FERNANDO ELIAS DE MELO BORGES

SELEÇÃO INTELIGENTE DE DADOS PARA TREINAMENTO DE MÁQUINA UTILIZANDO CURVAS PRINCIPAIS

LAVRAS – MG

2020

FERNANDO ELIAS DE MELO BORGES

SELEÇÃO INTELIGENTE DE DADOS PARA TREINAMENTO DE MÁQUINA UTILIZANDO CURVAS PRINCIPAIS

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do Curso de Graduação em Engenharia de Controle e Automação, para a obtenção do título de Engenheiro.

Prof. DSc. Danton Diego Ferreira
Orientador

LAVRAS – MG
2020

Ficha catalográfica elaborada pela Coordenadoria de Processos Técnicos

da Biblioteca Universitária da UFLA

Borges, Fernando Elias de Melo

Seleção Inteligente de Dados para Treinamento de Máquina Utilizando Curvas Principais / Fernando Elias de Melo Borges. 2^a ed. rev., atual. e ampl. – Lavras : UFLA, 2020.

59 p. : il.

Monografia–Universidade Federal de Lavras, 2020.

Orientador: Prof. DSc. Danton Diego Ferreira.

Bibliografia.

1. TCC. 2. Monografia. 3. Dissertação. 4. Tese. 5. Trabalho Científico – Normas. I. Universidade Federal de Lavras. II. Título.

CDD-808.066

FERNANDO ELIAS DE MELO BORGES

SELEÇÃO INTELIGENTE DE DADOS PARA TREINAMENTO DE MÁQUINA UTILIZANDO CURVAS PRINCIPAIS

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do Curso de Graduação em Engenharia de Controle e Automação, para a obtenção do título de Engenheiro.

APROVADA em 24 de Agosto de 2020.

Prof. DSc. Danton Diego Ferreira UFLA
Prof. DSc. Wilian Soares Lacerda UFLA

Prof. DSc. Danton Diego Ferreira
Orientador

LAVRAS – MG
2020

Em memória de minha avó, Maria Luiza Borges, pessoa de grande impacto na minha vida e bondade inigualável. Sei que onde estiver, estará olhando pela gente.

AGRADECIMENTOS

Neste momento, o que não faltam são motivos para agradecer, uma jornada que, creio eu, me fez crescer de uma forma incrível, desde o meu começo como calouro, até hoje, em minha última etapa como graduando. Primeiramente, agradecer aos meus pais que sempre me motivaram aos estudos, à dedicação e à busca por conhecimento, me dando apoio, carinho e me incentivando a crescer, não só como profissional, mas como ser humano. À minha irmã Letycia, que esteve comigo na UFLA na reta final do meu curso, por estar ao meu lado e me aturar neste tempo. Aos meus tios do Rio, Elvis, Jacinta e Welinton, por todo o apoio e incentivo, durante minha estadia aí, tanto no ensino médio, quanto durante meu estágio. Às minhas avós, Aquilea e Maria pelo carinho durante minha criaçao. Aos meus amigos de alojamento na UFV, durante meu tempo no ap. 2331, sei que foi pouco tempo, mas minha consideração por vocês é enorme, devo uma visita à vocês. Meus companheiros de 307, bloco 2 no alojamento da UFLA e, por fim, mas não menos importante, aos meus amigos e companheiros da República Vaca-H, que, neste trabalho, tiveram que me ouvir bastante. Aos amigos que a UFLA me deu, Andrey, Artur, Igor, Luan, Diogo e outros mais. Aos professores com quem trabalhei ao longo da graduação durante projetos de iniciação científica, monitoria e outros trabalhos, em especial ao professor Danton pelo incentivo, apoio e credibilidade que, graças ao teu incentivo, cheguei no ponto em que estou. Aos técnicos da UFLA, Olavo, Bruno, Alexandre e Fabiano por me ajudarem durante os experimentos na IC. Ao pessoal do LPS da UFRJ por me receber de braços abertos para a realização de meu estágio e dos estudos que chegaram à este trabalho, em especial ao professor Seixas por me conceder incrível oportunidade. Aos amigos que fiz na UFRJ durante minha mobilidade. Aos meus velhos amigos e companheiros do Rio, Pedrinho, Sebastian, Diogo, Larissa, Marlonn, Victor Luiz, Matheus, Zé e toda a turma. Aos demais que não citei meu muito obrigado e minhas singelas desculpas, pois estou escrevendo os agradecimentos de madrugada, mas tenham meu muito obrigado com todo carinho e sinceridade.

"Os sonhos das pessoas... não tem fim."
(Marshall D. Teach (Barba Negra), One Piece (Anime Episódio 147, Mangá Capítulo 225).)

RESUMO

Nos tempos atuais, sistemas inteligentes aplicados a ambientes envolvendo grande volume de dados em altas taxas de aquisição, vêm tendo sua importância e uso aumentados. Tais sistemas geram eventos com elevada dimensionalidade e complexidade e necessitam de processamento eficaz com elevados requisitos de tempo de processamento e consumo de memória. A fim de processar grandes volumes de dados, ferramentas de aprendizagem de máquina de alta complexidade vem sendo aplicadas nos ambientes de *Big-Data*. De maneira a reduzir a carga dos algoritmos de aprendizagem, mantendo os parâmetros de desempenho com redução no tempo de desenvolvimento do modelo, torna-se viável a proposta de métodos de redução no volume dados a serem utilizados no treinamento dos modelos. Neste trabalho é proposto um método de seleção inteligente de dados utilizando Curvas Principais que explora correlações não lineares nos dados por meio destas. Para a execução desta tarefa, é realizado o mapeamento das distâncias dos dados à sua respectiva Curva Principal e são propostas abordagens de seleção. Para o teste do método, foi utilizada uma base de dados real do sistema de filtragem *online* de elétrons do experimento ATLAS do CERN (Centro Europeu para a Pesquisa Nuclear). Realizada a seleção de dados, os conjuntos de dados reduzidos foram testados em uma Rede Neural objetivando simular o processo real do sistema. Resultados de testes preliminares mostraram um potencial do método, gerando resultados similares de desempenho com redução significativa de tempo de processamento em estudos comparativos entre todo o conjunto de dados e os conjuntos de dados reduzidos.

Palavras-chave: Curvas Principais. Seleção de Dados. *Big-Data*. Aprendizagem de Máquina. Redes Neurais.

ABSTRACT

Nowadays, smart systems applied to environments involving a large volume of data at high acquisition rates have increased in importance and use. Such systems generate events with high dimensionality and complexity and require efficient processing with high requirements for processing time and memory consumption. To processing big-data, machine learning tools with high complexity have been applied. In order to reduce the processing cost of the learning algorithms, maintaining the performance parameters with reduction in the development time of the model, it is feasible to propose methods to reduce the volume of data to be used for training. In this work, a method of smart data selection using Principal Curves is proposed, which exploits non-linear correlations in the data through them. To do this, the mapping of the data distances to their respective Principal Curve is realized and selection approaches are proposed. For the test of the method, a real dataset from the online electron trigger system of the ATLAS experiment at CERN (European Center for Nuclear Research) was used. After data selection, the reduced datasets were tested in a Neural Network in order to simulate the real process of the system. The results showed the potential of the method, generating similar performance results with significant reduction of processing time in comparison with studies including the complete data set.

Keywords: Principal Curves. Data Selection. Big-Data. Machine Learning. Neural Networks.

LISTA DE FIGURAS

Figura 2.1 – Conjunto de dados com representações por: (a) Regressão Linear, (b) PCA, (c) Regressão Não-Linear e (d) Curvas Principais. 29

Figura 2.2 – Representação em Diagrama de Blocos do Algoritmo k-segmentos para obtenção de Curvas Principais. 30

Figura 2.3 – Exemplo Gráfico de Uma CP obtida por meio do k-segmentos. 31

Figura 2.4 – Representação do Grafo do Neurônio artificial proposto por (MCCULLOCH; PITTS, 1943). 31

Figura 2.5 – Grafo de uma Rede Neural Muticamadas do tipo *feedforward* totalmente conectada. 33

Figura 2.6 – Grafo ilustrando o funcionamento do *backpropagation* e seus fluxos de sinais 33

Figura 2.7 – Desenho Esquemático do LHC contendo a representação da localização dos experimentos. 35

Figura 2.8 – Ilustração do detector ATLAS e seus sistemas de detecção de partículas. 36

Figura 2.9 – Desenho ilustrativo do Sistema de Filtragem *Online* do ATLAS. 37

Figura 2.10 – Desenho ilustrativo do Algoritmo Anelador para extração da informação do Sistema de Calorimetria no ATLAS. 38

Figura 3.1 – Exemplo de evento de sinal. 39

Figura 3.2 – Exemplo de evento de ruído. 40

Figura 3.3 – Exemplo do funcionamento da seleção de dados via CP. . . . 41

Figura 3.4 – Fluxograma do Método Aplicado. 42

Figura 4.1 – Gráfico de dispersão das distâncias para a classe de sinal. . . . 43

Figura 4.2 – Gráfico de dispersão das distâncias para a classe de ruído. . . 44

Figura 4.3 – Histograma de distâncias para o sinal nas baixas distâncias (distâncias menores que 0.03[u.a.]). 45

Figura 4.4 – Histograma de distâncias para o sinal nas altas distâncias (distâncias maiores que 0.03[u.a.]). 45
Figura 4.5 – Histograma de distâncias para o ruído nas baixas distâncias (distâncias menores que 0.03[u.a.]). 46
Figura 4.6 – Histograma de distâncias para o ruído nas altas distâncias (distâncias maiores que 0.03[u.a.]). 46

LISTA DE TABELAS

Tabela 2.1 – Distribuição dos Anéis em cada camada do sistema de calorimetria do ATLAS pelo algoritmo anelador 38

Tabela 3.1 – Tamanho do conjunto de dados utilizado no experimento . . . 39

Tabela 4.1 – Resultados percentuais da Rede Neural (média $\pm$ desvio-padrão) utilizando a Abordagem 1 de seleção. Sendo: N_t o tamanho do conjunto de treinamento das Redes Neurais reduzido pelas CP; SP_{max} o máximo índice SP obtido no treinamento com validação cruzada; P_D a probabilidade de detecção e P_F o falso alarme. 47

Tabela 4.2 – Resultados percentuais da Rede Neural (média $\pm$ desvio-padrão) utilizando a Abordagem 2 de seleção. Sendo: N_t o tamanho do conjunto de treinamento das Redes Neurais reduzido pelas CP; SP_{max} o máximo índice SP obtido no treinamento com validação cruzada; P_D a probabilidade de detecção e P_F o falso alarme. 48

Tabela 4.3 – Resultados percentuais da Rede Neural (média $\pm$ desvio-padrão) utilizando a Abordagem 3 de seleção. Sendo: N_t o tamanho do conjunto de treinamento das Redes Neurais reduzido pelas CP; pp a proporção dos dados mais próximos no conjunto reduzido; SP_{max} o máximo índice SP obtido no treinamento com validação cruzada; P_D a probabilidade de detecção e P_F o falso alarme. 48

Tabela 4.4 – Resultados percentuais da Rede Neural (média $\pm$ desvio-padrão) utilizando todo o conjunto de desenvolvimento da CP. Sendo: N_t o tamanho do conjunto de treinamento das Redes Neurais sem a redução pelas CP; SP_{max} o máximo índice SP obtido no treinamento com validação cruzada; P_D a probabilidade de detecção e P_F o falso alarme. 49

LISTA DE QUADROS

SUMÁRIO

1 INTRODUÇÃO 23

1.1 Objetivos 25

1.2 Estrutura do Trabalho 26

2 REVISÃO BIBLIOGRÁFICA 27

2.1 Problemas envolvendo *Big-Data* 27

2.2 Curvas Principais 28

2.3 Algoritmo k-segmentos para obtenção de Curvas Principais 29

2.4 Redes Neurais Artificiais 30

2.5 O Sistema de Filtragem do ATLAS e o Algoritmo *NeuralRinger* . . 34

3 MÉTODO PROPOSTO 39

3.1 Implementação 42

4 RESULTADOS E DISCUSSÃO 43

4.1 Resultados do Processo de Seleção de Dados 43

4.2 Discussão 49

5 CONCLUSÃO 53

REFERÊNCIAS 55

APENDICE A – Algoritmo para Seleção de Dados 59

1 INTRODUÇÃO

Com o avanço da era da informação, os sistemas computacionais atuais vêm trabalhando cada vez mais com os dados que os mesmos geram, por meio do uso de ferramentas de aprendizagem de máquina. Tais sistemas vêm realizando aquisições de grandes volumes de dados em curtos espaços de tempo (alta taxa de eventos) com dados de grande dimensionalidade. Tal problemática, se caracteriza em um problema de *big-data*, onde grandes volumes de dados de alta complexidade gerados em curto intervalo de tempo necessitam de processamento rápido (SAGIROGLU; SINANC, 2013).

