



UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA QUÍMICA
ENG07053 - TRABALHO DE DIPLOMAÇÃO EM ENGENHARIA
QUÍMICA



Desenvolvimento de Analisador Virtual para Predição da Pressão de Fundo em Poços de Petróleo Utilizando Rede Neural

Autor: Norton Faitão Farias

Orientador: Marcelo Farenzena

Coorientador: Jônathan Dambros

Porto Alegre, dezembro de 18

Sumário

Sumário	ii
Agradecimentos	iii
Resumo	iv
Lista de Figuras	v
Lista de Tabelas	vi
1 Introdução	1
2 Revisão Bibliográfica	3
2.1 Extração e produção de petróleo	3
2.1.1 Sistema de extração offshore	3
2.1.2 Escoamento multifásico e golfadas	3
2.1.3 Sensores para controle	5
2.2 Analisadores Virtuais	5
2.3 Modelo de rede neural	9
2.3.1 Hiperparâmetros	2
2.4 Método de regressão linear	5
2.5 Estruturas de redes neurais aplicadas à predição da Ppdg em sistemas <i>offshore</i>	6
3 Materiais e Métodos	7
3.1 Desenvolvimento do <i>Soft Sensor</i>	7
3.1.1 Primeira inspeção de dados seleção de dados históricos	7
3.1.2 Pré-tratamento dos dados	8
3.1.3 Seleção, treinamento e validação do modelo	9
4 Resultados	10
5 Conclusões e Trabalhos Futuros	15
6 Referências	16

Agradecimentos

À universidade UFRGS e seu corpo docente que me oportunizaram a estrutura base para meu aprendizado.

Ao meu orientador, pelos ensinamentos, sugestões e empenho dedicado à elaboração deste trabalho.

Ao meu co-orientador, que mesmo a distância e com fuso horário diferente pode me auxiliar, dar suporte e incentivar muito.

Obrigado a minha namorada, que mesmo com momentos em que eu estava ausente devido aos estudos, sempre me apoiou e me animou muito nas situações de dificuldade.

Aos meus pais, pelo amor, incentivo, suporte nos momentos de cansaço e apoio incondicional.

Meus agradecimentos aos amigos Daniela, Gabriele e Felipe, companheiros de trabalhos, irmãos na amizade, que fizeram toda a diferença na minha formação e que irão continuar sempre presentes em minha vida.

Resumo

Na exploração de petróleo a melhor variável para estabilização do fluxo via golfadas é a Pressão no chamado de *Downhole Pressure Gauge* (Ppdg). Porém este instrumento é instalado em ambiente agressivo e, portanto tende a estragar facilmente. Além disso, devido sua localização a manutenção/substituição é normalmente inviável economicamente. Sendo assim, uma solução para garantir esta medida é utilizar como sensor um analisador virtual. Nesse contexto, utilizando de um modelo de rede neural *feedforward* do tipo caixa preta com dados do modelo dinâmico semi-empírico simplificado FOWM (*Fast Offshore Wells Model*), o objetivo do presente trabalho é desenvolver um analisador virtual capaz de prever a Ppdg com o mínimo erro associado. Obteve-se como resultado otimizado o modelo com os hiperparâmetros: otimizador = Adadelta, inicialização = *glorot_normal*, ativação = *relu*, N° neurônios na 1° camada = 256 e N° neurônios na 2° camada = 0, sendo o coeficiente de determinação (R^2) igual a 0,9998969 e um MSE de $5,02 \times 10^{-6}$. Indicando assim, que o mesmo é capaz de substituir em uma planta real o sensor físico. Além disso, pode-se observar que um modelo do tipo caixa preta exige que a base de dados contemple a maioria dos cenários que a planta está sujeita a operar, tendo em vista que a capacidade preditiva extrapolava do modelo foi muito inferior a capacidade interpolativa, sendo o coeficiente de determinação da extrapolação inicial igual a 0,659041 e final igual a 0,699827. Por fim, quando testada a mesma base de dados no método clássico de predição, a regressão linear, obteve-se como $R^2 = 0,995022$ e o $MSE = 2 \times 10^{-4}$. Indicando assim, que o método de rede neural é mais acurado para modelagem preditiva.

Lista de Figuras

Figura 1: Desenho esquemático de uma plataforma de petróleo offshore. Fonte: adaptado de Di Meglio et al. (2012).	3
Figura 2: Representação de fluxo multifásico horizontal. Fonte: adaptado de Smart e Smith (1997).	4
Figura 3: Representação esquemática de uma região suscetível ao fenômeno da golfada severa. Fonte: Ashikawa (2017).....	4
Figura 4: Fluxo de criação de um soft sensor. Fonte: adaptado de Kadlec et al. (2009).	7
Figura 5: Representação a da técnica de validação cruzada <i>k-fold cross-validation</i> , com $K = 4$. Fonte: adaptado de Bishop (2006).....	8
Figura 6: Representação de um neurônio artificial. Fonte: adaptado de Agatonovic-kustrin e Beresford (2000).	9
Figura 7: Representação de uma rede neural típica. Fonte: Adaptado de Deka et. al (2015)	2
Figura 8: Representação da técnica de parada antecipada. Fonte: adaptado de Géron (2017).	3
Figura 9: Representação da função <i>Sigmoid</i> . Fonte: Autoria própria.	3
Figura 10: Representação da função <i>Tanh</i> . Fonte: Autoria própria.....	3
Figura 11: Representação da função <i>ReLU</i> . Fonte: Autoria própria.	4
Figura 12: Representação método de otimização chamado de Gradiente Descendente, Fonte: adaptado de GÉRON, 2017.....	5
Figura 13: Variação da Ppdg pelo tempo.	8
Figura 14: Normalização da PPDG.	8
Figura 15: Avaliação dos algoritmos de otimização.	11
Figura 16: Avaliação dos métodos de inicialização.	11
Figura 17: Avaliação das Funções de ativação.	11
Figura 18: Avaliação do número de neurônios totais no modelo, onde no eixo X é a soma dos numero de neurônios na 1ª camada e numero de neurônios na 2ª camada além da representação dos mesmos (numero de neurônios na 1ª camada/ numero de neurônios na 2ª camada).....	12
Figura 19: Capacidade do modelo em predizer a Ppdg frente aos valores reais.....	13
Figura 20: Reta que representa a precisão do modelo frente aos valores reais.	13
Figura 21: Precisão preditiva extrapolativa do modelo.....	14

Lista de Tabelas

Tabela 1: Dados de entrada do simulador FOWM.....	7
Tabela 2: Hiperparâmetros utilizados no estudo.....	9
Tabela 3: Melhores combinações de hiperparâmetros do modelo.....	10

1 Introdução

Nos dias atuais o petróleo e o gás ainda continuam sendo as mais importantes fontes de energia em todo o mundo. Mesmo com o aumento dos investimentos no desenvolvimento de novas tecnologias que utilizem de novas fontes de energia, incluindo as renováveis, ainda assim o mundo ainda será dependente de petróleo e gás por muitos anos (BRAGA e CAMPOS, 2012). No Brasil, essa importância ficou ainda mais evidente com a descoberta de potenciais reservas de grande escala localizadas no Pré-Sal. Segundo registro da Petrobrás (2014) o segmento de petróleo e gás natural representam cerca de 13% no PIB nacional e tem perspectivas de dobrar a atual produção petróleo até 2020.

No início da exploração de um novo poço de petróleo, a própria pressão do reservatório é capaz de fazer com que a mistura multifásica de óleo, gás e água flua até a superfície. Entretanto com o passar do tempo com o esgotamento das reservas se faz necessária a injeção de gás com o objetivo de reduzir a massa específica da mistura para que seja mantido o mesmo fluxo de produção (AGUIRRE et. al, 2017).

Com isso, surgem as golfadas severas ou *slugging*. Esse fenômeno é caracterizado por irregularidades do fluxo multifásico durante o transporte do reservatório para as instalações de superfície. É de forte interesse econômico prever e preveni-lo, já que sua presença aumenta o risco operacional, reduz a produtividade e a vida útil das instalações (JAHANSHAH, 2013).

O monitoramento e o controle efetivo do processo são soluções viáveis para atingir esse objetivo. Para isso, existem diversos sensores disponíveis em plataformas, entretanto a pressão mensurada no chamado *Permanent Downhole Gauge* (Ppdg) é a medida que melhor representa a dinâmica dos poços de extração, principalmente por sua localização e por estar diretamente ligada à estabilidade da produção. Todavia essa se encontra em local de intenso desgaste e, portanto, tem vida útil curta (FROTA e DESTRO, 2006). Além disso, quando é possível realizar manutenção, essa não só é de difícil acesso, exigindo uma logística muito onerosa, mas também exige que seja parada a produção para troca do instrumento.

Nesse contexto, o objetivo principal do presente trabalho é desenvolver um analisador virtual capaz de prever a Ppdg utilizando redes neurais artificiais. Uma vantagem deste tipo de modelagem é que ela frequentemente pode ser desenvolvida em pouco tempo e utilizando dados disponíveis com um custo muito baixo quando comparado com o desenvolvimento de *conventional expert systems*. A economia de tempo e custo é obtida por se substituir o processo de aquisição de conhecimento do processo por treinamento de redes (HERTZ et al., 1991; ZURADA, 1992). Outra vantagem, é que as redes neurais podem se ajustar com dados históricos e prever novas situações. Portanto, podem ser treinadas para resolver um problema mesmo antes que os especialistas no processo sejam capazes de formular seus conhecimentos de forma organizada, completa e consistente (HERTZ et al., 1991; ZURADA, 1992).

Os objetivos específicos desse trabalho contemplam:

i) A comparação de diferentes hiperparâmetros da rede a fim de estimar o modelo otimizado. Já que estes parâmetros definem a capacidade da rede neural e, por meio deles, podem-se minimizar os erros associados às predições. Uma boa compreensão dos

efeitos que cada um deles tem sobre o treino de uma rede neural auxilia muito na elaboração de um modelo eficaz (GÉRON, 2017).

ii) Estimar a capacidade preditiva interpolativa e extrapolativa do mesmo para que se tenha conhecimento da necessidade de dados que a base deve conter.

iii) A comparação do método de predição baseado em rede neural com o método clássico de regressão linear, tendo em vista sua simplicidade, baixo custo operacional e tempo para seu desenvolvimento e execução.

