Um Critério Baseado no Casamento de Distribuições Multivariadas para o Treinamento de Redes LSTM

Otávio Rodrigues de Oliveira, Henrique Luiz Voni Giuliani, Amanda Polastro, Denis Gustavo Fantinato
Centro de Matemática, Computação e Cognição (CMCC)
Universidade Federal do ABC (UFABC)
Santo André-SP, Brasil 09210–580
Email: otavio.rodrigues@aluno.ufabc.edu.br,{henrique.voni,amanda.leite,denis.fantinato}@ufabc.edu.br

***Resumo*—Em problemas de predição de séries temporais, as Redes Neurais Recorrentes (RNNs, do inglês *Recurrent Neural Networks*) despontam como importantes estruturas de processamento de informação. Em particular, as RNNs do tipo LSTM (do inglês *Long Short-Term Memory*) possuem distintos mecanismos para tratar concomitantemente memórias de curto e de longo prazo, o que lhes garante um enorme potencial para o tratamento da informação. No entanto, o uso do erro quadrático médio (MSE, do inglês *Mean Squared Error*) como critério de otimização pode trazer algumas limitações ao desempenho da LSTM. Nesse sentido, o presente trabalho propõe o uso de um critério baseado no casamento de distribuições multivariadas ao invés do MSE para o treinamento da LSTM. Os resultados envolvendo quatro diferentes *datasets* para predição são favoráveis a esta abordagem que abre novas perspectivas para o uso da LSTM.**

***Keywords*—Redes Neurais Recorrentes; LSTM; Casamento de PDF**

I. Introdução

As Redes Neurais Artificiais (RNAs) são poderosas estruturas de processamento da informação inspiradas nas células neuronais humanas, possuindo plasticidade para aprender a desempenhar uma determinada tarefa, ao invés de seguir de forma estrita um algoritmo sequencial na programação clássica [1]. Tal característica permitiu que tarefas antes dependentes da intuição ou percepção humana passassem a ser realizadas por máquinas capazes de aprender, alcançando desempenho notável em problemas como o de classificação e o de regressão [1], [2]. O grande interesses pelas RNAs contribuíram para sua rápida disseminação à diversas áreas do conhecimento, bem como para o desenvolvimento de estruturas de redes mais especializadas e com maior poder de processamento, como as redes neurais de aprendizado profundo (*Deep learning*) [3].

Para o processamento de séries temporais, principalmente na tarefa de predição, um dos tipos de estruturas de processamento mais utilizados é a chamada Rede Neural Recorrente (RNN, do inglês *Recurrent Neural Network*), que segue a mesma perspectiva conexionista das RNAs tradicionais, mas possui o diferencial dos laços de realimentação da informação [4]. Tais conexões retroativas permitem que a informação gerada pela RNN no passado seja utilizada para produzir as saídas presentes. Essa abordagem é um elemento chave para séries temporais, visto que estas geralmente possuem dependência entre as amostras [5].

Entretanto, um problema enfrentado pelas RNNs é a dificuldade em se priorizar dados do passado baseado em pontos diferentes em relação ao tempo presente, que podem ser de longo ou curto prazo. Torna-se necessário o uso de um mecanismo capaz de tratar concorrentemente esses dois tipos de informação. Como uma forma de se resolver esse problema, foi desenvolvida a chamada rede LSTM (do inglês *Long Short-Term Memory*), que é capaz de criar uma memória dedicada para eventos de curto prazo e outra para eventos de longo prazo [4].

Desde sua proposição em 1995 [6], a rede LSTM ganhou algumas mudanças em sua estrutura, como a utilização de três *gates* de processamento e as conexões *peephole*, que são capazes de traduzir o estado de memória que cada unidade da LSTM se encontra [7]. Apesar de importantes, essas e outras modificações na estrutura foram propostas, mas não levaram a um aumento significativo de desempenho [4]. Tais resultados são indícios de um possível elemento que esteja limitando o desempenho da LSTM: o critério de otimização. Geralmente, a função objetivo que deseja-se minimizar é a do Erro quadrático Médio (MSE, do inglês *Mean Squared Error*), entretanto, quando se lida com dados/sinais com dependência estatística (como é o caso de séries temporais ou imagens), o MSE pode se mostrar como uma medida limitada, incapaz de representar características importantes sobre os dados [8]. De fato, sob a perspectiva estatística, o MSE restringe-se ao momento estatístico de segunda-ordem sobre o sinal de erro [9].

Nesse âmbito, o arcabouço de aprendizado baseado na teoria da informação (ITL, do inglês *Information Theoretic Learning*) pode trazer conceitos e entidades capazes de descrever as características subjacentes aos dados de uma forma mais completa [10], [11]. Os critérios de ITL, como a entropia de Shannon [12], são capazes de usar toda a informação estatística da distribuição dos sinais – sem limitar-se ao momento de segunda-ordem. Particularmente, para sinais temporais, uma descrição mais rica da dependência estatística

pode ser alcançada através da manipulação de distribuições multivariadas [13], [14]. Dessa forma, desponta como promissor o critério de ITL baseado no casamento de distribuições, com a abordagem multivariada [15].