Na literatura há varios exemplos de estudos de caso envolvendo *big-data* nas mais variadas áreas, como em (NUAIMI et al., 2015), onde os autores fizeram um estudo acerca do uso de *big-data* no desenvolvimento de cidades inteligentes. (ONAL et al., 2017) fizeram uso de *big-data* no monitoramento de codições climáticas. Aplicações também no setor de saúde (KUMAR; SINGH, 2018) e em mobilidade urbana (JIANG; FERREIRA; GONZALEZ, 2017) também mostram o quão variado pode ser o uso de *big-data* em aplicações reais.

Entretanto, para que tais aplicações sejam implementadas, é necessário tratar a grande massa de dados altamente complexa de maneira que sejam atendidos os requisitos de desempenho computacional. Em meio a este problema, trabalhos como os reportados em (JIN et al., 2015) e (FAN; HAN; LIU, 2014) mostram os desafios a serem enfrentados em aplicações envolvendo *big-data* e também no uso de ferramentas de aprendizagem de máquina em tal cenário. Dentre estes desafios, são relatados os desafios de custo computacional, onde são necessários métodos que demandem a menor carga computacional possível aliado aos parâmetros de desempenho estipulados no projeto.

Imerso na problemática supracitada, o experimento ATLAS inserido no CERN (Centro Europeu para a Pesquisa Nuclear) tem, em seus trabalhos na área de física experimental de altas energias, problemas envolvendo *big-data*. Tais pro-

blemas encontram-se durante a aquisição dos eventos necessários para a observação dos eventos físicos de interesse ocorridos no interior do colisor de partículas LHC (*Large Hadron Collider*). Dentro do colisor são realizadas colisões por meio de cruzamento de prótons acelerados em sentidos opostos, estas colisões geram partículas instáveis, as quais podem ser analisadas por meio do seu decaimento em partículas mais estáveis, como, por exemplo, elétrons. No ATLAS, sinais oriundos de elétrons são de grande interesse de estudo para fins de reconstrução dos eventos físicos de interesse ocorridos no interior do colisor. Dentre tais eventos, inclui-se a observação do Bóson de Higgs ocorrida em 2012 (ATLAS COLLABORATION, 2012), (CMS COLLABORATION, 2012).

Para que sejam observados eventos raros, como o Bóson de Higgs ou outros de natureza ainda mais rara, os sistemas de aquisição de dados do ATLAS operam em uma taxa de eventos de, aproximadamente, 70TB/s para a geração da estatística necessária para a reconstrução dos eventos de interesse. Entretanto, a grande maioria desses dados é composta por ruído de fundo do experimento, sem interesse de análise.

Com o objetivo de evitar que uma grande massa de dados sem interesse de estudo sejam armazenados, o ATLAS implementou um sistema de filtragem *online* de eventos (*Trigger*). Tal sistema possui implementações em *software* e *hardware* e possui altos requisitos para sua operação. Como, por exemplo, requisitos de desempenho, em que o algoritmo deve filtrar somente os sinais de interesse em meio à grande massa de ruído e de latência, que se refere ao intervalo de tempo no qual o sistema de filtragem deve operar.

Dentre os algortimos implementados em *software*, inclui-se o *NeuralRinger*, desenvolvido pela COPPE/UFRJ, que é a atual referência na filtragem *online* na cadeia de elétrons do ATLAS. O *NeuralRinger* extrai a informação do sistema de calorimetria e a compacta, formando anéis concêntricos de energia e realiza a filtragem por meio de um *ensemble* de Redes Neurais (FREUND, 2018). O algo-

ritmo possui como características uma elevada taxa de detecção e baixo índice de falso alarme, além de outros fatores, como um baixo *bias* no sistema. Com isto, o modelo é capaz de extrair os sinais de elétrons e evitar que o ruído de fundo seja coletado erroneamente como sinal.

Entretanto, dada a problemática envolvendo *big-data* imersa no ATLAS, a realização dos ciclos de aprendizagem do modelo neural acaba por demandar elevada carga computacional para que o mesmo possa convergir e atingir os resultados esperados. Este trabalho visa contornar este problema, por meio da implementação de um sistema de seleção inteligente de dados. Tal proposta, tem por objetivo reduzir o número de eventos a serem treinados pelas Redes Neurais visando manter padrões de desempenho similares, podendo, assim, reduzir o tempo de processamento dos dados. Alguns exemplos de técnicas de redução de dados envolvendo *big-data* podem ser vistos em (GENENDER-FELTHEIMER, 2018).

Neste trabalho é prposto um método baseado em Curvas Principais (CP) (HASTIE; STUETZLE, 1989) via algoritmo k-segmentos (VERBEEK; VLASSIS; KRÖSE, 2002) para a realização da seleção dos eventos a serem treinados em uma Rede Neural Artificial (HAYKIN, 2007). A seleção se deu pelo mapeamento das distâncias de cada evento à sua respectiva Curva Principal e por meio deste foram propostas abordagens de seleção de dados. Neste trabalho, o enfoque foi dado à redução somente do número de eventos, logo, a dimensionalidade (número de variáveis) do problema foi mantida.

1.1 Objetivos

Este trabalho tem por objetivo geral propor um sistema de seleção inteligente de dados utilizando Curvas Principais. Como objetivos específicos podem ser listados: Obter o mapeamento dos eventos em relação à Curva Principal e, por meio deste, propor abordagens de seleção de dados; analisar os resultados das Redes Neurais, dando enfoque aos resultados de teste, verficando se os con-

juntos reduzidos de dados geram resultados similares de desempenho com todo o conjunto de dados; e verificar quais das abordagens de seleção possui o melhor desempenho, tanto em acertos da Rede Neural, quanto na redução do tempo de processamento.

1.2 Estrutura do Trabalho

O texto segue dividido em quatro capítulos posteriores: no capítulo 2 estão apresentadas as revisões a partir do problema relacionado além das ferramentas que foram utilizadas neste trabalho; o capítulo 3 apresenta como foi aplicado o método proposto, seus passos e sua implementação; os resultados e discussões a respeito do experimento estão contidos no capítulo 4 e, por fim, as conclusões, perspectivas e considerações finais do trabalho estão no capítulo 5.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo são apresentados os problemas encontrados em ambientes envolvendo *Big-Data* bem como uma revisão a respeito de Curvas Principais e o algoritmo k-segmentos. É apresentado também o modelo de Redes Neurais Artificiais, modelo de aprendizagem de máquina utilizado neste trabalho e, por fim, é mostrada uma revisão sobre o experimento ATLAS e o algoritmo *NeuralRinger*.

2.1 Problemas envolvendo *Big-Data*

Big-data consiste em uma terminologia recente que trata de problemas que envolvem grandes volumes de dados, contudo, não somente o tamanho se faz essencial. Para ser enquadrado em um problema de *big-data*, outros fatores precisam ser levados em conta, como os chamado "Vs"que englobam tal dinâmica. Em (SAGIROGLU; SINANC, 2013) são explorados 3 Vs: além do volume, foram apresentados a variedade, que se refere à complexidade dos dados no sistema e a velocidade de processamento dos dados, sendo, em alguns casos, processamento de tempo real. Revisões mais recentes como em (MIKALEF et al., 2018) apresentam alguns fatores a mais a serem considerados em *big-data*, como Variabilidade, Veracidade, Valor e Visualição.

No capítulo anterior foram mostrados alguns exemplos de aplicações que envolvem *big-data*. Além dos citados, vários outros podem ser encontrados na literatura nas mais variadas áreas do conhecimento como em empresas de telecomunicações para análise de clientes, recursos humanos (RABHI et al., 2019), microrredes (MOHARM, 2019), sistemas de manufatura inteligente (ZHANG; MING; YIN, 2020). Tais trabalhos mostram a diversidade de áreas em que se tem trabalhado com *big-data*.

Entretanto, vários desafios existem na área para processar tais dados, dentre os quais, uns que se destacam são os desafios envolvendo custo computacional como visto em (JIN et al., 2015) e (FAN; HAN; LIU, 2014). Para contornar tal

problema, algumas soluções são propostas na literatura, como redução de dimensionalidade por meio de seleção de variáveis (HASANIN et al., 2019), (FONG; WONG; VASILAKOS, 2015) e seleção de dados (GENENDER-FELTHEIMER, 2018), sendo este último tipo de solução, a trabalhada pelo método proposto neste trabalho.

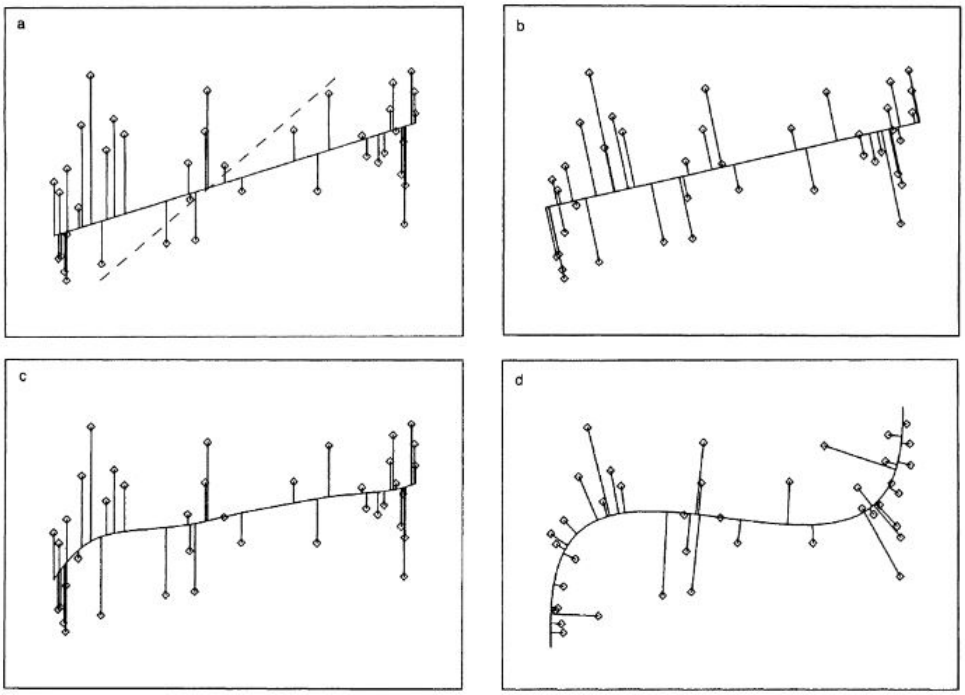
2.2 Curvas Principais

Curvas Principais (CP) consistem em uma generalização não linear da Análise de Componentes Principais (ou PCA - do inglês *Principal Component Analysis*) (HASTIE; STUETZLE, 1989). As CP geram um modelo compacto de dados, transpondo um determinado conjunto de dados de um espaço *n*-dimensional para uma representação unidimensional deste mesmo conjunto. As CP se baseiam no princípio da auto-consistência[1], não se interceptam e sua forma é sugerida pelos dados utilizados em sua projeção. Na Figura 2.1 são mostradas representações de modelos compactos de dados envolvendo Regressão Linear e Não-Linear, PCA e Curvas Principais. Observe que as Curvas Principais acompanham a distribuição dos dados no espaço, minimizando a distância dos dados à mesma.

