Cálculo do limite superior para a capacidade discriminante de modelos preditivos baseado na informação disponível

Variáveis dependentes dicotómicas

André Pestana Sampaio e Melo

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

Cálculo do limite superior para a capacidade discriminante de modelos preditivos baseado na informação disponível

Variáveis dependentes dicotómicas

Dissertação orientada por

Professor Doutor Fernando José Ferreira Lucas Bação
Professor Doutor Victor José de Almeida Sousa Lobo

Outubro 2011

Agradecimentos

Este é um trabalho que resulta não só de empenho individual mas também, em larga medida, do contributo de várias pessoas sem as quais não seria possível sequer ter iniciado esta tese.

Gostaria de agradecer, em primeiro lugar, aos meus Orientadores, o Prof. Doutor Fernando Bação, não só pelos conhecimentos que me passou nas áreas da Estatística e da Gestão de Informação mas também pela disponibilidade e pelo incentivo que me deu em momentos mais complicados durante a construção desta tese, e ao Prof. Doutor Victor Lobo, pelo apoio prestado no âmbito estrito deste trabalho e também pelos desafios intelectuais que me colocou e que me obrigaram sempre a procurar saber mais e melhor.

Agradeço também ao SAS Institute Portugal, nomeadamente ao seu Director Geral Álvaro Faria, ao Director da área de *Business Development* Rui Rosa, e aos meus colegas em geral por acomodarem e compreenderem a menor disponibilidade profissional, da minha parte, que pontualmente ocorreu.

And last but not the least, destaco o papel da minha família nesta conquista, como pilar fundamental para o sucesso de toda minha carreira académica, profissional e pessoal: a minha mulher Xana, o meu filho Henrique e os meus pais, Maria da Conceição e Alberto Sampaio.

Resumo

Quando se avalia o poder discriminante de um determinado modelo (com variável dependente dicotómica) recorrendo à curva ROC, é usual representar-se no mesmo gráfico o "Modelo perfeito" e o "Modelo aleatório" enquanto limites teóricos (superior e inferior) à capacidade discriminante. O presente trabalho propõe o cálculo de um limite superior complementar, derivado dos dados e conceptualmente distinto do obtido via o "Modelo perfeito". Este novo limite designar-se-á "Capacidade discriminante dos dados" utilizados no desenvolvimento do(s) modelo(s) e encontra-se associado ao modelo Classificador Probabilista AP (*Probabilistic a Posteriori Classifier*). A utilidade desta abordagem passa por permitir, numa vertente mais prática, a estimação *a priori* (antes do trabalho exaustivo de modelação propriamente dito) da qualidade potencial dos dados para endereçar o problema de previsão em questão, bem como ajudar na rápida triagem das variáveis mais promissoras a incluir no futuro modelo preditivo a desenvolver. Numa vertente mais teórica, esta abordagem possibilita uma avaliação e uma comparação da capacidade efectiva que diferentes modelos preditivos apresentam na captura da capacidade discriminante encerrada nos dados.

Complementa-se os resultados teóricos com ilustrações empíricas obtidas a partir do ajustamento de duas metodologias distintas - Regressão Logística e Redes Neuronais – a dados de um ficheiro contendo informação sobre o comportamento creditício de 46,000 Clientes. Os resultados práticos tornam ainda evidente como se relaciona o "novo" limite com o tema do *overfitting*.

Abstract

When assessing the discriminatory power of a given model (with a dichotomous dependent variable) through a ROC curve, it is a commonplace to use both the "Perfect model" and "Random model" as theoretical upper and lower bounds for the model's discriminatory capacity. The current work aims at computing an empirical upper limit, derived from the actual data being used for model development, conceptually distinct from the traditional (and theoretical) usage of the "perfect model".

This new limit will be referred as "Data Discriminatory Power" and is associated with the Probabilistic AP (a Posteriori) Classifier Model. In practical terms, with this approach it is possible to estimate roughly (before engaging in more cumbersome, formal modeling tasks) the potential quality of data being used in future model developments or, on a more theoretical mindset, to estimate the percentage of discriminatory power buried within the data that is not being captured by the learning algorithms, and thus, assess and compare their limitations.

To illustrate the dynamics of what is being proposed, the theoretical results are complemented with empirical evidence resulting from fitting two well-known competing methodologies - Logistic Regression and Neural Networks – to a publicly available credit bureau file with 46,000 customer records. The later results also show the relation of such a limit and the problem of model overfitting.

Palavras-Chave

Classificador Probabilista AP; Capacidade discriminante; Curva ROC; AUC; U Mann-Whitney; Regressão Logística; Redes Neuronais; Classificador MAP de *Bayes*; Capacidade preditiva; Capacidade de generalização

Key words

Probabilistic AP Classifier; Discriminatory power; ROC Curve; AUC; U Mann-Whitney; Logistic Regression; Neural Networks; MAP Bayes Classifier; Predictive Power

Índice de figuras

Siglas e abreviaturas

AUC – Area under (the ROC) curve

MAP – Maximum a Posteriori

AP – a Posteriori

MLP – Mutli-layer perceptrons

ROC – Receiver Operating Characteristic

Índice

Capítulo 1 - Introdução

"... Estamos afogados em informação e sedentos por conhecimento..."

Rutherford D. Roger

No presente capítulo dá-se uma perspectiva sobre a importância dos modelos preditivos na sociedade do conhecimento e sobre uma das suas utilizações mais típicas, como seja para previsão de variáveis dependentes dicotómicas. Nesse contexto e devido à sua extensa utilização nesta dissertação, introduzem-se duas abordagens conceptualmente distintas relativamente ao desenvolvimento de modelos preditivos - regressão logística (e.g. Hosmer & Lemeshow, 1989), redes neuronais (e.g. Bishop, 1995) - bem como as técnicas mais utilizadas para avaliação da capacidade discriminante desses modelos, como sejam a curva ROC (*Receiver Operating Characteristic*) e a sua estatística resumo, o AUC (*Area Under the (ROC) Curve*) (Hanley & McNeil, 1982). Apresenta-se também o e Classificador MAP (*Maximum a Posteriori*) de Bayes (Alpaydin, 2004) como ponto de ligação para o Classificador Probabilista AP (*a posteriori*), um dos pilares do presente trabalho. Após este macro enquadramento, dedica-se o resto do capítulo à explicitação dos objectivos da tese e da forma como esta se encontra organizada.

1.1. Dos dados ao conhecimento. Quantificar a utilidade dos dados na produção de conhecimento específico.

A tríade "dados, informação e conhecimento" está no centro de uma mudança fundamental e irreversível no mundo empresarial - a primazia do conhecimento - que gradualmente tem vindo a ganhar terreno face a abordagens mais tradicionais e intuitivas de resolver os desafios. No meio empresarial reconhece-se a necessidade de incorporar conhecimento factual, sobre as operações da Empresa e dos seus Clientes, nos processos de decisão e gestão das organizações, retirando vantagens competitivas por forma a prevalecer num ambiente altamente competitivo como são os mercados actuais (Laursen & Thorlund, 2010).

Sendo a abundância de dados registados digitalmente um fenómeno recente na escala da história humana, não é de estranhar que muita investigação actual se encontre ligada a temas relacionados com a transformação desses dados em conhecimento. Nesse sentido e como refere Hand *et al.* (e.g. Hand, Mannila, & Smyth, 2001), muita atenção tem sido dedicada à recente "ciência" de extrair informação e conhecimento de vastas quantidades de dados - *Datamining*.

O *Datamining* endereça, genericamente, vários tipos de objectivos ou tarefas (e.g. Hand, Mannila, & Smyth, 2001) como seja a) a análise exploratória de dados, b) a modelação descritiva, c) a descoberta de padrões, d) a aprendizagem baseada em instância e, por fim e) a modelação preditiva.

Para levar a cabo estes desideratos, esta nova "disciplina" resulta do cruzamento de várias áreas de investigação, como sejam a estatística, a aprendizagem máquina, a gestão de informação e de base de dados, Inteligência artificial, entre outras. Nesse contexto destacam-se três grandes áreas genéricas de trabalho que, no nosso entender, produzem maior eco no meio empresarial:

a) 1ª área: Qualidade dos algoritmos de modelação. Que famílias de modelos conseguem previsões de maior qualidade? Que algoritmos de optimização devem ser utilizados para estimar os parâmetros dos modelos?
b) 2ª área: Fiabilidade dos dados. Será que estes dados reflectem correctamente a realidade ? Como endereçar e corrigir eventuais erros?
c) 3ª área: Utilidade dos dados para o problema em questão. Em que grau os dados ajudam a resolver determinado problema que o utilizador necessita de ver resolvido? Até onde poderá chegar a capacidade preditiva/ discriminante de um modelo desenvolvido nestes dados?

É na terceira área que se inclui o presente trabalho. Encontrar formas expeditas de avaliar (e quantificar) a utilidade de um conjunto de dados para responder a problemas específicos que se colocam ao analista.

1.2 Enquadramento científico

De uma forma genérica, e como o próprio nome indica, o objectivo de um modelo preditivo será o de antecipar o resultado de um evento baseado em informação existente *a priori*. Dito de outra forma, e recorrendo a terminologia de Probabilidade, pretende-se saber qual será a realização da variável aleatória Y dispondo de informação de um vector aleatório $\boldsymbol{X}$ (e.g. Hastie, Tibshirani, & Friedman, 2001).

Relacionado (e muitas vezes confundido) com o exposto atrás, uma das tarefas mais usuais atribuídas a modelos preditivos é suportar a tomada de decisão com base na capacidade de classificar, com baixa margem de erro, observações em duas classes pré-identificadas (que, sem perda de generalidade, tomarão o valor 1 para registar a realização de um acontecimento e o valor 0 para registar a realização do complementar do referido acontecimento*) mediante informação disponível. Nestes casos, a capacidade preditiva coincide com a habilidade evidenciada pelo modelo em separar adequadamente as observações em classes distintas, designadamente a sua capacidade discriminante (e.g. Hand, Mannila, & Smyth, 2001).

No âmbito da presente tese recorrer-se-á a duas técnicas muito utilizadas para tarefas de modelação preditiva para variáveis dependentes binárias: a Regressão Logística e as Redes Neuronais MLP (*Multi-layer Perceptrons*) (e.g. Patterson, 1996).

A escolha destes métodos deve-se às seguintes razões:

a) São as metodologias amplamente estudadas e com enorme utilização no desenvolvimento de modelos preditivos para variáveis dicotómicas (e.g. Nisbet, Elder, & Miner, 2009),

b) Apresentam perspectivas distintas na tarefa de "ajustamento de funções a dados", com a Regressão Logística a incluir-se nos métodos baseados em pressupostos (onde a estrutura genérica da função se encontra pré-definida (e.g. Bação, 2009)), e as Redes Neuronais a incluírem-se em métodos sem pressupostos *a priori*, deixando os dados

(*) Doravante, por forma a simplificar a exposição utilizar-se-á jargão de uso corrente em *datamining* por contraponto a terminologia mais formal e conceptualmente mais correcta utilizada em probabilidade. No presente trabalho a expressão *evento* designa a realização de um acontecimento e não-evento designa a realização do acontecimento complementar.

"esculpir" o aspecto final da equação de regressão (abordagem mais flexível que a regressão logística (e.g. Bação, 2009)).

c) Representam campos distintos da interpretabilidade do modelo final, nomeadamente com a Regressão Logística a produzir resultados transversais e facilmente assimiláveis em contraponto com as Redes Neuronais cuja equação de regressão resulta de uma ponderação de múltiplas sigmóides, colocando maiores dificuldades a uma compreensão intuitiva por parte do analista (e.g. Engelmann & Rauhmeier, 2006).

Apresenta-se ainda o Classificador MAP de Bayes, importante nesta tese porque aborda o tema da classificação totalmente baseada nos dados do problema - sem nenhuma equação de regressão (e.g. Lobo, 2008) - o que liga directamente com o Classificador Probablista AP utilizado nas demostrações formais do capítulo 3.

Dispondo das técnicas de modelação, na fase de desenvolvimento de qualquer modelo propriamente dito (para variáveis binárias), tem-se não só o ajustamento de uma função aos dados observados, mas também uma avaliação global subsequente da sua qualidade via estimação da capacidade discriminante. Uma das abordagens mais utilizadas para avaliar essa última vertente é a denominada curva ROC (e.g. Engelmann & Rauhmeier, 2006). No entanto, sendo a curva ROC uma representação gráfica bidimensional da capacidade preditiva de um modelo, é importante dispor de estatísticas resumo que quantifiquem e sintetizem a informação contida nessa representação bidimensional. Existem diversas possibilidades (e.g. Basel Committee on Banking Supervision, 2005) mas a mais utilizada na prática é a AUC(e.g. Engelmann & Rauhmeier, 2006).

Tão ou mais importante que avaliar a capacidade discriminante que o modelo evidencia nos dados utilizados para treinar o modelo, é medir-se a capacidade preditiva observada em novos dados não utilizados no seu desenvolvimento. Esta abordagem denomina-se de validação cruzada (*cross-validation*) (Alpaydin, 2004) e o conjunto de dados utilizado para o efeito denomina-se de amostra de teste. Tal como na amostra de treino, a curva ROC (e o respectivo AUC) é novamente a ferramenta de eleição para desempenhar essas tarefas.

Quer se esteja a trabalhar com dados de treino, quer se esteja numa vertente de validação cruzada é usual representar-se conjuntamente com curva ROC do modelo, as curvas referentes a um modelo aleatório (sem capacidade discriminante) e a um modelo perfeito (capacidade discriminante máxima), representando estas curvas o limite inferior e o limite superior do que pode ser conseguido pelo modelo em termos de capacidade discriminante.

O que se propõe no presente trabalho é que a referência superior para a capacidade discriminante de um modelo não deva ser apenas a noção teórica de "modelo perfeito", mas também uma referência empírica, derivada (e especifica) dos dados utilizados para o desenvolvimento do modelo.

Nota: Para facilitar a exposição e tornar menos repetitivo o discurso, no presente trabalho falar-se-á de capacidade discriminante ou capacidade preditiva de um modelo quando aplicado à amostra onde este foi treinado e de capacidade de generalização de um modelo quando aplicado à amostra para *cross-validation*.

1.3 Objectivos do trabalho

Os objectivos principais deste trabalho são: a) Calcular um limite superior para a capacidade discriminante de qualquer modelo preditivo (para uma variável dependente dicotómica) com base na informação disponível e utilizada para a tarefa de modelação. A este limite chamar-se-á a "capacidade discriminante dos dados" para o exercício previsão em questão e é calculado tendo por base o "ajustamento" do modelo Classificador Probabilista AP (*Probabilistic AP (a posteriori) Classifier*) aos dados, b) Mostrar empiricamente como se comporta o modelo Classificador Probabilista AP relativamente a outros modelos no que se refere à capacidade de generalização para um conjunto de novos dados.

1.4 Relevância do trabalho

O Contributo de uma tese pode ser avaliado na medida em que os seus resultados permitem realizar algo novo ou fazer de forma mais eficiente o que já é feito com regularidade. O contributo desta tese toca um pouco nos dois aspectos, ilustrando-se na secção 1.4.1 aspectos de eficiência enquanto na secção 1.4.2 se ilustram formas adicionais de testar problemas mais gerais.

1.4.1 Avaliar, *a priori*, a qualidade dos dados a utilizar no desenvolvimento de modelos e seleccionar as variáveis mais promissoras

A matéria-prima para modelação estatística são dados. Como já se evidenciou, estes estarão cada vez disponíveis (*ad nauseum*) em variados formatos, qualidades e meios, mas não oferecem qualquer garantia, *a priori*, de ter capacidade para responder adequadamente às questões que se colocam e que levaram à necessidade conceptual de desenvolvimento do modelo ou modelos.

Num primeiro plano, importa então dispor de capacidade rápida para avaliar a qualidade da informação para produção de resultados úteis antes de aplicar metodologias estatísticas mais pesadas. Desta forma sabe-se antecipadamente o que esperar do exercício de modelação identificando-se a eventual necessidade de procurar mais e melhores dados.

Onde esta abordagem se poderá revestir de especial utilidade é precisamente nas ciências médicas onde a recolha dados é tradicionalmente onerosa, difícil e lenta. Nesta área é fundamental conseguir avaliar se os indicadores/marcadores tradicionalmente recolhidos serão suficientes para produzir modelos de qualidade ou se novos indicadores mostram potencial promissor.

Por outro lado, os resultados do presente trabalho permitem que se ordene as variáveis de qualquer conjunto de dados, de acordo com a sua capacidade discriminante, facilitando sobremaneira o trabalho de selecção de um subconjunto de variáveis a utilizar no modelo preditivo final que produzam resultados preditivos de qualidade (não implica que seja o subconjunto óptimo, apenas um subconjunto de grande qualidade obtido de forma expedita).

Implementou-se, em SAS® Enterprise Miner, um nó analítico que ajusta ao dados o modelo Classificador Probabilista AP. Assim, com este nó consegue estimar-se, não só, a capacidade discriminante de cada variável independente (ajudando a uma pré-selecção da variáveis a incluir num futuro modelo) mas também se calcula a capacidade discriminante máxima do conjunto de dados utilizado no treino do modelo preditivo final.

1.4.2 Referência para a avaliação da capacidade de modelos preditivos

Numa perspectiva inversa, dispondo de um conjunto de dados e sabendo qual o limite da capacidade discriminante intrínseca dos mesmos, torna-se mais simples avaliar e comparar a qualidade algorítmica de variadas técnicas de modelação para extrair a referida informação dos dados.

1.5 Questões de investigação

As questões fundamentais de investigação são:

a) Calcular o limite à capacidade discriminante de qualquer modelo condicionado aos dados utilizados para o ajustamento do mesmo.

b) Mostrar empiricamente as relações que podem existir entre esse limite e o tema do *overfitting*, quando na presença de conjuntos de dados não separáveis para o problema em questão.

1.6 Metodologia de investigação e dados

A metodologia seguida nesta dissertação passará pela introdução e demonstração de hipóteses teóricas, ilustrando-se posteriormente estas conclusões com resultados empíricos obtidos no desenvolvimento de modelos preditivos num ficheiro de dados reais.

O ficheiro utilizado para ilustrar as principais conclusões contém 46,000 registos de clientes, a quem foi concedido crédito por um Banco e para os quais se dispõe 27 variáveis observadas no momento da concessão de crédito. Dispõe-se igualmente de uma variável binária, calculada *a posteriori*, indicando se cada Cliente entrou ou não, nos 12 meses seguintes, em incumprimento.

Para que seja possível avaliar empiricamente a diferença entre "capacidade discriminante" e "capacidade de generalização" dos modelos, os 46,000 Clientes foram divididos 50%-50%, por amostragem aleatória estratificada, entre amostra de treino (para desenvolvimento dos modelos e cálculo da "capacidade discriminante") e amostra de teste (para teste dos modelos e cálculo da "capacidade de generalização").

As amostras de treino e de teste ficaram, cada uma, com 23,000 Clientes e 750 maus pagadores.

1.6.1 SAS® Enterprise Miner

O *Enterprise Miner* é um *software* desenhado pelo SAS® *Institute* orientado para o desenvolvimento de todo o tipo de modelos analíticos, sejam eles preditivos ou descritivos. O SAS® *Enterprise Miner* tem vários pontos muito fortes, como sejam:

a) A disponibilização de uma vasta gama de algoritmos oriundos das áreas da estatística ou de áreas relacionadas com *machine learning*, garantindo que se consegue realizar qualquer análise de A a Z sem necessidade de recorrer a outras ferramentas,

b) um interface com o utilizador que permite, de forma simples e visualmente agradável, estruturar qualquer processo analítico, desde o mais simples ao mais complexo,

c) permitir a realização de análises paralelas com vista à comparação de resultados e opção pela abordagem mais potente, entre outras.

O processo analítico utilizado nesta tese está exposto na figura 1.1

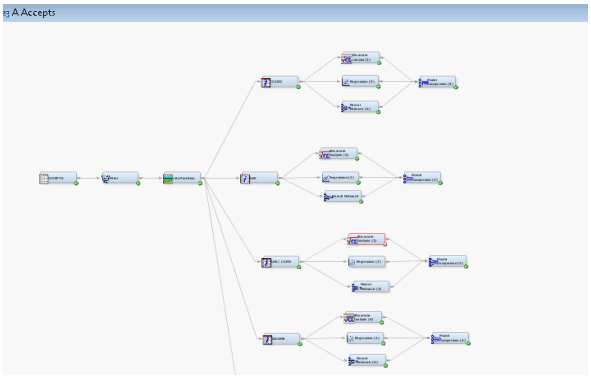


Figura 1.1 – Visão global do processo analítico

Na figura 1.2 destacam-se as principais macro-componentes do processo analítico.

Figura 1.2 – Macro-componentes do processo analítico

A macro-componente (A) refere-se ao *input* do ficheiro no diagrama e a um rápido e simples pré-processamento dos dados. As componentes de (B) a (E) referem-se ao desenvolvimento (paralelo) de vários modelos utilizados para a produção dos resultados mostrados neste trabalho.

Olhando mais em detalhe têm-se na macro-componente (A):

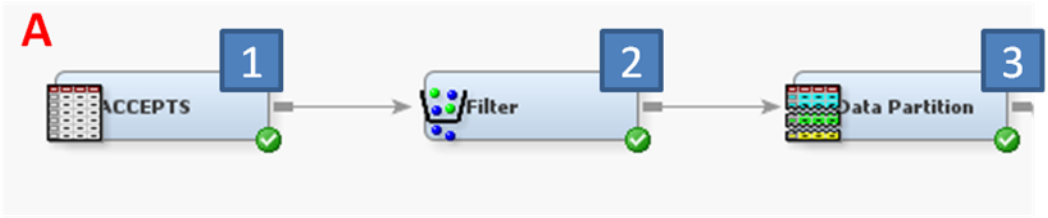


Figura 1.3 – Pré-processamento de dados para o processo analítico

Em 1) indica-se qual o ficheiro a utilizar e define-se toda a parte de *metadata* dos dados, nomeadamente: a) que variáveis desempenham o papel de variáveis de input e qual vai ser a variável dependente e b) qual o tipo de cada uma das variáveis: Binária, nominal, ordinal ou intervalar.

No passo 2) retiram-se alguns *outliers* mais grosseiros (23 observações em 46.000) e no passo 3) procede-se à partição do ficheiro em treino e teste (50%-50% via amostragem aleatória estratificada utilizando apenas a variável dependente).

As macro-componentes de (B) a (E) apresentam estruturas semelhantes, e referem-se ao desenvolvimento de modelos preditivos considerando conjuntos de variáveis independentes distintas com vista a ilustrar diferentes argumentos nesta tese. A título de exemplo, detalha-se apenas a macro-componente (C).

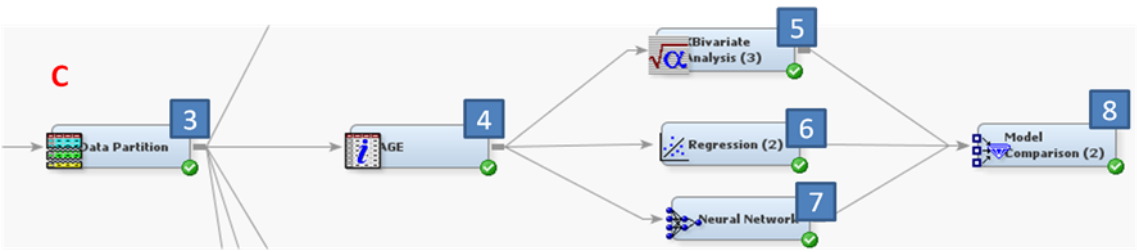


Figura 1.4 – Desenvolvimento paralelo de modelos preditivos para comparação de resultados

Nota: O ponto 3) pertence à macro-componente (A) e está aqui representada para fazer a ligação dos diagramas.

Da macro-componente C, no ponto 4) retém-se apenas as variáveis (neste caso é a idade), a utilizar nos 3 nós posteriores de modelação, no ponto 5) ajusta-se o modelo "Classificador Probabilista AP", nos pontos 6) e 7) ajustam-se os modelos regressão logística e redes neuronais, e o no ponto 8) comparam-se as curvas ROC (e as respectivas AUC) dos três modelos em competição.

1.7 Organização da tese

O capítulo 2 conterá uma breve apresentação teórica as metodologias utilizadas neste trabalho para prever variáveis dependentes dicotómicas, e apresentará uma fundamentação teórica sobre formas de avaliar a capacidade discriminante desses modelos, nomeadamente através da curva ROC e a sua estatística resumo, a AUC. Neste capítulo também se apresenta a estatística não paramétrica U de Mann-Whitney-Wilcoxon como estimativa da AUC (Hanley & McNeil, 1982).

No capítulo 3 detalha-se a forma de ajustar o Classificador Probabilista AP a um conjunto de dados e demonstra-se que este modelo majora a capacidade discriminante de qualquer modelo preditivo quando desenvolvido sobre a mesma informação.

No capítulo 4 ilustram-se empiricamente os resultados obtidos no capítulo 3 via o ajustamento de modelos a um conjunto de dados reais.

No capítulo 5 retiram-se as conclusões sobre os resultados obtidos.

No capítulo 6 propõe-se tópicos adicionais de estudo.

.

Capítulo 2. Conceitos e definições

No presente capítulo sintetizam-se alguns aspectos teóricos sobre modelação preditiva e detalham-se algumas implicações quando as variáveis dependentes são dicotómicas. Apresenta-se três das metodologias típicas para endereçar estes temas, nomeadamente a regressão logística, as redes neuronais e classificador MAP (*Maximum a Posteriori*) de Bayes. Termina-se o capítulo mostrando como se avalia a capacidade discriminante de modelos utilizando a curva ROC e a sua estatística resumo, a AUC. Destaca-se a utilização da estatística U de Mann-Whitney como estimador da AUC quando perante variáveis dependentes dicotómicas.

2.1 Modelos Preditivos

A utilização de modelos preditivos tem por objectivo antecipar a realização de um acontecimento dispondo, *a priori*, de informação de um conjunto de outros eventos já registados (e.g. Hand, Mannila, & Smyth, 2001).

Restringindo o tema da previsão a *inputs* e *outputs* quantitativos, encontramo-nos no mundo das variáveis aleatórias e dos espaços de probabilidade (e.g. Hastie, Tibshirani, & Friedman, 2001). O problema de previsão pode então ser definido da seguinte forma: seja $\boldsymbol{X} \in R^p$ vector aleatório real (*input*) de dimensão p e $Z \in R$ uma variável real aleatória de *output*, apresentando estas entidades uma distribuição conjunta $\mathrm{P}(\boldsymbol{X}, Z)$. De uma forma simplista pretende-se então estimar uma função

$$Z = f(\boldsymbol{X}; \boldsymbol{\theta}) \text{ (onde } \theta \text{ representa os parâmetros do modelo)}$$

que permita determinar Z quando se dispõe da informação $\boldsymbol{X}$.

Mas sendo o objecto do exercício de previsão uma variável aleatória, este é um exercício condenado ao insucesso se abordado de forma exacta, pois "variável aleatória" traduz precisamente uma dinâmica não determinística. O problema necessita de ser definido incorporando uma componente de erro ε. A equação que relaciona Z com $\boldsymbol{X}$ terá então uma expressão analítica (Alpaydin, 2004):

$$Z = f(\boldsymbol{X}; \boldsymbol{\theta}) + \varepsilon \text{ (supondo um erro aditivo e onde } \theta \text{ representa os parâmetros do modelo)}$$

O objectivo da modelação preditiva passa então por ajustar uma função $f(\boldsymbol{X};\boldsymbol{\theta})$ aos dados que produza desvios "pequenos" entre o valor Z observado e o valor $\hat{Z} = f(\boldsymbol{X};\boldsymbol{\theta})$ para várias repetições da mesma experiência aleatória. Para tratar, em termos genéricos, o problema da minimização do erro ε recorre-se à denominada função perda (*Loss Function*) $L(Z, f(\boldsymbol{X};\boldsymbol{\theta}))$ (e.g. Hastie, Tibshirani, & Friedman, 2001).

A abordagem mais comum é a utilização da função de perda (e.g. Hastie, Tibshirani, & Friedman, 2001)

$$L\big(Z, f(\boldsymbol{X};\boldsymbol{\theta})\big) = (Z - f(\boldsymbol{X};\boldsymbol{\theta}))^2$$

que penaliza erros mais grosseiros em detrimento de erros menores.

Considerando que se pretende minimizar a componente de erro aleatório, a função traduz-se no objectivo de minimização do erro quadrático médio, $E(\mathrm{Z} - f(\mathbf{X};\boldsymbol{\theta}))^2$.

Após alguma manipulação algébrica, conclui-se que a função que minimiza $E(\mathrm{Z} - f(\mathbf{X};\boldsymbol{\theta}))^2$ (e.g. Hastie, Tibshirani, & Friedman, 2001) é:

$$f(\boldsymbol{X};\boldsymbol{\theta}) = E(Z|\boldsymbol{X} = x)$$

sendo este um modelo adequado para prever a realização da variável aleatória Z, com base na informação **X**.

Antes de se abordar as metodologias que permitem estimar a função $f(\boldsymbol{X};\boldsymbol{\theta})$ importa então perceber qual o *modus operandi* de um Modelo Preditivo.

2.1.1 Modelos preditivos sem informação *a priori* – previsão *naive*

A denominação "previsão naive" está tradicionalmente explicitada numa perspectiva específica na literatura tradicional da área de *datamining* e *machine learning* mas, na minha opinião, a utilização do termo pode estendida também à situação de quando não se dispõe de informação $\boldsymbol{X}$ *a priori*. Nessa situação a função que minimiza o erro quadrático médio é a função constante $f(\boldsymbol{X};\boldsymbol{\theta}) = E(Z)$, ou seja, o melhor palpite que se pode dar sobre uma futura realização de Z é o seu valor médio (e.g. Mendenhall &

Sincich, 1996) e esse palpite acarreta um erro médio $E(Z - E(Z))^2 = Var(Z)$ (figura 2.1).

