

*IBM SPSS Neural Networks 31*



**Observação**

Antes de usar estas informações e o produto suportado por elas, leia as informações nos [“Avisos” na página 17](#).

**Informações sobre o produto**

Essa edição se aplica à versão 31, liberação 0, modificação 1 de IBM® SPSS Estatísticas e a todas as liberações e modificações subsequentes até que seja indicado de outra forma em novas edições.

© Copyright International Business Machines Corporation .

---

# Índice

<b>Capítulo 1. Redes neurais.....</b>	<b>1</b>
Introdução a redes neurais.....	1
O que é uma rede neural?.....	1
Estrutura da rede neural.....	2
Perceptron Multicamada.....	3
Partições .....	5
Arquitetura .....	6
Treinamento .....	7
Saída .....	8
Salvar .....	10
Exportar .....	10
Opções .....	10
Função de base radial.....	11
Partições .....	13
Arquitetura .....	14
Saída .....	14
Salvar .....	16
Exportar .....	16
Opções .....	16
<b>Avisos.....</b>	<b>17</b>
Marcas comerciais.....	18
<b>Índice remissivo.....</b>	<b>21</b>



---

# Capítulo 1. Redes neurais

Os seguintes recursos de rede neural estão incluídos em SPSS Statistics Edição Premium ou a opção Redes Neurais.

---

## Introdução a redes neurais

As redes neurais são a ferramenta preferida para muitos aplicativos de mineração de dados preditivos devido à sua força, flexibilidade e facilidade de uso. As redes neurais preditivas são úteis, principalmente em aplicativos nos quais o processo subjacente é complexo, como:

- Prever a demanda do consumidor para aperfeiçoar os custos de produção e entrega.
- Prever a probabilidade de resposta ao marketing direto para determinar a quais famílias em uma lista de correspondência deve ser enviada uma oferta.
- Escorar um requerente para determinar o risco de estender crédito a ele.
- Detectar transações fraudulentas em um banco de dados de solicitações de seguro.

As redes neurais usadas em aplicações preditivas, como as redes de perceptron multilayer (MLP) e de função de base radial (RBF), são supervisionadas no sentido de que os resultados previstos no modelo podem ser comparados contra valores conhecidos das variáveis de destino. A opção Redes neurais permite ajustar redes MLP e RBF e salvar os modelos resultantes para escoragem.

## O que é uma rede neural?

O termo **rede neural** aplica-se a uma família de modelos fracamente relacionada, caracterizada por um grande espaço paramétrica e uma estrutura flexível, originada de estudos de funcionamento cerebral. Conforme a família crescia, a maioria dos novos modelos era projetada para aplicativos não biológicos, embora muito da terminologia associada reflita a sua origem.

Definições específicas de redes neurais são tão variadas quanto os campos nos quais elas são usadas. Enquanto nenhuma definição única abrange adequadamente toda a família de modelos, por enquanto, considere a descrição a seguir<sup>1</sup>:

Uma rede neural é um processador paralelo massivamente distribuído que tem uma propensão natural para armazenar conhecimento de experiência e torná-lo disponível para uso. Ela se assemelha ao cérebro em dois aspectos:

- Conhecimento é adquirido pela rede através de um processo de aprendizado.
- Intensidades de conexão interneurônios conhecidas como ponderações sinápticas são usadas para armazenar conhecimento.

Para uma discussão sobre por que essa definição é talvez restritiva demais, consulte<sup>2</sup>.

Para diferenciar redes neurais de métodos estatísticos tradicionais usando essa definição, o que *não* é dito é simplesmente tão significativo quanto o texto real da definição. Por exemplo, o modelo de regressão linear tradicional pode adquirir conhecimento através do método de mínimos quadrados e armazenar esse conhecimento nos coeficientes de regressão. Nesse sentido, ele é uma rede neural. De fato, é possível argumentar que regressão linear é um caso especial de determinadas redes neurais. No entanto, regressão linear tem uma estrutura de modelo rígida e um conjunto de premissas que são impostas antes do aprendizado a partir dos dados.

Por contraste, a definição acima faz demandas mínimas na estrutura de modelo e premissas. Assim, uma rede neural pode aproximar um amplo intervalo de modelos estatísticos sem requerer que você

---

<sup>1</sup> Haykin, S. 1998. *Neural Networks: A Comprehensive Foundation*, 2ª ed. Nova York: Macmillan College Publishing.

<sup>2</sup> Ripley, B. D. 1996. *Pattern Recognition and Neural Networks*. Cambridge: Cambridge University Press.

crie hipóteses adiantadas de certos relacionamentos entre as variáveis dependentes e independentes. Em vez disso, a forma dos relacionamentos é determinada durante o processo de aprendizado. Se um relacionamento linear entre as variáveis dependentes e independentes for apropriado, os resultados da rede neural devem aproximar estreitamente aqueles do modelo de regressão linear. Se um relacionamento não linear for mais apropriado, a rede neural aproximará automaticamente a estrutura de modelo "correta".

O compromisso para essa flexibilidade é que as ponderações sinápticas de uma rede neural não são facilmente interpretáveis. Dessa forma, se você estiver tentando explicar um processo subjacente que produz os relacionamentos entre as variáveis dependentes e independentes, seria melhor usar um modelo estatístico mais tradicional. No entanto, se a interpretabilidade de modelo não for importante, muitas vezes, é possível obter bons resultados de modelo mais rapidamente usando uma rede neural.

## Estrutura da rede neural

Embora as redes neurais imponham demandas mínimas na estrutura de modelo e suposições, isso é útil para entender a **arquitetura de rede** geral. A rede perceptron multiestrato perceptron (MLP) ou de função de base radial (RBF) é uma função de preditores (também chamada de entradas ou variáveis independentes) que minimiza o erro de predição de variáveis de destino (também chamadas de saídas).

Considere o conjunto de dados *bankloan.sav* fornecido com o produto, no qual você deseja poder identificar possíveis inadimplentes entre um conjunto de requerentes de empréstimos. Uma rede MLP ou RBF aplicada a esse problema é uma função das medições que minimizam o erro no padrão de predição. A figura a seguir é útil para relacionar o formato desta função.

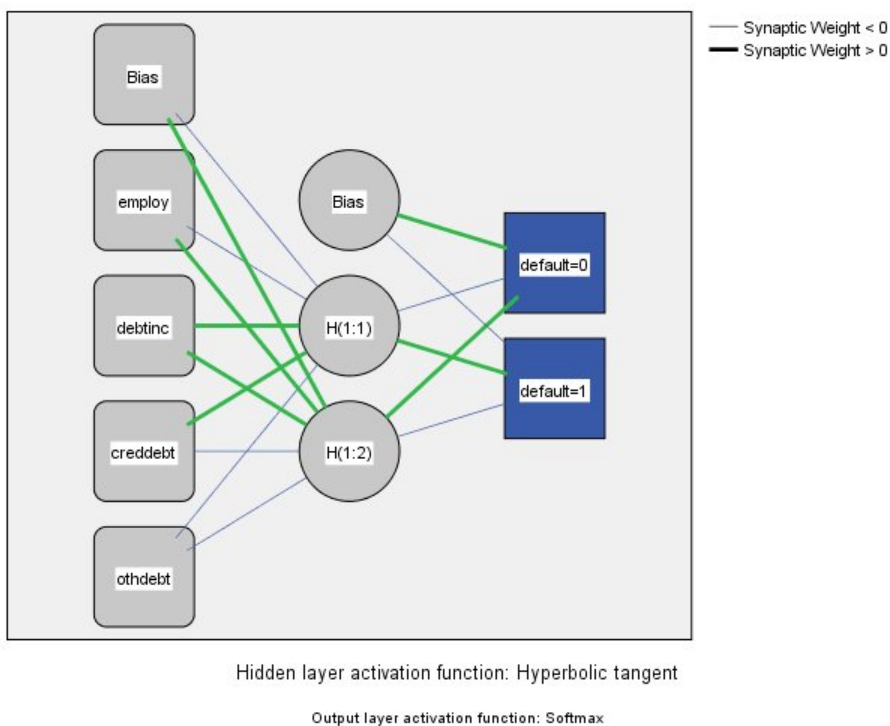


Figura 1. Arquitetura Feedforward com uma estrato oculta

Essa estrutura é conhecida como uma **arquitetura feedforward**, porque as conexões no fluxo de rede são encaminhadas da estrato de entrada para a estrato de saída sem nenhum loop de feedback. Nesta figura:

- A **estrato de entrada** contém os preditores.
- A **estrato oculta** contém nós ou unidades não observáveis. O valor de cada unidade oculta é alguma função dos preditores; a forma exata da função depende, em parte, do tipo de rede e, em parte, mediante especificações controláveis pelo usuário.

