



UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE ENERGIAS ALTERNATIVAS E RENOVÁVEIS
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
AUTOMAÇÃO INTELIGENTE



JOSEÍTO DE OLIVEIRA JÚNIOR

INTELIGENCIA ARTIFICIAL NA INDUSTRIA
PROPOSTAS DE MODELOS DE SOFT SENSORS APLICADOS
NA MEDIÇÃO DE VAZÃO EM UMA INDUSTRIA
SUCROALCOOLEIRA

JOÃO PESSOA
26 de Abril de 2021

Joseíto de Oliveira Júnior

INTELIGENCIA ARTIFICIAL NA INDUSTRIA
PROPOSTAS DE MODELOS DE SOFT SENSORS APLICADOS NA MEDIÇÃO
DE VAZÃO EM UMA INDUSTRIA SUCROALCOOLEIRA

Relatório referente ao experimento da disciplina Automação Inteligente, curso de Engenharia Elétrica da Universidade Federal da Paraíba.

Prof. Dr. JUAN MOISES MAURICIO VILLANUEVA

JOÃO PESSOA
26 de Abril de 2021

Lista de Figuras

1	Diagrama da planta.	3
2	Dados	5
3	Dados com anomalias interpoladas.	6
4	Dados após a filtragem.	6
5	Dados após a normalização	7
6	Diagrama do modelo 1.	8
7	Q2 e predição do modelo 1.	8
8	Diagrama do modelo 2.	9
9	Q2 e predição do modelo 2.	10
10	Zoom da figura (9).	11
11	Diagrama do modelo 3.	11
12	Q2 e predição do modelo 3.	12

Sumário

1	Descrição do problema	3
2	Soft sensor	4
3	Desenvolvimento	4
3.1	Dados	4
3.2	Pre-processamento	5
3.3	Normalização	7
3.4	Divisão dos dados	7
3.5	Modelo 1	7
3.6	Modelo 2	9
3.6.1	Análise e discussão dos resultados do modelo 2	10
3.7	Modelo 3	11
4	Conclusão	12

1 Descrição do problema

O objetivo deste trabalho é desenvolver um softsensor para medição indireta da vazão de um tanque na indústria sucroalcooleira.

A medição direta da vazão possui algumas problemáticas, como incerteza e necessidade de manutenção, além de outros como:

- Transdutor de Vazão Eletromagnético CONAUT (m³/s)
- A tubulação deve estar pressurizada para garantir que a medição seja correta
- Calibração a cada 2 anos (Incerteza= +/-?)
- Efeito de Ruídos e *Outliers*
- Alta incerteza da medição
- Tubulação de aço 1020
- Diâmetro da tubulação de 12 "

Na figura (1) é representada a planta do sistema, a variável de interesse é a da vazão Q_2 .