Este trabalho está estruturado em 5 capítulos: o Capítulo 1 traz motivações e os objetivos do trabalho; o Capítulo 2, evidencia aspectos que serão importantes para o entendimento do trabalho, desde o funcionamento de um poço de extração de petróleo até o desenvolvimento de um analisador virtual; o capítulo 3, expõe como foi desenvolvido o *softsensor* e o como serão atingidos os objetivos; o capítulo 4, apresenta o melhor conjunto de hiperparâmetros do modelo, o comparativo entre estes, além de gráficos explicativos referentes a capacidade preditiva do resultado mais otimizado; e o capítulo 5 com as principais conclusões do trabalho.

2 Revisão Bibliográfica

Neste capítulo, serão apresentados os conceitos necessários para uma melhor compreensão do trabalho.

2.1 Extração e produção de petróleo

2.1.1 Sistema de extração offshore

Um processo tradicional de extração de petróleo *offshore*, como ilustrado na Figura 1, inicia-se com a elevação, através dos tubos de transporte, de uma mistura multifásica composta de petróleo, gás e água desde os reservatórios até as instalações do fundo do mar, pelos poços. A próxima etapa, na chamada “cabeça do poço” encontra-se a árvore de natal molhada (ANM), onde estão presentes os instrumentos de medida e as válvulas operadas remotamente, que controlam o fluxo dos fluidos produzidos ou injetados no poço visando a segurança do processo. Além disso, as ANM reúnem as produções de diversos poços e realizam o transporte das mesmas por vários quilômetros, através das linhas de produção (*flowlines*) na superfície do solo marinho. A elevação dos produtos até a plataforma é realizada por meio de dutos verticais denominados *risers* (DI MEGLIO et al., 2012). Por fim, na plataforma está presente a válvula *choke* que é responsável pelo controle do fluxo de produção visando proteger os equipamentos da superfície que tenham restrição de pressão e/ou vazão.

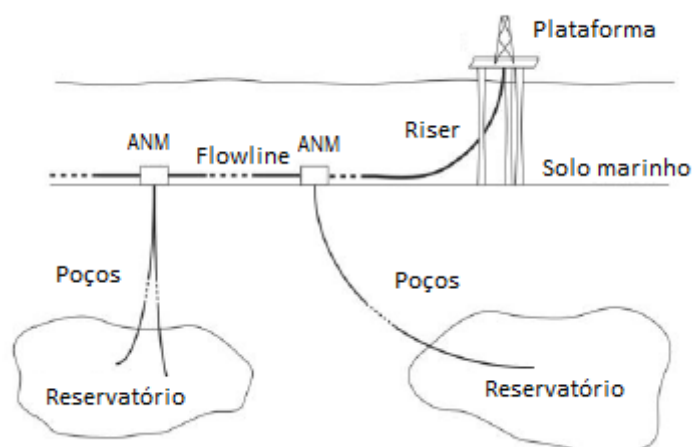


Figura 1: Desenho esquemático de uma plataforma de petróleo offshore. Fonte: adaptado de Di Meglio et al. (2012).

2.1.2 Escoamento multifásico e golfadas

Uma característica comum na produção de petróleo são as irregularidades de fluxo durante o transporte do reservatório para as instalações de superfície. Os principais problemas causados por essas instabilidades de fluxo estão ligados ao aumento do risco operacional, redução do sistema de produção, e dificuldades na condução dos poços para o funcionamento em condições ideais. Quando a produção ocorre em um ambiente do tipo *offshore*, os problemas de fluxo tornam-se ainda mais críticos (DIEHL et al. 2017).

Na produção em águas profundas e ultraprofundas, o *flowline* e o *riser* são compostos de tubos longos distribuídos em diferentes angulações. Essa distribuição, combinada à diferença de velocidade, viscosidade e densidade entre as fases gasosa e líquida pode

criar instabilidades no fluxo multifásico, denominadas golfadas (*slug flow*), resultando em riscos operacionais e redução da produção (SCHMIDT et al., 1979; FABRE et al., 1990; MEGLIO, 2011). Ainda, Di Maglio (2012) afirma que o regime de fluxo multifásico intermitente/oscilatório ocorre com maior frequência em poços de petróleo com maior tempo de exploração.

As golfadas são caracterizadas por bolsões de líquido (*slugs*) seguidos de bolsões de gás. Nos dutos horizontais (*flowline*) esse fenômeno, ilustrado na Figura 2, pode ocorrer devido à diferença de velocidade entre a fase líquida e a fase gasosa, e é denominado de golfada hidrodinâmica. Ainda, pela superfície irregular do leito marinho, denominado de golfada induzida pelo terreno, onde o líquido se acumula nas posições do duto com menor elevação e bloqueia a passagem de gás.

Já nos dutos verticais, os *risers*, quando associados a trechos horizontais subsequentes de inclinação descendente, situação representada na Figura 3, ou condições de baixa vazão de produção, ocorre à chamada golfada severa. Nesta, o líquido impede a passagem do gás que está no trecho inferior, até que este tenha pressão suficiente para vencer a coluna hidrostática (JAHANSHAH, 2013).

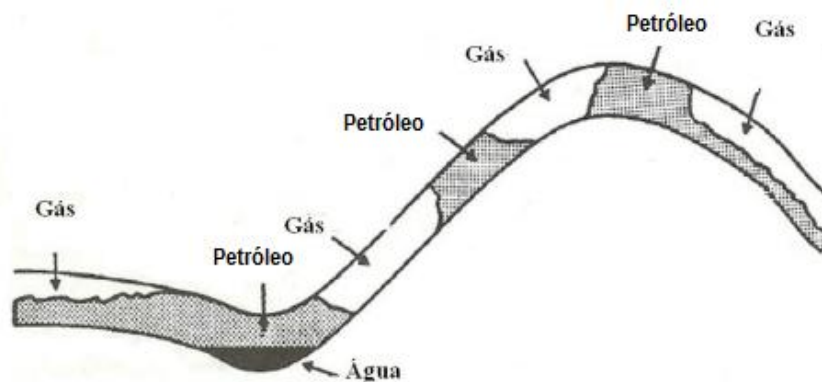


Figura 2: Representação de fluxo multifásico horizontal. Fonte: adaptado de Smart e Smith (1997).

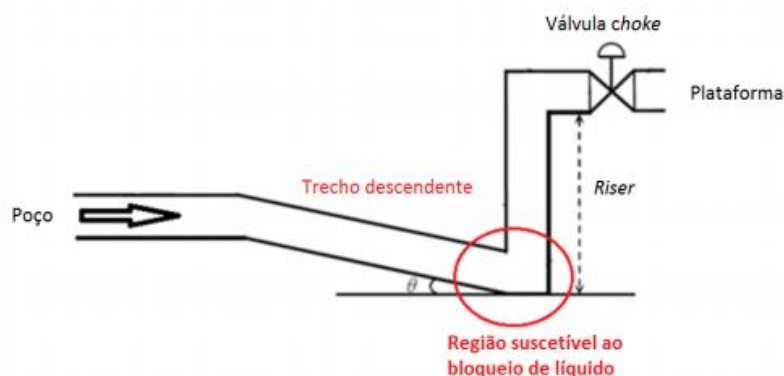


Figura 3: Representação esquemática de uma região suscetível ao fenômeno da golfada severa. Fonte: Ashikawa (2017).

Para reduzir tal problema, uma das soluções convencionais é reduzir a abertura da válvula *choke* de produção, porém essa atitude resulta na redução da taxa de produção do poço de petróleo. Outra possibilidade é aumentar a quantidade do gás injetado, entretanto isso acarreta em custos mais elevados (JAHANSHAH, 2012).

Melhorar essa operação é um ponto chave para otimizar o lucro do sistema de produção. Para alcançar este objetivo, os esforços devem ser concentrados em fornecer melhor operação do sistema de produção, aumentando o monitoramento, controle e conhecimento em tempo real para apoiar decisões rápidas sobre a operação. (DIEHL et al., 2017)

2.1.3 Sensores para controle

O primeiro a utilizar técnicas de controle foi Jansen et al. (1999), que fez uso de medições de pressão, vazão ou densidade do fluido como variáveis de controle, sendo a pressão do poço, denominada de *permanent downhole gauge* (PDG), a variável de controle mais importante. Porém, essa medição é de baixa confiabilidade e não está disponível em boa parte dos casos. Portanto, para solucionar esse infortúnio e otimizar o processo muitos estudos propõem sistemas predição baseados em analisadores virtuais (EIKREM et al., 2004; AAMO et al., 2005).

Para monitorar e controlar o sistema, dentre muitos sensores disponíveis em uma típica plataforma, um dos mais importantes é o sensor PDG, pois este tem uma localização que torna possível a obtenção de valiosos dados para a estabilização do fluxo via golfadas. Todavia, sua localização também resulta em muitos inconvenientes, como desgaste intenso, que ocasiona em uma curta vida útil. Frota e Destro (2006) afirmam que aproximadamente 30% dos sensores falham em até 5 anos após a instalação. Outro problema é a dificuldade, ou até mesmo a impossibilidade, de substituição ou manutenção dos instrumentos devido à sua localização. Além disso, ainda é necessário interromper a produção para executar essas tarefas, ocasionando grandes perdas monetárias. Nesse contexto, analisadores virtuais tornam-se boas alternativas para aumentar a confiabilidade dos dados e até mesmo, atuar como um substituto em casos de falha do sensor (BARBOSA et al., 2015).

2.2 Analisadores Virtuais

Em muitas situações de controle de processos, devido a limitações de amostragem e ausência ou perda do sensor desejado, impede-se a detecção antecipada de distúrbios de carga, como na composição do produto de colunas de destilação e reatores químicos. Isso pode resultar em grandes desvios dos *setpoints* e tempos longos de recuperação de distúrbios. Uma das maneiras de solucionar este problema é a inserção das informações fornecidas por outras variáveis facilmente mensuráveis em analisadores virtuais, a fim de fornecer uma estimativa da saída controlada. Os resultados estimados podem ser utilizados, então, para o controle total da planta (THAM et al., 1991).