O modelo de CP inicialmente proposto por (HASTIE; STUETZLE, 1989) foi de grande contribuição para a obtenção de novas representações de dados. Contudo, o algoritmo apresentava determinadas limitações, tais como tendência apresentada pela curva além de sua convergência não ser garantida para qualquer distribuição de dados. Visando contornar tais problemas, algoritmos alternativos foram propostos a fim de adquirir maior robustez no processo de extração das CP, como em (BANFIELD; RAFTERY, 1992), (TIBSHIRANI, 1992), (DELICADO, 2001) e (VERBEEK; VLASSIS; KRÖSE, 2002), sendo este útlimo citado, o algoritmo utilizado neste trabalho.

[1] O conceito de auto-consistência significa, no contexto de Curvas Principais, que um dado ponto da CP corresponde à média dos dados que o projetam (HASTIE; STUETZLE, 1989).

Figura 2.1 – Conjunto de dados com representações por: (a) Regressão Linear, (b) PCA, (c) Regressão Não-Linear e (d) Curvas Principais.



Fonte: (HASTIE; STUETZLE, 1989).

2.3 Algoritmo k-segmentos para obtenção de Curvas Principais

O algoritmo k-segmentos proposto por (VERBEEK; VLASSIS; KRÖSE, 2002) é um modelo alternativo de extração de Curvas Principais inicialmente proposto por (HASTIE; STUETZLE, 1989). Neste, a obtenção de uma determinada CP se faz de maneira incremental por meio de segmentos de reta que são ligados entre si. Este algoritmo se diferencia dos demais pela sua elevada robustez, possuindo menos tendência à mínimos locais e convergência prática garantida. O algoritmo consiste nos 3 seguintes passos:

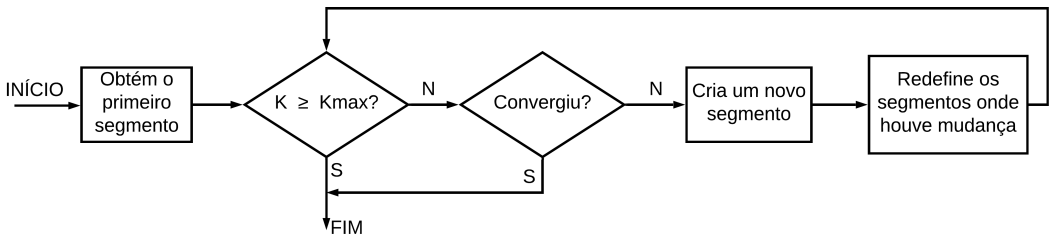
1. Inicia-se com a inserção do primeiro segmento, este possui a direção da primeira componente principal e comprimento equivalente à $3/2$ do desvio padrão dos dados.

2. Insere-se o segundo segmento, nesta nova inserção, são redefinidos os pontos centrais do agrupamento. Os eventos pertencentes à cada agrupamento são definidos por meio do algoritmo *k*-means baseando-se nas regiões de Voronoi[2]. Também são recalculados os tamanhos dos segmentos onde houve mudança no agrupamento.

3. Este último passo consiste na análise da convergência do algortimo, esta é realizada de duas formas: verificando se o número de segmentos atingiu o máximo estipulado pelo usuário ou se o maior agrupamento possui menos de 3 segmentos. Não atendendo tais condições, retorna-se ao passo 1.

Uma ilustração do funcionamento do algoritmo é mostrada na Figura 2.2 e um exemplo da obtenção de CP por meio do k-segmentos em um dado conjunto de dados é ilustrada na Figura 2.3.

Figura 2.2 – Representação em Diagrama de Blocos do Algoritmo k-segmentos para obtenção de Curvas Principais.



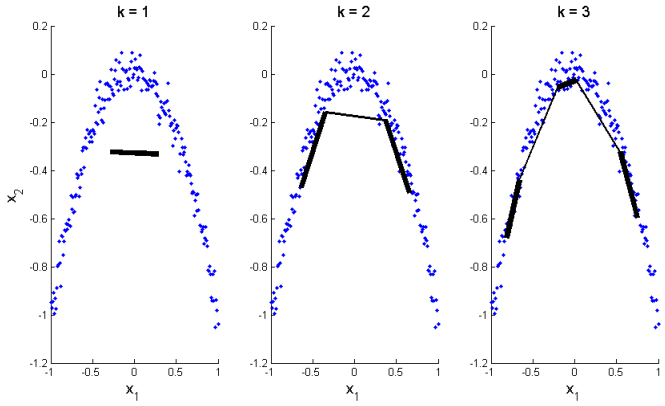
Fonte: do Autor.

2.4 Redes Neurais Artificiais

As Redes Neurais Artificiais, mais referidas como Redes Neurais, são modelos matemáticos baseados no princípio do funcionamento dos neurônios biológicos (HAYKIN, 2007). Seu primeiro modelo foi proposto em (MCCULLOCH;

[2] As regiões de Voronoi, no contexto do k-segmentos, são agrupamentos formados onde os eventos de um dado agrupamento (região) estão mais próximos do seu centro que de um dos segmentos da Curva Principal (VERBEEK; VLASSIS; KRÖSE, 2002).

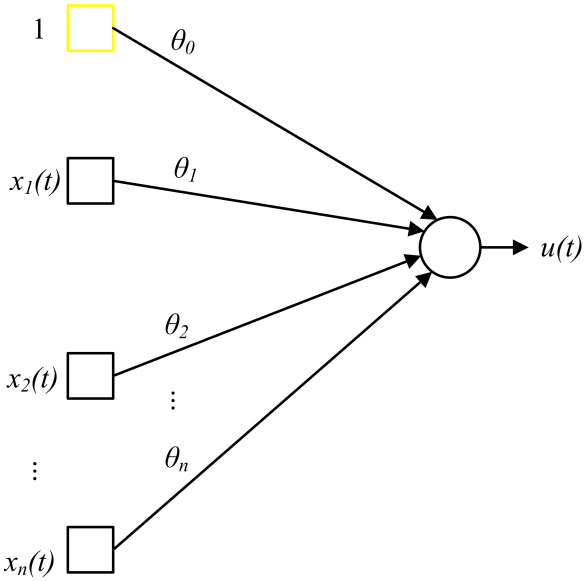
Figura 2.3 – Exemplo Gráfico de Uma CP obtida por meio do k-segmentos.



Fonte: do Autor.

PITTS, 1943) onde os autores apresentaram um modelo gráfico e matemático por meio de funções lógicas representando cada passo do processamento de um neurônio natural. Na Figura 2.4 é apresentado um grafo do modelo de um neurônio artificial.

Figura 2.4 – Representação do Grafo do Neurônio artificial proposto por (MCCULLOCH; PITTS, 1943).



Fonte: do Autor.

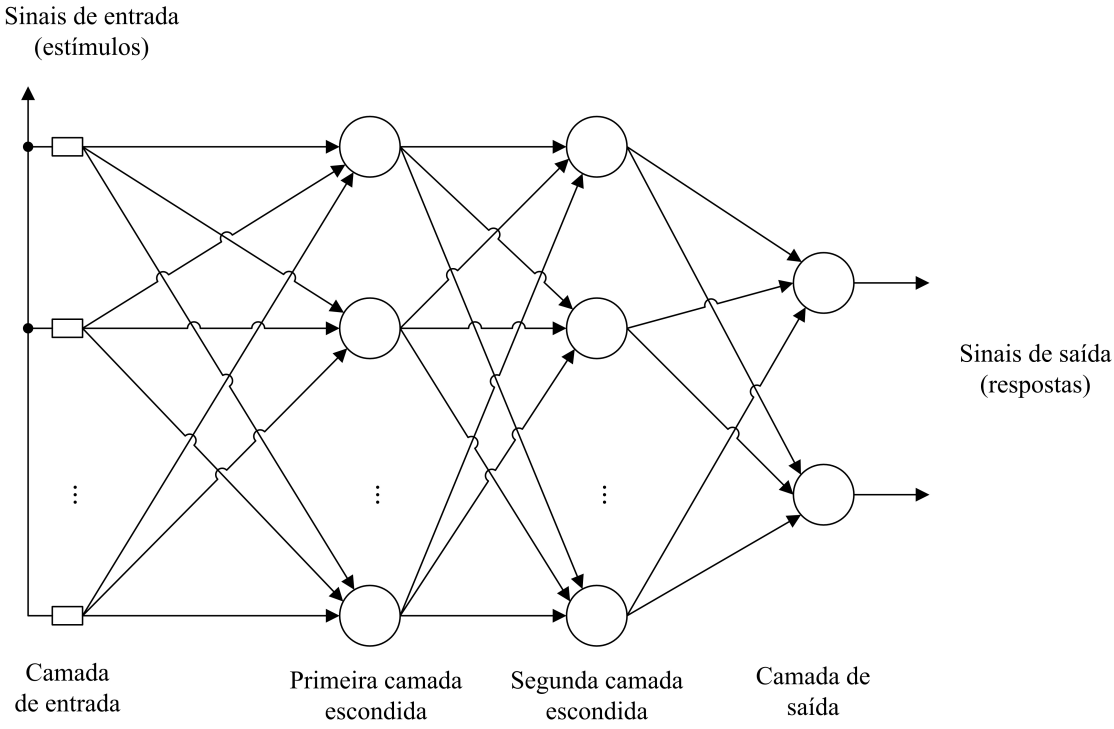
A partir do neurônio artificial, outros dois postulados importantes foram cruciais para o desenvolvimento das Redes Neurais atuais, como o aprendizado Hebbiano (HEBB, 1949) e o *perceptron* como modelo de aprendizagem (ROSENBLATT, 1958). Tais ideias iniciaram o modelo de Redes Neurais como conhecidas atualmente.