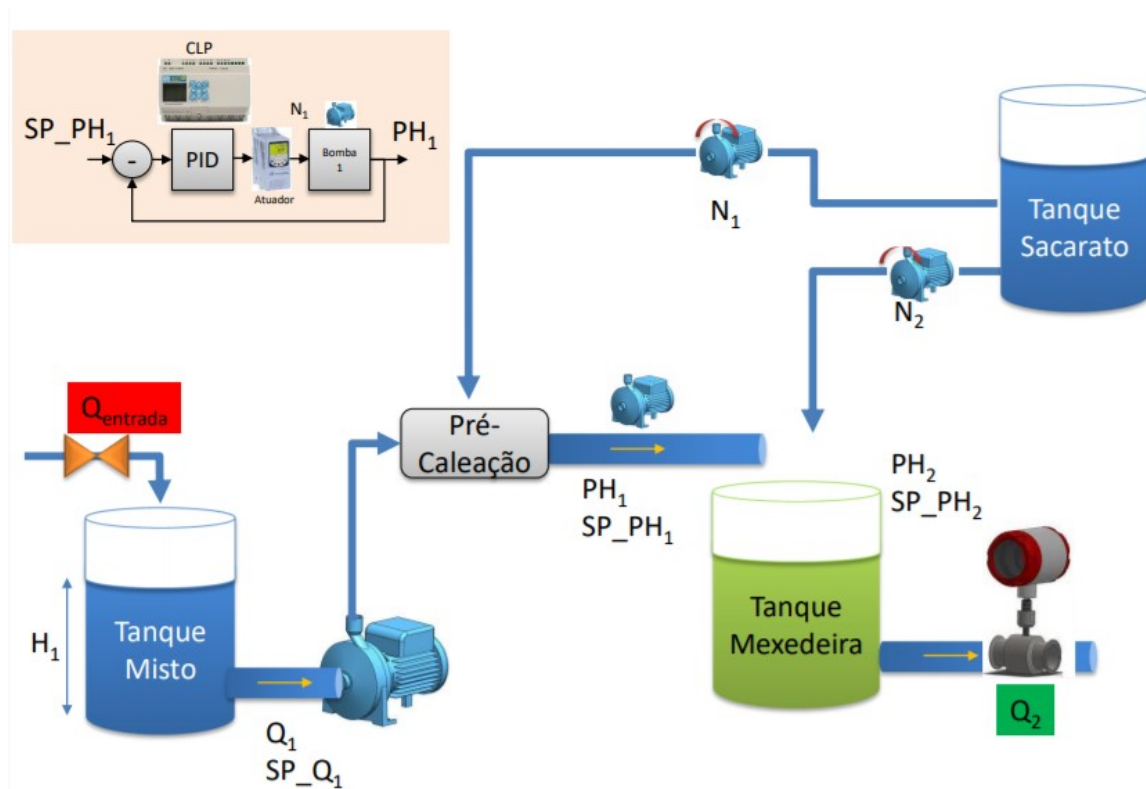


Figura 1: Diagrama da planta.

2 Soft sensor

Soft sensor ou sensor virtual é um nome comum para um software em que várias medições são processadas juntas. Geralmente *soft sensors* são baseados na teoria de controle e também recebem o nome de observador de estado. Pode haver dezenas ou até centenas de variáveis de entrada. A interação dos sinais pode ser usada para calcular novas quantidades que não precisam ser medidas. *Soft sensor* são especialmente úteis na fusão de dados, onde medições de diferentes características e dinâmicas são combinadas. Ele pode ser usado para diagnóstico de falhas e também para aplicações de controle.[5]

Algoritmos de software bem conhecidos que podem ser vistos como *soft sensors* incluem, por exemplo, filtros de Kalman. Implementações mais recentes de *soft sensors* usam redes neurais ou lógica fuzzy.[5]

3 Desenvolvimento

Os *soft sensors* propostos aqui são redes neurais artificiais profundas. O desenvolvimento dos modelos foi realizado com utilização da linguagem de programação Python [3] junto com algumas bibliotecas de inteligência artificial como Keras [6] e Scikit-learn [4], Pandas [2] para manipulação de dados e Numpy [1] para cálculos computacionais.

3.1 Dados

Os dados utilizados são de um conjunto de sensores com 8640 amostras obtidas com período de amostragem de 10 segundos, o que totalizam 24 horas de amostras. Entre as variáveis disponíveis temos as velocidades dos motores N1 e N2, o PH1, o PH2, a altura do tanque misto H1, vazão do tanque misto Q1 e vazão do tanque mexedeira Q2. Na figura (2) é representado parte desse conjunto de dados, as variáveis de PH foram omitidas e não utilizadas por não apresentarem variações significativas.

