GAM)

Proposto por (12) os quais posteriormente estenderam o trabalho em (13), trata-se de modelos de regressão não paramétricos desenvolvidos após o estudo sobre *Modelos Aditivos* em (27).

A classe dos GAM´s tem como fundamento a substituição da forma linear $\sum \beta_j \mathbf{x}_j$ pela soma de funções suavizadas das variáveis explicativas, $\sum f_j(\mathbf{x}_j)$. Trata-se de uma generalização ainda maior do que os MLG´s, como se mostra abaixo, figura 3.3, na estrutura dos modelos encontrada em (11).

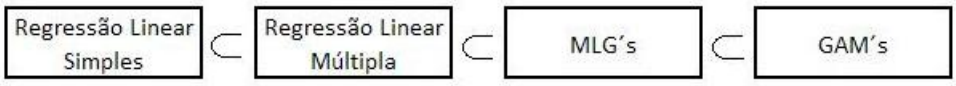

Figura 3.3: Hierarquia dos modelos

Os GAM´s são considerados modelos *semi-paramétricos*, pois, assim como os MLG´s, são paramétricos no que diz respeito a distribuição de probabilidade da variável dependente, a qual deve ser especificada, porém alguns preditores podem ser modelados de forma não-paramétrica através de termos lineares e polinomiais de outros preditores, podendo, desta maneira, mensurar relações não-lineares entre a variável dependente e as variáveis explicativas. Essa é, sem dúvida, sua maior vantagem.

Assim como os MLG´s a relação entre a média da variável dependente e, no caso dos GAM´s, as função suavizadas das variáveis explicativas é feita por uma função de ligação tendo como principal hipótese que aquelas sejam funções aditivas entre as covariáveis e as componentes sejam suaves. Desta

maneira, a forma de um GAM é apresentada da seguinte maneira

$$\mathbb{E}(\mathbf{y}|x_1, ..., x_p) = f_0 + f_1(x_1) + f_2(x_2) + ... + f_p(x_p) \tag{3-5}$$

onde o preditor linear dos MLG´s, $\eta = \sum \beta_j \mathbf{x}_j$, é substituído por $\eta = \sum f_j(\mathbf{x}_j)$, $j = 1, ..., p$.

Cada uma dessas funções é ajustada através de um diagrama de dispersão suavizado (*scatterplot smoother*) e utilizando um algoritmo se realiza a estimação das p funções simultaneamente. Conforme apresentado em (13), um diagrama de dispersão suavizado é uma função s de $\mathbf{x}$ e $\mathbf{y}$, com mesmo domínio que os valores em $\mathbf{x} : s = \mathbf{S}(\mathbf{y}|\mathbf{x})$, a qual tem como principais atributos a descrição visual da relação entre a variável dependente e as covariáveis, além da estimação da relação entre as mesmas, que nada mais é que o ajuste da reta suavizada, $f(x)$, que sintetize a dependência entre $\mathbf{y}$ e $\mathbf{x}$. Tal reta deve ser tal que minimize $\sum_{i=1}^{n}[y_i - f(x_i)]^2$.

Um suavizador usual e que será utilizado nas aplicações feitas nesse trabalho é o *Cubic Smoother Splines*, que faz a busca pela $f(x)$ que minimize a *Soma dos Quadrados dos Resíduos Penalizada - SQRP* (em inglês, *Penalized Residual Sum of Squares - PRSS*), denotada por

$$SQRP(f, \lambda) = \sum_{i=1}^{n}[y_i - f(x_i)]^2 + \lambda \int_a^b f''(t)^2 dt, \quad a \leq x_1 \leq ... \leq b \tag{3-6}$$

onde λ é o parâmetro de suavização que deve ser escolhido.

Quanto maior for λ ($\lambda \to \infty$) o termo que penaliza a SQRP é dominante, forçando $\int_a^b f''(t)^2 dt = 0$, sendo assim a reta ajustada por Mínimos Quadrados. Caso contrário, $\lambda \to 0$, a solução tende para qualquer função que faça a interpolação dos dados. Alguns métodos de seleção automática de λ são apresentados em (14).

Uma maneira intuitiva de escolher λ é através da determinação dos graus de liberdade (gl) para o suavizador, no caso o Cubic Smoothing Splines, e utilizar uma otimização numérica para determinar o valor do parâmetro que retorne tal número.

Os graus de liberdade de um suavizador são dados por

$$gl_\lambda = trace(\mathbf{S}_\lambda) \tag{3-7}$$

onde $\mathbf{S}_\lambda$ é um operador linear suavizado e a soma de seus autovalores definem os graus de liberdade. Por exemplo, quando usamos um suavizador com 4 gl, significa que para cada x_j o parâmetro λ_j é escolhido tal que

$trace[\mathbf{S}_j(\lambda_j)] - 1 = 4$.

3.2.1 Regressão Logística Aditiva

Com a substituição dos termos lineares da regressão logística pelas funções suavizadas, a expressão do modelo toma a forma

$$\log\left[\frac{\pi(\mathbf{x})}{1-\pi(\mathbf{x})}\right] = f_0 + f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2) + ... + f_p(\mathbf{x}_p). \tag{3-8}$$

As função de ligação utilizada, $g[\pi(x)]$, é a logito. Além dela os GAM´s também admitem as demais funções de ligação: probito, logística e, obviamente, a identidade.

Para especificar o modelo utiliza-se os critérios de forma semelhante àquela feita para os MLG´s na seção 2.2.1. Apenas a estimação é feita de forma diferente.

Estimação do modelo de Regressão Logística Aditiva

Para estimar uma Regressão Logística Aditiva sabendo que, dada a forma do modelo apresentada anteriormente, temos para apenas uma variável dependente, X, o modelo

$$\log\left[\frac{\mathbb{P}(Y=1|X=x)}{\mathbb{P}(Y=0|X=x)}\right] = f(x)$$

em que

$$\mathbb{P}(Y=1|X=x) = \frac{e^{f(x)}}{1+e^{f(x)}}.$$

Para tal, o método da *Máxima Verossimilhança Penalizada*, apresentado em detalhes em (14), é alocado. Tal método segue o mesmo princípio apresentado anteriormente onde se deve maximizar a log-verossimilhança, guardadas as devidas alterações com a inclusão do termo penalizador como segue

$$\begin{aligned} l(f;\lambda) &= \sum_{i=1}^{n}[y_i \log \pi(x_i) + (1-y_i)\log(1-\pi(x_i))] - \frac{1}{2}\lambda\int[f''(t)]^2dt \\ &= \sum_{i=1}^{n}[y_i f(x_i) - \log(1+e^{f(x_i)})] - \frac{1}{2}\lambda\int[f''(t)]^2dt \end{aligned}$$

onde $\pi(x) = \mathbb{P}(Y=1|X=x)$.

Representando $f(x) = \sum_{j=1}^{n} N_j(x)\theta_j$, chamado *natural spline*, em que $N_j(x)$ é um conjunto N-dimensional de funções bases, temos a primeira e

segunda derivadas representadas na forma matricial por

$$\frac{\partial l(\boldsymbol{\beta})}{\partial \beta} = \mathbf{N}'(\mathbf{y} - \mathbf{p}) - \lambda \boldsymbol{\Omega} \theta,$$

$$\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \beta \partial \beta'} = -\mathbf{N}'\mathbf{W}\mathbf{N} - \lambda \boldsymbol{\Omega},$$

onde $\{\mathbf{N}\}_{ij} = N_j(x_i)$ e $\{\boldsymbol{\Omega}_N\}_{jk} = \int N_j''(t) N_k''(t) dt$.

Assim os valores de θ e das funções ajustadas são obtidos iterativamente por meio de

$$\begin{aligned} \theta^{m+1} &= (\mathbf{N}'\mathbf{W}\mathbf{N} + \lambda\boldsymbol{\Omega})^{-1}\mathbf{N}'\mathbf{W}(\mathbf{N}\theta^m + \mathbf{W}^{-1}(\mathbf{y} - \mathbf{p})) \\ &= (\mathbf{N}'\mathbf{W}\mathbf{N} + \lambda\boldsymbol{\Omega})^{-1}\mathbf{N}'\mathbf{W}\mathbf{z} \end{aligned}$$

$$\begin{aligned} \mathbf{f}^{m+1} &= (\mathbf{N}'\mathbf{W}\mathbf{N} + \lambda\boldsymbol{\Omega})^{-1}\mathbf{N}'\mathbf{W}(\mathbf{f}^m + \mathbf{W}^{-1}(\mathbf{y} - \mathbf{p})) \\ &= \mathbf{S}_{\lambda,w}\mathbf{z}. \end{aligned}$$

3.3 k-Nearest Neighbor

Primeiramente apresentado em (9) e posteriormente desenvolvido e teoricamente provado em (6), trata-se de um classificador não-paramétrico, para o qual não é necessário um modelo a ser ajustado, onde, dado um ponto x_0 no espaço n-dimensional, que deva ser classificado, encontra-se k pontos pertencentes a amostra de treinamento $(x_{(i)}, i = 1, ..., k)$ mais próximos em distância (geralmente distância Euclidiana) do novo ponto. Assim, este tem sua classificação feita de acordo com a maioria das classificações existentes de seus k vizinhos com a finalidade de formar $\hat{Y}$ que é definido como

$$\hat{Y}(x_{(i)}) = \frac{1}{k} \sum_{x_0 \in N_k(x_{(i)})} y_0, \tag{3-9}$$

onde $N_k(x_{(i)})$ é a vizinhança de $x_{(i)}$ definida pelos k pontos amostrais próximos de x_0 na amostra de treinamento. Tal procedimento nada mais é do que encontrar as k observações mais próximas do novo ponto, x_0, e tirar a média dos valores de suas variáveis dependentes.

A figura 3.4 ilustra um exemplo de como a variação no valor de k influencia na classificação. Para $k = 5$ classificaríamos o novo ponto como sendo um círculo e caso o número aumentasse para $k = 15$, por exemplo, a

classificação mudaria para um quadrado.

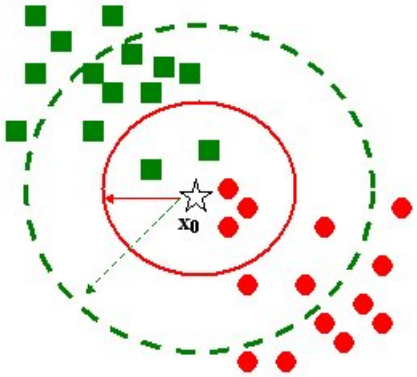

Figura 3.4: Exemplo: k-Nearest Neighbor

3.4 Análise Discriminante

A Análise Discriminante é uma metodologia que permite classificar duas ou mais populações e com esta separação prévia poder alocar um novo objeto a uma das classes existentes. Para tal é calculada uma função, que é a combinação linear das covariáveis, denominada *função discriminante*. Os principais pressupostos desta função são: a variável dependente deve seguir uma distribuição Normal multivariada e as matrizes de covariância (Σ) sejam iguais.

Utiliza-se a técnica através da *Função Discriminante Linear de Fisher* (em inglês, *Fisher Discriminant Linear - FDL*) que, conforme apresentado em (16), transforma as observações multivariadas $\mathbf{x}$ em observações univariadas y, tal que os y´s das populações P_1 e P_2 fossem separados o máximo possível.

Assim sendo, a função discriminante de Fisher tem a forma da combinação linear $\hat{\mathbf{y}} = \hat{\mathbf{a}}'\mathbf{x}$, onde $\hat{\mathbf{a}}' = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\mathbf{S}_p^{-1}$ e $\mathbf{S}_p = \frac{(n_1-1)S_1+(n_2-1)S_2}{n_1+n_2-2}$. Considerando os estimadores S e $\bar{\mathbf{x}}$ referentes a Σ e $\boldsymbol{\mu}$.

A expressão $\hat{\mathbf{y}} = (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)'\mathbf{S}_p^{-1}\mathbf{x}$ maximiza a razão

$$\frac{(\bar{\mathbf{y}}_1 - \bar{\mathbf{y}}_2)^2}{s_y^2} = \frac{(\hat{\mathbf{a}}'\bar{\mathbf{x}}_1 - \hat{\mathbf{a}}'\bar{\mathbf{x}}_2)^2}{\hat{\mathbf{a}}'\mathbf{S}_p\hat{\mathbf{a}}} = \frac{(\hat{\mathbf{a}}'\mathbf{d})^2}{\hat{\mathbf{a}}'\mathbf{S}_p\hat{\mathbf{a}}}$$

onde $\mathbf{d} = \bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2$. O resultado e prova da maximização são encontrados em (16) (pp. 610).

Para uma nova observação, x_0, a regra de alocação em uma das populações discriminadas pela função é a seguinte

– Alocação em P_1 se:

$$\hat{y}_0 = \hat{\mathbf{a}}'\mathbf{x}_0 \geq \frac{1}{2}\,\hat{\mathbf{a}}'(\bar{\mathbf{x}}_1 + \bar{\mathbf{x}}_2) = \frac{\bar{\mathbf{y}}_1 - \bar{\mathbf{y}}_2}{2};$$

– Alocação em P_2 se:

$$\hat{y}_0 < \frac{\bar{\mathbf{y}}_1 - \bar{\mathbf{y}}_2}{2}.$$

3.4.1
Estimação das Probabilidades de Classificação Incorreta

Em se tratando das populações P_1 e P_2, podem ser cometidos dois tipos de erros. Segundo (23) são eles:

– Erro 1: elementos provenientes da população 1 que são classificados como pertencentes à população 2;

– Erro 2: elementos provenientes da população 2 que são classificados como pertencentes à população 1.

Desta maneira define-se $\mathbb{P}(Erro\ 1) = p(2|1)$ e $\mathbb{P}(Erro\ 2) = p(1|2)$.

Uma forma de visualizar tais erros é através da *Matriz de Confusão*, que é um artifício semelhante à Tabela de Classificação, como se pode notar na tabela 3.2

Tabela 3.2: Matriz de Confusão

	População Classificada		
População Real	pop 1	pop 2	
pop 1	n_{11}	n_{12}	N_1
pop 2	n_{21}	n_{22}	N_2

Seus elementos são:

– n_{11}: itens de P_1 classificados corretamente em P_1;

– n_{12}: itens de P_1 classificados incorretamente em P_2;

– n_{21}: itens de P_2 classificados incorretamente em P_1;

– n_{22}: itens de P_2 classificados corretamente em P_2;

– N_1: total de itens em P_1;

– N_2: total de itens em P_2.

A Taxa Aparente de Erro (APER) definida em (16) é dada por

$$APER = \frac{n_{12} + n_{21}}{N_1 + N_2}$$

Ainda com a tabela 3.2 calculamos as estimativas das probabilidades dos erros, dadas por: $\hat{p}(1|2) = \frac{n_{21}}{N_2}$ e $\hat{p}(2|1) = \frac{n_{12}}{N_1}$. Quanto menores elas forem, melhor será a função de discriminação.