Portanto, neste trabalho pretendemos analisar o desempenho da LSTM trocando-se o critério baseado no MSE por aquele baseado no casamento de distribuições multivariadas. Serão realizados ensaios com quatro *datasets* diferentes de séries temporais, sendo comparado o desempenho da LSTM treinada com cada um dos dois critérios. Vale ressaltar que, diferentemente do MSE, o casamento de distribuições pode ser seguido através de uma abordagem semi-supervisionada, abrindo um amplo leque de possibilidades para o treinamento da LSTM. Espera-se que o desempenho alcançado seja equivalente ou superior ao MSE.

Este trabalho está organizado da seguinte forma - na Seção II, a estrutura da LSTM e o funcionamento de suas células são descritos. A Seção III apresenta o critério de ITL baseado no casamento de distribuições e sua estimação através do método de Parzen Window. A análise dos parâmetros do modelo e do critério são descritos na Seção IV e os resultados finais obtidos na Seção V. Finalmente, as conclusões do trabalho são apresentadas na Seção VI.

II. A Rede LSTM

A LSTM é um tipo específico de RNN que mantém os laços de realimentação, porém os estende para diferentes propósitos. O termo *Long Short-Term Memory* vem da intuição de que tais redes podem construir uma memória de longo prazo e, concomitantemente, uma memória de curto prazo, na forma de rápidas ativações dos nós consecutivos da rede. De forma bastante interessante, a LSTM é capaz de estabelecer um *tradeoff* entre as informações de longo e curto prazo através de suas células de memória [4], conforme apresentado na Fig. 1.

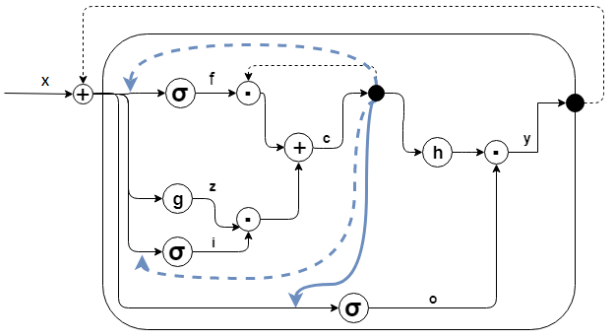

Figura 1. Célula da Rede LSTM

Elas possuem uma entrada de dados $\mathbf{x}^t$ para o bloco, três *gates* (input, forget e output), uma função de ativação de saída e conexões *peephole*, que permitem que os elementos da LSTM consigam obter informações do estado da célula (denotadas pelas linhas azuis). A realimentação do sinal de saída $\mathbf{y}^t$ acontece tanto para a entrada do bloco quanto para os gates.

No fluxo *forward*, considerando um vetor de entrada $\mathbf{x}^t$ no instante de tempo t, N o número de blocos LSTM e M o número de inputs, são descritos em uma camada LSTM os seguintes pesos: (*i*) Pesos de entrada - $\mathbf{W}_z$, $\mathbf{W}_i$, $\mathbf{W}_f$, $\mathbf{W}_o \in \mathbb{R}^{N\times M}$, (*ii*) Pesos Recorrentes - $\mathbf{R}_z$, $\mathbf{R}_i$, $\mathbf{R}_f$, $\mathbf{R}_o \in \mathbb{R}^{N\times N}$, (*iii*) Pesos peephole - $\mathbf{p}_i$, $\mathbf{p}_f$, $\mathbf{p}_o \in \mathbb{R}^N$ e (*iv*) Pesos dos bias - $\mathbf{b}_z$, $\mathbf{b}_i$, $\mathbf{b}_f$, $\mathbf{b}_o \in \mathbb{R}^N$; sendo os índices z, i, f e o relativos à entrada do bloco, input gate, forget gate e output gate, respectivamente.

E a partir deste ponto, tem-se as seguintes equações para o denominado *foward pass* de uma LSTM, que representa o movimento da transformação do dado de entrada ao longo da célula até sua saída:

$$\begin{aligned}
\mathbf{z}^t &= g\left(\mathbf{W}_z\mathbf{x}^t + \mathbf{R}_z\mathbf{y}^{t-1} + \mathbf{b}_z\right)\\
\mathbf{i}^t &= \sigma\left(\mathbf{W}_i\mathbf{x}^t + \mathbf{R}_i\mathbf{y}^{t-1} + \mathbf{p}_i \odot \mathbf{c}^{t-1} + \mathbf{b}_i\right)\\
\mathbf{f}^t &= \sigma\left(\mathbf{W}_f\mathbf{x}^t + \mathbf{R}_f\mathbf{y}^{t-1} + \mathbf{p}_f \odot \mathbf{c}^{t-1} + \mathbf{b}_f\right)\\
\mathbf{c}^t &= \mathbf{z}^t \odot \mathbf{i}^t + \mathbf{c}^{t-1} \odot \mathbf{f}^t\\
\mathbf{o}^t &= \sigma\left(\mathbf{W}_o\mathbf{x}^t + \mathbf{R}_o y^{t-1} + \mathbf{p}_o \odot \mathbf{c}^t + \mathbf{b}_o\right)\\
\mathbf{y}^t &= h(\mathbf{c}^t) \odot \mathbf{o}^t
\end{aligned} \tag{1}$$