Os analisadores virtuais também são chamados de *soft sensor*, ou seja, a combinação das palavras “*software*”, uma vez que os modelos são geralmente programas de computador e da palavra “*sensores*”, porque os modelos estão fornecendo informações semelhantes às suas contrapartes de *hardware* (KADLEC et al., 2009).

Dentre muitas aplicações este tipo de sensor pode utilizado como: um modelo de processo que fornece uma estimativa da variável desejada a partir das medições de outras variáveis; um modelo de observação que explica outra variável a partir da estimativa desejada da variável anterior e, então, usada para corrigir a saída do tipo de

processo; pode, ainda, ser utilizado como um estimador sequencial responsável por alimentar as entradas do modelo, ler a saída do modelo e calcular os finais ou estimar a variável desejada em cada iteração (TEIXEIRA et al., 2014).

Podem-se distinguir dois tipos de *Soft Sensors* de acordo com os princípios do modelo utilizado, sendo um deles baseado em modelos fenomenológicos e outro baseado em dados. Os primeiros são também chamados de modelo *white-box*, pois têm conhecimento fenomenológico sobre o processo, como por exemplo, uso de princípios de preservação de massa, balanços energéticos e cinéticas de reações. Entretanto, este modelo tem muitas desvantagens, são elas:

- i) Requer conhecimento avançado do processo;
- ii) Por vezes não há definições fenomenológicas suficientes;
- iii) Descreve uma base teórica do processo, ao invés de condições reais;
- iv) Geralmente centra-se na descrição do estado estacionário ótimo do processo, o que nem sempre é atingido (KADLEC et al., 2009).

Tendo em vista estas desvantagens, os modelos baseados em dados, chamados de modelo *black-box*, baseiam-se em observações do processo, ou seja, medições do processo real da planta que são gravados, armazenados e fornecidos como dados históricos. Este modelo normalmente está disponível para operadores de plantas e incorporados em sistemas de controle automatizados. Além disso, é muito utilizado para métodos de diagnóstico, prognóstico e apoio às decisões (KADLEC et al., 2009). Dentre os modelos que utilizam *soft sensor* do tipo *black-box*, Kadlec (2009) destaca os métodos preditivos empíricos como o Princípio de Regressão de Componentes Principais e o *Multi-layer Perceptron* (MLP).

Kadlec et al. (2009) descreve, de maneira geral, as etapas e os problemas típicos do procedimento de desenvolvimento de *Soft Sensor*. Esse procedimento segue o fluxo conforme a Figura 4.

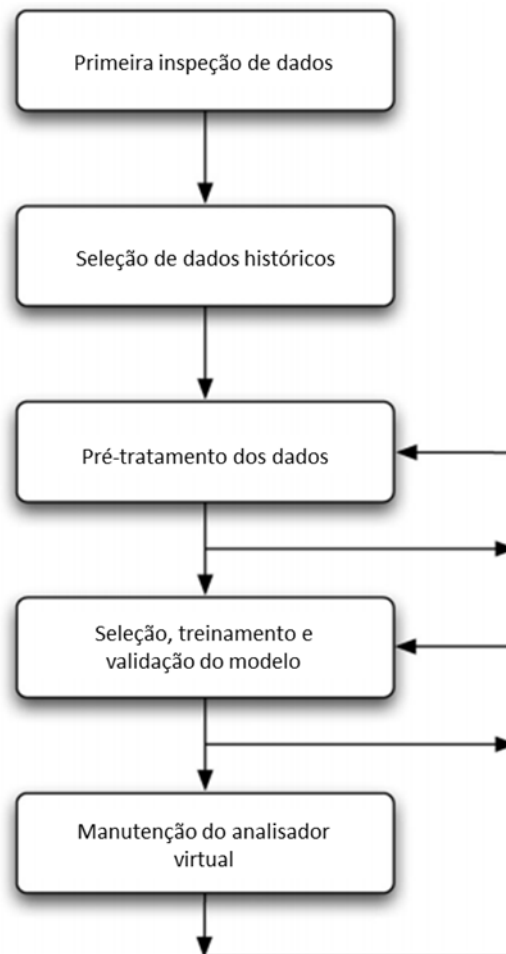


Figura 4: Fluxo de criação de um soft sensor. Fonte: adaptado de Kadlec et al. (2009).

- Primeira inspeção de dados

Ao gerar a base de dados podem ocorrer problemas, tanto na estrutura dos dados, quanto nas variáveis. Portanto, neste passo é realizada uma análise geral da base objetivando reconhecer os erros primários, como por exemplo, variáveis bloqueadas com valor constante. Assim como compreender a complexidade dos dados para escolha do melhor tipo e especificações do modelo (KADLEC et al., 2009).

- Seleção de dados históricos

Aqui, serão selecionados os dados que serão treinados, assim como, identificados e marcados os dados referentes aos estados estacionários para que sejam utilizados em uma modelagem adicional. Para modelos dinâmicos utiliza-se de lotes representativos ao invés da identificação dos estacionários (KADLEC et al., 2009).

- Pré-processamento de dados

No pré-processamento, o objetivo é transformar a base de dados a fim de que a rede seja treinada da forma mais eficaz possível, são exemplos desta etapa:

- i) A normalização dos dados para a média zero e variância unitária;
- ii) A manipulação de dados perdidos;
- iii) Detecção e substituição *outliers*;

iv) Seleção de variáveis relevantes.

Como apresentado na Figura 4, o pré-processamento é iterativo, sendo assim, mesmo após as próximas etapas do modelo, deve-se revisar os dados e atentar aos pontos abordados acima (KADLEC et al., 2009).

- Seleção, treinamento e validação do modelo

Esta é a etapa crucial para o desenvolvimento de um *soft sensor* que atenda os fins que se destina, pois é nela que serão selecionados o tipo de modelo, a divisão da base de dados, a métrica de avaliação da eficiência do modelo e os parâmetros utilizados (KADLEC et al., 2009).

Existem diversos tipos de modelos, desde os mais clássicos como o método de regressão linear, até métodos mais sofisticados como redes neurais. A sua escolha deve levar em consideração a complexidade do problema, o tempo de desenvolvimento requerido, além da robustez e precisão desejadas.

Para se estimar a capacidade preditiva do modelo, dentre vários métodos existentes dois são mais comumente utilizados. No modelo mais básico basta se dividir a base de dados em três grupos: treino, validação e teste. Sendo a base de treino utilizada para treinar o modelo, a de validação para minimizar o erro de bias e de variância, por fim, a de teste, para que seja analisado o resultado da predição (HASTIE et al., 2008). Essa metodologia é bastante simples, contudo utiliza informação da base de maneira ineficaz, já que parte da base é exclusiva para validação, além de variar seu resultado de acordo com a parte escolhida para treino e validação (JAMES et al., 2013).

Outro método, bastante utilizado é o *K-fold cross-validation*, representado na Figura 5, que surge como uma maneira de otimizar o uso das informações. Esse consiste em dividir a base de maneira aleatória em K partes. K-1 partes são utilizadas para treinamento e a parte remanescente é tratada na validação, sendo esse processo, então, repetido para todas K partes (James et al., 2013). Porém, é necessário escolher com cautela o valor de K parcelas, pois com $K = N$ a estimativa é aproximadamente imparcial em relação ao erro de previsão, mas pode ter alta variância, tendo em vista que os N "conjuntos de treinamento" são muito semelhantes, além de ser exigida uma carga computacional elevada para seu processamento. Com isso, Kohavi (1995) ao dividir a base em 10 partes obteve o melhor resultado nas estimativas do modelo.

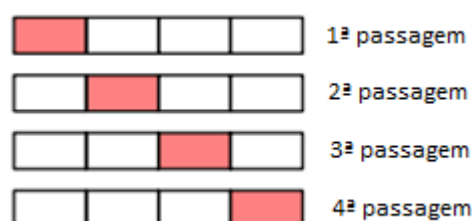


Figura 5: Representação da técnica de validação cruzada *k-fold cross-validation*, com $K = 4$. Fonte: adaptado de Bishop (2006).

Após decididos os parâmetros, o modelo é treinado e validado. A seguir é importante que se avalie o desempenho do modelo com suas especificações. Para avaliação do desempenho estatístico, o mais comum é empregar o erro quadrático médio (*Mean*

Squared Error - MSE), o qual mede a distância quadrática entre o valor estimado (\hat{y}) e o valor correto (y) ponderado pelo número de termos (n), conforme a Equação 1:

$$MSE = \frac{1}{n} \sum_i (y_i - \hat{y}_i)^2 \quad \text{Equação (1)}$$

- Manutenção

Por fim, após o desenvolvimento e implantação do *Soft Sensor*, ele deve ser mantido em observação e ajustado regularmente, por meio de compensação numérica. E caso o desempenho do *Soft Sensor* se deteriore com o passar do tempo o modelo terá que ser adaptado ou, até mesmo, redesenvolvido (KADLEC et al., 2009).

2.3 Modelo de rede neural

As redes neurais artificiais originalmente surgiram com o objetivo de construir modelos computacionais motivados pela operação similar a de neurônios biológicos, ou seja, as unidades básicas de processamento de informação em sistemas nervosos. (KADLEC et al., 2009).

O neurônio artificial é um importante componente de construção da rede neural. Nele, os sinais que chegam (entradas), associados aos pesos de conexão (ajustados no treinamento do modelo) são somados/combinados e transformados por uma função de transferência, ou função de ativação, para produzir a saída do neurônio, conforme ilustrado na Figura 6 (AGATONOVIC-KUSTRIN e BERESFORD, 2000).

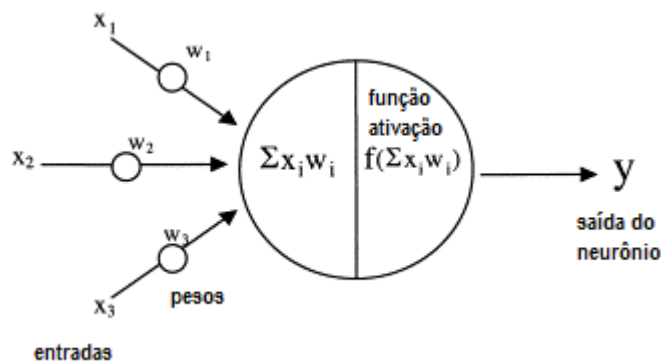


Figura 6: Representação de um neurônio artificial. Fonte: adaptado de Agatonovic-kustrin e Beresford (2000).

