Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

GUILHERME RODRIGUES DA SILVA

OPEN DATA SCIENCE: MACHINE LEARNING COM INICIATIVAS OPEN SOURCE

Assis/SP
2018

Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

GUILHERME RODRIGUES DA SILVA

OPEN DATA SCIENCE: MACHINE LEARNING COM INICIATIVAS OPEN SOURCE

Projeto de pesquisa apresentado ao curso de Ciência da Computação do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando(a): Guilherme Rodrigues da Silva
Orientador(a): Prof. MSc. Guilherme de Cleva Farto

Assis/SP
2018

FICHA CATALOGRÁFICA

SILVA, Guilherme

Open Data Science: Machine Learning com iniciativas Open Source / Guilherme Rodrigues da Silva. Fundação Educacional do Município de Assis – FEMA – Assis, 2018.

61p.

Orientador: Prof. MSc. Guilherme de Cleva Farto
Trabalho de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis – IMESA.

1. Data Science. 2. Machine Learning 3. Open Source

CDD: 001.6
Biblioteca da FEMA

OPEN DATA SCIENCE: MACHINE LEARNING COM INICIATIVAS OPEN SOURCE

GUILHERME RODRIGUES DA SILVA

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, avaliado pela seguinte comissão examinadora:

Orientador: _____
Prof. MSc. Guilherme de Cleva Farto

Examinador: _____
Prof. Dr. Luiz Carlos Begosso

Assis/SP
2018

RESUMO

Atualmente, tem-se enfrentado desafios de lidar com o grande volume e variedade de dados gerados principalmente pelos avanços tecnológicos nos processos de coleta e processamento de informações. Como consequência, houve um aumento na complexidade dos processos de análise. Responsável pela gestão desses dados vindos de várias fontes, por analisar e dar apoio à decisão através de procedimentos computacionais, aprendizagem de máquina, e análises estatísticas, a Ciência de Dados vem conquistando o interesse da comunidade científica e também no mercado de negócios. A proposta deste trabalho é a de pesquisar e compreender os conceitos de Ciência de Dados, Aprendizado de Máquina e ferramentas *Open Source* dentro do contexto, afim de explorar de maneira prática os conceitos estudados.

Palavras-chave: Ciência de Dados; Aprendizado de Máquina; Open Source; Algoritmos de Aprendizado

ABSTRACT

Challenges have now been faced to deal with the large volume and variety of data generated primarily by technological advances in the processes of data collection and processing. As a consequence, there was an increase in the complexity of the analysis processes. Responsible for the management of this data coming from various sources, for analyzing and supporting the decision through computational procedures, machine learning, and statistical analysis, Data Science has been winning the interest of the scientific community and also in the business market. The purpose of this work is to research and understand the concepts of Data Science, Machine Learning and Open Source tools within the context, in order to explore in a practical way the concepts studied.

Keywords: Data Science; Machine Learning; Open Source; Learning Algorithms

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

1. INTRODUÇÃO

Nos últimos anos, o Aprendizado de Máquina (em inglês, Machine Learning) que, de acordo com BRINK (2014), é uma ramificação da Inteligência Artificial (IA) voltada para o estudo e a construção de sistemas computacionais capazes de aprender de forma automatizada a partir de análise de dados. *Machine Learning* tem sido um dos focos de estudo dentro da área de tecnologia e com a grande disponibilidade de dados, o estudo torna-se ainda mais necessário, podendo-se aplicar técnicas para análise desses dados com inteligência, colaborando significativamente com o caminhar da tecnologia na atualidade. (SMOLA, 2008).

Pode-se encontrar estudos e soluções com *Machine Learning* diversas áreas, a área de *Marketing* com os sistemas de recomendações é uma delas. O sistema baseia-se em tradicionalmente em técnicas de Aprendizado de Máquina, com o objetivo de encontrar padrões de acordo com as ações dos usuários para que recomendações possam ser feitas. Em pesquisa realizada por KOBSA (2001), *websites* que ofereceram serviços personalizados conseguiram um aumento de 47% no número de novos clientes, reforçando a importância desse *marketing* direto por meio do sistema de recomendações.