- A **estrato de saída** contém as respostas. Como o histórico de padrão é uma variável categórica com duas categorias, ele é registrado como duas variáveis indicadoras. Cada unidade de saída é alguma função das unidades ocultas. Novamente, o formato exato da função depende, em parte, do tipo de rede e, em parte, de especificações controladas pelo usuário.

A rede MLP permite uma segunda estrato oculta; nesse caso, cada unidade da segunda estrato oculta é uma função das unidades na primeira estrato oculta, e cada resposta é uma função das unidades na segunda estrato oculta.

## Perceptron Multicamada

O procedimento Perceptron Multiestrato (MLP) produz um modelo preditivo para uma ou mais variáveis dependentes (destino) com base nos valores das variáveis preditoras.

**Exemplos.** A seguir estão dois cenários que usam o procedimento MLP:

Um responsável pelo empréstimo em um banco precisa ser capaz de identificar características que são indicativos de pessoas que são suscetíveis a dar calote em empréstimos e usar essas características para identificar os riscos de crédito bons e ruins. Usando uma amostra de clientes antigos, é possível treinar um perceptron multiestrato, validar a análise usando uma amostra de validação de clientes antigos e, em seguida, usar a rede para classificar possíveis clientes como riscos de crédito bons ou ruins.

Um sistema de hospital está interessado em rastrear custos e durações de internações de pacientes admitidos para tratamento de infarto do miocárdio (MI, ou "ataque cardíaco"). A obtenção de estimativas precisas dessas medidas permite que a administração gerencie de forma apropriada o espaço da cama disponível conforme os pacientes são tratados. Usando os registros de tratamento de uma amostra de pacientes que receberam tratamento para MI, o administrador pode treinar uma rede para prever o custo e a duração da internação.

Considerações de dados

**Variáveis dependentes.** As variáveis dependentes podem ser:

- *Nominal.* Uma variável pode ser tratada como nominal quando seus valores representam categorias sem uma classificação intrínseca (por exemplo, o departamento da empresa na qual um funcionário trabalha). Exemplos de variáveis nominais incluem região, código de endereçamento postal e filiação religiosa.
- *Ordinal.* Uma variável pode ser tratada como ordinal quando seus valores representam categorias com alguma classificação intrínseca (por exemplo, níveis de satisfação de serviço de altamente insatisfeito para altamente satisfeito). Exemplos de variáveis ordinais incluem escores de atitude que representam o grau de satisfação ou de confiança e os escores de classificação de preferência.
- *scale.* Uma variável pode ser tratada como escala (contínua) quando seus valores representam categorias ordenadas com uma métrica significativa, para que as comparações de distância entre valores sejam apropriadas. Exemplos de variáveis de escala incluem idade em anos e rendimento em milhares de dólares.

O procedimento supõe que o nível de medição apropriado foi designado a todas as variáveis dependentes; no entanto, é possível mudar temporariamente o nível de medição para uma variável, clicando com o botão direito na variável na lista de variáveis de origem e selecionando um nível de medição no menu pop-up.

Um ícone ao lado de cada variável na lista de variáveis identifica o nível de medição e o tipo de dados:




Tabela 1. Ícones do nível de medição				
	Numérico	Sequência	Data	Hora
Escala (Contínua)		n/d		

Tabela 1. Ícones do nível de medição (continuação)				
	Numérico	Sequência	Data	Hora
Ordinal				
Nominal				

**Variáveis predictoras.** Os preditores podem ser especificados como fatores (categóricos) ou covariáveis (escala).

**Codificação de variável categórica.** O procedimento recodifica temporariamente preditores categóricos e variáveis dependentes utilizando a codificação um-de-c durante o procedimento. Se houver c categorias de uma variável, então a variável é armazenada como c vetores, com a primeira categoria denotada (1,0,...,0), a próxima categoria (0,1,0,...,0),..., e a categoria final (0,0,...,0,1).

Esse esquema de codificação aumenta o número de ponderações sinápticas e pode resultar em um treinamento mais lento; no entanto, os métodos de codificação mais "compactos" geralmente levam a redes neurais com um ajuste inadequado. Se o seu treinamento em rede está prosseguindo muito lentamente, tente reduzir o número de categorias em seus preditores categóricos ao combinar categorias similares ou eliminando casos que tenham categorias extremamente raras.

Toda a codificação um-de-c é baseada nos dados de treinamento, mesmo que uma amostra de teste ou de validação esteja definida (consulte “Partições ” na página 5). Portanto, se as amostras de teste ou de validação contiverem casos com categorias do preditor que não estão presentes nos dados de treinamento, esses casos não serão usados pelo procedimento ou na escoragem. Se as amostras de teste ou de validação contiverem casos com categorias de variável dependente que não estão presentes nos dados de treinamento, esses casos não serão usados pelo procedimento, mas poderão ser escorados.

**Escalando novamente.** As variáveis dependentes de escala e covariáveis são escaladas novamente por padrão para melhorar o treinamento de rede. Toda a nova escala é executada com base nos dados de treinamento, mesmo que uma amostra de teste ou de validação esteja definida (consulte “Partições ” na página 5). Ou seja, dependendo do tipo de nova escala, a média, o desvio padrão, o valor mínimo ou o valor máximo de uma covariável ou variável dependente são calculados usando apenas os dados de treinamento. Se você especificar uma variável para definir partições, é importante que essas covariáveis ou variáveis dependentes tenham distribuições similares nas amostras de treinamento, teste ou validação.

**Ponderações de frequência.** As ponderações de frequência são ignoradas por este procedimento.

**Replicando resultados.** Se desejar replicar seus resultados de forma exata, use o mesmo valor de inicialização para o gerador de números aleatórios, a mesma ordem de dados e a mesma ordem da variável, além de usar as mesmas configurações de procedimento. A seguir estão mais detalhes sobre esse problema:

- **Geração de número aleatório.** O procedimento usa a geração de número aleatório durante a designação aleatória de partições, subamostragem aleatória para inicialização de ponderações sinápticas, subamostragem aleatória para seleção automática de arquitetura e o algoritmo de arrefecimento simulado usado na inicialização da ponderação e na seleção automática de arquitetura. Para reproduzir os mesmos resultados aleatórios no futuro, use o mesmo valor de inicialização para o gerador de números aleatórios antes de cada execução do procedimento Perceptron Multiestrato.
- **Ordem de casos.** Os métodos de treinamento On-line e de Minilote (consulte “Treinamento ” na página 7) são explicitamente dependentes da ordem de casos; no entanto, até mesmo o treinamento de Lote é dependente da ordem de casos, porque a inicialização de ponderações sinápticas envolve a subamostragem do conjunto de dados.

Para minimizar os efeitos da ordem, ordene aleatoriamente os casos. Para verificar a estabilidade de uma determinada solução, talvez você queira obter várias soluções diferentes com casos ordenados em



diferentes ordens aleatórias. Em situações com tamanhos de arquivos extremamente grandes, podem ser feitas várias execuções com uma amostra de casos ordenados em diferentes ordens aleatórias.

- **Ordem da variável.** Os resultados podem ser influenciados pela ordem de variáveis nas listas de fatores e de covariáveis, devido ao padrão diferente de valores iniciais designados quando a ordem da variável é mudada. Assim como os efeitos da ordem de casos, é possível tentar diferentes ordens de variáveis (apenas arrastar e soltar nas listas de fatores e de covariáveis) para avaliar a estabilidade de uma determinada solução.

Criando uma rede Perceptron Multiestrato

Nos menus, escolha:

**Analisar > Redes neurais > Perceptron Multiestrato...**

1. Selecione pelo menos uma variável dependente.
2. Selecione pelo menos um fator ou covariável.

Opcionalmente, na guia Variáveis, é possível mudar o método para escalar novamente as covariáveis. As opções são:

- **Padronizado.** Subtraia a média e divida pelo desvio padrão,  $(x - \text{mean})/s$ .
- **Normalizado.** Subtraia o mínimo e divida pelo intervalo,  $(x - \text{min})/(\text{max} - \text{min})$ . Os valores normalizados estão entre 0 e 1.
- **Normalizado ajustado.** Versão ajustada da subtração do mínimo e da divisão pelo intervalo,  $[2 * (x - \text{min})/(\text{max} - \text{min})] - 1$ . Os valores normalizados ajustados estão entre -1 e 1.
- **Nenhum.** Nenhuma nova escala de covariáveis.

Campos com nível de medição desconhecido

O alerta de Nível de Medição é exibido quando o nível de medição para uma ou mais variáveis (campos) no conjunto de dados é desconhecido. Como o nível de medição afeta o cálculo de resultados para este procedimento, todas as variáveis devem ter um nível de medição definido.

**Dados de varredura.** Lê os dados no conjunto de dados ativo e designa o nível de medição padrão para quaisquer campos com um nível de medição desconhecido atualmente. Se o conjunto de dados for grande, isso poderá demorar algum tempo.