em que $\sigma(\cdot)$, $g(\cdot)$ e $h(\cdot)$ são funções de ativação não-lineares, sendo $\sigma(x) = \frac{1}{1+e^{-x}}$, $g(x) = h(x) = tanh(x)$ e $\odot$ o produto da matriz elemento-a-elemento. O sinal $\mathbf{c}^t$ ocupa papel central na célula e representa o estado da célula, sendo controlado pelos gates, cada qual com sua função específica: o input gate gera o sinal $\mathbf{i}^t$ que atuará sobre o sinal $\mathbf{z}^t$, que é resultado da entrada da célula, e fornecerá a excitação para o estado da célula; o forget gate produz o sinal $\mathbf{f}^t$, que é responsável por zerar o estado da célula (pode atenuar o estado anterior) e, por fim, o output gate gera o sinal $\mathbf{o}^t$ que atua sobre a saída $\mathbf{y}^t$ da célula. As conexões peephole não são consideradas neste trabalho.

A Eq. (1) representa todas as conexões de uma camada da LSTM, podendo ser estendida para um número arbitrário de camadas. Neste caso, a saída $\mathbf{y}^t$ será a entrada $\mathbf{x}^t$ da próxima camada.

É possível observar que a LSTM pode possuir um elevado número de pesos (considerando aqueles que ponderam as entradas e os gates). Estes são geralmente treinados através do critério do Erro Quadrático Médio (MSE), cuja função custo é definida como:

$$J_{MSE} = \frac{1}{N_y}\sum_{i=1}^{N_y}\left|s^t - y^t\right|^2, \tag{2}$$

em que N_y é o número de amostras de y^t e s^t é o sinal desejado no instante t. O ajuste dos pesos é feito pelo algoritmo *Backpropagation Through Time*, que usa um gradiente aproximado de J_{MSE} em relação aos pesos considerando uma dependência temporal truncada dos sinais da recursão (usualmente considera-se um único atraso).

O MSE é uma medida bastante adequada para estruturas lineares [1]. Entretanto, as funções de ativação - bem como as atuações multiplicativas dos gates - geram um caráter fortemente não linear nas redes LSTM, sendo este um importante

indício de que o MSE pode ser um limitador de desempenho. Baseando-se nisso, buscaremos utilizar um critério alternativo.

III. Casamento de Distribuições

O treinamento das redes LSTM é classicamente realizado pelo MSE, que é um critério supervisionado dotado de interessantes propriedades, como continuidade e simples derivação [10]. Entretanto, em domínios de sinais que apresentam dependência estatística, como sinais temporais e imagens, essa medida mostra-se incompleta em termos estatísticos, pois restringe-se ao momento estatístico de segunda-ordem do erro, que pode representar uma caracterização estatística limitada e parcial [8], principalmente pelo caráter não linear das redes LSTM. Uma alternativa capaz de extrair as informações subjacentes aos dados de forma mais efetiva é a utilização da função densidade de probabilidade (PDF, do inglês *Probability Density Function*) do sinal, visto que esta é capaz de carregar todos os momentos estatísticos [9]. Dentro do aprendizado baseado na teoria da informação (ITL), esta é uma abordagem que se mostra bastante promissora para estruturas não lineares de processamento e com recursão [13], [15].

De forma mais aprofundada, para sinais temporais, foco do presente trabalho, é possível potencializar o uso da informação sobre dependência estatística através de PDFs multivariadas. Uma distribuição pode ser analisada de uma perspectiva temporal da seguinte maneira. Assume-se que $\mathbf{x}^t = \begin{bmatrix} x^t & x^{t-1} & \dots & x^{t-M} \end{bmatrix}^T$ é o vetor composto do sinal x^t no instante t junto das suas M versões atrasadas, tendo associado a si o vetor de coeficientes da variável aleatória $\underline{X} = \{X^t, X^{t-1}, \dots, X^{t-M}\}$ e PDF $f_{\underline{X}}(\mathbf{v})$. Assim, a distribuição $f_{\underline{X}}(\mathbf{v})$ carrega todas as informações estatísticas de x^t, bem como de sua relação de dependência com as demais versões atrasadas [15].