A partir do proposto pelos autores supracitados, foram propostos aperfeiçoamentos ao modelo de aprendizagem neural, sendo um dos modelos desenvolvidos, uma generalização do *perceptron* de camada única, o *perceptron* de múltiplas camadas (ou MLP, do inglês, *Multi-Layer Perceptron*) (HAYKIN, 2007). Uma das formas de Rede Neural MLP existentes, e utilizada neste trabalho, é a Rede Neural do tipo *feedforward* totalmente conectada. O termo *feedforward* é dado ao tipo de Rede Neural em que os estímulos surgidos na entrada seguem unidirecionalmente para a camada de saída. Enquanto o termo totalmente conectada se refere à ligação de todos os neurônios de uma determinada camada com todos os neurônios da camada anterior e a todos os neurônios da camada seguinte. Na Figura 2.5 é mostrado um grafo de uma rede MLP do tipo *feedforward* totalmente conectada.

O modelo MLP teve seu treinamento viabilizado por meio do algoritmo de retropropagação do erro (ou *error backpropagation*, do inglês, ou, somente, *backpropagation*) (HAYKIN, 2007). Neste processo, os pesos sinápticos são ajustados no momento em que o fluxo de sinal de erro ocorre contrariamente ao sinal funcional do modelo. Enquanto o sinal funcional percorre dos vetores de entrada sentido à camada de saída, o sinal de erro percorre da camada de saída rumo à camada de entrada se propagando neste sentido. Desta forma, é realizado o ajuste dos pesos sinápticos da rede em conjunto com a minimização do erro do modelo. Uma ilustração do fluxo do sinal do erro em relação ao sinal do estímulo da Rede Neural, exemplificando o *backpropagation* é mostrado na Figura 2.6.

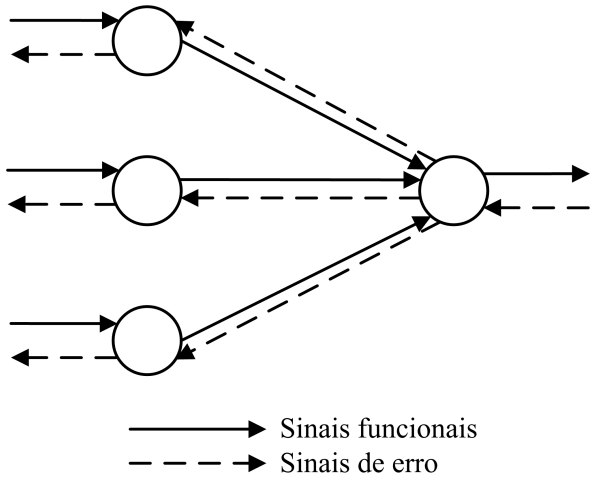
Baseando-se no grafo da Figura 2.5, o funcionamento do modelo de uma rede MLP consiste na soma dos sinais de entrada multiplicados aos respectivos

Figura 2.5 – Grafo de uma Rede Neural Muticamadas do tipo *feedforward* totalmente conectada.



Fonte: do Autor.

Figura 2.6 – Grafo ilustrando o funcionamento do *backpropagation* e seus fluxos de sinais



Fonte: do Autor.

pesos sinápticos tendo a adição de um valor de *bias*. Esta soma é, então, passada por uma função de ativação. A saída da função de ativação consiste na saída do neurônio. Tal processo é feito, camada por camada, até a camada de saída em que é gerada a saída do modelo, podendo ser a classe no qual um determinado evento pertence ou a probabilidade deste evento pertencer a uma determinada classe (em problemas de classificação) ou o valor de uma determinada função a qual a Rede Neural foi utilizada para aproximação (para problemas de regressão). De uma maneira generalizada, pode-se escrever a saída de todos os neurônios de uma determinada camada k da Rede Neural escrevendo a saída em função da entrada de forma matricial. Esta função é definida pela equação (2.1):

$$\mathbf{u}_k = f(\mathbf{W}_k\mathbf{x} + \mathbf{b}_k) \tag{2.1}$$

em que u é a saída de uma camada k, $\mathbf{W}_k$ é a matriz de pesos da camada k; $\mathbf{b}_k$ são os *bias* da camada e $\mathbf{x}$ corresponde à entrada da camada k, no caso da primeira camada, $\mathbf{x}$ serão os dados de entrada da rede, nos demais casos $\mathbf{x}$ corresponde a saída da camada anterior $k-1$. A funçao $f(.)$ refere-se à função de ativação da Rede Neural. Existem vários tipos de funções de ativação para Redes Neurais, podendo estas serem lineares, ou não. Algumas funções de ativação utilizadas em Redes Neurais são: função logística (ou sigmoide), função ReLu, tangente hiperbólica, entre outras. Esta última foi a função de ativação utilizada neste trabalho e sua função é dada pela equação (2.2):

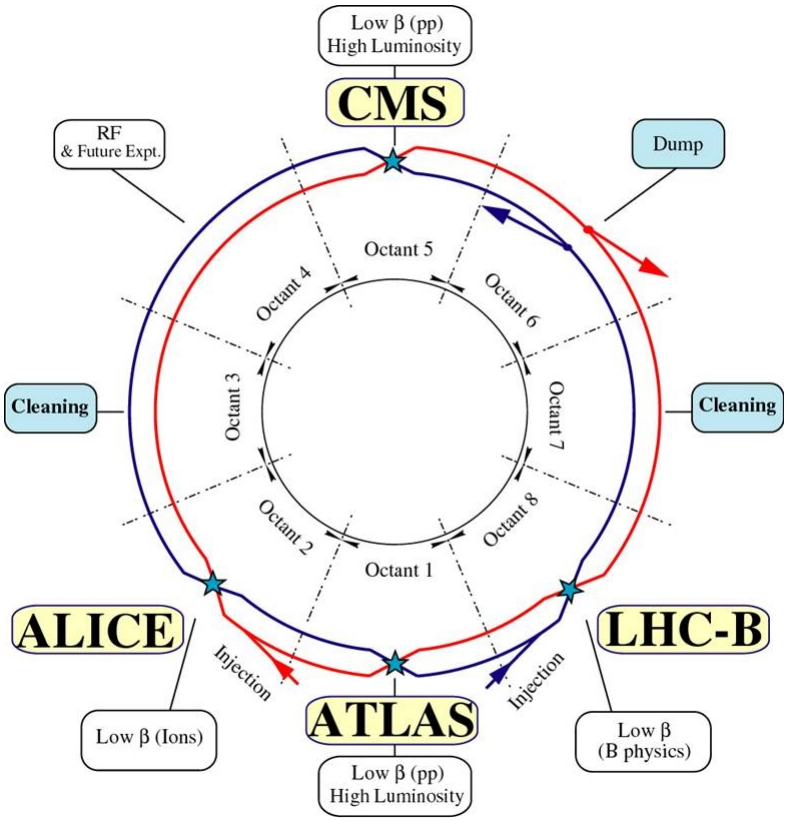
$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{2.2}$$

2.5 O Sistema de Filtragem do ATLAS e o Algoritmo *NeuralRinger*

O colisor de partículas LHC possui 4 Experimentos que consistem em pontos de colisão: LHCb, CMS, ALICE e ATLAS. Dentre estes, o Experimento

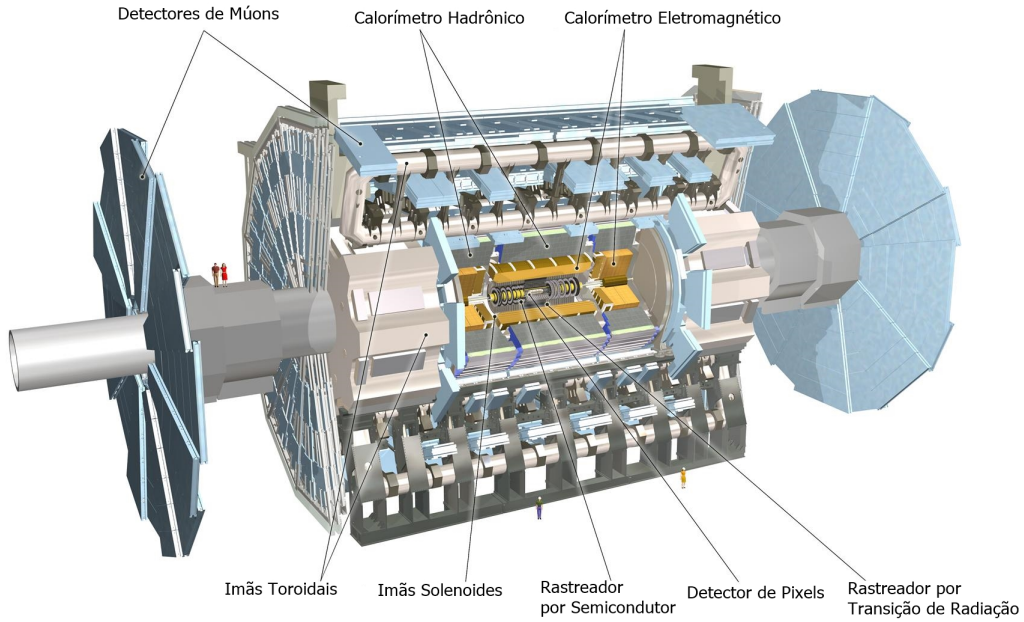
ATLAS (ATLAS COLLABORATION, 2008) consiste no maior destes detectores em que são realizados diversos experimentos e análises no campo de física de altas energias. Dentro do detector, existem três principais sistemas de deteção que são: o Detector de Traço (ID), o Sistema de Calorimetria e o Espectrômetro de Múons (FREUND, 2018), (ATLAS COLLABORATION, 2008). Neste trabalho, os estudos foram concentrados nos sinais de energia oriundos do Sistema de Calorimetria. Um desenho esquemático do LHC pode ser visto na Figura 2.7 enquanto na Figura 2.8 é mostrado um desenho do detector do ATLAS e seus sistemas de detectores.

Figura 2.7 – Desenho Esquemático do LHC contendo a representação da localização dos experimentos.



Fonte: (LHC, 2020).

Figura 2.8 – Ilustração do detector ATLAS e seus sistemas de detecção de partículas.



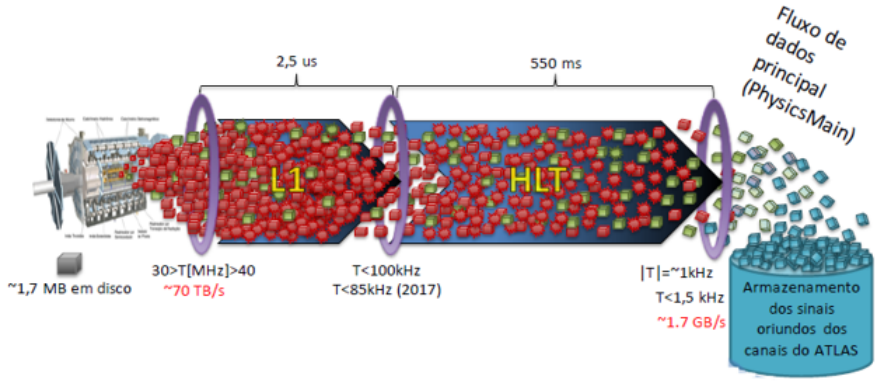
Fonte: Adaptado de (ATLAS COLLABORATION, 2008).

O Sistema de Calorimetria do ATLAS (ATLAS COLLABORATION, 1999) consiste em um dos sistemas de detecção em que são coletados os sinais de energia de partículas que incidem sobre o calorímetro. O sistema é composto, resumidamente, por 3 partes: o Pré-Amostrador (PS), o Calorímeto Eletromagnético e o calorímetro hadrônico. Tal sistema possui, no total, aproximadamene 190.000 canais de leitura em uma fina granularidade e segmentação o que geram dados de elevada complexidade.