A avaliação e construção da matriz de confusão serão feitas neste trabalho através do *Método da Ressubstituição* (ver (23)) em que os escores de cada elemento amostral observado de P_1 e P_2 são calculados, sendo a regra de discriminação utilizada para classificar os $N = N_1 + N_2$ elementos da amostra conjunta. Assim os mesmos elementos amostrais participam da estimação da regra de classificação e da estimação dos erros. Outros dois métodos utilizados são: *Método Holdout* e o *Método de Lachenbruch*, extensamente debatidos em (16) e (23).

4
Modelo de Regressão Logística com Transição Suave Estruturado por Árvore (STLR-Tree)

Na busca por uma melhor explicação de fenômenos complexos por modelos de regressão e de séries temporais, a utilização da modelagem não-linear vem crescendo ao longo dos anos amparada pelo avanço de recursos computacionais e a modernização dos pacotes estatísticos.

A modelagem não-linear vem para superar os métodos lineares de previsão de maior recorrência, indo além dos modelos estacionários Gaussianos e abrangendo as situações onde os dados apresentam um comportamento que não equivale ao linear tendo algumas características, citadas por (8), por exemplo: não-normalidade, ciclos assimétricos, bimodalidade, não-normalidade entre variáveis defasadas, variação do desempenho de previsão sobre o espaço de estado, irreversibilidade temporal etc. Podemos acrescentar a isso outras características como: ciclos-limite, salto de ressonância, amplitude dependente da freqüência e caos. Estes últimos inseridos no contexto da análise de séries temporais. No Apêndice A encontram-se alguns dos modelos não-lineares, em sua maioria utilizados na análise de séries temporais, que, de alguma forma, estão relacionados com o STLR-Tree, seja em sua forma estrutura, estimação e/ou previsão.

A idéia central deste trabalho é adaptar o modelo proposto em (7), usando-o como um método de classificação, para o caso em que nossa variável dependente, y_i, assuma apenas dois valores, 0 ou 1, caindo assim no contexto de uma Regressão Logística.

4.1
Revisão do STR-Tree

Proposto em (7) o modelo STR-Tree tem como idéia principal a substituição das transições abruptas nas árvores de regressão feitas através da função indicadora $I(\mathbf{x}_i; c)$ pela função logística definida por

$$G(x_i; \gamma, c) = \frac{1}{1 + e^{-\gamma(x_i - c)}}. \tag{4-1}$$

Definição 4.1 *Seja* $\boldsymbol{z}_i \subseteq \boldsymbol{x}_i$ *tal que* $\boldsymbol{x}_i \in \mathbb{X} \subseteq \mathbb{R}^q$ *e* $\boldsymbol{z}_i \in \mathbb{R}^p$ *onde* $p \leq q$. *Considere* $\tilde{\boldsymbol{z}}_i = (1, \boldsymbol{z}_i)'$. *Um modelo paramétrico* $\mathcal{M}$ *definido pela função* $H_{\mathbb{J}\mathbb{T}}(\boldsymbol{x}_i; \psi) : \mathbb{R}^{q+1} \to \mathbb{R}$, *indexado pelo vetor de parâmetros* $\psi \in \boldsymbol{\Psi}$, *um subconjunto compacto do espaço Euclidiano, é chamado modelo Smooth Transition Regression Tree (STR-Tree) se*

$$y_i = H_{\mathbb{J}\mathbb{T}}(\boldsymbol{x}_i; \psi) = \sum_{k \in \mathbb{T}} \boldsymbol{\beta}_k' \tilde{\boldsymbol{z}}_i B_{\mathbb{J}_k}(\boldsymbol{x}_i; \boldsymbol{\theta}_k) + \epsilon_i \tag{4-2}$$

onde

$$B_{\mathbb{J}_k}(\boldsymbol{x}_i; \boldsymbol{\theta}_k) = \prod_{j \in \mathbb{J}} G(x_{s_j,i}; \gamma_j, c_j)^{\frac{n_{k,j}(1+n_{k,j})}{2}} \left[1 - G(x_{s_j,i}; \gamma_j, c_j)\right]^{(1-n_{k,j})(1+n_{k,j})},$$

e $n_{i,j}$ *assume três valores:*

- -1, *se o caminho até o nó i não incluir o nó gerador j;*
- 0, *se caminho para o nó i incluir o nó filho da direita do nó gerador j;*
- 1, *se o caminho para o nó terminal i incluir o nó filho da esquerda do nó gerador j.*

Sendo $\mathbb{J}_k$ *o subconjunto de* $\mathbb{J}$ *que contém os índices dos nós geradores (ou nós pais) do caminho para o nó terminal k. Então,* $\boldsymbol{\theta}_k$ *é o vetor que contém todos os parâmetros não-lineares* (γ_t, c_t) *tal que* $t \in \mathbb{J}_k, k \in \mathbb{T}$ *(índice dos nós terminais).*

As funções $B_{\mathbb{J}_k}$ são tais que, $0 < B_{\mathbb{J}_k} < 1$ e $\sum_{j \in \mathbb{J}} B_{\mathbb{J}_k}(\mathbf{x}_i; \boldsymbol{\theta}_j) = 1, \forall \mathbf{x}_i \in \mathbb{R}^{q+1}$.

Vamos considerar agora um STR-Tree em uma árvore completamente crescida, ou cheio, com profundidade d, $K = 2d$, nós terminais (folhas) e $N = \sum_{k=1}^{d} 2^k$ nós geradores definido como

$$y_i = \sum_{k=1}^{K} \boldsymbol{\beta}_{K+k-2}' \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k) + \epsilon_i, \tag{4-3}$$

onde $y_i \in \mathbb{R}$, $\mathbf{x}_i = (x_{1i}, ...x_{qi})' \in \mathbb{X} \subseteq \mathbb{R}^q$ e considerando que o vetor $\mathbf{z}_i$ receba valores defasados das variáveis explicativas, temos que $\mathbf{z}_i = (x_{1,i}, ..., x_{1,i-p_1}, ..., x_{k,i}, ..., x_{k,i-p_k})' \in \mathbb{R}^m$, onde $m = p + \sum_{j=1}^{k}(p_j + 1)$, sem esquecer que $\tilde{\mathbf{z}}_i = (1, \mathbf{z}_i)'$.

O vetor de parâmetros $\boldsymbol{\psi} = (\boldsymbol{\beta}_{K-1}', ..., \boldsymbol{\beta}_{2K-2}', \boldsymbol{\theta}_1', ..., \boldsymbol{\theta}_K')' \in \mathbb{R}^r$ possui $r = (p+1)K + 2N$.

Dois exemplos de estruturas de árvore são dados a seguir. Primeiramente o caso mais simples, figura 4.1, com uma profundidade, ou seja, quando temos o nó raiz se dividindo em apenas dois nós terminais, assim $d = 1, K = 2$ e $N = 1$. Este modelo sofrerá uma reparametrização para que possamos, mais adiante, fazer as hipóteses em relação a divisão dos nós. O segundo exemplo, figura 4.2, é uma estrutura de árvore maior com profundidade $d = 2$, quatro nós terminais, $K = 4$, e $N = 3$. Abaixo se encontram as equações de cada um deles bem como suas figuras ilustrativas.

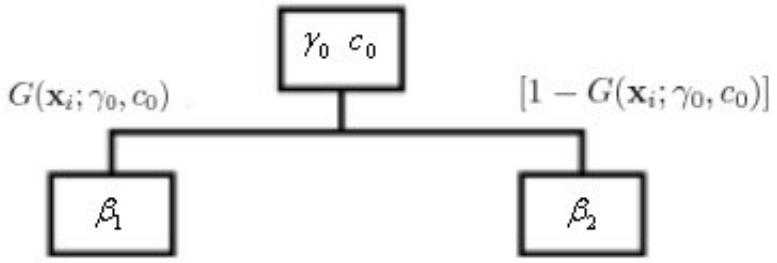

Figura 4.1: Exemplo Árvore 1

$$y_i = \boldsymbol{\beta}_1' \tilde{\mathbf{z}}_i G(\mathbf{x}_i; \gamma_0, c_0) + \boldsymbol{\beta}_2' \tilde{\mathbf{z}}_i [1 - G(\mathbf{x}_i; \gamma_0, c_0)] + \epsilon_i$$

Reparametrizando a fim de obter uma representação mais parcimoniosa,

$$y_i = \boldsymbol{\phi}_0' \tilde{\mathbf{z}}_i + \boldsymbol{\lambda}_0' \tilde{\mathbf{z}}_i G(\mathbf{x}_i; \gamma_0, c_0) + \epsilon_i \tag{4-4}$$

onde $\boldsymbol{\phi}_0 = \boldsymbol{\beta}_2$ e $\boldsymbol{\lambda}_0 = \boldsymbol{\beta}_1 - \boldsymbol{\beta}_2$

O outro exemplo é

$$\begin{aligned} y_i \quad &= \{\boldsymbol{\beta}_3' \tilde{\mathbf{z}}_i G(\mathbf{x}_i; \gamma_1, c_1) + \boldsymbol{\beta}_4' \tilde{\mathbf{z}}_i [1 - G(\mathbf{x}_i; \gamma_1, c_1)]\} G(\mathbf{x}_i; \gamma_0, c_0) \\ &+ \{\boldsymbol{\beta}_5' \tilde{\mathbf{z}}_i G(\mathbf{x}_i; \gamma_2, c_2) + \boldsymbol{\beta}_6' \tilde{\mathbf{z}}_i [1 - G(\mathbf{x}_i; \gamma_2, c_2)]\} [1 - G(\mathbf{x}_i; \gamma_0, c_0)] + \epsilon_i \end{aligned}$$

$$\begin{aligned} y_i \quad &= \boldsymbol{\beta}_3' \tilde{\mathbf{z}}_i G(\mathbf{x}_i; \gamma_0, c_0) G(\mathbf{x}_i; \gamma_1, c_1) + \boldsymbol{\beta}_4' \mathbf{z}_i G(\mathbf{x}_i; \gamma_0, c_0) [1 - G(\mathbf{x}_i; \gamma_1, c_1)] \\ &+ \boldsymbol{\beta}_5' \tilde{\mathbf{z}}_i [1 - G(\mathbf{x}_i; \gamma_0, c_0)] G(\mathbf{x}_i; \gamma_2, c_2) \\ &+ \boldsymbol{\beta}_6' \tilde{\mathbf{z}}_i [1 - G(\mathbf{x}_i; \gamma_0, c_0)][1 - G(\mathbf{x}_i; \gamma_2, c_2)] + \epsilon_i \end{aligned}$$

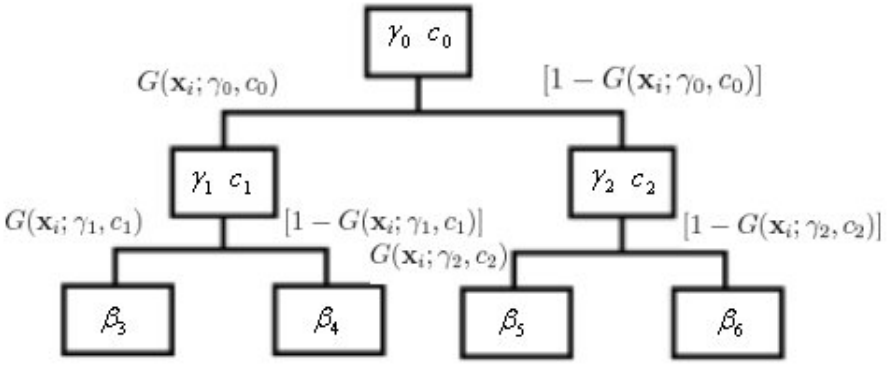

Figura 4.2: Exemplo Árvore 2

com isso podemos deduzir que

$$
\begin{aligned}
B_1(\mathbf{x}_i;\boldsymbol{\theta}_1) &= G(\mathbf{x}_i;\gamma_0,c_0)G(\mathbf{x}_i;\gamma_1,c_1)\\
B_2(\mathbf{x}_i;\boldsymbol{\theta}_2) &= G(\mathbf{x}_i;\gamma_0,c_0)[1-G(\mathbf{x}_i;\gamma_1,c_1)]\\
B_3(\mathbf{x}_i;\boldsymbol{\theta}_3) &= [1-G(\mathbf{x}_i;\gamma_0,c_0)]G(\mathbf{x}_i;\gamma_2,c_2)\\
B_4(\mathbf{x}_i;\boldsymbol{\theta}_4) &= [1-G(\mathbf{x}_i;\gamma_0,c_0)][1-G(\mathbf{x}_i;\gamma_2,c_2)]
\end{aligned}
$$

assim

$$
\begin{aligned}
y_i \quad &= \boldsymbol{\beta}_3'\tilde{\mathbf{z}}_i B_1(\mathbf{x}_i;\boldsymbol{\theta}_1) + \boldsymbol{\beta}_4'\tilde{\mathbf{z}}_i B_2(\mathbf{x}_i;\boldsymbol{\theta}_2)\\
&\quad +\boldsymbol{\beta}_5'\tilde{\mathbf{z}}_i B_3(\mathbf{x}_3;\boldsymbol{\theta}_3) + \boldsymbol{\beta}_6'\tilde{\mathbf{z}}_i B_4(\mathbf{x}_4;\boldsymbol{\theta}_4) + \epsilon_i
\end{aligned}
$$

$$
y_i = \sum_{k=1}^{4} \boldsymbol{\beta}_{K+k-2}'\tilde{\mathbf{z}}_i B_k(\mathbf{x}_i;\boldsymbol{\theta}_k) + \epsilon_i.
$$

4.2 Especificação do STLR-Tree

Ao especificar o modelo devemos primeiramente selecionar as variáveis relevantes, elementos de $\mathbf{z}_i$. A segunda etapa na especificação do modelo é a busca pelo nó a ser dividido e, na seqüência, a escolha da a variável de transição, elementos de $\mathbf{x}_i$.