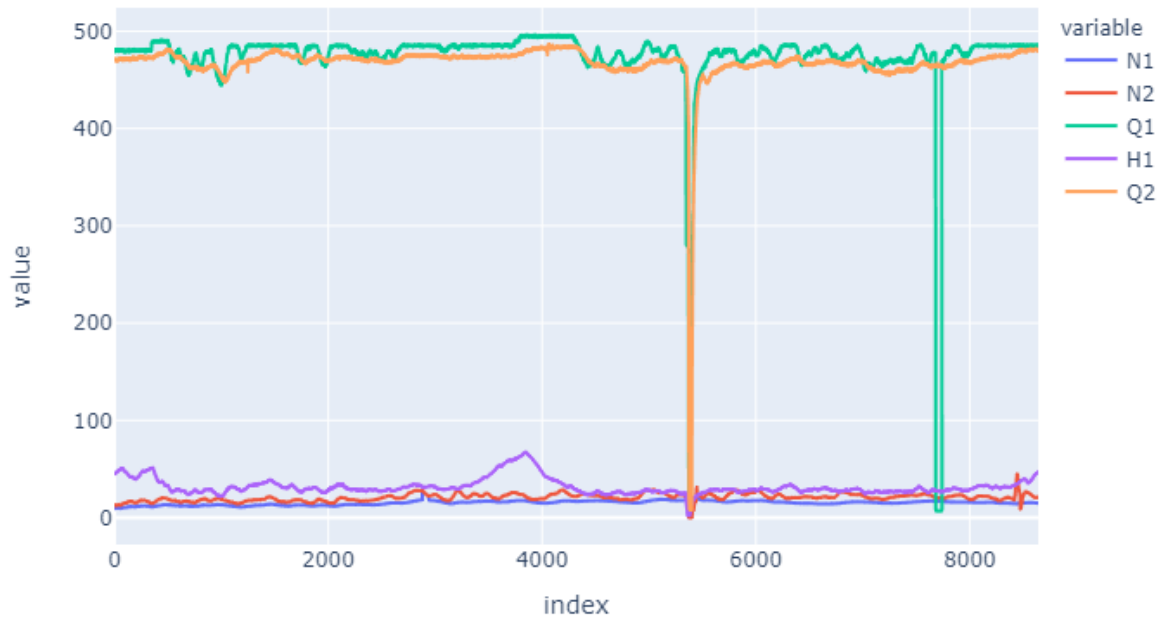


Figura 2: Dados

3.2 Pre-processamento

É importante que haja um pre-processamento dos dados para melhorar a qualidade dos dados utilizados no aprendizado dos modelos, nos dados originais, presentes na figura (2), podemos ver a presença de *outlier* na variável Q1 próximo ao índice 8000 e por volta do índice 5000 várias variáveis apresentam um afundamento que possivelmente representa um desligamento do sistema ou queda de energia, é interessante remover essas duas anomalias e para isso foi utilizado interpolação do tipo *spline* de ordem 3. O resultado pode ser visto na figura (3).

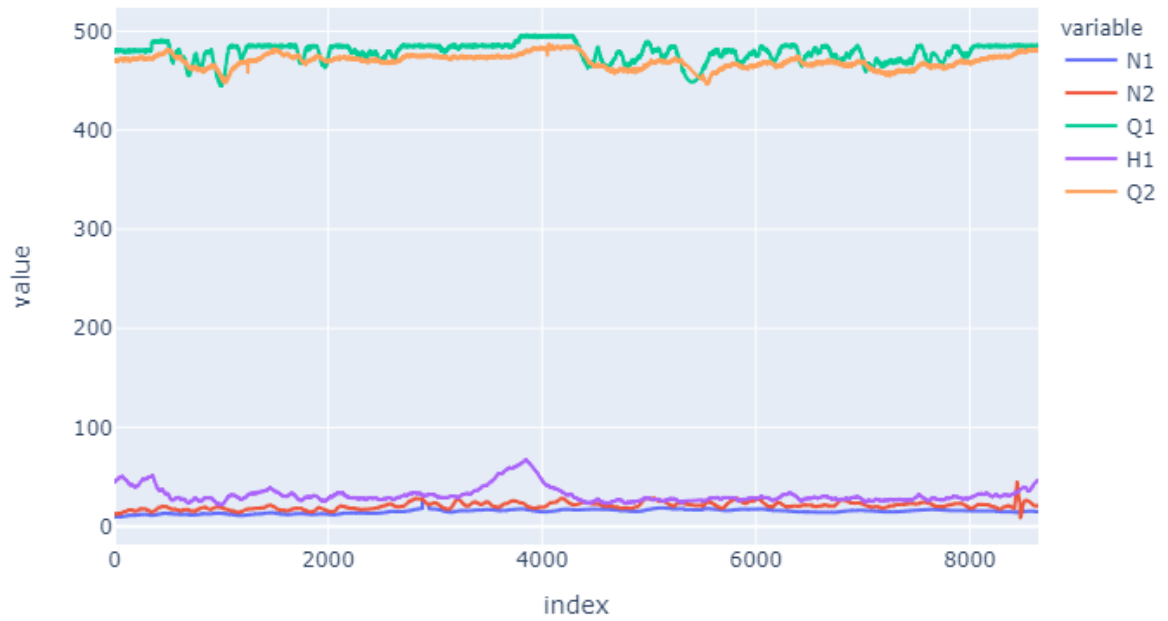


Figura 3: Dados com anomalias interpoladas.