Arquiteturas de redes neurais são tipicamente organizadas por camadas. Geralmente as camadas são divididas em três grupos, como representado na Figura 7:

- Camada de Entrada: onde os dados são inseridos à rede;
- Camadas Ocultas ou Intermediárias: onde é realizada a maior parte do processamento da rede neural;
- Camada de Saída: onde o resultado final é concluído e apresentado.

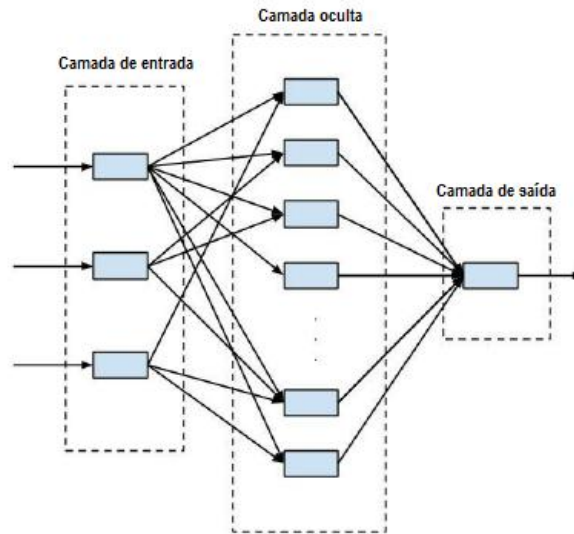


Figura 7: Representação de uma rede neural típica. Fonte: Adaptado de Deka et. al (2015)

2.3.1 Hiperparâmetros

Os hiperparâmetros são parâmetros que serão incorporados apenas pelo algoritmo de aprendizado e não pelo modelo. Com isso, devem ser definidos anteriormente ao treinamento e permanecem constantes durante o mesmo. A definição dos hiperparâmetros é uma etapa decisiva na construção de uma rede neural, e pode ser a diferença entre a construção de uma rede com boa capacidade de predição ou não (GÉRON, 2017).

Os principais parâmetros são:

i) Número de *epochs*:

É cada rodada de ciclos de iteração que o algoritmo de aprendizado realiza. Este hiperparâmetro é muito importante para determinar o fim do treino, pois enquanto passam as *epochs* o algoritmo aprende e seu erro de previsão (RMSE) no conjunto de treinamento naturalmente diminui, assim como o erro de previsão no conjunto de validação. Entretanto, após certo tempo, o erro de validação para de diminuir e passa a aumentar. Isso indica que o modelo começou a sobrecarregar os dados de treinamento, este fenômeno é chamado de *overfitting*. Com técnica de parada antecipada basta interromper o treinamento na *epoch* em que o erro de validação atingir o mínimo ou ainda, quando alcança um valor limite, suficientemente pequeno que indique convergência, como ilustrado na Figura 8 (GÉRON, 2017).

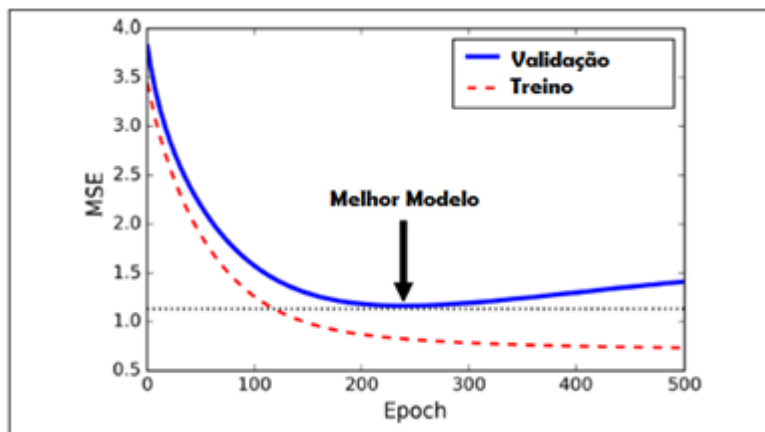


Figura 8: Representação da técnica de parada antecipada. Fonte: adaptado de Géron (2017).

ii) Função de ativação:

Realiza uma transformação não-linear nos dados de entrada, tornando-o modelo capaz de aprender e executar tarefas mais complexas. Além disso, diferencia informações recebidas pelo neurônio que serão utilizadas/ativadas ou rejeitadas. São utilizados para converter grandes saídas das unidades em um valor menor, além de promover a não linearidade na rede (GÉRON, 2017). Segundo Géron (2017) ReLu é a função que geralmente apresenta melhores resultados e Agatonovic-kustrin e Beresford (2000) afirmam que a mais comumente utilizada é a função *Sigmoid*. Os principais tipos são (GÉRON, 2017):

- Função *Sigmoid*:

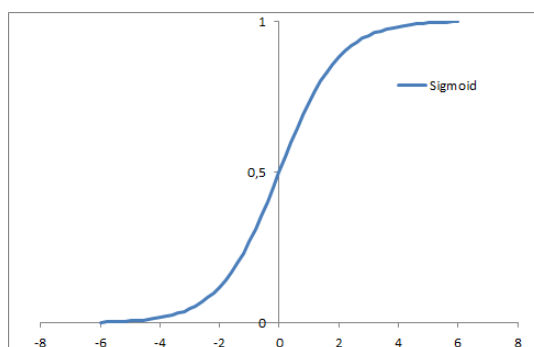


Figura 9: Representação da função *Sigmoid*. Fonte: Autoria própria.

- Função Tanh:

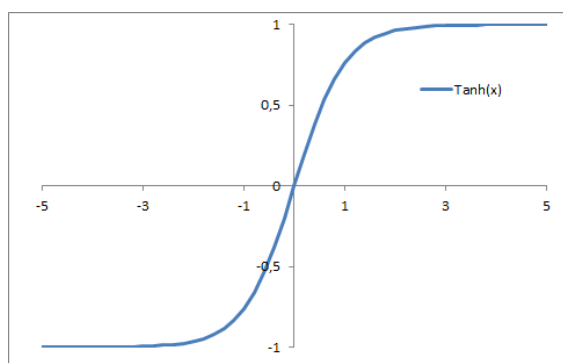


Figura 10: Representação da função Tanh. Fonte: Autoria própria.

- Função ReLU (*Rectified Linear Unit*):

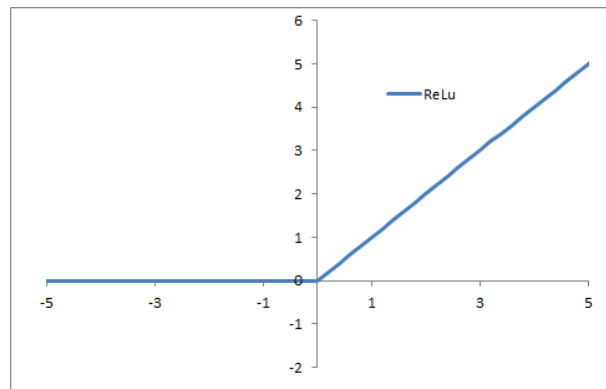


Figura 11: Representação da função ReLU. Fonte: Autoria própria.

iii) Número de camadas ocultas:

As chamadas camadas ocultas (*layers*) são as estruturas que contêm os neurônios artificiais. Elas são todas as camadas intermediárias entre a camada de entrada e a camada de saída. Segundo Géron (2017) para muitos problemas apenas uma ou duas camadas ocultas fará com que o modelo apresente capacidade preditiva adequada. Afirma, também, que com apenas uma camada oculta com algumas centenas de neurônios alcançam-se acuracidades superiores a 97%. E ainda, que é possível obter acuracidades acima de 98% utilizando 2 camadas ocultas com a mesma quantidade total de neurônios, em aproximadamente mesma quantidade de tempo de treinamento. Entretanto, para problemas mais complexos, deve-se gradualmente aumentar o número de camadas ocultas, sendo o valor limite onde ocorra o *overffiting* (GÉRON, 2017).

iv) Número de neurônios:

O número de neurônios nas camadas de entrada e saída é determinado pelo tipo de entrada e saída que a tarefa requer. Quanto às camadas ocultas, uma prática é dimensioná-los na forma de um funil, com menos e menos neurônios em cada camada sucessiva. Essa estrutura tem como base a lógica de unir muitos recursos de baixo nível em menos recursos do alto nível. Outra opção é estabelecer um número constante de neurônios por camada oculta (GÉRON, 2017).

Portanto, encontrar a quantidade perfeita de neurônios é uma tarefa que pode variar muito. Uma abordagem simples para auxiliar na escolha é a utilização de um modelo com mais camadas e neurônios do que o necessário, e a seguir fazer uso da parada antecipada, evitando o *overfitting* (GÉRON, 2017). Para Géron (2017), em geral obtém-se resultados mais efetivos aumentando o número de camadas do que o número de neurônios por camada.

V) Método de otimização:

Este algoritmo gradualmente reduz uma função erro (por exemplo, MSE) até que está convirja a um mínimo. O algoritmo mais tradicional, representado na Figura 12 é o Gradiente Descendente, este mede o gradiente local da função de erro em relação ao

vetor de parâmetro θ , e vai na direção do gradiente descendente, até que o valor nulo seja atingido (neste caso o mínimo da função). Pode-se destacar também outros algoritmos eficientes de otimização, como: AdaGrad, Adam Optimization e Adadelta. Géron (2017) afirma que na maioria das vezes o otimizador Adam será a melhor opção.