Nesse sentido, um possível critério alternativo ao MSE é o casamento de distribuições multivariado por distância quadrática (MQD, do inglês *Multivariate Quadratic Distance* [11], [15], que pode ser definido como:

$$\begin{aligned} J_{MQD} &= \int_D (f_{\underline{Y}}(\mathbf{v}) - f_{\underline{S}}(\mathbf{v}))^2 d\mathbf{v} \\ &= \int_D f^2_{\underline{Y}}(\mathbf{v}) d\mathbf{v} + \int_D f^2_{\underline{S}}(\mathbf{v}) d\mathbf{v} - 2\int_D f_{\underline{Y}}(\mathbf{v}) f_{\underline{S}}(\mathbf{v}) d\mathbf{v} \end{aligned} \tag{3}$$

em que $D \subseteq \mathbb{R}^{(M+1)}$ e $f_{\underline{Y}}(\mathbf{v})$ e $f_{\underline{S}}(\mathbf{v})$ são as PDFs associadas à saída da rede LSTM, y^t e ao sinal desejado s^t, respectivamente. Uma diferença fundamental em relação ao MSE é a ausência da comparação amostra-a-amostra, permitindo que uma abordagem semi-supervisionada seja seguida, i.e., em posse da distribuição alvo $f_{\underline{S}}(\mathbf{v})$, não são necessárias amostras de referência s^t para o MQD. Isto abre um amplo horizonte de possibilidades de aplicação. No entanto, por outro lado, o critério MQD pode não ser suficiente para garantir que y^t se assemelhe ao máximo a s^t, mas existem importantes indícios de que, ao usar M grande o suficiente para extrair a informação temporal, y^t aproxime-se suficientemente de s^t [13], [15]. Estes aspectos serão investigados futuramente.

Devido às realimentações presentes na LSTM, o conhecimento pleno de $f_{\underline{Y}}(\mathbf{v})$ é difícil de ser obtido analiticamente. Além disso, no contexto atual de aprendizado de máquina, geralmente conhece-se amostras de s^t, e não sua distribuição. Dessa forma, optamos por estimar as distribuições $f_{\underline{Y}}(\mathbf{v})$ e $f_{\underline{S}}(\mathbf{v})$ através do método de *Parzen Window*, que aplica funções kernel sobre os dados [11], [13], i.e.:

$$\hat{f}_{\underline{X}}(\mathbf{v}) = \frac{1}{N}\sum_{i=1}^{N} \kappa_\sigma \left(\frac{\mathbf{v} - x_i}{\sigma} \right), \tag{4}$$

onde $\kappa_\sigma(\cdot)$ é a função kernel multivariada e σ é o tamanho do kernel [11], [13].

Uma das funções kernel amplamente utilizadas é a Gaussiana, devido à sua maior ocorrência na natureza e sua relativa simplicidade [11]. Matematicamente, a versão multivariada desse kernel pode ser definida como [13]:

$$\begin{aligned} G_\Sigma(\mathbf{v} - \mathbf{x}_i) = &\frac{1}{\sqrt{(2\pi)^{M+1}\det(\Sigma)}} \\ &\cdot exp\left[\frac{-1}{2} (\mathbf{v} - x_i)^T \Sigma^{-1} (\mathbf{v} - x_i) \right], \end{aligned} \tag{5}$$

em que $\Sigma = \sigma^2 I_{M+1}$ é a matriz de covariância e $\det(\cdot)$ é a operação determinante. Como é contínua e diferenciável, essa função kernel é muito útil para ser usada em técnicas de otimização em Redes Neurais Artificiais [11].

Finalmente, aplicando o kernel Gaussiano multivariado, Eq. (5), em Eq. (4) e substituindo em Eq. (3), resulta

$$\begin{aligned} \hat{J}_{MQD} = &\frac{1}{N_y^2} \sum_{i=0}^{N_y-1} \sum_{j=0}^{N_y-1} G_{2\Sigma}(y^{t-i} - y^{t-j}) \\ &+ \frac{1}{N_s^2} \sum_{i=0}^{N_s-1} \sum_{j=0}^{N_s-1} G_{2\Sigma}(s^{t-i} - s^{t-j}) \\ &- \frac{2}{N_y N_s} \sum_{i=0}^{N_y-1} \sum_{j=0}^{N_s-1} G_{2\Sigma}(y^{t-i} - s^{t-j}) \end{aligned} \tag{6}$$

em que N_y e N_s é o número de vetores-amostras usado para estimar $f_{\underline{Y}}(\mathbf{v})$ e $f_{\underline{S}}(\mathbf{v})$, respectivamente. O objetivo é minimizar $\hat{J}_{MQD}$, que atinge seu valor mínimo quando as distribuições são equivalentes.

Vale ressaltar que o custo dado pela Eq. (6) é diferenciável e pode ser utilizado para o ajuste da LSTM através do método de backpropagation, de forma similar como é feito para o MSE [4]. Apesar de se tratar de um critério computacionalmente mais complexo, o MQD pode carregar a informação de dependência temporal de forma mais eficiente. Assim, pretendemos utilizá-lo como substituto ao MSE no treinamento da LSTM visando permitir que os pesos das realimentações sejam ajustados de forma mais precisa.

IV. Bases de Dados e Ajustes dos Parâmetros

Com o intuito de analisar o MQD como critério de treinamento para a LSTM, iremos compará-lo ao clássico critério de MSE para a tarefa de predição. O desempenho em cada

caso será avaliado em termo das seguintes métricas sobre o conjunto de testes:

- MAE, *Mean Absolute Error*:

$$MAE = \frac{\sum_{i=1}^{N} |s^i - y^i|}{N} \tag{7}$$

- RMSE, *Root Mean Square Error*:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(s^i - y^i)^2} \tag{8}$$

- SMAPE, *Symmetric Mean Absolute Percent Error*:

$$SMAPE = \frac{100\%}{N}\sum_{i=1}^{N}\frac{|y^i - s^i|}{(|s^i| + |y^i|)/2} \tag{9}$$

Em todos os casos, quanto menor o valor aferido, melhor o desempenho da rede LSTM. Serão utilizadas 4 bases de dados, conforme descrito a seguir.