Antes de descrever as etapas de especificação vamos reescrever o modelo STR-Tree levando em consideração que nossa variável dependente seja dicotômica, onde cada elemento tenha uma distribuição Bernoulli de parâmetro

π, e que tanto $\mathbf{z}_i$ quanto $\mathbf{x}_i$ sejam variáveis contínuas, independentes entre si, respeitando que $\mathbf{z}_i \subseteq \mathbf{x}_i$. Além disso, vamos considerar o modelo em uma árvore completamente crescida, com profundidade d e $K = 2d$, nós terminais. Desta maneira, podemos escrever o modelo de modo similar a uma Regressão Logística, utilizando como função de ligação a logito, denominado STLR-Tree, como se segue

$$\log\left[\frac{\pi(\tilde{\mathbf{z}}_i)}{1-\pi(\tilde{\mathbf{z}}_i)}\right] = \sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2}\tilde{\mathbf{z}}_i B_k(\mathbf{x}_i;\boldsymbol{\theta}_k), \tag{4-5}$$

como descrito anteriormente, após algumas contas, encontra-se o valor de $\pi(\tilde{\mathbf{z}}_i)$ igual a

$$\pi(\tilde{\mathbf{z}}_i) = \frac{e^{\sum_{k=1}^{K}\boldsymbol{\beta}'_{K+k-2}\tilde{\mathbf{Z}}_i B_k(\mathbf{X}_i;\boldsymbol{\theta}_k)}}{1+e^{\sum_{k=1}^{K}\boldsymbol{\beta}'_{K+k-2}\tilde{\mathbf{Z}}_i B_k(\mathbf{X}_i;\boldsymbol{\theta}_k)}}. \tag{4-6}$$

Por fim, temos a função de log-verossimilhança do STLR-Tree, semelhante à encontrada em (2-14), que será utilizada em seções posteriores.

$$\begin{aligned} l(\boldsymbol{\beta}) \quad &= \sum_{i=1}^{n}\left[y_i\sum_{k=1}^{K}\boldsymbol{\beta}'_{K+k-2}\tilde{\mathbf{z}}_i B_k(\mathbf{x}_i;\boldsymbol{\theta}_k) - \right. \\ &\quad \left. - \log\left(1+e^{\sum_{k=1}^{K}\boldsymbol{\beta}'_{K+k-2}\tilde{\mathbf{Z}}_i B_k(\mathbf{X}_i;\boldsymbol{\theta}_k)}\right)\right] \end{aligned}$$

4.2.1 Escolha das variáveis relevantes

Para a escolha dos elementos de $\mathbf{z}_i$ destacam-se três métodos: por critérios de informação, AIC e BIC, os quais já foram descritos em (2-12) e (2-13) respectivamente, sendo o melhor modelo aquele que minimiza tais critérios; aproximação polinomial do modelo, proposto em (26); outra opção é dada através de técnicas não paramétricas, porém esta classe é computacionalmente dispendiosa principalmente para um grande número de observações.

No caso o STLR-Tree será restrito para variáveis contínuas independentes. Se fossem consideradas variáveis categóricas, a extensão do modelo que contempla tal mudança é feita através da inclusão de um vetor contendo variáveis indicadoras, $\mathbf{D}_i(\mathbf{w}_i)$, que representa o vetor categórico denominado

por $\mathbf{w}_i$. Assim o modelo em (4-5) terá a forma:

$$\log\left[\frac{\pi(\tilde{\mathbf{z}}_i)}{1-\pi(\tilde{\mathbf{z}}_i)}\right] = \sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2}\tilde{\mathbf{z}}_i\mathbf{D}_i(\mathbf{w}_i)B_k(\mathbf{x}_i;\boldsymbol{\theta}_k), \tag{4-7}$$

4.2.2
Escolha do nó a ser dividido

Estudo da significância da divisão dos nós através de testes de hipótese, fundamentados em inferência estatística. A descrição a seguir é similar ao procedimento de estimação do modelo STAR seguindo a abordagem *específico-geral* partindo de um modelo simples para um modelo mais complexo à medida que testes de diagnóstico assim o permita. Maiores detalhes do ciclo de modelagem STAR em (32).

Estamos testando a linearidade do modelo, o que implica dizer se devemos ou não dividir um nó.

Neste ponto iremos voltar a escrever o modelo no caso clássico, considerando que $\epsilon_i \sim NID(0,\sigma^2)$, para justificar algumas manipulações algébricas entendidas com mais facilidade nesta situação. Assim, considerando K nós terminais e testando se o nó $k^* \in \mathbb{T}$ será dividido temos o modelo escrito como:

$$\begin{aligned} y_i = & \sum_{k\in\mathbb{T}-\{k^*\}} \boldsymbol{\beta}'_k\tilde{\mathbf{z}}_i B_{\mathbb{J}_k}(\mathbf{x}_i;\boldsymbol{\theta}_k) + \boldsymbol{\beta}'_{2k^*+1}\tilde{\mathbf{z}}_i B_{\mathbb{J}_{2k^*+1}}(\mathbf{x}_i;\boldsymbol{\theta}_{2k^*+1}) + \\ + & \boldsymbol{\beta}'_{2k^*+2}\tilde{\mathbf{z}}_i B_{\mathbb{J}_{2k^*+2}}(\mathbf{x}_i;\boldsymbol{\theta}_{2k^*+2}) + \epsilon_i \end{aligned}$$

onde

$$\begin{aligned} B_{\mathbb{J}_{2k^*+1}}(\mathbf{x}_i;\boldsymbol{\theta}_{2k^*+1}) &= B_{\mathbb{J}_{k^*}}(\mathbf{x}_i;\boldsymbol{\theta}_{k^*})G(x_{k^*i};\gamma_{k^*},c_{k^*}) \\ B_{\mathbb{J}_{2k^*+2}}(\mathbf{x}_i;\boldsymbol{\theta}_{2k^*+2}) &= B_{\mathbb{J}_{k^*}}(\mathbf{x}_i;\boldsymbol{\theta}_{k^*})[1-G(x_{k^*i};\gamma_{k^*},c_{k^*})] \end{aligned}$$

Como feito em (4-4) reparametrizamos o modelo de tal forma que

$$\begin{aligned} y_i = & \sum_{k\in\mathbb{T}-\{k^*\}} \boldsymbol{\beta}'_k\tilde{\mathbf{z}}_i B_{\mathbb{J}_k}(\mathbf{x}_i;\boldsymbol{\theta}_k) + \boldsymbol{\phi}'\tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i;\boldsymbol{\theta}_{k^*}) + \\ + & \boldsymbol{\lambda}'\tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i;\boldsymbol{\theta}_{k^*})G(x_{k^*i};\gamma_{k^*},c_{k^*}) + \epsilon_i \end{aligned}$$

onde $\boldsymbol{\phi} = \boldsymbol{\beta}_{2k^*+1}$ e $\boldsymbol{\lambda} = \boldsymbol{\beta}_{2k^*+1} - \boldsymbol{\beta}_{2k^*+2}$.

Deve-se testar a hipótese de significância dessa divisão (inclusão de um

novo regime), que é equivalente a testar as hipóteses:

$$\begin{cases} H_0 : \gamma_{k^*} = 0 \\ H_1 : \gamma_{k^*} > 0. \end{cases} \tag{4-8}$$

Porém, sob H_0 temos que enfrentar um problema de especificação do modelo, pois os parâmetros γ_{k^*} e c_{k^*} podem assumir diferentes valores sem alterar a função de verossimilhança. Para solucionar tal problema foi proposto em (18) uma aproximação da função de transição por uma expansão de Taylor de terceira ordem em torno de $\gamma_{k^*} = 0$ e assim, após manipulações algébricas, podemos reescrever o modelo como

$$\begin{aligned} y_i = & \sum_{k \in \mathbb{T} - \{k^*\}} \boldsymbol{\beta}_k' \tilde{\mathbf{z}}_i B_{\mathbb{J}_k}(\mathbf{x}_i; \boldsymbol{\theta}_k) + \boldsymbol{\alpha}_0' \tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i; \boldsymbol{\theta}_{k^*}) + \\ + & \boldsymbol{\alpha}_1' \tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i; \boldsymbol{\theta}_{k^*}) + \boldsymbol{\alpha}_2' \tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i; \boldsymbol{\theta}_{k^*}) + \boldsymbol{\alpha}_3' \tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i; \boldsymbol{\theta}_{k^*}) + e_i \end{aligned}$$

onde $e_i = \epsilon_i + \boldsymbol{\lambda}' \tilde{\mathbf{z}}_i R(x_{k^* i}; \gamma_{k^*}, c_{k^*})$ e $R(x_{k^* i}; \gamma_{k^*}, c_{k^*})$ é o termo restante da expansão de Taylor.

Assim podemos reescrever a hipótese de nulidade dos parâmetros como:

$$H_0 : \alpha_i = 0, \quad i = 1, 2 \text{ e } 3$$

e sob H_0 temos que $e_i = \epsilon_i$.

Com isso, pegando o último modelo escrito anteriormente e considerando novamente que a variável dependente é binária, voltamos para o modelo em que

$$\begin{aligned} \log\left[\frac{\pi(\tilde{\mathbf{z}}_i)}{1-\pi(\tilde{\mathbf{z}}_i)}\right] = & \sum_{k \in \mathbb{T} - \{k^*\}} \boldsymbol{\beta}_k' \tilde{\mathbf{z}}_i B_{\mathbb{J}_k}(\mathbf{x}_i; \boldsymbol{\theta}_k) + \boldsymbol{\alpha}_0' \tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i; \boldsymbol{\theta}_{k^*}) + \\ + & \boldsymbol{\alpha}_1' \tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i; \boldsymbol{\theta}_{k^*}) + \boldsymbol{\alpha}_2' \tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i; \boldsymbol{\theta}_{k^*}) + \boldsymbol{\alpha}_3' \tilde{\mathbf{z}}_i B_{\mathbb{J}_{k^*}}(\mathbf{x}_i; \boldsymbol{\theta}_{k^*}). \end{aligned}$$

Em seguida é feita uma seqüência de teste de *Razão de Verossimilhança* comparando o modelo sob H_0 contra o modelo irrestrito através da estatística de teste, que foram mostradas em (2-14) e (2-15). A primeira compara os valores do logaritmo da função de verossimilhança maximizada e a segunda está em termos da diferença entre as *deviances* dos modelos conforme mostrado a seguir

$$\xi_{RV} = 2\left[l(\hat{\boldsymbol{\beta}}; \mathbf{y}) - l(\boldsymbol{\beta}^0; \mathbf{y})\right] = \phi^{-1}\left[D(\mathbf{y}; \hat{\boldsymbol{\mu}}^0) - D(\mathbf{y}; \hat{\boldsymbol{\mu}})\right] \xrightarrow{a} \chi^2_{3(p+1)}, \tag{4-9}$$

em que $\hat{\boldsymbol{\mu}}^0 = \mathbf{g}^{-1}(\hat{\boldsymbol{\eta}})^0$ e $\hat{\boldsymbol{\eta}}^0 = \mathbf{X}\boldsymbol{\beta}^0$.

Se desconhecido o parâmetro de dispersão ϕ deve ser substituído por uma estimativa consistente, $\hat{\phi}$. Contudo para o caso binomial temos que $\phi = 1$, desta forma

$$\xi_{RV} = \left[D(\mathbf{y}; \hat{\boldsymbol{\mu}}^0) - D(\mathbf{y}; \hat{\boldsymbol{\mu}})\right] \xrightarrow{a} \chi^2_{3(p+1)}, \tag{4-10}$$

A inferência também pode ser baseada na estatística F da seguinte maneira:

$$F = \frac{\left[D(\mathbf{y}; \hat{\boldsymbol{\mu}}^0) - D(\mathbf{y}; \hat{\boldsymbol{\mu}})\right]}{3(p+1)} \xrightarrow{a} F_{3(p+1),N-4(p+1)}. \tag{4-11}$$

A idéia de utilizar a razão entre *deviances* para determinar o crescimento da árvore também é utilizada em (5) e (14).

4.2.3
Escolha das variáveis de transição

Aplicar os testes de RV para cada uma das variáveis explicativas e selecionar a variável $x_{s_0 t}$ que gere o menor *p-valor*, sob um nível de significância α. Sabe-se que $s_0 \in \mathbb{S} = \{1, 2, ..., m\}$, o conjunto dos índices dos elementos em $\mathbf{x}_i$.

4.3
Estimação do STLR-Tree

A estimação da parte linear é feita por Máxima Verossimilhança, utilizando o método iterativo de Newton-Raphson. Para isso é necessário o cálculo de $\frac{\partial l(\boldsymbol{\beta})}{\partial \beta}$ e $\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \beta \partial \beta'}$

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left[y_i \sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k) - \log\left(1 + e^{\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{Z}}_i B_k(\mathbf{X}_i; \boldsymbol{\theta}_k)}\right)\right]$$

$$\begin{aligned} \frac{\partial l(\boldsymbol{\beta})}{\partial \beta} = & \sum_{t=1}^{n} \left\{ \left[y_i \sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k) \right] - \right. \\ & - \left. \left[\frac{e^{\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{Z}}_i B_k(\mathbf{X}_i; \boldsymbol{\theta}_k)}}{\left(1 + e^{\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{Z}}_i B_k(\mathbf{X}_i; \boldsymbol{\theta}_k)}\right)} \right] \times \right. \end{aligned}$$

$$\times \quad \left[\frac{\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k)}{\left(1 + e^{\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k)}\right)} \right] \Bigg\}$$

$$\begin{aligned} \frac{\partial l(\boldsymbol{\beta})}{\partial \beta} = & \quad \sum_{i=1}^{n} \left\{ \left[\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k) \right] \times \right. \\ & \times \quad \left. \left[y_i - \frac{e^{\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k)}}{\left(1 + e^{\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k)}\right)} \right] \right\} \end{aligned}$$

$$\frac{\partial l(\boldsymbol{\beta})}{\partial \beta} = \sum_{i=1}^{n} \left\{ \left[\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k) \right] [y_i - \pi(\mathbf{z}_i)] \right\}.$$

Da mesma forma deduziremos o Hessiano

$$\begin{aligned} \frac{\partial^2 l(\boldsymbol{\beta})}{\partial \beta \partial \beta'} = & \quad -\sum_{i=1}^{n} \left\{ \pi(\mathbf{z}_i) \left[\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k) \right]^2 - \right. \\ & - \quad \left. [\pi(\mathbf{z}_i)]^2 \left[\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k) \right]^2 \right\}, \end{aligned}$$

simplificando

$$\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \beta \partial \beta'} = -\sum_{i=1}^{n} \left\{ \left[\sum_{k=1}^{K} \boldsymbol{\beta}'_{K+k-2} \tilde{\mathbf{z}}_i B_k(\mathbf{x}_i; \boldsymbol{\theta}_k) \right]^2 \pi(\mathbf{z}_i)[1 - \pi(\mathbf{z}_i)] \right\}.$$

Neste ponto fixamos os parâmetros não-lineares, γ e c, e obtemos seus valores iniciais através de uma procura em *grid*. Supondo conhecidos tais parâmetros o modelo pode ser encarado como uma Regressão Logística onde o vetor de parâmetros, $\boldsymbol{\beta}$, é estimado por Máxima Verossimilhança, que necessita da utilização do processo iterativo de Newton-Raphson, como mostrado em (2-20). Desta forma para um STLR-Tree completo temos que

$$\beta^{(m+1)} = [\mathbf{B}(\boldsymbol{\theta})' \mathbf{W}^{(m)} \mathbf{B}(\boldsymbol{\theta})]^{-1} \mathbf{B}(\boldsymbol{\theta})' \mathbf{W}^{(m)} \mathbf{z}^{(m)}$$

onde $\mathbf{z} = \mathbf{B}(\boldsymbol{\theta})\beta^{(m)} + \mathbf{W}^{-1}(\mathbf{y} - \boldsymbol{\pi})$ e

$$\mathbf{B}(\boldsymbol{\theta}) = \begin{pmatrix} B_1(\mathbf{x}_1;\boldsymbol{\theta}_1) & \dots & B_K(\mathbf{x}_1;\boldsymbol{\theta}_K) \\ \vdots & \ddots & \vdots \\ B_1(\mathbf{x}_N;\boldsymbol{\theta}_1) & \dots & B_K(\mathbf{x}_N;\boldsymbol{\theta}_K) \end{pmatrix}$$

Após estimados os parâmetros da parte linear, faz-se a estimação de γ e c também por Máxima Verossimilhança usando as estimativas de $\boldsymbol{\beta}$ nos cálculos.

