

UNIVERSIDADE FEDERAL FLUMINENSE
INSTITUTO DE CIÊNCIA E TECNOLOGIA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Luís Felipe de Oliveira Fernandes

PREDIÇÃO DE DEMANDA DE ENERGIA EM PRÉDIOS
COMERCIAIS USANDO APRENDIZADO ONLINE

Rio das Ostras-RJ

2019

LUÍS FELIPE DE OLIVEIRA FERNANDES

PREDIÇÃO DE DEMANDA DE ENERGIA EM PRÉDIOS COMERCIAIS USANDO
APRENDIZADO ONLINE

Monografia apresentada ao Curso de
Bacharelado em Ciência da Computação da
da Universidade Federal Fluminense, como
requisito parcial para obtenção do Grau de Ba-
charel.

Área de Concentração: Inteligência Artificial,
Aprendizado de Máquina Online.

Orientador: Prof. Dra. FLÁVIA CRISTINA BERNARDINI

Rio das Ostras-RJ

2019

LUÍS FELIPE DE OLIVEIRA FERNANDES

PREDIÇÃO DE DEMANDA DE ENERGIA EM PRÉDIOS COMERCIAIS USANDO
APRENDIZADO ONLINE

Monografia apresentada ao Curso de
Bacharelado em Ciência da Computação da
da Universidade Federal Fluminense, como
requisito parcial para obtenção do Grau de Ba-
charel. Área de Concentração: Inteligência Ar-
tificial.

Aprovada em JULHO de 2019.

BANCA EXAMINADORA

Prof. Dra. FLÁVIA CRISTINA BERNARDINI - Orientador
UFF

Prof. Dr. DALESSANDRO SOARES VIANNA
UFF

Prof. Dr. EDWIN BENITO MITACC MEZA
UFF

Rio das Ostras-RJ
2019

Dedico esse trabalho à minha família,
em especial aos meus pais e, em memória, à minha
avó

Lúcia Marcelina dos Santos Fernandes.

Agradecimentos

Primeiramente a todos os guias e espíritos de luz que me acompanham e puderam me direcionar nessa trajetória.

À minha avó, em memória, **Lúcia Marcelina dos Santos Fernandes** por toda a base que pôde me proporcionar, além de conhecimento e amor e foi a principal influenciadora para que eu estivesse no meio acadêmico.

Aos meus pais **Alessandra de Oliveira Marques** e **Marco Aurélio Fernandes** pois, sem eles, eu não estaria aqui hoje.

Aos meus irmãos, em especial à **Thaís de Oliveira Fernandes**, por terem sido meu suporte durante fases difíceis na minha formação acadêmica.

Aos meus amigos, em especial **Alessandro Sampaio**, **Camila Alves**, **Tatiane de Sousa** e **Thaynara Cardoso**, por terem sido parte fundamental no meu processo de crescimento e amadurecimento nessa fase da minha vida.

Aos professores que sempre demonstraram amor pelo que fazem e sempre empenharam-se, como puderam, para oferecer o melhor para todos nós, alunos, nesse processo de formação. Eu agradeço, em especial, à minha professora e orientadora **Flávia Cristina Bernardini** por todos os ensinamentos, amizade, as oportunidades oferecidas, e, que de alguma forma, suas ações e paixão pelo que faz me inspiraram a chegar onde estou.

A todos que de alguma forma me apoiaram e que contribuíram para esse meu processo de formação.

Lista de Figuras

2.1	Esquema geral de aprendizado indutivo de regras. Fonte: Adaptado de [Shaw and Gentry, 1990]).	4
2.2	A Hierarquia do Aprendizado. Fonte: Elaboração própria.	6
2.3	Um exemplo da distribuição de camadas do algoritmo Multilayer Perceptron. Fonte: Elaboração própria.	11
3.1	Publicações realizadas entre 2013 até 2017. Fonte: Elaboração própria.	23
3.2	Publicações por área. Fonte: Elaboração própria.	24
4.1	Consumo Médio Diário. Fonte: Elaboração própria.	30
4.2	Consumo Médio Semanal. Fonte: Elaboração própria.	31
4.3	Consumo nos Feriados. Fonte: Elaboração própria.	31
4.4	Média e Desvio Padrão dos dias de semana do Medidor 1	33
4.5	Média e Desvio Padrão dos dias de semana do Medidor 2. Fonte: Elaboração própria. . .	33
4.6	Média e Desvio Padrão dos dias de semana do Medidor 3. Fonte: Elaboração própria. . .	33
4.7	Média e Desvio Padrão dos fins de semana do Medidor 1. Fonte: Elaboração própria. . . .	34
4.8	Média e Desvio Padrão dos fins de semana do Medidor 2. Fonte: Elaboração própria. . . .	34
4.9	Média e Desvio Padrão dos fins de semana do Medidor 3. Fonte: Elaboração própria. . . .	34
4.10	Arquivo de extensão .arff. Fonte: Elaboração própria.	35
4.11	Correlação de Pearson para o Algoritmo FIMT-DD. Fonte: Elaboração própria.	36
4.12	Correlação de Pearson do Medidor 1. Fonte: Elaboração própria.	37
4.13	Correlação de Pearson do Medidor 2. Fonte: Elaboração própria.	37
4.14	Correlação de Pearson do Medidor 3. Fonte: Elaboração própria.	37
4.15	Medidor 1 - FIMT-DD. Fonte: Elaboração própria.	40
4.16	Medidor 2 - FIMT-DD. Fonte: Elaboração própria.	40
4.17	Medidor 3 - FIMT-DD. Fonte: Elaboração própria.	41
4.18	Medidor 1 - Perceptron Online. Fonte: Elaboração própria.	41
4.19	Medidor 2 - Perceptron Online. Fonte: Elaboração própria.	41
4.20	Medidor 3 - Perceptron Online. Fonte: Elaboração própria.	42

Lista de Tabelas

2.1	Comparação entre sistemas de Banco de Dados e Fluxo de dados. Fonte: [Gama, 2010] . . .	14
3.1	Autores com mais publicações entre 2013 e 2017. Fonte: Elaboração própria.	23
3.2	Países com mais publicações. Fonte: Elaboração própria.	24
3.3	Idiomas com mais publicações. Fonte: Elaboração própria.	25
3.4	Keywords com mais publicações. Fonte: Elaboração própria.	25
3.5	Relação das publicações mais relevantes. Fonte: Elaboração própria.	26
4.1	Descrição dos dados. Fonte: Elaboração própria.	30
4.2	Correlação entre os medidores. Fonte: Elaboração própria.	32
4.3	Exemplo dos dados durante o pré-processamento. Fonte: Elaboração própria.	32
4.4	Resultados do Perceptron Online para a janela de 5t. Fonte: Elaboração própria.	38
4.5	Resultados do Multilayer Perceptron janela de 5t. Fonte: Elaboração própria.	38
4.6	Resultados do FIMT-DD para a janela de 5t. Fonte: Elaboração própria.	38
4.7	Resultados do FIMT-DD para a janela de 5t. Fonte: Elaboração própria.	38
4.8	Resultados do Perceptron Online para a janela de 10t. Fonte: Elaboração própria.	38
4.9	Resultados do Multilayer Perceptron janela de 10t. Fonte: Elaboração própria.	39
4.10	Resultados do FIMT-DD para a janela de 10t. Fonte: Elaboração própria.	39
4.11	Resultados da Regressão Linear para a janela de 10t. Fonte: Elaboração própria.	39
4.12	Resultados do Perceptron Online para a janela de 15t. Fonte: Elaboração própria.	39
4.13	Resultados do Multilayer Perceptron janela de 15t. Fonte: Elaboração própria.	39
4.14	Resultados do FIMT-DD para a janela de 15t. Fonte: Elaboração própria.	39
4.15	Resultados da Regressão Linear para a janela de 15t. Fonte: Elaboração própria.	40

Sumário

Agradecimentos	v
Lista de Figuras	vi
Lista de Tabelas	vii
Resumo	x
Abstract	xi
1 Introdução	1
1.1 Objetivo e Metodologia	2
1.1.1 Objetivo geral	2
1.1.2 Objetivos específicos	2
1.1.3 Metodologia	2
1.2 Organização da Monografia	2
2 Aprendizado de Máquina Online	4
2.1 Aprendizado Indutivo	4
2.2 Aprendizado Supervisionado	6
2.2.1 Aprendizado em Lote	9
2.2.2 Regressão	10
2.2.3 O Algoritmo Multilayer Perceptron	11
2.3 Aprendizado Online	12
2.3.1 Desafios no Aprendizado Online	14
2.3.2 O Algoritmo FIMT-DD	15
2.3.3 O Algoritmo Perceptron Online	16
2.4 Avaliação	17
2.4.1 Problemas de Avaliação	17
2.4.2 Concepção dos Experimentos de Avaliação	18
2.5 Métricas	18
2.5.1 Correlação de Pearson	18
2.5.2 SMAPE	19

2.5.3	Técnicas de Avaliação	19
3	Revisão da Literatura	20
3.1	Metodologia	20
3.2	Análise dos Trabalhos Seleccionados	22
3.2.1	Descrição da Literatura	25
3.3	Considerações Finais	28
4	Experimentos Realizados	29
4.1	Ferramentas Utilizadas	29
4.2	Modelagem	29
4.2.1	Coleta de dados	29
4.2.2	Pré Processamento	31
4.3	Análise Experimental	35
5	Conclusão e Trabalhos Futuros	43

Resumo

Segundo [Cury and Marques, 2017], as cidades inteligentes devem ter sua construção pautada em relações harmoniosas com o meio ambiente com medidas racionais para utilização dos recursos ambientais das diversas formas possíveis, principalmente em condições essenciais à vida humana como água e energia. E com o avanço do uso de tecnologias, o ser humano passa a ser um dos principais fomentadores de informações e soluções para as cidades inteligentes. O Aprendizado de Máquina Online é um método onde os dados se tornam disponíveis sequencialmente, chamados de fluxo de dados, e são usados para atualizar o melhor algoritmo de predição a cada rodada de leitura de dados. Isso significa que no aprendizado de máquina online, os dados estão sendo inseridos a todo momento e em grandes quantidades. O objetivo deste trabalho é analisar uma base de dados aberta de consumo de energia predial e fazer uma predição dos próximos índices de consumos dos prédios. Para alcançar esse objetivo, será utilizada a ferramenta MOA para avaliação do desempenho dos algoritmos online selecionados e verificação da performance dos mesmos em comparação com os algoritmos em lote que serão analisados através da ferramenta WEKA. Além disso, neste trabalho é feito uma revisão da literatura com a construção de uma bibliometria que pudesse contribuir com o trabalho, destacando os trabalhos que mais se aproximam do tema proposto. Ao final do trabalho são mostrados os resultados das avaliações dos algoritmos com base nas métricas escolhidas, com partes gráficas comparando os dados gerados.

Palavras-chave: Aprendizado de Máquina Online, Aprendizado Supervisionado, Inteligência Artificial, Cidades Inteligentes

Abstract

Based on [Cury and Marques, 2017], smart cities should be built on harmonious relationships with the environment with rational measures to use environmental resources in many ways. Especially under condition essential to human life such as water and energy. And with the advance of the use of technologies, the human being becomes one of the main promoters of information and solutions for smart cities. Online Machine Learning is a method where data becomes available sequentially, called a data stream, and is used to update the best prediction algorithm with each round of data reading. This means that in online machine learning, data is being entered all the time and in large quantities. The objective of this work is to analyze an open database of energy consumption and building electricity and to make a prediction of the next rates of consumption of buildings. To achieve this goal, the MOA tool will be used to evaluate the performance of the selected online algorithms and verify their performance against the batch algorithms that will be analyzed using the WEKA tool. In addition, this paper reviews the literature with the construction of a bibliometrics that could contribute to the work, highlighting the works that are closest to the proposed theme. At the end of the work, the results of the algorithm evaluations are shown based on the chosen metrics, with graphical parts comparing the generated data.

Keywords: Artificial Intelligence, Online Learning, Data Stream, Supervised Learning, Smart Cities

Capítulo 1

Introdução

Com o crescimento exponencial de dados sendo processados e consumidos a todo momento na rede de computadores, o aprendizado de máquina online se torna um tema interessante para lidar com esse tipos de dados conhecidos por seu fluxo contínuo e por seus resultados modificarem com o tempo. E todo esse fenômeno crescente alerta pesquisadores e entusiastas à buscarem soluções que possam beneficiar não só um uso mais eficiente de dispositivos como o dia a dia da sociedade.

A predição está diretamente relacionada a um evento incerto, quase sempre, baseado em conhecimento ou experiência [Stevenson, 1998]. Segundo [Upton and Cook, 2008], na estatística, a predição é parte da inferência estatística que faz afirmações a partir de um conjunto de amostras sobre uma população. As técnicas estatísticas utilizadas para a predição incluem Análise de Regressão e várias das suas subcategorias como a Regressão Linear, modelos lineares generalizados etc. Quando esses ou os métodos de Aprendizado de Máquina são implantados em uso comercial, eles são conhecidos como Análise Preditiva [Siegel, 2013].

O consumo de energia final é geralmente dividido em 3 grandes setores: Indústria, Transporte e “outros” que englobam agricultura, setor de serviços e residências. Isso torna ainda mais difícil de coletar informações a respeito do consumo de energia em edifícios [Pérez-Lombard et al., 2008]. Em países desenvolvidos, onde edifícios correspondem entre 20 a 40% do total da energia final consumida, acredita-se que esses podem ser considerados como um setor principal e dividido entre edifícios domésticos e edifícios não domésticos. Com o crescimento da população mundial e o aprimoramento dos serviços dos edifícios junto ao crescimento do tempo gasto dentro deles tem elevado o consumo de energia aos níveis do transporte e indústrias. Numa era onde os recursos estão ficando cada vez mais escassos, a preocupação com esse consumo em específico acabou sendo uma das prioridades do conceito de Cidades Inteligentes [Pérez-Lombard et al., 2008]. Depois de acontecimentos históricos como o Apagão vivido no começo dos anos 2000, a ideia de redes elétricas inteligentes tem se tornado um dos assuntos mais debatidos no Brasil, nos últimos anos, no que tange às cidades inteligentes, já que o país ocupa um posição não tão agradável no cenário mundial [Guimarães et al., 2013].

As *Smart Grids* são uma nova arquitetura de energia elétrica com a intenção de distribuir melhor a energia, baseando-se nas informações bidirecionais fornecidas por sensores inteligentes instalados, os *Smart*

Meters, o que possibilita um melhor planejamento e uso mais eficiente da rede [Guimarães et al., 2013].

1.1 Objetivo e Metodologia

1.1.1 Objetivo geral

O objetivo geral deste trabalho é propor uma abordagem para predição do consumo de energia de prédios comerciais utilizando Aprendizado de Máquina Online comparando o desempenho dos algoritmos desse aprendizado com os algoritmos de Aprendizado de Máquina em Lote.

1.1.2 Objetivos específicos

Os objetivos específicos são:

- Realizar a pesquisa de toda a base teórica que leva ao conceito de Aprendizado de Máquina Online;
- Efetuar um levantamento bibliométrico para mapear toda a produção científica a cerca de Predição e Aprendizado de Máquina Online nos últimos 5 anos.
- Fazer uso de ferramentas para auxiliar na Predição de consumo com os dados obtidos para pesquisa.

1.1.3 Metodologia

Inicialmente foi feita uma revisão da literatura para que pudesse ter a base necessária para entendimento e construção do caso de estudo, seguido da análise bibliométrica para que trabalhos relacionados pudessem ser encontrados e fomentarem, ainda mais, o trabalho. Com isso, foram utilizados os algoritmos de aprendizado de máquina em Lote, Regressão Linear e Multilayer Perceptron, e os algoritmos de aprendizado de máquina online, FIMT-DD e Perceptron Online. A vantagem em utilizar esses algoritmos online em comparação com os algoritmos de aprendizado em lote é a melhor adaptação dos mesmos a um grande conjunto de dados com fluxo contínuo. Para a avaliação do método foram utilizadas as métricas Correlação de Pearson e *Symmetric Mean Absolute Percentage Error*, também conhecido como SMAPE.

1.2 Organização da Monografia

Este trabalho está estruturado em 5 capítulos:

No Capítulo 1 é apresentada a Introdução que demonstra a contextualização do tema, a elaboração do problema e os objetivos da pesquisa.