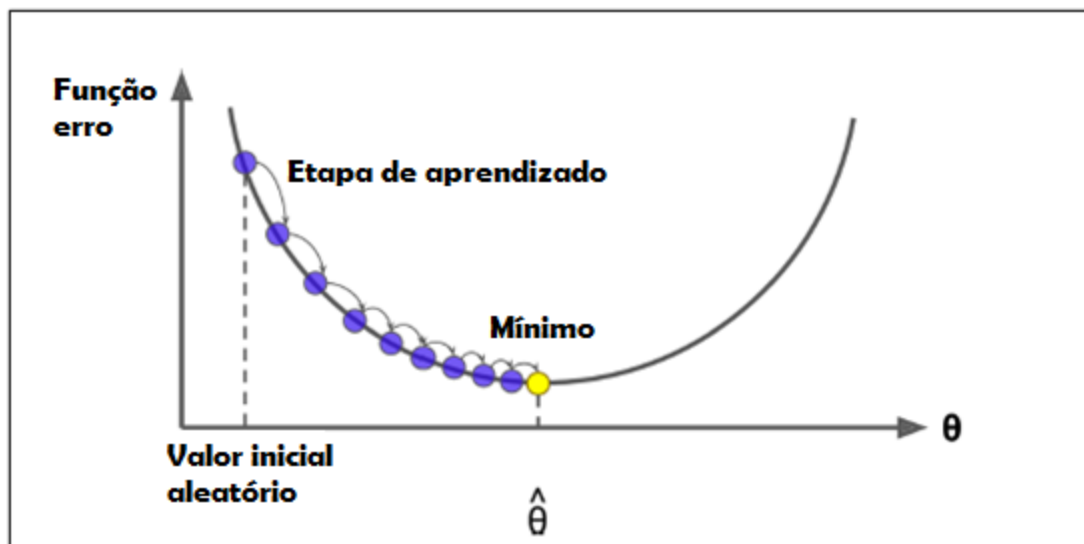


Figura 12: Representação método de otimização chamado de Gradiente Descendente, Fonte: adaptado de GÉRON, 2017.

VI) Método de inicialização:

É o método que será utilizado para determinar os pesos (bias) iniciais. Esta etapa afeta diretamente a convergência do algoritmo de otimização. Podem ocorrer 2 cenários: inicialização com pesos com valor igual a zero, o que faz com que o modelo seja equivalente a um modelo linear, tornando as unidades ocultas simétricas em todas as iterações. Ou então, para evitar este problema, é comumente utilizado inicializações aleatórias. Estas podem seguir a distribuição normal ou métodos mais otimizados, como por exemplo a inicialização proposta por Xavier & Yoshua (2010), onde, os pesos são inicializados levando em consideração o tamanho da camada anterior. Isso auxilia na busca do mínimo global da função erro com mais rapidez e eficiência. Os pesos ainda são aleatórios, mas dependem do tamanho da camada anterior de neurônios. Isso fornece uma inicialização controlada, e, portanto um gradiente de descida mais rápido e eficiente. (GÉRON, 2017).

Outro método otimizado é proposto por *He et al.* (2015), o qual é muito semelhante ao de Xavier & Yoshua (2010), apenas se diferencia pelo acréscimo de um fator de multiplicador de valor dois (GÉRON, 2017)..

2.4 Método de regressão linear

Modelos lineares foram amplamente desenvolvidos nos primórdios da estatística computadorizada, mas mesmo nos tempos atuais ainda existem boas razões para estudá-los e usá-los. Eles são simples e, muitas vezes, fornecem uma descrição adequada e interpretável de como as entradas afetam as saídas. Para fins de predição eles, as vezes, podem superar alguns dos modelos não-lineares, especialmente em situações que

contenham baixa relação sinal-ruído ou dados esparsos (HASTIE, TIBSHIRANI & FRIEDMAN, 2009).

A análise de regressão é um método que objetiva encontrar uma relação funcional entre variáveis dependentes (resposta) e variáveis independentes (preditoras) (FUMO & BISWAS, 2015). Neste modelo a função é uma equação linear e a variável dependente pode ser expressa como uma função de variáveis independentes segundo a Equação 2 (BILGILI & SAHIN, 2010):

$$Y = B_0 + B_1X_1 + B_2X_2 + \dots + B_nX_n \quad \text{Equação (2)}$$

onde Y é variável dependente, B₀ até B_n são parâmetros da equação e X₁ até X_n são as variáveis independentes.

2.5 Estruturas de redes neurais aplicadas à predição da Ppdg em sistemas *offshore*

Atualmente há alguns estudos (EIKREM et al., 2004; AAMO et al. 2005; AGUIRRE et al., 2017) direcionados a prever de maneira eficiente a pressão do poço visando o controle de processos de extração de petróleo, dentre estes é interessante evidenciar os estudos que se utilizam de modelos baseados em redes neurais e quais parâmetros são especificados para prover bons resultados.

Nesse contexto, Motke (2017) fez uso de modelo de predição baseado em rede neural do tipo *feed-forward* com duas camadas contendo 200 neurônios cada e obteve seu melhor resultado com um coeficiente de determinação (R²) igual a 0,98.

Por outro lado, Barbosa et al. (2015), objetivando estudar a dinâmica da PDG, a partir de uma estrutura de treinamento da rede composta por *multilayers perceptrons* com uma camada contendo dez neurônios, função de ativação sendo a tangente hiperbólica e uma saída de dados linear, pôde concluir que o modelo é suficiente para os fins que se destina.

Por fim, com o mesmo objetivo, Antonelo et al. (2017) utilizou dados reais de uma plataforma oriundos de instrumentos dispostos apenas na parte superior do leito marinho. Como parâmetros de treinamento fez uso do método de validação *cross-validation* com 9 partições, arbitrariamente 400 neurônios, e *Echo State Neural* como tipo de modelo. A partir disso, pôde, então, afirmar que mesmo um único modelo com tais especificações já é suficiente para desenvolver um sensor eficiente e robusto para tal predição.

3 Materiais e Métodos

Este capítulo descreve as etapas do desenvolvimento do *Soft Sensor*, a escolha da plataforma computacional, do tipo de rede neural, do tipo de modelo utilizado e os principais hiperparâmetros utilizados.

3.1 Desenvolvimento do *Soft Sensor*

A plataforma computacional utilizada para o desenvolvimento do modelo e avaliação da Rede Neural é o Python 3.5 juntamente com a biblioteca Keras (CHOLLET, 2015) e o *backend* TensorFlow (ABADI et al., 2015).

A base de dados utilizada como referência às informações de um poço de petróleo, foram extraídas de um modelo dinâmico semi-empírico simplificado chamado de FOWM (*Fast Offshore Wells Model*). Este modelo é capaz de simular comportamentos dinâmicos complexos como as golfadas e fora baseado em dados dos sistemas de produção da Petrobras em ambiente offshore de águas profundas e ultra profundas (DIEHL et al., 2017)

3.1.1 Primeira inspeção de dados seleção de dados históricos

Primeiramente, a base de dados foi extraída contendo as entradas do modelo FOWM, que são: a pressão do separador (P_s), a abertura da válvula *choke* ($\% choke$) e a vazão do gás *lift* (Gl), e as variáveis de saída, sendo elas: a pressão de topo *Ryser* (P_{rt}) a pressão na base do *Ryser* (P_{rb}), a pressão do topo do *Tubing* (P_{tt}) e a pressão medida no “olho” do poço, a P_{pdg} .

Foram simulados no Matlab 7.160.001 pontos do FOWM, resultado da combinação de 13 variações de vazão de gás *lift*, com 11 variações de abertura da válvula *choke* e 5 variações da pressão no separador, como representado na Tabela 1.

Tabela 1: Dados de entrada do simulador FOWM.

Vazão de gás <i>lift</i> (m^3 / dia)	Abertura da válvula <i>choke</i> (%)	Pressão do separador (Pa)
145000	1	613250
150000	3	713250
155000	5	813250
160000	7	913250
165000	9	1013300
170000	11	
175000	13	
180000	15	
185000	17	
190000	19	
195000	21	
200000		
205000		

Para ter conhecimento do comportamento da Ppdg com o tempo segue a Figura 13. Pode-se verificar que se trata de um modelo dinâmico com comportamento semelhante até tempo < 5860000 s, onde tem-se uma variação, seguida de uma estabilização em um “patamar” e assim sucessivamente. Somente após tempo > 5860000 s, onde a vazão de gás $lift = 145000$ m³/dia, abertura da válvula $choke = 19\%$ e Pressão do separador = 613250 Pa há mudança do comportamento para uma oscilação constante (ciclo-limite), representando o fenômeno chamado de golfada.

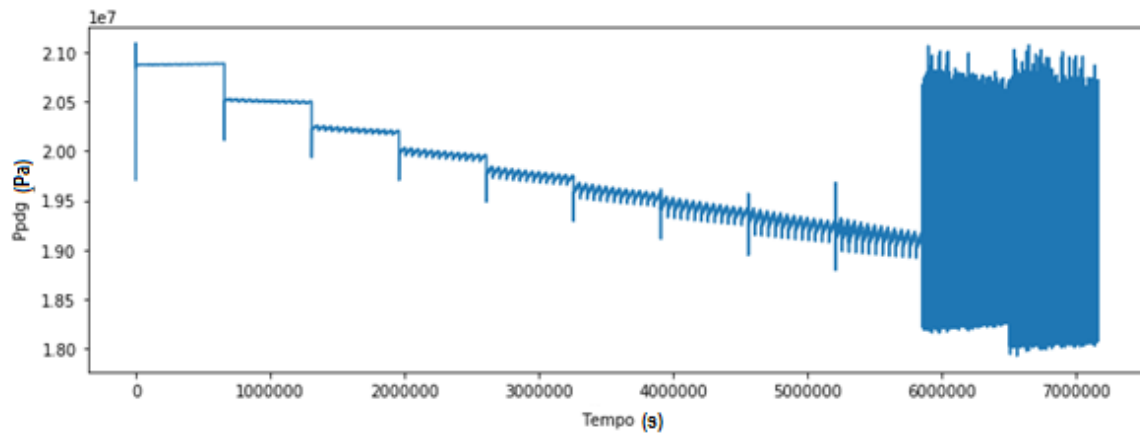


Figura 13: Variação da Ppdg pelo tempo.

3.1.2 Pré-tratamento dos dados

Tendo em vista a dimensão da base de dados, o pré-tratamento foi realizado no software Python. Visando, então, otimizar o funcionamento do algoritmo de aprendizado e, portanto, obter valores mais confiáveis do modelo optou-se por escalonar as variáveis de tal maneira que estas contivessem grandezas semelhantes. Essa metodologia foi empregada, pois é prejudicial para o algoritmo de aprendizado multiplicar pesos de variáveis tão distintas como, por exemplo, abertura da válvula $choke$ que varia de 1 a 21 e vazão de gás $lift$ que está entre 145000 Pa e 205000 Pa.