**Designar Manualmente.** Abre um diálogo que lista todos os campos com um nível de medição desconhecido. É possível utilizar este diálogo para designar o nível de medição para esses campos. Também é possível designar o nível de medição na Visualização de Variável do Editor de Dados.

Como o nível de medição é importante para este procedimento, não é possível acessar o diálogo para executar este procedimento até que todos os campos possuam um nível de medição definido.

## Partições

**Particionar conjunto de dados.** Esse grupo especifica o método de particionamento do conjunto de dados ativo em amostras de treinamento, de teste e de validação. A **amostra de treinamento** abrange os registros de dados usados para treinar a rede neural; alguma porcentagem dos casos no conjunto de dados deve ser designada à amostra de treinamento para obter um modelo. A **amostra de teste** é um conjunto independente de registros de dados usados para rastrear erros durante o treinamento para evitar o excesso de treinamento. É altamente recomendado que você crie uma amostra de teste, dessa forma, o treinamento de rede geralmente será mais eficiente se a amostra de teste for menor que a amostra de treinamento. A **amostra de validação** é outro conjunto independente de registros de dados usados para avaliar a rede neural final; o erro para a amostra de validação fornece uma estimativa "honesta" da capacidade preditiva do modelo, porque os casos de validação não foram usados para construir o modelo.

- **Designar casos aleatoriamente com base no número relativo de casos.** Especifique o número relativo (razão) de casos designados aleatoriamente a cada amostra (treinamento, teste e validação). A coluna % relata a porcentagem de casos que serão designados a cada amostra, com base nos números relativos especificados.

Por exemplo, especificar 7, 3, 0 como os números relativos para amostras de treinamento, teste e validação corresponde a 70%, 30% e 0%. Especificar 2, 1, 1 como os números relativos corresponde a 50%, 25% e 25%; 1, 1, 1 corresponde a dividir o conjunto de dados em terços iguais entre treinamento, teste e validação.

- **Usar variável de particionamento para designar casos.** Especifique uma variável numérica que designa cada caso no conjunto de dados ativo à amostra de treinamento, teste ou validação. Os casos com um valor positivo na variável são designados à amostra de treinamento, os casos com um valor 0 à amostra de teste, e os casos com um valor negativo à amostra de validação. Casos com um valor omissivo do sistema são excluídos da análise. Quaisquer valores omissivos do usuário para a variável de partição são sempre tratados como válidos.

*Nota:* Usar uma variável de particionamento não garantirá resultados idênticos em execuções sucessivas do procedimento. Consulte "Replicando resultados" no tópico principal [Perceptron multiestrato](#).

## Arquitetura

A guia Arquitetura é usada para especificar a estrutura da rede. O procedimento pode selecionar a "melhor" arquitetura automaticamente, ou é possível especificar uma arquitetura customizada.

A seleção automática de arquitetura constrói uma rede com uma estrato oculta. Especifique os números mínimo e máximo de unidades permitidas na estrato oculta, e a seleção automática de arquitetura calcula o "melhor" número de unidades na estrato oculta. A seleção automática de arquitetura usa as funções de ativação padrão para as estratos oculta e de saída.

A seleção de arquitetura customizada fornece controle de especialista sobre as estratos oculta e de saída e pode ser mais útil quando você sabe com antecedência qual a arquitetura desejada ou quando precisa ajustar os resultados da seleção automática de arquitetura.

### Camadas ocultas

A estrato oculta contém nós de rede não observáveis (unidades). Cada unidade oculta é uma função do soma ponderada das entradas. A função é a função de ativação e os valores das ponderações são determinados pelo algoritmo de estimação. Se a rede contiver uma segunda camada oculta, cada unidade oculta na segunda camada será uma função da soma ponderada das unidades na primeira camada oculta. A mesma função de ativação é utilizada em ambas as camadas.

*Número de camadas ocultas.* Um perceptron multicamadas pode ter uma ou duas camadas ocultas.

*Função de Ativação.* A função de ativação "liga" a soma ponderada das unidades em uma camada aos valores de unidades na camada seguinte.

- **Tangente hiperbólica.** Esta função tem o formato:  $\gamma(c) = \tanh(c) = (e^c - e^{-c}) / (e^c + e^{-c})$ . Ela usa argumentos com valores reais e transforma-os no intervalo  $(-1, 1)$ . Quando a seleção automática de arquitetura é usada, essa é a função de ativação para todas as unidades nas estratos ocultas.
- **Sigmoide.** Esta função tem o formato:  $\gamma(c) = 1 / (1 + e^{-c})$ . Ela usa argumentos com valores reais e transforma-os no intervalo  $(0, 1)$ .

*Número de unidades.* O número de unidades em cada camada oculta pode ser especificado explicitamente ou determinado automaticamente pelo algoritmo de estimação.

### Camada de saída

A estrato de saída contém as variáveis de destino (dependentes).

*Função de Ativação.* A função de ativação "liga" a soma ponderada das unidades em uma camada aos valores de unidades na camada seguinte.

- **Identidade.** Essa função tem o formato:  $\gamma(c) = c$ . Ela usa argumentos com valores reais e retorna-os inalterados. Quando a seleção automática de arquitetura for usada, essa será a função de ativação para unidades na estrato de saída, se houver variáveis dependentes de escala.
- **Softmax.** Essa função tem o formato:  $\gamma(c_k) = \exp(c_k) / \sum_j \exp(c_j)$ . Ela usa um vetor de argumentos com valores reais e transforma-o em um vetor cujos elementos estão no intervalo  $(0, 1)$  e são somados a 1. Softmax estará disponível somente se todas as variáveis dependentes forem categóricas. Quando a

seleção automática de arquitetura for usada, essa será a função de ativação para unidades na estrato de saída, se todas as variáveis dependentes forem categóricas.

- **Tangente hiperbólica.** Esta função tem o formato:  $\gamma(c) = \tanh(c) = (e^c - e^{-c}) / (e^c + e^{-c})$ . Ela usa argumentos com valores reais e transforma-os no intervalo  $(-1, 1)$ .
- **Sigmoide.** Esta função tem o formato:  $\gamma(c) = 1 / (1 + e^{-c})$ . Ela usa argumentos com valores reais e transforma-os no intervalo  $(0, 1)$ .

**Nova escala de variáveis dependentes de escala.** Esses controles estarão disponíveis somente se pelo menos uma variável dependente de escala tiver sido selecionada.

- **Padronizado.** Subtraia a média e divida pelo desvio padrão,  $(x - \text{mean}) / s$ .
- **Normalizado.** Subtraia o mínimo e divida pelo intervalo,  $(x - \text{min}) / (\text{max} - \text{min})$ . Os valores normalizados estão entre 0 e 1. Este é o método de reajuste de escala necessário para variáveis dependentes de escala se a camada de saída utilizar a função de ativação sigmoide. A opção de correção especifica um pequeno número  $\epsilon$  que é aplicado como uma correção à fórmula de nova escala; essa correção assegura que todos os valores de variável dependente escalada novamente estejam no intervalo da função de ativação. Em específico, os valores 0 e 1, que ocorrem na fórmula não corrigida quando  $x$  usa seus valores mínimo e máximo, definem os limites do intervalo da função sigmoide, mas não estão nesse intervalo. A fórmula corrigida é  $[x - (\text{min} - \epsilon)] / [(\text{max} + \epsilon) - (\text{min} - \epsilon)]$ . Especifique um número maior ou igual a 0.
- **Normalizado ajustado.** Versão ajustada da subtração do mínimo e da divisão pelo intervalo,  $[2 * (x - \text{min}) / (\text{max} - \text{min})] - 1$ . Os valores normalizados ajustados estão entre  $-1$  e  $1$ . Este é o método de reajuste de escala necessário para variáveis dependentes de escala se a camada de saída utilizar a função de ativação tangente hiperbólica. A opção de correção especifica um pequeno número  $\epsilon$  que é aplicado como uma correção à fórmula de nova escala; essa correção assegura que todos os valores de variável dependente escalada novamente estejam no intervalo da função de ativação. Em específico, os valores  $-1$  e  $1$ , que ocorrem na fórmula não corrigida quando  $x$  usa seus valores mínimo e máximo, definem os limites do intervalo da função de tangente hiperbólica, mas não estão nesse intervalo. A fórmula corrigida é  $\{2 * [(x - (\text{min} - \epsilon)) / ((\text{max} + \epsilon) - (\text{min} - \epsilon))] - 1\}$ . Especifique um número maior ou igual a 0.
- **Nenhum.** Nenhuma nova escala de variáveis dependentes de escala.

## Treinamento

A guia Treinamento é usada para especificar como a rede deve ser treinada. O tipo de treinamento e o algoritmo de otimização determinam quais opções de treinamento estão disponíveis.