A. Bases de Dados Utilizadas

A avaliação do modelo foi realizada usando 4 bases de dados reais: *Beijing PM2.5*[1], *Bike Sharing*[2], *NSW2016* e *TAS2016*[3]. Para obter o dado em intervalos anuais, foram concatenados os dados mensais de preço e demanda de eletricidade das regiões de New South Wales e Tasmania. Apresentados na Tab. I estão o número de amostras e número de variáveis de cada uma das bases de dados.

Tabela I
BASES DE DADOS UTILIZADAS

Base de Dados	Nº de Amostras	Qtd. Variáveis
Beijing PM2.5	43824	13
Bike Sharing	17379	17
NSW 2016	17568	5
TAS 2016	17568	5

Todas as bases são formadas por séries temporais reais, e o sinal desejado é a previsão de algum índice relativo a cada dado, ou seja, para a base *Beijing PM 2.5* será a previsão do índice do poluente PM 2.5 na cidade de *Beijing*, para *Bike Sharing* a previsão do número de aluguéis de bicicletas em determinado período e para as bases *NSW2016* e *TAS2016* a previsão da demanda de energia elétrica para as duas regiões.

Para cada base, os dados foram normalizados usando o método min-max - uma técnica de redimensionamento relativa aos valores mínimo e máximo, restringindo a base a um intervalo de 0 a 1.

Devido à relação temporal dos dados, as bases foram divididas em conjuntos de treino e teste, representando respectivamente partições de 75% e 25%.

[1]https://archive.ics.uci.edu/ml/datasets/Beijing+PM2.5+Data

[2]https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset

[3]https://www.aemo.com.au/Electricity/National-Electricity-Market-NEM/Data-dashboard#aggregated-data

B. Parâmetros da Rede LSTM e do Critério MQD

São considerados seis parâmetros para a LSTM e seu treinamento: (i) número de camadas, (ii) número de células por camada (ou tamanho da camada), (iii) *lag*, (iv) *learning rate*, (v) número de épocas e (vi) *Batch Size*. O número de camadas (intermediárias) e o número de células por camada definem estruturalmente a rede LSTM. O *lag* está relacionado ao intervalo de tempo entre o valor presente da entrada e o valor predito pela rede, antecipando-o em k unidades de tempo (i.e. quão futura será a previsão) para, por exemplo, um sinal presente x^t. O valor esperado é representado por $d^t = x^{t+k}$. O *learning rate*, número de épocas e *Batch Size* estão relacionados ao treinamento da LSTM. O *learning rate* é um parâmetro de treinamento que está associado à taxa de mudança dos valores dos pesos dentro das células. O número de épocas indica quantas vezes a base de dados inteira realizará o percorrimento de ida (*foward pass*) e de volta (*backward pass*) dentro do modelo. Por fim, visando o processamento dos dados em blocos (ou *Batches*), o *Batch Size* define a quantidade de amostras de entrada processadas para cada ajuste dos pesos.

Em relação ao critério MQD, há dois parâmetros a serem considerados: o tamanho do kernel σ e a quantidade M de versões atrasadas do sinal de entrada. O tamanho do kernel está associado à suavidade da PDF, enquanto que M define a dimensão da PDF. Quanto maior M, mais informação a PDF poderá carregar. No entanto, a complexidade computacional do estimador aumenta e também requer mais amostras para obter uma estimativa mais fidedigna.

C. Ajuste dos Parâmetros

A fim de encontrar os valores adequados dos parâmetros da LSTM e do critério MQD, percorreu-se um conjunto de valores para cada parâmetro de forma independente. De fato, essa abordagem pode levar a valores subótimos, mas consideramos que esta é uma opção considerável tendo em vista o elevado número de parâmetros.

De maneira a simplificar a comparação e padronizar os experimentos, partiu-se dos parâmetros mostrados na Tab. II para, em seguida, percorrer seus diferentes valores.

Tabela II
PARÂMETROS DE INICIAIS CONSIDERADOS

Nº de camadas	1
Tamanho das camadas	64
Lag	24
Learning Rate	10^{-4}
Nº de épocas	20
Batch Size	32
σ	1
M	10

A análise dos parâmetros será baseada no primeiro conjunto de dados, o *Beijing PM 2.5*. Começou-se considerando o *Lag* da rede, que podia assumir os valores $k = 24$ ou $k = 5$. Os resultados obtidos em termos das medidas MAE, RMSE e SMAPE são conforme mostra a Tab. III.