No Capítulo 2 são apresentados os conceitos de Aprendizado Indutivo e o Aprendizado Supervisionado. Além disso, é exposto o conteúdo de Aprendizado Online bem como seus desdobramentos e abordagens.

No Capítulo 3, é apresentada a revisão bibliográfica da literatura bem como seus dados estatísticos baseados em registros publicados ao redor do mundo.

No Capítulo 4 são mostrados os resultados obtidos através da execução do experimento utilizando dados estatísticos junto à ferramenta.

No Capítulo 5 é apresentada uma análise final dos resultados obtidos e sugestões para trabalhos futuros de acordo com as perspectivas obtidas no trabalho.

Capítulo 2

Aprendizado de Máquina Online

2.1 Aprendizado Indutivo

Segundo [Pozo, 2016], na área de Aprendizado de Máquina é possível encontrar o aprendizado indutivo, que é tido como uma busca heurística através do espaço de descrições simbólicas, gerado pela aplicação de várias regras de inferência sobre as características inicialmente observadas. As regras de inferência incluem regras de generalização que executam transformações em descrições e regras convencionais dedutivas com preservação da originalidade. A aplicação destas regras de inferência para descrições é restringida pelo problema do conhecimento prévio e guiada por critérios de evolução da qualidade das afirmações indutivas geradas. Programas de aprendizado indutivo podem prover melhorias nas técnicas de programação correntes e uma base para o desenvolvimento alternativo de métodos de aquisição de conhecimento. Em pequenos e selecionados domínios, programas indutivos já são hábeis para determinar regras de decisão por indução em exemplos de decisões especialistas. Este processo notoriamente simplifica a transferência de conhecimento de um perito para uma máquina. Na Figura 2.1, temos um esquema básico de um sistema de Aprendizado Indutivo.

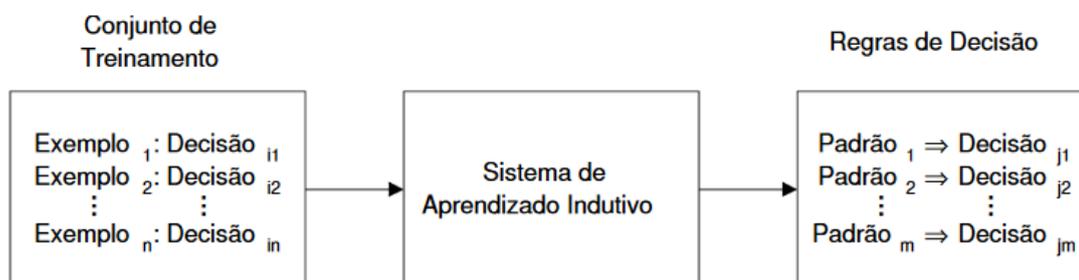


Figura 2.1: Esquema geral de aprendizado indutivo de regras. Fonte: Adaptado de [Shaw and Gentry, 1990]).

De uma maneira geral, um sistema que aprende a partir de exemplos recebe, como dados, informações na forma de situações específicas, cada uma delas devidamente classificadas (geralmente por especialista humano no domínio), caracterizando o que se convencionou chamar de aprendizado supervi-

sionado, e produz, como resultado, hipóteses que generalizam aquelas situações inicialmente fornecidas. Entre as principais características de Aprendizado Indutivo apresentadas por [Uchôa, 1998] encontram-se, além da Incrementabilidade já mencionada na seção anterior:

- **Teoria de Domínio:** No caso em que o sistema não possui informação a respeito do problema de aprendizado sendo tratado, o sistema induz a expressão do conceito apenas a partir dos exemplos disponíveis. Entretanto, para que a solução de problemas difíceis de aprendizado seja encontrada, é fundamental que um volume substancial de conhecimento sobre o problema esteja disponível, de maneira a subsidiar a indução do conceito. Esse conhecimento prévio existente é conhecido como Teoria de Domínio ou Conhecimento de Fundo.
- **Linguagem de Descrição:** Em inferência indutiva, os exemplos, a Teoria de Domínio e as hipóteses formuladas necessitam ser expressos em alguma linguagem e geralmente são utilizadas linguagens formais.
- **Precisão de Classificação:** É, geralmente, definida como o percentual de exemplos corretamente classificados pela hipótese induzida.
- **Transparência da Descrição Induzida do Conceito:** Geralmente, é essencial que a descrição do conceito, gerada por um sistema de Aprendizado de Máquina, seja compreensível pelo ser humano. O entendimento do conceito, por parte do ser humano, não apenas aumenta a credibilidade do sistema de Aprendizado de Máquina, mas também permite que o conceito possa ser assimilado e utilizado pelo especialista humano. Em muitas situações, a transparência da descrição é medida pelo número de descritores e operadores usados na descrição do conceito.
- **Complexidade Computacional:** Está relacionada com os recursos computacionais necessários (tempo e espaço) para realizar o aprendizado.

Para a expressão de um modelo de aprendizado indutivo são necessárias linguagens que descrevam os exemplos de treinamento, assim como linguagens que descrevam os conceitos aprendidos. Vários formalismos lógicos têm sido usados em sistemas de aprendizado indutivo para a representação de exemplos e conceitos [Bernardes, 2010]. O aprendizado indutivo pode ser dividido em aprendizado supervisionado e aprendizado não-supervisionado. Na Figura 2.2 é possível visualizar a hierarquia do aprendizado indutivo onde os sombreados nos levam ao aprendizado supervisionado utilizando Regressão que é o foco deste trabalho.

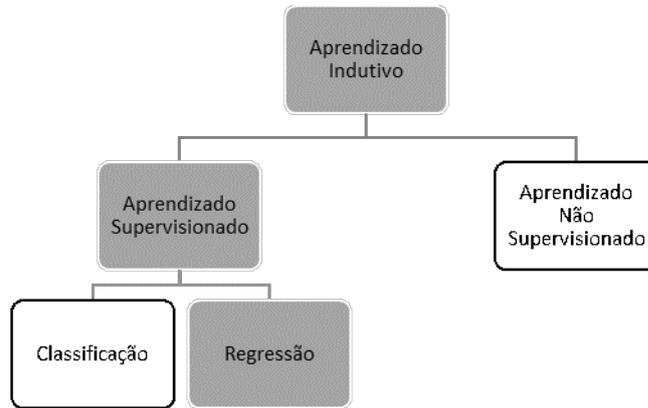


Figura 2.2: A Hierarquia do Aprendizado. Fonte: Elaboração própria.

O aprendizado não supervisionado é tido como a compreensão geral dos dados disponíveis. A chamada “descoberta de conhecimento” é, finalmente, a descoberta de padrões ocultos nos dados. Um cenário típico desse aprendizado é a segmentação, seja na indústria ou na segmentação de imagens. Nesse cenário, os dados são agrupados de acordo com suas similaridades. A segmentação pode ser utilizada para identificar novos grupos no conjunto de dados de forma que possa ser utilizado para reconhecimento de padrões.

O aprendizado não supervisionado analisa os exemplos fornecidos e tenta determinar se alguns deles podem ser agrupados de alguma maneira, formando agrupamentos [Chessman and Stutz, 1996]. Pode, também, proceder hierarquicamente das observações em níveis cada vez mais abstratos de representação. Cada hierarquia adicional precisa aprender apenas uma etapa e, portanto, o tempo de aprendizado aumenta (aproximadamente) linearmente no número de níveis na hierarquia do modelo [Valpola, 2002].

No aprendizado supervisionado, um conjunto de exemplos de treinamento é fornecido ao algoritmo de aprendizado para os quais o rótulo da classe associada é conhecido [Rezende, 2003]. Nesse modo, o algoritmo recebe um conjunto de dados rotulados como se fossem dados de treinamento, dados estes que consistem em um conjunto de exemplos de treinamento, onde cada exemplo é um par constituído por um objeto de entrada e um de saída, e fazem previsões acerca dos dados já avaliados. Como nesse trabalho será trabalhado a parte de aprendizado supervisionado, este tema será abordado de forma mais profunda na Seção 2.2.

2.2 Aprendizado Supervisionado

Um cenário ideal permitirá que o algoritmo determine corretamente os rótulos de classe para instâncias invisíveis. Isso requer que o algoritmo de aprendizagem generalize a partir dos dados de treinamento para situações não vistas de uma forma “razoável”. A fim de resolver um dado problema de aprendizagem supervisionada, segundo [Mohri et al., 2012], é preciso executar as seguintes etapas:

- **Determinar o Tipo de Exemplo de Treinamento:** Antes de fazer qualquer outra coisa, o usuário deve decidir que tipo de dados deve ser usado como um conjunto de treinamento.

- **Reunir um Conjunto de Treinamento:** O conjunto de treinamento precisa ser representativo do uso real da função. Assim, um conjunto de objetos de entrada é coletado e as saídas correspondentes também são coletadas, seja de especialistas humanos ou de medições.
- **Determinar a representação da entrada da função aprendida:** A precisão da função aprendida depende fortemente de como o objeto de entrada é representado. Normalmente, o objeto de entrada é transformado em um vetor de recurso que contém um número de recursos que são descritivos do objeto. O número não deve ser muito grande, por causa da maldição da dimensionalidade, mas deve conter informações suficientes para prever, com precisão, a saída.
- **Determinar a Estrutura da Função Aprendida e o Algoritmo Correspondente:** Por exemplo, a escolha do algoritmo entre máquinas de vetores de suporte ou árvores de decisão.
- **Terminar o Projeto:** Executar o algoritmo de aprendizagem no conjunto de treinamento reunido. Alguns algoritmos requerem que o usuário determine parâmetros de controle. Esses parâmetros podem ser ajustados ao otimizar o desempenho em um subconjunto do conjunto de treinamento, ou através de validação cruzada.
- **Avaliar a Precisão da Função Aprendida:** Após o ajuste dos parâmetros, o desempenho da função resultante deve ser medido num conjunto de teste separado do conjunto de treino.

No cenário mais geral do aprendizado supervisionado, a distribuição D é definida sobre $X \times Y$, e os dados de treinamento são uma amostra rotulada S dada de acordo com D :

$$S = ((X_1, Y_1), \dots, (X_m, Y_m)) \quad (2.1)$$

Vários algoritmos de aprendizado supervisionado estão disponíveis, cada um com seus pontos fortes e fracos. E o melhor algoritmo a ser utilizado é aquele que melhor se adapta ao seu conjunto de dados. Com isso, existem quatro questões que devem ser consideradas no aprendizado supervisionado:

- **Compensação de Desvio-Variância:** O primeiro problema é a compensação entre o desvio e a variância conhecido como Bias-Variance Tradeoff. Imagine que se tenha disponível vários conjuntos de dados de treinamento diferentes, porém igualmente bons. Um algoritmo de aprendizado é tendencioso para uma determinada entrada x se, quando treinado em cada um desses conjuntos de dados, é, sistematicamente, incorreto ao prever a saída correta para x . Um algoritmo de aprendizado tem alta variância para uma determinada entrada x se preder valores de saída diferentes quando treinados em diferentes conjuntos de treinamento. O erro de previsão de um classificador aprendido está relacionado com a soma do viés e da variância do algoritmo de aprendizagem. Geralmente, há uma compensação entre o desvio e a variância. Um algoritmo de aprendizagem com baixo viés deve ser flexível para que ele possa ajustar bem os dados. Entretanto, se o algoritmo de aprendizagem for muito flexível, ele irá condicionar cada conjunto de dados de treinamento de forma diferente, e, portanto, tem alta variação. Um aspecto chave de muitos métodos de aprendizado supervisionado é que eles são capazes de ajustar essa compensação entre o desvio e a variância tanto automaticamente quanto fornecendo parâmetros que possam ser ajustados [Geman et al., 2008].

- **Complexidade de Função e a Quantidade de Dados de Treinamento:** A segunda questão é a quantidade de dados de treinamento disponíveis relativos à complexidade da função (classificador ou regressão). Se a função é simples, então um algoritmo de aprendizado não flexível, com alto desvio e baixa variância, será capaz de aprendê-lo a partir de uma pequena quantidade de dados. Mas se a função verdadeira é altamente complexa (por envolver interações complexas como muitas características de entradas, por exemplo), então a função só será aprendida a partir de uma grande quantidade de dados de treinamento e usando um algoritmo de aprendizado flexível com baixo desvio e alta variância
- **Dimensionalidade do Espaço de Entrada:** A terceira questão é a dimensionalidade do espaço de entrada. Se os vetores de recursos de entrada tiverem dimensões muito altas, o problema de aprendizado pode ser difícil mesmo se a função verdadeira depende apenas de um pequeno número dessas características. Isso ocorre porque as muitas dimensões extras podem confundir o algoritmo de aprendizado e fazer com que ele tenha alta variação. Assim, a alta dimensionalidade de entrada, tipicamente, requer o ajuste do classificador para ter baixa variância e desvio elevado. Na prática, se as características dos dados de entrada forem removidas manualmente, provavelmente melhorará a precisão da função aprendida. Além disso, já muitos algoritmos para seleção de recursos que buscam identificar as características relevantes e descartar as irrelevantes. Esta é uma instância da estratégia mais geral de redução da dimensionalidade, que procura mapear os dados de entrada para um espaço de menor dimensão antes de executar o algoritmo de aprendizado supervisionado [Brodley and Friedl, 1999].
- **Ruído nos Valores de Saída:** A quarta questão é o grau de ruído nos valores de saída desejados (variáveis alvo da supervisão). Se os valores de saída desejados são frequentemente incorretos, seja por causa de erro humano ou erro em sensores), o algoritmo de aprendizado não deve tentar encontrar uma função que corresponda exatamente aos exemplos de treinamento. A tentativa de encaixar os dados com muito cuidado leva a um ajuste excessivo. Pode-se sobrepor mesmo quando não há erros de medição (ruído estocástico) se a função que de aprendizado for muito complexa para o modelo de aprendizagem. Em tal situação que parte da função de destino, que não pode ser modelado, corrompe seus dados de treinamento tem esse fenômeno conhecido como ruído determinístico. Quando um ou outro tipo de ruído está presente, é melhor ir com um desvio mais elevado e um estimador de variância mais baixo. Na prática existem várias abordagens para aliviar o ruído nos valores de saída tais como paragem antecipada para evitar a sobre alimentação bem como a detecção e remoção dos exemplos de treino ruidosos antes do treino do algoritmo de aprendizagem supervisionada. Existem vários algoritmos que identificam exemplos de treinamento ruidosos e, removendo os exemplos suspeitos de treinamento ruidoso antes do treinamento, diminuem o erro de generalização com significância [Smith and Martinez, 2011].

Outros fatores que são importantes na hora de escolher e aplicar um aprendizado de máquina são:

- **Heterogeneidade dos Dados:** Se os vetores de recurso incluem características de muitos tipos diferentes (discretos, discretos ordenados, contagens, valores contínuos), alguns algoritmos são mais

fáceis de aplicar do que outros. Muitos algoritmos, incluindo máquinas de vetores de suporte, regressão linear, regressão logística, redes neurais e métodos de vizinho mais próximo, requerem que os recursos de entrada sejam numéricos e escalados para intervalos similares. Métodos que empregam uma função de distância, tais como métodos de vizinho mais próximo e máquinas de suporte de vetores com kernel gaussiano, são, particularmente, sensíveis a isso. Árvores de decisão lidam, facilmente, com dados heterogêneos.

- **Redundância dos Dados:** Se os recursos de entrada contêm informações redundantes (como características altamente correlacionadas), alguns algoritmos de aprendizado (como regressão linear, por exemplo) irão apresentar um desempenho fraco devido a instabilidades numéricas. Estes problemas podem, muitas vezes, ser resolvidos através da imposição de uma regularização.
- **Presença de Interações e Não Linearidade:** Se cada um dos recursos faz uma contribuição independente para a saída, então algoritmos baseados em funções lineares e funções de distância geralmente funcionam bem. No entanto, se houver interações complexas entre os recursos, então algoritmos com árvores de decisão e redes neurais funcionam melhor, porque eles são especificamente projetados para descobrir essas interações. Métodos lineares podem ser aplicados também, mas deve especificar, manualmente, as interações quando usá-las.