**Tipo de treinamento.** O tipo de treinamento determina como a rede processa os registros. Selecione um dos seguintes tipos de treinamento:

- **Lote.** Atualiza as ponderações sinápticas somente após transmitir todos os registros de dados de treinamento; ou seja, o treinamento em lote usa informações de todos os registros no conjunto de dados de treinamento. O treinamento em lote geralmente é preferencial, porque minimiza diretamente o total de erros; no entanto, o treinamento em lote pode precisar atualizar muitas vezes as ponderações até que uma das regras de parada seja atendida e, portanto, pode precisar de muitas transmissões de dados. Ele é mais útil para conjuntos de dados "menores".
- **On-line.** Atualiza as ponderações sinápticas após cada registro de informações; ou seja, o treinamento on-line usa informações de um registro por vez. O treinamento on-line obtém continuamente um registro e atualiza as ponderações até que uma das regras de parada seja atendida. Se todos os registros forem usados uma vez e nenhuma das regras de parada for atendida, o processo continuará reciclando os registros de dados. O treinamento on-line é superior ao de lote para conjuntos de dados "maiores" com preditores associados; ou seja, se houver muitos registros e muitas entradas, e seus valores não forem independentes uns dos outros, então o treinamento on-line poderá obter mais rapidamente uma resposta razoável do que o treinamento em lote.
- **Minilote.** Divide os registros de dados de treinamento em grupos de tamanho aproximadamente igual, em seguida, atualiza as ponderações sinápticas após transmitir um grupo; ou seja, o treinamento de minilote usa informações de um grupo de registros. Em seguida, o processo recicla o grupo de dados, se necessário. O treinamento de minilote oferece um compromisso entre o treinamento em lote e

on-line, e pode ser melhor para conjuntos de dados de "médio porte". O procedimento pode determinar automaticamente o número de registros de treinamento por minilote, ou é possível especificar um número inteiro maior que 1 e menor ou igual ao número máximo de casos a serem armazenados na memória. É possível configurar o número máximo de casos a serem armazenados na memória na guia Opções.

**Algoritmo de otimização.** Esse é o método usado para estimar as ponderações sinápticas.

- **Gradiente conjugado escalado.** As suposições que justificam o uso de métodos gradientes conjugados se aplicam apenas a tipos de treinamento em lote, portanto, esse método não está disponível para treinamento on-line ou de minilote.
- **Descida de gradiente.** Esse método deve ser usado com treinamento on-line ou de minilote; ele também pode ser usado com treinamento em lote.

**Opções de treinamento.** As opções de treinamento permitem otimizar o algoritmo de otimização. Geralmente, não será necessário mudar essas configurações, a menos que a rede tenha problemas com a estimação.

As opções de treinamento para o algoritmo de gradiente conjugado escalado incluem:

- **Lambda inicial.** O valor inicial do parâmetro lambda para o algoritmo de gradiente conjugado escalado. Especifique um número maior que 0 e menor que 0,000001.
- **Sigma inicial.** O valor inicial do parâmetro sigma para o algoritmo de gradiente conjugado escalado. Especifique um número maior que 0 e menor que 0,0001.
- **Centro de intervalo e offset de intervalo.** O centro de intervalo ( $a_0$ ) e o offset de intervalo ( $a$ ) definem o intervalo  $[a_0 - a, a_0 + a]$ , no qual os vetores de ponderação são gerados aleatoriamente quando o arrefecimento simulado é usado. O arrefecimento simulado é usado para divisão fora de um mínimo local, com o objetivo de encontrar o mínimo global, durante a aplicação do algoritmo de otimização. Essa abordagem é usada na inicialização de ponderação e na seleção automática de arquitetura. Especifique um número para o centro do intervalo e um número maior que 0 para o offset do intervalo.

As opções de treinamento para o algoritmo de descida de gradiente incluem:

- **Taxa de aprendizado inicial.** O valor inicial da taxa de aprendizado para o algoritmo de descida de gradiente. Uma taxa de aprendizado mais alta significa que a rede irá treinar mais rápido, possivelmente ao custo de se tornar instável. Especifique um número maior que 0.
- **Limite inferior para taxa de aprendizado.** O limite inferior na taxa de aprendizado para o algoritmo de descida de gradiente. Essa configuração se aplica somente ao treinamento on-line e de minilote. Especifique um número maior que 0 e menor que a taxa de aprendizado inicial.
- **Momento.** O parâmetro de momento inicial para o algoritmo de descida de gradiente. O termo momento ajuda a evitar instabilidades causadas por uma taxa de aprendizado muito alta. Especifique um número maior que 0.
- **Redução da taxa de aprendizado, em épocas.** O número de épocas ( $p$ ), ou transmissões de dados da amostra de treinamento, necessárias para reduzir a taxa de aprendizado inicial para o limite inferior da taxa de aprendizado, quando a descida de gradiente é usada com treinamento on-line ou de minilote. Isso fornece controle do fator de declínio da taxa de aprendizado  $\beta = (1/p K) * \ln(\eta_0/\eta_{low})$ , em que  $\eta_0$  é a taxa de aprendizado inicial,  $\eta_{low}$  é o limite inferior na taxa de aprendizado e  $K$  é o número total de minilotes (ou o número de registros de treinamento para treinamento on-line) no conjunto de dados de treinamento. Especifique um número inteiro maior que 0.

## Saída

**Estrutura de rede.** Exibe informações de sumarização sobre a rede neural.

- **DESCRIPTION.** Exibe informações sobre a rede neural, incluindo as variáveis dependentes, o número de unidades de entrada e de saída, o número de camadas e unidades ocultas e as funções de ativação.
- **Diagrama.** Exibe o diagrama de rede como um gráfico não editável. Observe que, à medida que o número de covariáveis e os níveis de fatores aumentam, o diagrama se torna mais difícil de ser interpretado.

- **Ponderações sinápticas.** Exibe as estimativas de coeficiente que mostram o relacionamento entre as unidades em uma determinada estrato com as unidades na estrato seguinte. As ponderações sinápticas são baseadas na amostra de treinamento, mesmo que o conjunto de dados ativo seja particionado em dados de treinamento, de teste e de validação. Observe que o número de ponderações sinápticas pode se tornar bastante grande e que essas ponderações geralmente não são usadas para interpretar resultados de rede.

**Desempenho da rede.** Exibe os resultados usados para determinar se o modelo é "válido". *Nota:* Os gráficos nesse grupo são baseados nas amostras combinadas de treinamento e teste ou somente na amostra de treinamento, se não houver nenhuma amostra de teste.

- **Sumarização do modelo.** Exibe uma sumarização dos resultados da rede neural por partição e geral, incluindo o erro, o erro relativo ou porcentagem de predições incorretas, a regra de parada usada para parar o treinamento e o tempo de treinamento.

O erro é a soma dos quadrados dos erros quando a função de ativação de identidade, de sigmoide ou de tangente hiperbólica é aplicada à estrato de saída. É o erro de entropia cruzada quando a função de ativação softmax é aplicada à estrato de saída.

São exibidos erros relativos ou porcentagens de predições incorretas, dependendo dos níveis de medição da variável dependente. Se alguma variável dependente tiver nível de medição de escala, a média geral de erros relativos (relativos ao modelo de média) será exibida. Se todas as variáveis dependentes forem categóricas, a porcentagem média de predições incorretas será exibida. Os erros relativos ou porcentagens de predições incorretas também são exibidos para variáveis dependentes individuais.

- **Resultados de classificação.** Exibe uma tabela de classificação para cada variável dependente categórica por partição e geral. Cada tabela fornece o número de casos classificados corretamente e incorretamente para cada categoria de variável dependente. A porcentagem do total de casos que foram classificados corretamente também é relatada.
- **Curva ROC.** Exibe uma curva ROC (Receiver Operating Characteristic) para cada variável dependente categórica. Também exibe uma tabela que fornece a área em cada curva. Para uma determinada variável dependente, o gráfico ROC exibe uma curva para cada categoria. Se a variável dependente tiver duas categorias, cada curva tratará a categoria com problema como o estado positivo versus a outra categoria. Se a variável dependente tiver mais de duas categorias, cada curva tratará a categoria com problema como o estado positivo versus a agregação de todas as outras categorias.
- **Gráfico de ganhos acumulativos.** Exibe um gráfico de ganhos acumulativos para cada variável dependente categórica. A exibição de uma curva para cada categoria de variável dependente é a mesma que para curvas ROC.
- **Gráfico de ganho.** Exibe um gráfico de ganho para cada variável dependente categórica. A exibição de uma curva para cada categoria de variável dependente é a mesma que para curvas ROC.
- **Gráfico predito por observado.** Exibe um gráfico de valor predito por observado para cada variável dependente. Para variáveis dependentes categóricas, os boxplots em cluster de pseudoprobabilidades preditas são exibidos em cada categoria de resposta, com a categoria de resposta observada como a variável de cluster. Para variáveis dependentes de escala, é exibido um gráfico de dispersão.
- **Gráfico de resíduo por predito.** Exibe um gráfico de valor de resíduos por predito para cada variável dependente de escala. Não deve haver nenhum padrão visível entre valores de resíduos e preditos. Esse gráfico é produzido apenas para variáveis dependentes de escala.

