

ENG1456 - Redes Neurais - Trabalho 2

Previsão de Séries Temporais

Aluno: Matheus Carneiro Nogueira - 1810764

Professora: Marley Velasco

Sumário

1	Compreensão do Problema	2
1.1	Visualize, em forma de gráfico, a dinâmica temporal da série escolhida. A série é adequada para a modelagem usando Redes Neurais? Caso não seja, que técnicas podem ser aplicadas para ajustar o comportamento da série? .	2
1.2	No problema escolhido, usaremos uma variável exógena que representa o mês de previsão (i.e. no instante $t+1$). De que forma esta variável pode auxiliar na previsão da série temporal?	4
2	Previsão One-Step	4
2.1	Execute o script para a previsão one-step. Analise o resultado (conjunto de treinamento e teste), usando as métricas RMSE e MAE.	4
2.2	Modifique a técnica de codificação mensal de ‘numérico’ para ‘binário’. Qual a mudança existente na arquitetura da Rede Neural? Analise o resultado (conjunto de treinamento e teste), usando as métricas RMSE e MAE.	6
3	Previsão Multi-Step	7
3.1	Implemente o processo de previsão multi-step	7
3.2	Faça a previsão multi-step para o horizonte de previsão igual a 12 e compare com o resultado da previsão one-step.	7
3.3	Modifique o tamanho da janela de entrada e avalie os resultados	7
3.4	Modifique a topologia da rede para obter um melhor desempenho. Altere seus parâmetros (e.g. número de processadores na camada escondida, tipo de função na camada de saída) e avalie o desempenho em termos das métricas RMSE e MAE.	7
3.5	Implemente a codificação ‘1 of N’ e use-a para modificar a representação da variável ‘mês’. Qual a mudança existente na arquitetura da Rede Neural? Avalie o desempenho em termos das métricas RMSE e MAE.	7

Resumo

Este documento consiste no relatório do trabalho 2 do módulo de Redes Neurais da disciplina ENG1456 da PUC-Rio. Nele será explicada a implementação de

modelos de Redes Neurais MLP para a previsão de uma série temporal do dataset MicroClima2 , disponibilizado pela professora da disciplina. A seções do relatório são definidas de acordo com as perguntas principais que constam no arquivo Guia de Atividades II. Foram consultados os materiais de aula, o livro [1] e outros materiais devidamente referenciados.

1 Compreensão do Problema

1.1 Visualize, em forma de gráfico, a dinâmica temporal da série escolhida. A série é adequada para a modelagem usando Redes Neurais? Caso não seja, que técnicas podem ser aplicadas para ajustar o comportamento da série?

A série utilizada neste trabalho, denominada *microclima2* consiste em 144 temperaturas média mensais, ou seja, temos informação sobre as temperaturas dos últimos 12 anos.

Com o intuito de analisar a dinâmica temporal da série, foi gerado o gráfico da série em si e da decomposição dela, com o intuito de verificar *trending* e *sazonalidade*. As figuras abaixo ilustram esses resultados.