2.2.1 Aprendizado em Lote

Desde os trabalhos mais pioneiros em Aprendizado de Máquina, duas visões de lidar com um problema de aprendizado coexistiram: lotes e online. Segundo [Faceli et al., 2011], o aprendizado de máquina por lotes utiliza pequenos conjuntos de dados que estão disponíveis em memória e um algoritmo aprende um modelo de decisão após, iterativamente, processar todos os dados. A lógica por trás dessa prática é que os exemplos são gerados aleatoriamente, de acordo com alguma distribuição estacionária. Os paradigmas do Aprendizado em Lotes assumem direta e indiretamente as seguintes restrições:

- Todos os dados de treinamento podem ser acessados em ordem para ajustar o modelo. A todo momento, o processo de aprendizado precisa acessar todos os dados e o acesso é imediato e completo.
- Não há restrição de tempo. Isso significa que se tem tempo suficiente para esperar o modelo ser completamente ajustado.
- O processo não muda. Uma vez que o modelo esteja ajustado, nenhuma atualização é necessária para resultados mais precisos.

Ao assumir todas essas restrições, a aplicabilidade do Aprendizado de Máquina é limitada significativamente. Muitas aplicações importantes de aprendizado, durante os últimos 50 anos, poderiam ter sido impossíveis de resolver sem uma suavização dessas restrições [Fontenla-Romero et al., 2013]). Alguns exemplos podem ser vistos a seguir:

- **Meios onde os dados chegam continuamente:** Nesse tipo de aplicação, dados chegam constantemente e o modelo precisa ser atualizado a todo momento senão o processo de aprendizado seria

redundante. Nesse caso, o acesso a todo o conjunto de dados pode ser tanto completo (o volume de dados cresce a todo momento) ou imediato (deve esperar a chegada de dados). De fato, isso pode ser questionado acerca da perda do conceito do treinamento do conjunto de dados como pode ser assumido que nunca terá acesso a ele desde que nunca se complete.

- **Aplicações em Aprendizado de Máquina:** Quando um grande número de dados disponíveis faz a centralização ser impossível (o acesso não pode ser completado) ou impraticável (devido à restrição de tempo) duas abordagens podem ser tomadas: paralelizar o processo de treinamento enquanto se mantém a formulação de lotes ou seguir uma formulação diferente capaz de otimizar o modelo utilizando subconjuntos de dados reduzidos.
- **O Processo por Baixo da Geração de Dados:** Nesse caso, o conjunto de treinamento inicial perde sua validade conforme o tempo passa e devido às mudanças de condições dos objetivos desejados. Portanto, um mecanismo para atualizar o modelo dado de forma que adaptem as novas condições é necessário.

2.2.2 Regressão

Segundo [Mohri et al., 2012], a regressão, em Aprendizado de Máquina, é um problema supervisionado de saídas contínuas e é utilizada quando o registro é identificado por um valor numérico. Assim é possível estimar o valor de uma determinada variável observando e analisando os valores das demais variáveis. Seguindo a análise, o modelo é capaz de “dizer” qual será o valor que a variável de saída assumiria.

Considerando X que representa o objeto de entrada e Y um subconjunto dos reais, uma distribuição D desconhecida sobre X onde cada objeto de entrada atende à $f : X \rightarrow Y$, dada como função de destino. Isso corresponde a cenário determinístico de aprendizado que é adotado para simplificar a apresentação. Em todos os problemas de aprendizado supervisionado, o aprendiz recebe um conjunto rotulado $S = ((x_1, y_1), \dots, (x_m, y_m)) \in (X \times Y)^m$ com x_1, \dots, x_m de acordo com D e $y_i = f(x_i)$ para todo $i \in [1, m]$. Desde que os rótulos sejam números reais, não é razoável esperar que o aprendiz preveja, com precisão, o rótulo correto. Em vez disso, pode-se exigir que suas previsões sejam próximas das corretas. Em regressão, a medida de erro baseia-se na magnitude da diferença entre o rótulo de valor real previsto e o verdadeiro, e não baseado na igualdade ou desigualdade destes dois valores.

Denota-se como $L : Y \times Y \rightarrow R_+$ a função de perda utilizada para mensurar a magnitude do erro. A função de perda mais comum utilizada em Regressão é o *squared loss* L_2 definida como $L(y, y') = |y' - y|^2$ para todo $y, y' \in Y$, ou, generalizado, uma L_p definida como $L(y, y') = |y' - y|^p$ para algum $p \geq 1$ e para todo $y, y' \in Y$.

Dada um conjunto de hipóteses H de funções mapeando $X \rightarrow Y$, o problema de regressão consiste em utilizar o conjunto rotulado S para encontrar hipóteses $h \in H$ com uma pequena perda já esperada ou uma generalização do erro $R(h)$ a respeito do alvo f :

$$R(h) = E[L(h(x), f(x))] \tag{2.2}$$

O erro de $h \in H$ ou a perda empírica é indicado como $\hat{R}(h)$ e definido como

$$\hat{R}(h) = \frac{1}{m} \sum_{i=1}^m L(h(x_i), y_i) \quad (2.3)$$

No caso mais comum onde L é a *squared loss*, isto representa o *mean squared error* de h no conjunto S . Quando a função de perda L é limitada por algum $M > 0$, tal que $L(y', y) = < M$ para todo $y, y' \in Y$ ou, mais estritamente, $L(h(x), f(x)) = < M$ para todo $h \in H$ e $x \in X$, o problema é referente ao problema de regressão de fronteira (*bounded regression problem*).

2.2.3 O Algoritmo Multilayer Perceptron

Segundo [Russell and Norvig, 2004], o Multilayer Perceptron é uma rede neural consistente de um conjunto de unidades. A formação da rede neural é dado por uma camada de entrada, uma camada de saída e camadas ocultas. O número de camadas ocultas fica a critério da modelagem. Um número maior de camadas ocultas pode se tornar uma vantagem pois garante a representação de um número maior de hipóteses. Cada unidade dessas camadas ocultas representam um parâmetro e a ligação entre elas é chamada de peso sináptico. O multilayer perceptron faz uso do algoritmo *backpropagation*¹ com o objetivo de ajustar os pesos da rede para minimizar medidas de erro durante a fase de treinamento [Rezende, 2001]. Na Figura 2.3 é exibida uma representação de uma rede neural com duas camadas ocultas.

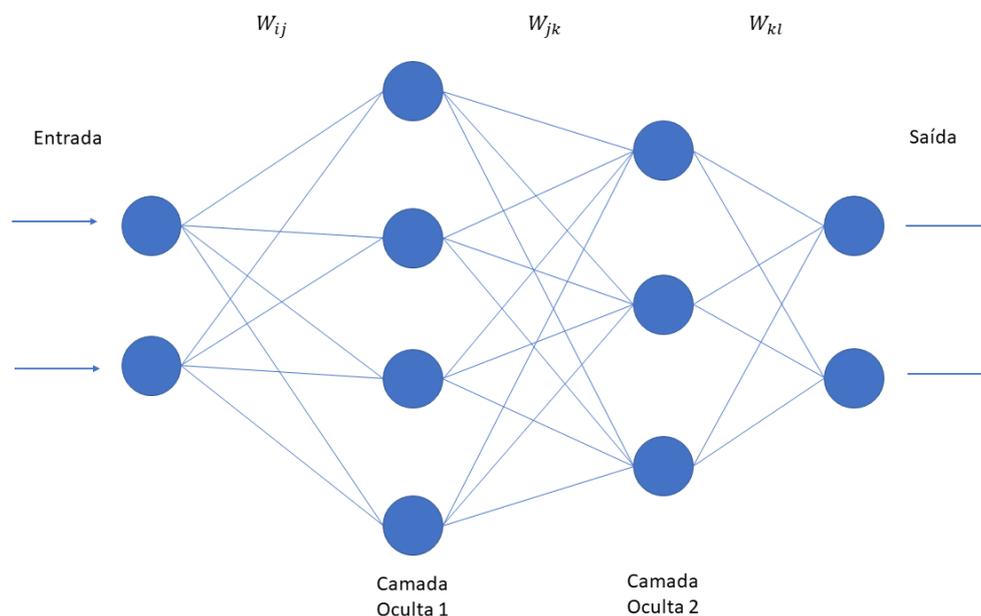


Figura 2.3: Um exemplo da distribuição de camadas do algoritmo Multilayer Perceptron. Fonte: Elaboração própria.

Logo a seguir é mostrado o Algoritmo 1 onde $W_{i,j}$ são os pesos sinápticos, x são os valores de entrada, a é a variável de ativação, g é tida como a função de ativação e \vec{y} é o vetor de saída:

¹O algoritmo de backpropagation é utilizado para obtenção do vetor gradiente associado ao peso da rede neural [Rezende, 2001].

Algoritmo 2: Fonte: [Riedmiller, 2010]

```
1 início
2   para todos os neurônios de entrada i faça
3     |  $a_i \leftarrow x_i$ ;
4   fim
5   para todos os neurônios ocultos e de saída i em ordem topológica faça
6     |  $Rede_i \leftarrow w_{i,0} \sum_j w_{i,j} a_j$ ;
7     |  $a_i \leftarrow g_{log}(Rede_i)$ ;
8   fim
9   para todos os neurônios de saída i faça
10    |  $\vec{y} \leftarrow a_i$ ;
11  fim
12  Retornar hipótese da rede neural;
13 fim
```

2.3 Aprendizado Online

O Aprendizado Online ou Aprendizado em Fluxo de Contínuo de Dados pode ser visto como processos estocásticos em que eventos ocorrem continuamente e independentemente um dos outros. A ideia principal de se trabalhar com fluxo contínuo de dados não impede o uso de dados da forma convencional: os dados podem ser transientes. Desde os primeiros desenvolvimentos criados, os métodos em Aprendizado Online têm encontrado suas aplicações em Controle Adaptativo, Filtro Adaptativo, Comunicação, Aprendizado de Máquina, Reconhecimento de Padrão, Redes Neurais etc. Atualmente, o Aprendizado Online está em voga também em outros campos de aplicação onde diferem da ideia original devido ao aumento dos dados produzidos nos últimos anos [Fontenla-Romero et al., 2013]. As redes sociais, por exemplo, geram dados na ordem de terabytes diariamente. E com essa quantidade de dados sendo produzida a todo momento, é preciso construir ferramentas que consigam lidar com tamanha quantidade de dados. Um alto volume de dados é armazenado hoje por toda o tipo de indústria e os modelos mais convencionais para tantos dados acabam sendo inviáveis. Assim como o volume, a velocidade acaba sendo um fator determinante pois os dados são recebidos e enviados com alta velocidade e demandam por algoritmos de aprendizado cada vez mais rápidos. E para se juntarem a esses dois, a variedade também é importante pois as fontes de dados podem diferir entre si e eles contribuem para a predição. Um bom algoritmo pode lidar com essa variedade sem muitos problemas [Fontenla-Romero et al., 2013].

Um algoritmo de predição simples como o *Random Forest*² executando aproximadamente 50 mil dados e 100 dimensões leva 10 minutos para ser executado em uma máquina de 12 GB de RAM. Se esse número de dados for aumentado para algo na casa dos milhões, será impossível de resolver usando essas máquinas mais comuns. Partindo desse pensamento, pode se pensar em ou usar uma

²É um algoritmo de aprendizagem de fácil utilização que consiste numa coleção de árvores de regressão aleatórias. [Biau, 2012].

máquina mais potente ou mudar o modo de como os algoritmos preditivos funcionam. Máquinas mais potentes, na maioria dos casos, não são viáveis [Srivastava, 2015]. De um ponto de vista matemático, um algoritmo online soluciona as restrições mencionadas anteriormente utilizando o seguinte princípio: Minimizar a função custo dada atualizando os parâmetros do modelo selecionando amostras de dados (ou um conjunto pequeno de dados) uma por vez. Basicamente, esses algoritmos de aprendizado compartilham uma formulação comum como na Equação 2.4

$$\hat{\theta}_t = \hat{\theta}_{t-1} + \Delta\hat{\theta}_{x_t} \quad (2.4)$$

Onde $\hat{\theta}_t$ é o vetor atualizado do parâmetro corrente, $\hat{\theta}_{t-1}$ é o parâmetro do vetor obtido do passo computacional anterior e o $\Delta\hat{\theta}_{x_t}$ é o ajuste ou atualização do termo. Essa metodologia de aprendizado é compartilhada por muitos algoritmos de Aprendizado Online na literatura, os quais podem ser encontrados sob uma variedade de termos como Aprendizado Online, Aprendizado de Filtro Adaptativo, Aprendizado Sequencial, Aprendizado Incremental e Aprendizado Estocástico.

Desenvolvimentos recentes nos algoritmos de Aprendizado Online têm focado no entorno dos seguintes algoritmos: performance Robusta ou Adaptativa, maximização de margens desde a introdução do suporte das máquinas vetoriais e a metodologia Kernel que tem sido usada para contornar problemas de classificação de linearidade não separável e problemas de regressão não linear. Muitos algoritmos têm sido propostos nos últimos anos, mas o Aprendizado Online não tem uma teoria de aprendizado consolidada e o entendimento das suas propriedades não é completa como no Aprendizado por Lotes. Portanto, é esperado que as pesquisas recentes ainda possam prover métodos melhores e mais completos do processo de Aprendizado Online num futuro próximo [Fontenla-Romero et al., 2013]. Nesses contextos, o objetivo para os algoritmos online consiste em manter um modelo de decisão preciso e consistente com o estado atual do processo que gera dados. Para isso, algoritmos de aprendizado incrementais são requisitados por poderem modificar o modelo atual sempre que uma nova informação for requisitada. Na presença de uma distribuição não estacionária, o sistema de aprendizado deve incorporar mecanismos de esquecimento, permitindo o descarte de informações e conceitos que já não representem o estado atual do problema. O aprendizado via fluxo de dados requer algoritmos de aprendizado incremental que funcionem com recursos computacionais limitados e que sejam capazes de levar em consideração, não só mudanças, mas também evoluções dos conceitos a aprender [Faceli et al., 2011]. A seguir, será mostrado os algoritmos que foram utilizados neste trabalho. As principais diferenças entre o fluxo de dados e o modelo relacional convencional incluem:

- Os elementos em um fluxo contínuo de dados chegam on-line;
- O sistema não tem controle sobre a ordem em que os elementos de dados chegam em um fluxo de dados ou através do fluxo;
- Fluxo de dados são potencialmente ilimitados em tamanho;
- Uma vez que um elemento do fluxo de dados é processado, ele é descartado ou arquivado. Isso não pode ser revertido facilmente, a menos que isso seja explicitado no armazenado em memória, que é pequeno em relação ao tamanho do fluxo de dados.

No modelo de fluxo descrito em [Muthukrishnan, 2005], os elementos de entrada $f_1, f_2, \dots, f_i, \dots$ chegam sequencialmente, item por item e descrevem uma função fundamental F .

Sistema de gestão de Banco de Dados	Sistema de gestão de Fluxo de Dados
Relações persistentes	Fluxos transientes
Uma requisição por vez	Requisições contínuas
Acesso aleatório	Acesso sequencial
Plano de acesso determinado pelo processador de requisição	Características de dados e padrões de chegada imprevisíveis

Tabela 2.1: Comparação entre sistemas de Banco de Dados e Fluxo de dados. Fonte: [Gama, 2010]

Modelos de fluxos se diferem em como a f_i descreve F . Em relação a esses modelos, nós podemos dividi-los entre:

- **Modelo de Inserção:** Uma vez que o elemento f_i é visto, ele não pode ser modificado;
- **Modelo de Inserir-Deletar:** Elementos f_i podem ser deletados ou atualizados;
- **Modelo Aditivo:** Cada f_i é um incremento para $F|j| = F|j| + f_i$.

Na Seção 2.3.1, são descritos os desafios no Aprendizado Online que é a base deste trabalho; e na Seção 2.4 sobre Avaliação.