A estimação dos parâmetros lineares e não-lineares segue basicamente o seguinte processo iterativo:

1. encontra-se os valores iniciais dos parâmetros não-lineares, γ e c, através de uma busca em *grid*, ao maximizar a log-verossimilhança concentrada;

2. aplica-se os valores encontrados no item anterior e estima-se por Máxima Verossimilhança os parâmetros lineares, $\boldsymbol{\beta}$;

3. as estimativas de $\boldsymbol{\beta}$ são usadas na estimação de γ e c também por Máxima Verossimilhança;

4. alternar os dois passos anteriores.

4.4 Avaliação do STLR-Tree

Feita após terem se encerrado as divisões e a árvore não possa mais crescer, assim nenhum nó terminal poderá ser dividido. A avaliação do ajuste é feita através dos métodos expostos em 2.2.3 onde são apresentadas as técnicas de avaliação do ajuste de uma Regressão Logística, que é basicamente a análise da Tabela de Classificação e da área abaixo da curva ROC.

Neste trabalho, entretanto, foi utilizada apenas a Tabela de Classificação (ou Matriz de Confusão) como medida da qualidade do ajuste.

4.5 Ciclo de Modelagem

Para minimizar a possibilidade de superestimar a quantidade de resultados significativos, evitando assim que sejam feitas mais divisões que o necessário, o nível de significância, α, sofre um desconto, ou defasagem, a medida que a árvore cresce da seguinte maneira

$$\alpha(d,n) = \frac{\alpha}{n^d} \tag{4-12}$$

onde n indica o n-ésimo teste aplicado e d a profundidade.

Assim, após a profundidade d=0, onde é aplicado somente um teste ($n = 1$) temos o nível de significância igual a α, a partir daí, na primeira profundidade ($d = 1$), são aplicados dois testes ($n = 2$) e o nível de significância passa a valer $\frac{\alpha}{2}$. A evolução de α de acordo com a profundidade e o número de testes se dá a partir do nó raiz como: $\alpha, \frac{\alpha}{2}, \frac{\alpha}{3}, \frac{\alpha}{4^2}, \frac{\alpha}{5^2}, \frac{\alpha}{6^2}, \frac{\alpha}{7^2}, \frac{\alpha}{8^3}, \frac{\alpha}{9^3}, ...$

Tal método, que aumenta o rigor da significância com o aumento de d, evita que se venha a utilizar técnicas de podagem da árvore a posteriori (*post prunning*).

Todas as etapas para obtenção do STLR-Tree devem ser aplicadas a cada nova profundidade da árvore. A seguir é apresentada a modelagem a partir do nó raiz, depois da primeira profundidade e de forma generalizada, para a *k-ésima* profundidade.

Criação da Primeira profundidade (a partir de $d = 0$)

Para cada variável explicativa aplicar os testes de RV comparando cada um dos modelos que, sob H_0, mantenha apenas a parte linear. Escolher as variável $x_{s_0 i}$ que gere o menor *p-valor*. Dado $s_0 \in \mathbb{S} = \{1, 2, ..., m\}$ é feita a estimação do vetor de parâmetros, $\boldsymbol{\psi} = (\gamma_0, c_0, \beta_1, \beta_2)'$ conforme os métodos de estimação especificados anteriormente e testa-se as hipóteses

$$\begin{cases} H_{01} : \beta_1 = 0 \\ H_{02} : \beta_2 = 0 \\ H_{03} : \beta_1 - \beta_2 = 0 | \beta_1, \beta_2 \neq 0. \end{cases} \tag{4-13}$$

Se pelo menos uma das hipóteses não for rejeitada, busca-se a próxima variável de transição que gerou o segundo menor *p-valor* e reestima-se os parâmetros. Se para todos os $s_0 \in \mathbb{S}$ não forem produzidas divisões estatisticamente significativas, ou seja, se as hipóteses de linearidade não forem rejeitadas, a raiz é declarada como nó terminal e o modelo apenas com a parte linear é estimado. Caso seja gerada uma divisão estatisticamente significativa, rejeitando-se as hipóteses nulas, dois nós filhotes são gerados.

Criação da Segunda profundidade (a partir de $d = 1$)

Os dois nós filhotes gerados a partir do nó raiz compõem a primeira profundidade. A partir desta, deve-se escolher além da variável de transição, um dos nós a ser dividido e, assim continuar a aplicação dos testes RV para verificar a significância da inclusão de um novo regime, utilizando-se ainda o critério de seleção do par de combinações entre o índice da variável de transição em $\mathbb{S} = \{1, 2, ..., m\}$ e o número do nó em $\mathbb{D} = \{1, 2\}$ que minimize

o *p-valor*. Sendo assim, estima-se os parâmetros e testa-se a significância da divisão através das hipóteses

$$\begin{cases} H_{01} : \beta_{2j_1+1} = 0 \\ H_{02} : \beta_{2j_1+2} = 0 \\ H_{03} : \beta_{2j_1+1} - \beta_{2j_1+2} = 0 | \beta_{2j_1+1}, \beta_{2j_1+2} \neq 0. \end{cases} \tag{4-14}$$

Se pelo menos uma das hipóteses não for rejeitada, busca-se o próximo par {nó; variável de transição}, que possua o segundo menor *p-valor*. Caso o contrário, a divisão seja aceita, testa-se a significância da divisão do nó $j_2 \in \mathbb{D} - \{j_1\}$, hipótese alternativa ao modelo com 3 nós terminais. Caso os dois nós da primeira profundidade gerem mais dois nós filhotes teremos, na segunda profundidade, 4 nós que serão: $2j_1 + 1, 2j_1 + 2, 2j_2 + 1$ e $2j_2 + 2$. Por outro lado, se nenhum dos dois nós gerarem divisões significativas paramos o crescimento da árvore e fazemos a avaliação do ajuste.

Criação da k-ésima profundidade

A generalização do processo de crescimento da árvore para uma profundidade $k > 0$, assumindo que foram criados N nós terminais anteriormente.

Para cada combinação $\{j_k; s_{j_k}\}$, de nó e variável de transição, é aplicado o teste de RV confrontando-os com o modelo apenas linear. As variáveis de transição pertencem ao conjunto $\mathbb{S} = \{1, 2, ..., m\}$ enquanto que os nós estão em $\mathbb{D}_k = \{2^k - 1, 2^k, ..., 2^{k+1} - 2\}$ e seleciona-se $j_k \in \mathbb{D}_k$ e $s_{j_k} \in \mathbb{S}$ que gere o menor *p-valor*. Com isso em mãos estima-se os parâmetros do modelo.

Como feito nas outras profundidades são aplicados os testes da significância das divisões iterativamente segundo a seqüência de nós: $j_2 \in \mathbb{D}_1 - \{j_1\}$, $j_3 \in \mathbb{D}_1 - \{j_1, j_2\}$, $j_4 \in \mathbb{D}_1 - \{j_1, j_2, j_3\}$,...

O ciclo é encerrado quando a profundidade em que se está aplicando os testes não gerem mais filhotes.

Simulação de dados através do STLR-Tree

Os dados foram gerados através de modelos STLR-Tree em sua forma mais simples, com apenas uma transição, alternando o parâmetro de suavidade entre os dois valores, $\gamma = 0.5$ e $\gamma = 50$. As covariáveis geradas foram: $x_1 \sim N(30, 8)$ e $x_2 \sim N(8.5, 6)$ e devido ao fato do parâmetro de suavidade ser dependente da escala, tais variáveis foram divididas por seus desvios padrão. Os valores dos parâmetros lineares para ambos os modelos são os seguintes: $\beta = \{2.5, -1.9, -0.2, 1.6, -0.17, 0.45\}$.

As figuras 4.3 e 4.4 correspondem ao modelo onde $c_0 = 4.3$ e $s_0 = 1$, para $\gamma = 0.5$ e $\gamma = 50$, respectivamente. Podemos notar a diferença entre as funções por meio da quebra que ocorre de acordo com a alteração do parâmetro de suavidade.

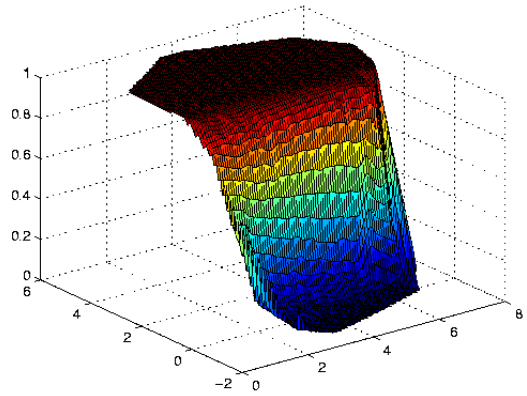

Figura 4.3: Dados gerados ($c_0 = 4.3$ e $s_0 = 1$ e $\gamma = 0.5$)

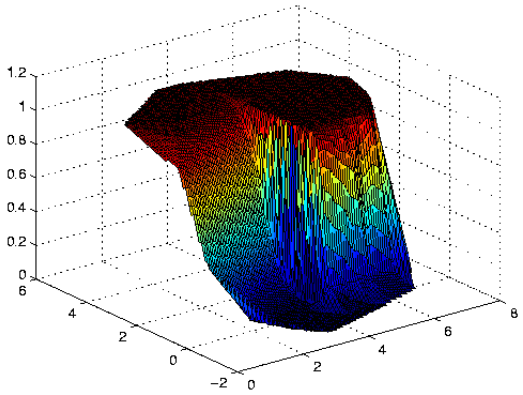

Figura 4.4: Dados gerados ($c_0 = 4.3$ e $s_0 = 1$ e $\gamma = 50$)

5
Aplicação e comparação dos métodos

Foram feitas comparações do STLR-Tree com os seguintes métodos de classificação: *Generalized Additive Models (GAM), Regressão Logística, Redes Neurais, Análise Discriminante, k-Nearest Neighbor (k-NN)* e *Classification and Regression Trees (CART)*. O STLR-Tree e as Redes Neurais foram programados no Matlab 6.1 e os demais métodos foram rodados no programa R 2.6.2, utilizando, além das funções existentes no próprio, bibliotecas tais como: tree (CART), VGAM (GAM) e kknn (k-NN), disponíveis na própria página do programa na internet:

http://cran.r-project.org.

No Apêndice B encontram-se alguns comandos do R utilizados para o CART, GAM, k-NN e Regressão Logística.

5.1
Bases de dados

A aplicação do STLR-Tree e a comparação com outros métodos de classificação foram feitas para três bases. Duas encontradas em exemplos do livro (14). Uma delas utilizada para fazer a distinção entre as mensagens que são realmente *e-mail* e as consideradas *spam* e outra para a ocorrência ou não de um possível infarto do miocárdio. Ambas disponibilizadas na página do próprio livro na internet:

http://www-stat.stanford.edu/ElemStatLearn.

A terceira provém de um estudo feito pra uma empresa do setor de energia elétrica no Estado do Rio de Janeiro, onde deseja-se classificar eventuais fraudes ou irregularidades de seus usuários. Este é o único exemplo no qual Redes Neurais foram utilizadas em comparação com os demais métodos de Classificação.

Uma breve descrição dos bancos segue a seguir:

– E-mail/Spam: contém dados de 4601 mensagens de e-mail, em que a variável dependente é igual a 0 se a mensagem foi considerada um e-mail

de fato ou 1 caso tinha sido caracterizada como um *spam* (mensagem que não é verdadeiramente um e-mail particular). Originalmente, a base possui 57 preditores dos quais foram selecionados 16 com base nos resultados de significância das mesmas disponíveis no livro onde aparecem como exemplo. Das 4601 mensagens, foram selecionadas 2000 aleatoriamente para compor a base *in sample* e 1000 para a base *out-of-sample*. As variáveis explicativas são as seguintes: percentual de palavras no e-mail que correspondam a: our, over, remove, internet, free, business, hp, hpl, george, 1999, re e edu; percentual de caracteres no e-mail que correspondam a: ! (char_!) e $ (char_$); comprimento da mais longa seqüência ininterrupta de letras maiúsculas (CAPMAX) e soma do comprimento das seqüências ininterruptas de letras maiúsculas (CAPTOT).

– Doenças Cardíacas na África do Sul (DCAS): possui informações de 462 indivíduos homens com idades entre 15 e 64 anos, para as variáveis: pressão arterial (sbp); consumo de tabaco em *kg* (tobacco); colesterol ldl (ldl); índice de obesidade (obesity); consumo de álcool (alcohol); idade em anos (age) e a variável resposta binária corresponde a ocorrência ou não de infarto do miocárdio até a data da coleta dos dados. Dessa 462 observações, 362 foram selecionadas para a base *in sample* e 100 para *out-of-sample*.

– Fraude/Irregularidade no Consumo de Energia Elétrica: a empresa possui cerca de 452 mil clientes inspecionados em baixa tensão com perfis de consumo de energia diferentes, distribuídos em 2 regiões de estudo (Leste e Oeste). Essas regiões estão subdivididas e foi uma dessas subdivisões que selecionamos nesse exemplo. Ela possui 2430 clientes (*in sample*) e 2941(*out-of-sample*) que são classificados através de uma variável binária (indic_irregul_cod) com o valor 0 para os clientes normais e 1 para os supostos clientes irregulares. As demais variáveis são: consumo no mês (consumo), consumo no ano anterior (comsumo_ano_ant), consumo no ano base (consumo_ano_base), média 3 meses (media_3), média 6 meses (media_6), média dos meses 1 ao 12 (media_12), média dos meses 13 ao 24 (media_12_24), indicador trimestral 1_2 (indic_trimestral_1), indicador trimestral 2_3 (indic_trimestral_2), indicador anual (indic_anual), indicador ajuste (indic_ajuste), indicador tendência (indic_tendencia), temperatura mínima (temperatura_min), temperatura máxima (temperatura_max), carga.

Para cada um dos métodos analisados e comparados tentou-se ajustar o melhor modelo para cada um deles, confrontando as melhores classificações

que cada um resultou.

Vale ressaltar que em todas as comparações com o GAM, ajustamos o mesmo usando um *Suavizador Spline Cúbico* com 4 graus de liberdade para cada preditor e o método k-NN foi ajustado para um $k = 10$.

No Apêndice C as tabelas C.1, C.2, e C.3 apresentam os valores de algumas Estatísticas Descritivas das variáveis de cada uma das bases descritas anteriormente.

5.1.1 Aplicação: E-mail/Spam

Todas as 16 covariáveis selecionadas previamente foram colocadas como candidatas à variável de transição, fazendo parte do conjunto $\mathbf{x}$ e o conjunto de variáveis $\mathbf{z}$ foi composto por: our, over, remove, internet, free, business, hp, hpl, george, re, edu, char_! e char_$, CAPMAX e CAPTOT.