**Sumarização do processamento de caso.** Exibe a tabela de sumarização de processamento de caso, que sumariza o número de casos incluídos e excluídos na análise, no total e por amostras de treinamento, de teste e de validação.

**Análise de importância da variável independente.** Executa uma análise de sensibilidade, que calcula a importância de cada preditor na determinação da rede neural. A análise é baseada nas amostras combinadas de treinamento e teste, ou apenas na amostra de treinamento, se não houver nenhuma amostra de teste. Isso cria uma tabela e um gráfico que exibem a importância e a importância normalizada para cada preditor. Observe que a análise de sensibilidade é dispendiosa e demorada do ponto de vista computacional, se houver grandes quantidades de preditores ou casos.

## Salvar

A guia Salvar é usada para salvar predições como variáveis no conjunto de dados.

- **Salvar valor predito ou categoria para cada variável dependente.** Isso salva o valor predito para variáveis dependentes de escala e a categoria predita para variáveis dependentes categóricas.
- **Salvar pseudoprobabilidade ou categoria predita para cada variável dependente.** Isso salva as pseudoprobabilidades preditas para variáveis dependentes categóricas. Uma variável separada é salva para cada uma das primeiras  $n$  categorias, em que  $n$  é especificado na coluna **Categorias a serem salvas**.

**Nomes de Variáveis Salvas.** A geração de nome automática assegura que você mantenha todo o seu trabalho. Nomes customizados permitem descartar/substituir resultados de execuções anteriores sem primeiro excluir as variáveis salvas no Editor de Dados.

Probabilidades e pseudoprobabilidades

As variáveis dependentes categóricas com ativação de softmax e erro de entropia cruzada terão um valor predito para cada categoria, em que cada valor predito é a probabilidade de que o caso pertença à categoria.

As variáveis dependentes categóricas com a soma dos quadrados dos erros terão um valor predito para cada categoria, mas os valores preditos não podem ser interpretados como probabilidades. O procedimento salva essas pseudoprobabilidades preditas, mesmo que algumas sejam menores que 0 ou maiores que 1, ou a soma para uma determinada variável dependente não seja 1.

O ROC, ganhos acumulativos e gráficos de ganho (consulte “Saída ” na página 8 ) são criados com base em pseudoprobabilidades. No caso de qualquer uma das pseudoprobabilidades ser inferior a 0 ou superior a 1 ou a soma de uma determinada variável não for 1, elas são primeiro redimensionadas a ficar entre 0 e 1 e a somar 1. As pseudoprobabilidades são redimensionadas dividindo-se por sua soma. Por exemplo, se um caso tiver pseudoprobabilidades preditas de 0,50, 0,60 e 0,40 para uma variável dependente de três categorias, cada pseudoprobabilidade será dividida pela soma 1,50 para obter 0,33, 0,40 e 0,27.

Se qualquer uma das pseudoprobabilidades for negativa, o valor absoluto da mais baixa será incluído em todas as pseudoprobabilidades antes da nova escala acima. Por exemplo, se as pseudoprobabilidades forem -0,30, 0,50 e 1,30, primeiro inclua 0,30 em cada valor para obter 0,00, 0,80 e 1,60. Em seguida, divida cada novo valor pela soma 2,40 para obter 0,00, 0,33 e 0,67.

## Exportar

A guia Exportar é usada para salvar as estimativas de ponderação sináptica para cada variável dependente em um arquivo XML (PMML). É possível usar esse arquivo de modelo para aplicar as informações de modelo a outros arquivos de dados para propósitos de escoragem. Essa opção não estará disponível se arquivos de divisão tiverem sido definidos.

## Opções

**Valores omissos de usuário.** Os fatores devem ter valores válidos para um caso a serem incluídos na análise. Estes controles permitem decidir se valores omissos de usuário são tratados como válidos entre fatores e variáveis dependentes categóricas.

**Regras de Parada.** Essas são as regras que determinam quando parar o treinamento da rede neural. O treinamento continua através de pelo menos uma passagem de dados. O treinamento pode então ser parado de acordo com os seguintes critérios, que são verificados na ordem listada. Nas definições de regra de parada que se seguem, um passo corresponde a uma passagem de dados para métodos on-line e de minilote e uma iteração para o método em lote.

- **Máximo de passos sem uma diminuição de erros.** O número de passos a serem permitidos antes da verificação de uma diminuição de erros. Se não houver nenhuma diminuição de erros após o número especificado de passos, o treinamento será parado. Especifique um número inteiro maior que 0. Também é possível especificar qual amostra de dados é usada para computar o erro. **Escolher**

**automaticamente** usa a amostra de teste, se ela existir; caso contrário, usa a amostra de treinamento. Observe que o treinamento em lote garante uma diminuição no erro de amostra de treinamento após cada passagem de dados; portanto, essa opção se aplica ao treinamento em lote apenas se existir uma amostra de teste. **Dados de treinamento e de teste** verifica o erro para cada uma dessas amostras; essa opção se aplica apenas se existir uma amostra de teste.

*Nota:* Após cada passagem de dados completa, o treinamento on-line e de minilote requer uma passagem de dados extra para calcular o erro de treinamento. Essa passagem de dados extra pode deixar o treinamento consideravelmente lento, portanto, geralmente é recomendado fornecer uma amostra de teste e selecionar **Escolher automaticamente** em qualquer caso.

- **Tempo máximo de treinamento.** Escolha se deseja especificar o número máximo de minutos para o algoritmo executar. Especifique um número maior que 0.
- **Máximo de épocas de treinamento.** O número máximo de épocas (passagens de dados) permitido. Se o número máximo de épocas for excedido, o treinamento será parado. Especifique um número inteiro maior que 0.
- **Mudança mínima relativa no erro de treinamento.** O treinamento será parado se a mudança relativa no erro de treinamento comparada com o passo anterior for menor que o valor de critério. Especifique um número maior que 0. Para treinamento on-line e de minilote, este critério é ignorado se apenas os dados de testes forem usados para computar o erro.
- **Mudança mínima relativa na razão de erro de treinamento.** O treinamento será parado se a razão do erro de treinamento para o erro do modelo nulo for menor que o valor de critério. O modelo nulo prevê o valor médio para todas as variáveis dependentes. Especifique um número maior que 0. Para treinamento on-line e de minilote, este critério é ignorado se apenas os dados de testes forem usados para computar o erro.

**Máximo de casos para armazenar na memória.** Isso controla as seguintes configurações nos algoritmos perceptron multiestrato. Especifique um número inteiro maior que 1.

- Na seleção automática de arquitetura, o tamanho da amostra usada para determinar a arquitetura de rede é  $\min(1000, \text{memsize})$ , em que *memsize* é o número máximo de casos para armazenar na memória.
- No treinamento de minilote com cálculo automático do número de minilotes, o número de minilotes é  $\min(\max(M/10, 2), \text{memsize})$ , em que *M* é o número de casos na amostra de treinamento.

## Função de base radial

---

O procedimento Função de Base Radial (RBF) produz um modelo preditivo para uma ou mais variáveis dependentes (destino) com base nos valores de variáveis preditoras.

**Exemplo,** Um provedor de telecomunicações segmentou sua base de clientes por padrões de uso de serviço, categorizando os clientes em quatro grupos. Uma rede RBF que usa dados demográficos para prever a associação ao grupo permite que a empresa customize ofertas para possíveis clientes individuais.

Considerações de dados












**Variáveis dependentes.** As variáveis dependentes podem ser:

- *Nominal.* Uma variável pode ser tratada como nominal quando seus valores representam categorias sem uma classificação intrínseca (por exemplo, o departamento da empresa na qual um funcionário trabalha) Exemplos de variáveis nominais incluem região, código de endereçamento postal e filiação religiosa.
- *Ordinal.* Uma variável pode ser tratada como ordinal quando seus valores representam categorias com alguma classificação intrínseca (por exemplo, níveis de satisfação de serviço de altamente insatisfeito para altamente satisfeito. Exemplos de variáveis ordinais incluem escores de atitude que representam o grau de satisfação ou de confiança e os escores de classificação de preferência.
- *scale.* Uma variável pode ser tratada como escala (contínua) quando seus valores representam categorias ordenadas com uma métrica significativa, para que as comparações de distância entre

valores sejam apropriadas. Exemplos de variáveis de escala incluem idade em anos e rendimento em milhares de dólares.

O procedimento supõe que o nível de medição apropriado foi designado a todas as variáveis dependentes; embora seja possível mudar temporariamente o nível de medição para uma variável, clicando com o botão direito na variável na lista de variáveis de origem e selecionando um nível de medição no menu pop-up.