Como introduzido no capítulo anterior, a maior parte da massa de dados gerada pelas colisões no ATLAS é composta por ruído e não possui interesse de estudo. Logo, para evitar que este ruído seja armazenado, sistemas de filtragem *online* (*Trigger*) foram implementados no ATLAS a fim de armazenar somente os sinais de interesse imersos na grande massa de ruído. O *Trigger* é composto por etapas de filtragem (níveis), onde cada um desses níveis atua de uma determinada forma, mas com o mesmo objetivo: descartar a massa de ruído e preservar os

sinais de interesse. O sistema de filtragem, possui uma divisão em duas partes: o L1, implementado em *hardware* e o HLT (*High Level Trigger*), implementado em *software*. Este último possui duas etapas: a etapa rápida e a etapa precisa. O algoritmo *NeuralRinger*, explorado neste trabalho, opera na etapa rápida do HLT. Um esboço do sistema de filtragem *online* é observado na Figura 2.9.

Figura 2.9 – Desenho ilustrativo do Sistema de Filtragem *Online* do ATLAS.



Fonte: Adaptado de (FREUND, 2018).

O *NeuralRinger*, algoritmo proposto para filtragem *online* de elétrons no ATLAS (FREUND, 2018) consiste no atual modelo de referência na cadeia de elétrons do experimento. O modelo é composto pelo algoritmo anelador que compacta a informação gerada pelos canais do sistema de calorimetria no formato de anéis concêntricos de energia, de maneira a explorar a geometria do calorímetro do ATLAS. A extração da informação gera como resultado a energia em cada anel, no total são 100 anéis de energia distribuídos em camadas no Pré-Amostrador (PS), no Calorímetro Eletromagnético (EM) e no Calorímetro Hadrônico (HAD). A distribuição dos anéis em cada camada está contida na Tabela 2.1 e uma ilustração do modelo anelador segue na Figura 2.10. O procedimento descrito nesta ilustração consiste na captação da célula mais energética em cada camada e, a partir desta, delimitada uma região de interesse ao longo do plano η x ϕ. Essa região já é pré-

estabelecida e, dentro da mesma, são obtidos os valores de energia das células e somados estes valores. Por fim, o valor da soma realizada é tido como o valor de cada anel. O mesmo procedimento é realizado em todos os 100 anéis.

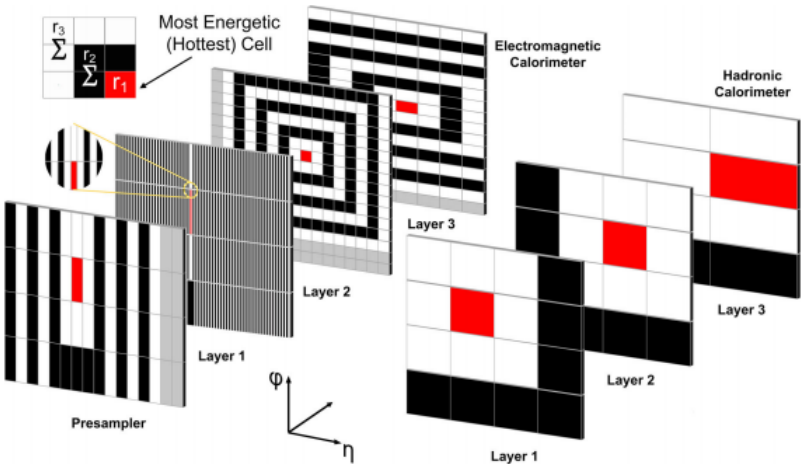
Após a extração da informação anelada de calorimetria, é realizado o processo de filtragem pelas Redes Neurais. Foi proposto um modelo de *ensemble* de maneira ao mesmo se ajustar melhor à geometria do sistema de calorimetria, provendo uma resposta mais suave do sistema. O *ensemble* que compõe o *NeuralRinger* consiste em Redes Neurais totalmente conectadas com uma camada escondida e função de ativação tangente hiperbólica (FREUND, 2018).

Tabela 2.1 – Distribuição dos Anéis em cada camada do sistema de calorimetria do ATLAS pelo algoritmo anelador

Camada	PS	EM1	EM2	EM3	HAD1	HAD2	HAD3
Número de Anéis	8	64	8	8	4	4	4

Fonte: do Autor.

Figura 2.10 – Desenho ilustrativo do Algoritmo Anelador para extração da informação do Sistema de Calorimetria no ATLAS.



Fonte: (ATLAS COLLABORATION, 2020).

3 MÉTODO PROPOSTO

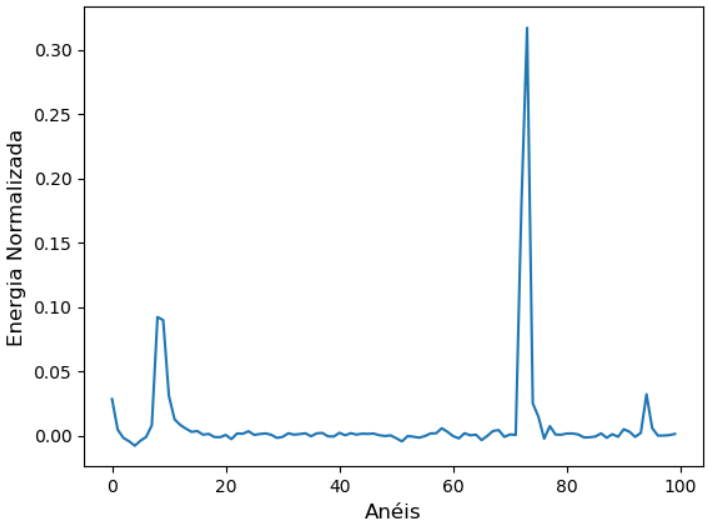
Como objeto de estudo do método de seleção de dados, foi utilizada uma base de dados real de colisões do tipo $Z \rightarrow ee$ coletadas do sistema de filtragem de elétrons do experimento ATLAS do CERN referente ao ano de 2018. A base consiste nos eventos de sinal de elétrons e ruído de fundo do experimento (*background*) na forma compactada pelo algoritmo anelador (*Ringer*) (FREUND, 2018). Cada evento é composto por 100 anéis de energia (100 dimensões) e a quantidade de eventos para cada classe (sinal e ruído) está contida na Tabela 3.1. Os dados foram normalizados pela norma 1 e exemplos de dados de sinal e ruído podem ser vistos, respectivamente, nas Figuras 3.1 e 3.2.

Tabela 3.1 – Tamanho do conjunto de dados utilizado no experimento

Classe	Desenvolvimento	Teste
Sinal	200.000	3.300.000
Ruído	200.000	7.808

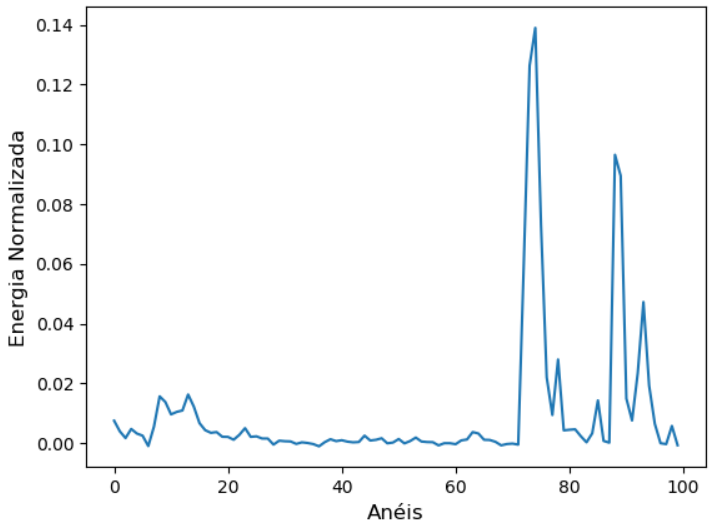
Fonte: do Autor.

Figura 3.1 – Exemplo de evento de sinal.



Fonte: do Autor.

Figura 3.2 – Exemplo de evento de ruído.



Fonte: do Autor.

A partir do conjunto de dados de desenvolvimento, foi projetada uma Curva Principal utilizando o algoritmo k-segmentos para cada classe. Após o projeto da CP, foi realizado o cálculo e o mapeamento das distâncias euclidianas de cada evento à sua respectiva Curva Principal. Por meio deste mapeamento, foram propostas 3 abordagens de seleção dos dados para o treinamento das Redes Neurais. A descrição das abordagens segue abaixo.

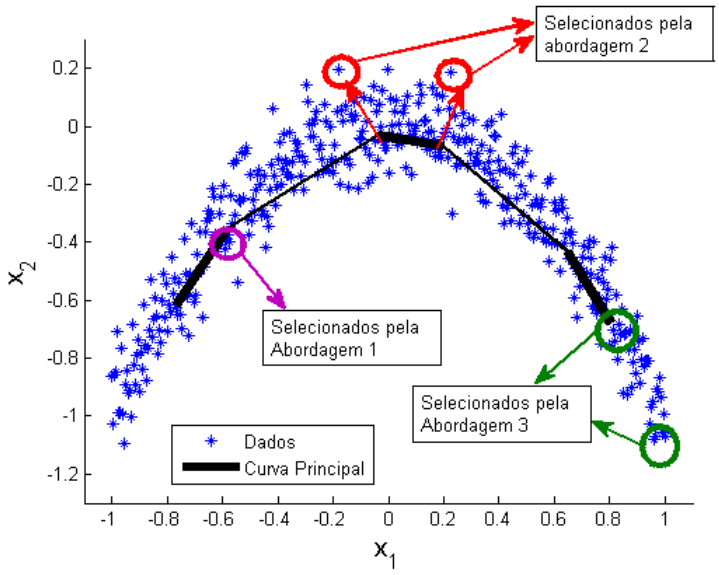
Abordagem 1: Foram selecionados os eventos mais próximos à CP, variando o tamanho do conjunto de dados reduzido (N_t);

Abordagem 2: Foram selecionados os eventos mais distantes à CP, variando o tamanho do conjunto de dados reduzido (N_t);

Abordagem 3: Foi feita uma mistura entre as abordagens 1 e 2, em que utilizaram-se os dados mais próximos com uma parcela do conjunto reduzido (pp) de dados mais distantes à CP. Nesta abordagem, tanto o tamanho do conjunto reduzido (N_t) quanto à parcela de dados mais distantes (pp) foi variada.

Como forma de ilustrar o procedimento de seleção em cada abordagem, foi gerado um conjunto de dados aleatório em duas dimensões (uma vez que os dados reais possuem 100 dimensões, não sendo possível sua visualização). Esta ilustração gerada, contendo a CP e exemplificando, em duas dimensões, o procedimento de seleção está inserida na Figura 3.3.

Figura 3.3 – Exemplo do funcionamento da seleção de dados via CP.



Fonte: do Autor.

Após o projeto da CP e a seleção de dados foi realizado o treinamento das Redes Neurais a fim de simular o procedimento do algoritmo *NeuralRinger*. A topologia utilizada foi uma Rede Neural do tipo Multicamadas (MLP - *Multi-Layer Perceptron*) contendo uma camada escondida com 10 neurônios e função de ativação de tangente hiperbólica. O treinamento foi realizado em processo de validação cruzada do tipo *k-fold* com 10 *folds*. Como medidas de avaliação foram utilizadas a Probabilidade de Detecção (P_D), a Probabilidade de Falso Alarme (P_F) e o índice Soma-Produto (SP), este dado pela equação (3.1) (FREUND, 2018). Durante o processo de validação cruzada são gerados 10 modelos neurais, sendo o modelo a ser escolhido o que apresentar o maior valor do índice SP.