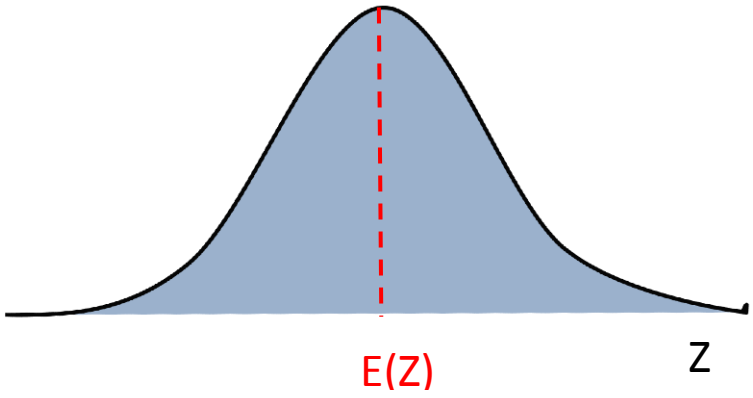


Figura 2.1 – Previsão *naive* na ausência de informação adicional

2.1.2 Modelos Preditivos com informação *a priori*

Dispondo de informação *a priori* **X**, um modelo preditivo será então aquele que produza previsões distintas, para alguns valores de ***X*=*x***, da abordagem *naive* E(*Z*). Assim um modelo apresentará uma capacidade preditiva superior, consoante mais níveis distintos do vector ***X*** corresponderem a níveis médios distintos de Z ($x_1 \neq x_2 \rightarrow E(Z|X = x_1) \neq E(Z|X = x_2)$) e cuja variância de $Z|X = x$ seja também tendencialmente inferior a $Var(Z)$, a variância total de Z. Essa dinâmica pode ser observada na figura 2.2.

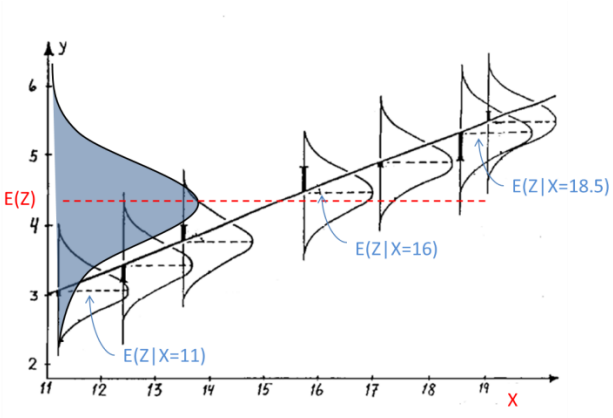


Figura 2.2 – Níveis médios E(*Z*) distintos em função da alteração do vector input *X*

2.1.3 Modelos preditivos para variáveis dependentes dicotómicas

Considerando, sem perda de generalidade, uma variável binária Y que assume os valores 1 e 0 (evento e não-evento) e que se verifica independência entre as várias realizações da variável, será expectável que variável Y tenha subjacente a seguinte lei probabilista de Bernoulli.

$$Y \sim Ber(\pi_1) = \begin{cases} 1 & p = \pi_1 \\ 0 & p = 1 - \pi_1 \end{cases}, \text{ com } E(Y) = \pi_1 \; e \; Var(Y) = \pi_1(1 - \pi_1)$$

Tem-se que π_1 reflecte a probabilidade de acontecer o evento e $(1 - \pi_1)$ a reflecte a probabilidade de acontecer o acontecimento complementar (não-evento).

Caso de disponha de informação *a priori* $\boldsymbol{X}$ então, em cada nível de $\boldsymbol{X} = \boldsymbol{x}$, ter-se-á

$$Y|\boldsymbol{X} = x \sim Ber(\pi_x) = \begin{cases} 1 & p = \pi_x \\ 0 & p = 1 - \pi_x \end{cases},$$

$com \; E(Y|\boldsymbol{X} = x) = \pi_x \;\; e \;\; Var(Y|\boldsymbol{X} = x) = \pi_x(1 - \pi_x)$ (e.g. Hosmer & Lemeshow, 1989)

Como corolário do que foi apresentado até agora, é trivial mostrar que a função $f(\boldsymbol{X}; \boldsymbol{\theta})$ estima π_x, como se depreende das igualdades:

$$Y = f(\boldsymbol{X}; \boldsymbol{\theta}) + \varepsilon = E(Y|\boldsymbol{X} = x) + \varepsilon = \pi_x + \varepsilon$$

A estimativa de Y, $\hat{Y} = \hat{f}(\boldsymbol{X}; \boldsymbol{\theta}) = \hat{\pi}_x$, apresenta então *outputs* quantitativos no intervalo [0,1], referentes à estimativa da proporção de eventos π_x (na população) em cada nível $\boldsymbol{X} = \boldsymbol{x}$ da variável independente (e.g. Hosmer & Lemeshow, 1989), apesar de *Y* apresentar valores {0,1}.

2.1.4 Modelos preditivos para variáveis dicotómicas, *Scores* $\hat{\boldsymbol{\pi}}_{\boldsymbol{x}}$ e capacidade de ordenação

Considerando que o modelo tem poder preditivo, então espera-se (Figura 2.2) obter proporções estimadas distintas $\hat{\pi}_x \neq \hat{\pi}$ para níveis distintos de $\boldsymbol{X} = \boldsymbol{x}$. Ter-se-á então

maior concentração de 1's na variável dependente, nas zonas de $\boldsymbol{X}=\boldsymbol{x}$ onde $\hat{\pi}_x \to 1$ e maior concentração de 0's na variável dependente, nas zonas de $\boldsymbol{X}=\boldsymbol{x}$ onde $\hat{\pi}_x \to 0$.

Diz-se então que o score $\hat{\pi}_x$ calculado pelo modelo $\hat{f}(\boldsymbol{X};\boldsymbol{\theta})$ permite discriminar a variável *Y* pois "separa tendencialmente" os 0's, dos 1's.

Visto de outra forma, ordenando os dados pelo *score* $\hat{\pi}_x$ produzido pelo modelo, e sendo este preditivo, está-se a discriminar os eventos dos não-eventos na variável dependente.

Em resumo, capacidade discriminante, capacidade de ordenação e capacidade preditiva são conceitos interligados em problemas com variáveis dependentes binárias.

Antes de prosseguir, importa realçar uma pequena nota. Caso se pretenda que o não-evento seja o evento a modelar, basta calcular a variável aleatória $Y' = 1 - Y$. Daí para a frente a abordagem é idêntica à apresentada para a variável *Y*.

2.2 Regressão logística, redes neuronais e classificadores MAP de Bayes

Em Probabilidade e Estatística existem diversas metodologias que permitem endereçar o tema da estimação de $f(\boldsymbol{X};\boldsymbol{\theta})$.

No caso de variáveis dependentes dicotómicas $Y \in \{0, 1\}$ que técnicas serão adequadas? Mesmo desenhando um gráfico simples (figura 2.3, onde se representa uma só variável independente *X* e uma variável dependente *Y* dicotómica) para ajudar à nossa intuição, não se antecipa que tipo de função ajustar a estes dados.

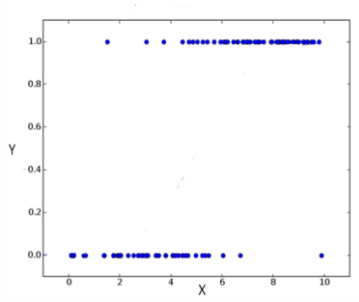


Figura 2.3 – Visualização da distribuição conjunta de uma variável dependente dicotómica Y e uma variável independente contínua X

De facto, o que se procurará modelar são os valores médios $E(Y|X = x)$ não observáveis directamente nos dados. Para "visualizar" esses pontos médios (Figura 2.4) imagine-se categorizar a variável X em 10 classes e calcular a média de Y para cada classe (e.g. Hosmer & Lemeshow, 1989). Já se obtêm os pontos $p_1, \dots, p_{10} \in [0,1]$. A função $f(\boldsymbol{X}; \boldsymbol{\theta})$ procurará então representar a esses pontos

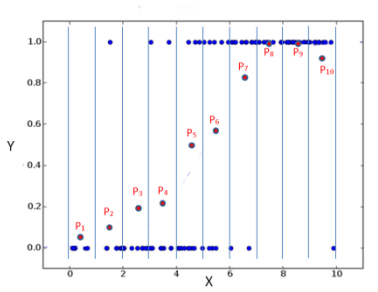


Figura 2.4 – Discretização da variável independente X e as respectivas proporções observadas na variável dependente Y em cada nível de X

2.2.1 Regressão logística

A regressão logística é uma das metodologias mais utilizados para endereçar o tema de modelos com variável dependente binária precisamente porque o seu *output* devolve valores no intervalo [0,1] (e.g. Witten & Frank, 2005). Como foi referido no capítulo introdutório, a regressão logística inclui-se nos métodos baseados em pressupostos, onde se encontra pré-definida a estrutura da função que vai ser utilizada para modelar os dados, sendo apenas necessários estimar alguns parâmetros (e.g. Bação, 2009).

A regressão logística propõe a seguinte expressão analítica para relacionar $\boldsymbol{X}$ com E($Y|\boldsymbol{X}=\boldsymbol{x}$) (e.g. Hosmer & Lemeshow, 1989):

$$\pi_x = E(Y|X = x) = \frac{e^{\alpha+\beta X}}{1+e^{\alpha+\beta X}}, \quad (1)$$

Esta função tem um aspecto sigmoidal e como se vê na figura 2.5, e quando estimada, apresenta visualmente um ajustamento aos valores médios p_x

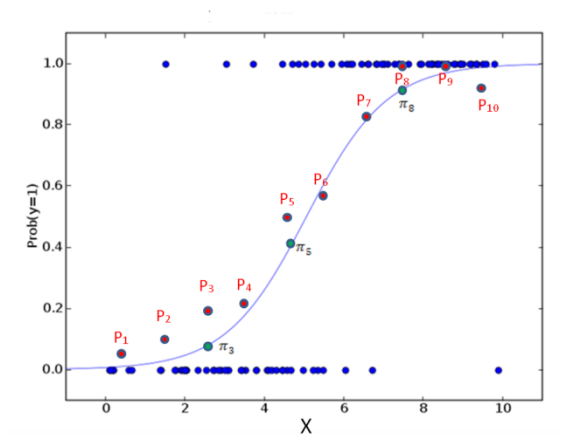


Figura 2.5 – Ajustamento de uma regressão logística a dados

Outra forma de olhar para a regressão logística passa por se trabalhar uma transformação conhecida como o "logaritmo das chances" (*Log Odds*). Prova-se que estimar (1) é equivalente a estimar (e.g. Hosmer & Lemeshow, 1989):

$$Ln(\frac{\pi_x}{1-\pi_x}) = \alpha + \beta X$$

Esta fórmula é conhecida como o LOGIT.

Como se depreende, o modelo logístico é então indicado para modelar dados em que os "logaritmos das chances" variem linearmente em função da variável independente **X**.

A forma de estimar os parâmetros α e β pode ser vista em Hosmer et al. (e.g. Hosmer & Lemeshow, 1989)

2.2.2 Redes Neuronais MLP

Outra abordagem ao tema do ajustamento de uma função que ligue $\boldsymbol{X}$ a $E(Y|\boldsymbol{X} = x)$ é utilizar as chamadas redes neuronais MLP (e.g. Patterson, 1996).

Por contraponto à regressão logística, as redes neuronais MLP nada assumem relativamente à função que se ajusta aos dados, sendo a "descoberta" desta relação guiada pelas observações utilizadas para treinar a rede (e.g. Bação, 2009). As Redes Neuronais MLP incluem-se nos denominados "aproximadores universais de funções". Em teoria, podem-se criar Redes Neuronais suficientemente complexas que produzam funções que se ajustam a qualquer conjunto de dados (Alpaydin, 2004).

As redes neuronais abordam o paradigma de ajustamento de funções na perspectiva da aprendizagem máquina (*Machine Learning*). Significa isto que partem duma representação do conhecimento "em branco" (melhor dizendo, inicializado de forma aleatória) e vão ajustando essa representação consoante os dados (sob a forma ($\boldsymbol{X}$,Y)) fornecidos por um supervisor – daí o nome de aprendizagem supervisionada. Na base das redes neuronais encontram-se algoritmos que permitem modificar a relação *input*/*output* $f(\boldsymbol{X};\boldsymbol{\theta})$ em função das diferenças $y_i - f(x_i)$ observadas à saída, procurando torná-las menores (e.g. Bishop, 1995).

Um exemplo ilustrativo de aprendizagem iterativa (*on-line training*) de uma rede MLP pode ser visto na sequência de figuras apresentadas. Na figura 2.6 mostra-se uma rede que inicializa aleatoriamente a representação de conhecimento, quando ainda não lhe foram apresentados dados. Posteriormente é apresentado o primeiro dado e a rede ajusta a função f por forma a diminuir $y_1 - f(x_1)$. Como se repara a função é ajustada sobretudo localmente, o que não é possível numa representação rígida como seja na regressão logística. Seguidamente apresenta-se um segundo dado e o algoritmo procede a novo ajustamento quasi-local e assim sucessivamente.

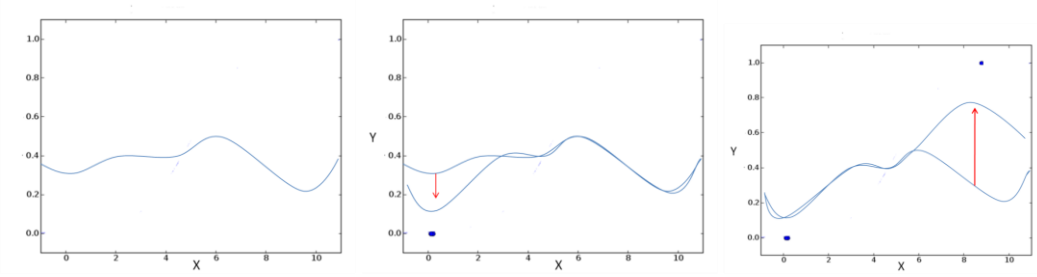


Figura 2.6 – Ajustamento de uma função a dados apresentados sequencialmente a uma rede neuronal

Após uma série de iterações, onde os dados foram apresentados várias vezes à rede, e após verificar-se determinado critério de paragem, surge então a representação final do conhecimento (figura 2.7)

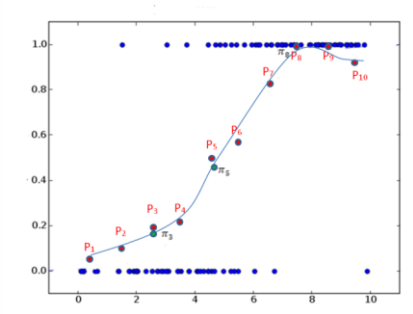


Figura 2.7 – Aspecto final do ajustamento de uma função a dados recorrendo a uma rede neuronal

As redes neuronais são tão flexíveis em termos de representação do conhecimento porque têm subjacente um modelo matemático que procura replicar a forma paralela como o cérebro processa e guarda informação (e.g. Patterson, 1996).

A tradução de todo este processo cognitivo para um modelo matemático pode ser visualizado na figura 2.8 e é conseguido através de (e.g. Patterson, 1996):

a) Definição de uma camada constituída por nós de *input* - equivalentes aos nossos sentidos – que recebem e estandardizam a informação,

b) definição de uma camada interna, equivalente a neurónios cerebrais, para modelação de informação. Essa camada é constituída por vários nós que desempenham funções de combinação de informação e posterior modulação/ampliação/redução do "sinal" recebido,

c) uma série de ligações entre os nós, representando as sinapses, que podem ser fortalecidas ou enfraquecidas em função das necessidades de aprendizagem para que o modelo final seja uma representação aproximada dos dados/factos observados,

d) uma camada de *output* que calcule o valor final estimado pela rede (o efeito) em função do que foi recebido via camada de input (as causas) e o compara com a realidade, por forma permitir correcções às ligações sinápticas para que a representação interna se aproxime da realidade externa.

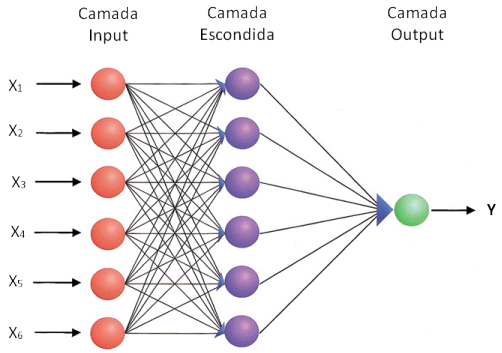


Figura 2.8 – Macro-componentes de uma rede neuronal (adaptado de http://iticsoftware.com/forex-neural-backpropagation)

O modelo matemático para cada neurónio está representado na figura 2.9. Cada nó na camada escondida ou na camada output apresentará uma estrutura semelhante deste tipo (e.g. Lobo, 2008):

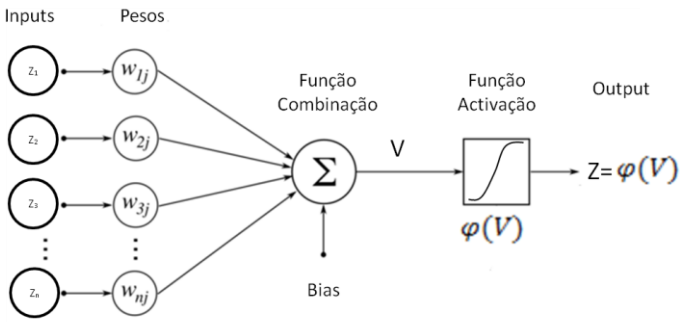


Figura 2.9 – Modelo matemático de um nó da rede neuronal, da camada escondida ou da camada *output* (adaptado de http://en.wikibooks.org/wiki/Artificial_Neural_Networks/Activation_Functions)

Os parâmetros a estimar no modelo serão as "forças da ligação" (representadas por pesos w_i) de cada sinapse, entidades que ligam os nós. O processo de estimação consiste na apresentação de dados à rede, no cálculo dos respectivos outputs (*feedforward*), seguido da retro-propagação (*backpropagation*) do erro observado na camada output - g($y_i - f(x_i)$) - para as camadas escondidas com vista à alteração dos pesos sinápticos no sentido de minimizar esse erro (e.g. Patterson, 1996).

2.2.3 Classificador MAP de Bayes

Num problema de classificação de novas observações em k classes pré-existentes C_i, e quando se dispõe da informação *a priori* $\boldsymbol{X}$ (vector) relativo a essas observações, o classificador *Bayesiano* MAP opta por classificar cada observação escolhendo a classe C_i que maximiza a probabilidade condicional $P(C_i|X = x), \forall i = 1, \dots, k$.

Resumidamente tem-se (Alpaydin, 2004):

$$P(C_i|X = x) = \frac{P(C_i) * P(x|C_i)}{P(x)}, \forall i = 1, \dots, k$$

onde $P(C_i)$ é a probabilidade *a priori* da classe C_i, $P(x|C_i)$ é a verosimilhança da classe C_i e $P(x)$ é a evidência. A regra de decisão baseia-se na escolha da classe C_i quando $\max_k P(C_k|X = x) = P(C_i|X = x)$.

Este problema está relacionado com estimar $E(Y|X = x)$ e utilizar esse resultado para tomar uma decisão. No caso de uma estimação não paramétrica, recorre-se directamente aos dados para estimar, em cada ponto **X=x**, $P(C_i|X = x)$, $\forall i = 1, \dots, k$ (Alpaydin, 2004).

Como indica Lobo (e.g. Lobo, 2008), o classificador MAP é sempre a escolha óptima no caso de problemas de classificação (e na ausência de outros dados, como seja dados de validação). Não é possível desenvolver outros modelos sobre os mesmos dados que consigam menor erro de classificação.

Um maior detalhe deste tema poderá ser visto em Alpaydin (Alpaydin, 2004).

2.3. Curva ROC: Avaliação da capacidade discriminante de modelos

Existem diversas formas de medir o poder discriminante de um modelo mas, como refere Engelmann *et Al.* (e.g. Engelmann & Rauhmeier, 2006), o mais utilizados na prática serão a curva ROC e a curva CAP (*Cumulative Accuracy Profile*) (e.g. Basel Committee on Banking Supervision, 2005)).

Como foi referido anteriormente, um modelo preditivo pondera múltiplas variáveis (vector $\boldsymbol{X}$) para produzir uma pontuação final (*score*) relacionada com o níveis médios distintos da variável dependente $Z|\boldsymbol{X}=x$, $E(Z|\boldsymbol{X}=\boldsymbol{x})$. No caso de variáveis dependentes dicotómicas Y, o *score* $\hat{\pi}_x$ está indexado à proporção média p_x de eventos, observada em cada nível $\boldsymbol{X} = \boldsymbol{x}$.

Ordenando decrescentemente as observações por esse *score* $\hat{\pi}_x$ está-se tendencialmente a escolher primeiro os eventos e só posteriormente começarão a surgir os não-eventos.

2.3.1 Curva ROC, Modelo Perfeito e Modelo Aleatório

A curva ROC permite avaliar o poder classificativo do modelo, para duas classes, em cada nível $\hat{\pi}_x$ do *score* (pontos de corte situadas na curva), marcando como eventos todas as observações acima do *score* e como não-eventos todas as observações abaixo do *score* (e.g. Thomas, 2007). Desta forma é possível conhecer, em cada ponto de corte, a percentagem de eventos (do total de eventos) correctamente capturados - *Sensitivity* - e a percentagem de não-eventos (do total de não-eventos) correctamente classificados - *Specificity*. A curva ROC é o gráfico resultante da marcação de "*Sensitivity"* e "*1-Specificity*" para cada ponto de corte (figura 2.10)

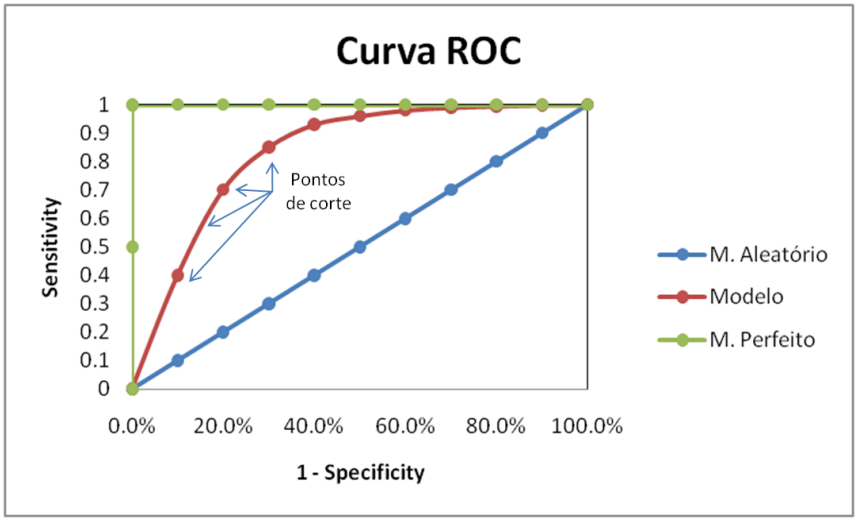


Figura 2.10 – Curva ROC

A par do desenho da curva ROC para o modelo específico em análise (a vermelho na figura 2.10) é usual representar-se, em simultâneo, um modelo aleatório (a diagonal a azul na figura 2.10) e um modelo perfeito (a verde no gráfico 2.10), sendo que o primeiro modelo não apresenta capacidade discriminante e o segundo produz separação total entre eventos e não-eventos. O objectivo destes dois modelos teóricos é o de tornar evidente a existência de um limite inferior e um limite superior à capacidade discriminante do modelo em análise.

Em termos de interpretação, quanto mais afastada estiver a curva do modelo em análise da diagonal maior será a capacidade discriminante do modelo. Na perspectiva inversa, a área que separa o modelo em análise do modelo perfeito representa o que o actual modelo ainda não consegue separar.

Como uma imagem vale por mil palavras, esta dinâmica encontra-se representada na figura 2.11

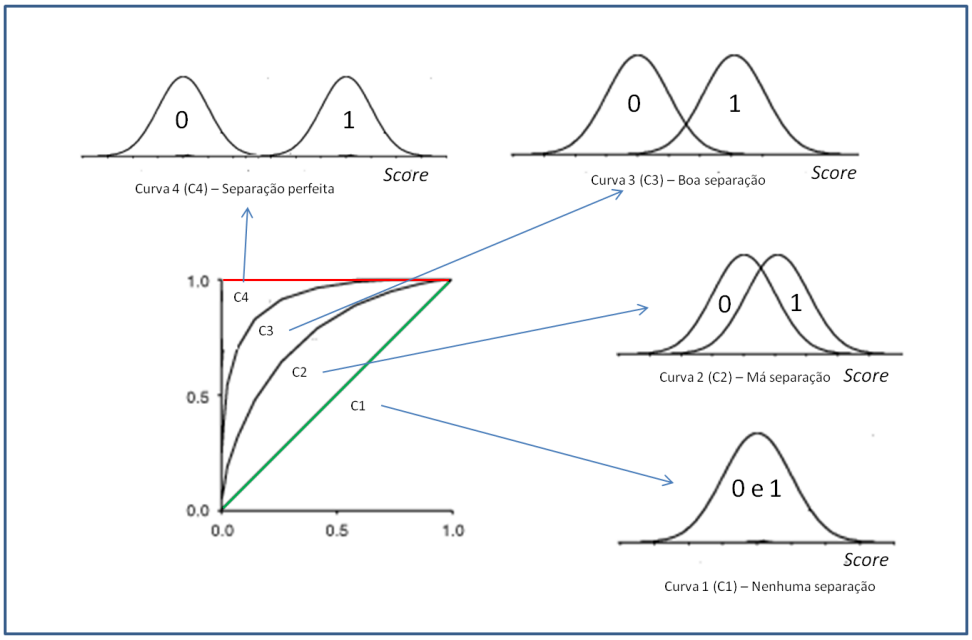


Figura 2.11 – Relação entre a curva ROC e a separação de duas populações (com base nos *scores*)

(adaptado de http://www-psych.stanford.edu/~lera/psych115s/notes/signal/)

2.3.2 Curva ROC, matriz de confusão e a tomada de decisão

Um dos resultados concretos que se pode retirar da análise da curva ROC é sobre eventuais "pontos de corte" óptimos conducentes à tomada de decisão, recorrendo-se à matriz de confusão para avaliar estas situações (e.g. Nisbet, Elder, & Miner, 2009). A título de exemplo, em análises clínicas, os resultados vêm muitas vezes sobre a forma de quantidades (ex: quantidade de PSA presente no sangue no teste de detecção de Cancro da Próstata), sendo importante definir limites a partir dos quais se "decide" que o paciente tem determinada doença e se actua em conformidade. É fundamental que esse ponto de corte seja definido com muito cuidado, para que seja detectada a maior percentagem de pessoas que se encontram doentes (eventos) mas, em simultâneo, evitar os chamados falsos alarmes (que conduzem a tratamentos onerosos e sobretudo prejudicais). Como a definição de pontos de corte está fora do âmbito do presente trabalho, uma breve exposição sobre o tema foi passado para apêndice (Apêndice A.1).

2.3.3. Curva ROC e a sua estatística resumo, o AUC

A qualidade do ponto de corte, visando contrabalançar custos e benefícios de determinada decisão, está altamente dependente do modelo conseguir discriminar

eventos de não-eventos na sua globalidade. O exercício de procura do ponto de corte tem de ser precedida de uma análise cuidada da efectividade do modelo em "separar as águas". Uma das estatísticas mais utilizadas para resumir globalmente a qualidade discriminante do modelo denomina-se AUC (*Area Under the (ROC) Curve*).

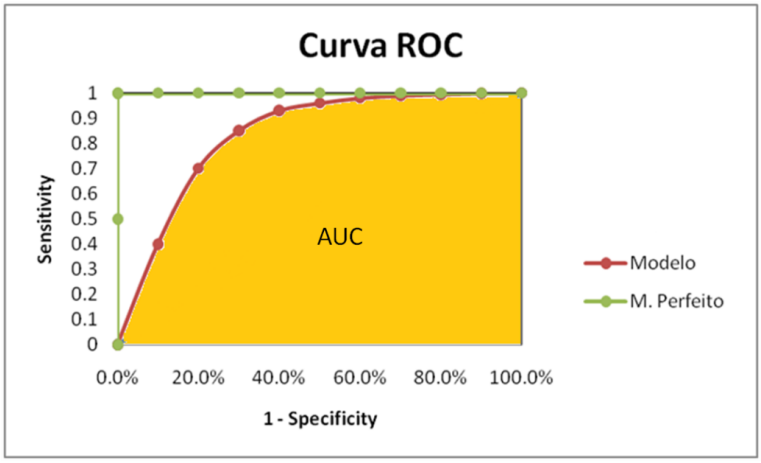


Figura 2.12 – AUC: *Area Under the ROC Curve*

Esta área tem uma interpretação probabilista e intuitiva. Engelman *et. Al* (e.g. Engelmann & Rauhmeier, 2006) demonstram que a área debaixo da curva ROC estima a probabilidade de num par [evento, não-evento], escolhido aleatoriamente entre todos os pares possíveis observados na amostra, o evento apresentar um *score* mais elevado que um não-evento.