Um filtro digital do tipo FIR de ordem 20 foi aplicado aos dados para reduzir o ruído, o resultado pode ser visto na figura (4).

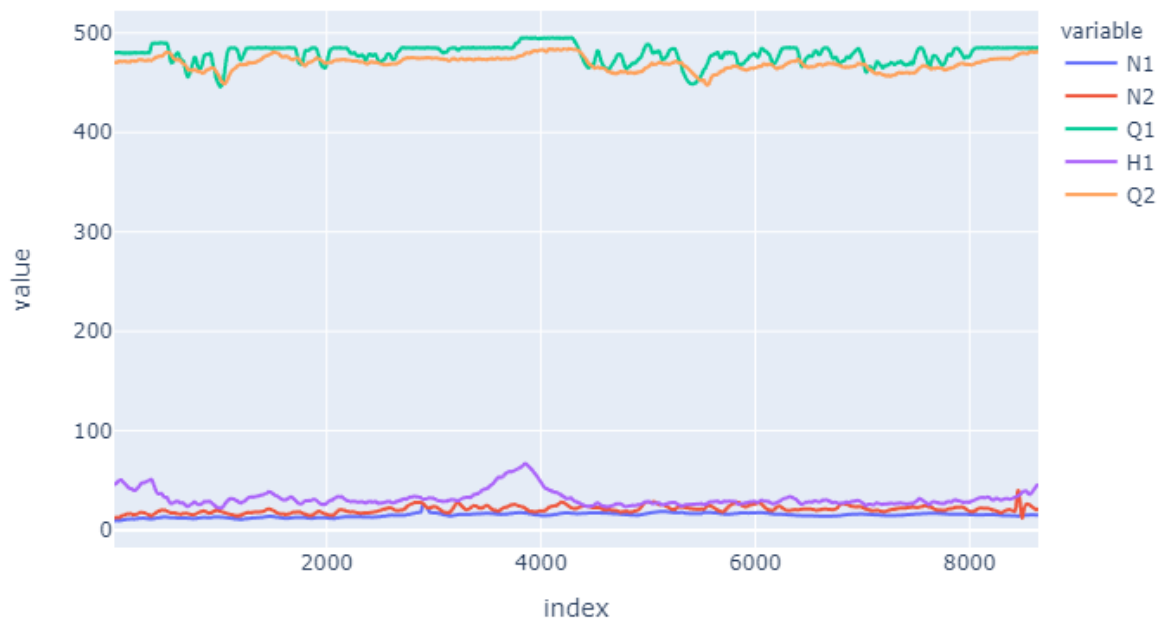


Figura 4: Dados após a filtragem.

3.3 Normalização

Devido as características de implementação dos modelos de RNA, é necessário que haja uma normalização dos dados antes que estes sejam utilizados no aprendizado propriamente dito, de modo que a magnitude das variáveis esteja limitada ao intervalo fechado 0 e 1, podendo possuir qualquer valor entre esses limites. Os resultados da normalização são exibidos na figura (5).



Figura 5: Dados após a normalização

3.4 Divisão dos dados

Para poder analisar o modelo desenvolvido, os dados são divididos em dois conjuntos onde um é utilizado para o treinamento e o outro para a validação, com isso é possível perceber o desempenho do modelo em dados inéditos. O conjunto de treino possui 75% do total das amostras agrupadas aleatoriamente e o conjunto de teste é composto pelos 25% complementares. Essas porcentagens da divisão foram escolhidos arbitrariamente e outros valores poderiam ter sido utilizados.

3.5 Modelo 1

O primeiro modelo proposto possui apenas 4 variáveis como entradas, sendo elas N1, N2, H1, e Q1 e possui 5 camadas, sendo 3 ocultas. As 4 camadas iniciais possuem função de ativação tangente hiperbólica e a ultima apresenta a função sigmoide para ativação. A função tangente hiperbólica foi escolhida por apresentar mais chances de convergências nos testes utilizados e sigmoide foi utilizada na última camada por possuir um valor limite entre 0 e 1, o que limita a saída da rede e evita *outlier*. As camadas deste modelo

possuem 4, 20, 15, 5 e 1 neurônios respectivamente. Na figura (6) o diagrama do modelo 1 é mostrado.