Encontra-se estudos também na área de trânsito urbano, tendo em vista a crescente urbanização e por consequência o aumento da quantidade de veículos circulando nos centros urbanos causando mais congestionamentos. Segundo CINTRA (2014), o desperdício de recursos por causa dos congestionamentos na cidade de São Paulo foi de 40 bilhões de reais em 2012, o que corresponde a 7,6% do PIB da cidade nesse ano. Uma redução de 27,63% no tempo de viagem dos motoristas da região metropolitana de São Paulo corresponderia a um aumento de 15,75% na produtividade dos trabalhadores na região, o que geraria um aumento Um dos estudos mais recentes na área (CASTRO, 2017), visa por meio da temporização inteligente dos semáforos, que será ajustada de acordo com o fluxo da via, com o intuito de otimizar os fluxos dos veículos, reduzindo o tempo de viagem e prevenindo congestionamentos.

Os meios de coleta de dados têm aumentado recentemente, seja com Internet das Coisas (do inglês, *Internet of Things*) com seus sensores, ou por meio de algoritmos capazes de retirar dados de acordo com o comportamento do usuário na internet e, como consequência

a quantidade de dados gerados também aumentou. Tecnologias como Hadoop, Apache Cassandra, dentre outras permitiram que os dados pudessem ser processados, armazenados e analisados. Este cenário gerou uma mudança nos desafios científicos. O antigo problema da escassez dos dados, foi substituído pela dificuldade em gerenciar o seu excesso, variedade e distribuição (HEY et al. 2009).

Posto isso, soluções que sejam capazes de obter informações por meio de dados para tomadas de decisões, crescem cada vez mais. Paralelamente tem-se também o crescimento da Ciência de Dados (em inglês, Data Science), que é, DHAR (2013), o campo de estudo interdisciplinar que utiliza técnicas de diversas áreas científicas para definir processos e sistemas para a extração de informação e conhecimento a partir dos dados. Uma área que possui diversas contribuições que trabalham com *Data Science* é a da saúde.

Métodos estatísticos, capazes de analisar grandes quantidades de dados biológicos, vêm sendo desenvolvidos e implementados há algum tempo na tentativa de identificar e predizer as funções dos genes e proteínas por eles codificados (WANG et al., 2003). Resultados obtidos por meio de análise *in silico* – por meio de simulação computacional - e confirmadas posteriormente em laboratório (SILVA, 2010), sugerem que é possível a identificação de uma família inteira de genes, por meio de processos de Ciência de Dados, a partir de informações genéticas depositadas em banco de dados públicos.

É comum presenciar *Machine Learning* e *Data Science* correlacionados, tendo em vista que os conceitos e definições destas duas áreas possuem certa relação, que pode ser vista no Diagrama Venn de Ciência de Dados (Conway, 2010), ilustrado na Figura 1. De acordo com ZHOU (2003), sem as técnicas de análise de dados do aprendizado de máquina, aplicar *Data Science* seria extremamente complexo.

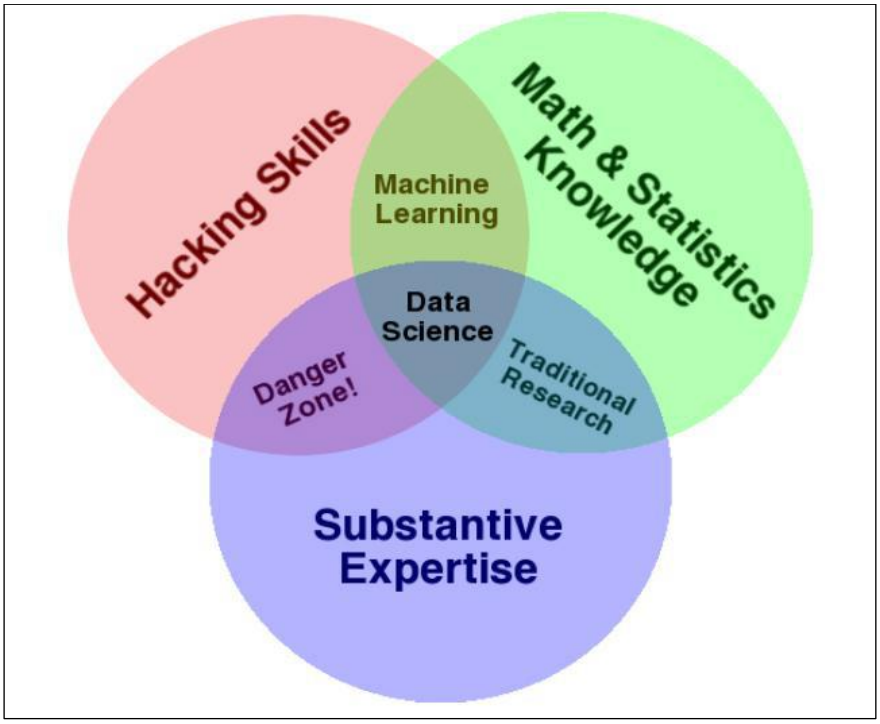

Figura 1: Diagrama Venn de Ciência de Dados (CONWAY, 2010)