Com base no exposto, estimar a área AUC pode ser encarado como um problema de combinatória. Separam-se os Eventos e os Não-Eventos (com os *scores* associados) em duas "urnas" distintas e procede-se a uma extracção aleatória de ambas as urnas para formar um par (evento, não-evento). Diz-se que:

a) O par é concordante quando $Score_{evento} > Score_{não-evento}$
b) O par é discordante quando $Score_{evento} < Score_{não-evento}$
c) O par é um empate (*tie*) quando $Score_{evento} = Score_{não-evento}$

para um número total de combinações (pares possíveis) igual a $n_{eventos} * n_{não-eventos}$.

Com base nesta informação poder-se-á calcular, por exemplo, a percentagem de pares concordantes ou a percentagem de pares discordantes com a afirmação "os eventos apresentam *scores* mais elevados que os não-eventos".

Estes são os conceitos que estão na base do cálculo da estatística U de Mann-Whitney que se apresenta de seguida.

2.3.4. A estatística U de Mann-Whitney

A estatística U de Mann-Whitney também conhecida como o Wilcoxon-Mann-Whitney (e.g. Marrocco, Duin, & Tortorella, 2008) é um estimador da AUC quando a variável resposta é binária (e.g. Engelmann & Rauhmeier, 2006).

Para se calcular a estatística U, agrupam-se as observações em **c** níveis distintos de *scores* $\hat{\pi}_i$ (número finito de *scores* produzidos por modelos) observados na amostra sendo que cada nível contém E_i eventos e $\bar{E}_i$ não-eventos. Considera-se $n_e = \sum_{i=1}^{c} E_i$ como representando o número total de eventos e $n_{\bar{e}} = \sum_{i=1}^{c} \bar{E}_\iota$ como representando o número total de não eventos e segue-se o protocolo (Hanley & McNeil, 1982):

a) Ordena-se a amostra decrescentemente pelos **c** níveis do *score* $\hat{\pi}_{(i)}$ com $i = 1, \dots, \boldsymbol{c}$ sendo que se tem $\hat{\pi}_{(1)} > \hat{\pi}_{(2)} > \cdots > \hat{\pi}_{(c)}$

Scores	Eventos	Não-Eventos
$\pi_{(1)}$	E_1	$\bar{E}_1$
$\pi_{(2)}$	E_2	$\bar{E}_2$
$\pi_{(3)}$	E_3	$\bar{E}_3$
$\pi_{(4)}$	E_4	$\bar{E}_4$
...	...	...
$\pi_{(C-1)}$	E_{C-1}	$\bar{E}_{C-1}$
$\pi_{(C)}$	E_C	$\bar{E}_C$
Total	n_e	$n_{\bar{e}}$

Figura 2.13 – Número de eventos e de não-eventos observados em cada nível de *score* $\pi_{(i)}$

b) Calcula-se número de pares concordantes (onde $Score_e > Score_{\bar{e}}$),

$$n_{conc} = \sum_{i=1}^{c-1} E_i * (\sum_{j=i+1}^{c} \bar{E}_j)$$

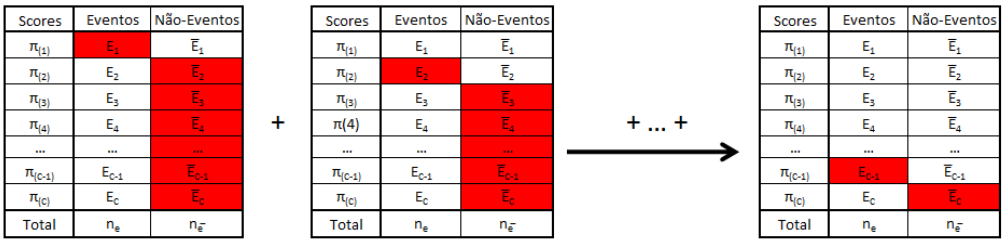


Figura 2.14 – Cálculo do número total de pares concordantes

c) Calcula-se o número de pares discordantes (onde $\text{Score}_e < \text{Score}_{\bar{e}}$),

$$n_{disc} = \sum_{i=1}^{c-1} \bar{E}_i * (\sum_{j=i+1}^{c} E_j)$$

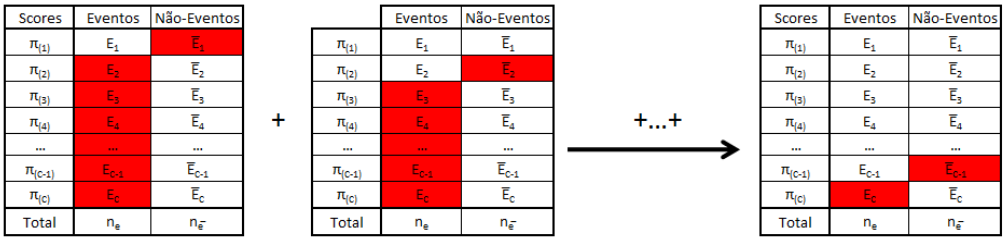


Figura 2.15 – Cálculo do número total de pares discordantes

d) Calcula-se o número de pares empatados (*tie*) ($\text{Score}_e = \text{Score}_{\bar{e}}$),

$$n_{tie} = \sum_{i=1}^{c} E_i * \bar{E}_i$$

Figura 2.16– Cálculo do número total de pares empatados

e) Calcula-se o número de pares $n_e * n_{\bar{e}}$ (evento, não-evento) passíveis de ser obtidos, sendo que $n_e * n_{\bar{e}} = n_{conc} + n_{disc} + n_{tie}$

$$\text{Pontuação para cada par} = \begin{cases} 1 & \text{se par concordante } (\text{Score}_e > \text{Score}_{\bar{e}}) \\ 0.5 & \text{se par empatado } (\text{Score}_e = \text{Score}_{\bar{e}}) \\ 0 & \text{se par discordante } (\text{Score}_e < \text{Score}_{\bar{e}}) \end{cases}$$

A estatística $U = \frac{n_{conc}+0.5*n_{tie}}{n_e*n_{\bar{e}}}$ é uma estimativa da AUC (Hanley & McNeil, 1982).

Em apêndice (Exemplo A.2) apresenta-se um exemplo prático muito simples de cálculo da estatística U.

A par da AUC, uma outra estatística muito utilizada é o denominado coeficiente de *Gini*, que representa a percentagem do modelo perfeito que é capturado pelo modelo em avaliação:

$$Coef.Gini = \frac{Area_A}{Area_A + Area_B} \text{ (figura 2.17)}$$

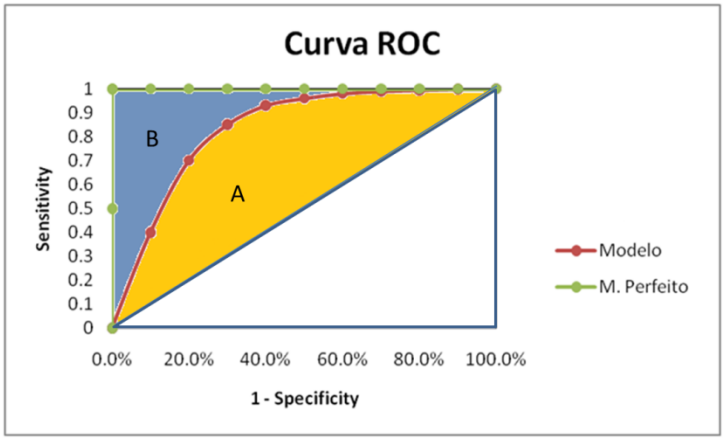


Figura 2.17 – Curva ROC e o cálculo do índice Gini

Thomas (e.g. Thomas, 2007) demonstra que $Coef.Gini = 2 * AUC - 1$.

2.3.5 Limitações do "Modelo perfeito"

Será o "modelo perfeito" a bitola adequada para uma comparação prática da qualidade de modelos desenvolvidos, o limite até onde se pode chegar?

O "modelo perfeito" sofre de dois problemas de percepção:

a) Implicitamente assume (ou cria as expectativa no analista) que o objectivo do modelo perfeito é atingível independentemente dos dados utilizados. Isto não é verdade, pois a qualidade dos dados é o principal "condicionador" da capacidade preditiva dos modelos.

b) Implicitamente assume que o objectivo do analista é extrair toda a capacidade discriminante dos dados, não tornando evidente que para se conseguir 100% capacidade discriminante nos dados de treino implica usualmente *overfitting* (se os dados não forem separáveis) e uma má capacidade de generalização.

Capítulo 3. Majoração da capacidade discriminante de um modelo

Neste capítulo propõe-se um limite prático, distinto do "modelo perfeito", para majoração da curva ROC (e da respectiva AUC) de qualquer modelo que seja desenvolvido num conjunto específico de dados e variáveis. O modelo associado a esse limite prático denominar-se-á de "Classificador Probabilista AP", e ao limite propriamente dito chamar-se-á de "capacidade discriminante dos dados". Neste capítulo derivam-se ainda algumas extensões e implicações da utilização do "Classificador Probabilista AP".

Nota importante: Relembra-se que, para facilitar a exposição e tornar menos repetitivo o discurso, no presente trabalho falar-se-á de capacidade discriminante ou capacidade preditiva de um modelo quando se calcula a curva ROC/AUC na amostra onde o modelo foi treinado e de capacidade de generalização quando se procede a uma *cross-validation*.

Definição : Define-se como Classificador Probabilista AP (*a posteriori*) o modelo que estima $P(Y = 1|\boldsymbol{X} = \boldsymbol{x})$, em cada ponto **X=x** observado no vector aleatório de variáveis independentes, baseado nas proporções empíricas calculadas directamente da amostra. O *score* produzido pelo Classificador Probabilista AP resulta do cálculo $p_x = \frac{E_x}{E_x + \bar{E}_x}$, a proporção empírica de eventos em cada ponto $\boldsymbol{X} = \boldsymbol{x}$ observado nos dados. Como é imediato, este *score* apenas se encontra definido nos pontos $\boldsymbol{X} = \boldsymbol{x}$.

O Classificador Probabilista AP deriva do Classificador MAP de *Bayes* apresentado no capítulo anterior:

$$\text{Classificador Probabilista AP:} \quad \hat{P}(Y = 1|\boldsymbol{X} = \boldsymbol{x}) = p_x$$

$$\text{Classificador MAP de } \textit{Bayes}\text{: } Se\ \hat{P}(Y = 1|\boldsymbol{X} = \boldsymbol{x}) \begin{cases} \geq 0.5 & então\ 1 \\ < 0.5 & então\ 0 \end{cases}$$

A função de *score* que suporta as classificações MAP é a mesma que suporta o Classificador Probabilista AP.

Importa definir, nesta fase, alguns conceitos chave e a respectiva terminologia a utilizar ao longo dos próximos capítulos:

$\boldsymbol{X}$ – Vector aleatório referente ao conjunto de variáveis independentes efectivamente utilizadas para o desenvolvimento de modelos.

$\boldsymbol{X} = \boldsymbol{x}$ – Valores concretos do vector aleatório **X** observados nos registos efectivamente utilizados para o desenvolvimento de modelos (como seja a amostra de treino e as variáveis *input*).

$E_x -$ Número de eventos ("1's") no ponto $\boldsymbol{X} = \boldsymbol{x}$.

$\bar{E}_x -$ Número de não-eventos ("0's") no ponto $\boldsymbol{X} = \boldsymbol{x}$.

$n_e -$ Número total de eventos.

$n_{\bar{e}} -$ Número total de não-eventos.

p_x – Proporção empírica de eventos observados na amostra de treino, em cada nível $\boldsymbol{X} = \boldsymbol{x}$. $p_x = \frac{E_x}{E_x + \bar{E}_x}$ é o *score* produzido pelo Classificador Probabilista AP na amostra de treino.

p_x^{teste} – Proporção empírica de eventos observados na amostra de teste, em cada nível $\boldsymbol{X} = \boldsymbol{x}$. Score produzido pelo Classificador Probabilista AP na amostra de teste.

$\hat{\pi}_x$ – *Score* produzido pelos modelos preditivos usuais (ex: regressão logística ou redes neuronais) no ponto $\boldsymbol{X} = \boldsymbol{x}$. Estimativa de $\pi_x = E(Y|\boldsymbol{X} = \boldsymbol{x})$ produzida por modelos.

ROC_{cp} – Curva ROC produzida pelo Classificador Probabilista AP

AUC_{cp} – AUC associada à curva $ROC_{cp.}$

ROC_M – Curva ROC produzida por modelos preditivos usuais.

AUC_M – AUC associada à curva ROC_M.

$p_{(i)}$ – Proporções empíricas / *Scores* p_x, calculados em cada ponto $\boldsymbol{X} = \boldsymbol{x}$, ordenados decrescentemente, primeiro por p_x e num segundo nível por E_x. Ter-se-á $p_{(1)} \geq p_{(2)} \geq \cdots \geq p_{(k)} \geq \cdots \geq p_{(s)} \geq \cdots \geq p_{(C)}$ porque poderão existir *scores* p_x idênticos para níveis distintos de $\boldsymbol{X} = \boldsymbol{x}$.

$E_{(i)}$ – Número de eventos ("1's") no *score* $p_{(i)}$

$\bar{E}_{(i)}$ – Número de não-eventos ("0's") no *score* $p_{(i)}$

Antes de se apresentar e demonstrar os principais teoremas da tese, estabelecem-se alguns resultados intermédios importantes para a sua demonstração.

Proposição 1: Considere-se as proporções p_x definido nos $\boldsymbol{c}$ níveis $\boldsymbol{X} = \boldsymbol{x}$ observados nos dados. Ordene-se p_x, obtendo-se as proporções $p_{(i)}(i = 1, \ldots, C)$, tal que $p_{(1)} \geq p_{(2)} \geq \cdots \geq p_{(k)} \geq \cdots \geq p_{(s)} \geq \cdots \geq p_{(C)}$.

$$\text{Quando } k < s\,,\ \textit{então}\ \ p_{(k)} \geq p_{(s)} \Leftrightarrow E_{(k)} * \bar{E}_{(s)} \geq E_{(s)} * \bar{E}_{(k)}$$

Demonstração: Tendo-se $p_{(1)} \geq p_{(2)} \geq \cdots \geq p_{(k)} \geq \cdots \geq p_{(s)} \geq \cdots \geq p_{(C)}$ e $k < s$ então:

$$p_{(k)} \geq p_{(s)} \Leftrightarrow \frac{p_{(k)}}{1 - p_{(k)}} \geq \frac{p_{(s)}}{1 - p_{(s)}} \Leftrightarrow$$

$$\Leftrightarrow \frac{E_{(k)}/\left(E_{(k)} + \bar{E}_{(k)}\right)}{1 - E_{(k)}/\left(E_{(k)} + \bar{E}_{(k)}\right)} \geq \frac{E_{(s)}/\left(E_{(s)} + \bar{E}_{(s)}\right)}{1 - E_{(s)}/\left(E_{(s)} + \bar{E}_{(s)}\right)} \Leftrightarrow$$

$$\Leftrightarrow \frac{E_{(k)}/\left(E_{(k)} + \bar{E}_{(k)}\right)}{\bar{E}_{(k)}/\left(E_{(k)} + \bar{E}_{(k)}\right)} \geq \frac{E_{(s)}/\left(E_{(s)} + \bar{E}_{(s)}\right)}{\bar{E}_{(s)}/\left(E_{(s)} + \bar{E}_{(s)}\right)} \Leftrightarrow$$

$$\Leftrightarrow \frac{E_{(k)}}{\bar{E}_{(k)}} \geq \frac{E_{(s)}}{\bar{E}_{(s)}} \Leftrightarrow E_{(k)} * \bar{E}_{(s)} \geq E_{(s)} * \bar{E}_{(k)} \quad QED.$$

Desta proposição retira-se facilmente que :

$$p_{(k)} = p_{(s)} \Leftrightarrow \frac{E_{(k)}}{\bar{E}_{(k)}} = \frac{E_{(s)}}{\bar{E}_{(s)}} \Leftrightarrow E_{(k)} * \bar{E}_{(s)} = E_{(s)} * \bar{E}_{(k)}$$

Proposição 2: A AUC_1 estimada associada à sequência de 3 scores $\hat{\pi}_{(1)} > \hat{\pi}_{(2)} > \hat{\pi}_{(3)}$ é algebricamente equivalente à AUC_2 estimada para a sequência de 4 scores, $\hat{\pi}_{(1)} > \hat{\pi}_{(2,1)} = \hat{\pi}_{(2,2)} > \hat{\pi}_{(3)}$, via separação de $\hat{\pi}_{(2)}$ em dois scores $\hat{\pi}_{(2,1)} = \hat{\pi}_{(2,2)}$, desde que as proporções empíricas observadas nesses "novos" *scores* sejam iguais ($p_{(2,1)} = p_{(2,2)}$).

Demonstração: Se $p_{(2,1)} = p_{(2,2)}$ então (da proposição 1) $E_{(2,1)} * \bar{E}_{(2,2)} = E_{(2,2)} * \bar{E}_{(2,1)}$.

Por outro lado têm-se $E_{(2)} = E_{(2,1)} + E_{(2,2)}$ e $\bar{E}_{(2)} = \bar{E}_{(2,1)} + \bar{E}_{(2,2)}$.

Utilizando a estatística U de Mann-Whitney, segue-se que $(n_e * n_{\bar{e}}) * AUC_1 =$

$$\left(E_{(1)} * \left(\bar{E}_{(2)} + \bar{E}_{(3)}\right) + E_{(2)} * \bar{E}_{(3)}\right) + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + E_{(2)} * \bar{E}_{(2)} + E_{(3)} * \bar{E}_{(3)}\right) =$$

$$= \left(E_{(1)} * \left(\left(\bar{E}_{(2,1)} + \bar{E}_{(2,2)}\right) + \bar{E}_{(3)}\right) + \left(E_{(2,1)} + E_{(2,2)}\right) * \bar{E}_{(3)}\right) + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + \left(E_{(2,1)} + E_{(2,2)}\right) * \left(\bar{E}_{(2,1)} + \bar{E}_{(2,2)}\right) + E_{(3)} * \bar{E}_{(3)}\right) =$$

$$= \left(E_{(1)} * \left(\bar{E}_{(2,1)} + \bar{E}_{(2,2)} + \bar{E}_{(3)}\right) + E_{(2,1)} * \bar{E}_{(3)} + E_{(2,2)} * \bar{E}_{(3)}\right) + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + E_{(2,1)} * \bar{E}_{(2,1)} + E_{(2,2)} * \bar{E}_{(2,2)} + E_{(2,2)} * \bar{E}_{(2,1)} + E_{(2,1)} * \bar{E}_{(2,2)} + E_{(3)} * \bar{E}_{(3)}\right) =$$

$$= \left(E_{(1)} * \left(\bar{E}_{(2,1)} + \bar{E}_{(2,2)} + \bar{E}_{(3)}\right) + E_{(2,1)} * \bar{E}_{(3)} + E_{(2,2)} * \bar{E}_{(3)}\right) + 0.5 * \left(E_{(2,2)} * \bar{E}_{(2,1)} + E_{(2,1)} * \bar{E}_{(2,2)}\right) + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + E_{(2,1)} * \bar{E}_{(2,1)} + E_{(2,2)} * \bar{E}_{(2,2)} + E_{(3)} * \bar{E}_{(3)}\right) =$$

$$= \left(E_{(1)} * \left(\bar{E}_{(2,1)} + \bar{E}_{(2,2)} + \bar{E}_{(3)}\right) + E_{(2,1)} * \bar{E}_{(3)} + E_{(2,2)} * \bar{E}_{(3)}\right) + 0.5 * \left(2 * E_{(2,1)} * \bar{E}_{(2,2)}\right) + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + E_{(2,1)} * \bar{E}_{(2,1)} + E_{(2,2)} * \bar{E}_{(2,2)} + E_{(3)} * \bar{E}_{(3)}\right) =$$

$= \left(E_{(1)} * \left(\bar{E}_{(2,1)} + \bar{E}_{(2.2)} + \bar{E}_{(3)}\right) + E_{(2,1)} * \left(\bar{E}_{(2,2)} + \bar{E}_{(3)}\right) + E_{(2,2)} * \bar{E}_{(3)}\right) + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + E_{(2,1)} * \bar{E}_{(2,1)} + E_{(2)} * \bar{E}_{(2,2)} + E_{(3)} * \bar{E}_{(3)}\right) =$

$= (n_e * n_{\bar{e}}) * AUC_2$ QED.

A extensão deste resultado a múltiplas separações é trivial (embora laborioso).

Este resultado é importante porque permite estabelecer a equivalência, para o cálculo da AUC do Classificador Probabilista AP, entre trabalhar com $p_{(1)}^{(A)} > p_{(2)}^{(A)} > \cdots > p_{(r)}^{(A)}$ (***r*** *scores* distintos, podendo cada *score* "Agregar" vários $\boldsymbol{X} = \boldsymbol{x}$ quando as proporções empíricas observadas em pontos distintos são idênticas e com a sequência $p_{(1)} \geq p_{(2)} \geq \cdots \geq p_{(C)}$ (sequência de *scores* mais granular possível de definir nos **c** pontos $\boldsymbol{X} = \boldsymbol{x}$ distintos observados na amostra).

3.1. Modelo Classificador Probabilista AP e a capacidade discriminante

3.1.1 Curva ROC do Classificador Probabilista AP

Teorema 1: A curva ROC do Classificador Probabilista AP majora a curva ROC de qualquer outro modelo desenvolvido sobre os mesmos dados.

Demonstração: *O score* p_x, definido nos **c** pontos $\boldsymbol{X} = \boldsymbol{x}$, quando devidamente ordenado (1º por p_x e numa 2ª ordem por E_x), traduz-se na sequência $p_{(1)} \geq p_{(2)} \geq \cdots \geq p_{(k)} \geq \cdots \geq p_{(s)} \geq \cdots \geq p_{(c)}$ que apresenta duas características fundamentais:

a) É a sequência mais granular de *scores* que é possível definir sobre os mesmos dados;

b) É a sequência que melhor ordena os Eventos, pela própria natureza da forma como foi definida acima.

Significa isto que a sequência "Cumulativa" $p^{(C)} = p_{(1)}^{(C)} \geq p_{(2)}^{(C)} \geq \cdots p_{(k)}^{(C)} \geq \cdots \geq p_{(s)}^{(C)} \geq \cdots \geq p_{(c)}^{(C)}$ é também a sequência cumulativa mais granular e que mais rapidamente captura os eventos presentes nos dados. Duas conclusões se tiram:

(1) Traduzindo esta realidade para uma curva ROC, qualquer sequência $\hat{\pi}^{(C)} = \hat{\pi}_{(1)}^{(C)} > \cdots > \hat{\pi}_{(k)}^{(C)} > \cdots > \hat{\pi}_{(s)}^{(C)} > \cdots > \hat{\pi}_{(r)}^{(C)}$ (com $\boldsymbol{r \leq c}$) de um modelo preditivo não pode ultrapassar os eventos capturados pela sequência $p^{(C)}$ para a mesma % de registos observados – a sequência $p^{(C)}$ não pode ser ultrapassada no eixo das ordenadas.

(2) Por outro lado como a sequência $p^{(C)}$ é a mais granular, obriga a que qualquer sequência $\hat{\pi}^{(c)}$ esteja definida num subconjunto dos pontos relativos à % registos observados.

Conjugando (1) e (2) demonstra-se o teorema QED.

3.1.2 AUC da curva ROC do Classificador Probabilista AP

Teorema 2: O modelo Classificador Probabilista AP cujo *score* p_x, está definido localmente para cada nível $\boldsymbol{X = x}$ observado nos dados, majora, em termos de AUC, qualquer modelo desenvolvido nos mesmos dados (registos e variáveis).

Demonstração:

Demonstrar que a AUC é máxima para o Classificador Probabilista AP equivale a demonstrar que estatística $U = \frac{n_{conc}+0.5*n_{tie}}{n_e * n_{\bar{e}}}$ do modelo não pode ser ultrapassada para

qualquer outro modelo que calcule *scores* com base na mesma informação (os mesmos pontos $\boldsymbol{X} = \boldsymbol{x}$).

De notar que para uma amostra específica se tem $n_e * n_{\bar{e}} = n_{conc} + n_{disc} + n_{tie}$ fixo. Significa isto que uma diminuição de n_{conc} implica uma transição para: a) n_{disc} ou b) n_{tie} , ambos correspondendo a uma diminuição de U. Logo demonstrar que AUC é máxima implica demonstrar que qualquer outro modelo cujos *scores* produzam alteração na ordem dos p_x diminui ou mantém n_{conc}.

$$n_{conc} = \sum_{i=1}^{c-1} E_i * (\sum_{j=i+1}^{c} \bar{E}_j)$$

Considere-se o *score* $p_x$definido nos $\boldsymbol{c}$ níveis distintos $\boldsymbol{X} = \boldsymbol{x}$ observados nos dados. Ordene-se p_x obtendo-se $\boldsymbol{c}$ níveis de *score* $p_{(i)}(i = 1, \dots, C)$, tal que $p_{(1)} \geq p_{(2)} \geq \cdots \geq p_{(k)} \geq \cdots \geq p_{(s)} \geq \cdots \geq p_{(C)}$ $(onde\ k < s)$

Suponha-se agora um outro modelo com *scores* $\hat{\pi}_x$, definidos nos mesmos pontos $\boldsymbol{X} = \boldsymbol{x}$ e alinhados em termos de ordem com p_x, à excepção da troca de dois pontos $p_{(k)}$ com $p_{(s)}$. Fazendo-se a expansão de n_{conc} para ambos os modelos (detalhado no apêndice 3), e lembrando que $k < s$, verifica-se que as parcelas são idênticas à excepção das referidas na seguinte tabela.

Parcelas únicas na expansão de n_{conc} para p_x	**Parcelas únicas na expansão de n_{conc} para $\hat{\pi}_x$**
$E_{(k)} * (\bar{E}_{(k+1)} + \cdots + \bar{E}_{(s)})$	$E_{(s)} * (\bar{E}_{(k+1)} + \cdots + \bar{E}_{(k)})$
$E_{(k+1)} * \bar{E}_{(s)}$	$E_{(k+1)} * \bar{E}_{(k)}$
$E_{(k+2)} * \bar{E}_{(s)}$	$E_{(k+2)} * \bar{E}_{(k)}$
…	…

$E_{(s-1)} * \bar{E}_{(s)}$	$E_{(s-1)} * \bar{E}_{(k)}$

Figura 3.1 – Parcelas distintas na expansão de $\boldsymbol{n_{conc}}$ para $\boldsymbol{p_x}$ e $\boldsymbol{n_{conc}}$ para $\boldsymbol{\hat{\pi}_x}$

Sendo n_{conc} uma soma de parcelas positivas e retirando-se, da proposição 1, que cada parcela da 1ª coluna (da figura 3.1) é maior ou igual a cada parcela da 2ª coluna (da figura 3.1), conclui-se que n_{conc} de $p_x \geq n_{conc}$ de $\hat{\pi}_x$ QED.

Visualmente, os dois teoremas aqui apresentados traduzem-se na figura 3.1

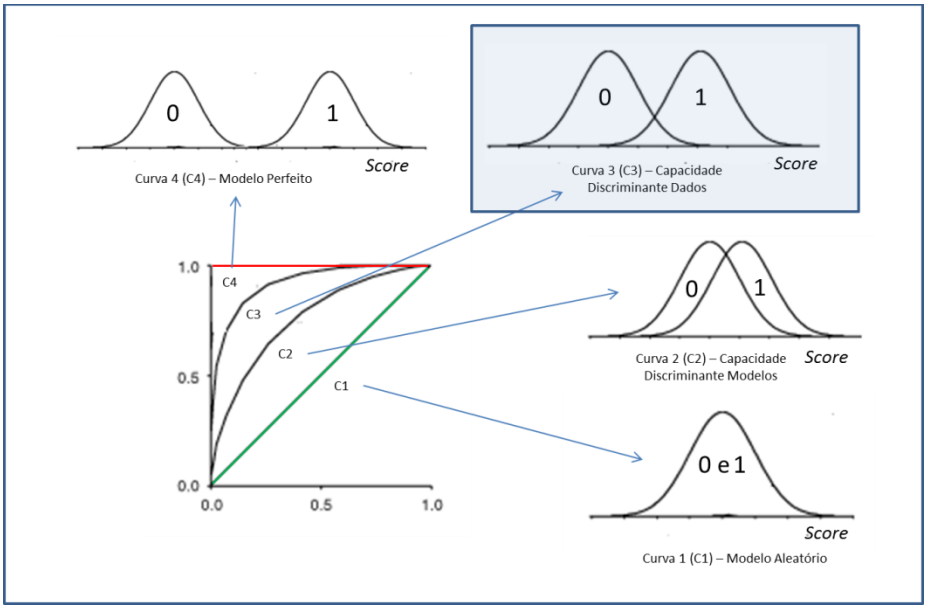


Figura 3.2 – Representação da capacidade discriminante dos dados

A curva ROC C3 associada ao Classificador Probabilista AP majora qualquer outra curva ROC de modelos desenvolvidos sobre os mesmos dados. A máxima separação possível das duas populações (1's e 0's) com base nos dados em concreto está destacada pelo rectângulo azul.

3.2. Índice CEDP - Proposta

Tal como o índice de *Gini* compara, na curva ROC, a área debaixo de um determinado modelo com a área associada do modelo perfeito, fará todo o sentido propor um outro índice que proceda da mesma forma em relação à capacidade discriminante dos dados.

Esse índice pode-se denominar como "índice CEDP" (*Captured Empirical Disciminant Power*)" de um modelo e terá a seguinte formulação:

$$CEDP_M = \frac{2 * AUC_M - 1}{2 * AUC_{CP} - 1}$$

onde AUC_M corresponde à AUC do modelo desenvolvido e AUC_{CP} corresponde à AUC do modelo Classificador Probabilista AP baseado nas mesmas variáveis.