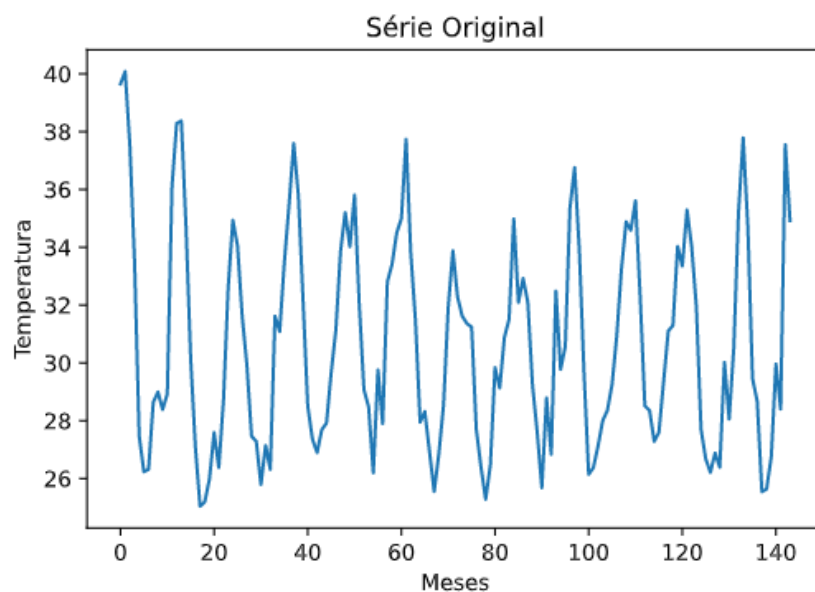


Figura 1: Série Original

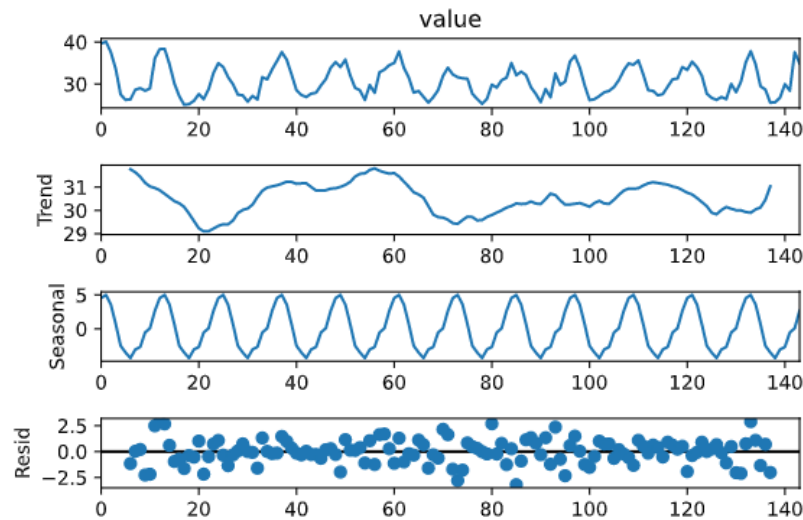


Figura 2: Decomposição da Série Original

Ao analisar a figura 1 notamos que a série possui um perfil estacionário ao longo dos 12 anos. Isso quer dizer que a temperatura média de cada mês do ano a é similar à temperatura média de cada mês no ano $a+1$. Além disso, a figura 2 revela, em seu campo *Trend*, a tendência da série ao longo do tempo. Embora existam oscilações nesse gráfico, não se percebe nenhuma tendência geral da série. Com o intuito de ir além da análise visual, foi executado o *Augmented Dickey-Fuller test* para verificar a probabilidade de existência de uma raiz unitária, que indicaria perfil não estacionário. O resultado é expresso na figura abaixo.

```

ADF Statistic: -3.340082
p-value: 0.013179
Critical Values:
    1%: -3.482
    5%: -2.884
   10%: -2.579

```

Figura 3: ADF Test

Nota-se que o *p-value* é muito pequeno e que o valor de *ADF Statistic* é menor que o valor crítico para 1%. Isso indica que esta série possui probabilidade baixíssima a alto grau de confiança em relação à inexistência de uma raiz unitária. Sendo assim, podemos tratá-la como uma série estacionária. Essa análise é importante pois, caso a série fosse não-estacionária, a rede neural precisaria, além de aprender o andamento temporal da série, aprender também a sua tendência, o que aumentaria a complexidade do modelo. Para corrigir esse problema, poderíamos tornar a série estacionária por meio de uma diferenciação, por exemplo, lembrando, apenas, de voltar aos valores originais ao final.

Como a série em questão possui valores de temperaturas médias mensais ao longo dos anos, é de se esperar que exista uma sazonalidade razoavelmente perceptível. Ambas as

figuras 1 e 2 mostram que essa sazonalidade existe de fato. Com essas imagens em mãos e supondo que o mês 1.0 é Janeiro, podemos inferir que as temperaturas referem-se a um local do hemisfério sul, onde é verão em janeiro, uma vez que as maiores temperaturas encontram-se próximas desse mês. Essa informação (o mês da temperatura) será útil para o treinamento da série, portanto deve constar nos dados de entrada. Como a sazonalidade não oferece problemas para a modelagem com um *MLP*, não precisamos fazer nenhum tipo de correção.

1.2 No problema escolhido, usaremos uma variável exógena que representa o mês de previsão (i.e. no instante $t+1$). De que forma esta variável pode auxiliar na previsão da série temporal?

Como comentado na seção 1.1, a série de temperaturas apresenta sazonalidade anual, isto é, o perfil de evolução da série se repete de ano em ano, o que é de se esperar dada a natureza da série. Desse modo, fornecer o mês do valor de entrada pode auxiliar bastante na qualidade da previsão da série, uma vez que meses como Dezembro a Fevereiro (12 a 3) geralmente apresentam temperaturas médias mais altas, enquanto meses como Maio a Setembro (5 a 9) apresentam temperaturas mais baixas. Ao fornecer essa informação para Rede Neural, ela possuirá mais informações para aprender o perfil de sazonalidade da série, aumentando sua qualidade de previsão.