Tabela III
MUDANÇA DO PARÂMETRO *Lag* - *Beijing PM 2.5*

Beijing PM2.5				
	Lag = 24		Lag = 5	
	MSE	MQD	MSE	MQD
MAE	14.20	13.95	13.12	12.81
RMSE	25.01	24.84	24.81	24.97
SMAPE	26.10	26.16	21.79	20.92

O valor inicial desse parâmetro (24) foi escolhido devido à base cujos registros foram realizados a cada uma hora. Logo, ao utilizar um intervalo de previsão de vinte e quatro horas, tem-se a previsão do próximo dia. Mas, como se tem um desempenho melhor num período menor de tempo, foi escolhido o período de cinco horas para minimizar grandes variações no treinamento da rede.

Variou-se também o tamanho da camada, conforme Tab. IV. Usando o critério MSE, teve-se um melhor resultado utilizando uma camada de tamanho 8. Enquanto isso, para o MQD, uma camada de tamanho 32 acabou se saindo melhor. Logo, esses serão os valores utilizados para os próximos testes.

Tabela IV
MUDANÇA DO PARÂMETRO TAMANHO DE CAMADAS - *Beijing PM 2.5*

Beijing PM2.5						
	Hidden = 8		Hidden = 32		Hidden = 64	
	MSE	MQD	MSE	MQD	MSE	MQD
MAE	12.75	15.02	13.12	12.81	13.31	13.33
RMSE	24.54	26.21	24.81	24.97	25.10	24.88
SMAPE	20.54	23.86	21.79	20.92	22.56	21.22

A Tab. V mostra a variação da *Learning Rate*. Com essa mudança, tem-se que o valor de 10^{-5} é o melhor valor para o treinamento da rede em ambos os casos. Como esse foi um caso de rápida convergência, o uso de um *Learning Rate* menor ocasionou em um resultado melhor, já que são realizadas mudanças menos abruptas e mais refinadas ao longo do treinamento.

Tabela V
MUDANÇA DO LEARNING RATE - *Beijing PM 2.5*

Beijing PM2.5			
Learning Rate $= 10^{-5}$	MAE	RMSE	SMAPE
MSE	12.75	24.54	20.54
MQD	12.81	24.97	20.92
Learning Rate $= 10^{-4}$	MAE	RMSE	SMAPE
MSE	14.70	25.32	27.70
MQD	13.04	24.60	22.21
Learning Rate $= 10^{-3}$	MAE	RMSE	SMAPE
MSE	13.37	24.48	27.95
MQD	15.04	25.22	26.94

Em seguida, foi-se estudado o efeito da mudança do valor σ do critério de avaliação conforme demonstrado na Tab. VI. Pode-se verificar que geralmente busca-se um valor de trade-off para esse parâmetro, já que ao mesmo tempo que um valor de σ alto é mais tolerante a uma maior variação na distribuição dos dados, a rede passa a ser menos precisa nos valores mais frequentes. O contrário também é valido, fazendo a estimativa ser mais precisa nos picos da distribuição e ignorando valores menos frequentes. Nos testes, o valor 0.75 foi escolhido devido a seu melhor desempenho. Logo, será o valor utilizado nos próximos testes.

Tabela VI
MUDANÇA DO σ DO CASAMENTO DE DISTRIBUIÇÕES - *Beijing PM 2.5*

Beijing PM2.5, Lag = 5, hidden = 32					
	$\sigma = 1$	$\sigma = 0.75$	$\sigma = 0.6$	$\sigma = 0.5$	$\sigma = 0.25$
MAE	12.81	12.80	12.89	13.33	13.39
RMSE	24.97	24.85	24.80	24.70	25.27
SMAPE	20.92	20.24	21.66	22.68	23.15

Outro parâmetro do critério MQD estudado foi o valor de M. Observando a Tab. VII, pode-se perceber que um valor próximo do *lag* da rede ocasiona em um desempenho melhor, já que o vetor é comparado dentro do intervalo de previsão da rede. Também é possível observar que valores mais altos para esse parâmetro pode ocasionar uma perda de precisão, já que torna-se necessário usar mais dados para estimar melhor as distribuições. Logo, o valor de 5 foi escolhido para esse parâmetro.

Tabela VII
MUDANÇA DO M DO CASAMENTO DE DISTRIBUIÇÕES - *Beijing PM 2.5*

Beijing PM2.5, Lag = 5, hidden = 32				
	M = 2	M = 5	M = 10	M = 15
MAE	13.98	12.80	13.17	14.40
RMSE	24.35	24.85	24.62	24.61
SMAPE	26.69	20.24	21.71	26.02

D. Inclusão de Atributos

Uma alternativa quando se lida com séries temporais é a geração de outros atributos, que serão tratados aqui como componentes dos seguintes tipos: sistemáticos, *i.e.* que são consistentes ou possuem uma recorrência e conseguem ser descritos e modelados; e um não-sistemático, que não pode ser diretamente modelado. Para isso, a série foi decomposta em dois sistemáticos: Tendência, se a série está crescendo ou decrescendo em dado momento; e Sazonalidade, ciclos que ocorrem em um curto espaço de tempo. Também decomposta em um não-sistemático: Ruído, variação aleatória na série. Com essa decomposição e passando cada elemento separado na LSTM foram obtidos os resultados exibidos na Tab. VIII.