O otimizador utilizado foi o adam e a função perda foi a de erro logarítmico médio quadrático.

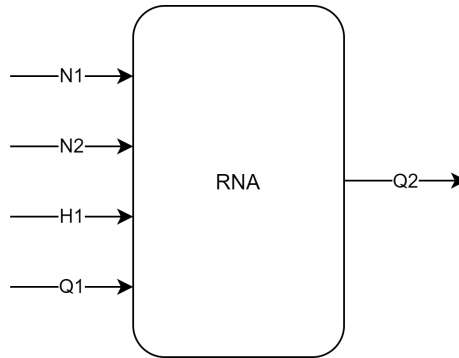


Figura 6: Diagrama do modelo 1.

Para evitar *overfitting*, em cada época do treinamento o modelo é testado com o conjunto de teste, e quando há uma estagnação na função perda ou esse valor começar a aumentar o treinamento é interrompido.

Após esse processo, o modelo foi testado com os dados originais apenas filtrados com um filtro FIR de ordem 10. Na figura (7) é mostrado o valor Q2 medido e o valor de Q2 estimado pelo modelo.

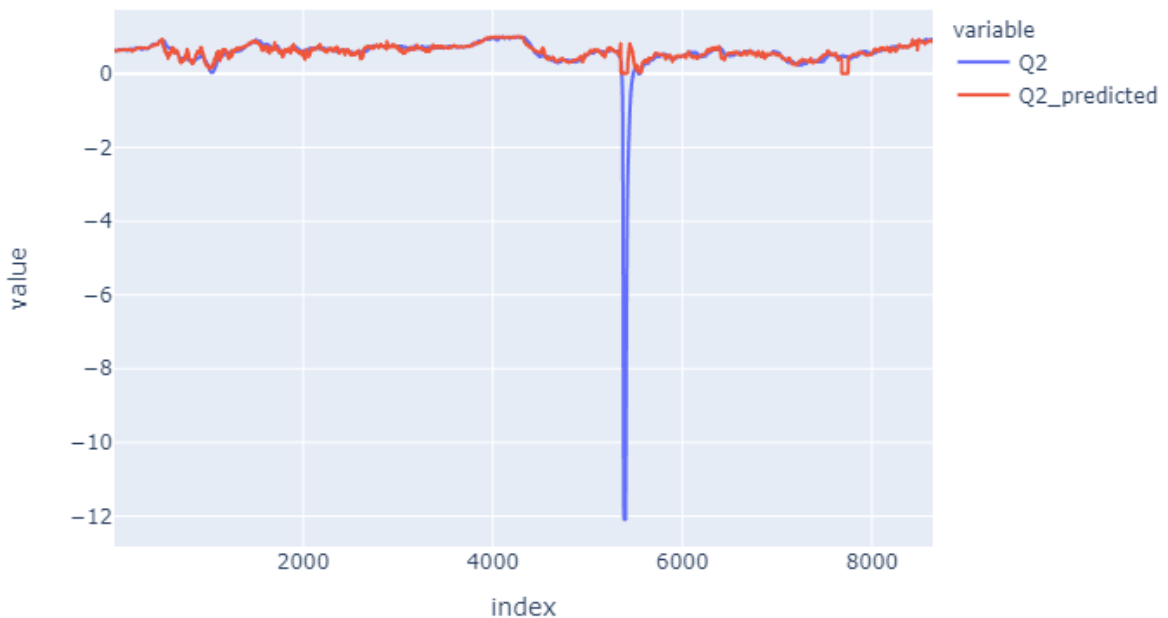


Figura 7: Q2 e predição do modelo 1.

3.6 Modelo 2

O primeiro modelo proposto possui apenas 9 variáveis como entradas, sendo elas N1, N2, H1, Q1, N1 atrasado de uma amostra, N2 atrasado de uma amostra, H1 atrasado de uma amostra, Q1 atrasado de uma amostra e Q2 atrasado de uma amostra e possui 5 camadas, sendo 3 ocultas. As 4 camadas iniciais possuem função de ativação tangente hiperbólica e a última apresenta a função sigmoide para ativação. A função tangente hiperbólica foi escolhida por apresentar mais chances de convergências nos testes utilizados e sigmoide foi utilizada na última camada por possuir um valor limite entre 0 e 1, o que limita a saída da rede e evita *outlier*. As camadas deste modelo possuem 9, 20, 30, 20 e 1 neurônios respectivamente. Na figura (8) o diagrama do modelo 2 é mostrado.