Um ícone ao lado de cada variável na lista de variáveis identifica o nível de medição e o tipo de dados:

Tabela 2. Ícones do nível de medição				
	Numérico	Sequência	Data	Hora
Escala (Contínua)		n/d		
Ordinal				
Nominal				

**Variáveis predictoras.** Os preditores podem ser especificados como fatores (categóricos) ou covariáveis (escala).

**Codificação de variável categórica.** O procedimento recodifica temporariamente preditores categóricos e variáveis dependentes utilizando a codificação um-de-c durante o procedimento. Se houver  $c$  categorias de uma variável, a variável será armazenada como vetores  $c$ , com a primeira categoria denotada (1, 0, ..., 0), a próxima categoria (0, 1, 0, ..., 0), ..., e a categoria final (0, 0, ..., 0, 1).

Esse esquema de codificação aumenta o número de ponderações sinápticas e pode resultar em um treinamento mais lento, mas os métodos de codificação mais "compactos" geralmente resultam em redes neurais com um ajuste mais simples. Se o seu treinamento em rede está prosseguindo muito lentamente, tente reduzir o número de categorias em seus preditores categóricos ao combinar categorias similares ou eliminando casos que tenham categorias extremamente raras.

Toda a codificação um-de-c é baseada nos dados de treinamento, mesmo que uma amostra de teste ou de validação esteja definida (consulte "Partições" na página 13). Portanto, se as amostras de teste ou de validação contiverem casos com categorias do preditor que não estão presentes nos dados de treinamento, esses casos não serão usados pelo procedimento ou na escoragem. Se as amostras de teste ou de validação contiverem casos com categorias de variável dependente que não estão presentes nos dados de treinamento, esses casos não serão usados pelo procedimento, mas poderão ser escorados.

**Escalando novamente.** As variáveis dependentes de escala e covariáveis são escaladas novamente por padrão para melhorar o treinamento de rede. Toda a nova escala é executada com base nos dados de treinamento, mesmo que uma amostra de teste ou de validação esteja definida (consulte "Partições" na página 13). Ou seja, dependendo do tipo de nova escala, a média, o desvio padrão, o valor mínimo ou o valor máximo de uma covariável ou variável dependente são calculados usando apenas os dados de treinamento. Se você especificar uma variável para definir partições, é importante que essas covariáveis ou variáveis dependentes tenham distribuições similares nas amostras de treinamento, teste ou validação.

**Ponderações de frequência.** As ponderações de frequência são ignoradas por este procedimento.

**Replicando resultados.** Se desejar replicar seus resultados de forma exata, use o mesmo valor de inicialização para o gerador de números aleatórios e a mesma ordem de dados, além de usar as mesmas configurações do procedimento. A seguir estão mais detalhes sobre esse problema:

- **Geração de número aleatório.** O procedimento usa a geração de número aleatório durante a designação aleatória de partições. Para reproduzir os mesmos resultados aleatórios no futuro, use



o mesmo valor de inicialização para o gerador de números aleatórios antes de cada execução do procedimento Função de base radial.

- **Ordem de casos.** Os resultados também são dependentes da ordem de dados, porque o algoritmo de cluster de dois passos é usado para determinar as funções de base radial.

Para minimizar os efeitos da ordem, ordene aleatoriamente os casos. Para verificar a estabilidade de uma determinada solução, talvez você queira obter várias soluções diferentes com casos ordenados em diferentes ordens aleatórios. Em situações com tamanhos de arquivos extremamente grandes, podem ser feitas várias execuções com uma amostra de casos ordenados em diferentes ordens aleatórios.

Criando uma rede de função de base radial

Nos menus, escolha:

**Analisar > Redes neurais > Função de base radial...**

1. Selecione pelo menos uma variável dependente.
2. Selecione pelo menos um fator ou covariável.

Opcionalmente, na guia Variáveis, é possível mudar o método para escalar novamente as covariáveis. As opções são:

- **Padronizado.** Subtraia a média e divida pelo desvio padrão,  $(x - \text{mean})/s$ .
- **Normalizado.** Subtraia o mínimo e divida pelo intervalo,  $(x - \text{min})/(\text{max} - \text{min})$ . Os valores normalizados estão entre 0 e 1.
- **Normalizado ajustado.** Versão ajustada da subtração do mínimo e da divisão pelo intervalo,  $[2 * (x - \text{min}) / (\text{max} - \text{min})] - 1$ . Os valores normalizados ajustados estão entre -1 e 1.
- **Nenhum.** Nenhuma nova escala de covariáveis.

Campos com nível de medição desconhecido

O alerta de Nível de Medição é exibido quando o nível de medição para uma ou mais variáveis (campos) no conjunto de dados é desconhecido. Como o nível de medição afeta o cálculo de resultados para este procedimento, todas as variáveis devem ter um nível de medição definido.

**Dados de varredura.** Lê os dados no conjunto de dados ativo e designa o nível de medição padrão para quaisquer campos com um nível de medição desconhecido atualmente. Se o conjunto de dados for grande, isso poderá demorar algum tempo.

**Designar Manualmente.** Abre um diálogo que lista todos os campos com um nível de medição desconhecido. É possível utilizar este diálogo para designar o nível de medição para esses campos. Também é possível designar o nível de medição na Visualização de Variável do Editor de Dados.

Como o nível de medição é importante para este procedimento, não é possível acessar o diálogo para executar este procedimento até que todos os campos possuam um nível de medição definido.

## Partições

**Particionar conjunto de dados.** Esse grupo especifica o método de particionamento do conjunto de dados ativo em amostras de treinamento, de teste e de validação. A **amostra de treinamento** abrange os registros de dados usados para treinar a rede neural; alguma porcentagem dos casos no conjunto de dados deve ser designada à amostra de treinamento para obter um modelo. A **amostra de teste** é um conjunto independente de registros de dados usados para rastrear erros durante o treinamento para evitar o excesso de treinamento. É altamente recomendado que você crie uma amostra de teste, dessa forma, o treinamento de rede geralmente será mais eficiente se a amostra de teste for menor que a amostra de treinamento. A **amostra de validação** é outro conjunto independente de registros de dados usados para avaliar a rede neural final; o erro para a amostra de validação fornece uma estimativa "honesta" da capacidade preditiva do modelo, porque os casos de validação não foram usados para construir o modelo.

- **Designar casos aleatoriamente com base no número relativo de casos.** Especifique o número relativo (razão) de casos designados aleatoriamente a cada amostra (treinamento, teste e validação). A coluna

% relata a porcentagem de casos que serão designados a cada amostra, com base nos números relativos especificados.

Por exemplo, especificar 7, 3, 0 como os números relativos para amostras de treinamento, teste e validação corresponde a 70%, 30% e 0%. Especificar 2, 1, 1 como os números relativos corresponde a 50%, 25% e 25%; 1, 1, 1 corresponde a dividir o conjunto de dados em terços iguais entre treinamento, teste e validação.

- **Usar variável de particionamento para designar casos.** Especifique uma variável numérica que designa cada caso no conjunto de dados ativo à amostra de treinamento, teste ou validação. Os casos com um valor positivo na variável são designados à amostra de treinamento, os casos com um valor 0 à amostra de teste, e os casos com um valor negativo à amostra de validação. Casos com um valor omissos do sistema são excluídos da análise. Quaisquer valores omissos do usuário para a variável de partição são sempre tratados como válidos.

## Arquitetura

A guia Arquitetura é usada para especificar a estrutura da rede. O procedimento cria uma rede neural com uma estrato de "função de base radial" oculta; em geral, não será necessário mudar essas configurações.

**Número de unidades na estrato oculta.** Existem três maneiras de escolher o número de unidades ocultas.

1. **Localizar o melhor número de unidades em um intervalo calculado automaticamente.** O procedimento calcula automaticamente os valores mínimo e máximo do intervalo e localiza o melhor número de unidades ocultas no intervalo.  
  
Se uma amostra de teste estiver definida, o procedimento usará o critério de dados de teste: O melhor número de unidades ocultas é o que gera o menor erro nos dados de teste. Se uma amostra de teste não estiver definida, o procedimento usará o critério de informações bayesiano (BIC): O melhor número de unidades ocultas é o que gera o menor BIC com base nos dados de treinamento.
2. **Localizar o melhor número de unidades em um intervalo especificado.** É possível fornecer seu próprio intervalo e o procedimento localizará o "melhor" número de unidades ocultas nesse intervalo. Como antes, o melhor número de unidades ocultas do intervalo é determinado usando o critério de dados de teste ou o BIC.
3. **Usar um número especificado de unidades.** É possível substituir o uso de um intervalo e especificar um número específico de unidades diretamente.

**Função de ativação para estrato oculta.** A função de ativação para a estrato oculta é a função de base radial, que "liga" as unidades em uma estrato aos valores de unidades na estrato seguinte. Para a estrato de saída, a função de ativação é a função de identidade; portanto, as unidades de saída são apenas somas ponderadas das unidades ocultas.