Nesta pesquisa será utilizado algumas iniciativas *Open Source*, sendo as principais TensorFlow, uma biblioteca para tarefas de aprendizagem de máquina, desenvolvida pelo Google, que utiliza grafos que podem ser usados para representar os nós dos modelos de redes neurais, bem como as arestas que representam os tensores. Keras, biblioteca para trabalhar com rede neural de alto nível escrita em Python e é utilizada como *frontend* em TensorFlow (KERAS, 2017).

Possui uma prototipagem rápida e fácil, suporte a redes convolucionais e recorrentes, suporte também para esquemas de conectividade arbitrária. Segundo PEDREGOSA et al. (2011), o Scikit-Learn é biblioteca de *Machine Learning* para programação em Python, ela inclui diversos algoritmos de classificação, regressão e agrupamento, e é projetada para interagir com as bibliotecas Python numéricas e científicas.

1.1 OBJETIVOS

O objetivo geral deste projeto de pesquisa é o de identificar iniciativas *open source* no contexto de *Machine Learning*, bem como explorar de maneira e aplicar conceitos de Aprendizagem de Máquina para contribuir com a democratização da Inteligência Artificial (IA). Entende-se por democratização na IA como sendo o compartilhamento de um estudo exploratório na área de *Data Science*. Fomentado novos estudos trabalhos a partir dos fundamentos identificados por esta pesquisa.

Como resultados desta pesquisa, espera-se contribuir com um material que identifica e explora, de maneira prática, ferramentas, métodos e tecnologias relacionadas à IA e *Machine Learning*. Também faz parte dos resultados, as discussões sobre desafios e dificuldades no uso de tais recursos para promover Open Data Science.

1.2 JUSTIFICATIVAS

Tem-se notado um aumento significativo no interesse da academia e empresas, sobre *Data Science* e *Machine Learning*, de acordo com DAVENPORT et al. (2006), o surgimento de uma nova forma de competição baseada no uso intensivo de análise, dados e tomada de decisões baseadas em fatos fez com que, no lugar de competir em fatores tradicionais, as empresas começassem a empregar estatística, análise quantitativa e modelagem preditiva como elementos primários de concorrência, ou seja, IA e Ciência de Dados.

Em uma pesquisa realizada por MCKINSEY (2016), analisando todas as profissões existentes nos Estados Unidos, um terço do tempo que as passam no local de trabalho é usado para atividades de coleta e processamento de dados, corroborando ainda mais de que se faz necessário pesquisas e estudos nessa área, afim de auxiliar nessa coleta e processamento dos dados.

Segundo FREIRE et al. (2014), o reuso dos dados publicados para a geração de novos conhecimentos também é essencial para a evolução da ciência, além de possibilitar a reanálise de evidências, a minimização da duplicação de esforços e a aceleração da inovação.

Diante do exposto, torna-se oportuno elaborar e compartilhar um material de apoio relacionado à IA e Aprendizado de Máquina, para auxiliar na simplificação da aprendizagem dessas áreas, de forma livre.

1.3 MOTIVAÇÃO

IA e *Data Science* tem crescido exponencialmente nos últimos anos, pode-se notar que ferramentas baseadas em Inteligência Artificial e Ciências de Dados estão presentes na atualidade, seja por reconhecimento de voz – Siri, Google Now, Cortana -, Alexa, Amazon Echo, funções de autocorreção, carros autônomos, reconhecimento facial e *chatbots.* Técnicas de *Machine Learning* são conhecidas há algum tempo, mas só com os avanços tecnológicos e com a grande quantidade de dados disponíveis é que se tornaram melhores praticáveis.