2.3.1 Desafios no Aprendizado Online

O Aprendizado a partir de fluxos contínuos de dados produzidos à uma alta velocidade em ambientes não estacionários e dinâmicos necessita de novas técnicas de amostragem, capacidade de processar dados na mesma velocidade em que são disponibilizados, capacidade de esquecer dados não úteis e novos algoritmos. As propriedades desejáveis para algoritmos que desenvolvem esse tipo de aprendizado são:

- **Incrementabilidade:** Em um sistema incremental, a expressão do conceito vai sendo construída exemplo a exemplo e implica constantemente a revisão do conceito até então formulados. Um novo exemplo, em algumas ocasiões, pode causar um rearranjo da expressão do conceito. A expressão do conceito vai se modificando à medida que novos exemplos vão se tornando disponíveis.
- **Aprendizado em tempo real:** No Aprendizado Online, as informações chegam a todo momento. Um algoritmo que esteja preparado para resolver lidar com toda a quantidade de dados em tempo real é um fator decisivo para melhor gestão dessas sequências de dados.
- **Capacidade para processar exemplos constantemente com poucos recursos:** Visto que aplicações com algoritmos que necessitem cada vez mais de recursos do computador, a demanda começa a ficar inviável dependendo do quanto cresce o algoritmo e com isso a necessidade de gerar algoritmos que sejam mais eficientes com um consumo menor de recursos passa a ser necessário.

- **Acesso limitado a exemplos já processados:** Algo muito importante no processamento de dados nesses algoritmos é o descarte de dados que já não são úteis para o processamento. Como a quantidade de recursos é importante a ser considerado, se determinada quantidade de dados não é mais interessante para o algoritmo, o descarte dessa sequência deve ser realizado de forma que suavize a utilização de recursos.
- **Detecção e Adaptação do Modelo:** O algoritmo que segue o fluxo contínuo de dados deve estar sempre pronto para mudanças visto que novos conjuntos estarão chegando a todo momento. Uma adaptação a esse recebimento deve estar sempre sendo criada para melhor funcionamento do algoritmo.

2.3.2 O Algoritmo FIMT-DD

Segundo [Gama et al., 2014], o *Fast Incremental Model Tree with Drift Detection* é um algoritmo de árvore de decisão para regressão em fluxo contínuo de dados que segue o método de divisão e conquista. O algoritmo começa com uma folha vazia e lê os exemplos conforme eles chegam. Cada exemplo é levado à folha onde as estatísticas são atualizadas. O *grace period* é um atributo da árvore de decisão que determina quantos exemplos serão processados até que a divisão seja realizada. Após a quantidade de instâncias alcançar o valor do *grace period*, o algoritmo encontra a melhor parte de cada atributo e, então, faz o ranqueamento dos atributos de acordo com a escala de avaliação. Se o critério de separação é satisfatório, então o melhor atributo é particionado, criando duas folhas, uma para cada ramo da divisão. Após a chegada de novas instâncias a uma divisão criada recentemente, elas são passadas ao longo dos ramos do teste na divisão dos seus valores. A detecção de uma mudança nos testes gera uma atualização com cada exemplo do fluxo. Se for detectada uma mudança, uma adaptação da estrutura da árvore é feita. Se não for encontrada, as estatísticas são atualizadas e os atributos classificados de forma a verificar se o critério de decisão é satisfeito ou não. E esse último procedimento é feito até que acabe os registros que não influenciaram a mudança de decisão. O Algoritmo 3 demonstra o procedimento de execução da árvore.

Algoritmo 3: Algoritmo FIMT-DD. Fonte: [Gama et al., 2014]

Entrada: σ Probabilidade definida pelo usuário e N_{min}

Saída: Árvore modelo

```

1 início
2   Começa com a folha vazia (raiz);
3   repita
4     Leia o próximo exemplo;
5     Analise o exemplo pela árvore até a folha;
6     Atualize os testes de detecção de mudanças de caminho;
7     se Mudança é detectada então
8       | Adapte a árvore do modelo;
9     fim
10    senão
11      | Atualize as estatísticas da folha;
12      | repita
13        | Encontre a melhor divisão por atributo;
14        | Classifique os atributos usando a mesma medida de avaliação;
15        | se Critério de divisão é satisfeito então
16          | | Faça uma divisão no melhor atributo;
17          | | Faça dois novos ramos conduzindo as folhas;
18        | | fim
19        | até Não ter mais  $N_{min}$  registros vistos na folha;
20      | fim
21    até Não ter mais exemplos no fluxo de dados;
22 fim

```

Segundo [Wisesa et al., 2016], a vantagem de se usar o FIMT-DD, em comparação com os outros algoritmos de regressão online, é que ele se adapta melhor ao fluxo de dados devido ao fato dele oferecer resultados de regressão rapidamente e com acurácia alta. Além de ser muito bom em se adaptar às mudanças do conjunto de dados.

2.3.3 O Algoritmo Perceptron Online

Segundo [Rezende, 2001], a versão online do Multilayer Perceptron sofre ajustes nos pesos sinápticos baseando de exemplo em exemplo. Assim o custo da função a ser minimizado é, instantaneamente, o erro energético. Se considerar uma porção de N exemplos de treinamento arranjados em ordem $x(1), d(1), x(2), d(2), \dots, x(N), d(N)$. O primeiro exemplo é apresentado à rede e os ajustes dos pesos são feitos usando o método do gradiente. O segundo exemplo é apresentado à rede para mais ajustes. Esse processo segue até o último exemplo fornecido. Infelizmente, esse procedimento vai de encontro ao Aprendizado Online. Dado um conjunto de condições iniciais, uma única realização da curva de aprendizado é obtido

ao plotar gráfico do valor final do erro energético e o número de porções usadas no treinamento. Assim como no aprendizado em lote, a curva de aprendizado do Aprendizado Online é computado pela média das realizações sobre um número grande de condições iniciais escolhidas aleatoriamente. Entretanto, a curva de aprendizado do Aprendizado Online será diferente da versão em lote. Dados exemplos de treinamento apresentados à rede de maneira aleatória, o uso do Aprendizado Online faz com que a busca no espaço multidimensional seja estocástica naturalmente. E é por isso que, muitas vezes, o Aprendizado Online é tratado como *método estocástico*. Quando os dados de treinamento são redundantes, o algoritmo reduz essas ocorrências pois os registros são apresentados um a um. [Rezende, 2001]. Por essas características, o Perceptron Online se sobressai diante do Multilayer

2.4 Avaliação

Ao observar toda a problemática que envolve o Aprendizado Online, é possível identificar três dimensões que influenciam no processo de aprendizado:

- **Espaço:** A memória disponível é fixa;
- **Tempo de aprendizado:** A taxa que os exemplos de processo chegam;
- **Poder de generalização:** Quão efetivo o modelo é ao capturar o verdadeiro conceito oculto.

O fato é que modelos de decisão que evoluem com o tempo tem implicações fortes nas técnicas de avaliação que avaliam a efetividade do processo de aprendizado. Outro aspecto relevante é a resiliência à superposição: cada exemplo é processado uma única vez.

2.4.1 Problemas de Avaliação

Sistemas de aprendizado geram representações compactas do que está sendo observado. Eles devem estar prontos para melhorar com a experiência e, continuamente, auto modificar seu estado interno. A avaliação é utilizada em dois contextos: dentro do sistema de aprendizado para a avaliar as hipóteses, e como um *wrapper* sobre o sistema de aprendizado para estimar a aplicabilidade de um algoritmo em particular em um dado problema. Três aspectos fundamentais são:

- O que são os objetivos das tarefas de aprendizado?
- Quais são as métricas de avaliação?
- Como conceber os experimentos para estimar as métricas de avaliação?

Para tarefas preditivas de aprendizado, classificação e regressão, o objetivo é induzir a função $\hat{y} = f(\vec{x})$. A dimensão mais relevante é o erro de generalização. Esse erro é um estimador da diferença entre \hat{f} e a f desconhecida, e a perda estimada do que pode ser esperado quando o modelo é aplicado ao modelo para exemplos futuros.

2.4.2 Concepção dos Experimentos de Avaliação

Um aspecto na criação dos experimentos que não foi abordado de forma conveniente é que os algoritmos de aprendizagem são executados em dispositivos computacionais que tem poder computacional limitado. Por exemplo, os algoritmos de aprendizado existentes assumem que os dados comportam na memória disponível o que não é aceito na presença de fluxo de dados. Esse problema se torna ainda mais relevante quando a análise de dados deve ser feita dependendo da situação. Em redes de sensores por exemplo, onde o fluxo de dados vem em alta velocidade e os recursos são muito limitados. Pouquíssimos algoritmos limitam a memória. Uma exceção notável é o *Very Fast Decision Tree* (VFDT) [Domingos and Hulten, 2000] que pode economizar memória congelando folhas menos promissoras sem que a memória chegue ao seu limite. No aprendizado por lotes utilizando um conjunto de dados de treinamento finitos, cross-validation e variantes são métodos padrões para avaliar sistemas de aprendizado. Já no contexto Online, onde os dados são potencialmente infinitos, a distribuição gera exemplos e os modelos de decisão evoluem com o passar do tempo e o cross-validation e outras estratégias não se aplicam.

2.5 Métricas

Nesse trabalho, as métricas utilizadas foram o SMAPE (*Symmetric Mean Absolute Percentage Error*) e a Correlação de Pearson.

2.5.1 Correlação de Pearson

Segundo [Filho and Júnior, 2009], a Correlação de Pearson é uma medida associativa entre duas variáveis X e Y . Considerando n como o tamanho do conjunto de dados, x_i e y_i são valores de seus respectivos conjuntos de dados, \bar{x} e \bar{y} são as médias do seus respectivos conjunto de dados e S_x e S_y são os valores de desvio padrão do seu respectivo conjunto de dados, a Equação 2.5 é dada da seguinte forma:

$$r_{xy} = \frac{1}{n-1} \sum \left(\frac{x_i - \bar{X}}{S_x} \right) \left(\frac{Y_i - \bar{Y}}{S_y} \right) \quad (2.5)$$

Isto é, os valores do coeficiente indicam se as variáveis analisadas possuem algum grau de semelhança. Segundo [Mukaka, 2012], a classificação de valores do coeficiente podem ser dispostos da seguinte forma:

- r entre 0 e 0,30 é dito como desprezível
- r entre 0,30 e 0,50 é dito como fraca
- r entre 0,5 e 0,7 é dito como moderada
- r entre 0,7 a 0,9 é dito como forte
- r maior que 0,9 é dita como muito forte

2.5.2 SMAPE

Segundo [Armstrong, 1985], o SMAPE é uma modificação do *MeanAbsolutePercentageError* onde o divisor é metade da soma dos valores atuais e os preditos. Possui limite mínimo e máximo, fornecendo valores entre 0% e 200%. Considerando que sua fórmula é dada por:

$$SMAPE = \frac{100\%}{n} \sum_{j=1}^n \frac{|P_{ij} - T_j|}{|P_{ij}| + |T_j| \frac{1}{2}} \quad (2.6)$$

Onde n é o tamanho do conjunto de dados, $P_{(ij)}$ é o valor predito pelo programa i e T_j é o valor atual.

2.5.3 Técnicas de Avaliação

Para avaliar um modelo no contexto de fluxo de dados, duas alternativas são viáveis:

- **Holdout:** Os dados, nessa métrica de avaliação, são separados em dois subconjuntos onde um é treinado e o outro é utilizado para testes. Com essa separação, o desempenho é avaliado periodicamente utilizando porções de fluxo de dados diferentes para treinamento e testes do modelo [Bifet et al., 2011]. Esse método é bem aproveitado quando se tem um conjunto de dados muito grande, caso contrário, o erro calculado na predição pode sofrer muitas variações [Kohavi, 1995].
- **Prequential:** Onde o erro do modelo é computado a partir de uma sequência de exemplos onde, para cada exemplo no fluxo, o modelo atual faz uma predição baseada apenas nos valores dos atributos exemplo.

Nesse trabalho, foi utilizado o modelo Holdout, tendo em vista que a avaliação foi feita periodicamente e os registros do fluxo de dados utilizados foram divididos em porções: 66% para treinar o modelo e 34% para testá-lo.

Capítulo 3

Revisão da Literatura

Nesse capítulo é apresentada a pesquisa bibliométrica realizada no contexto de aprendizado de máquina online, considerando o consumo de gás como fonte energética.

3.1 Metodologia

Para realizar o estudo bibliométrico, a amostra foi pesquisada no portal da CAPES de setembro de 2017 a outubro de 2017. O Portal de periódicos da CAPES é uma das soluções mais utilizadas no meio acadêmico nacional para busca de diversos artigos, periódicos, livros etc. em várias bases de dados. Para realizar uma pesquisa aprofundada, a base SCOPUS e a Science Direct foram utilizadas. O tema dessa monografia foi dividido em filtros para procurar similaridades já pesquisadas. A pesquisa bibliométrica foi auxiliada com a ferramenta Parsifal [Parsifal, 2015].

O Parsifal é uma ferramenta online desenvolvida para apoiar pesquisadores na revisão sistemática da literatura. Pesquisadores distribuídos geograficamente podem trabalhar juntos num mesmo projeto compartilhado designando o protocolo e a condução da pesquisa. Assim como provê uma forma de documentar todo o processo, a ferramenta ajuda a lembrar o que é importante durante a revisão sistemática [Parsifal, 2015].

Utilizando a ferramenta, o processo iniciou com o preenchimento do PICOC. O PICOC é um método usado para descrever os cinco elementos para uma pesquisa. O acrônimo indica:

- População (*Population*): “Quem?”, é a população em que a evidência é coletada.
- Intervenção (*Intervention*): “O quê ou como?”, é a intervenção aplicada ao estudo empírico.
- Comparação (*Comparison*): “Comparado ao quê?”, é a comparação em que a intervenção é comparada.
- Efeito (*Outcome*): “O que você está tentando melhorar / fazer?”, é o efeito que o experimento deveria não só ser estatisticamente significativo, mas também ser significativo de um ponto de vista prático.

- Contexto (*Context*): “Em que tipo de organização / circunstâncias?”, é o contexto do estudo que deve ser definido, em que é uma visão estendida da população. Incluindo se é realizado na academia ou na indústria, em que segmento industrial e, também, os incentivos para os assuntos tratados [Parsifal, 2015].

O pensamento, por baixo, é que todos os cinco elementos são importantes numa pesquisa focada por evidências e cada mudança nos elementos levam a evidências diferentes e também a um efeito diferente [Center for Evidence-Based Management, 2016]. No caso da bibliometria dessa monografia, foram selecionados termos para a *população*, *intervenção* e *comparação*.

- Population = “energy consumption”.
- Intervention = “online learning”
- Comparison = “prediction”

Em seguida foi realizado o preenchimento das Questões de Pesquisa (*Research Questions*). Essa parte foca na identificação dos estudos primários, a extração de dados dos estudos e a análise. Portanto, as questões de pesquisa devem ser bem pensadas e formuladas. Um dos aspectos que devem ser considerados na construção das questões é o PICOC.

Neste passo, foram realizadas cinco questões de pesquisa:

- Quais métodos / algoritmos / ferramentas de aprendizado online são utilizados para predição de consumo de energia?
- Quais atributos são utilizados para o aprendizado?
- Qual a diferença entre os problemas encontrados de consumo de energia?
- Qual(is) base(s) de dados é (são) usada(s)? Dados reais ou simulados? Está disponível na internet?
- Como é feita a modelagem do rótulo para predição?

Com essas perguntas, ao realizar a pesquisa, é possível analisar as fontes encontradas e respondê-las propriamente.

A próxima etapa da construção da pesquisa se dá com a inserção de palavras chave e sinônimos. As palavras chave são, basicamente, extraídas da *população*, *intervenção*, *comparação* e *efeito*. Cada palavra chave deve estar relacionada a um acrônimo do PICOC. Depois de determinar todas as palavras chave relevantes, deve-se relacioná-las a todos os seus sinônimos que possam tornar a string de busca mais ampla. Nesse trabalho, a adição de sinônimos se deu com o “energy consumption” onde foi inserido o sinônimo “gas consumption”. Como os dados obtidos para o experimento são de gás em forma de energia, a intenção é que a pesquisa encontre alternativas entre energia e gás.

A próxima etapa é a formação da string de busca. Nessa parte do planejamento, a ferramenta irá pegar todos os dados preenchidos para *população*, *intervenção*, *comparação* e *efeito* para compor a string. A regra de ouro para montar uma boa string de busca é separar as palavras chave e os sinônimos com o operador lógico OU e separar os termos do PICOC utilizados com parênteses e utilizando o operador

E. O exemplo a seguir demonstra como que a string de busca fica organizada. Na geração da string de busca dessa monografia se deu com a *população, intervenção e comparação*. Dessa forma:

(*“energyconsumption”OR“gasconsumption”*)AND(*“onlinelearning”*)AND(*“prediction”*)

Depois de obter a string de busca, restou selecionar as fontes de pesquisa. O Parsifal conta com diversas fontes para realizar as pesquisas. As fontes que foram selecionadas para a pesquisa nesse trabalho são a “Scopus” e a “Science@Direct”.