Esta padronização foi realizada segundo a Equação 3, e a representação de tal efeito pode ser observado na Figura 14, onde a variação máxima da Ppdg agora é de -0,6 até 0,4.

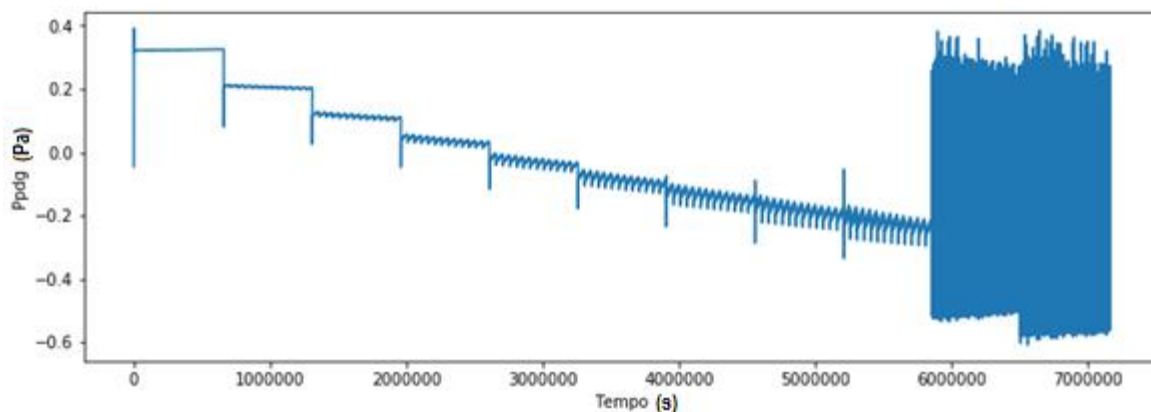


Figura 14: Normalização da PPDG.

$$V = \frac{V - \text{média}(V)}{\text{máx}(V) - \text{mín}(V)} \quad \text{Equação (3)}$$

onde, V é a variável que será normalizada.

3.1.3 Seleção, treinamento e validação do modelo

Para o desenvolvimento do modelo foi utilizada a rede neural do tipo *feedforward*, tendo em vista sua recorrência em pesquisas (ANTONELO, CAMPONOGARA, & FOSS, 2017) e sua simplicidade. Além disso, outro recurso usado foi a biblioteca Keras, a qual traz grandes recursos que auxiliam no desenvolvimento do modelo, como funções de ativação, métricas e otimizadores. Os dados de entrada utilizados foram todas as variáveis da base de dados exceto a Ppdg, já que esta é a variável de interesse.

No presente trabalho 20% da base foi previamente reservada para os testes. Testes esses que indicarão se o modelo é capaz de realizar previsões mesmo com dados de entrada que não foram treinados, ou seja que o modelo nunca “viu”. O restante da base é o conjunto de dados que será utilizado para o treino e validação do modelo. Para o treino e validação da rede neural, objetivando utilizar a base de maneira otimizada implementou-se o recurso *crossvalidation* com N = 3. Além disso, visando reduzir o consumo de dados, o tempo e evitar o *overffiting*, foi utilizada a parada antecipada (*early stopping*). Ou seja, no momento em que a função erro da validação começar a aumentar novamente o modelo para, mesmo antes de terminar o número total de *epochs*.

Os hiperparâmetros testados no estudo são apresentados na Tabela 2, são eles: o número de neurônios da primeira camada, o número de neurônios da segunda camada, o algoritmo de otimização e o algoritmo de inicialização. Como métrica da função erro foi utilizada a média do erro ao quadrado (MSE), representada na Equação 1.

Tabela 2: Hiperparâmetros utilizados no estudo.

Hiperparâmetro	Valor/funções
Algoritmo de otimização	SGD, Adagrad, Adadelta, Adam
Método de inicialização	glorot_normal, he_normal
Função ativação	Relu, tanh, sigmoid, elu
Número de neurônios 1° camada	16, 64, 256
Número de neurônios 2° camada	0, 16, 64, 256

Objetivando obter o conjunto de hiperparâmetros com o menor erro associado e um resultado que contemplasse todas as combinações possíveis optou-se por utilizar a busca chamada de *grid search*.

Após treinadas e validadas todas as combinações de hiperparâmetros, fora realizada uma análise de qual resultava em menor erro e está combinação, então, utilizada para implementação no modelo que utilizará como dados de entrada a base de teste reservada anteriormente.

Ainda, em uma nova análise, selecionaram-se os 60% intermediários dados da base para treinar, validar e testar novamente o modelo e os demais 20% iniciais e 20% finais da base para testar a precisão preditiva extrapolativa do modelo. Ou seja, o quanto o modelo é capaz de prever dados com comportamentos distintos do que o modelo fora antes treinado.

Por fim, 80% dos dados foram utilizados, aleatoriamente, para constituir a equação de regressão. Então, permanecendo aleatoriamente selecionados 20% dos dados que foram testados com o uso da equação formada.

4 Resultados

Treinaram-se 385 modelos e obtiveram-se os valores de MSE de validação, conforme a Tabela 3.

Tabela 3: Melhores combinações de hiperparâmetros do modelo.

	Otimizador	Inicialização	Ativação	N° neurônios 1° camada	N° neurônios 2° camada	MSE validação
1	Adadelta	glorot_normal	relu	256	0	4,01E-06
2	SGD	glorot_normal	relu	256	0	4,60E-06
3	Adagrad	he_normal	tanh	16	0	5,20E-06
4	Adagrad	he_normal	sigmoid	64	0	6,61E-06
5	Adagrad	glorot_normal	sigmoid	16	0	7,63E-06
6	Adagrad	glorot_normal	tanh	16	0	7,88E-06
7	Adadelta	glorot_normal	relu	64	0	8,19E-06
8	Adagrad	he_normal	sigmoid	16	0	8,37E-06
9	Adagrad	glorot_normal	tanh	256	0	8,65E-06
10	Adagrad	he_normal	sigmoid	16	64	9,04E-06
11	Adagrad	he_normal	elu	16	0	9,51E-06
12	Adagrad	he_normal	relu	64	0	9,57E-06
13	Adagrad	glorot_normal	elu	16	0	9,79E-06
14	Adagrad	he_normal	tanh	64	0	1,06E-05
15	Adagrad	he_normal	sigmoid	256	0	1,09E-05
16	SGD	glorot_normal	relu	256	256	1,12E-05
17	SGD	glorot_normal	relu	64	16	1,15E-05
18	Adagrad	glorot_normal	relu	16	0	1,17E-05
19	Adagrad	he_normal	sigmoid	64	64	1,17E-05
20	Adagrad	glorot_normal	sigmoid	64	0	1,29E-05

Pode-se observar que foram atingidos resultados com erros muito baixos na ordem de 10^{-6} ; que os hiperparâmetros exercem grande influência no resultado do modelo; ainda, que os parâmetros estejam correlacionados, ou seja, não basta encontrar a melhor otimização de um deles e deixa-lo constante para as demais otimizações, é necessário testar todas as combinações para posterior comparação de resultados.

A fim de avaliar impacto individual de cada um dos hiperparâmetros, selecionaram-se os 100 melhores resultados e avaliou-se a frequência em que os parâmetros estavam presentes nos mesmos. Evitou-se realizar a média dos erros, pois algumas combinações não convergiram e, ainda, haveria muitos *outliers* que comprometeriam a média.

Para verificar a probabilidade estatística de obter bons resultados com: os algoritmos de otimização construiu-se o gráfico da Figura 15, para os métodos de inicialização a Figura 16 e para as funções ativação a Figura 17.

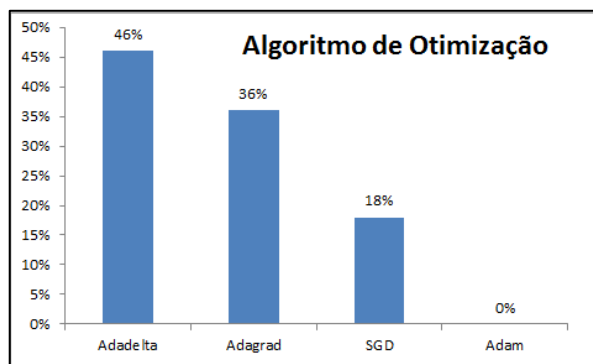


Figura 15: Avaliação dos algoritmos de otimização.

Pode-se observar que a maioria dos melhores resultados obtidos utilizaram como algoritmo de otimização o Adadelta (46%), e ainda que o Adam não é efetivo (0%) para esta modelagem. Isso está em desacordo com o que é proposto por Géron (2017), que relata que na maioria dos casos o algoritmo Adam será a melhor escolha.

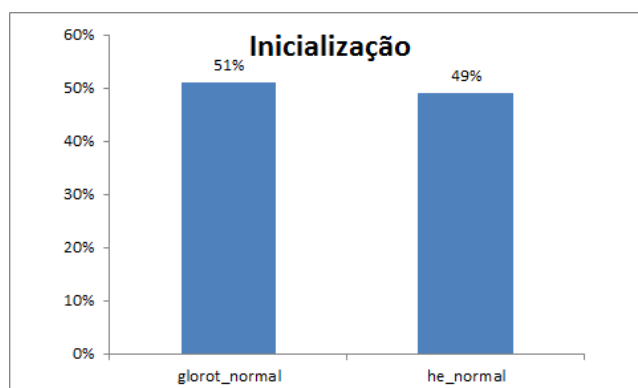


Figura 16: Avaliação dos métodos de inicialização.

Nos métodos de inicialização, pode-se verificar que a variação do parâmetro não tem grande impacto sobre a frequência em que estão presentes nos melhores resultados obtidos. Isso pode ser explicado tendo em vista que o método he_normal é apenas o método gloriot_normal multiplicado por um fator igual a 2.

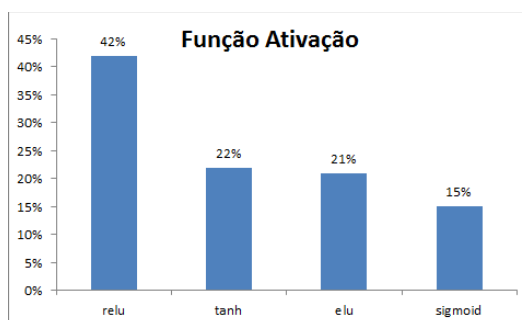


Figura 17: Avaliação das Funções de ativação.