O otimizador utilizado foi o adam e a função perda foi a de erro logarítmico médio quadrático.

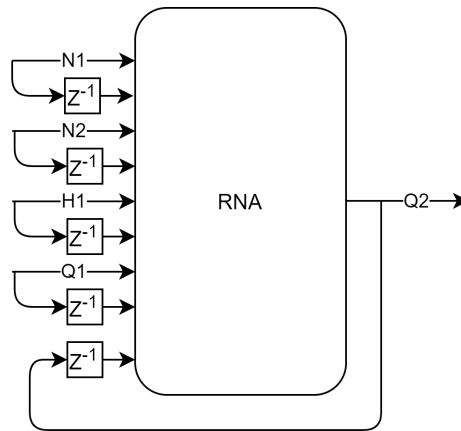


Figura 8: Diagrama do modelo 2.

De forma similar ao modelo 1, para evitar *overfitting*, em cada época do treinamento o modelo é testado com o conjunto de teste, e quando há uma estagnação na função perda ou esse valor começar a aumentar o treinamento é interrompido.

Após esse processo, o modelo foi testado com os dados originais apenas filtrados com um filtro FIR de ordem 10. Na figura (9) é mostrado o valor Q2 medido e o valor de Q2 estimado pelo modelo.

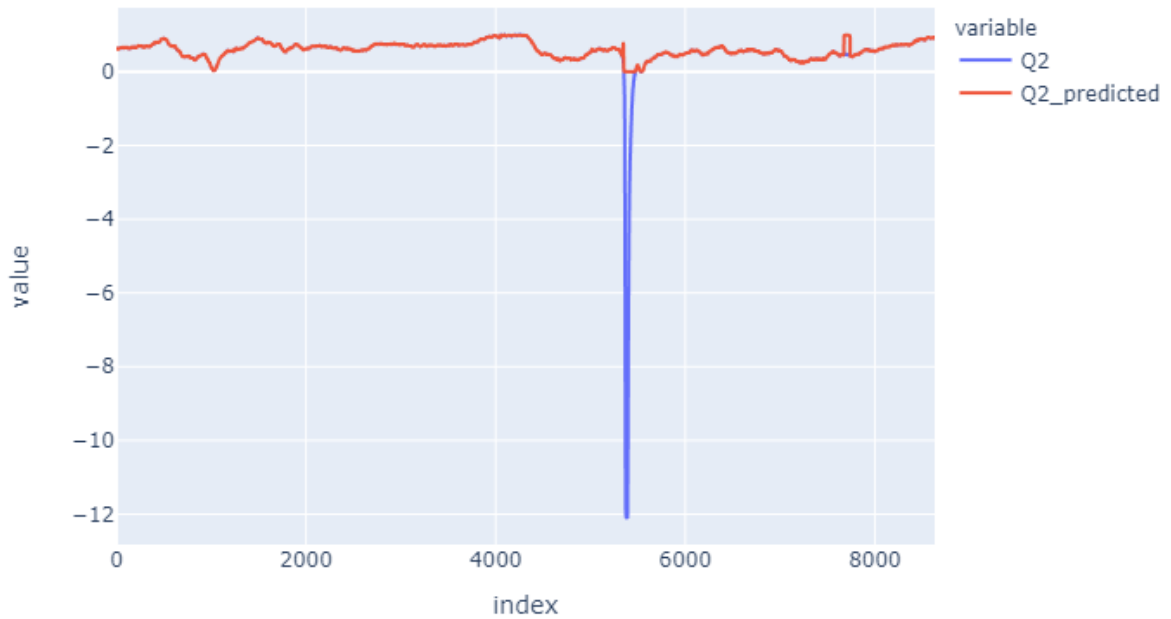


Figura 9: Q2 e predição do modelo 2.