E o planejamento se encerra com os critérios de seleção. Para essa análise, os critérios de inclusão são:

- Artigo de Conferência;
- Artigo de Periódico;
- Tema próximo ao da monografia;
- Trata do problema de predição de energia ou gás usando aprendizado online.

Já os critérios de exclusão são:

- Trabalho que não seja de conferência ou periódico;
- O foco da predição não é próximo ao tema;
- Tema foge de IA e/ou Predição;
- Usa aprendizado de máquina ou outras técnicas de IA que não aprendizado online.

No processo de condução é feito uma análise de cada umas das pesquisas em cada base escolhida e onde será feita a seleção de das publicações mais próximas do tema pesquisado neste trabalho.

3.2 Análise dos Trabalhos Selecionados

Na Figura 3.1 é possível observar que o número de publicações dentro dos termos pesquisados foram aumentando nos últimos anos, com exceção da queda que teve em 2015. O grande crescimento entre os anos de 2016 e 2017 dão a entender que a pesquisa na área tem crescido e que é possível que o número aumente ainda mais no de 2018 que já contam com cinco publicações previstas.

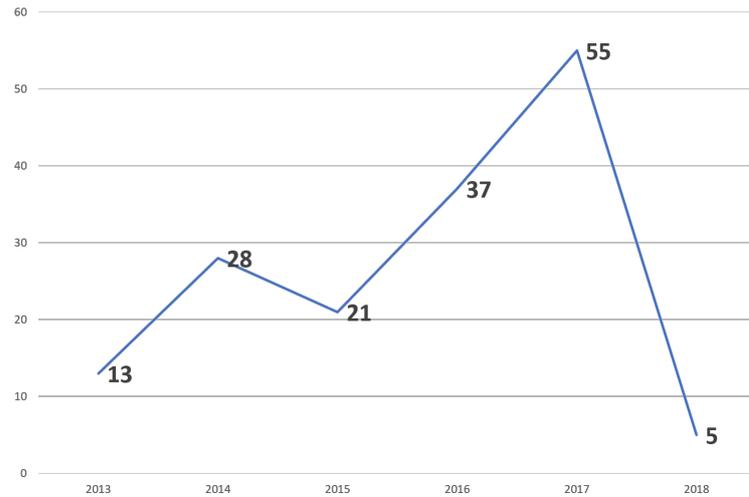


Figura 3.1: Publicações realizadas entre 2013 até 2017. Fonte: Elaboração própria.

A Tabela 3.1 mostra os autores que mais tiveram publicações sobre o tema proposto. Os autores que mais se destacaram possuem 3 publicações, são eles Al-Hashimi, Ghahramani e Merrett.

Autor	Nº de Publicações
Al-Hashimi, B.M.	3
Ghahramani, A.	3
Merrett, G.V.	3
Aduda, K.	2
Becerik-Gerber, B.	2
Chen, D.	2
Chen, Y.L..	2
Das, A.	2
Jiang, H.	2
Maeda-Nunez, L.A.	2

Tabela 3.1: Autores com mais publicações entre 2013 e 2017. Fonte: Elaboração própria.

A Figura 3.2 é o gráfico que relaciona o quantitativo de publicações com as áreas de pesquisa sobre o tema. Aqui contém apenas os dados da Base SCOPUS que fornecem esses filtros. As áreas com mais destaque são Ciência da Computação, Engenharia e Matemática.

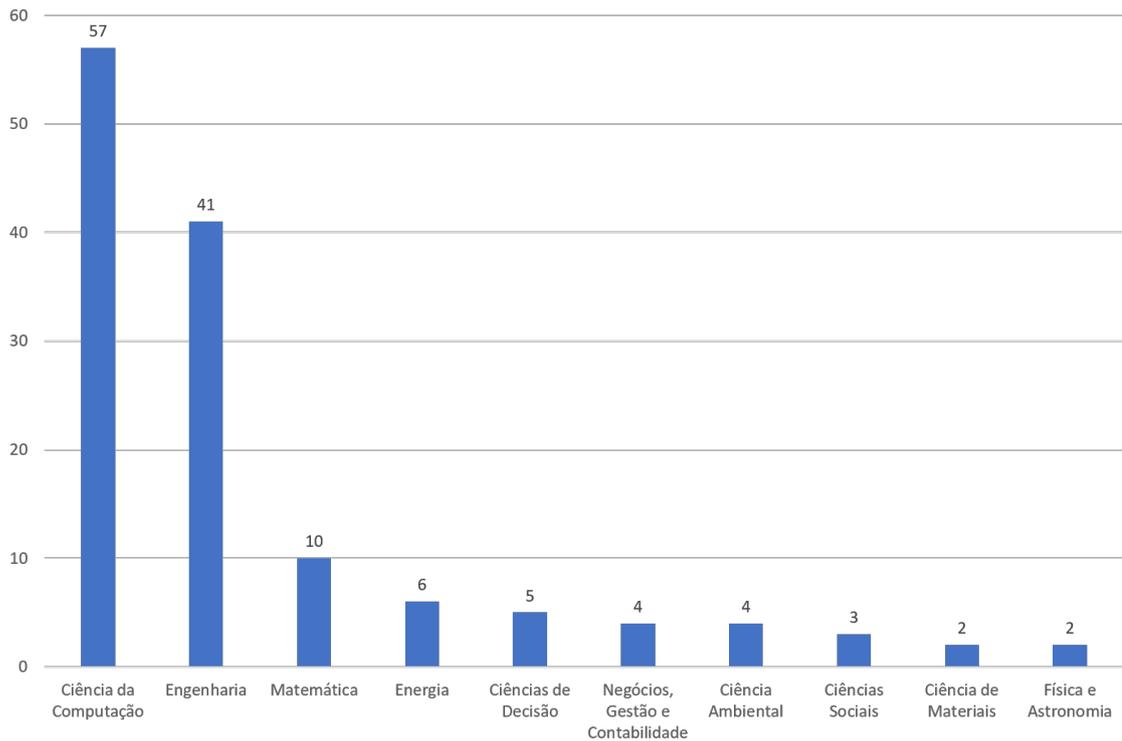


Figura 3.2: Publicações por área. Fonte: Elaboração própria.

A Tabela 3.2 demonstra os países que mais obtiveram publicações na base SCOPUS. Estados Unidos e China se destacam como os países que mais registraram publicações (57 e 45 respectivamente). O Brasil possui 1 publicação sobre o tema.

País	Nº de Publicações
Estados Unidos	24
China	22
Reino Unido	9
Índia	6
Canadá	5
Espanha	5
Alemanha	4
Holanda	4
França	3
Grécia	3
Brasil	1

Tabela 3.2: Países com mais publicações. Fonte: Elaboração própria.

A Tabela 3.3 informa os idiomas das publicações na base SCOPUS. O Inglês domina as publicações com quase 96% das publicações.

A Tabela 3.4 apresenta as 8 palavras chave que mais são encontradas sobre o tema abordado.

Idioma	Nº de Publicações
Inglês	82
Chinês	4

Tabela 3.3: Idiomas com mais publicações. Fonte: Elaboração própria.

As palavras que mais se destacaram nas buscas foram: “Utilização de Energia” e “Eficiência de Energia”. Adicionadas à Tabela, os termos Aprendizado de Máquina e Aprendizado de Máquina Online tiveram 7 e 5 publicações respectivamente.

Keyword	Nº de Publicações
Utilização de Energia	34
Eficiência Energética	33
Inteligência Artificial	12
Redes Neurais	11
Previsão	10
Gerenciamento de Energia	9
Sistemas de Aprendizado	9
Otimização	9
Aprendizado de Máquina	7
Aprendizado de Máquina Online	5

Tabela 3.4: Keywords com mais publicações. Fonte: Elaboração própria.

3.2.1 Descrição da Literatura

A Tabela 3.5 apresenta as 5 publicações mais relevantes de acordo com os artigos encontrados e aceitos com base nos critérios de inclusão. Os artigos aqui descritos possuem informações que foram consideradas pertinentes quanto ao tema e até mesmo com informações que puderam ajudar no desenvolvimento de trabalho de certa forma.

Nº	Título	Autores	Ano
1	On-line learning of indoor temperature forecasting models towards energy efficiency	Zamora-Martínez, F. and Romeu, P. and Botella-Rocamora, P. and Pardo, J.	2014
2	Up by their bootstraps: On-line learning in Artificial Neural Networks for CMP uncore power management	Won, J.-Y. and Chen, X. and Gratz, P. and Hu, J. and Soteriou, V.	2014
3	An online learning approach for quantifying personalized thermal comfort via adaptive stochastic modeling	Ali Ghahramani and Chao Tang and Burcin Becerik-Gerber	2015
4	Towards next-generation heterogeneous mobile data stream mining applications: Opportunities, challenges, and future research directions	Muhammad Habib ur Rehman and Chee Sun Liew and Teh Ying Wah and Muhammad Khurram Khan	2017
5	Smart grid load forecasting using online support vector regression	Vrablecová, P. and Ezzeddine, A.B. and Rozinajová, V. and Sárík, S. and Sangaiah, A.K.	2017

Tabela 3.5: Relação das publicações mais relevantes. Fonte: Elaboração própria.

A seguir, é apresentado um resumo sobre as publicações selecionadas na 3.5, na ordem de relevância.

- [Zamora-Martínez et al., 2014] - Nesse trabalho foi descrito uma visão geral a respeito da monitoração e detecção da energia solar em uma casa. Os sistemas AVAC¹ correspondem a 53,89% do consumo total da casa. Nesse trabalho, uma abordagem do Aprendizado Online foi apresentada com a integração de modelos preditivos. Foram utilizados e comparados modelos de regressão diferentes, um Bayesiano online e um Gradient Descent (GD) online. Segundo os resultados, os dois conseguiram ser promissores apesar dos modelos mais simples (como RNAs de uma ou duas camadas ocultas) serem mais eficientes. Foi verificado que os modelos preditivos online podem divergir pois podem alcançar seu melhor desempenho num determinado tempo, mas se perdendo conforme o treinamento continua. Algumas estratégias podem ser adotadas como adaptação da velocidade de aprendizado dependendo de como o modelo se comporta e memorização de padrões.
- [Won et al., 2014] - Esse trabalho busca melhorar a eficiência energética do *uncore* (partes não núcleo) dos Chips Multi Processadores (CMP) focando na tensão dinâmica e a escala de frequência

¹AVAC é uma sigla para Aquecimento, Ventilação e Ar-Condicionado, o que consiste na tecnologia destinada ao conforto ambiental interior.

(DVFS) já que trabalhos anteriores centravam apenas nos núcleos. Foi considerado que todo o uncore compreendem uma única relação de tensão / frequência, assim os dados não precisam experimentar atrasos de sincronização. Foram utilizadas RNAs offline e duas adaptações própria online: *feedback adaptation* e *self-sharpening*. A primeira funciona como um aprendizado de máquina supervisionado onde o alvo é a frequência do uncore obtida de forma online. Já o segundo é baseado na observação que o RNA deveria ter idealmente. Um valor de saída 1 enquanto os outros deveriam ter valor 0. Foi concluído que as abordagens são benéficas para o CMP, conseguindo reduzir o atraso de energia em 27%.

- [Ghahramani et al., 2015] - Nesse artigo foi apresentado um procedimento matemático para modelar o conforto térmico de indivíduos com o Aprendizado Online. O conforto térmico foi quantificado com base nas condições em que o indivíduo se sente confortável ou não. É considerado Online pois ele aprende a cada ponto de entrada coletado do indivíduo. Os dados brutos foram transformados em três conjuntos: desconfortavelmente quente, confortável e desconfortavelmente legal. Utilizou-se o método de máxima verossimilhança para tratar das probabilidades e utilizaram uma rede bayesiana com a combinação das distribuições feitas. Os resultados foram muito satisfatórios, visto que eles conseguiram uma precisão acima de 70% na média e foi relativamente superior a outras técnicas de classificação utilizadas na comparação. Concluíram que a abordagem pode ser utilizada para alterar problemas convencionais de otimização como os de conforto, energia, HVAC.
- [ur Rehman et al., 2017] - Nesse trabalho foi apresentado a revisão da literatura a cerca de MDSM (Mobile Device Systems Management) onde foi constatado que as plataformas de MDSM são encontradas em configurações de múltiplas topologias. Os modelos de comunicação incluem dispositivos de computação e sistemas que tenham diferentes fatores. Esses sistemas precisam liberar múltiplas funcionalidades como particionamento, gestão de dados e processamento leve e pesado de dados. Muitas literaturas destacam que essas aplicações têm de lidar com seis fatores críticos quando se trata de fluxo de dados: tamanho do fluxo de dados, velocidade dos dados, fusão inicial dos dados, seleção de métodos de pré-processamento, estratégias de desenvolvimento do modelo de aprendizado e seleção dos algoritmos de mineração de dados. E lidando com esses fatores críticos, fica mais fácil de conquistar alguns objetivos como eficiência energética.
- [Vrablecová et al., 2017] - Neste artigo é trabalhado o uso de vetores de suporte em smart grids. Por ser uma parte das Cidades Inteligentes, provê novas oportunidades em eficiência energética. Considerando que medidores inteligentes produzem fluxo de dados, os métodos de processamento técnicas de mineração que são diferentemente dos clássicos. Entretanto, essa chegada contínua de dados dos medidores demandam a transição do método de aprendizado em lotes para o incremental e online. Apresentando uma aplicação com capacidades de precisão e acurácia altas e uma comparação entre os métodos de previsão de dados públicos onde encontraram um número muito pequeno de artigos que comparassem as mesmas técnicas.

3.3 Considerações Finais

O contexto de Aprendizado Online ainda é recente no ramo da pesquisa e, dependendo do foco, os trabalhos podem ser ainda mais escassos. Como visto durante a revisão, os artigos, principalmente nacionais, não abordam tanto o mesmo objetivo que o desse trabalho em questão. Muitos dos algoritmos utilizados nos artigos analisados não se encaixavam na proposta dessa monografia, visto que os procedimentos não são os mesmos e serviram para verificar comportamentos dos dados no aprendizado de máquina online, com exceção do trabalho do [Won et al., 2014] que explora os algoritmos de RNAs adaptados. Uma conclusão geral e positiva dentre todos os trabalhos foi a eficiência dos algoritmos de aprendizado online que sempre foram positivos numa comparação com os algoritmos ditos offline.

Capítulo 4

Experimentos Realizados

Nesse trabalho serão utilizadas técnicas de Aprendizado Online para poder prever, baseado em dados abertos, o consumo de gás como energia de uma empresa.

4.1 Ferramentas Utilizadas

- **WEKA:** O WEKA é uma coleção de algoritmos de Aprendizado de Máquina para tarefas de mineração de dados. Contém ferramentas para preparação, classificação, regressão, clusterização, visualização e regras de associação para mineração de dados. É uma ferramenta de código aberto [Hall et al., 2009].
- **MOA:** O MOA (*Massive Online Analysis*) é um dos frameworks mais populares para trabalhar com mineração de fluxo de dados. Sua coleção de algoritmos de aprendizado de máquina incluem algoritmos de Classificação, Regressão, Clusterização, Detecção de Outliers além de ferramentas para avaliação. Relacionado ao projeto WEKA, o MOA também é escrito em Java, enquanto é escalado para problemas de grandes demandas [Bifet et al., 2010].

4.2 Modelagem

Nas seções que seguem, são descritas as etapas da construção do método proposto com detalhes. Na Seção 4.2.1 é descrito como foi feita a coleta dos dados. Na Seção 4.2.2 é relatado como foi feito o pré processamento de dados, bem como o processo de construção de novos atributos a partir dos que já continham nos dados. Na Seção 4.3 é descrito todo o processo da análise experimental, a preparação de dos dados para aplicação dos algoritmos online e em lote nas ferramentas mencionadas anteriormente.