- **Função de base radial normalizada.** Usa a função de ativação softmax para que as ativações de todas as unidades ocultas sejam normalizadas para serem somadas a 1.
- **Função de base radial ordinária.** Usa a função de ativação exponencial para que a ativação da unidade oculta seja um "impulso" Gaussiano como uma função das entradas.

**Sobrepor entre unidades ocultas.** O fator de sobreposição é um multiplicador aplicado à largura das funções de base radial. O valor calculado automaticamente do fator de sobreposição é  $1+0.1d$ , em que  $d$  é o número de unidades de entrada (a soma do número de categorias em todos os fatores e o número de covariáveis).

## Saída

**Estrutura de rede.** Exibe informações de sumarização sobre a rede neural.

- **DESCRIPTION.** Exibe informações sobre a rede neural, incluindo as variáveis dependentes, o número de unidades de entrada e de saída, o número de camadas e unidades ocultas e as funções de ativação.

- **Diagrama.** Exibe o diagrama de rede como um gráfico não editável. Observe que, à medida que o número de covariáveis e os níveis de fatores aumentam, o diagrama se torna mais difícil de ser interpretado.
- **Ponderações sinápticas.** Exibe as estimativas de coeficiente que mostram o relacionamento entre as unidades em uma determinada estrato com as unidades na estrato seguinte. As ponderações sinápticas são baseadas na amostra de treinamento, mesmo que o conjunto de dados ativo seja particionado em dados de treinamento, de teste e de validação. Observe que o número de ponderações sinápticas pode ficar muito grande e essas ponderações geralmente não são usadas para interpretar resultados de rede.

**Desempenho da rede.** Exibe os resultados usados para determinar se o modelo é "válido." *Nota:* Os gráficos nesse grupo são baseados nas amostras combinadas de treinamento e teste ou somente na amostra de treinamento, se não houver nenhuma amostra de teste.

- **Sumarização do modelo.** Exibe uma sumarização dos resultados da rede neural por partição e geral, incluindo o erro, o erro relativo ou a porcentagem de predições incorretas e o tempo de treinamento.  
O erro é a soma dos quadrados dos erros. Além disso, são exibidos os erros relativos ou porcentagens de predições incorretas, dependendo dos níveis de medição da variável dependente. Se alguma variável dependente tiver nível de medição de escala, a média geral de erros relativos (relativos ao modelo de média) será exibida. Se todas as variáveis dependentes forem categóricas, a porcentagem média de predições incorretas será exibida. Os erros relativos ou porcentagens de predições incorretas também são exibidos para variáveis dependentes individuais.
- **Resultados de classificação.** Exibe uma tabela de classificação para cada variável dependente categórica. Cada tabela fornece o número de casos classificados corretamente e incorretamente para cada categoria de variável dependente. A porcentagem do total de casos que foram classificados corretamente também é relatada.
- **Curva ROC.** Exibe uma curva ROC (Receiver Operating Characteristic) para cada variável dependente categórica. Também exibe uma tabela que fornece a área em cada curva. Para uma determinada variável dependente, o gráfico ROC exibe uma curva para cada categoria. Se a variável dependente tiver duas categorias, cada curva tratará a categoria com problema como o estado positivo versus a outra categoria. Se a variável dependente tiver mais de duas categorias, cada curva tratará a categoria com problema como o estado positivo versus a agregação de todas as outras categorias.
- **Gráfico de ganhos acumulativos.** Exibe um gráfico de ganhos acumulativos para cada variável dependente categórica. A exibição de uma curva para cada categoria de variável dependente é a mesma que para curvas ROC.
- **Gráfico de ganho.** Exibe um gráfico de ganho para cada variável dependente categórica. A exibição de uma curva para cada categoria de variável dependente é a mesma que para curvas ROC.
- **Gráfico predito por observado.** Exibe um gráfico de valor predito por observado para cada variável dependente. Para variáveis dependentes categóricas, os boxplots em cluster de pseudoprobabilidades preditas são exibidos em cada categoria de resposta, com a categoria de resposta observada como a variável de cluster. Para variáveis dependentes de escala, é exibido um gráfico de dispersão.
- **Gráfico de resíduo por predito.** Exibe um gráfico de resíduo por valor predito para cada variável dependente de escala. Não deve haver nenhum padrão visível entre valores de resíduos e preditos. Esse gráfico é produzido apenas para variáveis dependentes de escala.

**Sumarização do processamento de caso.** Exibe a tabela de sumarização de processamento de caso, que sumariza o número de casos incluídos e excluídos na análise, no total e por amostras de treinamento, de teste e de validação.

**Análise de importância da variável independente.** Executa uma análise de sensibilidade, que calcula a importância de cada preditor na determinação da rede neural. A análise é baseada nas amostras combinadas de treinamento e teste, ou apenas na amostra de treinamento, se não houver nenhuma amostra de teste. Isso cria uma tabela e um gráfico que exibem a importância e a importância normalizada para cada preditor. Observe que a análise de sensibilidade é dispendiosa e demorada do ponto de vista computacional, se houver uma grande quantidade de preditores ou casos.

## Salvar

A guia Salvar é usada para salvar previsões como variáveis no conjunto de dados.

- **Salvar valor predito ou categoria para cada variável dependente.** Isso salva o valor predito para variáveis dependentes de escala e a categoria predita para variáveis dependentes categóricas.
- **Salvar pseudoprobabilidade predita para cada variável dependente.** Isso salva as pseudoprobabilidades preditas para variáveis dependentes categóricas. Uma variável separada é salva para cada uma das primeiras  $n$  categorias, em que  $n$  é especificado na coluna *Categorias a serem salvas*.

**Nomes de Variáveis Salvas.** A geração de nome automática assegura que você mantenha todo o seu trabalho. Os nomes customizados permitem descartar ou substituir resultados de execuções anteriores sem primeiro excluir as variáveis salvas no Editor de dados.

### Probabilidades e pseudoprobabilidades

As pseudoprobabilidades preditas não podem ser interpretadas como probabilidades porque o procedimento Função de base radial usa a soma dos quadrados dos erros e a função de ativação de identidade para a estrato de saída. O procedimento salva essas pseudoprobabilidades preditas mesmo que algumas sejam menores que 0 ou maiores que 1 ou a soma para uma determinada variável dependente não seja 1.

O ROC, ganhos acumulativos e gráficos de ganho (consulte “Saída ” na [página 14](#) ) são criados com base em pseudoprobabilidades. No caso de qualquer uma das pseudoprobabilidades ser inferior a 0 ou superior a 1 ou a soma de uma determinada variável não for 1, eles são primeiros redimensionados a ficar entre 0 e 1 e a somar 1. As pseudoprobabilidades são redimensionadas dividindo-se por sua soma. Por exemplo, se um caso tiver pseudoprobabilidades preditas de 0,50, 0,60 e 0,40 para uma variável dependente de três categorias, cada pseudoprobabilidade será dividida pela soma 1,50 para obter 0,33, 0,40 e 0,27.

Se qualquer uma das pseudoprobabilidades for negativa, o valor absoluto da mais baixa será incluído em todas as pseudoprobabilidades antes da nova escala acima. Por exemplo, se as pseudoprobabilidades forem -0,30, 0,50 e 1,30, primeiro inclua 0,30 em cada valor para obter 0,00, 0,80 e 1,60. Em seguida, divida cada novo valor pela soma 2,40 para obter 0,00, 0,33 e 0,67.

## Exportar

A guia Exportar é usada para salvar as estimativas de ponderação sináptica para cada variável dependente em um arquivo XML (PMML). É possível usar esse arquivo de modelo para aplicar as informações de modelo a outros arquivos de dados para propósitos de escoragem. Essa opção não estará disponível se arquivos de divisão tiverem sido definidos.

## Opções

**Valores omissos de usuário.** Os fatores devem ter valores válidos para um caso a serem incluídos na análise. Estes controles permitem decidir se valores omissos de usuário são tratados como válidos entre fatores e variáveis dependentes categóricas.

## Avisos

---

Essas informações foram desenvolvidas para produtos e serviços oferecidos nos Estados Unidos. Esse material pode estar disponível a partir da IBM em outros idiomas. No entanto, pode ser necessário possuir uma cópia do produto ou da versão do produto nesse idioma para acessá-lo.

É possível que a IBM não ofereça os produtos, serviços ou recursos discutidos nesta publicação em outros países. Consulte um representante IBM local para obter informações sobre produtos e serviços disponíveis atualmente em sua área. Qualquer referência a produtos, programas ou serviços IBM não significa que apenas produtos, programas ou serviços IBM possam ser utilizados. Qualquer outro produto, programa ou serviço, funcionalmente equivalente, poderá ser utilizado em substituição daqueles, desde que não infrinja nenhum direito de propriedade intelectual da IBM. Entretanto, a avaliação e verificação da operação de qualquer produto, programa ou serviço não IBM são de responsabilidade do Cliente.