3.6.1 Análise e discussão dos resultados do modelo 2

O modelo 2 apresentou um desempenho ótimo, mas há um problema associado que precisa ser observado, na figura (10) é mostrado um zoom da figura (9) e podemos ver que Q2 predito pela RNA é muito próximo ou igual ao valor anterior de Q2 que foi fornecido na entrada e de fato a planta apresenta pouca variação entre uma amostra e outro de modo que esse comportamento apresenta um valor perda muito baixo.

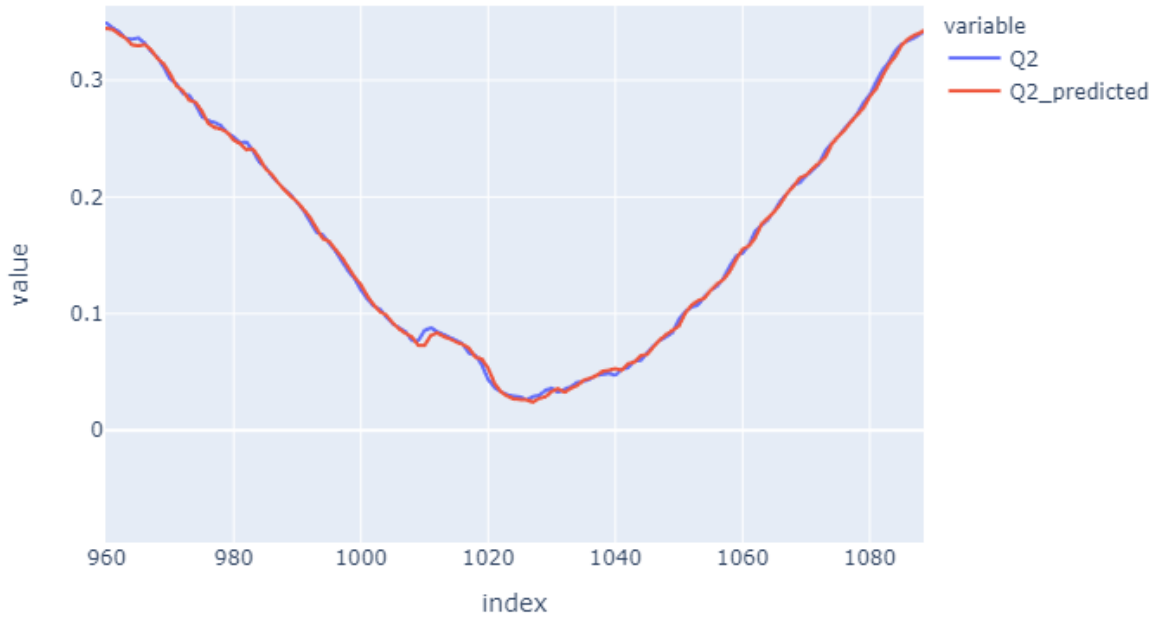


Figura 10: Zoom da figura (9).

Devido a necessidade de possuir o valor anterior de $Q2$, um sensor para medir essa grandeza ainda é necessário e todos os problemas associados levantados na seção (1) não são resolvidos.

3.7 Modelo 3

Um terceiro modelo é sugerido idêntico ao modelo 2, porem distinguindo-se apenas nas entradas, onde a variável $Q2$ atrasada não é utilizada e as demais variáveis atrasadas são atrasadas em 10 ciclos, na figura (11) é mostrado o diagrama do modelo 3.

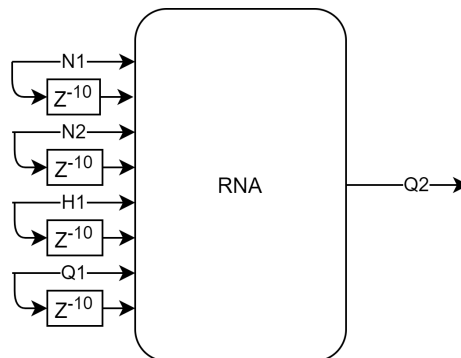


Figura 11: Diagrama do modelo 3.

De forma similar aos modelos anteriores, para evitar overfitting, em cada época do treinamento o modelo é testado com o conjunto de teste, e quando há uma estagnação na função perda ou esse valor começar a aumentar o treinamento é interrompido. Após esse processo, o modelo foi testado com os dados originais apenas filtrados com um filtro FIR

de ordem 10. Na figura (12) é mostrado o valor Q2 medido e o valor de Q2 estimado pelo modelo.