3.4. Classificador Probabilista AP e capacidade de generalização (numa amostra de teste)

Principio: O Modelo Classificador probabilista AP desenvolvido sobre a amostra de treino, quando aplicado a novos dados traduzir-se-á (quase certamente), não só, numa AUC inferior ao obtido na amostra de treino, mas também numa AUC inferior à AUC obtida por outros modelos quando aplicada aos novos dados.

Explicação: O Classificador Probabilista AP faz um ajustamento perfeito à informação observada nos dados de treino, o que, em caso de dados não separáveis, se traduz usualmente em *overfitting*, donde não se estranha a sua teórica fraca capacidade de generalização.

Nota: Para se aplicar o Classificador Probabilista AP a novos dados, agrega-se o ficheiro de treino pelos níveis individuais de $\boldsymbol{X}$=$\boldsymbol{x}$ e calcula-se os p_x. Faz-se posteriormente o cruzamento dos ficheiros via $\boldsymbol{X}$ (utilizando, por exemplo, o SQL) e aplica-se directamente os p_x a cada observação da amostra de teste. Para os níveis $\boldsymbol{X}$=$\boldsymbol{x}$ da amostra de novos dados para os quais não existe correspondência na amostra de treino, atribui-se um *score* constante equivalente à taxa média da amostra de treino.

3.5. Classificador Probabilista AP aplicado sobre a amostra de teste e a majoração da generalização

Lema: O Classificador Probabilista AP calculado directamente numa amostra de novos dados (para um conjunto específico de variáveis) e produzindo *scores* p_x^{teste} apresenta um AUC que majora o AUC de qualquer modelo testado nesses dados e que tenha sido desenvolvido noutros dados (como seja o ficheiro de treino) desde que com base nas mesmas variáveis.

Demonstração: Como já se demonstrou anteriormente, a ordenação aplicada à variável dependente pelo Classificador Probabilista AP desenvolvido sobre a própria amostra maximiza a AUC.

3.6. Granularidade da informação e monotonia não-decrescente da capacidade discriminante do Classificador Probabilista AP

Lema: O AUC_{CP} do Classificador Probabilista AP é monótono não-decrescente à medida que aumenta a granularidade da informação, isto é, à medida que se aumenta o número de níveis $\boldsymbol{X} = \boldsymbol{x}$ do vector de variáveis independentes.

Demonstração: Far-se-á a demonstração para o caso mais simples, embora a extensão para cenários mais complexos seja trivial (embora laboriosa).

Considere-se um ficheiro com $n = n_e + n_{\bar{e}}$ registos, onde após discretização de uma variável intervalar (ex: Idade) se observam apenas dois pontos (**c**=2,correspondendo a, por exemplo, Idade<=30 vs. Idade>30), e obtendo-se os seguintes *scores* $p_{(1)} > p_{(2)}$ (em que, para efeitos de ilustração, o $p_{(2)}$ corresponde a Idade>30) e calcula-se a AUC_1.

Considere-se agora que se categoriza a mesma variável, em 3 níveis (ex: Idade <=30, 30<Idade<=40 e Idade>40), via divisão do $p_{(2)}$ em dois *scores* $p_{(2,1)}$ *e* $p_{(2,2)}$.

Três possibilidades se colocam:

a) $p_{(1)} > p_{(2,1)} = p_{(2,2)}$: Da proposição 2, AUC_1=AUC_2.

b) $p_{(2,1)} > p_{(1)} > p_{(2,2)}$ ou $p_{(2,2)} > p_{(1)} > p_{(2,1)}$: Uma vez que a ordenação imposta por $p_{(1)} > p_{(2)}$ deixou de ser óptima numa granularidade superior, pelos teoremas 1 e 2 tem-se AUC_1<AUC_2.

c) $p_{(1)} > p_{(2,1)} > p_{(2,2)}$: Aqui não houve alterações de ordenação, onde se impõe uma demonstração analítica.

$$(n_e * n_{\bar{e}}) * AUC_1 = E_{(1)} * \bar{E}_{(2)} + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + E_{(2)} * \bar{E}_{(2)}\right) =$$

$$= E_{(1)} * (\bar{E}_{(2,1)} + \bar{E}_{(2,2)}) + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + \left(E_{(2,1)} + E_{(2,2)}\right) * (\bar{E}_{(2,1)} + \bar{E}_{(2,2)})\right) =$$

$$= E_{(1)} * \bar{E}_{(2,1)} + E_{(1)} * \bar{E}_{(2,2)} + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + \left(E_{(2,1)} * \bar{E}_{(2,1)} + E_{(2,1)} * \bar{E}_{(2,2)} + E_{(2,2)} * \bar{E}_{(2,1)} + E_{(2,2)} * \bar{E}_{(2,2)}\right)\right)$$

$$(n_e * n_{\bar{e}}) * AUC_2 = E_{(1)} * \left(\bar{E}_{(2,1)} + \bar{E}_{(2,2)}\right) + E_{(2,1)} * \bar{E}_{(2,2)} + 0.5 * \left(E_{(1)} * \bar{E}_{(1)} + E_{(2,1)} * \bar{E}_{(2,1)} + E_{(2,2)} * \bar{E}_{(2,2)}\right)$$

$$(n_e * n_{\bar{e}}) * (AUC_1 - AUC_2) = 0.5 * \left(E_{(2,1)} * \bar{E}_{(2,2)} + E_{(2,2)} * \bar{E}_{(2,1)}\right) - E_{(2),1} * \bar{E}_{(2,2)} =$$

$= 0.5 * \left(E_{(2,2)} * \bar{E}_{(2,1)} - E_{(2,1)} * \bar{E}_{(2,2)}\right) < 0$ donde, ($Proposição$ 1), AUC_1<AUC_2. QED.

3.7. Granularidade da informação e a tendência decrescente da capacidade de generalização do Classificador Probabilista AP

Principio: O Classificador Probabilista AP da amostra de treino, quando aplicado a novos dados produzirá tendencialmente um AUC mais baixo que o observado na amostra de treino. Esta diferença tenderá a ser tanto maior quanto maior o número de níveis $\boldsymbol{X} = \boldsymbol{x}$ do vector de variáveis independentes.

Explicação: O Classificador Probabilista AP faz *overfitting* aos dados treino, extraindo toda a sua capacidade discriminante, donde não se estranha a sua teórica fraca capacidade de generalização.

O efeito do *overfitting* tende a piorar à medida que o número de níveis de $\boldsymbol{X} = \boldsymbol{x}$ observado na amostra de treino aumenta. Se assim não fosse, significaria que os *scores* p_x preservavam determinada capacidade de generalização (sob a forma da manutenção da ordenação em ambas as amostras) para cada vez mais estimativas p_x (referentes a mais patamares $\boldsymbol{X} = \boldsymbol{x}$) estimadas com menor qualidade (uma vez que, fixada a dimensão da amostra, o número de observações em cada nível $\boldsymbol{X} = \boldsymbol{x}$ tende a diminuir com o aumento das dimensões do vector **X**), o que é manifestamente um contra-senso.

Capítulo 4. Visualização empírica das hipóteses enunciadas

Neste capítulo apresenta-se resultados empíricos que ilustram as conclusões obtidas no capítulo anterior. Na abordagem a cada hipótese, inicia-se o processo considerando usualmente casos mais simples, como sejam modelos preditivos univariados, complexificando-se posteriormente os exemplos via uma abordagem multivariada.

4.1 Classificador Probabilista AP e a majoração da capacidade discriminante

O objectivo deste tópico é ilustrar como a curva ROC (e a respectiva AUC) do Classificador Probabilista AP majora a curva ROC (e a respectiva AUC) de outros modelos desenvolvidos sobre os mesmos dados.

4.1.1 Modelo univariado com uma variável independente binária

Variável independente = Variável binária posse de cartão de crédito, que toma o valor 0 quando o Cliente não tem cartão e o valor 1 quando o Cliente tem cartão.

Para se ajustar o Classificador Probabilista AP sumariza-se a informação ao nível dos pontos $\boldsymbol{X} = \boldsymbol{x}$ observados nos dados e calculam-se os p_x (figura 4.1)

	Classificador Probabilista AP		
Cartão	# Eventos	# Não-Eventos	p_x
0	613	14880	0.03956626
1	131	7440	0.01730287

Figura 4.1 – Ajustamento do Classificador Probabilista AP aos dados com a variável independente "posse cartão crédito"

Esta tabela permite a apresentação de um gráfico de frequências mas também da curva ROC do Classificador probabilista AP (ambas apresentadas na figura 4.2).

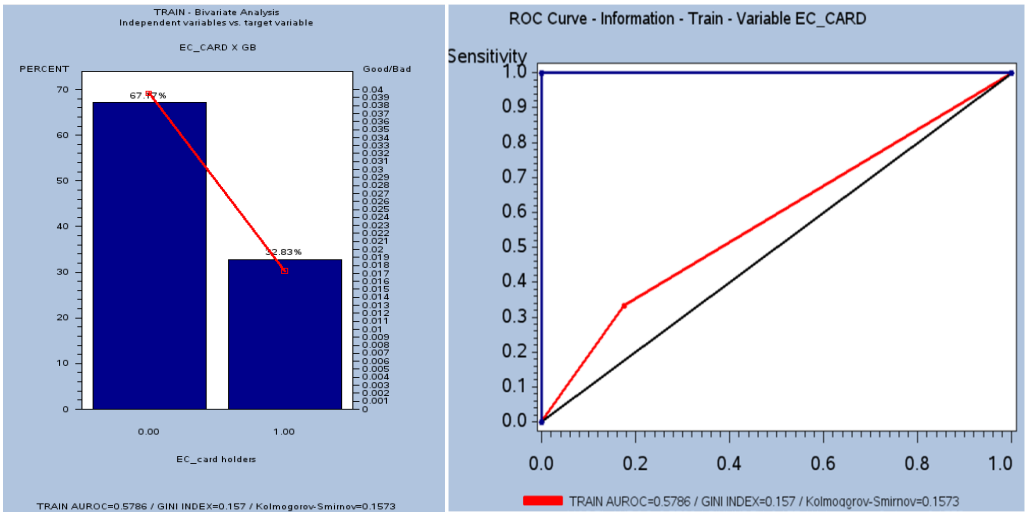


Figura 4.2 – Ajustamento do Classificador Probabilista AP a dados com apenas uma variável independente binária (posse cartão crédito)

Utilizando a estatística U de Mann-Whitney estima-se a AUC da curva ROC, que é 0.5786 (o que pode ser visto em rodapé na figura 4.2).

Em paralelo desenvolveu-se um modelo preditivo sobre a mesma informação, mas recorrendo a Redes Neuronais. Intuitivamente, e quando na presença de um problema com apenas uma variável independente binária, antecipa-se que qualquer modelo desenvolvido sobre esta mesma informação irá atribuir *scores* $\hat{\pi}_x$ iguais a p_x .Nas figuras 4.3 resumem-se os resultados da aplicação da Rede Neuronal aos mesmos dados.

	Modelo Rede Neuronal			
Cartão	Score π_x	# Eventos	# Não-Eventos	p_x
0	0.039566	613	14880	0.039566
1	0.017303	131	7440	0.017303

Figura 4.3 - Ajustamento de uma Rede Neuronal a dados com apenas uma variável independente binária (posse cartão crédito) e comparação do score $\hat{\pi}_x$ com o score empírico p_x

Como o *score* de um modelo estatístico baseado apenas numa variável binária independente preserva a ordenação das proporções empíricas, as curvas ROC_M e ROC_{CP}

coincidem exactamente, com a estatística AUC_M da rede neuronal a ser igual à AUC_{CP} da informação (Figura 4.4)

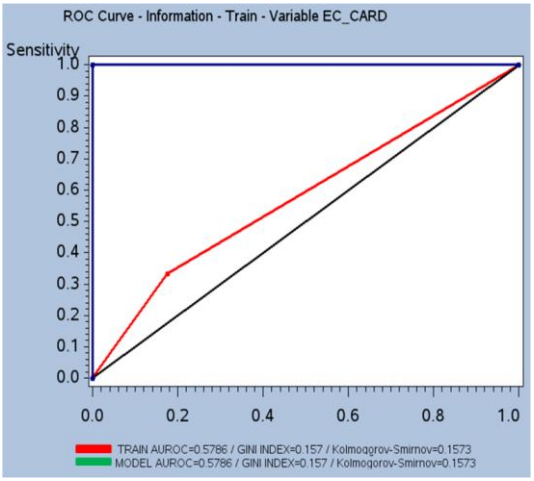


Figura 4.4 – Curva ROC da Rede Neuronal

4.1.2 Modelo univariado com uma variável independente intervalar

Variável independente intervalar = Idade do Cliente (em anos)

Repete-se o procedimento do tópico anterior, sumarizando-se a informação ao nível da granularidade máxima da informação, calculando-se o número de eventos, o número de não-eventos, as proporções empíricas de eventos p_x e ordenando-se decrescentemente (figura 4.5).

	Classificador Probabilista AP		
Idade	# Eventos	# Não-Eventos	p_x
22	47	300	0.135447
20	30	270	0.1
26	53	600	0.081164
23	43	630	0.063893
19,27	69	1140	0.057072
24,25	88	1620	0.051522
...	...	...	...
59	1	240	0.004149
61,71	0	210	0

Figura 4.5 - Ajustamento do Classificador Probabilista AP aos dados (variável independente "idade")

De forma equivalente poder-se-á representar esta tabela em gráfico de frequências (Figura 4.6), o que torna mais intuitiva a visualização e interpretação da dinâmica subjacente aos dados. A azul, no diagrama, surge a frequência relativa do número de Clientes com determinada idade sendo os p_x representados pela linha vermelha.

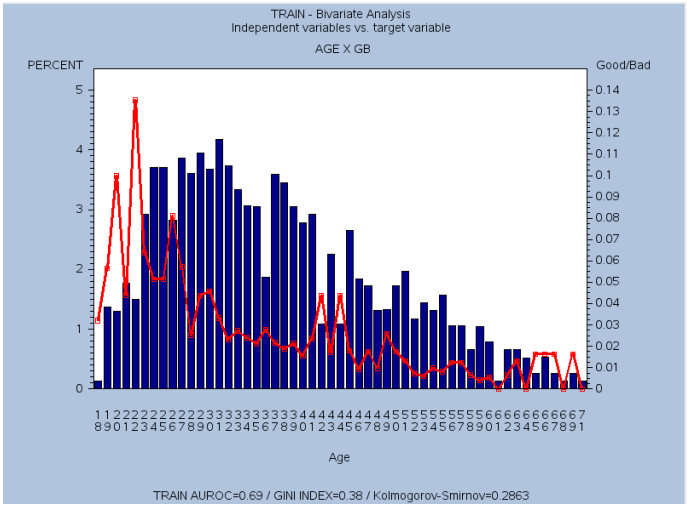


Figura 4.6 – Diagrama de frequências e as respectivas proporções empíricas referentes a cada nível da variável "idade"

A tabela obtida permite calcular o AUC_{CP} que é de 0.69 (ver rodapé figura 4.6), a capacidade total de separação do Classificador Probabilista AP baseado na variável idade.

Entretanto também se estimou um modelo com base em redes neuronais, sendo interessante comparar os *scores* produzidos por ambos os modelos para os mesmos pontos $\boldsymbol{X} = \boldsymbol{x}$.

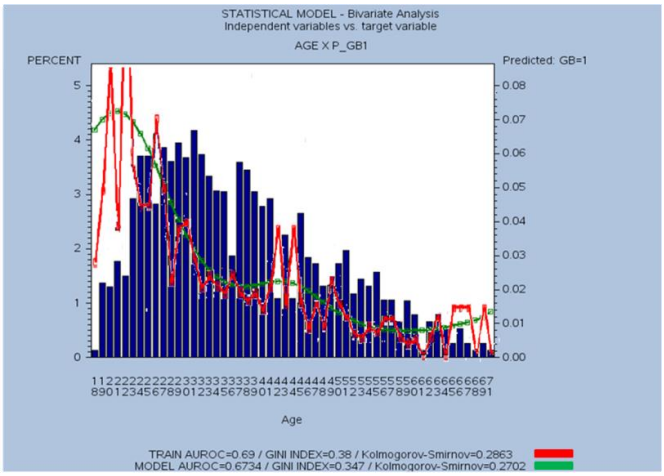


Figura 4.7 – Sobreposição dos *scores* produzidos por uma rede neuronal (a verde) sobre *scores* empíricos (a vermelho)

Olhando para a figura 4.7, a verde surgem os *scores* $\hat{\pi}_x$ da rede neuronal e a vermelho os *scores* p_x produzidos pelo Classificador Probabilista. Como seria de esperar a função estimada pela rede neuronal é muito mais suave que a curva do Classificador Probabilista AP, mas não ordena de forma tão eficiente as proporções empíricas p_x na amostra de treino. Esta realidade pode ser vista ver formalmente pela figura 4.8 (melhores 6 *scores* de p_x *e* $\hat{\pi}_x$e como se encontram alinhados). Registe-se que a rede neuronal "ordena mal" logo no a partir do 1º *score*.

	Classificador Probabilista AP		
Idade	# Eventos	# Não-Eventos	p_x
22	47	300	0.135447
20	30	270	0.1
26	53	600	0.081164
23	43	630	0.063893
19,27	69	1140	0.057072
24,25	88	1620	0.051522
...	...	...	...

	Rede Neuronal			
Idade	Scores π_x	# Eventos	# Não-Eventos	p_x
21	0.073	18	390	0.044118
20,22	0.072	77	570	0.119011
19	0.07	18	300	0.056604
23	0.069	43	630	0.063893
18	0.067	1	30	0.032258
24	0.066	44	810	0.051522
...	...	...	...	...

Figura 4.8 – Relação entre os scores p_x *e* $\hat{\pi}_x$ produzidos pelo Classificador Probabilista e por uma Rede Neuronal

Mais uma vez olhando para o rodapé da figura 4.7 verifica-se o que se esperava: o AUC_{CP} é superior ao AUC_M da rede neuronal (0.69 vs. 0.6734).

Desenhando a curva ROC para ambos os modelos obtém-se o gráfico 4.9

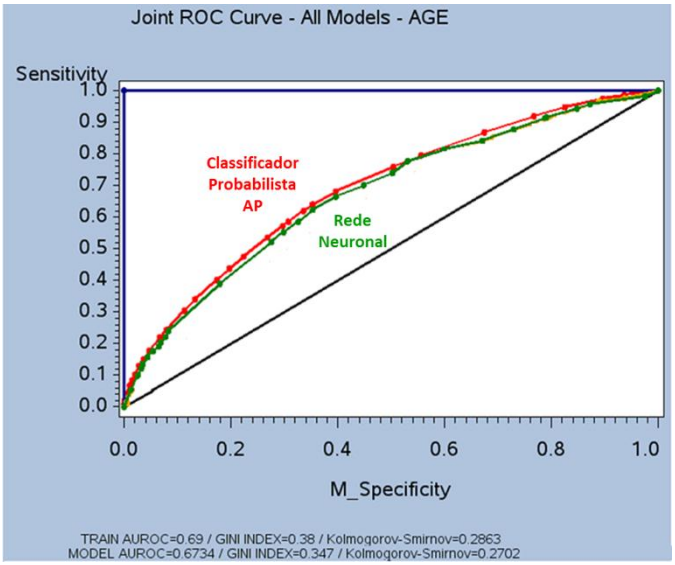


Figura 4.9 – Curvas ROC do Classificador Probabilista AP versus Rede Neuronal (variável idade)

Observa-se a majoração da curva ROC (verde) referente à rede neuronal pela curva ROC (vermelha) do Classificador Probabilista AP

Em termos de CEDP para a rede neuronal, ter-se-á $CEDP_M$=(2*0.6734-1)/(2*0.69-1)=91.2%, que contrasta sobremaneira com o índice de *Gini* de 34.68% (que indica a percentagem do "Modelo Perfeito" capturado pela rede neuronal).

A título de exemplo, como se comportou a regressão logística?

Sendo um método menos flexível que as redes neuronais (e que o Classificador Probabilista AP) espera-se uma função monótona mais bem comportada (a verde na figura 4.10) que qualquer uma das anteriores (comparar figura 4.7)

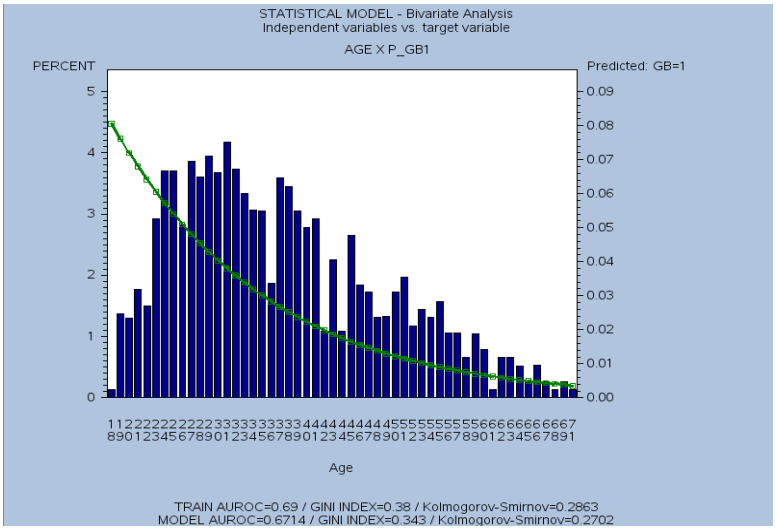


Figura 4.10 – Ajustamento da regressão logística aos dados e os *scores* obtidos (linha verde)

Ao sobrepor-se o Classificador Probabilista AP intui-se que este modelo ainda deva perder mais capacidade de ordenação que rede neuronal, o que se confirma pelo AUC_M da regressão logística de 0.6714 (<0.6734 da RN e <0.69 do Classificador) e que pode ser visto no rodapé da figura 4.10.

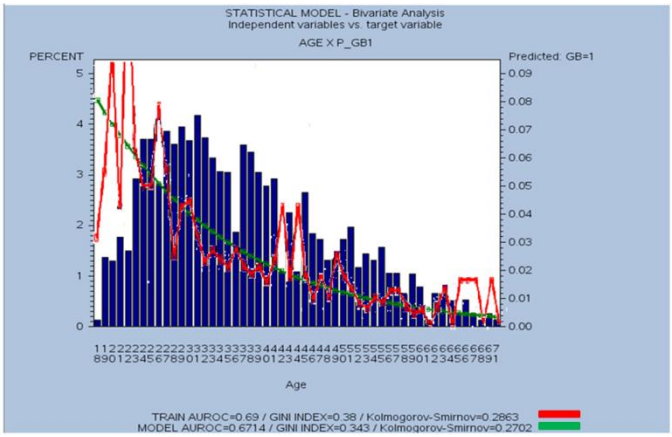


Figura 4.11- Sobreposição do Classificador Probabilista AP sobre os resultados obtidos pela regressão logística

Em termos de CEDP, observa-se o $CEDP_M$=(2*0.6714-1)/(2*0.69-1)=90.2%

4.1.3 Modelo multivariado com cinco variáveis independentes

Variáveis independentes = Idade do Cliente / Posse cartão de crédito / Informação na CRC / Número de pessoas no agregado familiar / Estado Civil

Para se poder ter uma perspectiva comparativa das curvas ROC e das respectivas AUC, relativas ao Classificador Probabilista AP e rede neuronal/regressão logística, em cenários multivariados, considerou-se alargar o ficheiro a 5 variáveis independentes, onde duas delas são já as variáveis trabalhadas no caso univariado (Idade e posse de cartão de crédito). A curva ROC surge na figura 4.12.

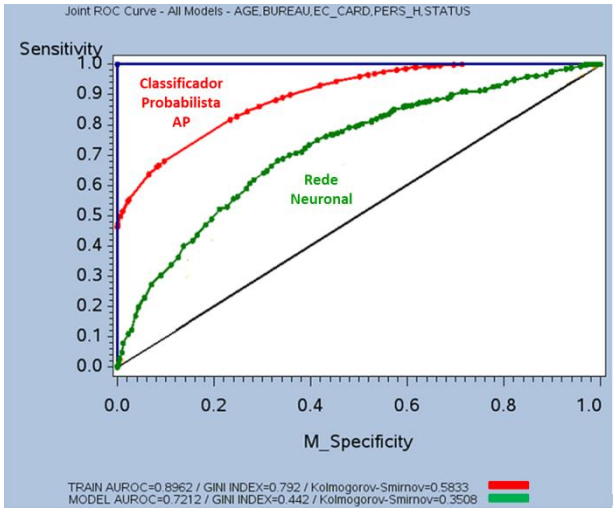


Figura 4.12 – Curva ROC para Classificador Probabilista AP (a vermelho) e Rede Neuronal (a verde) tendo por base 5 variáveis independentes.

Observa-se então uma AUC_{CP}=0.8962 versus uma AUC_M=0.7212, para um $CEDP_M$=55% (que compara com 44.2% de coeficiente de *Gini*).

Um dos aspectos que importa destacar nesta fase é o seguinte: da abordagem univarida (só com a variável idade) para a abordagem multivariada, a AUC_{CP} aumenta de 0.69 para 0.8962, por contraponto à AUC_M , que passou de 0.6734 para 0.7212. Apesar da "capacidade discriminante dos dados" aumentar cerca de 20 p.p. (de 0.69 para 0.8962), a capacidade discriminante do modelo aumenta menos de 5 p.p (de 0.6734 para 0.7212). Esta observação conduzirá a propostas para investigação posterior.

4.2. Capacidade de generalização do Classificador Probabilista AP

O objectivo deste tópico é ilustrar como, tendencialmente, o Classificador Probabilista AP desenvolvido sobre determinados dados de treino e aplicado a uma amostra de teste apresenta um AUC inferior (na amostra de teste) a outros modelos preditivos desenvolvidos sobre os mesmos dados de treino e aplicados ao mesmo ficheiro de teste.

4.2.1 Modelo univariado com uma variável independente intervalar

Tal como no tópico anterior, aborda-se faseadamente a ilustração desta dinâmica, iniciando-se o processo com uma só variável independente – a variável idade. Mais uma vez se calcula o diagrama de frequência, as respectivas proporções empíricas p_x^{teste} na amostra de teste (a laranja na figura 4.13).

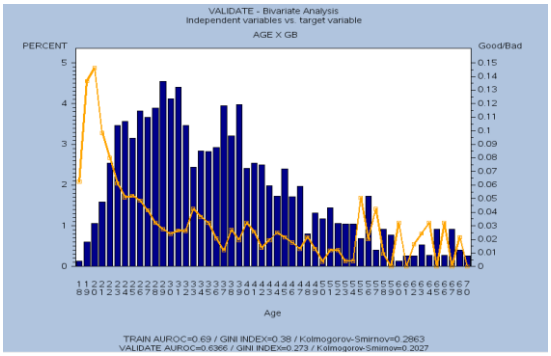


Figura 4.13 – Proporções empíricas p_x^{teste} (a laranja)

Aplicando-se os *scores* p_x (calculados na amostra de treino) aos dados de teste e sobrepondo-se essa informação (a vermelho) à curva dos p_x^{teste} (a laranja), obtêm-se o seguinte gráfico (figura 4.14)

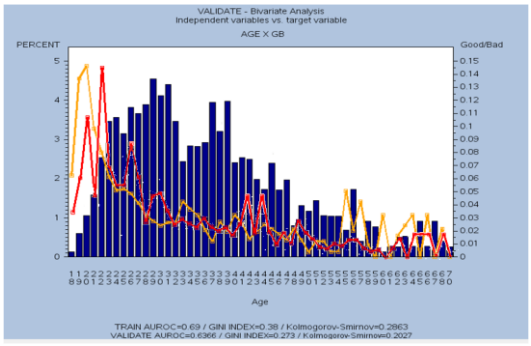


Figura 4.14 – Aplicação do Classificador Probabilista AP sobre a amostra de teste. Comparação dos p_x com p_x^{teste}

Visualiza-se um desencontro entre as curvas, especialmente em áreas onde existem poucas observações. De facto o que se está a comparar, na figura 4.14, são as taxas de eventos entre a amostra de treino (via *score* p_x) e as taxas de eventos entre a amostra de teste (via *score* p_x^{teste}). Esse desencontro em termos de taxa de eventos, com implicação na ordenação da variável dependente, traduz-se num decréscimo da AUC_{CP} de 0.69 na amostra de treino para 0.6366 na amostra de validação (ver rodapé da figura 4.15). Ilustra-se este decréscimo visualmente via curvas ROC, embora não seja usual representar, num mesmo gráfico, curvas ROC para modelos desenvolvidos em ficheiros distintos (que é o que sucede na figura 4.15).

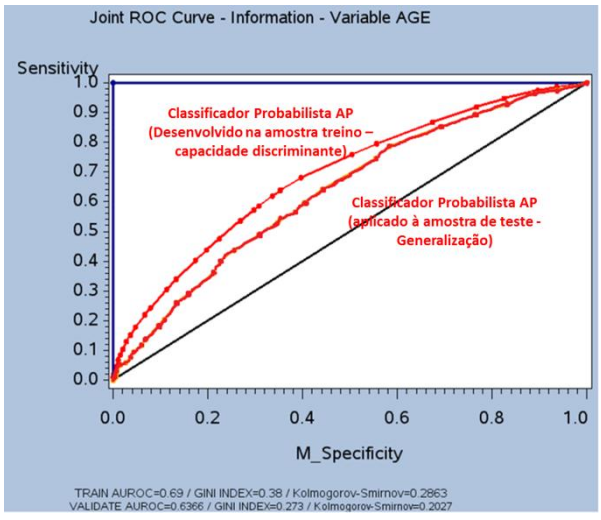


Figura 4.15 – Capacidade discriminante versus capacidade de generalização do Classificador Probabilista AP

Como se comporta a rede neuronal em termos de capacidade de generalização?