O desenvolvimento deste projeto de pesquisa é motivado pelo fato de que a área de Inteligência Artificial e *Data Science* têm sido exploradas de maneira crescente nos últimos anos, além de serem disciplinas que, não só agrega no meio tecnológico, mas também com todas as outras áreas imagináveis. Apesar desta pesquisa investigar áreas emergentes, destaca-se que os resultados deste estudo contribuirão com trabalhos futuros que demandam por iniciativas *open source* em *Data Science*.

Outra motivação para realização desta pesquisa é o interesse em contribuir com o compartilhamento de conhecimento das iniciativas *open source* em *Data Science,* auxiliando na formação inicial de profissionais que buscam conteúdo sobre tal assunto.

1.4 PERSPECTIVAS DE CONTRIBUIÇÃO

O presente trabalho tem como perspectivas de contribuição:

- Material teórico explorando conceitos de IA, *Machine Learning* e *Data Science;*
- Material prático com exemplos de implementações de algoritmos de Aprendizado de Máquina e Ciência de Dados com iniciativas *Open Source;*
- Ambiente configurado com resultados do trabalho para futuras pesquisas;
- Publicá-la em forma de artigos;

- Divulgá-la em instituições de ensino com o objetivo de promover e compartilhar os conhecimentos e resultados alcançados;

1.5 METODOLOGIA DE PESQUISA

A proposta e objetivo deste trabalho acadêmico serão alcançados por meio de pesquisas teóricas, de forma a adquirir os conhecimentos necessários por meio da leitura de materiais científicos, livros, monografias, dissertações, teses, guias práticos e técnicos e fontes digitais confiáveis, tornando possível a elaboração e implementação de um referencial teórico que contempla os assuntos estudados.

Posteriormente à revisão da literatura, serão realizados estudos de métodos específicos com base em iniciativas *open source* no contexto de *Data Science.* Dessa forma, trabalhos futuros poderão investigar, em detalhes, ferramentas, algoritmos e demais soluções de IA e *Data Science* mencionados por esta pesquisa.

Além da revisão teórica, também serão realizados estudos e desenvolvimentos práticos para aplicar parte das iniciativas descritas, auxiliando a apresentação e discussão dos resultados identificados, bem como das limitações impostas por tais recursos.

1.6 ESTRUTURA DO TRABALHO

A estrutura deste trabalho será composta das seguintes partes:

- **Capítulo 1 – Introdução:** Neste capítulo, serão brevemente contextualizadas as principais áreas desta pesquisa, bem como relatados os objetivos, as motivações, as justificativas e a metodologia de pesquisa adotada;
- **Capítulo 2 – Data Science:** Neste capítulo, serão apresentados os fundamentos e principais conceitos de *Data Science* como abordagem para análise de dados;
- **Capítulo 3 – Machine Learning:** Neste capítulo, maior ênfase será direcionada à *Machine Learning,* apresentando recursos que possibilitam a implementação de algoritmos que apoiam *Data Science;*

- **Capítulo 4 – Iniciativa Open Source em Data Science:** Neste capítulo, serão investigadas e relatadas as iniciativas *open source* no contexto de *Data Science* e *Machine Learning;*
- **Capítulo 5 – Proposta e Desenvolvimento do Trabalho**: Neste capítulo, será modelada uma proposta de avaliação de iniciativas *open source* em *Data Science,* descrevendo o que se pretende experimentar de maneira prática, também serão apresentados os artefatos desenvolvidos, em forma de exemplos e aplicações, com base nas iniciativas identificas;
- **Capítulo 6 – Conclusão:** Neste capítulo, serão revisitados os principais tópicos explorados nesta pesquisa, bem como os desenvolvimentos conduzidos para avaliar experimentalmente o contexto de *Data Science e Machine Learning.*
- **Referências**

2. DATA SCIENCE

A ciência de dados (*Data Science*) tem sido estudada e considerada como uma área com característica interdisciplinar por parte dos pesquisadores da temática (CONWAY, 2010; STANTON, 2012; ZHU, XIONG, 2015) ou multidisciplinar (TIERNEY, 2016). Dentro do contexto de grande volume de dados (*Big Data*), há três linhas de pesquisas que podem ser exploradas com vistas à consolidação da área de *Data Science*, a saber:

- Gerência de dados;
- Análise de dados;
- Análise de Redes Complexas.

Nesses aspectos fundamentais de análise de dados em larga escala, há também um grande potencial tecnológico na pesquisa aplicada em ciência de dados com impacto em diferentes áreas do conhecimento e do conhecimento e de setores de atuação. (PORTO; ZIVIANI, 2014).