O ajuste a esses dados gerou uma árvore com 2 nós terminais e profundidade 1, contra um CART com 13 nós terminais e profundidades igual a 7. A figura 5.1 ilustra a estrutura do STLR-Tree para o ajuste final onde as pertinências de cada regime estão sendo mostradas.

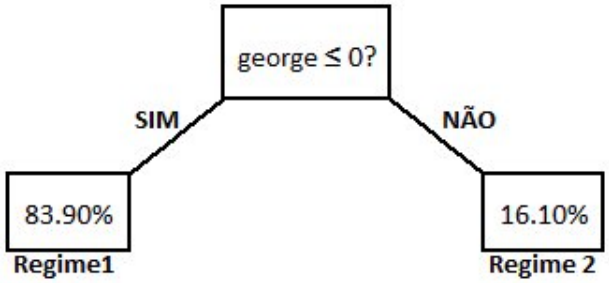

Figura 5.1: Estrutura do modelo - Spam

As equações referentes a cada um dos regimes encontrado são dadas por

$$\begin{aligned} Regime\ \ 1 = \ & (-0.39 - 3.15our - 10.77over + 9.87remove - 5.67internet + \\ + \ & 3.11free + 0.04business + 2.4hp + 1.8hpl - 2.88george - \\ - \ & 7.04remove - 14.61edu + 3.45char\_! + 19.25char\_\$ + \\ + \ & 1.53CAPMAX + 0.02CAPTOT) * G(george; 0, 8, 0, 1) \end{aligned}$$

$$\begin{aligned} Regime\ \ 2 = \ & (+150.51 + 784.57our - 658.04over + 7299.91remove + \\ + \ & 396.22internet + 48.4free - 21.48business - \\ - \ & 722.10hp + 148.85hpl + 379.96re - 38.98edu - \end{aligned}$$

$$- \quad 265.25char\_! - 955.31char\_\$ - 1.59CAPMAX +$$
$$+ \quad 2.3CAPTOT) * [1 - G(george; 0.8, 0.1)]$$

A seguir apresenta-se a tabela de classificação da análise *in sample*, tabela 5.1, e na seqüência a consolidação dos valores de sensitividade, especificidade e total de acertos, tabela 5.2, além da tabela com os métodos ordenados pelas taxas de acerto, tabela 5.3.

O STLR-Tree apresenta o segundo melhor desempenho para a taxa total de acertos 91.20%, ficando apenas atrás do GAM que obteve 95.20%.

Analisando a tabela 5.2 ressaltamos ainda as taxas de erro total (100% - taxa de acerto total) de classificação: GAM (4.80%), STLR-Tree (8.80%), Regressão Logística (8.95%), CART (10.05%), k-NN (28.85%) e Análise Discriminante (12.25%).

Tabela 5.1: Tabela de Classificação (*in sample*) - Spam

	Predito (ŷ)		
Observado (y)	0	1	
0	1142	70	1212
1	105	683	788
	1247	753	2000

Tabela 5.2: Comparação das Taxas de Acerto (*in sample*) - Spam

	Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
GAM	95.20%	92.26%	97.11%
STLR-Tree	91.25%	86.68%	94.22%
Reg. Logística	91.05%	86.80%	93.81%
CART	89.95%	80.46%	96.12%
k-NN	89.21%	81.32%	94.15%
Análise Discrim.	87.75%	78.17%	93.98%

Tabela 5.3: Métodos de Classificação Ordenados por Taxas de Acerto (*in sample*) - Spam

Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
GAM (0.952)	GAM (0.922)	GAM (0.971)
STLR-Tree (0.912)	Reg. Logística (0.86)	CART (0.961)
Reg. Logística (0.910)	STLR-Tree (0.866)	STLR-Tree (0.942)
CART (0.899)	k-NN (0.813)	k-NN (0.941)
k-NN (0.892)	CART (0.804)	Análise Discrim. (0.939)
Análise Discrim. (0.877)	Análise Discrim. (0.781)	Reg. Logística (0.938)

Nas tabelas seguintes temos as mesmas informações, porém para a análise *out-of-sample*, 5.4 e 5.5. Como esperado o valor das taxas diminui, mantendo a mesma ordem encontrada na análise (*in sample*) para a taxa total de acertos.

Tabela 5.4: Comparação das Taxas de Acerto (*out-of-sample*) - Spam

	Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
GAM	84,40%	83,65%	91,58%
STLR-Tree	85,70%	85,08%	91,58%
Reg. Logística	85,60%	84,86%	92,63%
K-NN	87,30%	87,29%	87,37%
Análise Discrim.	78,20%	76,91%	90,53%

Tabela 5.5: Métodos de Classificação Ordenados por Taxas de Acerto (*out-of-sample*) - Spam

Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
K-NN (0,873)	K-NN (0,873)	Reg. Logística (0,926)
STLR-Tree (0,857)	STLR-Tree (0,851)	STLR-Tree (0,916)
Reg. Logística (0,856)	Reg. Logística (0,849)	GAM (0,916)
GAM (0,844)	GAM (0,836)	Análise Discrim. (0,905)
Análise Discrim. (0,782)	Análise Discrim. (0,769)	K-NN (0,874)

A quantidade de parâmetros do STLR-Tree foi de 16 para cada nó terminal, que somados aos outros 2 parâmetros não-lineares, resultam em um total de r=34 parâmetros. Gam estimou 16 parâmetros lineares e a parte não-linear possui um parâmetro para cada observação. Regressão Logística e Análise Discriminante 15 cada.

A tabela D.1 do Apêndice D apresenta os parâmetros lineares de cada um dos métodos e a tabela D.5 os coeficientes dos parâmetros não-lineares do STLR-Tree.

5.1.2 Aplicação: Doenças Cardíacas na África do Sul (DCAS)

No modelo selecionado o conjunto de variáveis **z** é composto por: sbp, tobacco, ldl, alcohol e age. Seus resultados são mostrados a seguir.

O ajuste a esses dados gerou um modelo com 2 nós terminais e uma profundidade, contra um CART com 13 nós terminais e profundidades igual a 7, figura 5.2.

Os modelos encontrados foram

$$Regime \quad 1 = \qquad (-288.658 + 2.477sbp + 2.293tobacco -$$

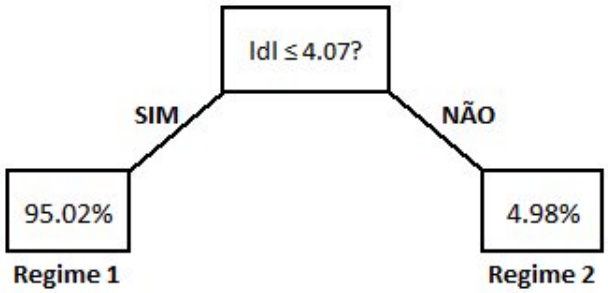

Figura 5.2: Estrutura do modelo - DCAS

$$\begin{aligned} & - \quad 6.156obesity - 1.255alcohol + \\ & + \quad 2.937age) * G(ldl; 4.07, 50) \end{aligned}$$

$$\begin{aligned} Regime \quad 2 = \quad & (-2.743 - 0.006sbp + 0.083tobacco + \\ & + \quad 0.008obesity + 0.005alcohol + \\ & + \quad 0.05age) * [1 - G(ldl; 4.07, 50)] \end{aligned}$$

Na tabela 5.6 é mostrada a tabela de classificação para os dados desta base (*in sample*).

Tabela 5.6: Tabela de Classificação (*in sample*) - DCAS

Observado (y)	Predito (ŷ)		
	0	1	
0	268	34	302
1	85	75	160
	353	109	462

As duas tabelas seguintes, 5.7 e 5.8, pode ser verificado que, para esta base, as taxas de classificação de todos os métodos não foram tão eficientes quanto aquelas apresentadas para a base anterior. Estas tabelas se referem à análise *in sample*.

O STLR-Tree apresenta a terceira melhor taxa de acerto total com 72.65%. Os dois métodos que melhora classificaram foram GAM (82.60%) e CART (75.14%).

Com a tabela 5.7 podemos concluir que as taxas de erro total foram razoavelmente altas para todos os métodos: GAM (17.40%), STLR-Tree (27.35%), CART (24.86%), Regressão Logística (28.73%), Análise Discriminante (32.32%) e k-NN (29.87%).

As mesmas comparações anteriores foram feitas para a base *out-of-sample*, 5.9 e 5.10.

Tabela 5.7: Comparação das Taxas de Acerto (*in sample*) - DCAS

	Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
GAM	78.35%	60.63%	87.75%
STLR-Tree	74.24%	46.88%	88.74%
CART	72.94%	50.63%	84.77%
Reg. Logística	70.78%	47.50%	83.11%
Análise Discrim.	69.05%	71.88%	67.55%
K-NN	66.23%	56.52%	70.37%

Tabela 5.8: Métodos de Classificação Ordenados por Taxas de Acerto (*in sample*) - DCAS

Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
GAM (0.783)	Análise Discrim. (0.718)	STLR-Tree (0.887)
STLR-Tree (0.742)	GAM (0.606)	GAM (0.877)
CART (0.729)	K-NN (0.565)	CART (0.847)
Reg. Logística (0.707)	CART (0.506)	Reg. Logística (0.831)
Análise Discrim. (0.690)	Reg. Logística (0.47)	K-NN (0.703)
K-NN (0.662)	STLR-Tree (0.468)	Análise Discrim. (0.675)

Como na aplicação anterior, a ordem de classificação que diz respeito a taxa total de acertos, se manteve, porém com diminuição dos valores percentuais.

Tabela 5.9: Comparação das Taxas de Acerto (*out-of-sample*) - DCAS

	Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
GAM	77,00%	51,72%	87,32%
STLR-Tree	71,00%	55,17%	77,46%
Reg. Logística	74,00%	41,38%	87,32%
Análise Discrim.	69,00%	72,41%	67,61%
K-NN	70,00%	55,17%	76,06%

Tabela 5.10: Métodos de Classificação Ordenados por Taxas de Acerto (*out-of-sample*) - DCAS

Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
GAM (0,77)	Análise Discrim. (0,724)	GAM (0,873)
Reg. Logística (0,74)	STLR-Tree (0,552)	Reg. Logística (0,873)
STLR-Tree (0,71)	K-NN (0,552)	STLR-Tree (0,775)
K-NN (0,7)	GAM (0,517)	K-NN (0,761)
Análise Discrim. (0,69)	Reg. Logística (0,414)	Análise Discrim. (0,676)

A quantidade de parâmetros em cada um dos dois nós terminais foi 6, assim o STLR-Tree tem 12 parâmetros lineares além de mais 2 não-lineares, em um total de r=14 parâmetros. O número de parâmetros lineares estimados pelo

GAM foi 6 e, como dito anteriormente, o número de parâmetros não lineares é igual ao número de observações de cada variável. Em Regressão Logística o número encontrado foi igual ao encontrado na parte linear do GAM, 6. Já Análise Discriminante estimou um total de 5. Na tabela D.2 do Apêndice D podemos verificar seus valores. Os coeficientes não-lineares do STLR-Tree são apresentados na tabela D.5.

5.1.3
Aplicação: Fraude/Irregularidade no Consumo de Energia Elétrica

Como este foi o único exemplo em que os métodos comparados incluíram Redes Neurais cabe fazer algumas observações quanto à sua programação.

No estudo original sobre as fraudes e irregularidades do setor Elétrico, encontrado em (25), o autor, visando um melhor treinamento das Redes Neurais, dividiu em cinco bases de treinamento/validação para o aprendizado (ou ajuste) do modelo e um arquivo de teste. Os dados relativos ao treinamento/validação foram coletados de dois períodos distintos: de janeiro de 2001 a dezembro de 2005 e os meses de março a junho de 2006, a fim de avaliar períodos distintos (verão e inverno de 2006).

Através de um comitê, onde foram gerados cinco modelos a partir das cinco bases de treinamento, onde cada um deles foi testado com a base de teste. Essas cinco redes treinadas possuíam uma camada de 8 neurônios escondidos.

Dentre elas, a que obteve a melhor classificação foi a utilizada para fazer as comparações deste trabalho.

O modelo para os dados de Fraude/Irregularidade tem seus resultados mostrados a seguir, com a mesma seqüência de figura e tabelas mostrada nos modelos anteriores. Nele o conjunto de variáveis $\mathbf{z}$ contém todas as variáveis de $\mathbf{x}$ menos a indic_trimestral_1 e a indic_trimestral_2.

Sua estrutura foi a maior dentre todas as estruturas dos demais exemplos, tendo 5 nós terminais e profundidade igual a 4, figura 5.3. A estrutura do CART tem 4 nós terminais e profundidade 3.

Para cada regime abaixo está descrito a equação dos modelos

$$\begin{aligned}
Regime \quad 1 = \quad & (-2.38 + 6.4consumo - 1.48comsumo_ano_ant + \\
+ \quad & 35.2consumo_ano_base + 0.28media_3 + 0.14media_6 - \\
- \quad & 0.78media_12 + 0.03media_12_24 - 1.51indic_anual - \\
- \quad & 0.55indic_ajuste + 2.79indic_tendencia - \\
- \quad & 1.62temperatura_min + 0.95temperatura_max + \\
+ \quad & 0.59carga) * [1 - G(temperatura_min; 3.54, 10.21)]
\end{aligned}$$