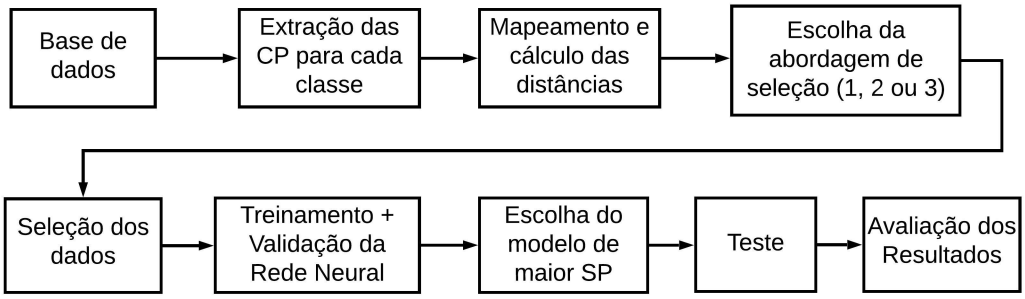
$$SP = \sqrt{\sqrt{P_D(1-P_F)}\frac{P_D+(1-P_F)}{2}} \tag{3.1}$$

3.1 Implementação

O algorimo k-segmentos proposto por (VERBEEK; VLASSIS; KRÖSE, 2002) possui sua implementação original em *software* MatLab®. Dado que a plataforma é um programa proprietário, foi proposta uma tradução *open-source* do algoritmo utilizando a linguagem Python. O código foi escrito utilizando a versão 3 do Python por meio da IDE Spyder, esta escolhida por possuir fácil acesso às variáveis do programa além da possibilidade de execução do código por trechos, facilitando o processo de escrita e depuração do código.

Com o objetivo de ilustrar o funcionamento do algoritmo implementado, foi feito um diagrama de blocos indicando os passos do método aplicado de maneira sequencial. A ilustração está contida na Figura 3.4 e um pseudocódigo do método pode ser visto no Apêndice A.

Figura 3.4 – Fluxograma do Método Aplicado.



Fonte: do Autor.

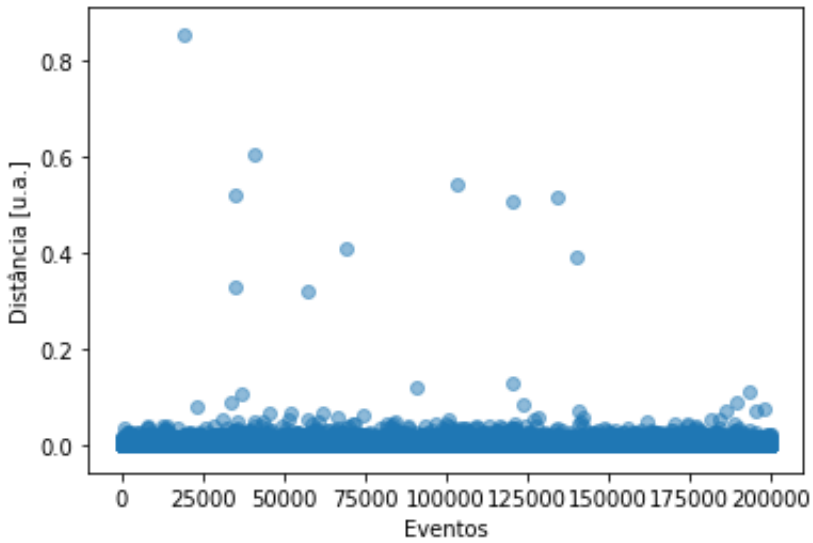
4 RESULTADOS E DISCUSSÃO

4.1 Resultados do Processo de Seleção de Dados

Após o procedimento experimental supracitado na seção 3, foram gerados os resultados do processo de seleção de dados. No tocante à geração das CP, a topologia escolhida foi com 15 segmentos, pois apresentou os melhores resultados com custo computacional menor, uma vez que aumentando o número de segmentos, não havia ganho de desempenho na seleçao de dados. Portanto, foram geradas uma CP de 15 segmentos para cada classe.

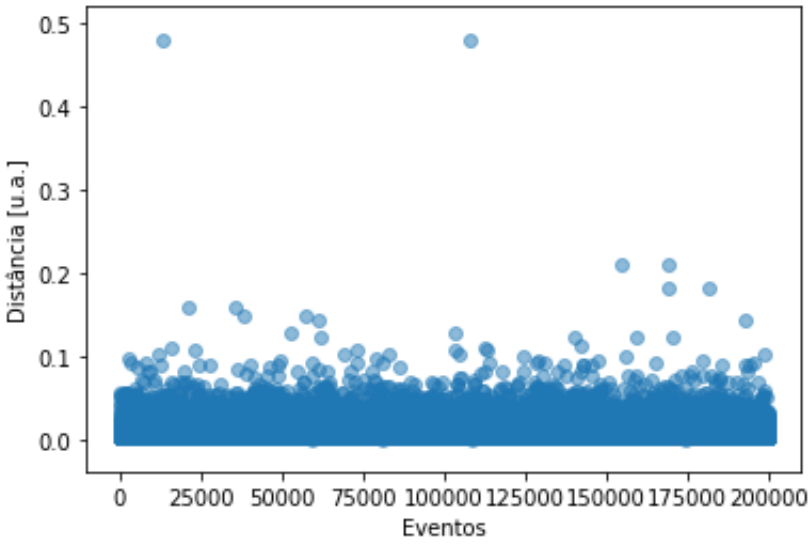
Após a obtenção das CP foi realizado o cálculo e mapeamento das distâncias euclidianas de cada evento à respectiva Curva. Em posse dos valores das distâncias, foram gerados gráficos de dispersão e histogramas das distâncias. Os gráficos de dispersão para o sinal e o ruído encontram-se, respectivamente, nas Figuras 4.1 e 4.2.

Figura 4.1 – Gráfico de dispersão das distâncias para a classe de sinal.



Fonte: do Autor.

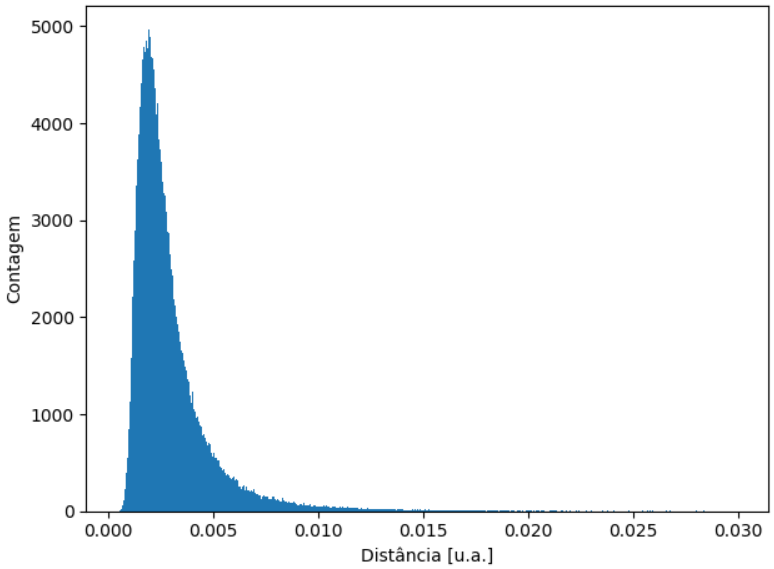
Figura 4.2 – Gráfico de dispersão das distâncias para a classe de ruído.



Fonte: do Autor.

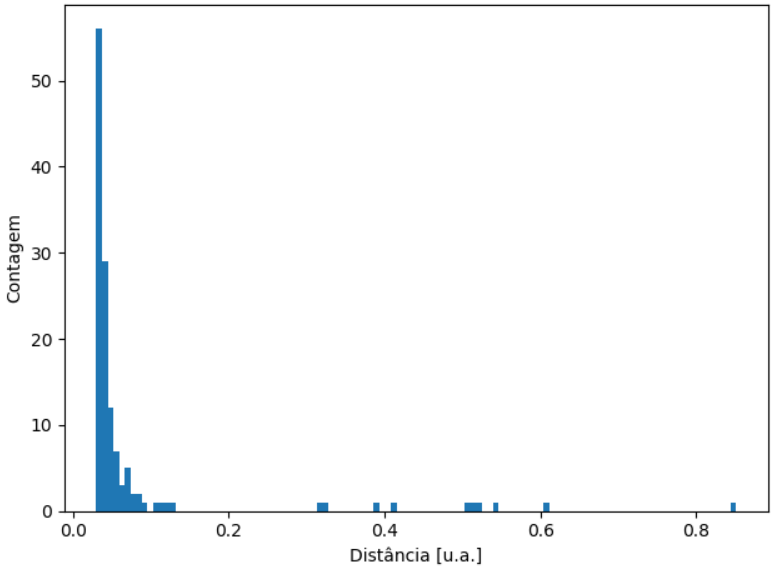
Devido à concentração da maior parte dos eventos em baixas distâncias, para a geração dos histogramas, foi realizada uma divisão do mesmo em 2 partes para que se permita uma melhor visualização da distribuição dos dados. Para isto, foi estabelecido um ponto de corte na distância $0.03[u.a.]$, assim, cada classe possuirá duas partes do histograma, uma parte para as menores distâncias ($d < 0.03[u.a.]$) e outra para as maiores distâncias ($d > 0.03[u.a.]$). Desta forma, para a classe de sinal, o histograma para as distâncias menores que o ponto de corte é mostrado na Figura 4.3, enquanto o histograma para as distâncias maiores que o ponto de corte pode ser visto na Figura 4.4. Para a classe de ruído, o histograma para as distâncias menores que o ponto de corte é apresentado na Figura 4.5 e o histograma para as distâncias maiores que o ponto de corte é mostrado na Figura 4.6.

Figura 4.3 – Histograma de distâncias para o sinal nas baixas distâncias (distâncias menores que 0.03[u.a.]).



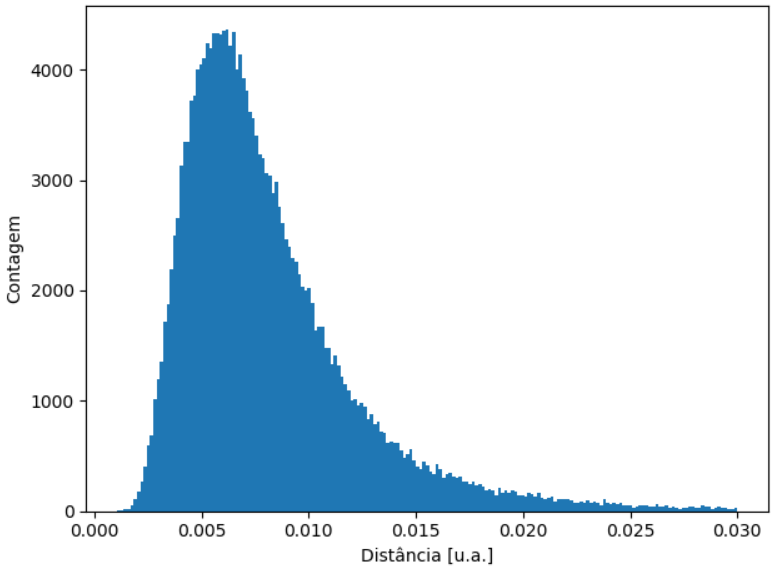
Fonte: do Autor.