Data Science utiliza métodos e técnicas semelhantes ao da Ciência da Computação, que incluem aquisição dos dados, gestão, armazenamento, segurança, análise e visualização de dado, sendo que de modos distintos. Nesta perspectiva, COWNAY (2010) criou o Diagrama de Venn para especificar as habilidades que compete à área (Figura 1) englobando outras disciplinas. No diagrama, a Ciência de Dados aparece no centro, em lugar de destaque, indicando a ascensão da área e a correlação com outras capacidades como conhecimento de matemática e estatística, habilidades hacker e expertise substantiva, além de aprendizagem de máquina. Aspectos como conhecimentos em estatística e matemática e habilidades de *hacking* e aprendizagem de máquinas aparecem em aspectos cruzados para operacionalizar e gerenciar grandes quantidades de dados em contexto de Ciência de Dados.

2.1 DADOS, INFORMAÇÃO E CONHECIMENTO

Primeiramente, é importante definir dados, informação e conhecimento, que apesar de serem termos relacionados, possuem diferente definições na literatura.

Os dados são as representações de fatos na forma de textos, números, gráficos, imagens, sons ou vídeos. Fatos são capturados, armazenados e expressados como dados (DAMA INTERNACIONAL, 2009).

A informação são os dados em um contexto. Sem um contexto, os dados não possuem sentido. Uma informação significativa é criada a partir da interpretação do contexto relacionado aos dados. Este contexto deve incluir:

- O significado dos elementos e dos termos relacionados;
- O formato no qual os dados estão representados;
- O período representado pelos dados;
- Os objetivos buscados durante a geração destes dados;
- A relevância dos dados para um determinado uso.

Os dados são a matéria-prima que os consumidores de dados interpretam para continuamente gerar informação, conforme mostrado na Figura 2. A informação resultante deste processo direciona nossas decisões.

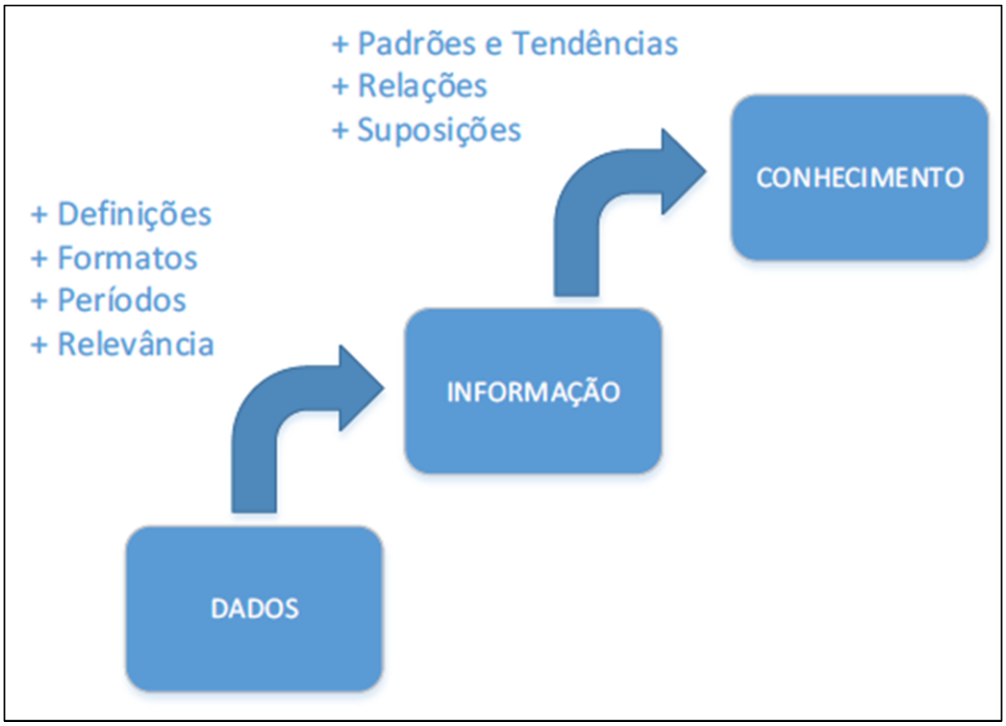

Figura 2: Dados, informação e conhecimento (Dama Internacional, 2009)

As informações contribuem para o conhecimento. O conhecimento é o entendimento, consciência e o reconhecimento de uma situação. O conhecimento é a informação em uma perspectiva, integrado a um ponto de vista com base no reconhecimento e interpretação de padrões, tais como tendências, formadas a partir de outras informações e experiências. Podendo ser incluídas também hipóteses e teorias. O conhecimento é obtido quando o significado das informações é compreendido (DAMA INTERNACIONAL, 2009).