Na avaliação das funções ativação a ReLu se mostrou a melhor opção sendo a melhor escolha em 42% dos casos, ou seja, estatisticamente duas vezes mais efetiva do que a segunda colocada Tanh (22%). Assim como sugerido na literatura, em que Géron (2017) declara que ReLu em grande parte dos casos será a melhor escolha.

Com o objetivo de utilizar-se da menor quantidade de dados possíveis e realizar o estudo no menor tempo, analisou-se o desempenho dos modelos em função do número de parâmetros, ou seja a soma total dos neurônios e obteve-se a Figura 18.

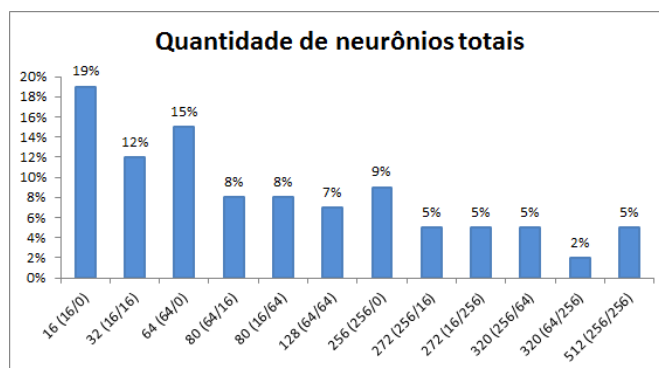


Figura 18: Avaliação do número de neurônios totais no modelo, onde no eixo X é a soma dos número de neurônios na 1ª camada e numero de neurônios na 2ª camada além da representação dos mesmos (número de neurônios na 1ª camada/ número de neurônios na 2ª camada).

Pode-se constatar que para a complexidade desta base de dados o uso de poucos parâmetros é mais relevante, tendo em vista que 19% das melhores respostas utilizaram apenas o mínimo de 16 neurônios na primeira camada e nenhum neurônio na segunda, e ainda que o aumento de parâmetros representa uma redução na probabilidade de ser um modelo com erros baixos. Além disso, quando acrescentou-se parâmetros na segunda camada obteve-se redução da eficácia do modelo, como no ponto 32 (16/16), o qual tem um total de neurônios reduzido e ainda assim é menos representativo (12%) em comparação ao ponto 64 (64/0), 15%, o qual não tem neurônios na segunda camada

A seguir, com o resultado otimizado da Tabela 3, utilizando os hiperparâmetros otimizador = Adadelta, inicialização = glorot_normal, ativação = relu, N° neurônios na 1° camada = 256 e N° neurônios na 2° camada = 0 realizou-se a etapa de teste, utilizando a base que estava previamente reservada. Com isso obteve-se as Figura 19 e Figura 20, que ilustram a capacidade do modelo em prever a Ppdg frente os dados de teste. Pode-se observar que o modelo é capaz de prever a Ppdg com erro muito baixo, sendo o MSE = $5,02 \times 10^{-6}$ e o $R^2=0,9998969$.

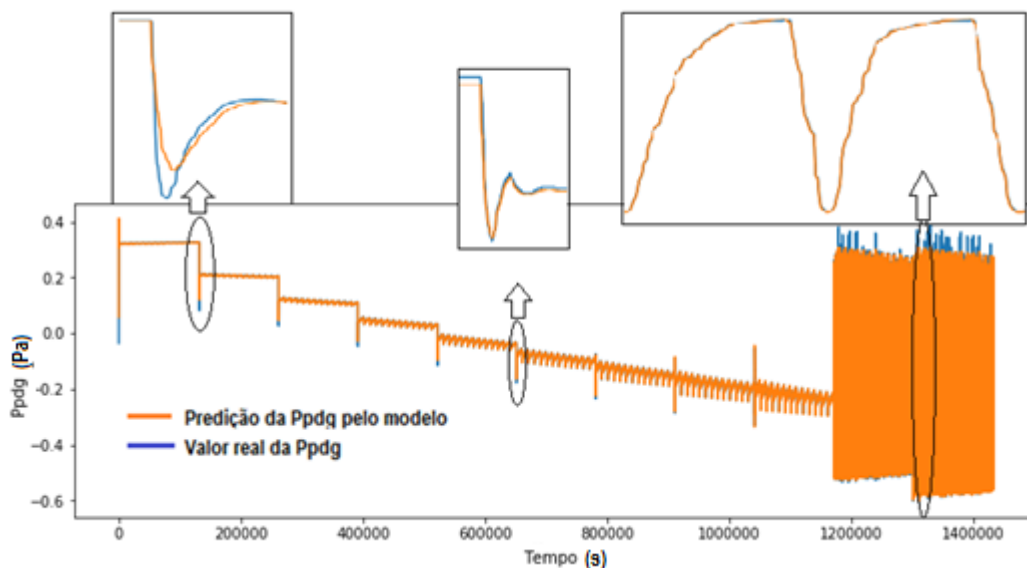


Figura 19: Capacidade do modelo em prever a Ppdg frente aos valores de teste.

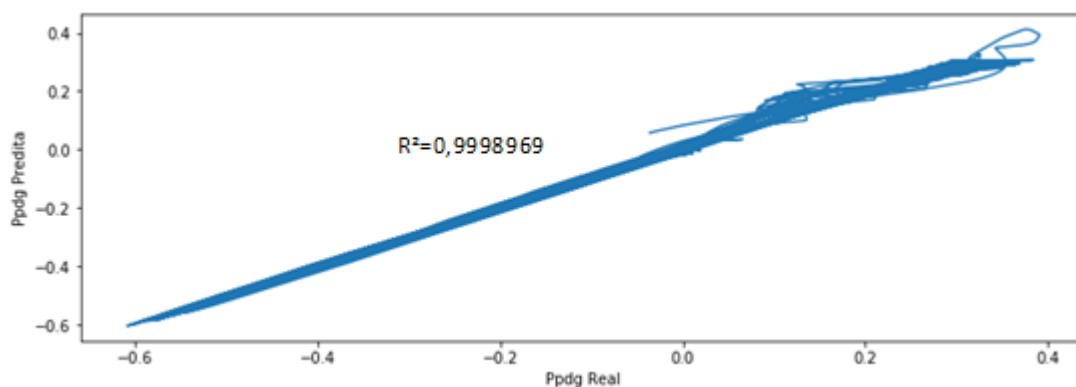


Figura 20: Reta que representa a precisão do modelo frente aos valores reais.

Ainda, pode-se constatar que o modelo ao ser extrapolado perde muito sua capacidade preditiva, como mostrado na Figura 21, pois na parte central da base (onde o modelo foi treinado e validado) o R^2 de teste foi de 0,999891. Já na extremidade inicial o $R^2_i = 0,66$ e na final $R^2_f = 0,70$. Este fenômeno pode ser explicado devido ao modelo ser do tipo caixa preta, ou seja, está baseado em dados anteriormente treinados para prever, e neste caso as extrapolações tem comportamentos distintos dos quais este fora preparado para operar.

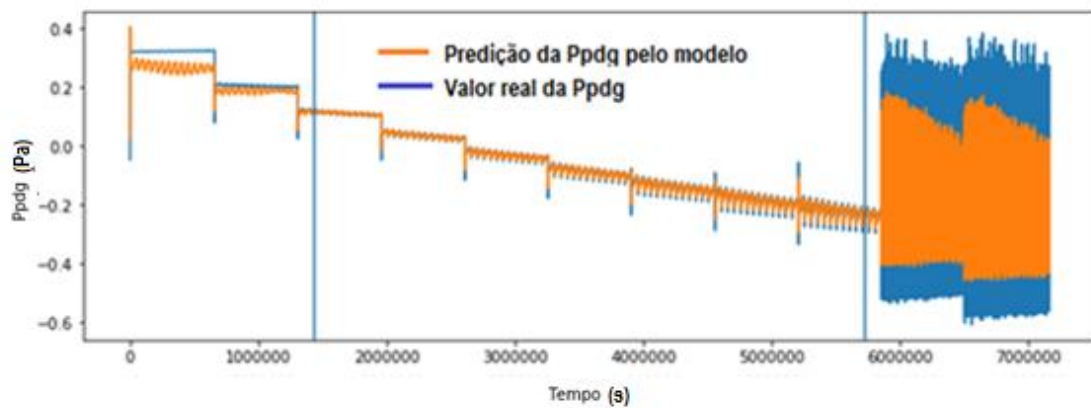


Figura 21: Precisão preditiva extrapolativa do modelo.

Por fim, foi treinado o modelo baseado no método de regressão linear (RL) e a equação de predição obtida foi a Equação 4. O R^2 obtido foi de 0,995022 e o MSE igual a 2×10^{-4} . Quando comparados os resultados de regressão linear e rede neural, pode-se verificar que a modelagem de RL oferece resultados menos acurados, tendo em vista seu MSE ser 50 vezes maior que o obtido na otimização da rede neural.

Entretanto, quando comparado o tempo de desenvolvimento e a capacidade computacional do modelo a RL se mostra mais eficiente. Tendo em vista que o desenvolvimento e otimização da rede neural foi realizado em 7 dias utilizando 3 computadores com bons processadores, enquanto a RL foi desenvolvida e testada em 20 minutos.

$$Ppdg = -4,282 * 10^{-6} - 8,681 * 10^{-3} * Gl - 3,340 * 10^{-2} * Choke - 8,496 * 10^{-4} * Ps - 1,463 * 10^{-1} * Prt + 1,301 * Prb - 1,657 * 10^{-1} * Ptt$$

Equação (4)

onde Gl a vazão do gás *lift*, *choke* é % de abertura da válvula *choke*, Ps é a pressão do separador, Prt é a pressão de topo *Ryser*, Prb é a pressão na base do *Ryser*, Ptt é a pressão do topo do *Tubing* e $Ppdg$ é a pressão medida no “olho” do poço.