4.2.1 Coleta de dados

Os dados foram obtidos através do portal governamental do Reino Unido. Os dados disponíveis são de cinco edifícios comerciais da Bristol onde é descrito seu consumo a cada trinta minutos. Cada edifício teve seu consumo anual armazenado, entre os dias 09 de março de 2013 a 05 de março de 2014, assim como

cada um dos seus três medidores inteligentes. Na Tabela 4.1 é mostrado uma faixa de dados coletados em sua forma bruta:

Building	Utility	Unit	Meter no.	Date	00:30	01:00	01:30
Building 1	Gas	KWh	Meter 1	09/03/2013	6.226	5.6034	14.3198
Building 1	Gas	KWh	Meter 1	10/03/2013	12.452	12.7633	11.8294
Building 1	Gas	KWh	Meter 1	11/03/2013	14.3198	12.7633	14.3198
Building 1	Gas	KWh	Meter 1	12/03/2013	16.4689	17.4328	19.6119

Tabela 4.1: Descrição dos dados. Fonte: Elaboração própria.

Ao analisar os dados dos cinco prédios, foi verificado que apenas o primeiro edifício possuía dados de mais de um medidor. Então, o foco do trabalho passou a se concentrar nesse prédio que possuía três medidores ao invés dos outros. Em uma análise primária desses dados, foram feitas médias de todo o consumo anual de um dos medidores para que fosse possível ter uma dimensão do que seria necessário verificar no que tange ao consumo energético de gás. Na Figura 4.1 pode ser visto que, no consumo diário, os dias úteis representam a maior taxa de consumo, o que era esperado por ser dias de maior movimentação na empresa.

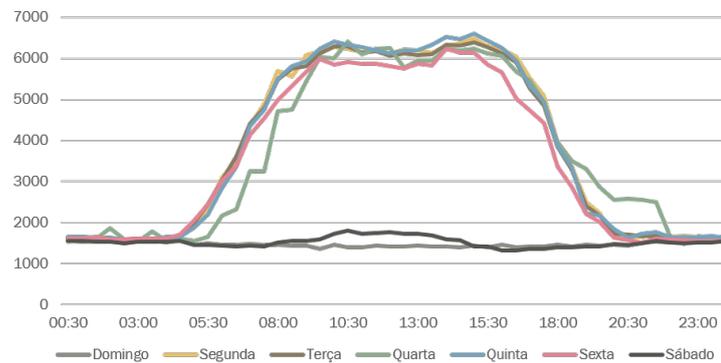


Figura 4.1: Consumo Médio Diário. Fonte: Elaboração própria.

Numa visão menos detalhada, a Figura 4.2 mostra o consumo semanal, isto é, a média entre os "dias de semana" e "fins de semana" como comparativo. Nessa figura, também como esperado, a linha correspondente aos dias de semana demonstra seu maior consumo com picos na casa de 8000 KWh.

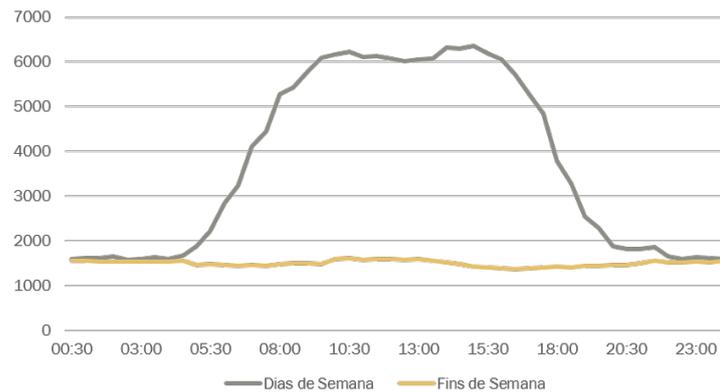


Figura 4.2: Consumo Médio Semanal. Fonte: Elaboração própria.

E, ainda, a Figura 4.3 mostra o consumo para os feriados do Reino Unido. São exibidos 7 feriados nacionais e que possuem um impacto significativo no consumo, visto que são dias bastante movimentados. É possível reparar que a linha de maior consumo é referente à virada de ano, e a menor de todas é no feriado de Páscoa. O *Boxing Day* é o dia seguinte ao Natal que é responsável pelo “desempacotar” dos presentes.

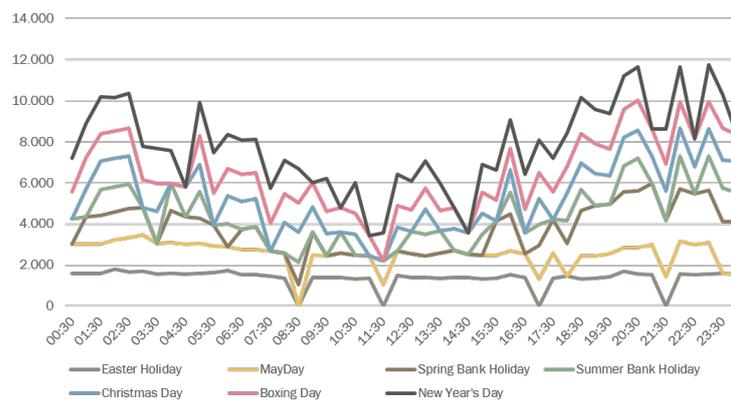


Figura 4.3: Consumo nos Feriados. Fonte: Elaboração própria.

Para que pudesse dar seguimento para o pré-processamento dos dados, foi necessário realizar a correlação entre os medidores do edifício. A intenção de realizar esse cálculo foi de tentar determinar se era necessário fazer a análise de cada um dos medidores, caso eles não possuam correlação entre si, ou se seria feito de apenas um deles, caso possuam correlação entre si. Para isso, foi utilizado o Coeficiente de Correlação de postos de Spearman. Na Tabela 4.2 mostra os valores de correlação e fica claro que não possuem correlação, já que os valores não são iguais a um. Isso indica que nossa análise será feita em cima de cada um dos três medidores.

4.2.2 Pré Processamento

Nesta etapa, foi realizada toda a engenharia de atributos. Com base nos dados coletados, conforme exibido no Quadro 4.1, alguns desses foram descartados como **Building, Utility, Unit, Meter no.** e

Coeficientes	Valor de ρ
Coeficiente de Spearman (M1 e M2)	-0,07004
Coeficiente de Spearman (M1 e M3)	-0,22439
Coeficiente de Spearman (M2 e M3)	-0,43590

Tabela 4.2: Correlação entre os medidores. Fonte: Elaboração própria.

Date. E, para uma melhor análise, foram criados os seguintes atributos:

- **Dia de Semana:** Binário, com valores de 1 para dia da semana;
- **Fim de Semana:** Binário, com valores de 1 para finais de semana;
- **Estação:** Com valores de 1 a 4 indicando em qual estação de ano a sequência está, verão, outono, inverno e primavera respectivamente.
- **Feriado:** Binário, com valores 1 para os feriados. Esse atributo foi pautado em cima dos feriados nacionais do Reino Unido, já que os dados foram coletados da empresa alocada nessa região.

A Tabela 4.3 mostra um pequeno trecho dos dados já nessa etapa de pré-processamento.

00:30	01:00	...	00:00	Dia de Se- mana	Fim de Se- mana	Estação do Ano	Feriado
6.22	56.03	...	115.18	0	1	4	0
12.45	127.63	...	143.19	1	0	4	0
143.19	174.32	...	140.08	1	0	4	0
164.98	136.97	...	164.98	1	0	4	0
171.21	143.19	...	161.87	1	0	4	0

Tabela 4.3: Exemplo dos dados durante o pré-processamento. Fonte: Elaboração própria.

Para dar seguimento na análise dos dados coletados, foram geradas figuras que demonstram o comportamento deles em diferentes situações. As Figuras 4.4, 4.5 e 4.6 mostram o desvio padrão para os dias de semana baseados nos medidores. Ao observá-las, o desvio padrão mais linear é o do medidor 2, enquanto o medidor 3 segue um ritmo de queda do meio para o final.

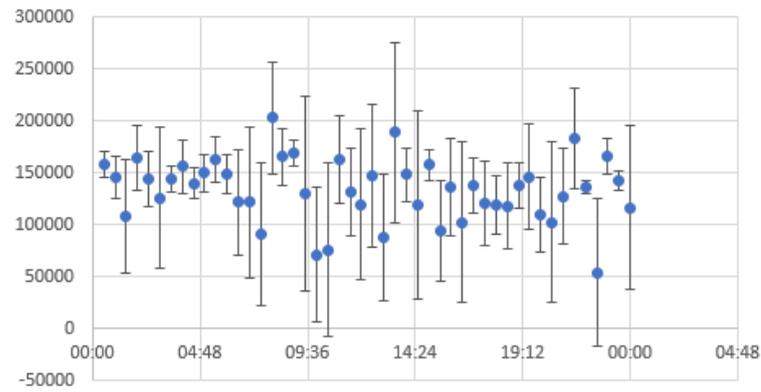


Figura 4.4: Média e Desvio Padrão dos dias de semana do Medidor 1
. Fonte: Elaboração própria.

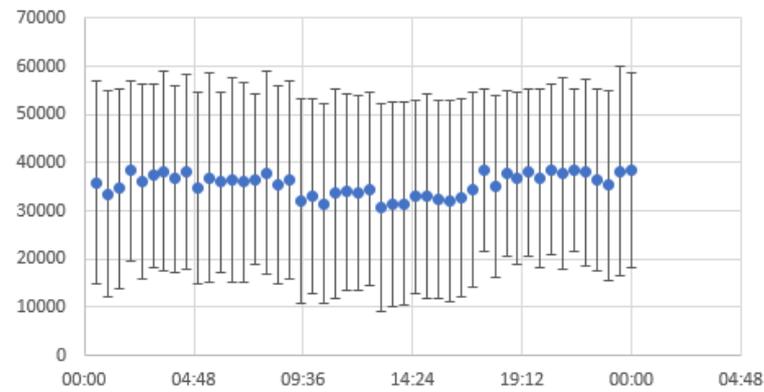


Figura 4.5: Média e Desvio Padrão dos dias de semana do Medidor 2. Fonte: Elaboração própria.

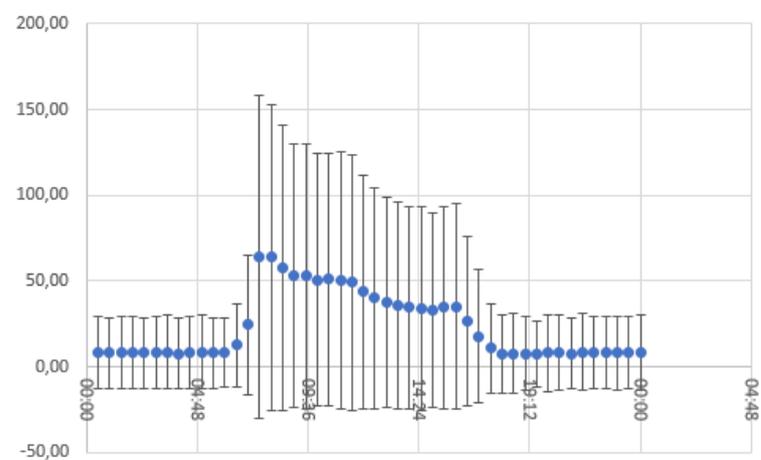


Figura 4.6: Média e Desvio Padrão dos dias de semana do Medidor 3. Fonte: Elaboração própria.

As Figuras 4.7, 4.8 e 4.9 exibem o desvio padrão para os finais de semana, onde é possível verificar que os dados do medidor 1 são os mais variados.

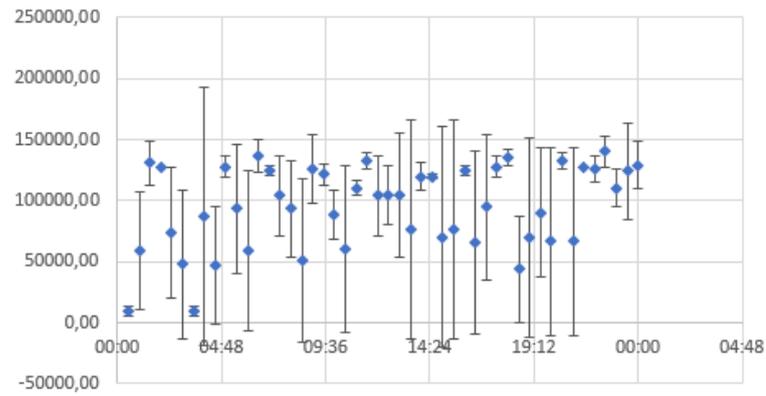


Figura 4.7: Média e Desvio Padrão dos fins de semana do Medidor 1. Fonte: Elaboração própria.

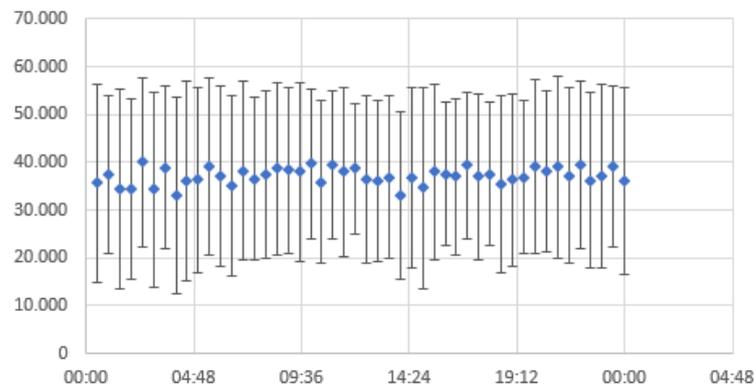


Figura 4.8: Média e Desvio Padrão dos fins de semana do Medidor 2. Fonte: Elaboração própria.

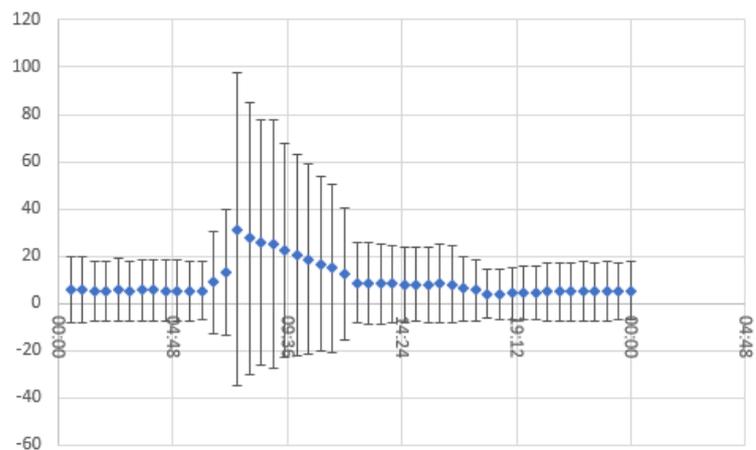


Figura 4.9: Média e Desvio Padrão dos fins de semana do Medidor 3. Fonte: Elaboração própria.

Durante essa fase, verificou-se que o uso dos feriados não trazia impacto significativo para o estudo, visto que puderam ser absorvidos pelas medidas de dias de semana e fins de semana.