Os dados são a base da informação e do conhecimento, mas dados imprecisos, incompletos, desatualizados e incompreensíveis podem gerar falsas afirmações, que irão apoiar decisões incorretas. Por isso, a grande importância do reconhecimento da gestão de dados como parte fundamental dos processos realizadas nas organizações.

Vivencia-se o quarto paradigma da ciência, o qual tem redefinido o modo de operar da ciência como consequência dos desafios impostos pela produção de dados em larga escala. A era *Big Data* também revolucionou o mundo dos negócios e vem exigindo uma nova postura das organizações para lidar com o grande volume e variedade de dados tanto estruturados, quanto não-estruturados, produzidos diariamente, de modo a subsidiar melhores decisões estratégicas (GONÇALVES e CERVANTES, 2016).

Como resultado destas transformações na ciência e no mundo dos negócios, e como forma de responder às demandas existentes, observa-se a expansão de uma área de estudo, interdisciplinar e intensivamente computacional: a ciência de dados. A ciência orientada a dados se vale do potencial de robustas ciberinfraestruturas de informação e comunicação, incluindo tecnologias de grids, e padrões que possibilitam a interoperabilidade e a interligação de dados. Tais padrões e ciberinfraestruturas sustentam as diferentes fases do ciclo de vida de grandes e heterogêneas coleções de dados disponíveis na web e em repositórios de dados digitais, com vistas a atribuir sentido e extrair *insights* de dados aplicáveis a diferentes domínios e contextos, para a resolução de problemas práticos e reais (GONÇALVES e CERVANTES, 2016).

2.2 GESTÃO DE DADOS

A gestão de dados, do inglês, *Data Management,* é a disciplina responsável por definir, planejar, implantar e executar estratégias, procedimentos e práticas necessárias para a

gestão efetiva dos recursos de dados e informações das organizações, incluindo planos para a sua definição, padronização, organização, proteção e utilização (DAMA INTERNACIONAL, 2009). As atividades de gestão de dados são consideradas nas diversas áreas de uma instituição, estendendo-se aos fornecedores, parceiros e consumidores. No nível institucional, envolve desde os gestores, que utilizam dados para a tomada de decisões, até profissionais de nível operacional, que são responsáveis pela coleta, produção e análise de dados.

Entre os objetivos da gestão de dados os principais são (DAMA INTERNACIONAL, 2009; STRASSER et al., 2012):

- Garantia da qualidade dos dados;
- Correta utilização e reutilização dos dados;
- Decisões ágeis e corretas baseadas nos dados;
- Confiabilidade e proveniência dos dados;
- Segurança e gestão de riscos;
- Manutenção dos dados a longo prazo.

2.3 BIG DATA

O termo *Big Data* é amplo e ainda não existe um consenso em sua definição.

Para MADDEN (2012), o conceito é definido em relação à existência e aplicabilidade dos três V’s:

- Velocidade: Consiste em processar o mais rápido quanto possível o conjunto, independentemente de seu volume de dados, ou seja, os algoritmos de processamento, idealmente, precisam ter complexidade sublinear;
- Variedade: Corresponde ao formato dos dados provenientes de diversas fontes que precisam ser integrados, possuindo esquemas diferentes em cada repositório e domínios diferentes, tal como imagens, vídeos, áudios, documentos, dentre outros.
- Volume: Compreende o conceito que os conjuntos de dados vêm acumulando grande quantidade de itens (cardinalidade) e medidas (dimensionalidade).

Segundo SMITH (2012), *Big Data* refere-se ao processamento e análise de repositórios de dados extremamente grandes e que não seriam possíveis se processar ou analisar com as ferramentas convencionais de análise de dados. MAYER-SCHONB e CUKIER (2014), mencionam que *Big Data* compete a grandes conjuntos de dados que são difíceis de armazenar, pesquisar, visualizar e analisar como, por exemplo, uma empresa aérea que coleta 10 *terabytes* de dados de sensores durante 30 minutos de vôo do avião.