Analisando os resultados da Tab. VIII, pode-se perceber que o MQD só obteve um desempenho melhor no componente de Tendência. Esse teste foi interessante pois pode-se perceber que é possível modelar o dado dessa maneira e realizar o treinamento utilizando o Casamento de Distribuições e se obter a convergência da rede. Uma análise mais detalhada desses atributos será deixada para trabalhos futuros.

Tabela VIII
DECOMPOSIÇÃO DA SÉRIE TEMPORAL - *Beijing PM 2.5*

Beijing PM2.5, Lag = 5, hidden = 32			
Tendência	MAE	RMSE	SMAPE
MSE	2.43	3.15	6.15
MQD	2.30	3.13	5.82
Ruído	MAE	RMSE	SMAPE
MSE	10.43	18.01	130.90
MQD	10.76	18.24	132.58
Sazonalidade	MAE	RMSE	SMAPE
MSE	0.0003	0.0003	0.0685
MQD	0.0011	0.0013	0.3807
Total	MAE	RMSE	SMAPE
MSE	11.04	18.07	21.73
MQD	11.18	18.46	22.70

E. Análise Gráfica

Utilizando o dataset *Beijing PM 2.5* e os parâmetros ótimos ajustados previamente, foram gerados histogramas, onde Casamento e MSE representa a saída da LSTM para cada um dos dois critérios e o valor esperado representa o a saída esperada da rede, como mostrado na Fig. 2.

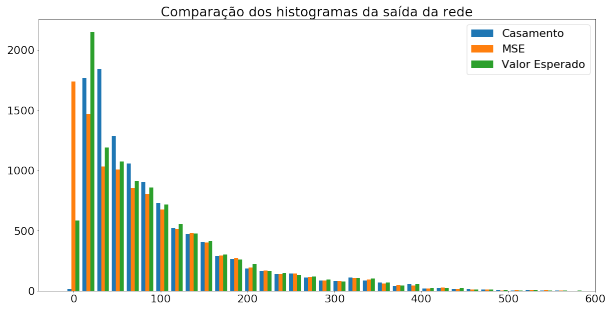

Figura 2. Comparação dos histogramas resultantes do treinamento completo da rede

Para o treinamento completo (Fig. 2), é possível observar que apesar dos dois métodos possuírem bastante divergência do valor esperado no começo da distribuição, a MSE acaba errando mais para os valores mais frequentes.

Acerca dos resultados é possível perceber que o MQD age de acordo com o esperado, obtendo uma distribuição mais próxima à da série original. Enquanto isso, o MSE é assertivo quando se trata apenas dos valores puros, ao mesmo tempo que, todavia, possam ocorrer divergências na distribuição da saída da rede.

Utilizando os atributos de decomposição da série temporal, como apresentado na Fig. 3, o MSE consegue aproximar-se mais da distribuição desejada, e o mesmo ocorre para o MQD. Neste caso, os erros entre os dois critério tornam-se bastante similares, apesar de haver maior divergência em relação aos valores mais baixos em ambos os critérios.

V. RESULTADOS

Para as demais bases de dados, o procedimento de ajuste dos parâmetros foi repetido, tendo sido selecionado os parâmetros para o MSE e para o MQD conforme a Tab. IX. Nesta etapa,

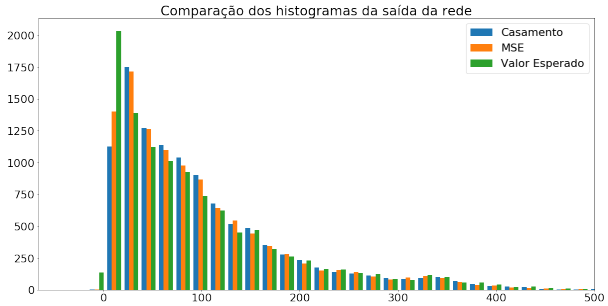

Figura 3. Comparação dos histogramas resultantes do treinamento da rede utilizando a decomposição

não foram considerados os atributos adicionais (tendência, ruído e sazonalidade).

Tabela IX
PARÂMETROS DE TREINAMENTO

Parâmetros da Rede - MSE e MQD				
Parâmetros	Beijing PM 2.5	Bike Sharing	NSW2016	TAS2016
Tam. camada	32	64	64	32
Lag	5	24	24	24
Learn. Rate	10^{-5}	10^{-4}	10^{-4}	10^{-4}
Parâmetros Específicos do MQD				
Parâmetros	Beijing PM 2.5	Bike Sharing	NSW2016	TAS2016
σ	0.75	0.1	0.25	0.25
M	5	30	20	20

Após o ajuste, a LSTM foi treinada com os dois critérios para cada base de dados. Os resultados obtidos para o conjunto de testes estão mostrados na Tab. X. De uma forma geral,

Tabela X
RESULTADOS

Base	Beijing PM2.5		Bike Sharing	
	MSE	MQD	MSE	MQD
MAE	12.75	12.80	36.51	32.14
RMSE	24.54	24.85	54.81	48.91
SMAPE	20.54	20.24	28.31	25.59
Base	NSW 2016		TAS 2016	
	MSE	MQD	MSE	MQD
MAE	72	84.15	14.89	15.26
RMSE	92.00	107.29	21.01	21.52
SMAPE	0.96	1.12	1.42	1.46

o desempenho do MQD e MSE ficaram bastante próximos, principalmente para as bases de dados *Beijing PM 2.5* e *TAS 2016*. Entretanto, para a base *Bike Sharing*, o MQD foi capaz de superar o desempenho do MSE, aumentando a qualidade da predição. Por outro lado, para a base de dados *NSW 2016*, o MSE se mostrou superior.