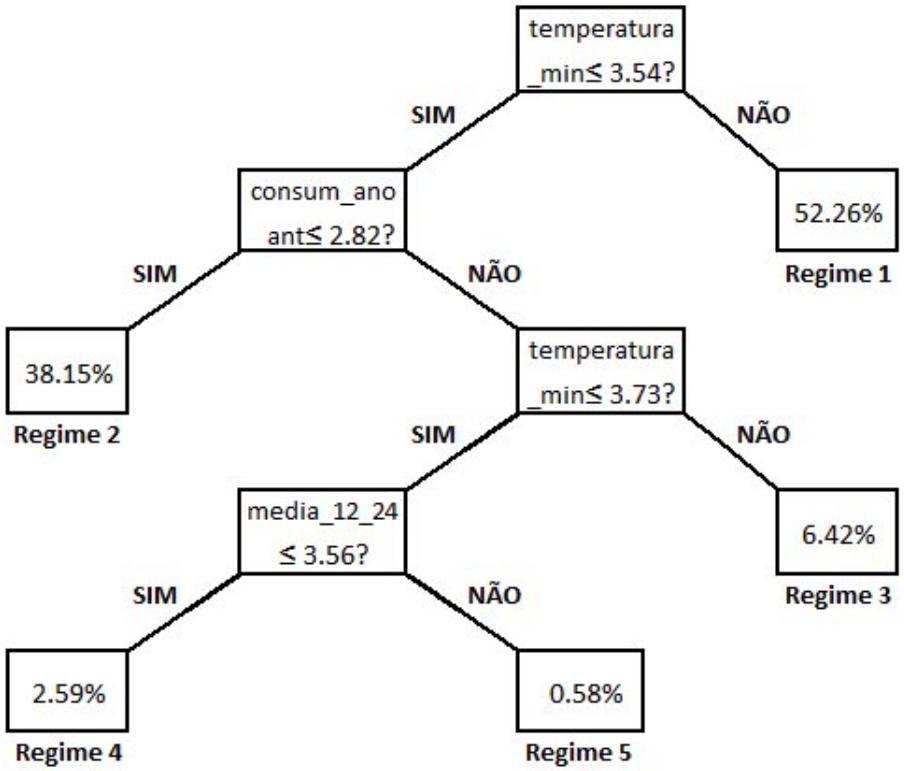

Figura 5.3: Estrutura do modelo - Consumo de Energia

$$
\begin{aligned}
Regime\ \ 2 = \quad & (-0.54 + 33.76 consumo - 23.65 comsumo\_ano\_ant - \\
- \quad & 6.64 consumo\_ano\_base + 0.46 media\_3 - \\
- \quad & 7.7 media\_6 - 0.12 media\_12 - 1.54 media\_12\_24 + \\
+ \quad & 0.15 indic\_anual + 3.56 indic\_ajuste - \\
- \quad & 4.65 indic\_tendencia + 3.62 temperatura\_min - \\
- \quad & 0.59 temperatura\_max + 0.74 carga) * \\
* \quad & G(comsumo\_ano\_ant; 2.82, 50) * \\
* \quad & G(temperatura\_min; 3.54, 10)
\end{aligned}
$$

$$
\begin{aligned}
Regime\ \ 3 = \quad & (-3.86 + 7.82 consumo - 1.65 comsumo\_ano\_ant + \\
+ \quad & 1.03 consumo\_ano\_base - 0.11 media\_3 - 0.03 media\_6 - \\
- \quad & 3.32 media\_12 + 6.5 media\_12\_24 - 5.74 indic\_anual + \\
+ \quad & 1.96 indic\_ajuste - 0.92 indic\_tendencia + \\
+ \quad & 0.6 temperatura\_min - 0.36 temperatura\_max - 4.7 carga) * \\
* \quad & [1 - G(temperatura\_min; 3.73, 50)] *
\end{aligned}
$$

$$
\begin{aligned}
& * \quad [1 - G(comsumo\_ano\_ant; 2.82, 50)] * \\
& * \quad G(temperatura\_min; 3.54, 10.21)
\end{aligned}
$$

$$
\begin{aligned}
Regime \quad 4 = \quad & (33.85 - 35.36consumo - 32.45comsumo\_ano\_ant + \\
& + \quad 15.22consumo\_ano\_base + 0.98media\_3 - 19.19media\_6 + \\
& + \quad 6.09media\_12 + 40.83media\_12\_24 - 75.25indic\_anual + \\
& + \quad 35.18indic\_ajuste - 25.79indic\_tendencia - \\
& - \quad 18temperatura\_min - 29.1temperatura\_max + 56.38carga) * \\
& * \quad G(media\_12\_24; 3.56, 100) * G(temperatura\_min; 3.73, 50) * \\
& * \quad [1 - G(comsumo\_ano\_ant; 2.82, 50)] * G(temperatura\_min; 3.54, 10)
\end{aligned}
$$

$$
\begin{aligned}
Regime \quad 5 = \quad & (-16.84 - 0.69consumo - 0.23comsumo\_ano\_ant - \\
& - \quad 0.01consumo\_ano\_base + 0.98media\_3 - 2.61media\_6 + \\
& + \quad 1.51media\_12 - 0.06media\_12\_24 - 0.72indic\_anual - \\
& - \quad 0.24indic\_ajuste - 0.15indic\_tendencia + \\
& + \quad 1.9temperatura\_min + 0.34temperatura\_max - 1.23carga) * \\
& * \quad [1 - G(media\_12\_24; 3.56, 100)] * G(temperatura\_min; 3.73, 50) * \\
& * \quad [1 - G(comsumo\_ano\_ant; 2.82, 50)] * G(temperatura\_min; 3.54, 10)
\end{aligned}
$$

A seguir está a tabela de classificação, tabela 5.11.

Tabela 5.11: Tabela de Classificação (*in sample*) - Consumo de Energia

Observado (y)	Predito (ŷ) 0	1	
0	775	440	1215
1	397	818	1215
	1172	1258	2430

Nas tabelas 5.12 e 5.13 (análise *in sample*) podemos verificar o bom desempenho do modelo na taxa de acertos total, ficando com um percentual de 68.48%, estando abaixo apenas de Redes Neurais (71.77%). Aqui GAM, que estava sempre com um melhor desempenho, detém o terceiro melhor resultado para a referida taxa, 67.74%.

Analisando ainda a tabela 5.12, nota-se que as taxas de erro total de classificação de todos os métodos foram: Redes Neurais (28.23%),

GAM (32.26%), STLR-Tree (31.52%), Análise Discriminante (39.84%), CART (40.08%), Regressão Logística (40.21%), e k-NN (48.85%).

Tabela 5.12: Comparação das Taxas de Acerto (*in sample*) - Fraude no Consumo de Energia Elétrica

	Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
Redes Neurais	71.77%	69.38%	74.16%
GAM	66.58%	64.20%	68.97%
STLR-Tree	65.56%	67.33%	63.79%
Análise Discrim.	60.16%	58.85%	61.48%
CART	59.92%	29.71%	90.12%
Reg. Logística	59.79%	57.20%	62.39%
K-NN	54.94%	56.90%	53.41%

Tabela 5.13: Métodos de Classificação Ordenados por Taxas de Acerto (*in sample*) - Fraude no Consumo de Energia

Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
Redes Neurais (0.717)	Redes Neurais (0.693)	CART (0.901)
GAM (0.665)	STLR-Tree (0.673)	Redes Neurais (0.741)
STLR-Tree (0.655)	GAM (0.642)	GAM (0.689)
Análise Discrim. (0.601)	Análise Discrim. (0.588)	STLR-Tree (0.637)
CART (0.599)	Reg. Logística (0.57)	Reg. Logística (0.623)
Reg. Logística (0.597)	K-NN (0.56)	Análise Discrim. (0.614)
K-NN (0.549)	CART (0.297)	K-NN (0.534)

A análise *out-of-sample* aplicada aos dados de consumo de energia, consta nas tabelas a seguir, 5.14 e 5.15. O STLR-Tree tem um desempenho não tão satisfatório quanto o apresentado para a base *in sample*, tendo sua taxa de acertos total caindo de 68.48% para 56.81%.

Tabela 5.14: Comparação das Taxas de Acerto (*out-of-sample*) - Fraude no Consumo de Energia Elétrica

	Tx. de acertos Total	Tx. de acertos para 1 (Sensitividade)	Tx. de acertos para 0 (Especificidade)
GAM	55,19%	55,25%	55,15%
STLR-Tree	56,81%	40,63%	68,70%
Redes Neurais	59,54%	46,47%	66,77%
Reg. Logística	52,98%	40,36%	59,96%
Análise Discrim.	57,43%	55,92%	58,27%
K-NN	50,80%	65,46%	42,68%

Em cada nó terminal foram estimados 14 parâmetros totalizando 70 coeficientes calculados através do STLR-Tree para sua parte linear. Com

Tabela 5.15: Métodos de Classificação Ordenados por Taxas de Acerto (*out-of-sample*) - Fraude no Consumo de Energia Elétrica

Tx. de acertos Total	**Tx. de acertos para 1 (Sensitividade)**	**Tx. de acertos para 0 (Especificidade)**
Redes Neurais (0,595)	K-NN (0,655)	STLR-Tree (0,687)
Análise Discrim. (0,574)	Análise Discrim. (0,559)	Redes Neurais (0,668)
STLR-Tree (0,568)	GAM (0,552)	Reg. Logística (0,6)
GAM (0,552)	Redes Neurais (0,465)	Análise Discrim. (0,583)
Reg. Logística (0,53)	STLR-Tree (0,406)	GAM (0,552)
K-NN (0,508)	Reg. Logística (0,404)	K-NN (0,427)

esses tivemos os outros 8 parâmetros não lineares, que, juntos, somam 78 parâmetros no total. GAM tem 13 na parte linear, Regressão Logística 9 e Análise Discriminante 15, como pode ser visto na tabela D.3 do Apêndice D. No mesmo apêndice, tabela D.5, encontram-se os valores dos coeficientes não-lineares do STLR-Tree.

Os pesos das variáveis em cada um dos 8 neurônios da camada oculta calculados pela Rede Neural em um total de 128 valores estão na tabela D.4 do mesmo apêndice.

Para ilustrar o que foi mostrado anteriormente, as figuras 5.4, 5.5 e 5.6 a seguir representam os valores das taxas de erro total de cada um dos métodos comparados, que foram plotadas para cada um dos exemplos onde se pode ver claramente a posição do modelo STLR-Tree em relação aos demais.

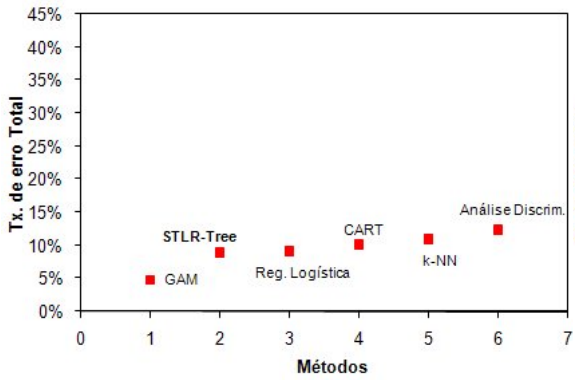

Figura 5.4: Gráfico das taxas de erro total - E-mail/Spam

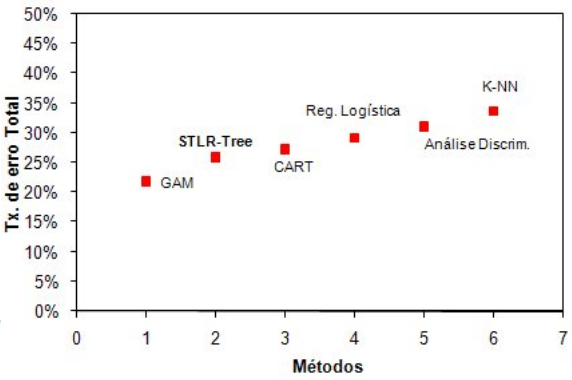

Figura 5.5: Gráfico das taxas de erro total - DCAS

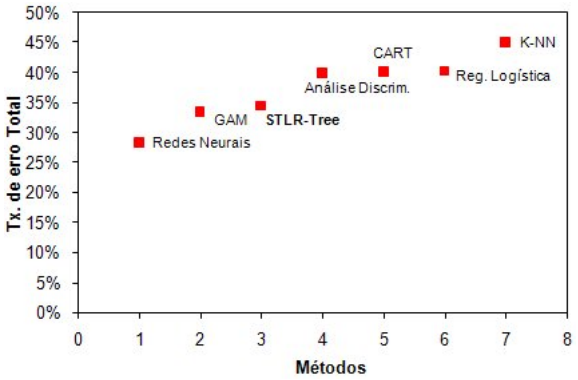

Figura 5.6: Gráfico das taxas de erro total - Fraude no Consumo de Energia Elétrica

6
Conclusão

No presente trabalho, foram comparadas as seguintes metodologias utilizadas para classificação: *Generalized Additive Models (GAM), Regressão Logística, Redes Neurais, Análise Discriminante, k-Vizinhos mais Próximos (k-NN)* e *Classification and Regression Trees (CART)*, com o modelo STLR-Tree, o qual é uma adaptação do modelo STR-Tree aplicado na classificação de variáveis binárias.

Para este fim, algumas alterações foram realizadas em sua forma estrutural e na estimação. A parte linear passou a ser estimada por Máxima Verossimilhança, através do método iterativo de Newton-Raphson, tal como em uma Regressão Logística e seus parâmetros não-lineares são estimados através de uma procura em *grid*. Os testes Multiplicadores de Lagrange, que determinam a inclusão de novos regimes no modelo, eram realizados através de estatísticas de teste onde se calculava a soma dos quadrados dos resíduos no caso Gaussiano, passou a ser feita a ser feito com o cálculo da função desvio, que é o equivalente para o caso não Gaussiano, onde as variáveis dependentes pertençam à família exponencial. Essas mudanças se devem ao fato do modelo STLR-Tree não apresentar uma estrutura sistemática do erro, dado a distribuição de sua variável dependente.

Com as duas bases de dados utilizadas, selecionou-se dois conjuntos de variáveis explicativas de cada uma delas. Esses conjuntos foram escolhidos pois, na avaliação das tabelas de classificação, apresentaram os melhores resultados. Sendo assim, os demais métodos foram aplicados a esses quatro conjuntos de variáveis.

Os resultados obtidos pelo STLR-Tree foram bastante satisfatórios, sendo ele detentor da segunda melhor taxa de acerto total nas aplicações: *Spam* e *Consumo de Energia*, na análise *in sample*. Ambos provavelmente conseguiram captar uma relação não-linear entre as variáveis que os demais não fizeram. Na aplicação *DCAS* o modelo ficou atrás apenas atrás de GAM e CART para a mesma taxa. Atentamos para o fato dos resultados do exemplo *Fraude no Consumo de Energia Elétrica* terem sido trabalhados e ajustados especificamente em um estudo sobre Redes Neurais.

Na análise *out-of-sample*, o modelo se manteve na mesma posição em comparação aos demais métodos nas aplicações *Spam* e *DCAS*, porém, para no *Fraude no Consumo de Energia Elétrica* acabou sendo ultrapassado poucos pontos percentuais por Análise Discriminante.

Além disso, o modelo se apresentou bastante parcimonioso nos dois primeiros exemplos, não passando de dois nós terminais na estrutura das árvores. Como comparação, temos o CART que se estruturou com 13 nós terminais para os exemplos da base *Spam* e na base *DCAS*. Já no último exemplo, *Fraude no Consumo de Energia Elétrica*, o número de nós terminais do STLR-Tree foi maior do que o CART, 5 e 4 nós respectivamente. Entretanto o primeiro foi bastante superior em duas das três taxas de acertos avaliadas (total, para 1 (sucesso), obtendo um acerto total de 68.48% contra 59.92%.

Concluímos com isto, que o modelo se adaptou bem as alterações realizadas, mostrando-se uma alternativa em análises não-lineares a ser utilizada para classificação. Porém deve ainda ter seu desempenho computacional melhorado, objetivando minimizar o tempo de duração de suas aplicações e, além disso, ser adaptado para classificação não apenas binária. Neste caso a distribuição da variável dependente não é Binomial, e sim *Multinomial*.

Referências Bibliográficas

[1] BOX, G. E. P.; JENKINS, G. M. **Time Series Analysis: Forcasting and Control**. San Francisco: Holden-Day, San Francisco, revised edition, 1976.

[2] BREIMAN, L.; FRIEDMAN, J.; OLSHEN, R. ; STONE, C. J. **Classification and Regression Trees**. Wadsworth and Brooks, Monterey, CA, 1984.

[3] CHAN, K. S.; TONG, H. **On estimating threshols in autoregressive models**. J. Times Series Anal., 7:179–190, 1986.