4.3 Análise Experimental

Dando seguimento à etapa de pré processamento, foi feita a construção de arquivos *.arff* que foram utilizados pelas ferramentas *MOA* e *WEKA*. A estruturação de um dos arquivos utilizados é demonstrado através da Figura 4.10.

```
@RELATION medidor_tres_janela_quinze

@ATTRIBUTE medida_um      REAL
@ATTRIBUTE medida_dois   REAL
@ATTRIBUTE medida_tres   REAL
@ATTRIBUTE medida_quatro REAL
@ATTRIBUTE medida_cinco  REAL
@ATTRIBUTE medida_seis   REAL
@ATTRIBUTE medida_sete   REAL
@ATTRIBUTE medida_oito   REAL
@ATTRIBUTE medida_nove   REAL
@ATTRIBUTE medida_dez    REAL
@ATTRIBUTE medida_onze   REAL
@ATTRIBUTE medida_doze   REAL
@ATTRIBUTE medida_treze  REAL
@ATTRIBUTE medida_quatorze REAL
@ATTRIBUTE valor_predito  REAL
@ATTRIBUTE dia_de_semana REAL
@ATTRIBUTE fim_de_semana REAL
@ATTRIBUTE estacao_do_ano REAL

@DATA
48.400,35.200,49.500,47.300,45.100,40.700,41.800,52.800,44.000,50.600,14.300,25.300,45.100,46.200,46.200,0,1,4
35.200,49.500,47.300,45.100,40.700,41.800,52.800,44.000,50.600,14.300,25.300,45.100,46.200,46.200,46.200,0,1,4
49.500,47.300,45.100,40.700,41.800,52.800,44.000,50.600,14.300,25.300,45.100,46.200,46.200,46.200,46.200,0,1,4
47.300,45.100,40.700,41.800,52.800,44.000,50.600,14.300,25.300,45.100,46.200,46.200,46.200,46.200,46.200,0,1,4
45.100,40.700,41.800,52.800,44.000,50.600,14.300,25.300,45.100,46.200,46.200,46.200,46.200,46.200,46.200,0,1,4
40.700,41.800,52.800,44.000,50.600,14.300,25.300,45.100,46.200,46.200,46.200,46.200,46.200,46.200,46.200,0,1,4
41.800,52.800,44.000,50.600,14.300,25.300,45.100,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,0,1,4
52.800,44.000,50.600,14.300,25.300,45.100,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,48.400,0,1,4
44.000,50.600,14.300,25.300,45.100,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,48.400,46.200,0,1,4
50.600,14.300,25.300,45.100,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,48.400,46.200,37.400,0,1,4
14.300,25.300,45.100,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,48.400,46.200,37.400,0,1,4
25.300,45.100,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,48.400,46.200,37.400,37.400,46.200,0,1,4
45.100,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,48.400,46.200,37.400,37.400,46.200,45.100,0,1,4
46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,48.400,46.200,37.400,37.400,46.200,45.100,31.900,0,1,4
46.200,46.200,46.200,46.200,46.200,46.200,46.200,46.200,48.400,46.200,37.400,37.400,46.200,45.100,31.900,45.100,0,1,4
46.200,46.200,46.200,46.200,46.200,46.200,48.400,46.200,37.400,37.400,46.200,45.100,31.900,45.100,46.200,0,1,4
```

Figura 4.10: Arquivo de extensão *.arff*. Fonte: Elaboração própria.

Para que fosse feita uma análise numa faixa temporal, visto que o conjunto contém valores a cada trinta minutos, foram feitos três novos conjuntos de dados consistindo de janelas de tempo:

- Janela de 5t, onde t corresponde a uma sequência de 5 valores, consistindo de um período de 2:30h;
- Janela de 10t, onde t corresponde a uma sequência de 10 valores, consistindo de um período de 5:00h;
- Janela de 15t, onde t corresponde a uma sequência de 15 valores, consistindo de um período de 7:30h.

Para dar seguimento a essa etapa, conforme uso padrão para algoritmos de regressão do aprendizado de máquina, foi estabelecido que seriam utilizados dois terços dos dados para treinamento e o restante para testar o modelo, seguindo a técnica Holdout como visto na Seção 2.5.

Na ferramenta *MOA*, foi feita a execução dos dois algoritmos mencionados anteriormente, o Perceptron Online e o FIMT-DD. Para esse último, foi feita uma avaliação, enquanto a construção dos seus modelos, sobre o *grace period*, utilizando os valores 20 a 200, com intervalos de 20 unidades. A escolha e análise deles foram feitas para avaliar se o comportamento do modelo, como um todo, sofria

variações com esses valores. E por terem variações expressivas, para o trabalho, foi utilizado o *grace period* com valor 100 pois foi o que, na análise, teve o melhor comportamento.

Após a execução de cada um dos modelos, o MOA faz a geração de um arquivo com os valores preditos, nos arquivos originais, e os reais para a execução feita de cada um dos registros. Com base nessas informações, é possível realizar o cálculo das métricas.

Na Figura 4.11 é possível verificar a Correção de Pearson para os três medidores com suas respectivas janelas de tempo na execução do FIMT-DD. Em quase todas as medidas, a correlação é tida como forte pois os valores oscilam entre 0.7 e 0.9. Entretanto, a janela de tempo 15t do medidor três foi a que se comportou da pior forma. Com apenas 0.15, sua correlação é tida como desprezível. Numa análise mais geral, é possível afirmar que a esquematização da janela de 15t foi a que teve a pior correlação das 3 para cada medidor. Também fica evidente que a janela de tempo 10t foi a que melhor se comportou na execução do experimento dos três medidores.

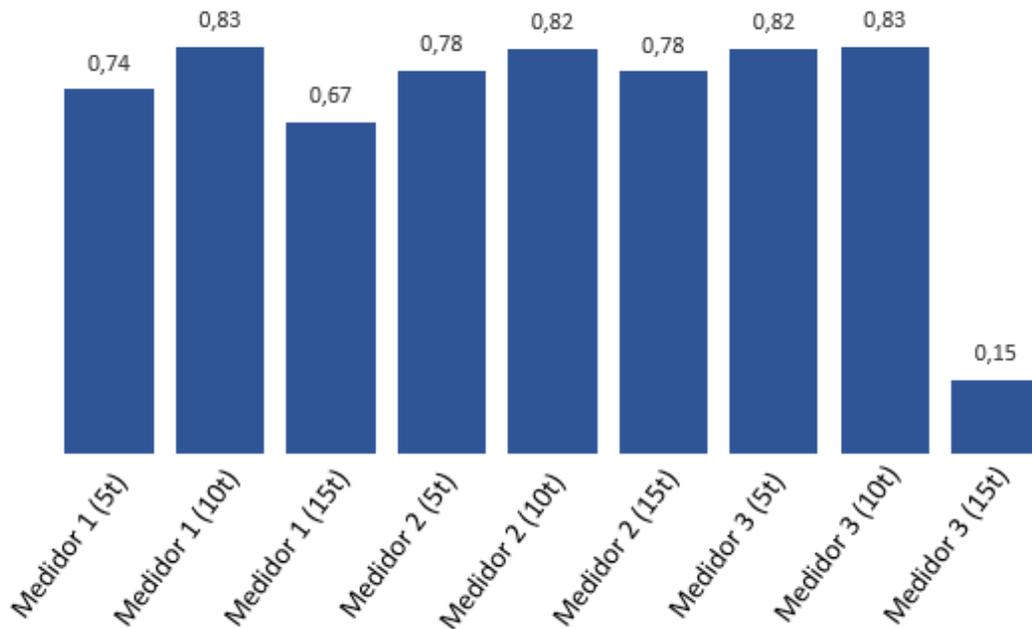


Figura 4.11: Correlação de Pearson para o Algoritmo FIMT-DD. Fonte: Elaboração própria.

Para a execução dos algoritmos em lote, a ferramenta WEKA foi utilizada com os algoritmos Regressão Linear e Multilayer Perceptron. A seguir temos três Figuras que fazem um comparativo entre as três janelas de tempo para cada um dos três medidores quanto à sua Correlação de Pearson. Em todos os três, o FIMT-DD se mostrou melhor que os demais, já as redes neurais sofreram em ambas as situações para se ajustarem aos fluxos de dados dos medidores.

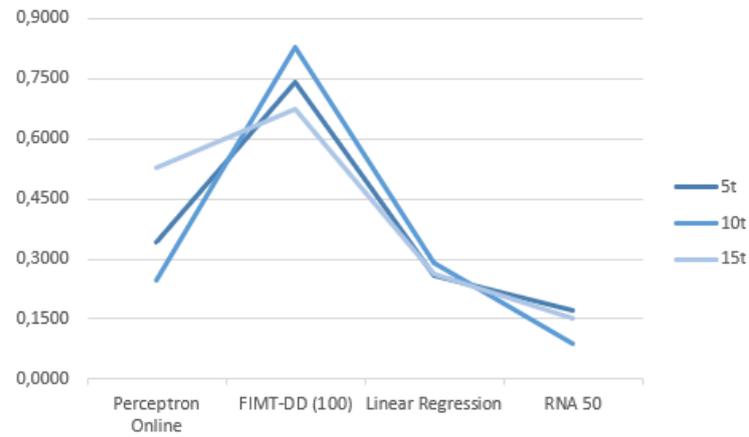


Figura 4.12: Correlação de Pearson do Medidor 1. Fonte: Elaboração própria.

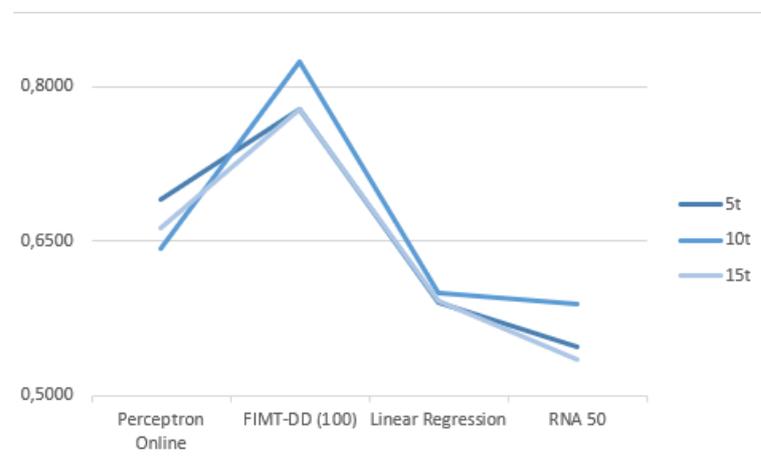


Figura 4.13: Correlação de Pearson do Medidor 2. Fonte: Elaboração própria.

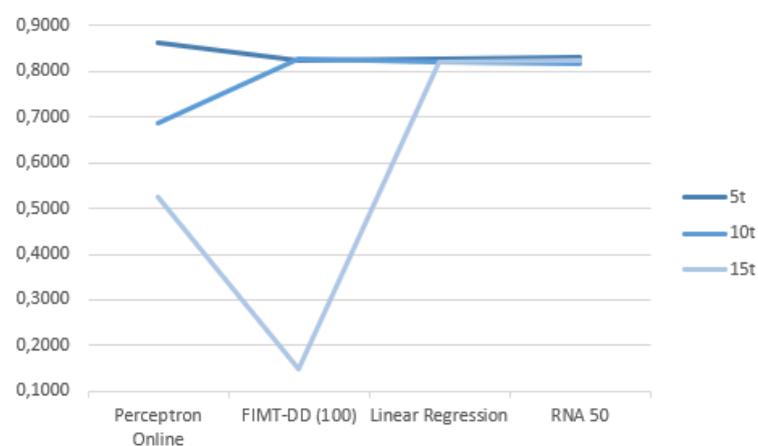


Figura 4.14: Correlação de Pearson do Medidor 3. Fonte: Elaboração própria.

Nas tabelas abaixo é possível ver os resultados das métricas para janela de 5t com suas respectivas métricas. Para essas tabelas, fica evidente o comportamento irregular dos algoritmos para os valores do Medidor 1, com ressalva para o FIMT-DD que obteve um bom resultado para todos os medidores. Num

comparativo entre as redes neurais, o Perceptron Online se saiu melhor do que o Multilayer Perceptron em todos os casos.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.3410	0.6898	0.8621
SMAPE	0.128	0.2903	0.9446

Tabela 4.4: Resultados do Perceptron Online para a janela de 5t. Fonte: Elaboração própria.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.1721	0.5472	0.8303
SMAPE	1.1437	0.3748	1.0251

Tabela 4.5: Resultados do Multilayer Perceptron janela de 5t. Fonte: Elaboração própria.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.7406	0.7786	0.8248
SMAPE	1.1476	1.05170	1.5214

Tabela 4.6: Resultados do FIMT-DD para a janela de 5t. Fonte: Elaboração própria.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.2604	0.5903	0.8258
SMAPE	0.3587	0.3197	1.3857

Tabela 4.7: Resultados do FIMT-DD para a janela de 5t. Fonte: Elaboração própria.

Analisando o resultado das métricas da janela 10t, o FIMT-DD continuou se sobressaindo diante dos outros, numa visão geral. O Multilayer Perceptron enfrentou dificuldades na predição dos dados do Medidor 1, em específico, pela grande variação dos dados coletados, além de ter um desempenho inferior ao Perceptron Online, num comparativo entre as redes neurais.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.2451	0.6424	0.6875
SMAPE	0.3512	0.3568	1.0363

Tabela 4.8: Resultados do Perceptron Online para a janela de 10t. Fonte: Elaboração própria.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.0871	0.5890	0.8171
SMAPE	1.1355	1.1875	1.6612

Tabela 4.9: Resultados do Multilayer Perceptron janela de 10t. Fonte: Elaboração própria.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.8274	0.8239	0.8276
SMAPE	1.0333	1.0095	1.4866

Tabela 4.10: Resultados do FIMT-DD para a janela de 10t. Fonte: Elaboração própria.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.28899	0.5998	0.8203
SMAPE	0.3039	0.2924	1.2827

Tabela 4.11: Resultados da Regressão Linear para a janela de 10t. Fonte: Elaboração própria.

Já para a janela 15t, o algoritmo FIMT-DD obteve um desempenho ruim comparado aos demais para o Medidor 3, mas foi o que melhor se adequou ao conjunto de dados nos outros medidores. Ainda para o Medidor 3, os algoritmos em Lote foram muito bem com esse conjunto de dados com pouca variação de informações. Mas, ainda, um pouco abaixo do que os algoritmos online numa visão geral.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.5264	0.6633	0.5247
SMAPE	0.1116	1.0363	0.8637

Tabela 4.12: Resultados do Perceptron Online para a janela de 15t. Fonte: Elaboração própria.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.1506	0.5345	0.8238
SMAPE	0.2560	0.5506	1.7779

Tabela 4.13: Resultados do Multilayer Perceptron janela de 15t. Fonte: Elaboração própria.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.6734	0.7778	0.1499
SMAPE	1.1936	1.0702	1.5165

Tabela 4.14: Resultados do FIMT-DD para a janela de 15t. Fonte: Elaboração própria.

Sumário	Medidor 1	Medidor 2	Medidor 3
Correlação de Pearson	0.2612	0.5924	0.8191
SMAPE	0.3277	0.2944	1.1851

Tabela 4.15: Resultados da Regressão Linear para a janela de 15t. Fonte: Elaboração própria.

A seguir é possível visualizar a convergência dos dados reais e preditos para os três medidores, tanto com o FIMT-DD quanto com o Perceptron Online. O intuito desses gráficos é mostrar o comportamento do algoritmo para absorver o conjunto de dados e prever o valor com base neles e na modelagem feita para cada um.

Nas Figuras 4.15, 4.16 e 4.17 que representam a convergência dos resultados do algoritmo FIMT-DD, fica evidente a discrepância de informações do Medidor 1 que tinha uma variação de dados muito grande num curto espaço de tempo. Já para os outros medidores fica mais evidente o bom comportamento do modelo e a proximidade de valores com a realidade, com destaque para o ótimo comportamento do modelo com os dados do Medidor 2. As cores em azul indicam o valor real e as cores em laranja indicam o valor predito pelo algoritmo.

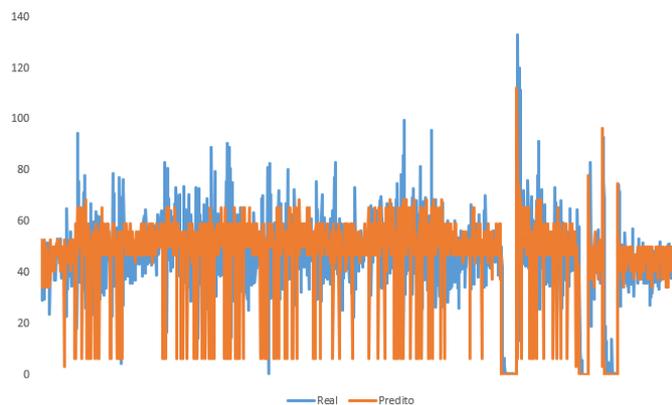


Figura 4.15: Medidor 1 - FIMT-DD. Fonte: Elaboração própria.

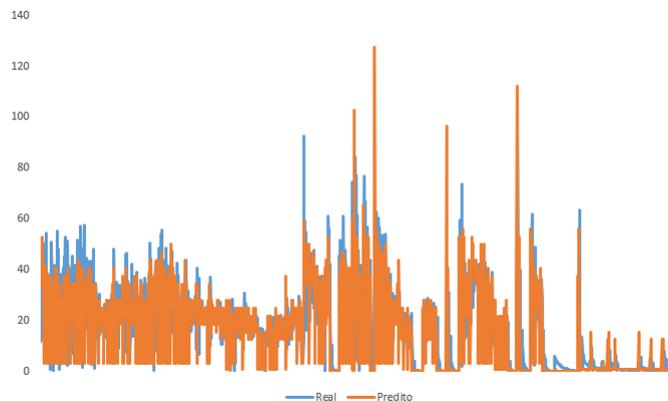


Figura 4.16: Medidor 2 - FIMT-DD. Fonte: Elaboração própria.