Esses resultados indicam que o MQD pode ser equiparável ao MSE ou até mesmo superior, mas sofre por dois principais motivos: (*i*) exige o ajuste dos parâmetros adicionais σ e M, que, dependendo da base de dados, pode exigir um ajuste mais fino – muito provavelmente, este foi o motivo

de degradação de desempenho para a base *NSW 2016*; e (*ii*) por se tratar de um método semi-supervisionado, não realiza a comparação amostra-a-amostra – nesse sentido, pode ser interessante iniciar o treinamento pelo critério MQD e, após convergência, fazer a troca pelo MSE; isso permitirá, por exemplo, que durante a etapa de treinamento com o MQD a distribuição da saída se acomode de forma melhor ao esperado (como observado nos histogramas da análise gráfica) e, usando o MSE, o ajuste da saída pode ser refinado, i.e., o MQD poderia auxiliar o MSE a evitar a convergência para soluções locais.

VI. Conclusão

A fim de evitar possíveis limitações causadas pelo uso do MSE como critério de treinamento para a LSTM, este trabalho propôs o uso de um critério de ITL baseado no casamento de distribuições multivariada, o MQD. Para a estimação das distribuições multivariadas, adotou-se o método de Parzen Window com kernels Gaussianos. Além dos parâmetros tradicionais da rede LSTM, como o número de camadas, número de células e taxa de aprendizado, o critério MQD depende do tamanho do kernel e da dimensão da distribuição multivariada. Após realizada uma varredura para alguns valores dos parâmetros, detectou-se aqueles que levavam a um melhor desempenho em termos da medida MAE. Os resultados mostraram que o uso do MQD pode levar a um melhor desempenho em relação ao MSE, pois é capaz de representar a dependência temporal de forma mais eficaz, como observado pela análise dos histogramas. No entanto, as características dos dados podem influenciar fortemente, pois para casos mais simples, o MSE já é suficiente, sendo recomendado o MQD para problemas mais complexos. Além disso, pode ser necessário um ajuste mais fino dos parâmetros do MQD.

Por ser um critério semi-supervisionado, o uso do MQD abre novas perspectivas para o treinamento da LSTM, sendo possível utilizar apenas uma distribuição alvo; ao invés de usar amostras da resposta desejada. Da mesma forma, esta abordagem permite também uma atuação intracelular da LSTM. Entretanto, tais perspectivas e o estudo em conjuntos de dados mais expressivos serão deixados para trabalhos futuros.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Referências

[1] S. S. Haykin *et al.*, *Neural networks and learning machines/Simon Haykin.* New York: Prentice Hall,, 2009.

[2] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[4] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2016.

[5] J. L. Elman, "Finding structure in time," *Cognitive science*, vol. 14, no. 2, pp. 179–211, 1990.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[7] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," in *1999 Ninth International Conference on Artificial Neural Networks (ICANN)*, vol. 2, no. 470, 1999, pp. 850–855.

[8] Z. Wang and A. C. Bovik, "Mean squared error: Love it or leave it? a new look at signal fidelity measures," *IEEE signal processing magazine*, vol. 26, no. 1, pp. 98–117, 2009.

[9] D. G. Fantinato, A. Neves, and R. Attux, "Analysis of a novel density matching criterion within the itl framework for blind channel equalization," *Circuits, Systems, and Signal Processing*, vol. 37, no. 1, pp. 203–231, 2018.

[10] R. Attux, L. Boccato, D. Fantinato, J. Montalvao Filho, A. Neves, R. Suyama, K. Nose Filho, and D. Silva, "Signals and images: Advances and results in speech, estimation, compression, recognition, filtering, and processing, chapter bio-inspired and information-theoretic signal processing," 2015.

[11] J. C. Principe, *Information theoretic learning: Renyi's entropy and kernel perspectives*. Springer Science & Business Media, 2010.

[12] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.

[13] D. W. Scott, *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015.

[14] D. G. Fantinato, L. Boccato, R. Attux, and A. Neves, "Multivariate pdf matching via kernel density estimation," in *2014 IEEE Symposium on Computational Intelligence for Multimedia, Signal and Vision Processing (CIMSIVP)*. IEEE, 2014, pp. 1–8.

[15] D. Fantinato, D. G. e Silva, R. Attux, and A. Neves, "Multivariate shannon's entropy for adaptive iir filtering via kernel density estimators," *Electronics Letters*, 2019.