[4] CHANDHURI, P.; LO, W.; LOH, W. Y. ; YANG, C. C. **Generalized regression trees**. Statistica Sinica, 5:641–666, 1995.

[5] CIAMPI, A. **Generalized regression trees**. Computational Statistics and Data Analysis, 12:57–78, 1991.

[6] COVER, T. M.; HART, P. E. **Nearest neighbor pattern classification**. IEEE Transactions on Information Theory, 13:21–27, 1967.

[7] DA ROSA, J. C.; VEIGA, A. ; MEDEIROS, M. C. **Tree-structured smooth transition regression models**. Computational Statistics and Data Analysis, 52:2469–2488, 2008.

[8] FAN, J.; YAO, Q. **Nonlinear Time Series: Nonparametric and Parametric Methods**. New York, 2003.

[9] FIX, E.; HODGES, J. L. **Nonparametric discrimination: Consistency properties**. Randolph Field, Texas, USA, 1951.

[10] GRANGER, C.; TERÄSVIRTA, T. **Modelling Nonlinear Economic Relationships**. Oxford, UK, 1993.

[11] GUISAN, A.; EDWARDS, T. ; HASTIE, T. **Generalized linear and generalized additive models in studies of species distributions: setting the scene**. Ecological Modeling, 157:89–100, 2002.

[12] HASTIE, T.; TIBSHIRANI, R. **Generalized additive models**. Statistical Science, 1(3):297–318, 1986.

[13] HASTIE, T.; TIBSHIRANI, R. **Generalized additive models**. London, UK, 1990.

[14] HASTIE, T.; TIBSHIRANI, R. ; FRIEDMAN, J. **The Elements of Statistical Learning: Data Mining, Inference, and Prediction**. New York, USA, 2001.

[15] HOSMER, D.; LEMESHOW, S. **Applied Logistic Regression**. New York, USA, 2nd edition, 2000.

[16] JOHNSON, R. A.; WICHERN, D. W. **Applied Multivariate Statistical Analysis**. London, UK, 5th edition, 2002.

[17] LEWIS, P. A. W.; STEVENS, J. G. **Nonlinear modelong of time series using multivariate adaptative regression splines**. Journal of American Statistical Association, p. 864–877, 1986.

[18] LUUKKONEN, R.; SAIKKONEN, P. ; TERÄSVIRTA, T. **Testing linearity againt smooth transition autoregressive models**. Biometrika, 75:491–499, 1988.

[19] MCCULLAGH, P.; NELDER, J. A. **Generalized Linear Models**. London, UK, 2nd edition, 1989.

[20] MEDEIROS, M.; VEIGA, A. **Diagnostic checking in a flexible nonlinear time series model**. Journal of Time Series Analysis, 24:461–482, 2003.

[21] MEDEIROS, M. C.; VEIGA, A. **A hybrid linear-neural model for time series forecasting**. IEEE Transactions on Neural Networks, 11(6):1402–1412, 2000.

[22] MEDEIROS, M. C.; VEIGA, A. **A flexible coefficient smooth transition time series models**. IEEE Transactions on Neural Networks, 16(1):97–113, 2005.

[23] MINGOTI, S. A. **Análise de Dados Através de Métodos de Estatística Multivariada**. Belo Horizonte, BR, 2005.

[24] NELDER, J. A.; WEDDERBURN, R. W. M. **Generalized linear models**. Journal of the Royal Statistical Society. Series A (General), 135(3):370–384, 1972.

[25] ORTEGA, G. V. C. **Redes neurais na identificação de perdas comerciais do setor elétrico**. Dissertação de Mestrado, Departamento de Engenharia Elétrica - PUC-Rio, Rio de Janeiro, Brasil, 2008.

[26] RECH, G.; TERÄSVIRTA, T. ; TSCHERNING, R. **A simple variable selection technique for nonlinear models**. Commun. Statist., Theory and Methods, 30:1227–1241, 2001.

[27] STONE, C. **Additive regression and other nonparametric models**. Ann. Statist, 13:689–705, 1985.

[28] TONG, H. **On a threshold model**. 1978. Leyden, Netherlands: Sijthoff and Noordhoff, 1978.

[29] TONG, H. **Non-linear Time Series: A Dynamical System Approach**, volumen 6. Oxford Statistical Science Series, Oxford, 1990.

[30] TONG, H.; LIM, K. **Threshold autoregression, limit cycles and cyclical data**. Royal Statistical Society, Series B, Methodological, (42):245–292, 1980.

[31] VAN DIJK, D.; FRANSES, P. **Modelling multiple regimes in the business cycle**. Macroeconomic Dynamics, 3:311–340, 1999.

[32] VAN DIJK, D.; TERÄSVIRTA, T. ; FRANSES, P. H. **Smooth transition autoregressive models - a survey of recent developments**. Econometric Rev., 21:1–47, 2002.

[33] WHITE, H. **A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity**. Econometrica, 48(4):817–838, 1980.

[34] YULE, G. U. **On a method of investigating periodicities in disturbed series with special reference to wolfer´s sunspot numbers**. Philosophical Transactions of th Royal Society, 226:267–298, 1927.

A
Alguns Modelos Não-lineares

Serão brevemente explicados alguns dos modelos não-lineares existentes na literatura, muitos deles utilizado na análise de séries temporais não enquadrados no contexto de classificação, porém foram escolhidos aqueles que, de alguma maneira, estão relacionados com este trabalho em sua forma estruturas, estimação e/ou previsão.

A.1
Threshold Auto Regressive (TAR)

Proposto em (28) e mais tarde desenvolvido em (30) e extensamente discutido em (29).

A idéia principal do modelo TAR é a de mudar os parâmetros de um modelo linear Auto Regressivo (AR) (ver (34) e (1)), de acordo com o valor de uma variável observável chamada variável de transição ou limiar (do inglês *threshold variable*).

Basicamente fazem uma divisão do espaço Euclidiano unidimensional de modo a obter *L* regimes, os quais são liderados por um modelo Auto Regressivo de odem $k_i, \quad i = 1, ..., L$. Tal divisão é feita de forma abrupta sendo regida por uma função indicadora, $I(z_t)$.

Formulação Matemática

$$y_t = \begin{cases} \beta_0^{(1)} + \sum_{i=1}^{k_1} \beta_i^{(1)} y_{t-i} + \epsilon_t^{(1)} & , \; se \; z_t \in \mathbb{R}_1 \\ \beta_0^{(2)} + \sum_{i=1}^{k_2} \beta_i^{(2)} y_{t-i} + \epsilon_t^{(2)} & , \; se \; z_t \in \mathbb{R}_2 \\ \vdots & \\ \beta_0^{(L)} + \sum_{i=1}^{k_L} \beta_i^{(L)} y_{t-i} + \epsilon_t^{(L)} & , \; se \; z_t \in \mathbb{R}_L \end{cases}$$

$$y_t = \sum_{j=1}^{L} \left[\beta_0^{(j)} + \sum_{i=1}^{p} \beta_i^{(j)} y_{t-i} + \epsilon_t^{(j)} \right] I^{(j)}(z_t) \qquad \text{(A-1)}$$

onde z_t é a variável de transição, $\epsilon_t^{(j)} \rightarrow (0, \sigma^2)$ o erro aleatório (ruído branco) e o vetor de parâmetros lineares $\boldsymbol{\beta} = (\beta_0^{(j)}, \beta_1^{(j)}, ..., \beta_p^{(j)})'$. A função indicadora é tal

que

$$I^{(j)}(z_t) = \begin{cases} 1 & , se\ z_t \in \mathbb{R}_j \\ 0 & , se\ z_t \notin \mathbb{R}_j. \end{cases}$$

A variável de transição pode ser governada pelo tempo ($z_t = t$), por uma variável exógena ($z_t = x_{t-d}$) ou ainda por um valor defasado da variável dependente, ou seja, um auto-regressor de y_t ($z_t = y_{t-d}$). A letra d representa o parâmetro de defasagem.

A.2
Self-Exiting Threshold Auto Regressive (SETAR)

A escolha da variável de transição como um auto-regressor de y_t caracteriza um modelo SETAR (ver (29)), o qual, da mesma forma que o TAR, divide o espaço das variáveis de forma abrupta, em subespaços ortogonais a somente um auto-regressor, y_{t-d}.

Formulação Matemática

$$y_t = \begin{cases} \beta_0^{(1)} + \sum_{i=1}^{k_1} \beta_i^{(1)} y_{t-i} + \epsilon_t^{(1)} & , se\ y_{t-d} \in \mathbb{R}_1 \\ \beta_0^{(2)} + \sum_{i=1}^{k_2} \beta_i^{(2)} y_{t-i} + \epsilon_t^{(2)} & , se\ y_{t-d} \in \mathbb{R}_2 \\ \vdots & \\ \beta_0^{(L)} + \sum_{i=1}^{k_L} \beta_i^{(L)} y_{t-i} + \epsilon_t^{(L)} & , se\ y_{t-d} \in \mathbb{R}_L \end{cases}$$

$$y_t = \sum_{j=1}^{L} \left[\beta_0^{(j)} + \sum_{i=1}^{p} \beta_i^{(j)} y_{t-i} + \epsilon_t^{(j)} \right] I^{(j)}(y_{t-d}) \qquad \text{(A-2)}$$

onde y_{t-d} é a variável de transição e

$$I^{(j)}(y_{t-d}) = \begin{cases} 1 & , se\ y_{t-d} \in \mathbb{R}_j \\ 0 & , se\ y_{t-d} \notin \mathbb{R}_j. \end{cases}$$

A.3
Smooth Transition Autoregression (STAR)

Uma alteração no modelo SETAR proposta por (3), onde passamos de uma transição abrupta para uma transição suave, substituindo a função indicadora por uma função não-linear, contínua e limitada entre 0 e 1 denominada por $G(z_t; \gamma, c)$. Tais modelos limitam-se a dois regimes apenas.

Formulação Matemática

$$y_t = \beta_0^{(1)} + \beta_1^{(1)} y_{t-1} + ... + \beta_{k1}^{(1)} y_{t-k_1} + \left(\beta_0^{(2)} + \beta_1^{(2)} y_{t-1} + ... + \beta_{k1}^{(2)} y_{t-k_2}\right) G(z_t; \gamma, c) + \epsilon_t$$

$$y_t = \beta_0^{(1)} + \sum_{i=1}^{k_1} \beta_i^{(1)} y_{t-i} + \left(\beta_0^{(2)} + \sum_{j=1}^{p} \beta_j^{(2)} y_{t-j}\right) G(z_t; \gamma, c) + \epsilon_t$$

A formulação para dois regimes pode ser expressa, quando $z_t = y_{t-1}$, por

$$y_t = (\alpha_0 + \beta_0 y_{t-1}) G(y_{t-1}; \gamma, c) + (\alpha_1 + \beta_1 y_{t-1})[1 - G(y_{t-1}; \gamma, c)] + \epsilon_t$$

onde $G(y_{t-1}; \gamma, c)$ é a função de transição, γ é chamado de parâmetro de suavização e c de parâmetro de localização ou limiar.

Especificação da Função de Transição

Nos modelos de transição suave pode-se especificar a função de transição de modo a modelar os dados sem assumir inicialmente que haverá uma mudança abrupta entre os regimes. As funções podem ser escolhidas como, por exemplo:

- Função Logística: $G(z_t; \gamma, c) = \frac{e^{-\gamma(z_t - c)}}{1 + e^{-\gamma(z_t - c)}}, \quad \gamma > 0$;
- Função Exponencial: $G(z_t; \gamma, c) = 1 - e^{-\gamma(z_t - c)}, \quad \gamma > 0$.

Dependendo da função de transição e dos valores do parâmetro de suavização da mesma, o STAR é definido de formas diferentes.

A.4
Logistic Smooth Transition Autoregression (LSTAR)

Quando a função de transição utilizada para suavizar a mudança entre os regimes for a função logística, trata-se do modelo LSTAR (ver (18)).

A variação no valor do parâmetro do grau de suavidade da função, γ, remete a casos particulares do modelo LSTAR.

Se $\gamma \to \infty$ a função de transição toma a forma de uma função degrau, ou seja

$$G(z_t; \gamma, c) = \begin{cases} 1 & , se\ z_t \leq c \\ 0 & , se\ z_t > c \end{cases}$$

Esta situação caracteriza um modelo TAR em que o limiar é regido e determinado por c. Caso $z_t = c$ a observação fará parte de ambos os regimes com o mesmo grau de pertinência.

No caso em que $\gamma \to 0$ a função logística é igual a $0,5$ e teremos um processo Auto Regressivo (AR).

A.5
Exponencial Smooth Transition Autoregression (ESTAR)

Ao utilizarmos uma função de transição exponencial teremos um modelo denominado ESTAR (ver (10)).

Neste, tanto para valores de $\gamma \to \infty$ quanto para $\gamma \to 0$, teremos um modelo AR.

A.6
Multiple Regime Smooth Transition Autoregression (MRSTAR)

Como o modelo STAR abrange somente até dois regimes, vemos em (31) o tratamento da multiplicidade de regimes através do modelo MRSTAR.

Formulação Matemática

Para um modelo com 4 regimes, por exemplo, teremos a representação de um MRSTAR onde $z_t = y_{t-1}$ dada por

$$\begin{aligned} y_t = \quad & \{(\alpha_0 + \beta_0 y_{t-1})G_0(y_{t-1};\gamma,c) + (\alpha_1 + \beta_1 y_{t-1})[1 - G_0(y_{t-1};\gamma,c)]\} \\ & + \{(\alpha_2 + \beta_2 y_{t-1})G_1(y_{t-1};\gamma,c) + (\alpha_3 + \beta_3 y_{t-1})[1 - G_1(y_{t-1};\gamma,c)]\} + \epsilon_t \end{aligned}$$

A.7
Neural Coefficient Smooth Transition Autoregressive (NCSTAR)

Ainda no contexto dos modelos de múltiplos regimes cabe destacar aquele proposto em (21). Trata-se de um modelo híbrido, pois mescla os parâmetros autoregressivos de um STAR, os quais variam ao longo do tempo conforme a saída de uma Rede Neural Artificial (RNA).

O modelo STR-Tree, principal enfoque deste trabalho, tem seus processos de especificação e estimação muito semelhantes ao que é feito no caso do NCSTAR.

As técnicas de diagnóstico do ajuste de um NCSTAR encontram-se em (20) e o tema volta a ser abordado em (22).