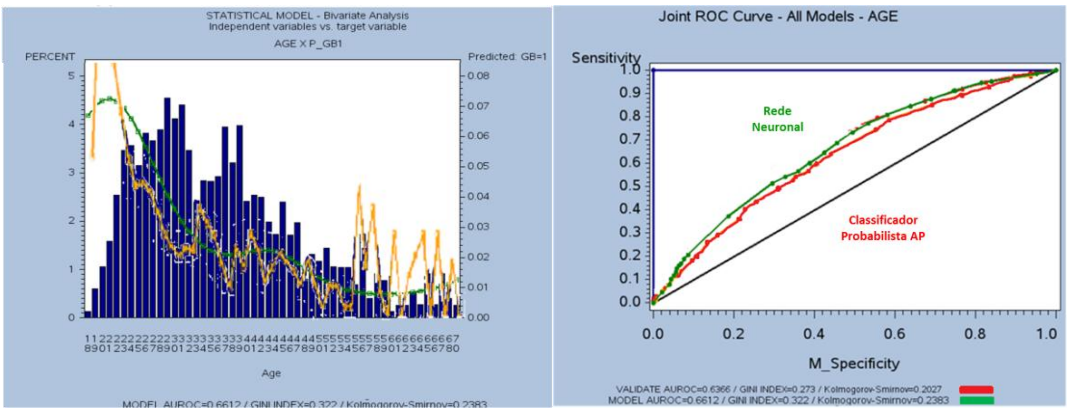


Figura 4.16 – Capacidade de generalização da rede neuronal versus Classificador Probabilista AP

O facto da curva dos *scores* $\hat{\pi}_x$ (parte esquerda da figura 4.16) ser substancialmente menos volátil que o Classificador Probabilista AP (a vermelho na figura 4.14) torna a rede neuronal tendencialmente mais robusta no que toca a capacidade de generalização.

Não é de estranhar a menor degradação da capacidade discriminante (0.6734) para a capacidade de generalização (0.6612) da rede neuronal - correspondendo a 1.8 p.p. de perda - relativamente à perda associada ao Classificador Probabilista (de 0.69 para 0.6366), correspondendo a 6 p.p. de perda.

Será que se verifica o mesmo em relação às previsões da regressão logística?

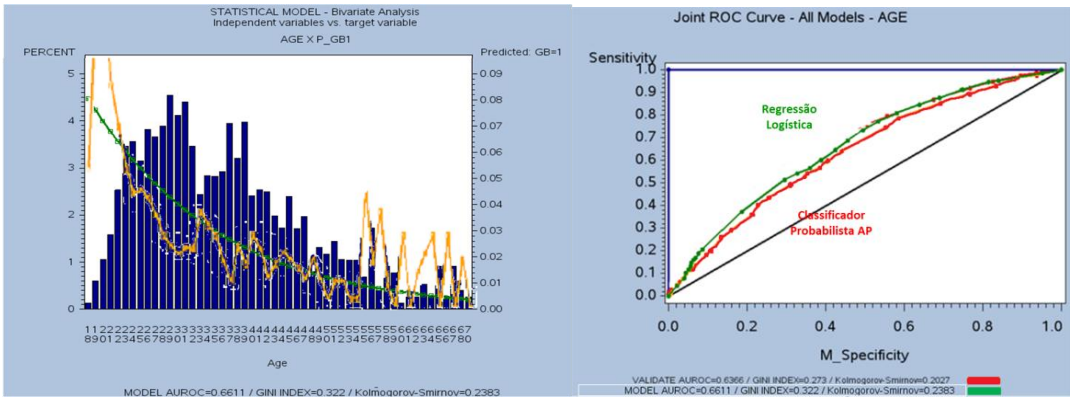


Figura 4.17 – Capacidade de generalização da regressão logística versus Classificador Probabilista AP

A regressão logística apresenta um comportamento semelhante à rede neuronal, classificando-se sempre acima do Classificador Probabilista AP embora com AUC_M ligeiramente inferiores (0.6714 para 0.6611) à Rede Neuronal.

4.2.2 Modelo multivariado com cinco variáveis independentes

Passando do caso univariado para o caso multivariado observa-se o que já se esperava. Os *scores* da rede neuronal generalizam substancialmente melhor que o Classificador Probabilista AP sendo que a passagem do caso univariado para o caso multivariado implica uma melhoria de 6.54 p.p nas redes neuronais (de 0.6612 para 0.7266) por contraponto a uma diminuição de 1.2 p.p. no caso do Classificador Probabilista (de 0.6366 para 0.624).

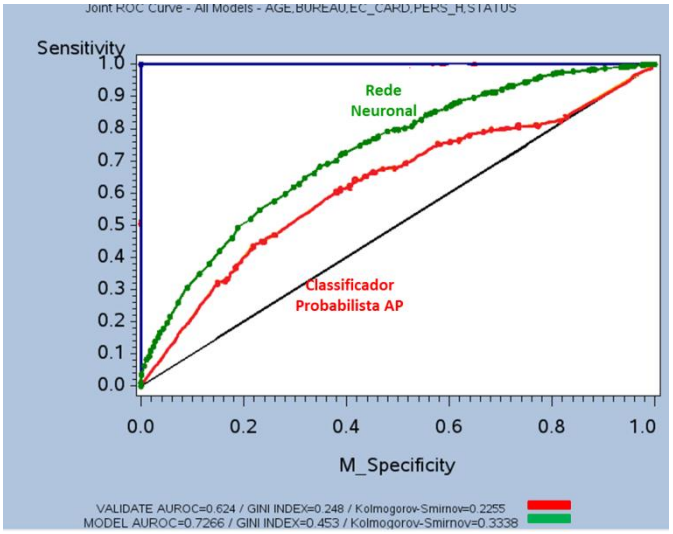


Figura 4.18 - Capacidade de generalização da rede neuronal versus Classificador Probabilista AP no caso multivariado

4.3 Classificador Probabilista AP ajustado directamente na amostra de teste, proporções p_x^{teste} e a majoração da generalização

Para se saber qual o limite superior ao poder de generalização de qualquer modelo em determinada amostra de teste, basta aplicar o conceito de Classificador Probabilista AP a essa mesma amostra de teste.

Caso a amostra de teste tenha sido devidamente "particionada", espera-se uma "capacidade discriminante dos dados" semelhante à da amostra de treino. No exemplo concreto, e olhando para o caso univariado (variável idade), observa-se uma AUC_{CP} de 0.6824 na amostra de validação que compara com a AUC_{CP} de 0.69 da amostra de treino

Em linha com as conclusões evidenciadas neste trabalho, a capacidade discriminante do Classificador Probabilista AP desenvolvido sobre amostra teste (0.6824) limita superiormente a capacidade de generalização da rede neuronal (0.6612) e do Classificador Probabilista AP (0.6366), ambos desenvolvidos sobre a amostra de treino - Figura 4.19

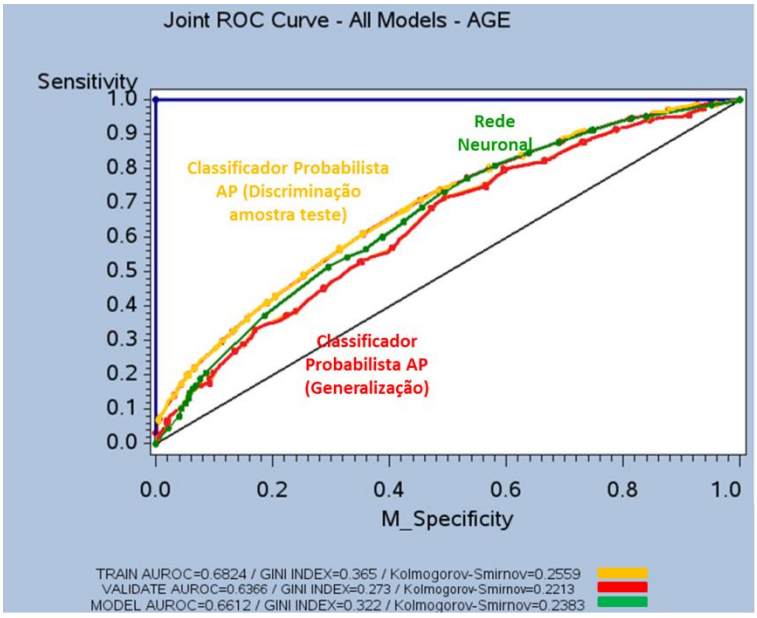


Figura 4.19 – Majoração da capacidade de generalização na amostra de teste

Passando para a abordagem multivariada (com 5 variáveis) verifica-se que a capacidade máxima discriminante entre as duas amostras é semelhante (0.8962 para a amostra de treino versus 0.9111 para a amostra de teste). Observa-se também a hierarquização típica com a capacidade discriminante (na amostra de teste) do Classificador Probabilista AP a ultrapassar a capacidade de generalização da rede neuronal que ultrapassa por sua vez a capacidade de generalização do Classificador Probabilista AP (figura 4.20).

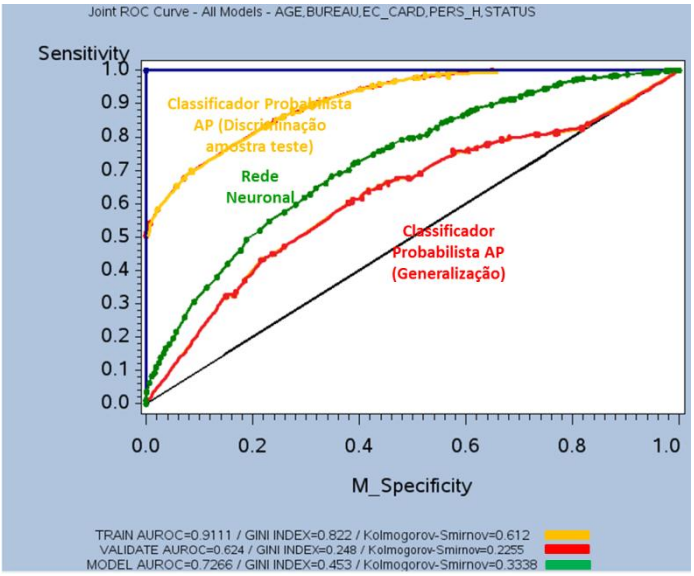


Figura 4.20 – Curva ROC comparando a capacidade discriminante do Classificador Probabilista AP na amostra de teste, com a capacidade de generalização da rede neuronal e do Classificador Probabilista AP

4.4. Aumento da granularidade e a monotonia não-decrescente da capacidade discriminante do Classificador Probabilista AP

4.4.1. Monotonia num contexto univariado

Para se observar o impacto que a granularidade do vector $\boldsymbol{X} = \boldsymbol{x}$ tem na capacidade discriminante do Classificador Probabilista AP optou-se por utilizar uma variável intervalar (Idade) e categorizá-la de diferentes formas (2,7,12,26 e 53 classes particionadas com base em quantis), calculando para cada instância os *scores* p_x (a vermelho na figura 4.21) e sua capacidade discriminante via AUC (Figura 4.21). Essa sequência pode ser observada nos seguintes gráficos:

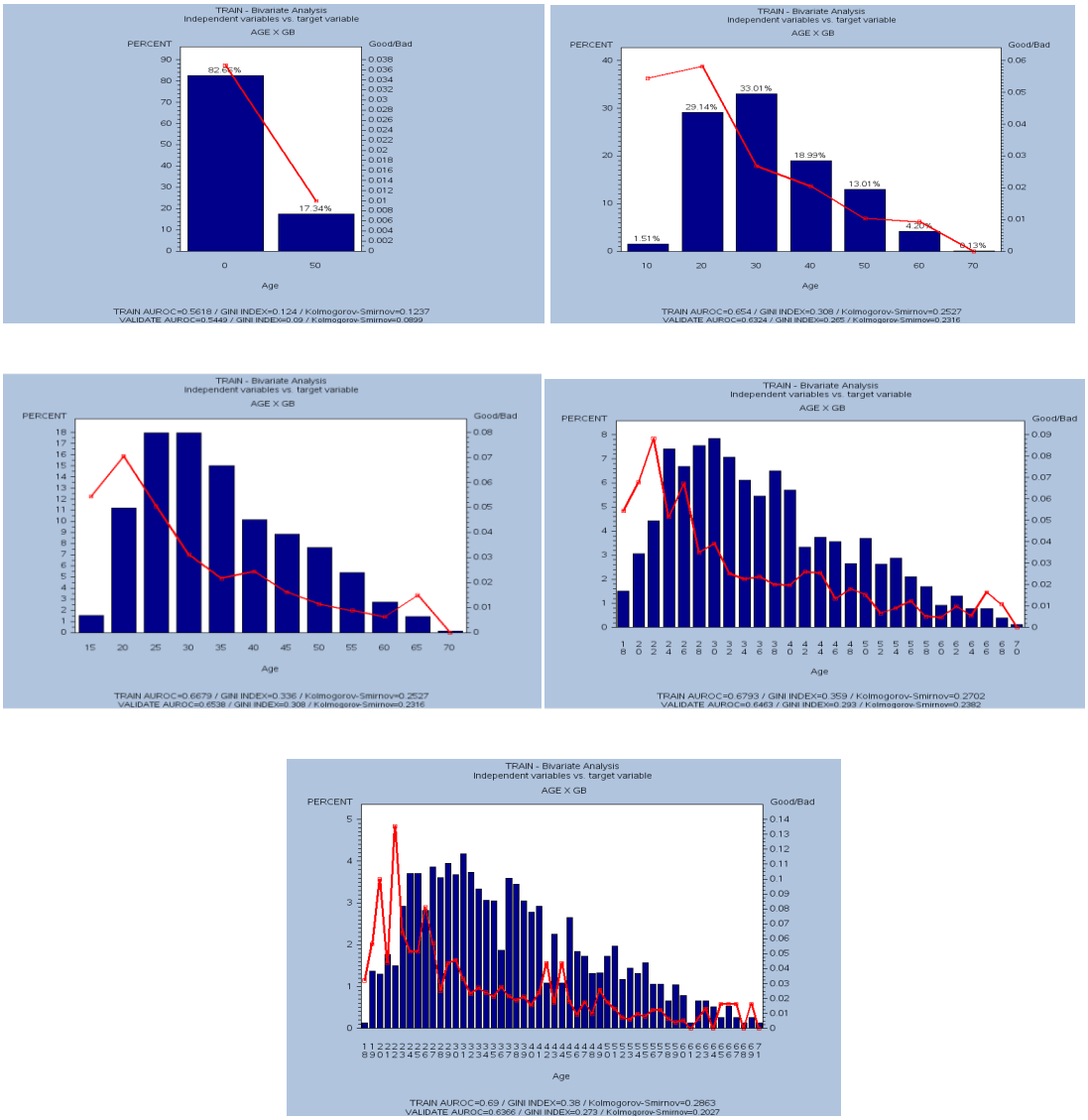


Figura 4.21 - Impacto da granularidade dos dados na "curva" da proporções empíricas.

	C. Disc.
1 classe	0.5
2 Classes	0.5618
7 Classes	0.654
12 Classes	0.6679
26 Classes	0.6793
Sem categorização	0.69

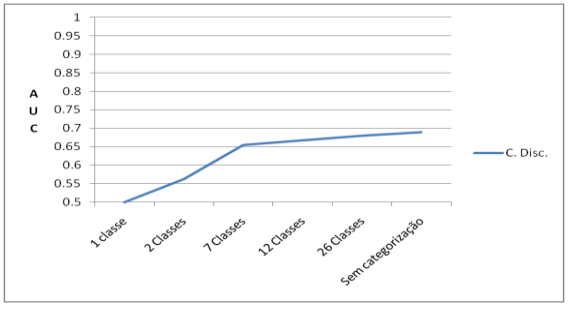


Figura 4.22 – Impacto da granularidade dos dados na AUC do Classificador Probabilista AP

Observa-se, não só, um aumento da capacidade discriminante do Classificador Probabilista AP à medida que se consideram mais classes nos dados (figura 4.22) mas também eventuais indícios de menor capacidade de generalização devido à volatilidade crescente das curvas dos *scores* (a vermelho na figura 4.21).

4.4.2. Monotonia num contexto multivariado

Passando de um contexto univariado para um contexto multivariado, a monotonia não-decrescente da capacidade discriminante do Classificador Probabilista AP mantêm-se à medida que o vector X das variáveis independentes ganha novas dimensões.

	C. Disc.
1 classe	0.5
2 Classes	0.5618
7 Classes	0.654
12 Classes	0.6679
26 Classes	0.6793
Sem categorização	0.69
2 Dimensões	0.7279
3 Dimensões	0.7662
4 Dimensões	0.8353
5 Dimensões	0.8962

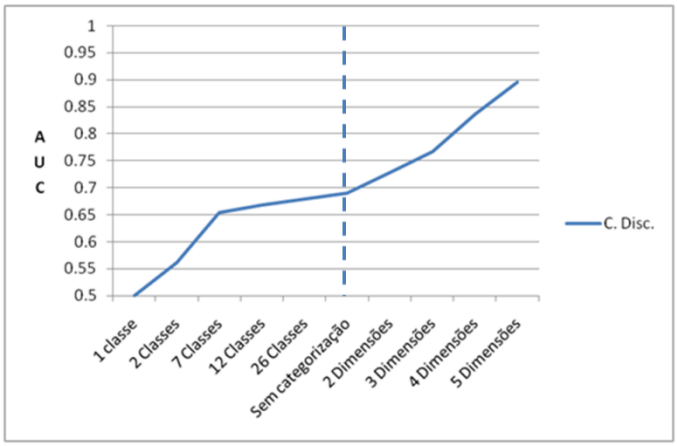


Figura 4.23 – Impacto na AUC na passagem de um Classificador Probabilista AP baseado em uma variável independente para um contexto multivariado

4.5. Aumento da granularidade e a tendência decrescente da capacidade de generalização do Classificador Probabilista AP

Se é certo que a capacidade discriminante do Classificador Probabilista AP é monótona não-decrescente, como se comportará a sua capacidade de generalização ?

Como seria de esperar, a capacidade de generalização do Classificador Probabilista AP acompanha a sua capacidade discriminante numa fase inicial (até às 12 classes) mas, à medida que a maldição da dimensionalidade ganha terreno, rapidamente se afastam. Essa dinâmica encontra-se resumida na figura 4.23.

	C. Disc.	C. Gener.
1 classe	0.5	0.5
2 Classes	0.5618	0.5449
7 Classes	0.654	0.6324
12 Classes	0.6679	0.6538
26 Classes	0.6793	0.6463
Sem categorização	0.69	0.6366
2 Dimensões	0.7279	0.6475
3 Dimensões	0.7662	0.6283
4 Dimensões	0.8353	0.61
5 Dimensões	0.8962	0.6

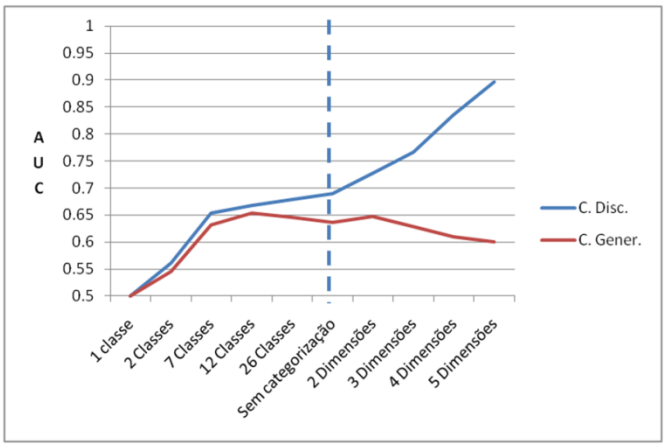
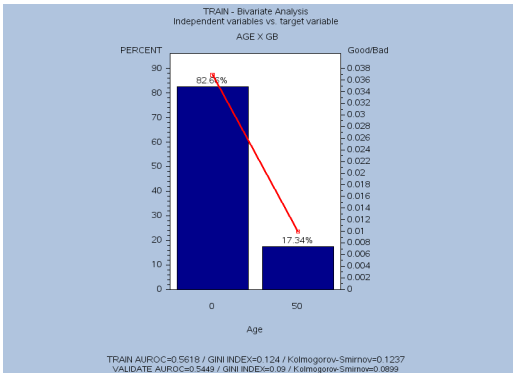
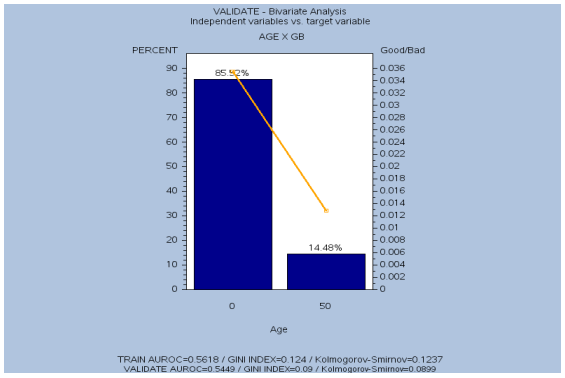


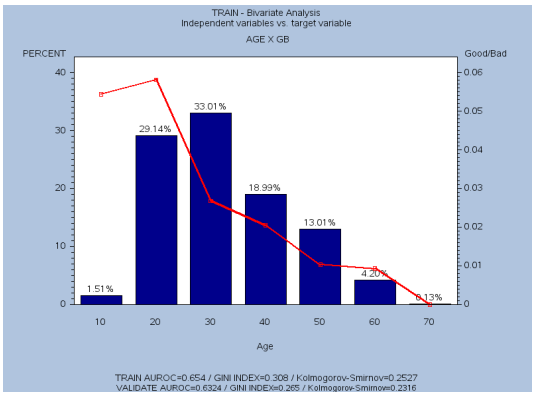
Figura 4.24 – Capacidade discriminante *versus* capacidade de generalização do Classificador Probabilista AP face à alteração de granularidade dos dados

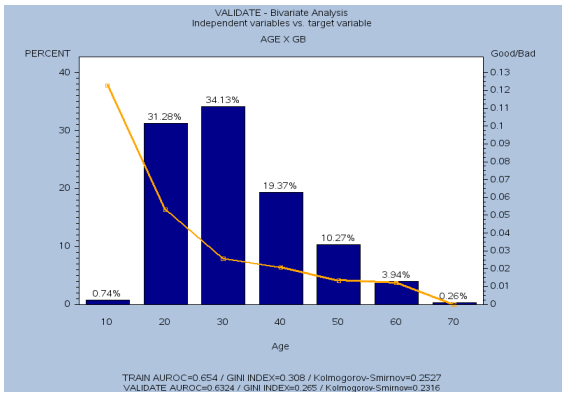
Porque sucede esta convergência inicial seguida de uma divergência abrupta? Precisamente porque os *scores* das duas amostras (p_x e p_x^{teste}) começam a não estar alinhados, à medida que a granularidade aumenta, uma vez que esses *scores* são estimados com cada vez menos robustez (porque baseados em cada vez menos observações para cada ponto **X=x**).

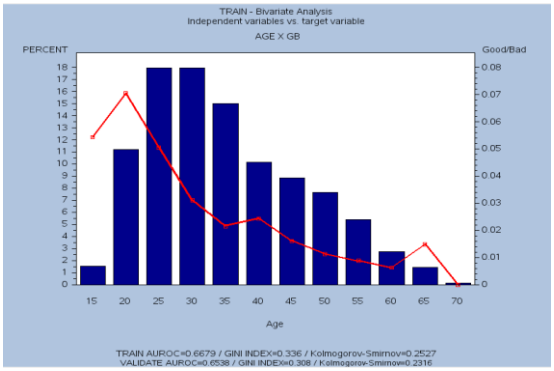
Para facilitar a visualização/comparação do aumento do desencontro dos p_x e p_x^{teste}, colocam-se lado a lado os diagramas de frequências de ambas as amostras onde se sobrepõem as curvas que reflectem p_x (a vermelho) e os p_x^{teste} (a laranja).

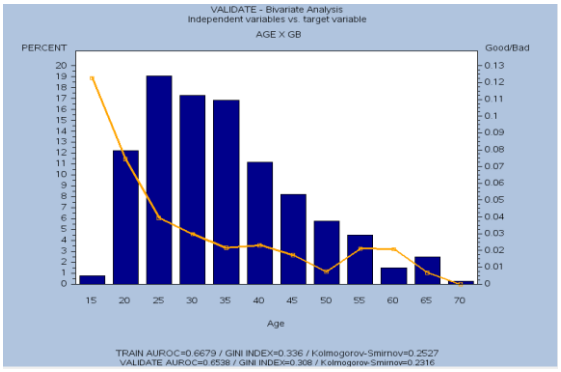

TRAIN - Bivariate Analysis
Independent variables vs. target variable
AGE X GB
PERCENT
Good/Bad
82.66%
17.34%
Age
TRAIN AUROC=0.5618 / GINI INDEX=0.124 / Kolmogorov-Smirnov=0.1237
VALIDATE AUROC=0.5449 / GINI INDEX=0.09 / Kolmogorov-Smirnov=0.0899

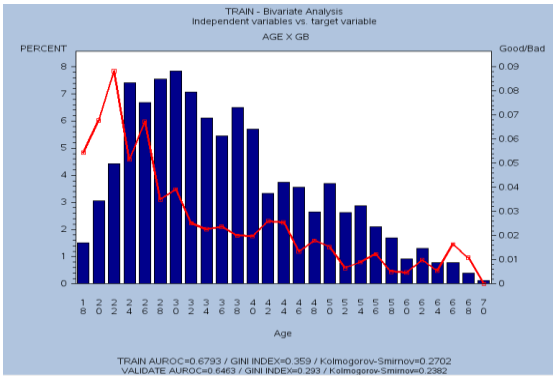

VALIDATE - Bivariate Analysis
Independent variables vs. target variable
AGE X GB
PERCENT
Good/Bad
85.52%
14.48%
Age
TRAIN AUROC=0.5618 / GINI INDEX=0.124 / Kolmogorov-Smirnov=0.1237
VALIDATE AUROC=0.5449 / GINI INDEX=0.09 / Kolmogorov-Smirnov=0.0899

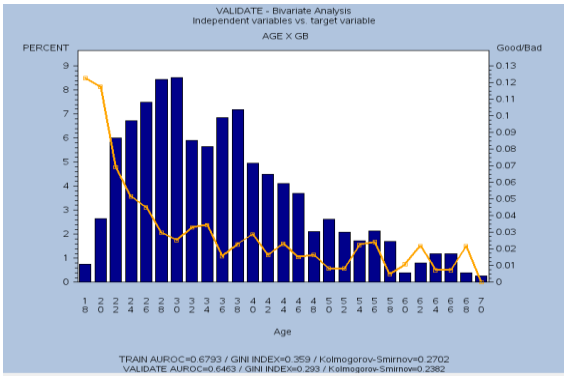

TRAIN - Bivariate Analysis
Independent variables vs. target variable
AGE X GB
PERCENT
Good/Bad
1.51%
29.14%
33.01%
18.99%
13.01%
4.20%
0.13%
Age
TRAIN AUROC=0.654 / GINI INDEX=0.308 / Kolmogorov-Smirnov=0.2527
VALIDATE AUROC=0.6324 / GINI INDEX=0.265 / Kolmogorov-Smirnov=0.2316


VALIDATE - Bivariate Analysis
Independent variables vs. target variable
AGE X GB
PERCENT
Good/Bad
0.74%
31.28%
34.13%
19.37%
10.27%
3.94%
0.26%
Age
TRAIN AUROC=0.654 / GINI INDEX=0.308 / Kolmogorov-Smirnov=0.2527
VALIDATE AUROC=0.6324 / GINI INDEX=0.265 / Kolmogorov-Smirnov=0.2316


TRAIN - Bivariate Analysis
Independent variables vs. target variable
AGE X GB
PERCENT
Good/Bad
Age
TRAIN AUROC=0.6679 / GINI INDEX=0.336 / Kolmogorov-Smirnov=0.2527
VALIDATE AUROC=0.6538 / GINI INDEX=0.308 / Kolmogorov-Smirnov=0.2316


VALIDATE - Bivariate Analysis
Independent variables vs. target variable
AGE X GB
PERCENT
Good/Bad
Age
TRAIN AUROC=0.6679 / GINI INDEX=0.336 / Kolmogorov-Smirnov=0.2527
VALIDATE AUROC=0.6538 / GINI INDEX=0.308 / Kolmogorov-Smirnov=0.2316


TRAIN - Bivariate Analysis
Independent variables vs. target variable
AGE X GB
PERCENT
Good/Bad
Age
TRAIN AUROC=0.6793 / GINI INDEX=0.359 / Kolmogorov-Smirnov=0.2702
VALIDATE AUROC=0.6463 / GINI INDEX=0.293 / Kolmogorov-Smirnov=0.2382


VALIDATE - Bivariate Analysis
Independent variables vs. target variable
AGE X GB
PERCENT
Good/Bad
Age
TRAIN AUROC=0.6793 / GINI INDEX=0.359 / Kolmogorov-Smirnov=0.2702
VALIDATE AUROC=0.6463 / GINI INDEX=0.293 / Kolmogorov-Smirnov=0.2382

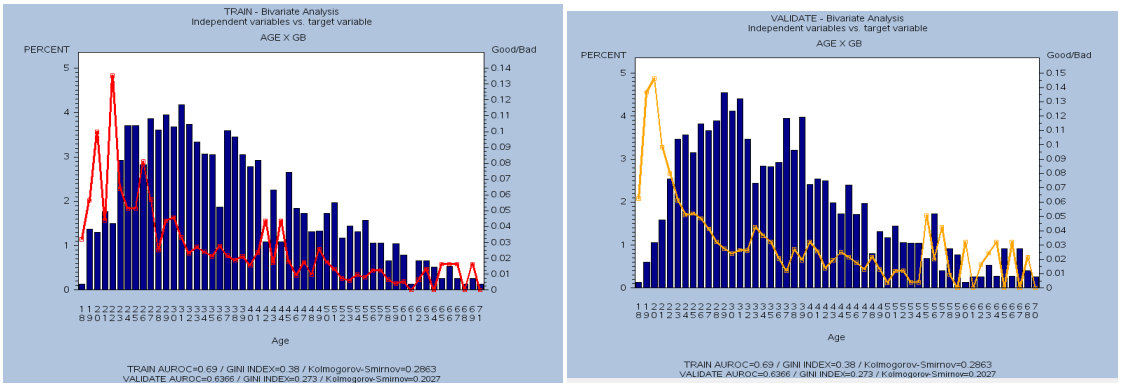


Figura 4.25 – Evolução (e desencontro) das proporções empíricas na amostra treino (p_x) versus a amostra de teste (p_x^{teste}) face à alteração da granularidade

Uma das observações mais interessantes a retirar deste tópico é o facto da categorização das variáveis tornar mais estável e semelhante o binómio capacidade discriminante / capacidade de generalização da informação.