A IBM pode ter patentes ou solicitações de patentes pendentes relativas a assuntos tratados nesta publicação. O fornecimento desta publicação não lhe garante direito algum sobre tais patentes. Pedidos de licença devem ser enviados, por escrito, para:

*Gerência de Relações Comerciais e Industriais da IBM Brasil*  
*IBM Brasil Ltda*  
*Botafogo*  
*Rio de Janeiro, RJ*  
*CEP 22290-240*

Para pedidos de licença relacionados a informações de DBCS (Conjunto de Caracteres de Byte Duplo), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie pedidos de licença, por escrito, para:

*Intellectual Property Licensing*  
*Legal and Intellectual Property Law*  
*IBM Japan Ltd.*  
*19-21, Nihonbashi-Hakozakicho, Chuo-ku*  
*Tokyo 103-8510, Japan*

A INTERNATIONAL BUSINESS MACHINES CORPORATION FORNECE ESTA PUBLICAÇÃO "NO ESTADO EM QUE SE ENCONTRA", SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS A ELAS NÃO SE LIMITANDO, AS GARANTIAS IMPLÍCITAS DE NÃO INFRAÇÃO, COMERCIALIZAÇÃO OU ADEQUAÇÃO A UM DETERMINADO PROPÓSITO. Alguns países não permitem a exclusão de garantias expressas ou implícitas em certas transações; portanto, essa disposição pode não se aplicar ao Cliente.

Essas informações podem conter imprecisões técnicas ou erros tipográficos. São feitas alterações periódicas nas informações aqui contidas; tais alterações serão incorporadas em futuras edições desta publicação. A IBM pode, a qualquer momento, aperfeiçoar e/ou alterar os produtos e/ou programas descritos nesta publicação, sem aviso prévio.

Todas as referências nestas informações para websites não IBM são fornecidas apenas para conveniência e não de forma a servir como um endosso a esses websites. Os materiais contidos nesses websites não fazem parte dos materiais desse produto IBM e a utilização desses websites é de inteira responsabilidade do Cliente.

A IBM por usar ou distribuir as informações fornecidas da forma que julgar apropriada sem incorrer em qualquer obrigação para com o Cliente.

Licenciados deste programa que desejam obter informações sobre este assunto com objetivo de permitir: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) a utilização mútua das informações trocadas, devem entrar em contato com:

*Gerência de Relações Comerciais e Industriais da IBM Brasil*  
*IBM Brasil Ltda*

*Botafogo  
Rio de Janeiro, RJ  
CEP 22290-240*

Tais informações podem estar disponíveis, sujeitas a termos e condições apropriadas, incluindo em alguns casos o pagamento de uma taxa.

O programa licenciado descrito nesta publicação e todo o material licenciado disponível são fornecidos pela IBM sob os termos do Contrato com o Cliente IBM, do Contrato Internacional de Licença do Programa IBM ou de qualquer outro contrato equivalente.

Os exemplos de dados de desempenho e do Cliente citados são apresentados apenas para propósitos ilustrativos. Resultados de desempenho reais podem variar dependendo das configurações específicas e das condições operacionais.

Informações relativas a produtos não IBM foram obtidas junto aos fornecedores dos respectivos produtos, de seus anúncios publicados ou de outras fontes disponíveis publicamente. A IBM não testou esses produtos e não pode confirmar a precisão de seu desempenho ou de sua compatibilidade, nem quaisquer reclamações relacionadas aos produtos não IBM. Perguntas sobre os recursos de produtos não IBM devem ser endereçadas aos fornecedores desses produtos.

Instruções relativas à direção futura ou intento da IBM estão sujeitas a mudança ou retirada sem aviso e representam metas e objetivos apenas.

Estas informações contêm exemplos de dados e relatórios utilizados nas operações diárias de negócios. Para ilustrá-los da forma mais completa possível, os exemplos incluem nomes de indivíduos, empresas, marcas e produtos. Todos esses nomes são fictícios e qualquer semelhança com pessoas ou empresas reais é mera coincidência.

#### LICENÇA DE COPYRIGHT:

Estas informações contêm programas de aplicativos de amostra na linguagem fonte, ilustrando as técnicas de programação em diversas plataformas operacionais. O Cliente pode copiar, modificar e distribuir estes programas de amostra sem a necessidade de pagar à IBM, com objetivos de desenvolvimento, utilização, marketing ou distribuição de programas aplicativos em conformidade com a interface de programação de aplicativo para a plataforma operacional para a qual os programas de amostra são criados. Esses exemplos não foram testados completamente em todas as condições. Portanto, a IBM não pode garantir ou implicar a confiabilidade, manutenção ou função destes programas. Os programas de amostra são fornecidos "NO ESTADO EM QUE SE ENCONTRAM", sem garantia de qualquer tipo. A IBM não será responsabilizada por quaisquer danos decorrentes do uso dos programas de amostra.

Cada cópia ou parte destes programas de amostra ou qualquer trabalho derivado deve incluir um aviso de copyright com os dizeres:

© Copyright IBM Corp. 2021. Partes deste código são derivadas de Programas de Amostra da IBM Corp. Programas de amostra.

© Copyright IBM Corp. 1989 - 2021. Todos os direitos reservados.

## Marcas comerciais

---

IBM, o logotipo IBM e [ibm.com](http://ibm.com) são marcas comerciais ou marcas registradas da International Business Machines Corp., registradas em várias jurisdições no mundo todo. Outros nomes de produto e de serviço podem ser marcas registradas da IBM ou de outras empresas. A lista atual de marcas comerciais da IBM está disponível na web em "Copyright and trademark information" em [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, o logotipo Adobe, PostScript e o logotipo PostScript são marcas registradas ou marcas comerciais da Adobe Systems Incorporated nos Estados Unidos e/ou em outros países.

Intel, o logotipo Intel, Intel Inside, o logotipo Intel Inside, Intel Centrino, o logotipo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium e Pentium são marcas comerciais ou marcas registradas da Intel Corporation ou de suas subsidiárias nos Estados Unidos e em outros países.

Linux é uma marca registrada da Linus Torvalds nos Estados Unidos, e/ou em outros países.

Microsoft, Windows, Windows NT e o logotipo Windows são marcas comerciais da Microsoft Corporation nos Estados Unidos e/ou em outros países.

UNIX é uma marca registrada da The Open Group nos Estados Unidos e em outros países.

Java e todas as marcas comerciais e logotipos baseados em Java são marcas comerciais ou marcas registradas da Oracle e/ou suas afiliadas.





---

# Índice remissivo

## A

amostra de teste  
na Função de Base Radial [13](#)  
no Perceptron multiestrato [5](#)  
amostra de treinamento  
na Função de Base Radial [13](#)  
no Perceptron multiestrato [5](#)  
amostra de validação  
na Função de Base Radial [13](#)  
no Perceptron multiestrato [5](#)  
arquitetura  
redes neurais [2](#)  
arquitetura de rede  
na Função de Base Radial [14](#)  
no Perceptron multiestrato [6](#)

## C

Curva ROC  
na Função de Base Radial [14](#)  
no Perceptron multiestrato [8](#)

## D

diagrama de rede  
na Função de Base Radial [14](#)  
no Perceptron multiestrato [8](#)

## E

estrato de saída  
na Função de Base Radial [14](#)  
no Perceptron multiestrato [6](#)  
estrato oculta  
na Função de Base Radial [14](#)  
no Perceptron multiestrato [6](#)

## F

função de ativação  
na Função de Base Radial [14](#)  
no Perceptron multiestrato [6](#)  
Função de base radial  
arquitetura de rede [14](#)  
exportação de modelo [16](#)  
opções [16](#)  
partições [13](#)  
saída [14](#)  
salvar variáveis no conjunto de dados ativo [16](#)

## G

gráfico de ganho  
na Função de Base Radial [14](#)  
no Perceptron multiestrato [8](#)

gráfico de ganhos  
na Função de Base Radial [14](#)  
no Perceptron multiestrato [8](#)

## P

Perceptron Multicamada  
arquitetura de rede [6](#)  
exportação de modelo [10](#)  
opções [10](#)  
partições [5](#)  
saída [8](#)  
salvar variáveis no conjunto de dados ativo [10](#)  
treinamento [7](#)

## R

redes neurais  
arquitetura [2](#)  
regras de parada  
no Perceptron multiestrato [10](#)

## T

treinamento de rede  
no Perceptron multiestrato [7](#)  
treinamento em lote  
no Perceptron multiestrato [7](#)  
treinamento em minilote  
no Perceptron multiestrato [7](#)  
treinamento on-line  
no Perceptron multiestrato [7](#)

## V

valores omissos  
no Perceptron multiestrato [10](#)