Formulação Matemática

$$y_t \quad = G(\mathbf{z}_t, \mathbf{x}_t; \boldsymbol{\psi}) + \epsilon_t$$

$$= \alpha_0 + \sum_{j=1}^{p} \alpha_j y_{t-j} + \sum_{i=1}^{h} \lambda_{0i} F(\boldsymbol{\omega}_i' \mathbf{x}_t - c_i)$$
$$+ \sum_{j=1}^{p} \left\{ \sum_{i=1}^{h} \lambda_{ji} F(\boldsymbol{\omega}_i' \mathbf{x}_t - c_i) \right\} y_{t-j} + \epsilon_t$$

que tem a forma vetorial dada

$$y_t = \boldsymbol{\alpha}' \mathbf{z}_t + \sum_{i=1}^{h} \boldsymbol{\lambda}' \mathbf{z}_t F(\boldsymbol{\omega}_i' \mathbf{x}_t - c_i),$$

onde $\boldsymbol{\psi} = (\boldsymbol{\alpha}', \boldsymbol{\lambda}_1', ..., \boldsymbol{\lambda}_h', \boldsymbol{\omega}_1', ..., \boldsymbol{\omega}_h', c_i, ..., c_h)' \in \mathbb{R}^r$ é o vetor de parâmetros, $r = (q+1)h + (p+1)(h+1)$, $\boldsymbol{\alpha} = (\alpha_0, ..., \alpha_p)' = (-\lambda_{00}, ..., \lambda_{p0})'$ e $\boldsymbol{\lambda}_i = (\lambda_{0i}, ..., \lambda_{pi})'$. $F(\boldsymbol{\omega}_j' \mathbf{x}_t - c_i)$ é a função de transição logística onde $\mathbf{x}_t \in \mathbb{R}^q$ é o vetor de variáveis de transição, $\boldsymbol{\omega}_i = (\omega 1i, ..., \omega_{qi})' \in \mathbb{R}^q$ e $c_i \in \mathbb{R}$ são os parâmetros não-lineares.

A.8
Smooth Transition Regression (STR)

Semelhante ao STAR onde a transição não é mais governada por autor-regressores de y_t e sim por outras variáveis explicativas, $\mathbf{x}_t$, as quais podem ser dependentes do tempo, no caso de estarmos tratando de séries temporais, ou simplesmente covariáveis independentes do tempo e entre si, que serão as regressões não-lineares.

Esses modelos também acompanham as variações de acordo com o tipo de funções de ligação onde teremos os modelos LSTR e ESTR. No caso de múltiplos regimes o MSTR (ver (31)).

O modelo STR e suas variações tem sua especificação, estimação e avaliação extensamente comentadas em (10).

B
Comando do programa R 2.6.2

Serão mostrados os comandos utilizados para o caso de Fraude/Irregularidade no Consumo de Energia Elétrica.

B.1
Comandos para GAM

```
library(VGAM)

gam1 = vgam(y ~ s(x15,df=4)+s(x16,df=4)+s(x17,df=4)+s(x18,df=4)+s(x19,df=4)+
+s(x20,df=4)+s(x21,df=4)+s(x22,df=4)+s(x23,df=4)+s(x25,df=4)+s(x26,df=4)
+s(x27,df=4),binomialff, lt)

phat=fitted.values(gam1)
Dev=deviance(gam1)
betas=as.matrix(coefficients(gam1)
cutoff=0.5 yhat=lt$y dim(yhat)=c(length(lt$y),1)
for (i in 1:length(lt$y))if (phat[[i]]>=cutoff) yhat[[i]]=1 else yhat[[i]]=0
phat_out=predict.vglm(gam1, newdata=nd, type="response")
dim(phat_out)=c(length(nd$y),1) yhat_out=nd$y dim(yhat_out)=c(length(nd$y),1)
for (i in 1:length(nd$y))if (phat_out[[i]]>=cutoff) yhat_out[[i]]=1 else
yhat_out[[i]]=0
```

B.2
Comandos para CART

```
library(tree)

cart1 = tree(y  x15+x16+x17+x18+x19+x20+x21+x22+x23+x25+x26+x27+x29+x30+x31,
data=lt)

plot(cart1) text(cart1) cv.cart1=cv.tree(cart1,rand=1:10,K=10,FUN=prune.tree)
prune.cart1=prune.tree(cart1,best=NULL)
phat=predict(cart1) yhat=predict(cart1, type=c("class")) dim(yhat)=c(length(lt$y),1)
```

B.3
Comandos para k-NN

```
library(kknn)
lt=as.data.frame(light[1:2430,]) nd=as.data.frame(light[-c(1:2430),])
knn1.learn ¡- lt knn1.valid ¡- nd

modknn1 = kknn(y  x15+x16+x17+x18+x19+x20+x21+x22+x23+x25+x26+x27+
+x29+x30+x31,knn1.learn,knn1.valid)

phat=fitted(modknn1) table(knn1.valid$y, phat)
cutoff=0.5 yhat=lt$y dim(yhat)=c(length(lt$y),1)
for (i in 1:length(lt$y))if (phat[[i]]>=cutoff) yhat[[i]]=1 else yhat[[i]]=0
phat_out = predict(modknn1, newdata = nd, type = "response")
dim(phat_out)=c(length(nd$y),1) yhat_out=nd$y dim(yhat_out)=c(length(nd$y),1)
for (i in 1:length(nd$y))if (phat_out[[i]]>=cutoff) yhat_out[[i]]=1 else
yhat_out[[i]]=0
```

B.4
Comandos para Regressão Logística

```
lt = as.data.frame(light[1:2430,]) nd = as.data.frame(light[-c(1:2430),])

glm1=glm(y  x15+x16+x17+x18+x19+x20+x21+x22+x23+x25+x26+x27+x29+x30+x31,
family=binomial(link=logit), data=lt)

X=model.matrix(glm1) betas=as.matrix(coefficients(glm1)) Dev=deviance(glm1)
phat=fitted(glm1) dim(phat)=c(length(lt$y),1)
cutoff=0.5 yhat=lt$y dim(yhat)=c(length(lt$y),1)
for (i in 1:length(lt$y))if (phat[[i]]>=cutoff) yhat[[i]]=1 else yhat[[i]]=0
phat_out=predict(glm1, newdata=nd, type="response")
dim(phat_out)=c(length(nd$y),1) yhat_out=nd$y dim(yhat_out)=c(length(nd$y),1)
for (i in 1:length(nd$y))if (phat_out[[i]]>=cutoff) yhat_out[[i]]=1 else
yhat_out[[i]]=0
```

C
Estatísticas Descritivas

C.1
E-mail/Spam

Tabela C.1: Estatísticas Descritivas - Spam

	Máximo	Mínimo	Média	Variância	Desvio Padrão
our	10.00	0.00	0.32	0.49	0.70
over	5.88	0.00	0.09	0.08	0.28
remove	5.40	0.00	0.12	0.16	0.40
internet	4.68	0.00	0.10	0.13	0.35
free	20.00	0.00	0.27	0.91	0.95
business	5.12	0.00	0.15	0.21	0.45
hp	18.18	0.00	0.55	2.90	1.70
hpl	10.86	0.00	0.27	0.80	0.89
george	33.33	0.00	0.75	10.92	3.30
1999	4.54	0.00	0.13	0.14	0.37
remove	20.00	0.00	0.28	0.97	0.99
edu	22.05	0.00	0.21	1.04	1.02
char_!	19.13	0.00	0.28	0.57	0.75
char_$	5.30	0.00	0.08	0.06	0.24
CAPMAX	2204.00	1.00	50.31	18348.72	135.46
CAPTOT	9163.00	1.00	267.41	265164.09	514.94

C.2
Doenças Cardíacas na África do Sul

Tabela C.2: Estatísticas Descritivas - DCAS

	Máximo	Mínimo	Média	Variância	Desvio Padrão
sbp	218.00	101.00	138.33	420.10	20.50
tobacco	31.20	0.00	3.64	21.10	4.59
ldl	15.33	0.98	4.74	4.29	2.07
obesity	46.58	14.70	26.04	17.76	4.21
alcohol	147.19	0.00	17.04	599.32	24.48
age	64.00	15.00	42.82	213.42	14.61

C.3
Fraude/Irregularidade no Consumo de Energia Elétrica

Tabela C.3: Estatísticas Descritivas - Fraude no Consumo de Energia

	Máximo	Mínimo	Média	Variância	Desvio Padrão
consumo	0.92	0.00	0.32	0.07	0.27
comsumo_ano_ant	0.94	0.00	0.36	0.07	0.26
consumo_ano_base	51.83	0.00	0.63	3.41	1.85
media_3	0.91	0.00	0.33	0.06	0.24
media_6	0.91	0.00	0.34	0.04	0.21
media_12	0.91	0.00	0.34	0.03	0.18
media_12_24	0.91	0.00	0.37	0.03	0.18
indic_trimestral_1	1.00	0.00	0.09	0.04	0.19
indic_trimestral_2	1.00	0.00	0.09	0.03	0.19
indic_trimestral_3	1.00	0.00	0.09	0.04	0.19
indic_anual	1.00	0.00	0.16	0.06	0.25
indic_ajuste	1.00	0.00	0.20	0.09	0.30
indic_tendencia	1.00	0.00	0.24	0.11	0.33
temperatura_min	0.81	0.33	0.51	0.02	0.13
temperatura_max	0.82	0.29	0.44	0.01	0.11
carga	0.70	0.19	0.36	0.02	0.13

D
Estimativas dos Coeficientes

D.1
E-mail/Spam

Tabela D.1: Coeficientes - Spam

	GAM	Regressão Logística	Análise Discrimina	STLR-Tree (REGIME 1)	STLR-Tree (REGIME 2)
Intercepto	-1.3979	-1.577	-	-0.391	-0.014
our	0.0110	0.010	0.250	-3.153	11.598
over	0.0184	0.018	0.243	-10.772	-7.960
remove	0.0350	0.042	0.414	9.870	12.162
internet	0.0157	0.020	0.243	-5.669	-1.185
free	0.0141	0.015	0.257	3.111	2.307
business	0.0130	0.012	0.142	0.037	-0.962
hp	-0.0313	-0.025	-0.205	2.401	-5.604
hpl	0.0020	-0.019	-0.122	1.802	0.145
george	-0.0428	-0.060	-0.210	-2.879	-328.586
1999	-	-	-0.148	-	-
remove	-0.0164	-	-0.149	-7.036	13.064
edu	-0.0198	-0.014	-0.168	-14.609	-7.917
char_!	0.0291	-0.034	0.266	3.451	-18.344
char_$	0.0819	0.027	0.328	19.250	14.834
CAPMAX	-0.0051	0.096	0.125	1.528	0.014
CAPTOT	0.0002	-0.003	0.282	0.017	0.017

D.2
Doenças Cardíacas na África do Sul

Tabela D.2: Coeficientes - DCAS

	GAM	Regressão Logística	Análise Discrimina	STLR-Tree (REGIME 1)	STLR-Tree (REGIME 2)
Intercepto	-4.1713	-4.3762	-	130.491	-3.629
sbp	0.0067	0.0051	0.104	-3.583	0.002
tobacco	0.0027	0.0033	0.389	2.664	0.081
ldl	0.0029	0.0026	0.393	6.386	-0.004
alcohol	-0.0003	-0.0001	0.016	-8.911	0.002
k-NN	0.0440	0.0514	0.563	-3.926	0.051

D.3
Fraude/Irregularidade no Consumo de Energia Elétrica

Tabela D.3: Coeficientes - Fraude no Consumo de Energia

	GAM	Regressão Logística	Análise Discrimina	STLR-Tree (REGIME 1)	STLR-Tree (REGIME 2)	STLR-Tree (REGIME 3)	STLR-Tree (REGIME 4)	STLR-Tree (REGIME 5)
Intercepto	0.52	-1.23	-	-2.38	-0.54	-3.86	33.85	-16.84
consumo	1.47	-	0.14	6.40	33.76	7.82	-35.36	-0.69
consumo_ano_ant	0.27	-	0.19	-1.48	-23.65	-1.65	-32.45	-0.23
consumo_ano_base	0.00	0.00	0.05	35.20	-6.64	1.03	15.22	-0.01
media_3	-0.83	-	-0.44	0.28	0.46	-0.11	0.98	0.98
media_6	-2.76	-1.55	-0.55	0.14	-7.70	-0.03	-19.19	-2.61
media_12	0.98	1.13	0.48	-0.78	-0.12	-3.32	6.09	1.51
media_12_24	-0.45	-0.77	-0.36	0.03	-1.54	6.50	40.83	-0.06
indic_trimestral_1	1.24	0.60	0.35	-	-	-	-	-
indic_trimestral_2	1.04	0.61	0.27	-	-	-	-	-
indic_anual	-0.34	-	0.10	-1.51	0.15	-5.74	-75.25	-0.72
indic_ajuste	-0.19	-	-0.34	-0.55	3.56	1.96	35.18	-0.24
indic_tendencia	0.14	-	-0.08	2.79	-4.65	-0.92	-25.79	-0.15
temperatura_min	-	5.27	1.34	-1.62	3.62	0.60	-18.00	1.90
temperatura_max	-	-	0.01	0.95	-0.59	-0.36	-29.10	0.34
carga	-	-2.76	-0.70	0.59	0.74	-4.70	56.38	-1.23

Tabela D.4: Pesos Redes Neurais - Fraude no Consumo de Energia

	Neurônio 1	Neurônio 2	Neurônio 3	Neurônio 4	Neurônio 5	Neurônio 6	Neurônio 7	Neurônio 8
consumo	-1.13	-31.57	-1.96	5.49	-3.29	2.38	2.91	-3.51
comsumo_ano_ant	0.37	-6.63	-9.18	-5.30	3.31	-0.48	1.11	-3.30
consumo_ano_base	-0.04	3.68	-2.92	-1.66	1.20	-0.19	-0.19	0.33
media_3	10.60	20.32	5.88	-3.43	1.59	-4.19	-8.30	10.77
media_6	-23.91	40.65	-1.44	-5.18	3.62	-0.02	12.74	-17.20
media_12	4.05	70.20	24.88	3.61	-2.35	-1.29	-4.75	8.71
media_12_24	0.38	-4.99	-12.80	-1.84	1.61	1.26	-5.31	5.01
indic_trimestral_1	8.61	-27.56	-0.36	2.37	-2.55	2.60	-1.82	1.02
indic_trimestral_2	2.65	18.67	5.08	5.79	-3.97	-0.21	-3.29	4.99
indic_trimestral_3	10.73	-26.45	0.05	4.47	-2.38	0.69	1.37	0.41
indic_anual	5.15	-20.05	5.52	2.48	-2.14	-2.34	8.68	-9.34
indic_ajuste	9.35	21.16	0.59	-1.99	1.48	-0.41	0.51	-0.77
indic_tendencia	-3.57	-0.91	2.39	0.83	-0.65	0.54	-0.14	0.61
temperatura_min	44.50	5.67	3.68	-24.13	18.24	-79.18	4.18	-2.26
temperatura_max	22.60	-1.91	42.95	21.04	-17.69	74.52	-17.42	21.25
carga	-48.36	-7.46	-31.66	-5.02	4.59	-8.68	13.64	-19.26

D.4
Coeficientes dos parâmetros não-lineares

Tabela D.5: Coeficientes Não-lineares

		c	γ
E-mail/Spam	1ª divisão	0.00	50.00
DCAS	1ª divisão	4.08	50.00
Consumo Energia	1ª divisão	3.54	10.22
	2ª divisão	2.82	50.03
	3ª divisão	3.74	50.00
	4ª divisão	3.57	99.99