4.6. Classificador Probabilista AP vs. Rede Neuronal: Granularidade, capacidade discriminante e capacidade de generalização.

Para finalizar o capítulo, importa estender a análise efectuada relativamente ao Classificador Probabilista AP para os modelos preditivos propriamente ditos. De uma forma sintética, e trabalhando com os resultados das redes neuronais, apresentam-se os resultados na figura 4.26.

	C. Disc.	C. Gener.	RN C. Disc.	RN C. Gener.
1 classe	0.5	0.5	0.5	0.5
2 Classes	0.5618	0.5449	0.5618	0.5449
7 Classes	0.654	0.6324	0.654	0.6324
12 Classes	0.6679	0.6538	0.6663	0.6526
26 Classes	0.6793	0.6463	0.6722	0.6562
Sem categorização	0.69	0.6366	0.6734	0.6612
2 Dimensões	0.7279	0.6475	0.6914	0.6837
3 Dimensões	0.7662	0.6283	0.6999	0.6862
4 Dimensões	0.8353	0.61	0.7164	0.6969
5 Dimensões	0.8962	0.6	0.7212	0.7019

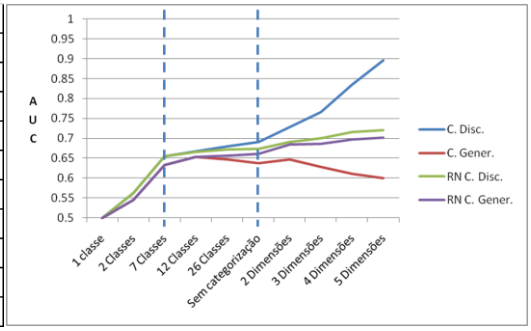


Figura 4.26 - Capacidade discriminante *versus* capacidade de generalização do Classificador Probabilista AP e da Rede Neuronal face à alteração de granularidade dos dados

Da análise da figura 4.26 destaca-se o caminho inevitavelmente divergente entre a capacidade discriminante e a capacidade de generalização do Classificador Probabilista AP (*overfitting*) por contraponto a um alinhamento continuado entre a capacidade discriminante e a capacidade de generalização do modelo preditivo (captura efectiva do padrão existente em ambas as amostras).

Capítulo 5. Conclusões

Várias conclusões interessantes se retiram do exposto.

Demonstrou-se analiticamente que existe um limite superior, empírico, à capacidade discriminante de qualquer modelo e que esse limite reside na qualidade da informação que está a ser utilizada para desenvolver o modelo. Ao modelo associado a essa capacidade discriminante dos dados chamou-se Classificador Probabilista AP, sendo este uma derivação possível do Classificador MAP de *Bayes*.

Considerando um conjunto de dados específico (com variável dependente binária), se o Classificador MAP de *Bayes* é o modelo que minimiza o erro de classificação, o Classificador Probabilista AP é o modelo que melhor ordena a variável dependente.

Propôs-se o índice CEDP (*Captured Empirical Discriminatory Power*) como estatística paralela ao índice de *Gini*. Enquanto o índice de *Gini* indica a percentagem da capacidade discriminante do modelo perfeito que o actual modelo consegue capturar, o CEDP indica a percentagem da capacidade discriminante dos dados capturada pelo modelo.

Também se demonstrou analiticamente que a capacidade discriminante dos dados é monótona não decrescente em relação ao aumento da granularidade da informação utilizada no desenvolvimento de modelos preditivos.

Por último mostrou-se empiricamente a relação entre capacidade máxima de discriminação de um modelo numa amostra e as implicações em termos de *overfitting* e má capacidade de generalização para outras amostras. Esta dinâmica não está devidamente acautelada quando se utiliza o conceito de modelo perfeito.

Capítulo 6. Trabalho futuro

O presente trabalho respondeu às questões centrais da tese, mas levantou outras hipóteses que seria interessante abordar, nomeadamente:

a) Será a categorização das variáveis uma estratégia adequada na modelação permitindo um equilíbrio entre a capacidade discriminante e a capacidade de generalização?

b) A distância que separa a capacidade discriminante da informação (a azul na figura 6.1) da capacidade discriminante do modelo (a verde) representa apenas ruído/*overfitting* ou engloba também uma zona de incapacidade dos modelos para extrair padrões? Será possível estimar esta fronteira?

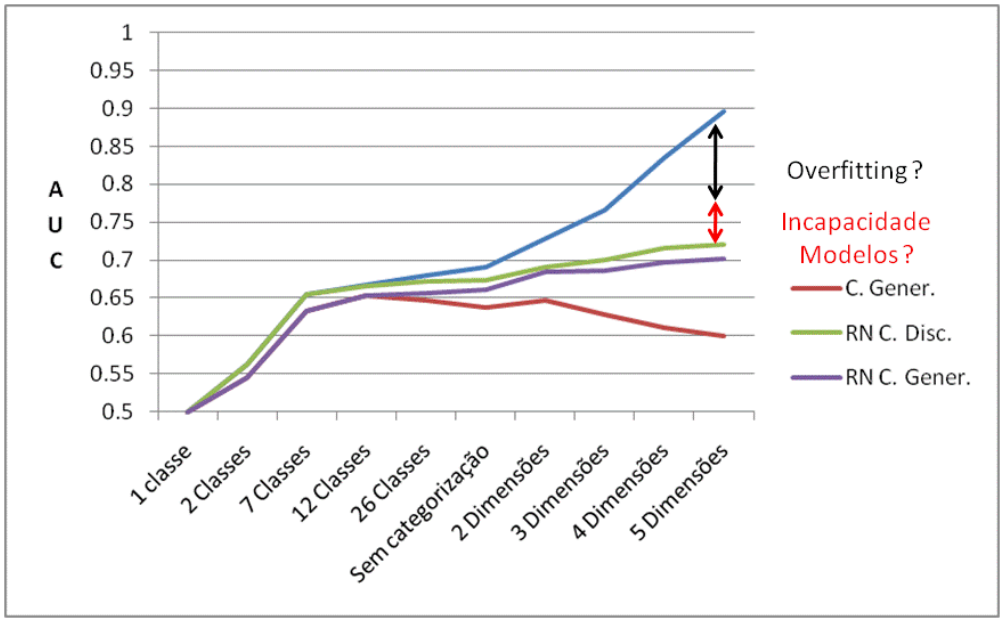


Figura 6.1 – Delta entre a capacidade discriminante dos dados e a capacidade discriminante dos modelos

Bibliografia

Alpaydin, E. (2004). *Introduction to Machine Learning.* Cambrige, Massachusets and London,England: The MIT Press.

Bação, F. (2009). *Data Mining - Apontamentos Licenciatura.* ISEGI, Lisbon.

Basel Committee on Banking Supervision. (2005). *Working Paper No. 14: Studies on the validation of Internal Rating Systems.* Bank for International Settlements.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition.* Oxford University Press.

Engelmann, B., & Rauhmeier, R. (2006). *The Basel II Risk Parameters, Estimation, Validation, Stress Testing.* Heildlberg: Springer Berlin.

Graziano, A., & Raulin, M. (1997). *Reserach Methods, A process of inquiry, 3rd edition.* New York: Addison Wesley Longman, Inc.

Hand, D., Mannila, H., & Smyth, P. (2001). *Principles of Data Mining.* Cambridge, Massachusetts and London,England: The MIT Press.

Hanley, J., & McNeil, B. (1982). *The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve, RADIOLOGY Vol. 143 No. 1.* Radiological Society of North America.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The Elements of Statistical Learning Data mining, Inference and prediction.* New York: Springer.

Hosmer, D., & Lemeshow, S. (1989). *Applied Logistic Regression.* Massachusetts: John Wiley & Sons.

Laursen, G., & Thorlund, J. (2010). *Business Analytics for Managers: Taking Business Intelligence Beyond Reporting.* New Jersey: John Wiley & Sons, Inc.

Lobo, V. (2008). *Sistemas de apoio à Decisão, Técnicas e Algoritmos.* Lisbon.

Mendenhall, W., & Sincich, T. (1996). *A second course in Statistics: Regression Analysis.* New Jersey: Prentice-Hall Inc.

Nisbet, R., Elder, J., & Miner, G. (2009). *Handbook of Statistical Analysis & Datamining Applications.* Elsevier Inc.

Patterson, D. (1996). *Artificial Neural Networks, Theory and Applications.* Singapore: Prentice Hall.

Thomas, L. (2007). *Measuring the Discrimination Quality of Suites of Scorecards: ROCs, Ginis, Bounds and Segmentation.* Edinburgh.

Witten, I., & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools & Technics.* San Francisco: Elsevier, Inc.

Apêndices

Apêndice 1: Curva ROC e a tomada de decisão

Na figura A.1, observa-se que caso se opte pelo ponto de corte *score*=0.7, identificam-se cerca de 70% dos eventos mas cometendo o erro de marcar falsamente 20% de não-eventos em eventos. Por outro lado, a escolha deste ponto de corte implica que 30% dos eventos ficam por identificar.

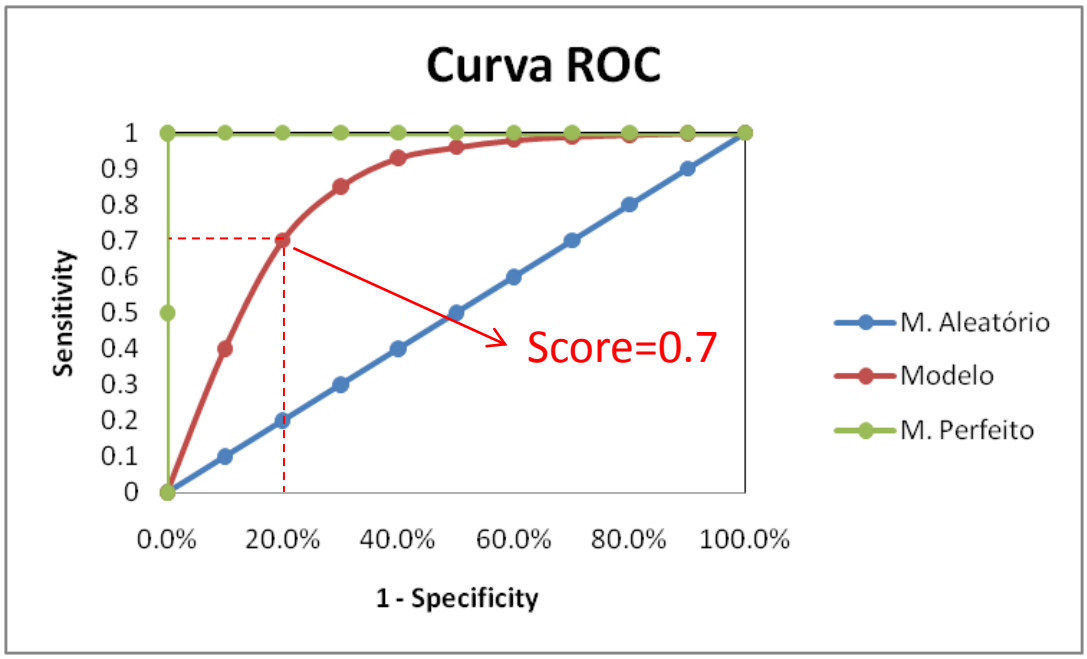


Figura A.1 – Curva ROC e pontos de corte

Focando num ponto de corte específico (*score*=0.7 neste exemplo) é possível estimar as implicações das nossas decisões, como se visualiza na figura A.2

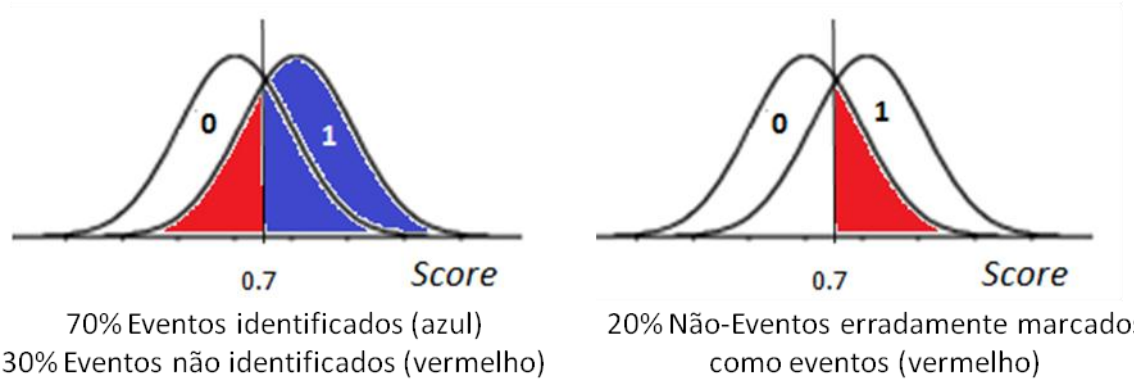


Figura A.2 – Pontos de corte e erro associado

A matriz de confusão (figura A.3) é o método mais utilizado para uma análise aprofundada das implicações da escolha de determinado ponto de corte.

		Valor actual	
	Score=0.7	0	1
Previsão	0	720	30
	1	180	70
	Total	900	100

Figura A.3 – Matriz de confusão

Os constituintes base da matriz são: Positivos (P) = 100 (Eventos) , Negativos (N) = 900 (não-eventos), *True Positives* (TP) = 70, *True Negatives* (TN) = 720, *False Positives* (FP) = 180 e *False Negatives* (FN) = 30.

Uma série de estatísticas podem ser retiradas de uma matriz de confusão. Enumeram-se algumas das mais comuns: *Sensitivity* ou *True Positive (hit) Rate* (TPR) = TP / P = 70%, *Specificity* ou *True Negative (tit) Rate* (TNR) = TN / N =80%, *1-Specificity* ou *False Positive (hit) Rate* (FPR) = FP / N = 20%, *Accuracy* = (TP+TN)/(P+N) = (720+70) / (900+100) = 79% entre outras.

Apêndice 2: Cálculo prático da estatística U Mann-Whitney

A título de exemplo, suponha-se uma amostra com 245 (n) observações onde se observam 130 eventos (n_e) e 115 não-eventos (n_{ne}) e para o qual se desenvolveu um modelo que devolve 3 *scores*.

Score	# Eventos	# Não-Eventos
0.99	90	5
0.7	30	22
0.6	10	88
Total	130	115

Figura A.4 – Matriz para cálculo da estatística U

Construindo uma tabela com contagens agregadas da amostra e posteriormente ordenada decrescentemente por *score* (figura A.4) calcula-se a estatística U de Mann-Whitney da seguinte forma:

Da tabela em A.4 tem-se $n = 245,\ n_e = 130\ , n_{\bar{e}} = 115,$

$n_e * n_{\bar{e}} = 130 * 115 = 14950$ (número total de pares possíveis)

$n_{conc} = 90 * (22 + 88) + \ (30 * 88) = 12540$ (número de pares concordantes em que $\text{Score}_{\text{e}} > \text{Score}_{\bar{\text{e}}}$)

$n_{disc} = 5 * (30 + 10) + \ (22 * 10) = 420$ (número de pares discordantes em que $\text{Score}_{\text{e}} < \text{Score}_{\bar{\text{e}}}$)

$n_{tie} = (90 * 5) + \ (30 * 22) + \ (10 * 88) = 1990$ (número de pares empatados (*tied*) em que $\text{Score}_{\text{e}} = \text{Score}_{\bar{\text{e}}}$)

Relembra-se que o total de pares possíveis se divide em pares concordantes, pares discordantes e pares empatados donde: $n_{conc} + n_{disc} + n_{tie} = n_e * n_{\bar{e}}$

Tem-se então $U = \frac{(12540+0.5*1990)}{14950} = 90.5\%$ = AUC

Apêndice 3: Desdobramento das parcelas

Nesta secção detalha-se um pouco mais a expansão de n_{conc} para ambos os modelos s_x e π_x.

Parcelas na expansão de n_{conc} para p_x	**Parcelas na expansão de n_{conc} para π_x**
$E_{(1)} * (\bar{E}_{(2)} + \cdots + \bar{E}_{(k)} + \cdots + \bar{E}_{(s)} + \cdots + \bar{E}_{(c)})$	$E_{(1)} * (\bar{E}_{(2)} + \cdots + \bar{E}_{(s)} + \cdots + \bar{E}_{(k)} + \cdots + \bar{E}_{(c)})$
$E_{(2)} * (\bar{E}_{(3)} + \cdots + \bar{E}_{(k)} + \cdots + \bar{E}_{(s)} + \cdots + \bar{E}_{(c)})$	$E_{(2)} * (\bar{E}_{(3)} + \cdots + \bar{E}_{(s)} + \cdots + \bar{E}_{(k)} + \cdots + \bar{E}_{(c)})$
…	…
$E_{(k)} * (\bar{E}_{(k+1)} + \cdots + \bar{E}_{(s)} + \cdots + \bar{E}_{(c)})$	$E_{(s)} * (\bar{E}_{(k+1)} + \cdots + \bar{E}_{(k)} + \cdots + \bar{E}_{(c)})$
…	…
$E_{(s-1)} * (\bar{E}_{(s)} + \ \ldots + \bar{E}_{(c)})$	$E_{(s-1)} * (\bar{E}_{(k)} + \cdots + \bar{E}_{(c)})$
$E_{(s)} * (\bar{E}_{(s+1)} + \ \ldots + \bar{E}_{(c)})$	$E_{(k)} * (\bar{E}_{(s+1)} + \ \ldots + \bar{E}_{(c)})$
…	…
$E_{(c-1)} * \bar{E}_{(c)}$	$E_{(c-1)} * \bar{E}_{(c)}$

