

3

Modelos Comparativos: Teoria e Metodologia

Para avaliar o desempenho do modelo STAR-Tree, foram estimados os modelos Naive, ARMAX e Redes Neurais. O ajuste dos modelos ARMAX e das redes neurais foi feito a partir de dados *in-sample*. Para cada metodologia é apresentada a teoria básica sobre o assunto abordado e a metodologia de estimação dos parâmetros de cada modelo.

A performance dos modelos foi avaliada por medidas estatísticas e financeiras, que são explicadas detalhadamente no capítulo (4).

3.1

Naive

O método Naive se baseia no conceito de que a melhor previsão para o dia seguinte será o valor observado no dia atual. Ou seja, se um ativo estiver em uma tendência de alta, a previsão é que ele continuará em alta e vice-versa. A previsão Naive é definida por

$$\hat{R}_{t+1} = R_t \quad (3-1)$$

onde \hat{R}_{t+1} é a previsão feita em t para o dia $t + 1$ e R_t é o valor do retorno observado no dia t .

3.2

ARMAX

Os modelos ARMAX (Autoregressive Moving Average with Exogenous inputs) são uma extensão dos modelos ARMA, pois consideram a possibilidade de inclusão de variáveis exógenas como regressores.

Os modelos ARMA, também conhecidos como Box-Jenkins, são usados para modelar séries temporais univariadas, nas quais os dados utilizados

são estacionários¹. São formados por uma parte autoregressiva AR(p), definida como uma regressão linear do valor atual da série contra p valores passados ponderados; e uma parte médias móveis MA(q), uma soma ponderada da observação atual e de q observações do passado de um processo ruído branco. Para maior aprofundamento no assunto, recomenda-se recorrer a Box e Jenkins [1], ou Hamilton [7].

Os modelos ARMAX(p,q,Nx) apresentam a forma:

$$R_t = c + \sum_{i=1}^p \phi_i R_{t-i} + \sum_{j=1}^q \theta_j \epsilon_{t-j} + \epsilon_t + \sum_{k=1}^{Nx} \beta_k X_{t,k} \quad (3-2)$$

onde R_t é o processo modelado; c é uma constante; ϕ_1, \dots, ϕ_p são os coeficientes dos termos autoregressivos; $R_{t-1}, R_{t-2}, \dots, R_{t-p}$ são os termos autoregressivos; ϵ_t é a componente aleatória do modelo, com $\epsilon_t \sim N(0, \sigma^2)$; $\theta_1, \theta_2, \dots, \theta_q$ são os coeficientes da componente aleatória; $\epsilon_{t-1}, \epsilon_{t-2}, \dots, \epsilon_{t-q}$ são as componentes aleatórias defasados; \mathbf{X} é uma matriz de variáveis exógenas na qual cada coluna é uma série temporal e $X(t, k)$ é o elemento alocado na t -ésima linha e k -ésima coluna; e β_1, \dots, β_p são os coeficientes dos termos de variáveis exógenas.

Especificação dos Modelos ARMAX

Seguindo a abordagem adotada por Dunnis e Williams [4], e estendendo-a para modelos ARMAX, a especificação foi feita com a ordem $p = 10$ e $q = 10$ de forma a levar em conta as duas últimas semanas de negócios. Testes de correlação cruzada indicaram que apenas a primeira defasagem das variáveis derivadas de índices externos apresentava correlação significativa com o retorno dos ativos. A descrição de cada variável é apresentada no capítulo (4).

Inicialmente, a estimação foi feita incluindo todas as variáveis como variáveis de entrada. Porém, a estatística t dos coeficientes estimados não se mostrou significativa para diversas variáveis. Os coeficientes que apresentaram as maiores estatísticas t foram os primeiros a serem suprimidos da equação. Este processo de eliminação de coeficientes foi adotado até o momento em que todos os valores estimados para os parâmetros restantes mostraram-se significantes.

¹Caso a série seja não estacionária, deve-se fazer uma transformação nos dados. Geralmente aplica-se a primeira diferença na série, podendo ser necessário mais de uma diferenciação.