Figura 4.4 – Histograma de distâncias para o sinal nas altas distâncias (distâncias maiores que 0.03[u.a.]).



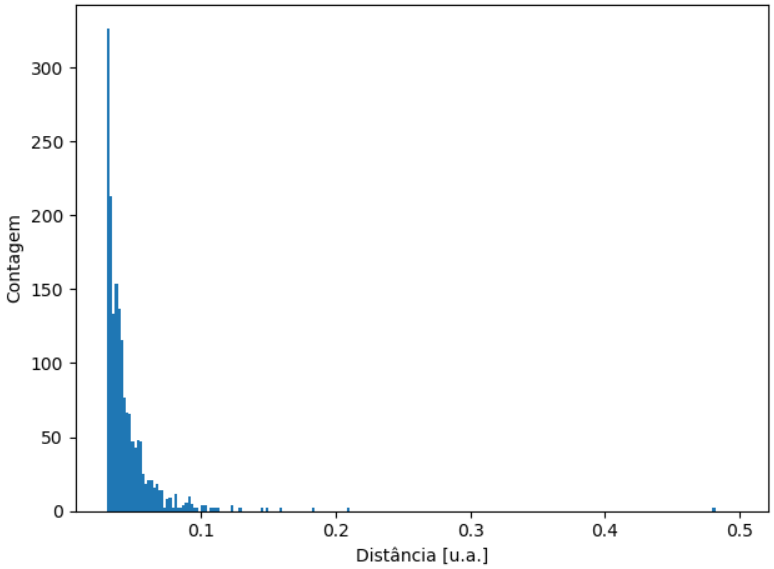
Fonte: do Autor.

Figura 4.5 – Histograma de distâncias para o ruído nas baixas distâncias (distâncias menores que 0.03[u.a.]).



Fonte: do Autor.

Figura 4.6 – Histograma de distâncias para o ruído nas altas distâncias (distâncias maiores que 0.03[u.a.]).



Fonte: do Autor.

Após o mapeamento e cálculo das distâncias dos eventos à CP e a seleção dos dados, foi realizado o treinamento das Redes Neurais e o teste das mesmas com os dados de teste. Os resultados de treinamento e teste para as Abordagens de seleção 1, 2 e 3 seguem, respectivamente, contidos nas Tabelas 4.1, 4.2, 4.3 enquanto os resultados da Rede Neural treinada com todo o conjunto de desenvolvimento, sem a realização do procedimento de seleção de dados, encontra-se na Tabela 4.4.

Tabela 4.1 – Resultados percentuais da Rede Neural (média $\pm$ desvio-padrão) utilizando a Abordagem 1 de seleção. Sendo: N_t o tamanho do conjunto de treinamento das Redes Neurais reduzido pelas CP; SP_{max} o máximo índice SP obtido no treinamento com validação cruzada; P_D a probabilidade de detecção e P_F o falso alarme.

	Treino			Teste	
N_t	SP_{max}	P_D	P_F	P_D	P_F
20.000	99,99 $\pm$ 0,02	99,99 $\pm$ 0,01	0,00 $\pm$ 0,04	94,19 $\pm$ 20,73	1,13 $\pm$ 9,57
40.000	99,99 $\pm$ 0,02	99,99 $\pm$ 0,02	0,00 $\pm$ 0,02	93,76 $\pm$ 22,58	1,19 $\pm$ 10,28
60.000	99,99 $\pm$ 0,19	99,99 $\pm$ 0,34	0,01 $\pm$ 0,17	93,36 $\pm$ 22,82	1,22 $\pm$ 10,21
80.000	99,99 $\pm$ 0,31	99,99 $\pm$ 0,52	0,00 $\pm$ 0,31	93,34 $\pm$ 23,85	1,28 $\pm$ 10,85
100.000	99,94 $\pm$ 1,23	99,95 $\pm$ 1,22	0,06 $\pm$ 1,75	94,05 $\pm$ 21,94	1,48 $\pm$ 11,52
120.000	99,84 $\pm$ 1,93	99,83 $\pm$ 2,27	0,11 $\pm$ 2,35	94,00 $\pm$ 21,24	1,75 $\pm$ 12,29
140.000	99,64 $\pm$ 3,41	99,68 $\pm$ 2,85	0,31 $\pm$ 4,50	95,17 $\pm$ 18,28	1,84 $\pm$ 12,42
160.000	99,30 $\pm$ 4,74	99,29 $\pm$ 4,63	0,51 $\pm$ 5,78	96,25 $\pm$ 15,44	2,11 $\pm$ 13,23
180.000	98,92 $\pm$ 6,21	99,05 $\pm$ 5,03	0,90 $\pm$ 8,01	97,21 $\pm$ 12,21	2,24 $\pm$ 13,39

Fonte: do Autor.

Tabela 4.2 – Resultados percentuais da Rede Neural (média ± desvio-padrão) utilizando a Abordagem 2 de seleção. Sendo: N_t o tamanho do conjunto de treinamento das Redes Neurais reduzido pelas CP; SP_{max} o máximo índice SP obtido no treinamento com validação cruzada; P_D a probabilidade de detecção e P_F o falso alarme.

	Treino			Teste	
N_t	SP_{max}	P_D	P_F	P_D	P_F
20.000	96,14 ± 12,33	96,81 ± 8,55	3,26 ± 15,78	98,36 ± 3,05	5,28 ± 19,09
40.000	96,37 ± 11,08	96,65 ± 8,96	2,92 ± 14,36	97,90 ± 4,12	3,35 ± 15,12
60.000	96,17 ± 11,28	96,53 ± 8,61	3,16 ± 15,10	98,08 ± 5,00	2,41 ± 12,36
80.000	96,37 ± 10,75	96,67 ± 8,85	2,99 ± 14,26	98,18 ± 5,77	2,13 ± 10,65
100.000	96,71 ± 10,69	97,18 ± 8,28	2,82 ± 14,06	98,22 ± 6,14	1,47 ± 8,60
120.000	97,08 ± 9,87	97,37 ± 8,15	2,43 ± 12,95	98,21 ± 6,60	1,22 ± 7,43
140.000	97,31 ± 9,47	97,67 ± 7,81	2,33 ± 12,46	98,19 ± 6,84	0,87 ± 5,92
160.000	97,51 ± 9,20	97,51 ± 9,20	2,16 ± 11,77	97,98 ± 7,87	0,92 ± 5,38
180.000	97,71 ± 8,95	97,98 ± 8,09	1,93 ± 11,26	98,21 ± 7,31	0,43 ± 3,49

Fonte: Do Autor

Tabela 4.3 – Resultados percentuais da Rede Neural (média ± desvio-padrão) utilizando a Abordagem 3 de seleção. Sendo: N_t o tamanho do conjunto de treinamento das Redes Neurais reduzido pelas CP; pp a proporção dos dados mais próximos no conjunto reduzido; SP_{max} o máximo índice SP obtido no treinamento com validação cruzada; P_D a probabilidade de detecção e P_F o falso alarme.

		Treino			Teste	
N_t	pp	SP_{max}	P_D	P_F	P_D	P_F
80.000	0.9	98,99 ± 5,97	99,13 ± 6,47	0,86 ± 6,70	98,85 ± 6,62	2,78 ± 14,25
80.000	0.8	98,52 ± 7,27	98,76 ± 6,64	1,29 ± 8,79	99,00 ± 5,33	2,97 ± 14,54
80.000	0.7	98,08 ± 8,17	98,31 ± 7,59	1,61 ± 9,76	98,87 ± 5,15	2,83 ± 14,01
80.000	0.6	97,79 ± 8,78	97,89 ± 8,19	1,67 ± 10,35	98,79 ± 5,04	2,68 ± 13,59
120.000	0.9	98,79 ± 6,22	98,91 ± 6,69	1,04 ± 7,32	98,67 ± 6,84	3,05 ± 14,87
120.000	0.8	98,33 ± 7,69	98,60 ± 7,09	1,47 ± 9,55	98,68 ± 6,09	2,89 ± 14,32
120.000	0.7	98,11 ± 7,95	98,30 ± 6,97	1,57 ± 10,08	98,66 ± 6,02	2,63 ± 13,57
120.000	0.6	97,88 ± 8,48	98,16 ± 7,25	1,82 ± 10,79	97,96 ± 7,58	3,53 ± 15,22
160.000	0.9	98,45 ± 6,99	98,52 ± 7,36	1,24 ± 8,29	98,31 ± 7,58	3,10 ± 14,93
160.000	0.8	98,24 ± 7,83	98,36 ± 7,53	1,39 ± 9,21	98,46 ± 6,92	2,91 ± 14,44
160.000	0.7	98,11 ± 7,85	98,29 ± 7,21	1,57 ± 9,89	98,44 ± 6,70	2,64 ± 13,53
160.000	0.6	97,88 ± 8,67	98,29 ± 6,86	1,92 ± 11,37	98,24 ± 6,97	2,37 ± 12,60

Fonte: do Autor.

Tabela 4.4 – Resultados percentuais da Rede Neural (média $\pm$ desvio-padrão) utilizando todo o conjunto de desenvolvimento da CP. Sendo: N_t o tamanho do conjunto de treinamento das Redes Neurais sem a redução pelas CP; SP_{max} o máximo índice SP obtido no treinamento com validação cruzada; P_D a probabilidade de detecção e P_F o falso alarme.

	Treino			Teste	
N_t	SP_{max}	P_D	P_F	P_D	P_F
200.000	98,01 $\pm$ 8,22	98,32 $\pm$ 6,82	1,76 $\pm$ 10,61	98,23 $\pm$ 7,37	1,70 $\pm$ 10,80

Fonte: do Autor.

4.2 Discussão

Analisando os resultados obtidos e apresentados na seção anterior, podem ser levantados alguns pontos em relação ao procedimento de seleção de dados: (i) o método de seleção levou à resultados similares entre o conjunto de dados sem uso redução realizada pelo método e os conjuntos de dados reduzidos pelo método de seleção; (ii) os resultados obtidos por meio da Abordagem 1 apresentaram as menores taxas de detecção, contudo, obtiveram as menores taxas de falso alarme; (iii) as abordagens de seleção 2 e 3 apresentaram melhores taxas de detecção, entretanto, seu falso alarme foi, relativamente, maior para a maioria dos valores de N_t que os resultados utilizando a Abordagem 1. (iv) em alguns valores de N_t, a abordagem 2 apresenta superioridade nos índices de P_D e P_F, contudo, não é possível apontá-la, categoricamente, como a abordagem mais apropriada para a seleção.

Outro fator a ser analisado são os resultados obtidos pelas Figuras 4.1, 4.2 (gráficos de dispersão) e pelas Figuras 4.3, 4.4, 4.5, 4.6 (histogramas). Foi apresentado por ambas as figuras uma concentração dos eventos nas baixas distâncias. Outro ponto também mostrado pelas figuras foi a elevada discrepância relativa entre as baixas e altas distâncias dos eventos. Tal circunstância sugere uma presença de *outliers*, contudo, não seria possível uma afirmação em primeira instância, sendo necessários estudos adicionais acerca desta situação, uma vez que,

a presença de eventos presentes nas altas distâncias (Abordagens 2 e 3) acarretaram em maiores valores de P_D em relação ao uso somente de eventos contidos nas baixas distâncias (Abordagem 1). Isto pode ser observado por meio do comparativo entre os valores presentes nas Tabelas 4.2 e 4.3 com os valores da Tabela 4.1.