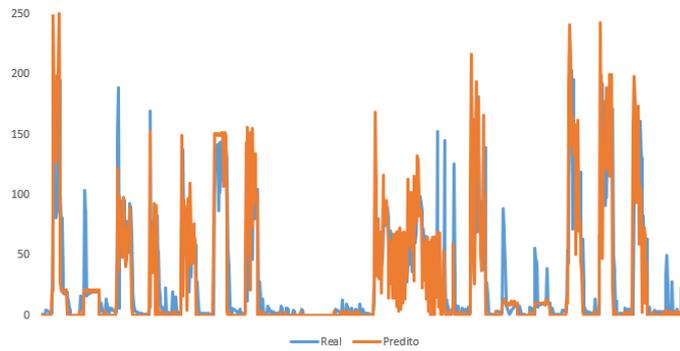


Figura 4.17: Medidor 3 - FIMT-DD. Fonte: Elaboração própria.

Analisando as Figuras 4.18, 4.19 e 4.20, do Perceptron Online, é possível observar sua melhor atuação no Medidor 3 que também se justifica ao observar os valores das tabelas anteriores, já que os dados desse medidor em específico são mais homogêneos.

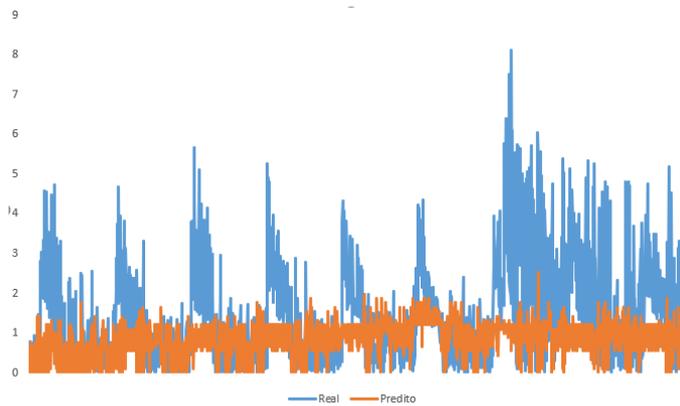


Figura 4.18: Medidor 1 - Perceptron Online. Fonte: Elaboração própria.

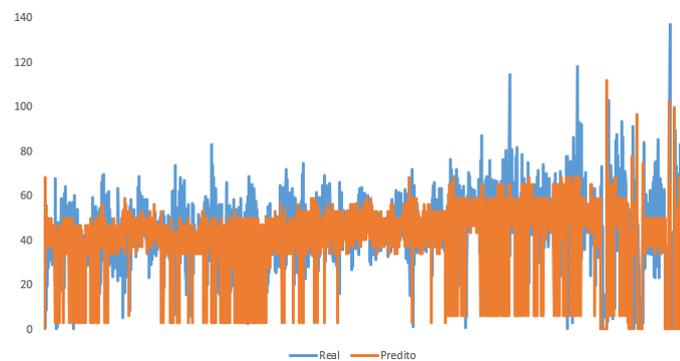


Figura 4.19: Medidor 2 - Perceptron Online. Fonte: Elaboração própria.

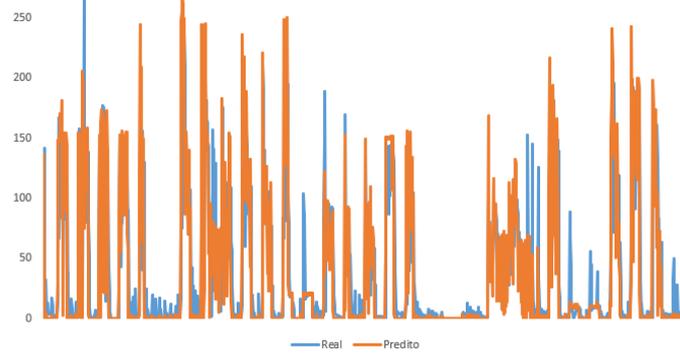


Figura 4.20: Medidor 3 - Perceptron Online. Fonte: Elaboração própria.

Em uma visão geral, com base nos gráficos de convergência, ficou um pouco mais evidente a melhor performance do algoritmo FIMT-DD com relação ao Perceptron Online para lidar com esses três conjuntos de dados.

Capítulo 5

Conclusão e Trabalhos Futuros

O consumo de energia, independente da fonte, é algo que movimenta a humanidade como um todo. A previsão de consumo se torna algo muito necessário no que tange à tendência de Cidades Inteligentes. O uso de algoritmos de Aprendizagem Online demonstra uma melhor adaptação ao fluxo contínuo de dados do que os algoritmos de aprendizado em lote. Neste trabalho foi mostrado uma vertente de previsão para consumo de um tipo de energia.

A modelagem do problema foi feita baseada em janelas de tempo e os algoritmos utilizados foram os de regressão. Com a comparação com os algoritmos de aprendizagem em lote, o trabalho demonstrou a eficiência dos algoritmos online em lidar com um conjunto de dados de fluxo contínuo. Em uma análise geral, os algoritmos que usam Redes Neurais tiveram um desempenho muito abaixo, tanto no aprendizado em lote quanto no online. Na comparação direta entre os algoritmos online, o FIMT-DD demonstrou superioridade comparado com o Perceptron Online, salvo o caso da janela de $15t$ do medidor 3. E a Regressão Linear teve performance melhor que o Multilayer Perceptron na maioria dos casos. No caso da árvore de decisão, o seu bom desempenho era esperado pois, segundo [Faceli et al., 2011], as árvores são mais suscetíveis a mudanças de conceito (*concept drift*). Os dados obtidos estão disponíveis no conjunto de dados abertos do governo britânico. As ferramentas MOA e WEKA foram utilizadas para qualificar os regressores e as métricas Correlação de Pearson e SMAPE foram utilizados para avaliação dos resultados obtidos com os regressores. No geral, o FIMT-DD obteve melhor resultado nos três medidores com suas janelas de tempo, com exceção da janela de tempo $15t$ do Medidor 3, justamente pela sua facilidade em lidar com a heterogeneidade dos dados. No aprendizado em lote, a Regressão Linear se mostrou adaptável ao conjunto de dados ficando um pouco mais atrás que a árvore de decisão e superior ao Multilayer Perceptron. O Perceptron Online foi superior no confronto com a sua versão em lote e teve a melhor performance na janela de tempo para $15t$ do terceiro medidor. Foi possível verificar, também, durante a análise bibliométrica que a quantidade de trabalhos que exploraram o tema de aprendizado online ainda é pequena, principalmente na quantidade de pesquisas nacionais.

Para os trabalhos futuros, várias situações podem ser colocadas na análise. A expansão de um cenário, como, por exemplo, um grande centro de um bairro onde é possível obter diversos dados a respeito de consumo de um tipo de energia considerando as condições climáticas, com muitos medidores espalhados

dentre casas, edifícios, indústrias e afins. Com esse grande número de informações, a possibilidade de estender a metodologia coletando de um conjunto de bairros para verificar a variação de consumo dentre cada um deles considerando não só a região, como também a densidade populacional, número e tipos de medidores aplicados à edifícios domésticos e comerciais etc. Uma nova pesquisa pode ser feita para a Revisão da Literatura quanto à string de busca com o Aprendizado de Fluxo de Dados, já que foi uma palavra que era associada ao Aprendizado Online em alguns trabalhos e fontes de pesquisa. No comparativo dos algoritmos, os ajustes dos algoritmos em Lote geravam valores diferentes para as métricas, gerando um questionamento sobre uma versão ótima de distribuição de nós para maior eficiência do algoritmo de rede neural, por exemplo. O ajuste de janela de tempo que, no caso desse trabalho a janela de $10t$ foi a que obteve melhor performance, pode ser mais eficiente.

Referências Bibliográficas

- [Armstrong, 1985] Armstrong, J. S. (1985). Long-range forecasting: From crystal ball to computer. *Journal of the Operational Research Society*.
- [Bernardes, 2010] Bernardes, J. A. B. (2010). Algoritmo de aprendizado de máquina e representação de incerteza em sistemas baseados em conhecimento sob a ótica de funções de pertinência aproximada. Monografia (Bacharel em Ciência da Computação), UFLA (Universidade Federal de Lavras), Minas Gerais, Brazil.
- [Biau, 2012] Biau, G. (2012). Analysis of a random forests model. *Journal of Machine Learning Research* 13 (2012).
- [Bifet et al., 2010] Bifet, A., Holmes, G., Kirkby, R., and Pfahringer, B. (2010). MOA: massive online analysis. *J. Mach. Learn. Res.*, 11:1601–1604.
- [Bifet et al., 2011] Bifet, A., Holmes, G., Kirkby, R., and Pfahringer, B. (2011). Data stream mining: A practical approach. <https://sourceforge.net/projects/moa-datastream/files/documentation/StreamMining.pdf>. Acessado em 03/04/2017.
- [Brodley and Friedl, 1999] Brodley, C. E. and Friedl, M. A. (1999). Identifying mislabeled training data. *Journal of artificial intelligence research*, 11:131–167.
- [Center for Evidence-Based Management, 2016] Center for Evidence-Based Management, C. (2016). What is a picoc? <https://www.cebma.org/faq/what-is-a-picoc/>. Acessado em 21/03/2017.
- [Chessman and Stutz, 1996] Chessman, P. and Stutz, J. (1996). Bayesian classification (autoclass): Theory and results. *Advances in Knowledge Discovery and Data Mining*.
- [Cury and Marques, 2017] Cury, M. J. F. and Marques, J. A. L. F. (2017). A cidade inteligente: uma reterritorialização. *Revista do Desenvolvimento Regional (REDES)*.
- [Domingos and Hulten, 2000] Domingos, P. and Hulten, G. (2000). Mining high-speed data streams. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 71–80. ACM.
- [Faceli et al., 2011] Faceli, K., Lorena, A. C., Gama, J., and Carvalho, A. (2011). Innovation and intellectual property rights. In Fagerberg, J., Mowery, D. C., and Nelson, R. R., editors, *Inteligência Artificial: Uma abordagem de Aprendizado de Máquina*, pages 260–269. LTC, Rio de Janeiro.

- [Filho and Júnior, 2009] Filho, D. B. F. and Júnior, J. A. d. S. (2009). Desvendando os mistérios do coeficiente de correlação de pearson. *Revista Política Hoje*.
- [Fontenla-Romero et al., 2013] Fontenla-Romero, Ó., Guijarro-Berdiñas, B., Martínez-Rego, D., Pérez-Sánchez, B., and Peteiro-Barral, D. (2013). Online machine learning. *Efficiency and Scalability Methods for Computational Intellect*, 27.
- [Gama, 2010] Gama, J. (2010). *Knowledge discovery from data streams*. CRC Press.
- [Gama et al., 2014] Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., and Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, 46(4):44.
- [Geman et al., 2008] Geman, S., Bienenstock, E., and Doursat, R. (2008). Neural networks and the bias/variance dilemma. *Neural Networks*, 4(1).
- [Ghahramani et al., 2015] Ghahramani, A., Tang, C., and Becerik-Gerber, B. (2015). An online learning approach for quantifying personalized thermal comfort via adaptive stochastic modeling. *Building and Environment*, 92:86 – 96.
- [Guimarães et al., 2013] Guimarães, P. H. V., Murillo, A., Andreoni, M., Mattos, D. M. F., Ferraz, L. H. G., Pinto, F. A. V., Costa, L. H. M. K., and Duarte, O. C. M. B. (2013). Comunicação em redes elétricas inteligentes: Eficiência, confiabilidade, segurança e escalabilidade.
- [Hall et al., 2009] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA data mining software: an update. *SIGKDD Explorations*, 11(1):10–18.
- [Kohavi, 1995] Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *International Joint Conference on Artificial Intelligence*.
- [Mohri et al., 2012] Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2012). *Foundations of machine learning*. MIT press.
- [Mukaka, 2012] Mukaka, M. (2012). A guide to appropriate use of correlation coefficient in medical research. *Malawi Medical Journal*.
- [Muthukrishnan, 2005] Muthukrishnan, S. (2005). Data streams: Algorithms and applications. *Foundations and Trends® in Theoretical Computer Science*, 1(2):117–236.
- [Parsifal, 2015] Parsifal, L. (2015). <https://parsif.al/help>. Acessado em 17/04/2019.
- [Pérez-Lombard et al., 2008] Pérez-Lombard, L., Ortiz, J., and Pout, C. (2008). A review on buildings energy consumption information. *Energy and buildings*, 40(3):394–398.
- [Pozo, 2016] Pozo, A. T. R. (2016). Uma teoria e metodologia de aprendizado indutivo. <http://www.inf.ufpr.br/aurora/tutoriais/aprendizadomaq/>. Acessado em 16/08/2017.
- [Rezende, 2001] Rezende, S. O. (2001). *Redes Neurais e Máquinas de Aprendizado*. Editora Bookman Companhia.

- [Rezende, 2003] Rezende, S. O. (2003). *Sistemas inteligentes: fundamentos e aplicações*. Editora Manole Ltda.
- [Riedmiller, 2010] Riedmiller, M. (2010?). Machine learning: Multi layer perceptrons. *AG Maschinelles Lernen*.
- [Russell and Norvig, 2004] Russell, S. and Norvig, P. (2004). *Inteligência Artificial*. Editora Campus.
- [Shaw and Gentry, 1990] Shaw, M. J. and Gentry, J. A. (1990). Inductive learning for risk classification. *IEEE Expert*, 5(1):47–53.
- [Siegel, 2013] Siegel, E. (2013). Predictive analysis: The power to predict who will click, buy, lie, or die. Hoboken.
- [Smith and Martinez, 2011] Smith, M. R. and Martinez, T. (2011). Improving classification accuracy by identifying and removing instances that should be misclassified. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pages 2690–2697. IEEE.
- [Srivastava, 2015] Srivastava, T. (2015). Analytics vidhya. <https://www.analyticsvidhya.com/blog/2015/01/introduction-to-online-machine-learning-simplified-2/>. Acessado em 12/04/2017.
- [Stevenson, 1998] Stevenson, H. (1998). Do lunch or be lunch. boston. *Harvard Business School Press*.
- [Uchôa, 1998] Uchôa, J. Q. (1998). Representação e indução de conhecimento usando teoria de conjuntos aproximados.
- [Upton and Cook, 2008] Upton, G. and Cook, I. (2008). Oxford dictionary of statistics. Oxford.
- [ur Rehman et al., 2017] ur Rehman, M. H., Liew, C. S., Wah, T. Y., and Khan, M. K. (2017). Towards next-generation heterogeneous mobile data stream mining applications: Opportunities, challenges, and future research directions. *Journal of Network and Computer Applications*, 79:1 – 24.
- [Valpola, 2002] Valpola, H. (2002). Bayesian ensemble learning for nonlinear factor analysis. *Acta Polytechnica Scandinavica Mathematics and Computing Series*.
- [Vrablecová et al., 2017] Vrablecová, P., Ezzeddine, A. B., Rozinajová, V., ?árik, S., and Sangaiah, A. K. (2017). Smart grid load forecasting using online support vector regression. *Computers and Electrical Engineering*.
- [Wisesa et al., 2016] Wisesa, H. A., Ma’sum, M. A., and Wibisono, A. (2016). Adaptive range in fimt-dd tree for large data streams. In *2016 International Workshop on Big Data and Information Security (IW BIS)*, pages 109–114.
- [Won et al., 2014] Won, J.-Y., Chen, X., Gratz, P., Hu, J., and Soteriou, V. (2014). Up by their boots-traps: Online learning in artificial neural networks for cmp uncore power management. pages 308–319. cited By 13.

[Zamora-Martínez et al., 2014] Zamora-Martínez, F., Romeu, P., Botella-Rocamora, P., and Pardo, J. (2014). On-line learning of indoor temperature forecasting models towards energy efficiency. *Energy and Buildings*, 83:162–172. cited By 14.