Diversos modelos foram gerados para cada ativo. O critério de escolha de um modelo em detrimento dos demais foi o AIC (Akaike information Criteria)(3-3), que é uma medida que indica o nível de ajuste do modelo gerado aos dados levando em consideração a quantidade de parâmetros utilizados. O modelo que apresentar o menor AIC deve ser o escolhido.

$$AIC = -2\frac{l}{T} + 2\frac{k}{T} \quad (3-3)$$

onde l é o log da verossimilhança, T é o número de dados e k é o número de parâmetros.

Em algumas situações, quando a diferença entre os AIC's dos modelos concorrentes era menor que 0.01, foi escolhido o que apresentava a menor quantidade de parâmetros. O modelo final ainda foi submetido ao teste LM de correlação dos resíduos e ao teste F, que verifica a significância da regressão.

Para o cálculo das previsões no período de estimação (*in-sample*), é necessário estimar os valores iniciais das variáveis usadas nas equações, os quais foram obtidos através da geração de números aleatórios de uma distribuição Normal, sendo a média e a variância calculadas a partir das respectivas séries de observações. Os primeiros 252 termos da série de previsões gerada pelo modelo ARMAX foram desprezados para anular qualquer efeito de inicialização.

3.3 Redes Neurais

As redes neurais são modelos não lineares utilizados principalmente para reconhecimento e classificação de padrões, previsão de séries temporais e aproximação de funções.

As redes neurais são formadas por neurônios e organizadas em camadas. Geralmente é utilizado um nível de entrada, onde os padrões são apresentados à rede, uma camada escondida ou oculta e uma camada de saída, na qual a informação que se deseja é apresentada. Pode haver a necessidade do emprego de mais de uma camada escondida, mas não é usual. Os neurônios ocultos capacitam a rede a aprender tarefas complexas, extraíndo, progressivamente, as características mais significativas dos padrões de entrada.

Em cada neurônio está presente a função de ativação. Esta função define a saída do neurônio de acordo com a sua entrada. Os principais tipos

utilizados são: sigmóide e tangente hiperbólica, linear e radial. A função linear é utilizada principalmente na camada de saída, onde não é desejável o efeito de saturação presente nas funções sigmóides e tangentes hiperbólicas.

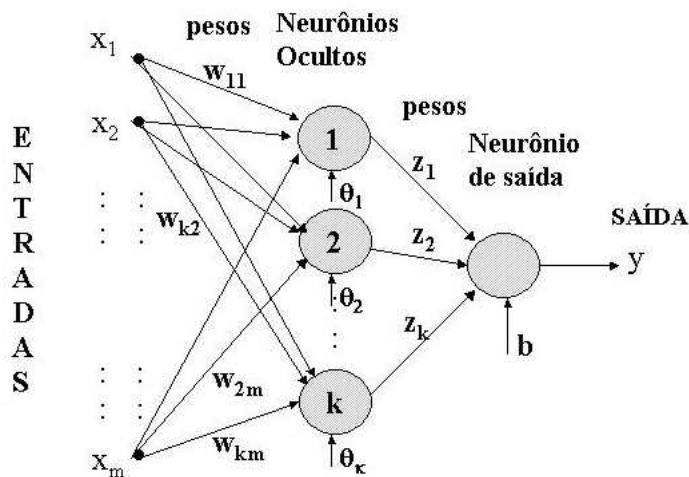


Figura 3.1: Representação de uma Rede Neural

O modelo matemático do neurônio de uma rede neural está representado em

$$t_k = \varphi \left(\sum_{j=1}^m w_{kj} x_j + \theta_k \right) \quad (3-4)$$

onde t_k é o valor do sinal na saída do neurônio k ; w_{kj} são os pesos; x_1, x_2, \dots, x_m são as variáveis de entrada; θ_k é o bias; e $\varphi(\cdot)$ é a função de ativação.

Em notação matricial, (3-4) assume a forma

$$t_k = \varphi(\mathbf{w}_k' \mathbf{x} + \theta_k) \quad (3-5)$$

onde $\mathbf{x} = (x_1, x_2, \dots, x_m)'$ e $\mathbf{w}_k = (w_{k1}, w_{k2}, \dots, w_{km})'$.