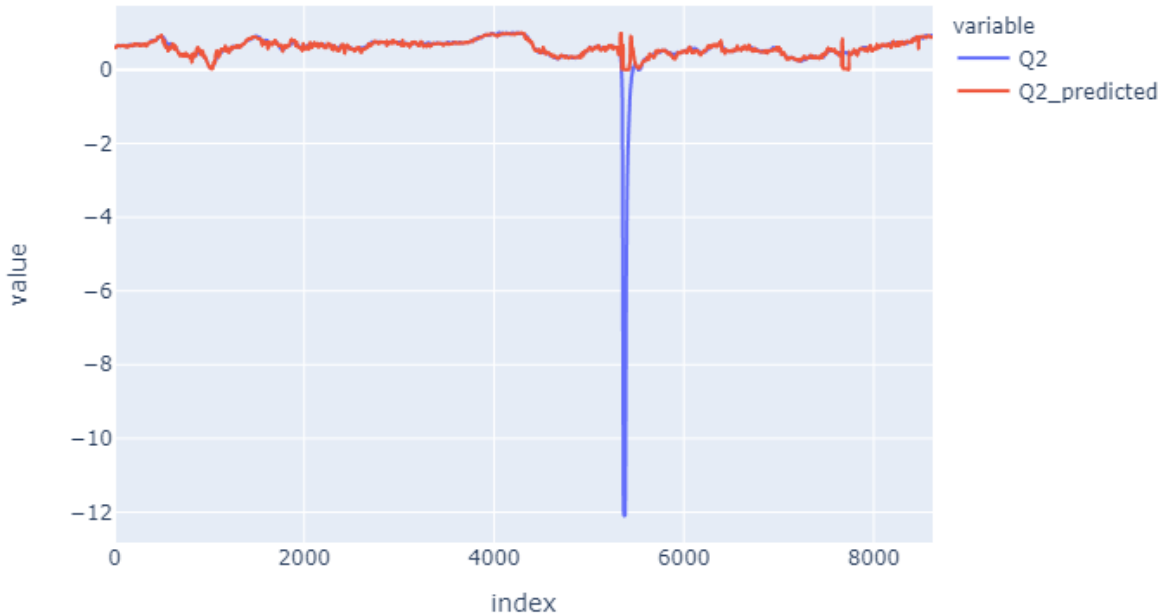


Figura 12: Q2 e predição do modelo 3.

4 Conclusão

Nesse trabalho foram propostos 3 modelos de *soft sensors* baseados em redes neurais artificiais profundas para medição de vazão em uma indústria sucroalcooleira, os dados foram tratados, normalizados e divididos.

O primeiro modelo apresentou desempenho satisfatório, porem com alto ruído e uma incerteza considerável. O segundo modelo apresentou o melhor desempenho mas utiliza uma amostras anteriores dos sinais incluindo a medição da vazão anterior, o que faz necessário a existência de um sensor e os problemas apresentados na seção 1 não são resolvidos.

Outra característica do modelo 2 observada foi a quase cópia direta da entrada Q2 atrasada diretamente para a saída, o que pode não justificar o custo computacional do modelo.

Por fim o modelo 3 apresentou um desempenho superior ao modelo 1 e inferior ao modelo 2, porem não utiliza leituras anteriores da variável predita, o que possibilita a retirada do sensor físico.

Sugere-se aqui a aplicação de mais dados para o treino antes de implementar um modelo definitivo na industria, visto que um dia de dados não representa em totalidade os possíveis cenários de operação da planta e uma maior quantidade de dados de treino ajuda na generalização do modelo.

Referências

- [1] URL <https://numpy.org/>. page.44
- [2] URL <https://pandas.pydata.org/>. page.44
- [3] Python 3.9.4 documentation. URL <https://docs.python.org/3/>. page.44
- [4] Scikit-learn. URL <https://scikit-learn.org/stable/>. page.44
- [5] Soft sensor, Sep 2020. URL https://en.wikipedia.org/wiki/Soft_sensor. page.44
- [6] Keras. Simple. flexible. powerful. URL <https://keras.io/>. page.44