Figura A.5 – Expansão de n_{conc} para p_x e n_{conc} para π_x

Apêndice 4: Implementação do Nó Enteprise Miner

```
/* ------------------------------------------------------------ */
/*                                COMMOM MACROS                 */
/* ------------------------------------------------------------ */


Filename Macros catalog 'Catalogs.Em_Ext61_am.Macros_Gerais.Source';
%Include Macros;


Filename Macros catalog
'Catalogs.Em_Ext61_am.Macros_Estatistica.Source';
%Include Macros;


Filename Macros catalog 'Catalogs.Em_Ext61_am.A000_Include.Source';
%Include Macros;



/* ------------------------------------------------------------ */
/*                              INITIALIZATION PHASE            */
/*------------------------------------------------------------- */



%Macro Init_Explore_BIVARIATE;
 %Let Macro_Name=Init_Explore_BIVARIATE;

 %Put WARNING- [&Sysuserid.:&Macro_Name.] Entering MACRO &Macro_Name.;


 Data Work.Log; Format Date Datetime19. Log $128.;
       Date=Datetime(); Log="Version 2.0 AM"; Output;
       Date=Datetime(); Log="Entering MACRO &Macro_Name."; Output;
 Run;

  /* Initialization for TOOL Properties because users can use EM5BATCH
macro which bypasses UI validations */
  %Log(Text=---Initializing tool property macro
variables.,Macro_Name=&Macro_Name.);



  %Em_Checkmacro(name=EM_PROPERTY_VARROLE,global=Y,Value=INPUT);
   %Em_Checkmacro(name=EM_PROPERTY_VARLEVEL,global=Y,Value=INTERVAL);
    %Em_Checkmacro(name=EM_PROPERTY_HIGHSCORE,global=Y,Value=0);
     %Em_Checkmacro(name=EM_PROPERTY_EVENT,global=Y,Value=1);
          %Em_Checkmacro(name=EM_PROPERTY_VARROC,global=Y,Value=N);

%Em_Checkmacro(name=EM_PROPERTY_VARLEVEL,global=Y,Value=INTERVAL);
      %Em_Checkmacro(name=EM_PROPERTY_MAXFREQ,global=Y,Value=30000);
      %Em_Checkmacro(name=EM_PROPERTY_JOINTROC,global=Y,Value=N);
      %Em_Checkmacro(name=EM_PROPERTY_MODELROC,global=Y,Value=Y);
```

```
      %Em_Checkmacro(name=EM_PROPERTY_MODEL,global=Y,Value=N);

%Em_Checkmacro(name=EM_PROPERTY_GENERALIZATION,global=Y,Value=N);
      %Em_Checkmacro(name=EM_PROPERTY_QUANTILE,global=Y,Value=N);
      %Em_Checkmacro(name=EM_PROPERTY_NUMQUANT,global=Y,Value=N);
      %Em_Checkmacro(name=EM_PROPERTY_SCOREVAR,global=Y,Value=P_GB1);

  %Em_Checkmacro(name=EM_PROPERTY_SAMPLE,global=Y,Value=Y);
   %Em_Checkmacro(name=EM_PROPERTY_SAMPLEPER,global=Y,Value=10);
    %Em_Checkmacro(name=EM_PROPERTY_SEED,global=Y,Value=12345);
     %Em_Checkmacro(name=EM_PROPERTY_WOE,global=Y,Value=N);
      %Em_Checkmacro(name=EM_PROPERTY_HTEXT,global=Y,Value=5);
       %Em_Checkmacro(name=EM_PROPERTY_NUMBINS,global=Y,Value=10);
        %Em_Checkmacro(name=EM_PROPERTY_SCALEWOE,global=Y,Value=2);

%Em_Checkmacro(name=EM_PROPERTY_SCALEHIST,global=Y,Value=100);


%If "&EM_PROPERTY_VARLEVEL." Eq "CLASS" %Then %Let
EM_PROPERTY_QUANTILE=N;


  %Log(Text=---
EM_PROPERTY_VARROLE=&EM_PROPERTY_VARROLE.,Macro_Name=&Macro_Name.);
   %Log(Text=---
EM_PROPERTY_VARLEVEL=&EM_PROPERTY_VARLEVEL.,Macro_Name=&Macro_Name.);
    %Log(Text=---
EM_PROPERTY_HIGHSCORE=&EM_PROPERTY_HIGHSCORE.,Macro_Name=&Macro_Name.)
;
     %Log(Text=---
EM_PROPERTY_EVENT=&EM_PROPERTY_EVENT.,Macro_Name=&Macro_Name.);
      %Log(Text=---
EM_PROPERTY_VARROC=&EM_PROPERTY_VARROC.,Macro_Name=&Macro_Name.);
       %Log(Text=---
EM_PROPERTY_JOINTROC=&EM_PROPERTY_JOINTROC.,Macro_Name=&Macro_Name.);
       %Log(Text=---
EM_PROPERTY_MODELROC=&EM_PROPERTY_MODELROC.,Macro_Name=&Macro_Name.);
       %Log(Text=---
EM_PROPERTY_MODEL=&EM_PROPERTY_MODEL.,Macro_Name=&Macro_Name.);
       %Log(Text=---
EM_PROPERTY_GENERALIZATION=&EM_PROPERTY_GENERALIZATION.,Macro_Name=&Ma
cro_Name.);

       %Log(Text=---
EM_PROPERTY_QUANTILE=&EM_PROPERTY_QUANTILE.,Macro_Name=&Macro_Name.);
       %Log(Text=---
EM_PROPERTY_NUMQUANT=&EM_PROPERTY_NUMQUANT.,Macro_Name=&Macro_Name.);


       %Log(Text=---
EM_PROPERTY_SCOREVAR=&EM_PROPERTY_SCOREVAR.,Macro_Name=&Macro_Name.);


  %Log(Text=---
EM_PROPERTY_SAMPLE=&EM_PROPERTY_SAMPLE.,Macro_Name=&Macro_Name.);
```

```
   %Log(Text=---
EM_PROPERTY_SAMPLESIZE=&EM_PROPERTY_SAMPLEPER.,Macro_Name=&Macro_Name.
);
    %Log(Text=---
EM_PROPERTY_SEED=&EM_PROPERTY_SEED.,Macro_Name=&Macro_Name.);

    %Log(Text=---
EM_PROPERTY_BIVARIATE=&EM_PROPERTY_WOE.,Macro_Name=&Macro_Name.);
    %Log(Text=---
EM_PROPERTY_BIVARIATE=&EM_PROPERTY_NUMBINS.,Macro_Name=&Macro_Name.);

    %Log(Text=---
EM_PROPERTY_BIVARIATE=&EM_PROPERTY_SCALEHIST.,Macro_Name=&Macro_Name.)
;
        %Log(Text=---
EM_PROPERTY_BIVARIATE=&EM_PROPERTY_SCALEWOE.,Macro_Name=&Macro_Name.);
         %Log(Text=---
EM_PROPERTY_SASTable=&EM_PROPERTY_SASTable.,Macro_Name=&Macro_Name.);


  %If &EM_PROPERTY_SAMPLEPER. GT 100 %Then %Let
EM_PROPERTY_SAMPLEPER=100;
  %If &EM_PROPERTY_SAMPLEPER. LT   1 %Then %Let
EM_PROPERTY_SAMPLEPER=1;


  /* Check if mandatory files are present */

  %Log(Text=---Check Input File for this
node.,Macro_Name=&Macro_Name.);
   %Log(Text=---
EM_IMPORT_DATA:&Em_Import_DATA.,Macro_Name=&Macro_Name.);


  %If "&Em_Import_Data." eq ""
   %Then %Do;
      %Error_Routine(Text=PROBLEM: No TRAIN input for this
NODE.,Macro_Name=&Macro_Name.);
   %End;
   %Else %Do;
      %Log(Text=***OK. TRAIN input for this
NODE,Macro_Name=&Macro_Name.);
   %End;


   %If "&EM_Num_Binary_Target." ne "1"
      %Then %Do;
      %Error_Routine(Text=PROBLEM: Number of BINARY TARGETS not equal
to 1,Macro_Name=&Macro_Name.);
   %End;
   %Else %Do;
      %Log(Text=***OK. Number of BINARY TARGETS equals
1,Macro_Name=&Macro_Name.);
   %End;
```

```
Data &EM_LIB..&EM_NODEID._Rank_ROC;

  Format Variable $32.;

  Variable='ALL';
  ROC_Train=1;
  Gini_Train=1;
  KS_Train=1;
  Granular_Train=1000;
  RObs_Train=5;
  ROC_Val=1;
  Gini_Val=1;
  KS_Val=1;
  Granular_Val=1000;
  RObs_Val=5;
  Output;

Run;

%Mend Init_Explore_BIVARIATE;
```

```
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
/*                                        TRAINING PHASE                */
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */

%Macro Train_Explore_BIVARIATE;
 %Let Macro_Name=Train_Explore_BIVARIATE;

 %Log(Text=Entering MACRO &Macro_Name.,Macro_Name=&Macro_Name.);


%Log(Text=Retrieving transformations code from
NOTES,Macro_Name=&Macro_Name.);

%Let TransfCode=;

 Data In_Ficheiro_transf;
       %let _EFIERR_ = 0; /* set the ERROR detection macro variable */
        infile "&EM_FILE_EMNOTES." Delimiter = ';' MISSOVER DSD
lrecl=32767 firstobs=1 obs=100;
          informat Code   $1024.;

          format Code  $1024.;

          input  Code   $;

          If code='' Then Delete; Else
```

```
          Code=Trim(Code)||";";

      if _ERROR_ then call symputx('_EFIERR_',1);  /* set ERROR
detection macro variable */

      run;



      Proc Sql Noprint;
        Select Code Into:TransfCode separated by ';' from
In_Ficheiro_transf;
      Quit;







%If "&EM_PROPERTY_SAMPLE." Eq "Y" %Then %Do;

  %Log(Text=---SAMPLING - TRAIN DATASET,Macro_Name=&Macro_Name.);



  Data &EM_NODEID._BIVARIATEFILE(Drop=Random);
   Set &EM_IMPORT_DATA;

     &TransfCode.;

     Random=Ranuni(&EM_PROPERTY_SEED.);

     If Random<=&EM_PROPERTY_SAMPLEPER./100 Then Output;

  Run;

    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: SAMPLING
TRAIN,Macro_Name=&Macro_Name.);
    %End;
    %Else %Do;
        %Log(Text=---OK,Macro_Name=&Macro_Name.);
    %End;




%If "&EM_IMPORT_VALIDATE." ne "" %Then %Do; /* VALIDATE  */


%Log(Text=---SAMPLING - VALIDATE DATASET,Macro_Name=&Macro_Name.);

 Data &EM_NODEID._BIVARIATEFILE_VAL(Drop=Random);
   Set &EM_IMPORT_VALIDATE.;
```

```
      &TransfCode.;

      Random=Ranuni(&EM_PROPERTY_SEED.);

      If Random<=&EM_PROPERTY_SAMPLEPER./100 Then Output;

   Run;


     %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
         %Error_Routine(Text=PROBLEM: SAMPLING
VALIDATE,Macro_Name=&Macro_Name.);
     %End;
     %Else %Do;
         %Log(Text=---OK,Macro_Name=&Macro_Name.);
     %End;


  %End; /* VALIDATE */




%If "&EM_PROPERTY_MODEL." Eq "Y" and "&EM_PROPERTY_SASTable." ne ""
%Then %Do; /* EXTERNAL MODEL DATASET USAGE */


  %Log(Text=---SAMPLING - SCORED DATASET,Macro_Name=&Macro_Name.);



   Data &EM_NODEID._BIVARIATEFILE_SCORED(Drop=Random);
    Set &EM_PROPERTY_SASTable.;

      &TransfCode.;

      Random=Ranuni(&EM_PROPERTY_SEED.);

      If Random<=&EM_PROPERTY_SAMPLEPER./100 Then Output;

   Run;

     %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
         %Error_Routine(Text=PROBLEM: SAMPLING SCORED
DATASET,Macro_Name=&Macro_Name.);
     %End;
     %Else %Do;
         %Log(Text=---OK,Macro_Name=&Macro_Name.);
     %End;


%End; /* EXTERNAL MODEL DATASET USAGE */
```

```
  Proc Sql Noprint;
   Select Count(*) into:TEMP_OBS From &EM_NODEID._BIVARIATEFILE;
  Quit;

  %Let TEMP_OBS=&TEMP_OBS.;

  %Log(Text=---Observations in TRAIN
&EM_NODEID._BIVARIATEFILE=&TEMP_OBS.,Macro_Name=&Macro_Name.);



%If "&EM_IMPORT_VALIDATE." ne "" %Then %Do;

  Proc Sql Noprint;
   Select Count(*) into:TEMP_OBS From &EM_NODEID._BIVARIATEFILE_VAL;
  Quit;

  %Let TEMP_OBS=&TEMP_OBS.;

  %Log(Text=---Observations in VALIDATE
&EM_NODEID._BIVARIATEFILE_VAL=&TEMP_OBS.,Macro_Name=&Macro_Name.);

 %End; /* VALIDATE */



%If "&EM_PROPERTY_MODEL." Eq "Y" and "&EM_PROPERTY_SASTable." ne ""
%Then %Do;

  Proc Sql Noprint;
   Select Count(*) into:TEMP_OBS From
&EM_NODEID._BIVARIATEFILE_SCORED;
  Quit;

  %Let TEMP_OBS=&TEMP_OBS.;

  %Log(Text=---Observations in SCORE
&EM_NODEID._BIVARIATEFILE_SCORED=&TEMP_OBS.,Macro_Name=&Macro_Name.);


%End; /* EXTERNAL MODEL DATASET USAGE */



%End;/* SAMPLING */

%Else %Do;   /* NO SAMPLING */
```

```
  %Log(Text=---NO SAMPLING - TRAIN DATASET,Macro_Name=&Macro_Name.);

  Proc Sql Noprint;
   Select Count(*) into:TEMP_OBS From &EM_IMPORT_DATA.;
  Quit;

  %Let TEMP_OBS=&TEMP_OBS.;



  Data &EM_NODEID._BIVARIATEFILE;
   Set &EM_IMPORT_DATA;

     &TransfCode.;

  Run;



    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: NO
SAMPLING,Macro_Name=&Macro_Name.);
    %End;
    %Else %Do;
        %Log(Text=---Observations in TRAIN
&EM_NODEID._BIVARIATEFILE=&TEMP_OBS.,Macro_Name=&Macro_Name.);
    %End;



%If "&EM_IMPORT_VALIDATE." ne "" %Then %Do; /* VALIDATE */



  %Log(Text=---NO SAMPLING - VALIDATE
DATASET,Macro_Name=&Macro_Name.);

  Data &EM_NODEID._BIVARIATEFILE_VAL;
   Set &EM_IMPORT_VALIDATE;

      &TransfCode.;

  Run;


    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: NO SAMPLING
VALIDATE,Macro_Name=&Macro_Name.);
    %End;
    %Else %Do;
        %Log(Text=---Observations in VALIDATION
&EM_NODEID._BIVARIATEFILE_VAL=&TEMP_OBS.,Macro_Name=&Macro_Name.);
    %End;
```

```
 %End; /* VALIDATE */



%If "&EM_PROPERTY_MODEL." Eq "Y" and "&EM_PROPERTY_SASTable." ne ""
%Then %Do; /* EXTERNAL MODEL DATASET USAGE */

  %Log(Text=---NO SAMPLING - SCORED DATASET,Macro_Name=&Macro_Name.);
  %Put EM_PROPERTY_SASTable=&EM_PROPERTY_SASTable.=;


  Data &EM_NODEID._BIVARIATEFILE_SCORED;
   Set &EM_PROPERTY_SASTable.;

      &TransfCode.;

  Run;


    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: CREATING SCORED
DATASET,Macro_Name=&Macro_Name.);
    %End;
    %Else %Do;
        %Log(Text=---OK,Macro_Name=&Macro_Name.);
    %End;


%End; /* EXTERNAL MODEL DATASET USAGE */


%End; /* NO SAMPLING */
```

```
/* Create standard macro variables names with the list and number of
variables to be analyzed. */


  %If "&EM_PROPERTY_VARROLE" Eq "ALL" %Then %Do;
      %If "&EM_PROPERTY_VARLEVEL" Eq "INTERVAL" %Then %Do;
        %Let AM_NUM_VARS=&EM_NUM_INTERVAL;
        %Let AM_VARS=%EM_INTERVAL;
      %End;
      %Else %Do;
        %Let AM_NUM_VARS=&EM_NUM_CLASS;
        %Let AM_VARS=%EM_CLASS;
      %End;
  %End;
  %Else %Do;
      %If "&EM_PROPERTY_VARLEVEL" Eq "INTERVAL" %Then %Do;
        %Let AM_NUM_VARS=&EM_NUM_INTERVAL_INPUT.;
        %Let AM_VARS=%EM_INTERVAL_INPUT;
      %End;
      %Else %Do;
        %Let
AM_NUM_VARS=%Eval(&EM_NUM_BINARY_INPUT.+&EM_NUM_ORDINAL_INPUT.+&EM_NUM
_NOMINAL_INPUT.);
        %Let AM_VARS=%EM_BINARY_INPUT %EM_ORDINAL_INPUT
%EM_NOMINAL_INPUT;
      %End;
  %End;


 %Log(Text=# of variables=&AM_NUM_VARS.,Macro_Name=&Macro_Name.);
 %Log(Text=List of variables=&AM_VARS.,Macro_Name=&Macro_Name.);
```

```
/*                                BEGIN                                 */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
```

```
%If "&EM_PROPERTY_VARLEVEL." EQ "INTERVAL" %Then %Do; /* INTERVAL
UNIVARIATE STATS CALCULATION */

   %Log(Text=PROC UNIVARIATE,Macro_Name=&Macro_Name.);
```

```
   Proc Sort Data=&EM_NODEID._BIVARIATEFILE out=Univariate_Sorted;
      By %EM_TARGET;
    Run;

 %Do I=1 %to &AM_NUM_VARS.; /* DO FOR ALL INTERVAL VARS */

  %Let Var=%Scan(&AM_VARS., &I., ' ');


    Proc Univariate Data=Univariate_Sorted Noprint;
        Var &VAR.;
        By %EM_TARGET;
         output out=&EM_LIB..&EM_NODEID._Stats&I.
                    NMiss=NMiss Mean=Mean Std=Std Min=Min   Max=Max
P1=P1     Q1=Q1 Median=Median  Q3=Q3     P99=P99;

     Run;




     Data &EM_LIB..&EM_NODEID._Stats&I.;
       Format Var $32.
              NMISS
              Mean
              Std
              Min
              P1
              Q1
              Median
              Q3
              P99
              Max  Best12.;

       Set &EM_LIB..&EM_NODEID._Stats&I.;

        Var="&Var.";

        Label NMISS=
              Mean=
              Std=
              Min=
              P1=
              Q1=
              Median=
              Q3=
              P99=
              Max=;

     Run;


     %If &I. Eq 1 %Then %Do;
```

```
        Data &EM_LIB..&EM_NODEID._Stats;
           Set &EM_LIB..&EM_NODEID._Stats&I.;

         Run;

      %End;
      %Else %Do;
        Data &EM_LIB..&EM_NODEID._Stats;
           Set &EM_LIB..&EM_NODEID._Stats
               &EM_LIB..&EM_NODEID._Stats&I.;

         Run;

      %End;


 %End; /* DO FOR ALL INTERVAL VARS */
```

```
/* BEGIN - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */
/* BEGIN - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */
/* BEGIN - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */
/* BEGIN - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */
```

```
 %Log(Text=---Calculating the PEARSON correlation index for interval
variables.);
  %Put [&Sysuserid.:&Macro_Name.]---Calculating the PEARSON
correlation index for interval variables.;


    proc corr data=&EM_IMPORT_DATA. nosimple
outp=&EM_LIB..&EM_NODEID._USER_CORR_PEARSON noprint;
      var %EM_INTERVAL;
    run;


    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: Calculating the PEARSON
correlation index for interval variables.);
    %End;
    %Else %Do;
        %Log(Text=***OK.);
        %Put [&Sysuserid.:&Macro_Name.]***OK.;
    %End;
```

```
    %limpa_labels(dir=&EM_LIB.,fich=&EM_NODEID._USER_CORR_PEARSON);



  %Log(Text=---Calculating the SPEARMAN correlation index for interval
variables.);
  %Put [&Sysuserid.:&Macro_Name.]---Calculating the SPEARMAN
correlation index for interval variables.;

    proc corr data=&EM_IMPORT_DATA. nosimple
outs=&EM_LIB..&EM_NODEID._USER_CORR_SPEARMAN noprint;
      var %EM_INTERVAL;
    run;


    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: Calculating the SPEARMAN
correlation index for interval variables.);
    %End;
    %Else %Do;
        %Log(Text=***OK.);
        %Put [&Sysuserid.:&Macro_Name.]***OK.;
    %End;



    %limpa_labels(dir=&EM_LIB.,fich=&EM_NODEID._USER_CORR_SPEARMAN);




%Log(Text=---Calculating the KENDALL correlation index for interval
variables.);
 %Put [&Sysuserid.:&Macro_Name.]---Calculating the KENDALL correlation
index for interval variables.;

    proc corr data=&EM_IMPORT_DATA. nosimple
outk=&EM_LIB..&EM_NODEID._USER_CORR_KENDALL noprint;
      var %EM_INTERVAL;
    run;


    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: Calculating the KENDALL
correlation index for interval variables.);
    %End;
    %Else %Do;
        %Log(Text=***OK.);
        %Put [&Sysuserid.:&Macro_Name.]***OK.;
    %End;
```

```
    %limpa_labels(dir=&EM_LIB.,fich=&EM_NODEID._USER_CORR_KENDALL);



%Log(Text=---Calculating the HOEFFDING correlation index for interval
variables.);
 %Put [&Sysuserid.:&Macro_Name.]---Calculating the HOEFFDING
correlation index for interval variables.;



    proc corr data=&EM_IMPORT_DATA. nosimple
outh=&EM_LIB..&EM_NODEID._USER_CORR_HOEFFDING noprint;
      var %EM_INTERVAL;
    run;


    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: Calculating the HOEFFDING
correlation index for interval variables.);
    %End;
    %Else %Do;
        %Log(Text=***OK.);
        %Put [&Sysuserid.:&Macro_Name.]***OK.;
    %End;

/* END - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */
/* END - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */
/* END - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */
/* END - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */
/* END - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */
/* END - Correlation Calculation if INTERVAL VARIABLE ANALYSIS */



    %limpa_labels(dir=&EM_LIB.,fich=&EM_NODEID._USER_CORR_HOEFFDING);

%End; /* INTERVAL UNIVARIATE STATS CALCULATION */



/*                                END                                  */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR INTERVAL VARIABLES WITH BINARY TARGET */
```

```
/*                                BEGIN                              */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */

%If "&EM_PROPERTY_VARLEVEL." EQ "CLASS" %Then %Do; /* CLASS VARIABLES
- UNIVARIATE */

  %Log(Text=---Start Initial Enumeration.,Macro_Name=&Macro_Name.);

   Data &EM_LIB..&EM_NODEID._Proc_Freq;
      Format Name $32.;
      %Do I=1 %to &AM_NUM_VARS.;
        Name=Upcase("%Scan(&AM_VARS., &I., ' ')");
        Output;
      %End;

   Run;



  %Log(Text=---Load Variable Macros.,Macro_Name=&Macro_Name.);


 Proc Sql noprint;
    Select name,
           Name  into:name1-:name&AM_NUM_VARS.,
                     :Desc1-:Desc&AM_NUM_VARS.
    from &EM_LIB..&EM_NODEID._Proc_Freq;
  Quit;



  Proc Sql noprint;
    Select "Tables "||Compress(name)||"/ out=WORK."||Compress(name)||"
missing" into: freq separated by ";"
    from &EM_LIB..&EM_NODEID._Proc_Freq;
  Quit;
```

```
  %Put [&Sysuserid.:&Macro_Name.] PROC FREQ RESULTS -&Freq..-;


  Proc freq data=&EM_NODEID._BIVARIATEFILE noprint;
       &Freq.;
  Run;



 %Do i=1 %to &AM_NUM_VARS.; /* Para cada ficheiro criado pelo PROC
FREQ ...       */
                           /* "Ajustar" o formato da coluna PERCENT
ou ...       */
                           /* reduzir o ficheiro caso tenha mais de
30.000 obs. */

    %Log(Text=---Variable -&i.-&&name&i-&&Desc&i-
,Macro_Name=&Macro_Name.);


    Proc sql noprint; Select count(*) into: numobs from WORK.&&name&i;
Quit;
    %let n=&numobs.;

     %if &n. gt &EM_PROPERTY_MAXFREQ. %Then %Do;

        %Log(Text=------Final corrections.More than
&EM_PROPERTY_MAXFREQ. unique observations.,Macro_Name=&Macro_Name.);

         Data WORK.&&name&i(Keep=&&name&i COUNT);
           Format &&name&i $32.;
           Set WORK.&&name&i(obs=1 Drop=&&name&i);
           &&name&i="More than &EM_PROPERTY_MAXFREQ. unique obs.";
           COUNT=1;
          Run;


         %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
            %Error_Routine(Text=PROBLEM: Final corrections.More than
&EM_PROPERTY_MAXFREQ. unique observations.,Macro_Name=&Macro_Name.);
         %End;
         %Else %Do;
            %Log(Text=***OK.,Macro_Name=&Macro_Name.);
         %End;


     %end;
     %else %do;

        %Log(Text=------Final corrections.,Macro_Name=&Macro_Name.);

         Data WORK.&&name&i(Drop=Percent);
            Set Work.&&name&i;
             Percentage=Percent/100;
```

```
          Run;

         %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
           %Error_Routine(Text=PROBLEM: Final
corrections.,Macro_Name=&Macro_Name.);
         %End;
         %Else %Do;
            %Log(Text=***OK.,Macro_Name=&Macro_Name.);
         %End;

      %End;


       %EM_Register(Key=IE&i.,Type=DATA);


        %Log(Text=------Display table.,Macro_Name=&Macro_Name.);


         Data &&EM_USER_IE&i..;
            Set WORK.&&name&i;
            Format  Percentage Percent10.2;
          Run;

          %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
            %Error_Routine(Text=PROBLEM: Display
table.,Macro_Name=&Macro_Name.);
          %End;
          %Else %Do;
            %Log(Text=***OK.,Macro_Name=&Macro_Name.);
         %End;

         %Let J=%Eval((&i./30)*30);


         %EM_Report(Key=IE&i.,Viewtype=DATA,Autodisplay=N,Block=Proc
Freq &j.,Description=&i. &&name&i..);


         %If "&EM_PROPERTY_VARLEVEL." NE "INTERVAL" %Then %Do;

              Proc SQL Noprint;
                Select Sum(Count),
                       Count(*),
                       Sum(&&name&i.. is Missing),
                       Max(&&name&i..),
                       Min(&&name&i..)
                                        Into:Reg,
                                            :Levels,
                                            :Miss,
                                            :Max,
                                            :Min
                From WORK.&&name&i;
              Quit;

              %Let Reg=&Reg.;
```

```
            %Let Levels=&Levels.;
            %Let Miss=&Miss.;
            %Let Max=&Max.;
            %Let Min=&Min.;


            %If &i. Eq 1 %Then %Do;

              Data &EM_LIB..&EM_NODEID._Stats;
                Format Var    $32.
                       Records
                       Levels
                       Miss   Best12.
                       Min
                       Max    $50.;

                Var="&&name&i..";
                Records=&Reg.;
                Levels=&Levels.;
                Miss=&Miss.;
                Min="&Min.";
                Max="&Max.";
                Output;
              Run;

            %End;
            %Else %Do;

              Data &EM_LIB..&EM_NODEID._Stats;
                Set &EM_LIB..&EM_NODEID._Stats end=fim;


                Output;
                If fim Then Do;
                   Var="&&name&i..";
                   Records=&Reg.;
                   Levels=&Levels.;
                   Miss=&Miss.;
                   Min="&Min.";
                   Max="&Max.";
                   Output;
                End;
              Run;
            %End;

        %End;

%End;


%End; /* CLASS VARIABLES - UNIVARIATE */


/*                                  END                                    */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
```

```
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
/* PRODUCE STATISTICS FOR CLASS VARIABLES WITH BINARY TARGET */
```

```
%If "&EM_PROPERTY_MODEL." Eq "Y" and "&EM_PROPERTY_SASTable." ne ""
%Then %Do;
/* MODEL ROC - NEEDS TO BE HERE BECAUSE SCOREDAUROC WILL BE USED IN
THE NEXT BLOCK*/
/* MARKER 3333*/

%Log(Text=Calculating AUROC / GINI / KS FOR SCORED
MODEL,Macro_Name=&Macro_Name.);


%If &Em_Num_Freq. eq 1 %Then %Do;
  %Let FREQX=%EM_FREQ;
  %Let TARGETX=%EM_TARGET;
%End;
%Else %Do;
   %Let FREQX=1;
   %Let TARGETX=%EM_TARGET;
%End;

%If &FREQX. eq 1 %Then %Let FREQX=;


%MACRO_BinModelEval(/* HELP */
                    MHelp=N,
                    MDisplay=Y,

                    /* DATASET */
                    MLib=work,
                    MDSet=&EM_NODEID._BIVARIATEFILE_SCORED,

                    /* VARIABLES */
```

```
                        /*MIndVar=&Var.,*/
                        MDescVar=Statistical Model,
                        MScoreVar=&EM_PROPERTY_SCOREVAR.,
                        MDepVar=&TARGETX.,
                        MFreqVar=&FREQX.,

                        /* EVENT */
                        MHighScore=&EM_PROPERTY_HIGHSCORE.,
                        MEvent=&EM_PROPERTY_EVENT.,


                        /* PARAMETERS */
                        MRiskB=10,

                        /* EMINER */
                        MEMiner52=N,
                        MEMiner61=N,


                        /* EXCEL */
                        MExcel=N,

                        /* REPORTS */
                        MReports=Y,

                        /* ODS */

                               MODS=N,
                               MHTML=Y,
                               MPDF=N,
                               MRTF=N,
                               MODSGRAPH=N,

MODSPATH=K:\SASBusinessAnalytics\Comum\Logs_HTML\HTML,
                               MODSFILE=BME,
                               MODSSTYLE=Default,

                               /* SYSTEM */
                               MOPSYS=W,

                               /* VERSIONING */
                               MFamily=Statistical,
                               MVersion=V1);


Data &EM_LIB..&EM_NODEID._SCORE_SMODEL;
 Set SCORES;
Run;

Data ModelScoredStat;
 Set &MVOUTDS1.;


 Run;
```

```
Data ModelScoredRoc;
 Set &MVOUTDS2.(firstobs=2);

   /*M_SpecificityR=Round(M_Specificity,0.01); */

 Run;


%Put SCOREDAUROC=&MLOUTV1.=&MVOUTV1. / GINI
INDEX=%Sysfunc(Round(%Sysevalf(2*&MVOUTV1.-1),0.001)) /
&MLOUTV2.=&MVOUTV2.;
%Let SCOREDVAR=S_Sensitivity;
%Let SCORED=Y;


%End;

/* MODEL ROC - NEEDS TO BE HERE BECAUSE SCOREDAUROC WILL BE USED IN
THE NEXT BLOCK*/
/* MARKER 3333 */









/* PRODUCE VARIABLE INTERVAL/CLASS BIVARIATE/ROC GRAPH  and ROC
Calculation*/



 %Let EM_PROPERTY_HTEXT=%SYSEVALF(&EM_PROPERTY_HTEXT./10);

%Do i=1 %to &AM_NUM_VARS.;  /* DO For All VARIABLES */

%Let Var=%Scan(&AM_VARS., &I., ' ');

 %PUT =BIVARIATE=&Var.=%EM_TARGET=&EM_PROPERTY_HTEXT.(text Height);
```

```
%Log(Text=Calculating AUROC / GINI / KS FOR
TRAIN,Macro_Name=&Macro_Name.); /* ROC Calc TRAIN */


%If &Em_Num_Freq. eq 1 %Then %Do;
  %Let FREQX=%EM_FREQ;
  %Let TARGETX=%EM_TARGET;
%End;
%Else %Do;
   %Let FREQX=1;
   %Let TARGETX=%EM_TARGET;
%End;


 %Log(Text=TARGET=&TARGETX. and FREQ=&FREQX.,Macro_Name=&Macro_Name.);


Proc Sql;
  Create Table InfModelTRAIN As
  Select &Var.,
         Sum((1-&TARGETX.)*&FREQX.)
As   Event_0,
         Sum((&TARGETX.)*&FREQX.)
As   Event_1,
         Sum(&FREQX.)
As   TotObs,
         Sum((&TARGETX.=&EM_PROPERTY_HIGHSCORE.)*&FREQX.)/Sum(&FREQX.)
As   P_&EM_PROPERTY_HIGHSCORE.
  From &EM_NODEID._BIVARIATEFILE
  Group By &Var.;

Quit;






Proc Sql;

  Create Table TRAIN As
  Select A.*,
         B.P_&EM_PROPERTY_HIGHSCORE.
  From  &EM_NODEID._BIVARIATEFILE As A Join InfModelTRAIN As B
  On    A.&Var.=B.&Var.;

Quit;
```

```
%If "&EM_PROPERTY_QUANTILE." Eq "Y" %Then %Do;


Proc Rank data=InfModelTRAIN Out=InfModelTRAIN_Rank
Groups=&EM_PROPERTY_NUMQUANT.;
 Var &Var.;
 Ranks RankedVar;
Run;




proc Sql;

  Create Table InfModelTRAIN_Rank2  As
  Select RankedVar,
         Min(&Var.)       As Min_Var,
         Max(&Var.)       As Max_Var,
         Sum(Event_0)     As Event_0,
         Sum(Event_1)     As Event_1,
         Sum(TotObs)      As TotObs,
         Calculated Event_&EM_PROPERTY_HIGHSCORE./Calculated TotObs
As Score


  From  InfModelTRAIN_Rank
  Group By RankedVar;
Quit;




Data InfModelTRAIN_Rank2;
  Set InfModelTRAIN_Rank2;
   Format Code $1000.;

  Code ="If
"||put(Min_Var,Best12.)||"<=&Var.<="||put(Max_Var,Best12.)||" Then
P_&EM_PROPERTY_HIGHSCORE.="||put(Score,Best12.);


Run;




Proc Sql Noprint;

  Select code into:Codex Separated by ";" From InfModelTRAIN_Rank2;

Quit;
```

```
%Put OLAX1=&Codex.=;



Data TRAIN;
 Set &EM_NODEID._BIVARIATEFILE;

  &Codex.;

 Run;

%End; /* END QUANTILE */









%If &FREQX. eq 1 %Then %Let FREQX=;



%If &i. Eq 1 %Then %Let Aux_Report=Y; %Else %Let
Aux_Report=&EM_PROPERTY_VARROC.;
/* Tem que correr os graficos 1 vez para configurar ambiente para o
JOINT ROC */


%MACRO_BinModelEval(/* HELP */
                    MHelp=N,
                    MDisplay=Y,

                    /* DATASET */
                    MLib=work,
                    MDSet=TRAIN,

                    /* VARIABLES */

                    MDescVar=Information - Train - Variable &Var.,
                    MScoreVar=P_&EM_PROPERTY_HIGHSCORE.,
                    MDepVar=&TARGETX.,
                    MFreqVar=&FREQX.,

                    /* EVENT */
                    MHighScore=&EM_PROPERTY_HIGHSCORE.,
```

```
                        MEvent=&EM_PROPERTY_EVENT.,


                        /* PARAMETERS */
                        MRiskB=10,

                        /* EMINER */
                        MEMiner52=N,
                        MEMiner61=N,


                        /* EXCEL */
                        MExcel=N,

                        /* REPORTS */
                        MReports=&Aux_Report.,

                        /* ODS */

                                MODS=N,
                                MHTML=Y,
                                MPDF=N,
                                MRTF=N,
                                MODSGRAPH=N,

MODSPATH=K:\SASBusinessAnalytics\Comum\Logs_HTML\HTML,
                                MODSFILE=BME,