Outro ponto em que as discrepâncias entre as altas e baixas distâncias podem influenciar são nos percentuais das incertezas das medidas de desempenho (desvios-padrão). Os resultados apresentados possuem alto desvio-padrão, principalmente para os valores de falso alarme (P_F). Tal situação pode sugerir novos estudos relacionados às abordagens de seleção, como, por exemplo, por meio de análise de agrupamentos gerados pelos segmentos das CP projetadas. Por meio destas análises, poderão ser propostas novas abordagens de seleção de dados via Curvas Principais. Buscando, assim, alternativas de reduzir as discrepâncias ocorridas no mapeamento dos eventos, acarretando em conjuntos mais uniformes.

Cabe ressaltar também, os efeitos do processo de seleção nos tempos de processamento dos dados, tal redução no tempo de processamento sendo ocasionada pela redução do conjunto de dados a serem treinados pela Rede Neural. Tais reduções possuem valores diferentes em cada abordagem, logo, para um mesmo valor de N_t, os tempos de processamento são diferentes dependendo da abordagem de seleção. Para efeitos de comparação, para um valor de N_t de 120.000 eventos Abordagem 1 mostrou uma redução de 60% no tempo de processamento em relação ao uso do conjunto total de dados de desenvolvimento, enquanto, que, nas Abordagens 2 e 3, para o mesmo número de eventos, a redução foi, respectivamente, 33% e 37%. Tal situação reforça que cada abordagem possui suas vantagens e desvantagens durante um estudo comparativo, sendo necessário levar tais pontos supracitados em conta durante a escolha de uma destas durante um processo real. Contudo, tais resultados apresentados mostraram-se promissores para um avanço nos estudos das CP como método de seleção de dados para um proce-

dimento *online* com relação à redução aos tempos de processamento sem perdas significativas da capacidade de generalização do modelo neural de aprendizagem.

5 CONCLUSÃO

No presente trabalho foi proposto um método de seleção inteligente de dados com o objetivo de reduzir tempos de processamento de Redes Neurais em grande volume de dados sem perdas de generalização do modelo, utilizando, para isto, Curvas Principais. Por meio dos testes realizados e os resultados apresentados e discutidos, foi possível identificar um potencial uso das CP como um método de seleção de dados que proporciona redução de tempo de processamento com resultados de desempenho similares entre o conjunto total de dados e os conjuntos de dados reduzidos por meio das abordagens de seleção propostas.

Tal situação pode ser observada por meio de um estudo realizado utilizando dados reais de sinais de elétrons e ruído de fundo do experimento ATLAS do CERN. Foi realizada uma simulação *offline* do processo que ocorre de maneira *online* pelo algoritmo *NeuralRinger*. Os resultados de desempenho apresentados mostram que os conjuntos compactos de dados gerados pelo método geraram valores de desempenho iguais ou, em alguns casos, superiores aos valores utilizando o conjunto de dados total. Podendo, assim, viabilizar seu uso em mais testes para, assim, poder chegar na implementação *online* do método.

Para trabalhos futuros, serão vistos testes do método no algoritmo *NeuralRinger* e análises mais aprofundadas, com o objetivo de analisar, além dos índices de desempenho, possíveis tendências que podem ser implicadas, ou não, pela seleção de dados. Tais tendências poderão ser vistas por meio de análises de quadrante e impacto. Desta forma, poderá ser avaliada de maneira detalhada a viabilidade do método de seleção para sua aplicação *online*. Tais estudos, tem como finalidade contribuir com melhorias computacionais ao sistema de filtragem online da cadeia de elétrons de maneira a manter os requisitos de desempenho e reduzir a carga de processamento em meio a um ambiente com elevadas demandas de processamento.

REFERÊNCIAS

ATLAS COLLABORATION. **ATLAS detector and physics performance: Technical Design Report, 1. Technical Design Report**. Geneva, CERN, 1999. 460 p. Disponível em: <https://cds.cern.ch/record/391176>. Acesso em: 14 ago. 2020.

ATLAS COLLABORATION. The ATLAS Experiment at the CERN Large Hadron Collider. **Jinst**, v. 3, n. 08, p. S08003, 2008.

ATLAS COLLABORATION. **Computer generated image of the whole ATLAS detector**. Geneva, CERN, 2008. Disponível em: <https://cds.cern.ch/record/1095924>. Acesso em: 11 ago. 2020.

ATLAS COLLABORATION. Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC. **Physics Letters B**, Elsevier, v. 716, n. 1, p. 1–29, 2012.

ATLAS COLLABORATION. Performance of electron and photon triggers in atlas during lhc run 2. **The European Physical Journal C**, Springer, v. 80, n. 01, p. 47, 2020.

BANFIELD, J. D.; RAFTERY, A. E. Ice floe identification in satellite images using mathematical morphology and clustering about principal curves. **Journal of the American Statistical Association**, Taylor & Francis Group, v. 87, n. 417, p. 7–16, 1992.

CMS COLLABORATION. Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC. **Physics Letters B**, Elsevier, v. 716, n. 1, p. 30–61, 2012.

DELICADO, P. Another look at principal curves and surfaces. **Journal of Multivariate Analysis**, Elsevier, v. 77, n. 1, p. 84–116, 2001.

FAN, J.; HAN, F.; LIU, H. Challenges of big data analysis. **National science review**, Oxford University Press, v. 1, n. 2, p. 293–314, 2014.

FONG, S.; WONG, R.; VASILAKOS, A. V. Accelerated pso swarm search feature selection for data stream mining big data. **IEEE transactions on services computing**, IEEE, v. 9, n. 1, p. 33–45, 2015.

FREUND, W. S. **Identificação de elétrons baseada em um calorímetro de altas energias finamente segmentado**. Tese (Doutorado) — Universidade Federal do Rio de Janeiro, 2018.

GENENDER-FELTHEIMER, A. Visualizing high dimensional and big data. **Procedia Computer Science**, Elsevier, v. 140, p. 112–121, 2018.

HASANIN, T. et al. Investigating random undersampling and feature selection on bioinformatics big data. In: IEEE. **2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService)**. Newark, 2019. p. 346–356.

HASTIE, T.; STUETZLE, W. Principal curves. **Journal of the American Statistical Association**, Taylor & Francis, v. 84, n. 406, p. 502–516, 1989.

HAYKIN, S. **Redes neurais: princípios e prática**. Porto Alegre: Bookman Editora, 2007.

HEBB, D. O. **The organization of behavior: a neuropsychological theory**. New York: J. Wiley; Chapman & Hall, 1949.

JIANG, S.; FERREIRA, J.; GONZALEZ, M. C. Activity-based human mobility patterns inferred from mobile phone data: A case study of Singapore. **IEEE Transactions on Big Data**, IEEE, v. 3, n. 2, p. 208–219, 2017.

JIN, X. et al. Significance and challenges of big data research. **Big Data Research**, Elsevier, v. 2, n. 2, p. 59–64, 2015.

KUMAR, S.; SINGH, M. Big data analytics for healthcare industry: impact, applications, and tools. **Big Data Mining and Analytics**, TUP, v. 2, n. 1, p. 48–57, 2018.

LHC. **Portal Oficial de Divulgação do LHC**. [S.l.], 2020. Disponível em: <https://lhc-machine-outreach.web.cern.ch/>. Acesso em: 11 ago. 2020.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, Springer, v. 5, n. 4, p. 115–133, 1943.

MIKALEF, P. et al. Big data analytics capabilities: a systematic literature review and research agenda. **Information Systems and e-Business Management**, Springer, v. 16, n. 3, p. 547–578, 2018.

MOHARM, K. State of the art in big data applications in microgrid: a review. **Advanced Engineering Informatics**, Elsevier, v. 42, p. 100945, 2019.

NUAIMI, E. A. et al. Applications of big data to smart cities. **Journal of Internet Services and Applications**, Springer, v. 6, n. 1, p. 25, 2015.

ONAL, A. C. et al. Weather data analysis and sensor fault detection using an extended iot framework with semantics, big data, and machine learning. In: IEEE. **2017 IEEE International Conference on Big Data (Big Data)**. Boston, 2017. p. 2037–2046.

RABHI, L. et al. Big data approach and its applications in various fields. **Procedia Computer Science**, Elsevier, v. 155, p. 599–605, 2019.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological review**, American Psychological Association, v. 65, n. 6, p. 386, 1958.

SAGIROGLU, S.; SINANC, D. Big data: A review. In: IEEE. **2013 international conference on collaboration technologies and systems (CTS)**. San Diego, 2013. p. 42–47.

TIBSHIRANI, R. Principal curves revisited. **Statistics and computing**, Springer, v. 2, n. 4, p. 183–190, 1992.

VERBEEK, J. J.; VLASSIS, N.; KRÖSE, B. A k-segments algorithm for finding principal curves. **Pattern Recognition Letters**, Elsevier, v. 23, n. 8, p. 1009–1017, 2002.

ZHANG, X.; MING, X.; YIN, D. Application of industrial big data for smart manufacturing in product service system based on system engineering using fuzzy dematel. **Journal of Cleaner Production**, Elsevier, p. 121863, 2020.

APÊNDICE A – Algoritmo para Seleção de Dados

Como forma de mostrar o procedimento de seleção de dados de maneira simplificada, é apresentado um pseudocódigo do algoritmo implementado para a seleção de dados, partindo desde a extração das CP até o teste das Redes Neurais. O pseudocódico é apresentado por meio do Algoritmo 1.

Algoritmo 1: Pseudocódigo da Seleção de Dados via CP.

Entrada: $sinal, ruido$: matrizes contendo os dados de sinal e ruído;
$dados_{teste}$: matriz contendo dados de sinal e ruído para teste da Rede Neural;
k_{max}: tamanho máximo de segmentos;
$abord$: Variável de escolha da abordagem de seleção (1,2 ou 3);
N_t: Tamanho do conjunto de dados a ser reduzido;
pp: quantidade de dados mais próximos à CP (utilizado somente na Abordagem 3).
Saída: SP_{max}: Maior SP gerado no processo de validação cruzada;
$P_DTreino$, P_DTeste: Probabilidade de detecção (treino e teste);
$P_FTreino$, P_FTeste: Probabilidade de falso alarme (treino e teste).

```
1  início
2      CP_sinal ← GeraCurva(sinal, k_max);
3      CP_ruido ← GeraCurva(ruido, k_max);
4      dist_sinal ← CalculaDistancia(sinal, CP_sinal);
5      dist_ruido ← CalculaDistancia(ruido, CP_ruido);
6      sinal_reduzido ← SelecionaDados(sinal, dist_sinal, abord, N_t, pp);
7      ruido_reduzido ← SelecionaDados(ruido, dist_ruido, abord, N_t, pp);
8      SP_max, P_DTreino, P_FTreino ←
         TreinaRedeNeural(sinal_reduzido, ruido_reduzido); //Treino feito em
         conjunto com a validação cruzada.
9      P_DTeste, P_FTeste ← TestaRedeNeural(dados_teste);
10 fim
```