2 Previsão One-Step

2.1 Execute o script para a previsão one-step. Analise o resultado (conjunto de treinamento e teste), usando as métricas RMSE e MAE.

Para realizar a previsão de um passo à frente, foram testadas diversas configurações de redes neurais, variando a quantidade de neurônios na camada escondida. Os erros *RMSE* e *MAE* de cada uma dessas configurações estão apresentados na tabela a seguir. Vale comentar que, no arquivo *jupyter notebook* enviado junto deste relatório está presente apenas o modelo final escolhido. Além disso, o script definia a métrica *MSE*, então, para obter a métrica desejada, *RMSE* foi calculada a raiz quadrada da *MSE* fornecida.

# neurônios	5	10	15	20	25	30
RMSE	3.007	2.871	3.022	2.584	2.598	2.936
MAE	2.090	2.105	2.434	1.938	1.968	2.343

Tabela 1: Comparação dos Erros para diferentes números de neurônios

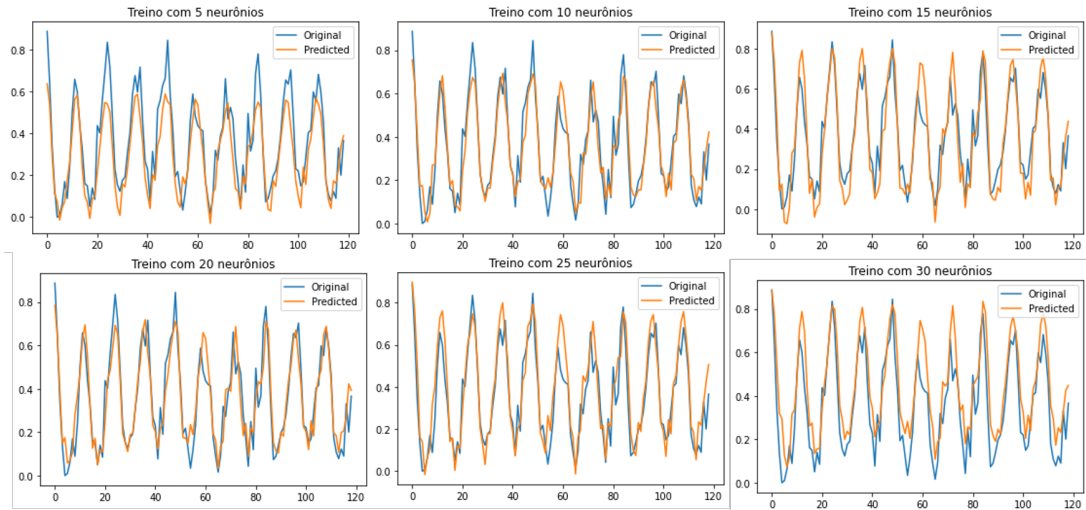


Figura 4: Comparação de Treinamento para diferentes quantidades de neurônios

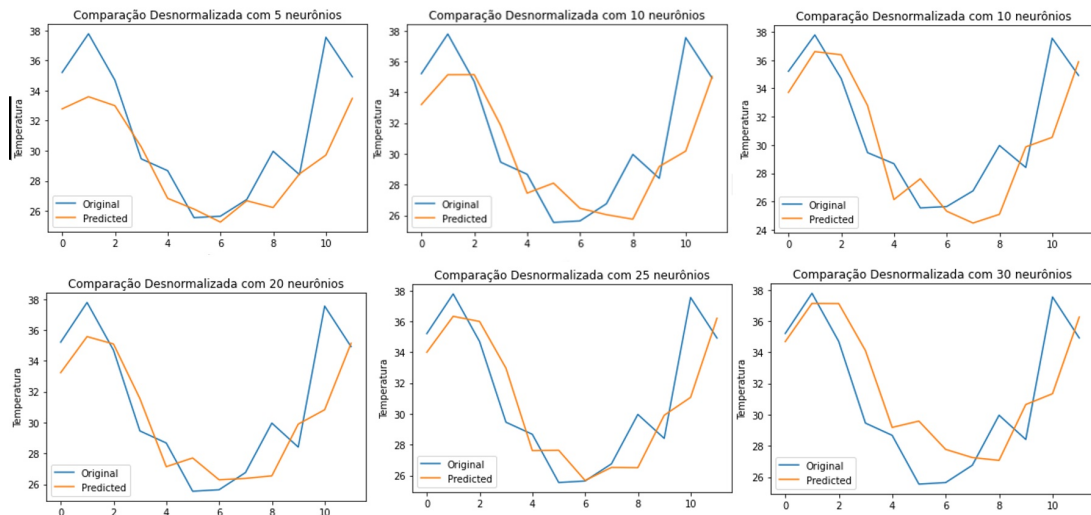


Figura 5: Comparação de testes para diferentes quantidades de neurônios

Com base na análise da tabela 2.1 e das imagens 4 e 5, percebe-se que para valores de 20 e 25 neurônios os erros são muito similares. Sendo assim, foi optado por eleger a rede com 20 neurônios na camada escondida por apresentar os melhores valores para as métricas analisadas e ser um modelo mais simples do que o de 25 neurônios.