                                MODSSTYLE=Default,

                                /* SYSTEM */
                                MOPSYS=W,

                                /* VERSIONING */
                                MFamily=Statistical,
                                MVersion=V1);



Data TrainStat;
 Set &MVOUTDS1.;


 Run;


Data TrainRoc;
 Set &MVOUTDS2.(firstobs=2);

   /*M_SpecificityR=Round(M_Specificity,0.01);*/

 Run;

%Let XTRAINAUROC=%Sysfunc(Round(%Sysevalf(&MVOUTV1.),0.001));
%Let XTRAINGINI=%Sysfunc(Round(%Sysevalf(2*&MVOUTV1.-1),0.001));
%Let XTRAINKS=%Sysfunc(Round(%Sysevalf(&MVOUTV2.),0.001));
%Let XTRAINGRAN=&MVOUTV5.;
```

```
%Let
XTRAINRACIO=%Sysfunc(Round(%Sysevalf(&MVOUTV4./&MVOUTV5.),0.001));


%Let XVALAUROC=.;
%Let XVALGINI=.;
%Let XVALKS=.;
%Let XVALGRAN=.;
%Let XVALRACIO=.;

%Put TRAINAUROC=&MLOUTV1.=&MVOUTV1. / GINI
INDEX=%Sysfunc(Round(%Sysevalf(2*&MVOUTV1.-1),0.001)) /
&MLOUTV2.=&MVOUTV2.;
%Let VALAUROC=;
%Let SCOREAUROC=;



/* ROC Calc TRAIN */




%If "&Em_Import_Validate." ne "" %Then %Do; /* IF ROC VALIDATE */

%Log(Text=Calculating AUROC / GINI / KS FOR
VALIDATE,Macro_Name=&Macro_Name.);


%If "&EM_PROPERTY_GENERALIZATION." eq "N" %Then %Do;

Proc Sql;

  Create Table VALIDATE As
  Select A.*,
         B.P_&EM_PROPERTY_HIGHSCORE.
  From   &EM_NODEID._BIVARIATEFILE_VAL As A Join InfModelTRAIN As B
  On     A.&Var.=B.&Var.;

Quit;




%If "&EM_PROPERTY_QUANTILE." Eq "Y" %Then %Do;



Data VALIDATE;
```

```
 Set &EM_NODEID._BIVARIATEFILE_VAL ;

  &Codex.;

 Run;

%End;



%End;
%Else %Do;
%End;




%MACRO_BinModelEval(/* HELP */
                    MHelp=N,
                    MDisplay=Y,

                    /* DATASET */
                    MLib=work,
                    MDSet=VALIDATE,

                    /* VARIABLES */

                    MDescVar=Information - Validation - Variable &Var.,
                    MScoreVar=P_&EM_PROPERTY_HIGHSCORE.,
                    MDepVar=&TARGETX.,
                    MFreqVar=&FREQX.,

                    /* EVENT */
                    MHighScore=&EM_PROPERTY_HIGHSCORE.,
                    MEvent=&EM_PROPERTY_EVENT.,


                    /* PARAMETERS */
                    MRiskB=10,

                    /* EMINER */
                    MEMiner52=N,
                    MEMiner61=N,


                    /* EXCEL */
                    MExcel=N,

                    /* REPORTS */
```

```
                    MReports=&EM_PROPERTY_VARROC.,

                    /* ODS */

                            MODS=N,
                            MHTML=Y,
                            MPDF=N,
                            MRTF=N,
                            MODSGRAPH=N,

MODSPATH=K:\SASBusinessAnalytics\Comum\Logs_HTML\HTML,
                            MODSFILE=BME,
                            MODSSTYLE=Default,

                            /* SYSTEM */
                            MOPSYS=W,

                            /* VERSIONING */
                            MFamily=Statistical,
                            MVersion=V1);



Data ValidateStat;
 Set &MVOUTDS1.;


 Run;


Data ValidateRoc;
 Set &MVOUTDS2.(firstobs=2);

   /*M_SpecificityR=Round(M_Specificity,0.01);*/

 Run;

%Let XVALAUROC=%Sysfunc(Round(%Sysevalf(&MVOUTV1.),0.001));
%Let XVALGINI=%Sysfunc(Round(%Sysevalf(2*&MVOUTV1.-1),0.001));
%Let XVALKS=%Sysfunc(Round(%Sysevalf(&MVOUTV2.),0.001));
%Let XVALGRAN=&MVOUTV5.;
%Let XVALRACIO=%Sysfunc(Round(%Sysevalf(&MVOUTV4./&MVOUTV5.),0.001));


%Put VALAUROC=&MLOUTV1.=&MVOUTV1. / GINI
INDEX=%Sysfunc(Round(%Sysevalf(2*&MVOUTV1.-1),0.001)) /
&MLOUTV2.=&MVOUTV2.;



%End; /* IF ROC VALIDATE */



Data Auxiliar;
```

```
  Format Variable $32.;

  Variable="&Var.";
  ROC_Train=&XTRAINAUROC.;
  Gini_Train=&XTRAINGINI.;
  KS_Train=&XTRAINKS.;
  Granular_Train=&XTRAINGRAN.;
  RObs_Train=&XTRAINRACIO.;
  ROC_Val=&XVALAUROC.;
  Gini_Val=&XVALGINI.;
  KS_Val=&XVALKS.;
  Granular_Val=&XVALGRAN.;
  RObs_Val=&XVALRACIO.;
  Output;

Run;



Data &EM_LIB..&EM_NODEID._Rank_ROC;
 Set &EM_LIB..&EM_NODEID._Rank_ROC
     Auxiliar;

run;




%If "&EM_PROPERTY_JOINTROC." eq "Y" %Then %Do; /* IF JOINT GRAPH -
VARIABLE LEVEL*/


 Proc Sql;
   Create Table GraphRocAll As
   Select Coalesce(A.XScore,B.XScore)                  As T_Score,
          Coalesce(A.M_Specificity,B.M_Specificity)    As
M_Specificity,
          A.Sensitivity                                As Sensitivity,
          B.Sensitivity                                As
V_Sensitivity
    From TrainRoc As A Full Join ValidateRoc As B
    On   A.M_Specificity =B.M_Specificity;

 Quit;



Data RocAuxIni;

      T_Score=0;
      M_Specificity=0;
      Sensitivity=0;
```

```
      V_Sensitivity=0;
      P_Sensitivity=0;
      Output;

      T_Score=0;
      M_Specificity=0;
      Sensitivity=0;
      V_Sensitivity=0;
      P_Sensitivity=1;
      Output;
Run;



Data RocAuxEnd;

      T_Score=.;
      M_Specificity=1;
      Sensitivity=.;
      V_Sensitivity=.;
      P_Sensitivity=1;
      Output;

Run;


Data GraphRocAll;
 Set RocAuxIni GraphRocAll RocAuxEnd;


Run;



Title1 "Joint ROC Curve - Information - Variable &Var.";
footnote1 j=c "TRAIN &TRAINAUROC.";

footnote2 j=c "VALIDATE &VALAUROC.";

  symbol1 interpol=join
        value=dot;

  proc gplot data=GraphRocAll;
   plot (Sensitivity M_Specificity P_Sensitivity
V_Sensitivity)*M_Specificity / Overlay;
run;
quit;

%End; /* IF JOINT GRAPH - VARIABLE LEVEL */
```

```
/* VARIABLE BAR LINE GRAPH */

Data &EM_NODEID._BIVARIATEFILE;
 Set &EM_NODEID._BIVARIATEFILE;

   %EM_TARGET=(%EM_TARGET = &EM_PROPERTY_EVENT.);

Run;


%If %upcase("&Var.") Eq "AGE" %Then %Do;



%If &Em_Num_Freq. eq 1 %Then %Do;
  %Let FREQX=%EM_FREQ;
  %Let TARGETX=%EM_TARGET;
%End;
%Else %Do;
   %Let FREQX=1;
   %Let TARGETX=%EM_TARGET;
%End;


 Proc Sql;
   Create Table &EM_LIB..&EM_NODEID._AGE As
   Select &VAR.,
          Sum((1-&TARGETX.)*&FREQX.)
As   Event_0,
          Sum((&TARGETX.)*&FREQX.)
As   Event_1,
          Sum(&FREQX.)
As   TotObs,
          Sum(&TARGETX.*&FREQX.)/Sum(&FREQX.)
As   AvgDefault
   From &EM_NODEID._BIVARIATEFILE
   Group By &VAR.
   /*Order By AvgDefault Desc*/;
 Quit;

%End;


%MACRO_BARLINE(
                        /* HELP */
                        MHelp=N,

                        /* OUPUT CATALOG */
                        MGout=Work.Temp,
                        MReplace=N,

                        /* ODS */
```

```
                        /* DATASET */
                        MLib=Work,
                        MDSet=&EM_NODEID._BIVARIATEFILE,

                        /* VARIABLES */
                        MIndVar=&Var.,
                        MDepVar=%EM_TARGET,
                        MGroup=,
                        MBy=,
                        MFreq=%EM_FREQ,

                        /* GRAPH */
                        MChart=VBAR,
                        MShape=Block,
                        MFText=Helvetic,
                        MHText=&EM_PROPERTY_HTEXT.,

                        /* TITLE & FOOTNOTES */

                        MTitle1=TRAIN - Bivariate Analysis,
                        MTitle2=Independent variables vs. target
variable,

                        MFoot1=TRAIN &TRAINAUROC.,
                        MFoot2=VALIDATE &VALAUROC.,

                        MNote=,

                        /* X AXIS */

                        MAscending=N,
                        MSpace=2,
                        MLevels=&EM_PROPERTY_NUMBINS.,
                        MWidth=10,
                        MMissing=N,
                        MScale=Y,


                        /* Y AXIS */

                        MAutoref=N,
                        MType=PCT,
                        MOutside=Y,

                        /* Y2 AXIS */
                        MInterpol=Spline,
                        MLine=1,
                        MSymbol=Square,
                        MCLine=RED,
                        MLineType=MEAN,
                        MNOMARKER=,


                        /* COLORS */


                      MCText=Black,
```

```
                    MCBACK=LightSteelBlue,
                    MCPAT1=DarkBlue,
                    MCPAT2=Blue,
                    MCPAT3=Blue,
                    MCPAT4=Yellow,
                    MCPAT5=Black,
                    MCREF=Black,
                    MCFRAME=White,
                    MCOUTLINE=Black,
                    MCAXIS1=Black,
                    MCAXIS2=Black,
                    MCAXIS3=Black,

                      MVersion=V1);



%If "&Em_Import_Validate." ne "" %Then %Do; /* IF BARLINE VALIDATE */


Data &EM_NODEID._BIVARIATEFILE_VAL;
 Set &EM_NODEID._BIVARIATEFILE_VAL;

   %EM_TARGET=(%EM_TARGET = &EM_PROPERTY_EVENT.);

Run;


%If %upcase("&Var.") Eq "AGE" %Then %Do;



%If &Em_Num_Freq. eq 1 %Then %Do;
 %Let FREQX=%EM_FREQ;
 %Let TARGETX=%EM_TARGET;
%End;
%Else %Do;
  %Let FREQX=1;
  %Let TARGETX=%EM_TARGET;
%End;


 Proc Sql;
   Create Table &EM_LIB..&EM_NODEID._AGE_VAL As
   Select &VAR.,
         Sum((1-%EM_TARGET)*&FREQX.)
As   Event_0,
         Sum((%EM_TARGET)*&FREQX.)
As   Event_1,
         Sum(&FREQX.)
As   TotObs,
```

```
          Sum(%EM_TARGET*&FREQX.)/Sum(&FREQX.)
As    AvgDefault
   From &EM_NODEID._BIVARIATEFILE_VAL
   Group By &VAR.
   /*Order By AvgDefault Desc*/;
 Quit;


%End;


%MACRO_BARLINE(
                        /* HELP */
                        MHelp=N,

                        /* OUPUT CATALOG */
                        MGout=Work.Temp,
                        MReplace=N,

                        /* ODS */


                        /* DATASET */
                        MLib=Work,
                        MDSet=&EM_NODEID._BIVARIATEFILE_VAL,

                        /* VARIABLES */
                        MIndVar=&Var.,
                        MDepVar=%EM_TARGET,
                        MGroup=,
                        MBy=,
                        MFreq=%EM_FREQ,

                        /* GRAPH */
                        MChart=VBAR,
                        MShape=Block,
                        MFText=Helvetic,
                        MHText=&EM_PROPERTY_HTEXT.,

                        /* TITLE & FOOTNOTES */

                        MTitle1=VALIDATE - Bivariate Analysis,
                        MTitle2=Independent variables vs. target
variable,

                        MFoot1=TRAIN &TRAINAUROC.,
                        MFoot2=VALIDATE &VALAUROC.,

                        MNote=,

                        /* X AXIS */

                        MAscending=N,
                        MSpace=2,
                        MLevels=&EM_PROPERTY_NUMBINS.,
                        MWidth=10,
                        MMissing=N,
```

```
                        MScale=Y,


                        /* Y AXIS */

                        MAutoref=N,
                        MType=PCT,
                        MOutside=Y,

                        /* Y2 AXIS */
                        MInterpol=Spline,
                        MLine=1,
                        MSymbol=Square,
                        MCLine=ORANGE,
                        MLineType=MEAN,
                        MNOMARKER=,


                        /* COLORS */


                      MCText=Black,
                      MCBACK=LightSteelBlue,
                      MCPAT1=DarkBlue,
                      MCPAT2=Blue,
                      MCPAT3=Blue,
                      MCPAT4=Yellow,
                      MCPAT5=Black,
                      MCREF=Black,
                      MCFRAME=White,
                      MCOUTLINE=Black,
                      MCAXIS1=Black,
                      MCAXIS2=Black,
                      MCAXIS3=Black,

                        MVersion=V1);

%End; /* IF BARLINE VALIDATE */


%If "&EM_PROPERTY_MODEL." Eq "Y" and "&EM_PROPERTY_SASTable." ne ""
%Then %Do; /* IF STATISTICAL MODEL BARLINE */


Data &EM_NODEID._BIVARIATEFILE_SCORED;
 Set &EM_NODEID._BIVARIATEFILE_SCORED;

   %EM_TARGET=(%EM_TARGET = &EM_PROPERTY_EVENT.);

Run;

%If %upcase("&Var.") Eq "AGE" %Then %Do;

 Proc Sql;
   Create Table &EM_LIB..&EM_NODEID._AGE_SCORE As
```

```
   Select &VAR.,
          Sum((1-%EM_TARGET)*%EM_FREQ)
As   Event_0,
          Sum((%EM_TARGET)*%EM_FREQ)
As   Event_1,
          Sum(%EM_FREQ)
As   TotObs,
          Sum(&EM_PROPERTY_SCOREVAR.*%EM_FREQ)/Sum(%EM_FREQ)
As   AvgScore
   From &EM_NODEID._BIVARIATEFILE_SCORED(Keep=&VAR.
&EM_PROPERTY_SCOREVAR. %EM_FREQ %EM_TARGET)
   Group By &VAR.
   /*Order By AvgScore Desc*/;
 Quit;

%End;


%MACRO_BARLINE(
                        /* HELP */
                        MHelp=N,

                        /* OUPUT CATALOG */
                        MGout=Work.Temp,
                        MReplace=N,

                        /* ODS */


                        /* DATASET */
                        MLib=Work,
                        MDSet=&EM_NODEID._BIVARIATEFILE_SCORED,

                        /* VARIABLES */
                        MIndVar=&Var.,
                        MDepVar=&EM_PROPERTY_SCOREVAR.,
                        MGroup=,
                        MBy=,
                        MFreq=%EM_FREQ,

                        /* GRAPH */
                        MChart=VBAR,
                        MShape=Block,
                        MFText=Helvetic,
                        MHText=&EM_PROPERTY_HTEXT.,

                        /* TITLE & FOOTNOTES */

                        MTitle1=STATISTICAL MODEL - Bivariate Analysis,
                        MTitle2=Independent variables vs. target
variable,

                        MFoot1=TRAIN &TRAINAUROC.,
   /* MARKER 3333 */    MFoot2=MODEL &SCOREDAUROC.,

                        MNote=,
```

```
                        /* X AXIS */

                        MAscending=N,
                        MSpace=2,
                        MLevels=&EM_PROPERTY_NUMBINS.,
                        MWidth=10,
                        MMissing=N,
                        MScale=Y,


                        /* Y AXIS */

                        MAutoref=N,
                        MType=PCT,
                        MOutside=Y,

                        /* Y2 AXIS */
                        MInterpol=Spline,
                        MLine=1,
                        MSymbol=Square,
                        MCLine=GREEN,
                        MLineType=MEAN,
                        MNOMARKER=,


                        /* COLORS */


                      MCText=Black,
                      MCBACK=LightSteelBlue,
                      MCPAT1=DarkBlue,
                      MCPAT2=Blue,
                      MCPAT3=Blue,
                      MCPAT4=Yellow,
                      MCPAT5=Black,
                      MCREF=Black,
                      MCFRAME=White,
                      MCOUTLINE=Black,
                      MCAXIS1=Black,
                      MCAXIS2=Black,
                      MCAXIS3=Black,

                        MVersion=V1);

%End; /* IF STATISTICAL MODEL BARLINE */


/* VARIABLE BAR LINE GRAPH */
```

```
%End; /* DO For All VARIABLES */
```

```
%If "&EM_PROPERTY_MODELROC." eq "Y" %Then %Do; /* MODEL ROC Curve
Calculation */
```

```
Data Vars;
 Set &EM_DATA_VARIABLESET.(Keep=NAME ROLE USE Where=(ROLE='INPUT' and
USE in ('Y','D')));
Format JOIN $128.;

Join=Compress("A."||compress(NAME)||"=B."||compress(NAME));

Run;


Proc Sql Noprint;

 Select NAME Into:Var separated by ','  From Vars;

Quit;



Proc Sql Noprint;

 Select JOIN Into:SQL separated by ' and '  From Vars;

Quit;


%Put =&Var.=;
%Put =&SQL.=;
```

```
%Log(Text=Calculating AUROC / GINI / KS ,Macro_Name=&Macro_Name.);


%If &Em_Num_Freq. eq 1 %Then %Do;
 %Let FREQX=%EM_FREQ;
 %Let TARGETX=%EM_TARGET;
%End;
%Else %Do;
  %Let FREQX=1;
  %Let TARGETX=%EM_TARGET;
%End;


 %Log(Text=TARGET=&TARGETX. and FREQ=&FREQX.,Macro_Name=&Macro_Name.);


Proc Sql;
  Create Table InfModelTRAIN As
  Select &Var.,
         Sum((1-&TARGETX.)*&FREQX.)
As   Event_0,
         Sum((&TARGETX.)*&FREQX.)
As   Event_1,
         Sum(&FREQX.)
As   TotObs,
         Sum((&TARGETX.=&EM_PROPERTY_HIGHSCORE.)*&FREQX.)/Sum(&FREQX.)
As   P_&EM_PROPERTY_HIGHSCORE.
  From &EM_NODEID._BIVARIATEFILE
  Group By &Var.;

Quit;



Proc Sql;

  Create Table TRAIN As
  Select A.*,
         B.P_&EM_PROPERTY_HIGHSCORE.
  From  &EM_NODEID._BIVARIATEFILE As A Join InfModelTRAIN As B
  On    &SQL.;

Quit;



%If &FREQX. eq 1 %Then %Let FREQX=;

%MACRO_BinModelEval(/* HELP */
```

```
                        MHelp=N,
                        MDisplay=Y,

                        /* DATASET */
                        MLib=work,
                        MDSet=TRAIN,

                        /* VARIABLES */
                        /*MIndVar=&Var.,*/
                        MDescVar=Information - TRAIN - All Variables,
                        MScoreVar=P_&EM_PROPERTY_HIGHSCORE.,
                        MDepVar=&TARGETX.,
                        MFreqVar=&FREQX.,

                        /* EVENT */
                        MHighScore=&EM_PROPERTY_HIGHSCORE.,
                        MEvent=&EM_PROPERTY_EVENT.,


                        /* PARAMETERS */
                        MRiskB=10,

                        /* EMINER */
                        MEMiner52=N,
                        MEMiner61=N,


                        /* EXCEL */
                        MExcel=N,

                        /* REPORTS */
                        MReports=Y,

                        /* ODS */

                               MODS=N,
                               MHTML=Y,
                               MPDF=N,
                               MRTF=N,
                               MODSGRAPH=N,

MODSPATH=K:\SASBusinessAnalytics\Comum\Logs_HTML\HTML,
                               MODSFILE=BME,
                               MODSSTYLE=Default,

                               /* SYSTEM */
                               MOPSYS=W,

                               /* VERSIONING */
                               MFamily=Statistical,
                               MVersion=V1);

Data &EM_LIB..&EM_NODEID._SCORE_TMODEL;
 Set SCORES;
Run;
```

```
Data ModelTrainStat;
 Set &MVOUTDS1.;


 Run;


Data ModelTrainRoc;
 Set &MVOUTDS2.(firstobs=2);

   /*M_SpecificityR=Round(M_Specificity,0.01);*/

 Run;




Data &EM_LIB..&EM_NODEID._GraphRocAll;
 Set ModelTrainRoc(Keep=XScore M_Specificity Sensitivity);


 Run;


%Let TRAINAUROC=&MLOUTV1.=&MVOUTV1. / GINI
INDEX=%Sysfunc(Round(%Sysevalf(2*&MVOUTV1.-1),0.001)) /
&MLOUTV2.=&MVOUTV2.;

%Let VALAUROC=;
%Let VALVAR=;
%Let VAL=N;









%If "&Em_Import_Validate." ne "" %Then %Do; /* IF Validate DATASET
Exists */



%If "&EM_PROPERTY_GENERALIZATION." eq "N" %Then %Do;

Proc Sql;

  Create Table VALIDATE As
  Select A.*,
         B.P_&EM_PROPERTY_HIGHSCORE.
  From   &EM_NODEID._BIVARIATEFILE_VAL As A Join InfModelTRAIN As B
  On     &SQL.;
```

```
Quit;

%End;
%Else %Do;


%End;



%MACRO_BinModelEval(/* HELP */
                    MHelp=N,
                    MDisplay=Y,

                    /* DATASET */
                    MLib=work,
                    MDSet=VALIDATE,

                    /* VARIABLES */
                   /* MIndVar=&Var.,*/
                    MDescVar=Information - VALIDATE - All Variables,
                    MScoreVar=P_&EM_PROPERTY_HIGHSCORE.,
                    MDepVar=&TARGETX.,
                    MFreqVar=&FREQX.,

                    /* EVENT */
                    MHighScore=&EM_PROPERTY_HIGHSCORE.,
                    MEvent=&EM_PROPERTY_EVENT.,


                    /* PARAMETERS */
                    MRiskB=10,

                    /* EMINER */
                    MEMiner52=N,
                    MEMiner61=N,


                    /* EXCEL */
                    MExcel=N,

                    /* REPORTS */
                    MReports=Y,

                    /* ODS */

                           MODS=N,
                           MHTML=Y,
                           MPDF=N,
                           MRTF=N,
                           MODSGRAPH=N,

MODSPATH=K:\SASBusinessAnalytics\Comum\Logs_HTML\HTML,
                           MODSFILE=BME,
```

```
                              MODSSTYLE=Default,

                              /* SYSTEM */
                              MOPSYS=W,

                              /* VERSIONING */
                              MFamily=Statistical,
                              MVersion=V1);


Data &EM_LIB..&EM_NODEID._SCORE_VMODEL;
 Set SCORES;
Run;



Data ModelValidateStat;
 Set &MVOUTDS1.;


 Run;


Data ModelValidateRoc;
 Set &MVOUTDS2.(firstobs=2);

   /*M_SpecificityR=Round(M_Specificity,0.01);*/

 Run;


%Let VALAUROC=&MLOUTV1.=&MVOUTV1. / GINI
INDEX=%Sysfunc(Round(%Sysevalf(2*&MVOUTV1.-1),0.001)) /
&MLOUTV2.=&MVOUTV2.;
%Let VALVAR=V_Sensitivity;
%Let VAL=Y;


 Proc Sql;
   Create Table &EM_LIB..&EM_NODEID._GraphRocAll_1 As
   Select Coalesce(A.XScore,B.XScore)                  As XScore,
          Coalesce(A.M_Specificity,B.M_Specificity)    As
M_Specificity,
          A.Sensitivity                                As Sensitivity,
          B.Sensitivity                                As
V_Sensitivity
    From &EM_LIB..&EM_NODEID._GraphRocAll As A Full Join
ModelValidateRoc As B
    On   A.M_Specificity =B.M_Specificity;

 Quit;
```

```
%End; /* IF Validate DATASET Exists */
%Else %Do;

 Data &EM_LIB..&EM_NODEID._GraphRocAll_1;
  Set &EM_LIB..&EM_NODEID._GraphRocAll;

    V_Sensitivity=0;

 Run;

%End;
```

```
%If "&EM_PROPERTY_MODEL." Eq "Y" and "&EM_PROPERTY_SASTable." ne ""
%Then %Do; /* IF STATISTICAL MODEL */


 Proc Sql;
   Create Table &EM_LIB..&EM_NODEID._GraphRocAll_2 As
   Select Coalesce(A.XScore,B.XScore)                    As XScore,
          Coalesce(A.M_Specificity,B.M_Specificity)  As M_Specificity,
          A.Sensitivity                                  As Sensitivity,
          A.V_Sensitivity                                As
V_Sensitivity,
          B.Sensitivity                                  As
S_Sensitivity
    From &EM_LIB..&EM_NODEID._GraphRocAll_1 As A Full Join
ModelScoredRoc As B
    On   A.M_Specificity =B.M_Specificity;

 Quit;


%End; /* IF STATISTICAL MODEL */
%Else %Do;

 Data &EM_LIB..&EM_NODEID._GraphRocAll_2;
  Set &EM_LIB..&EM_NODEID._GraphRocAll_1;

    S_Sensitivity=0;

 Run;

%End;


Data RocAuxIni;

      XScore=0;
      M_Specificity=0;
```

```
      Sensitivity=0;
      V_Sensitivity=0;
      S_Sensitivity=0;
      P_Sensitivity=0;
      Output;

      XScore=0;
      M_Specificity=0;
      Sensitivity=0;
      V_Sensitivity=0;
      S_Sensitivity=0;
      P_Sensitivity=1;
      Output;

Run;



Data RocAuxEnd;

      XScore=.;
      M_Specificity=1;
      Sensitivity=.;
      V_Sensitivity=.;
      S_Sensitivity=.;
      P_Sensitivity=1;
      Output;

Run;




Data &EM_LIB..&EM_NODEID._GraphRocAll_2;
 Set RocAuxIni &EM_LIB..&EM_NODEID._GraphRocAll_2 RocAuxEnd;



Run;



Title1 "Joint ROC Curve - All Models - &Var.";
footnote1 j=c "TRAIN &TRAINAUROC.";

footnote2 j=c "VALIDATE &VALAUROC.";

footnote3 j=c "MODEL &SCOREDAUROC.";

  symbol1 interpol=join
        value=dot;

  proc gplot data=&EM_LIB..&EM_NODEID._GraphRocAll_2;
   plot (Sensitivity M_Specificity P_Sensitivity V_Sensitivity
S_Sensitivity)*M_Specificity / Overlay;
run;
quit;
```

```
%End;/* MODEL ROC Curve Calculation */
```

```
 %Log(Text=Leaving MACRO &Macro_Name.,Macro_Name=&Macro_Name.);
 %Log(Text=---------------------------,Macro_Name=&Macro_Name.);
```

```
%Mend Train_Explore_BIVARIATE;
```

```
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
/*                                    REPORT MACROS                    */
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
```

```
%Macro Report_Explore_BIVARIATE;

 %Let Macro_Name=Report_Explore_BIVARIATE;

 %Log(Text=Entering MACRO &Macro_Name.,Macro_Name=&Macro_Name.);
```

```
%EM_Register(Key=LOG,Type=DATA);

DATA &EM_USER_LOG;
 Set Work.Log;
```

```
Run;


%EM_Report(Key=LOG,Viewtype=DATA,Autodisplay=Y,Block=USER
REPORTS,Description=LOG);







%EM_Register(Key=STATS,Type=DATA);

DATA &EM_USER_STATS;
 Set &EM_LIB..&EM_NODEID._Stats;

Run;


%EM_Report(Key=STATS,Viewtype=DATA,Autodisplay=Y,Block=USER
REPORTS,Description=Statistics);



Proc Sort Data=&EM_LIB..&EM_NODEID._Rank_ROC
          Out=&EM_LIB..&EM_NODEID._Rank_ROC_Sorted;
  By Descending ROC_Train;

Run;




%EM_Register(Key=All_ROC,Type=DATA);

DATA &EM_USER_All_ROC;
 Set &EM_LIB..&EM_NODEID._Rank_ROC_Sorted(Where=(Variable ne 'ALL'));

Run;


%EM_Report(Key=All_ROC,Viewtype=DATA,Autodisplay=Y,Block=USER
REPORTS,Description=AUC Statistics);
```

```
%If "&EM_PROPERTY_VARLEVEL." EQ "INTERVAL" %Then %Do; /* BEGIN DISPLAY
CORRELATION */


 %Log(Text=---Creating the PEARSON Correlation Matrix report.);
 %Put [&Sysuserid.:&Macro_Name.]---Creating the PEARSON Correlation
Matrix report.;



      %EM_Register(Key=PEARSON,Type=DATA);

      Data &EM_USER_PEARSON(drop=_Type_);
          Set &EM_LIB..&EM_NODEID._USER_CORR_PEARSON;

           If _Type_ ne 'CORR' Then Delete;

       Run;



    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: Creating the PEARSON Correlation
Matrix report.);
    %End;
    %Else %Do;
        %Log(Text=***OK.);
        %Put [&Sysuserid.:&Macro_Name.]***OK.;
    %End;


     %EM_Report(Key=PEARSON,
                Viewtype=DATA,
                Autodisplay=Y,
                Block=Correlation,
                Description=PEARSON);







 %Log(Text=---Creating the SPEARMAN Correlation Matrix report.);
 %Put [&Sysuserid.:&Macro_Name.]---Creating the SPEARMAN Correlation
Matrix report.;



      %EM_Register(Key=SPEARMAN,Type=DATA);

      Data &EM_USER_SPEARMAN(drop=_Type_);
          Set &EM_LIB..&EM_NODEID._USER_CORR_SPEARMAN;

           If _Type_ ne 'CORR' Then Delete;

       Run;
```

```
    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: Creating the SPEARMAN Correlation
Matrix report.);
    %End;
    %Else %Do;
        %Log(Text=***OK.);
        %Put [&Sysuserid.:&Macro_Name.]***OK.;
    %End;


     %EM_Report(Key=SPEARMAN,
                Viewtype=DATA,
                Autodisplay=Y,
                Block=Correlation,
                Description=SPEARMAN);




 %Log(Text=---Creating the KENDALL Correlation Matrix report.);
 %Put [&Sysuserid.:&Macro_Name.]---Creating the KENDALL Correlation
Matrix report.;



      %EM_Register(Key=KENDALL,Type=DATA);

      Data &EM_USER_KENDALL(drop=_Type_);
          Set &EM_LIB..&EM_NODEID._USER_CORR_KENDALL;

           If _Type_ ne 'CORR' Then Delete;

       Run;



    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: Creating the KENDALL Correlation
Matrix report.);
    %End;
    %Else %Do;
        %Log(Text=***OK.);
        %Put [&Sysuserid.:&Macro_Name.]***OK.;
    %End;

     %EM_Report(Key=KENDALL,
                Viewtype=DATA,
                Autodisplay=N,
                Block=Correlation,
                Description=KENDALL);
```

```
 %Log(Text=---Creating the HOEFFDING Correlation Matrix report.);
 %Put [&Sysuserid.:&Macro_Name.]---Creating the HOEFFDING Correlation
Matrix report.;


      %EM_Register(Key=HOEFFDING,Type=DATA);

      Data &EM_USER_HOEFFDING(drop=_Type_);
          Set &EM_LIB..&EM_NODEID._USER_CORR_HOEFFDING;

           If _Type_ ne 'CORR' Then Delete;

       Run;


    %if (&syserr. ne 0 and &syserr. ne 4) %then %Do;
        %Error_Routine(Text=PROBLEM: Creating the HOEFFDING
Correlation Matrix report.);
    %End;
    %Else %Do;
        %Log(Text=***OK.);
        %Put [&Sysuserid.:&Macro_Name.]***OK.;
    %End;


     %EM_Report(Key=HOEFFDING,
                Viewtype=DATA,
                Autodisplay=N,
                Block=Correlation,
                Description=HOEFFDING);


%End; /* END DISPLAY CORRELATION */


 %Log(Text=Leaving MACRO &Macro_Name.,Macro_Name=&Macro_Name.);
 %Log(Text=--------------------------,Macro_Name=&Macro_Name.);

%Mend Report_Explore_BIVARIATE;
```

```
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
/*                                   MAIN PROCEDURE                    */
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */


%Macro Main_Explore_BIVARIATE;

   %Global Node_Name
           Count
           Error;

   %Let Node_Name=Explore_BIVARIATE;
   %Let Count=1;
   %Let Error=0;




 %Put [&Sysuserid.:&Node_Name.] STARTING NODE Explore_BIVARIATE;


*Options mprint mlogic;

    %Init_Explore_BIVARIATE;
    %If &Error. Eq 1 %Then %Goto MAIN_ERROR;

*Options nomprint nomlogic;


    %Train_Explore_BIVARIATE;
    %If &Error. Eq 1 %Then %Goto MAIN_ERROR;


    %Report_Explore_BIVARIATE;
    %If &Error. Eq 1 %Then %Goto MAIN_ERROR;


%Goto END_Explore_BIVARIATE;

%MAIN_ERROR:


   %Log(Text=ABNORMAL TERMINATION,Macro_Name=MAIN_Explore_BIVARIATE);
   %Put [&Sysuserid.:&Node_Name.] ABNORMAL TERMINATION;


%END_Explore_BIVARIATE:

      %Log(Text=TERMINATING NODE
Explore_BIVARIATE,Macro_Name=MAIN_Explore_BIVARIATE);
```

```
     Proc Sql; Select * From Work.Log; Quit;


%Mend Main_Explore_BIVARIATE;

%Main_Explore_BIVARIATE;
```