5 Conclusões e Trabalhos Futuros

No presente trabalho, foi desenvolvido e otimizado um modelo de rede neural do tipo *feedforward*, baseado em variáveis facilmente medidas (Vazão de gás *lift*, Ps, Prt, Prb, Ptt, abertura da *Choke*) em uma planta de extração de petróleo e este se mostrou capaz de prever a Ppdg com erro muito baixo, sendo o MSE observado na validação de $4,01 \times 10^{-6}$. O melhor resultado foi obtido com a seguinte combinação de hiperparâmetros: otimizador = Adadelta, inicialização = *glorot_normal*, ativação = *relu*, N° neurônios na 1° camada = 256 e N° neurônios na 2° camada = 0.

Além disso, foi realizada a verificação da eficiência do modelo frente à base de teste, e o mesmo se mostrou capaz de se adaptar a dados não apresentados anteriormente. Comprovando ser uma ferramenta muito útil no caso de uma substituição do sensor físico da Ppdg por um analisador virtual. O coeficiente de determinação (R^2) da base de testes foi de 0.9998969.

Também, foram verificados os hiperparâmetros com a melhor probabilidade de apresentarem bons resultados para este modelo e conclui-se que para função ativação a melhor escolha seria a função ReLu, para o algoritmo de otimização o Adadelta e para o método de inicialização o *Glorot_normal*. Pode-se, ainda, verificar que o aumento do número total de parâmetros/neurônios diminui a eficiência do modelo.

Para mais, tendo em vista a utilização de um modelo do tipo caixa preta foi observado uma baixa precisão preditiva extrapolativa com $R^2_i = 0,659041$ e $R^2_f = 0,699827$. Evidenciando que para operar com este tipo de modelo é necessário que se tenha uma base de dados que contemple a maioria dos cenários possíveis de operação.

Por fim, ao testar o modelo de regressão linear obteve-se um $R^2 = 0,995022$ e o $MSE = 2 \times 10^{-4}$. Pode-se constatar que esta modelagem oferece resultados menos precisos, sendo seu MSE ser 50 vezes maior que o obtido na otimização da rede neural. Representando, assim, que o modelo escolhido para o estudo fornece uma acuracidade muito melhor que, até mesmo, os métodos clássicos. Entretanto a RL requer um tempo e uma capacidade computacional muito menor e, dependendo da acuracidade exigida pelo processo pode ser uma metodologia eficaz.

Sugere-se para trabalhos futuros a modelagem e teste de outros tipos de modelos como o Filtro de Kalman estendido (EKF) e modelos de árvores de decisão; além do uso de outras estruturas de redes neurais que não foram abordadas no trabalho, como *Echo State Neural*. Ainda, trabalhar com modelagem do tipo caixa cinza ou branca.

6 Referências

AAMO, O.; EIKREM, G.; SIAHAAN, H.; FOSS, B. Observer design for multiphase flow in vertical pipes with gas-lift - theory and experiments. **Journal of Process Control**, v. 15, p. 247-257, 2005.

ABADI, M.; BARHAM, P.; CHEN, J.; CHEN, Z.; DAVIS, A.; DEAN, J. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, 2015.

AGATONOVIC-KUSTRIN, S. e BERESFORD, R. Review: Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. **Journal of Pharmaceutical and Biomedical Analysis**, v. 22, p. 717–727, 2000.

AGUIRRE, L.; TEIXEIRA, B.; BARBOSA, B.; TEIXEIRA, A.; CAMPOS, M.; MENDES, E. Development of soft sensors for permanent downhole Gauges in deepwater oil wells. **Control Engineering Practice**, v. 65, p. 83–99, 2017.

ANTONELO, E. A.; CAMPONOGARA, E.; FOSS, B. Echo State Networks for data-driven downhole pressure estimation in gas-lift oil wells. **Neural Networks**, v. 85, p. 106-117, 2017.

ASHIKAWA, F. H. **Mitigação de golfadas em sistemas offshore utilizando modelo dinâmico simplificado**. Dissertação para Escola Politécnica da Universidade de São Paulo, 2017.

BARBOSA, B. H. G.; GOMES, L. P.; TEIXEIRA, A. F.; AGUIRRE, L. A. Downhole pressure estimation using committee machines and neural networks. **IFAC-PapersOnLine**, v.48, p. 286–291, 2015.

BARBOSA, B.H.G.; GOMES, L. P.; Teixeira, A. F.; Aguirre, L. A. Downhole Pressure Estimation Using Committee Machines and Neural Networks. **IFAC Workshop on Automatic Control in Offshore Oil and Gas Production**, 2015.

BISHOP, C. M. Pattern Recognition and Machine Learning. New York. 2006.

BRAGA, L. P.; CAMPOS, T. N. A Comparative Study Of Bid Models Adopted By Brazil, Peru, Colombia And Uruguay To Granting Petroleum Exploration And Production Rights. **5 Journal of World Energy Law & Business**, v. 99, p. 1-19, 2012.

CHOLLET, F., & others. Keras, 2015.

DEKA, A.; HAMTA, N.; ESMAEILIAN, B.; BEHDAD, S. Predictive Modeling Techniques to Forecast Energy Demand in the United States: A Focus on Economic and Demographic Factors. **Journal of Energy Resources Technology**, p. 138, 2015.

DI MEGLIO, F. et al. Stabilization of slugging in oil production facilities with or without upstream pressure sensors. **Journal of Process Control**, v. 22, p. 809-822, 2012.

DIEHL, F.; ANZAI, K. T.; ALMEIDA, S. C.; MEIEN, F. V. O.; NETO, S. S.; ROSA, R. V.; CAMPOS, C. M. M. M.; REOLON, F.; GEREVINI, G.; RANZAN, C.; FARENZENA, M.; TRIERWEILER, J. Fast Offshore Wells Model (FOWM): A practical dynamic model for

multiphase oil production systems in deepwater and ultra-deepwater scenarios. **Computers & Chemical Engineering**, v. 99, p. 304-313, 2017.

EIKREM, G.O.; IMSLAND, L.; FOSS, B. Stabilization of gas lifted wells based on state estimation. In **IFAC International Symposium on Advanced Control of Chemical Processes**. Hong Kong, China, 2004.

FABRE, J.; PERESSON, L.L.; CORTEVILLE, J.; ODELLO, R.; BOURGEOIS, T. Severe slugging in pipeline/riser systems. **SPE Prod. Eng.** v. 5, p. 299–305, 1990.

FROTA, H.M.; DESTRO, W. Reliability evolution of permanent downhole gauges for campos basin sub sea wells: A 10-years case study. In **2006 SPE Annual Technical Conference and Exhibition**. Santo Antonio: Society of Petroleum Engineers, 2006.

GÉRON, A. Hands-On Machine Learning with Scikit-Learn and TensorFlow Concepts, Tools, and Techniques to Build Intelligent Systems. **O’ Reilly Media**. v.1, 2017.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. H. The elements of statistical learning: data mining, inference, and prediction. New York: Springer, v.2, 2009.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. The elements of statistical learning: Data Mining, Inference, and Prediction. New York: Springer series in statistics, v.2, 2008.

HE, K. “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification.” **2015 IEEE International Conference on Computer Vision (ICCV)**. p. 1026-1034, 2015.

HERTZ, J.; KROGH, A.; PALMER, R.G. Introduction to the Theory of Neural Computation. Redwood City: Addison-Wesley, 1991.

JAHANSHAH, E.; SKOGESTAD, S. E.; HANSEN, H. Control structure design for stabilizing unstable gas-lift oil wells. **8th IFAC Symposium on Advanced Control of Chemical Processes**, SI – Furama Riverfront, 2012.

JAHANSHAH, E. Control Solutions for Multiphase Flow Linear and nonlinear approaches to anti-slug control. Tese de doutorado. Trondheim, 2013.

JAMES, Gareth; WITTEN, Daniela; HASTIE, Trevor; TIBSHIRANI, Robert. An introduction to statistical learning. New York: springer, 2013.

Jansen, B., Dalsmo, M., Nokleberg, L., Havre, K., Kristiansen, V., and Lemetayer, P. Automatic control of unstable gas lifted wells. In **SPE Annual Technical Conference and Exhibition**. Houston, Texas, 1999.

KADLEC, P.; GABRYS, B.; STRANDT, S. Data-driven soft sensors in the process industry. v. 33, p. 795–814, 2009.

KOHAVI, Ron. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *Ijcai*. p. 1137-1145, 1995.

MEGLIO, F. Dynamics and Control of Slugging in Oil Production, Doctorate Thesis. **Paris Institute of Technology**, 2011.

MOTKE M. B. Predição de pressão de fundo em poços de petróleo via redes neurais. UFRGS, 2017.

PETROBRAS. **Participação do setor de petróleo e gás chega a 13% do PIB brasileiro.** Disponível em: <<http://www.petrobras.com.br/fatos-e-dados/participacao-do-setor-de-petroleo-e-gas-chega-a-13-do-pib-brasileiro.htm>> Acesso em: 16/11/2018

SCHMIDT, Z.; BRILL, J.P.; BEGGS, D.H. Choking can eliminate severe pipelineslugging. *Oil Gas J.* 12, 230–238, 1979.

SMART, J. S.; SMITH, G.S. Pigging and Chemical Treatment for Oil and Gas Pipelines. **Richardson: Society of Petroleum Engineers**, 1997.

TEIXEIRA, B. O.; CASTRO, W. S.; TEIXEIRA, A. F.; AGUIRRE, L. A. Data-driven soft sensor of downhole pressure for a gas-lift oil well. **Control Engineering Practice**, v. 22, p. 34–43, 2014.

THAM, M. T.; MONTAGUE, C. A.; MORRIS, A. J.; LANT, P. A. Soft sensors for process estimation and inferential control. **J. Process Control.**, v.1, p 3-14, 1991.

WILLINGS, M. J.; MONTAGUE, G. A.; DI MASSIMO, C.; THAM, M. T.; MORRIS, A. J. Artificial neural networks in process estimation and control. **Automatica**, v. 28, p. 1181–1187, 1992.

XAVIER G., YOSHUA B. Understanding the difficulty of training deep feedforward neural networks, p. 249-256, 2010.

ZURADA, J. M. Introduction to Artificial Neural Systems. **St. Paul, MINN: West Publishing Company**, 1992.