Podemos perceber perfeitamente, com as imagens acima, a importância da generalização e da especialização da rede. Note que para 5 neurônios a curva de treino e de teste é muito suave, o que indica que a rede não conseguiu aprender o suficiente por possuir poucos pesos treináveis. Ou seja, não adquiriu especialização. Por outro lado, com 30 neurônio a curva passa a ser desnecessariamente ruidosa devido à quantidade grande de parâmetros treináveis, causando perda de generalização.

A figura abaixo exhibe as métricas da rede escolhida.

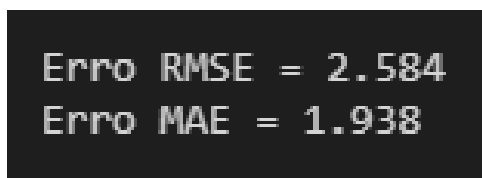


Figura 6: Erros da Rede Neural escolhida com 20 neurônios

2.2 Modifique a técnica de codificação mensal de ‘numérico’ para ‘binário’. Qual a mudança existente na arquitetura da Rede Neural? Analise o resultado (conjunto de treinamento e teste), usando as métricas RMSE e MAE.

As redes treinadas até então já estavam com a codificação mensal binária. Sendo assim, foi criado um modelo com exatamente os mesmos parâmetros da rede escolhida na seção 2.1 mas com a codificação numérica para os meses. Ao treinar essa nova rede pela primeira vez, os valores para as métricas *RMSE* e *MAE* foram melhores que aqueles apresentados pela rede com codificação binária. Para ter maior certeza dessa melhora, foram treinadas 6 redes com as mesmas configurações, variando, obviamente, apenas os pesos iniciais, que são aleatórios. Os erros para as 6 redes encontram-se na tabela abaixo. A primeira coluna é a referência da rede com configuração binária.

#Rede	Binária	1	2	3	4	5	6
RMSE	2.584	2.448	2.276	2.572	2.386	2.587	2.827
MAE	1.938	1.751	1.741	1.910	1.842	1.871	2.104

Tabela 2: Comparação dos erros das redes com codificação binária e numérica

Ao analisar a tabela 2.2, não é possível perceber grande melhora, ou piora, no desempenho da rede neural em relação ao tipo de codificação utilizado. Tentemos encontrar uma explicação para essa consistência.

Como visto na seção 1.1, a série em questão é bem comportada, inclusive em relação ao padrão de sazonalidade. O fato do tipo de codificação da entrada “mês” não ter influenciado na qualidade da previsão pode ser explicado, justamente, por esse bom comportamento. Independente do tipo de codificação a Rede parece já possuir informações suficientes para fazer a previsão com uma boa qualidade. Além disso, o tipo de codificação também altera a quantidade de entradas da rede neural. No caso numérico há apenas 1 entrada para indicar o mês, enquanto no caso binário são 4 entradas. Essa diferença aumenta o número de pesos a serem treinados e aumenta a complexidade da rede, podendo gerar perda de generalização. No entanto, uma vez que as métricas obtidas não foram muito distintas, isso não ocorreu.

3 Previsão Multi-Step

- 3.1 Implemente o processo de previsão multi-step
- 3.2 Faça a previsão multi-step para o horizonte de previsão igual a 12 e compare com o resultado da previsão one-step.
- 3.3 Modifique o tamanho da janela de entrada e avalie os resultados
- 3.4 Modifique a topologia da rede para obter um melhor desempenho. Altere seus parâmetros (e.g. número de processadores na camada escondida, tipo de função na camada de saída) e avalie o desempenho em termos das métricas RMSE e MAE.
- 3.5 Implemente a codificação ‘1 of N’ e use-a para modificar a representação da variável ‘mês’. Qual a mudança existente na arquitetura da Rede Neural? Avalie o desempenho em termos das métricas RMSE e MAE.

Referências

- [1] Ivan Nunes da Silva, Danilo Hernane Spatti, Rogerio Andrade Flauzino, Luisa Helena Bartocci Liboni, and Silas Franco dos Reis Alves. *Artificial Neural Networks*. Springer, 08 2016.