Aplicando a equação (3-5) em todos os neurônios, obtém-se a equação para a saída da rede, apresentada a seguir.

$$y = \varphi_2(\mathbf{z}' \varphi_1(\mathbf{W}\mathbf{x} + \boldsymbol{\theta}) + b) \quad (3-6)$$

onde $\varphi_2(\cdot)$ e b são, respectivamente, a função de ativação e o bias da camada de saída; $\mathbf{z} = (z_1, z_2, \dots, z_k)'$ é o vetor de pesos entre a camada oculta e a saída; \mathbf{W} é uma matriz $k \times m$ de pesos onde k é o número de neurônios ocultos e m o número de variáveis de entrada.

O conhecimento nas redes neurais é obtido através de um processo de treinamento, que funciona a partir de um algoritmo capaz de ajustar os

pesos sinápticos de modo a atingir o objetivo desejado, e é produzido com a combinação de não linearidades fazendo uso destes pesos. Os pesos recebem um valor de acordo com o efeito que a entrada exerce sobre o processador.

Especificação das Redes Neurais

O conjunto de dados foi separado em 3 grupos: treinamento, validação e teste. O treinamento tem a finalidade de ajustar o modelo aos dados. O grupo de validação monitora o erro medido na rede e, no primeiro sinal de aumento desta medida, o treinamento é finalizado. Este procedimento é importante para evitar que a rede seja super-treinada e aprenda ruído ao invés de informação útil. Em seguida, com a mesma quantidade de épocas (conjunto de padrões de treinamento), a rede é treinada novamente e os pesos são ajustados. Na fase de teste, o modelo é avaliado com padrões que ainda não foram apresentados à rede. O treinamento recebeu 2/3 dos dados e a validação e o teste 1/6 cada.

Em seguida, definiram-se quais seriam as variáveis que fariam parte da rede neural. Existem 42 variáveis disponíveis, entre elas defasagens do retorno, defasagens da volatilidade, defasagens da variação de volume entre dois dias consecutivos, médias móveis e variáveis exógenas, como o retorno dos índices BOVESPA, S&P 500, da cotação do dólar e outras. O capítulo (4) apresenta uma explicação detalhada do significado de cada variável.

A função de ativação utilizada foi a tangente hiperbólica, cuja equação está definida em (3-7) e seu gráfico em (3.2). Com o intuito de melhor aproveitar a sua não linearidade, aplicou-se uma função do *MatLab* chamada *premnmx*, que faz com que os dados ocupem o intervalo $[-1, 1]$.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad (3-7)$$

Apesar de ser uma medida linear, a correlação cruzada entre as variáveis foi usada como ponto inicial para a escolha das que farão parte do modelo. A partir daí, as variáveis foram escolhidas através da inclusão e retirada das mesmas. O critério de manutenção da variável como entrada da rede foi o RMSE (Root Mean Square Error)(3-8), que fornece uma medida para a dispersão dos erros da estimação no treinamento. Quanto menor o RMSE, melhor é o ajuste do modelo aos dados.

$$RMSE = \sqrt{\frac{1}{T} \sum_{t=1}^T (\tilde{y}_t - y_t)^2} \quad (3-8)$$

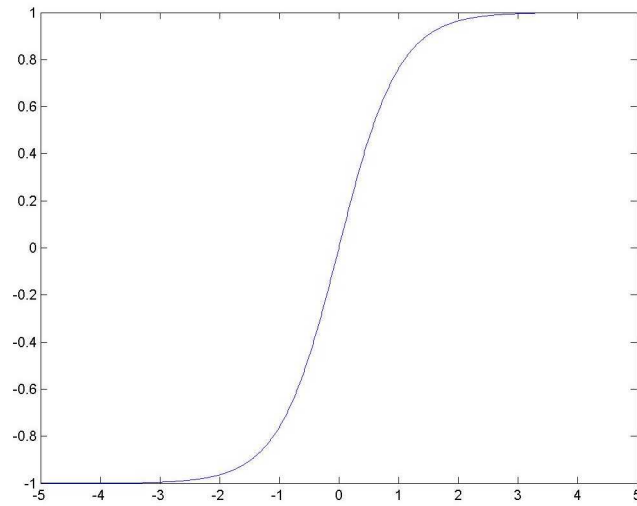


Figura 3.2: Tangente Hiperbólica

O número de neurônios escolhido para a camada escondida foi 14. Este valor foi obtido a partir da simulação da rede para diversos tamanhos da camada oculta, sendo que a rede que apresentou melhor desempenho foi a que possuía 14 unidades ocultas.