Outra definição para *Big Data* é realizada por LOUKIDES (2010) e está relacionada ao fato de que quando o tamanho do conjunto de dados faz parte do problema ou as técnicas existentes deixam de ser eficientes, trata-se de *Big Data*. Seguindo a mesma linha, JACOBS (2009) define que pode-se chamar de *big* qualquer volume de dados que requisite a utilização ou criação de novas metodologias de processamento.

2.4 PROCESSO DE DESCOBERTA DE CONHECIMENTO EM BANCOS DE DADOS

O processo de descoberta de conhecimento em bancos de dados (KDD – *Knowledge Discovery in Databases*) é formalmente definido por FAYYAD (1996) como um processo não trivial de identificação de padrões contidos nos dados que sejam válidos, novos, potencialmente úteis e compreensíveis. De acordo com HAN, KAMBER e PEI (2011), trata-se de um processo que pode ser dividido em sete etapas, como mostrado na Figura 3: limpeza dos dados e integração (realizadas no pré-processamento), seleção, transformação, mineração, avaliação dos padrões e apresentação do conhecimento.

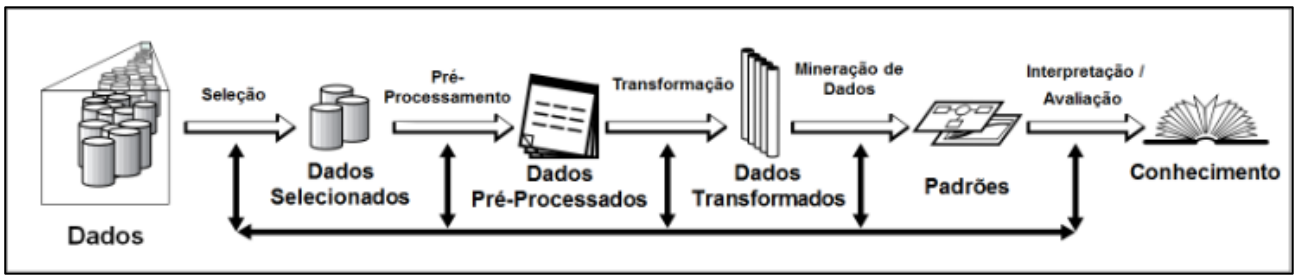

Figura 3: Conhecimento em Bancos de Dados (Fayyad, 1996)

2.4.1 SELEÇÃO DE DADOS

A fase de seleção dos dados é a primeira no processo de descobrimento de informação e possui impacto significativo sobre a qualidade do resultado final, uma vez que nesta fase é escolhido o conjunto de dados contendo todas as possíveis variáveis e registros que farão parte da análise. Normalmente essa escolha dos dados fica a critério de um especialista do domínio.

O processo de seleção é bastante complexo, uma vez que os dados podem vim de diversas fontes diferentes (*Data Warehouses,* planilhas, sistemas legados) e podem possuir os mais diferentes formatos.

2.4.2 PRÉ-PROCESSAMENTO E LIMPEZA

Esta é uma parte crucial no processo, pois a qualidade dos dados vai determinar a eficiência dos algoritmos de mineração. Nesta etapa deverão ser realizadas tarefas que eliminem dados redundantes e inconsistentes, recuperem dados incompletos e avaliem possíveis dados discrepantes ao conjunto. O auxílio de um especialista é fundamental.

Nesta fase também são utilizados métodos de redução ou transformação para diminuir o número de variáveis envolvidas no processo, visando com isto melhorar o desempenho do algoritmo de análise. (HAN, KAMBER e PEI 2011).

Um problema constante nesta fase é a ausência de valores (*missing values*) para determinadas variáveis, ou seja, registros com dados incompletos, seja por falhas no processo de seleção ou de revisão. O tratamento desses casos é necessário para que os resultados do processo de mineração sejam confiáveis. Como solução deste problema, FAYYAD (1996) propõe três alternativas:

- Usar técnicas de imputação (fazer a previsão dos dados ausentes e completa-los individualmente);
- Substituir o valor faltante pela média aritmética da variável;
- Excluir o registro inteiro.

Dados que possuem valores extremos, atípicos ou com características bastante distintas dos demais registros são chamados de discrepantes, ou *outliers.*

Normalmente, registros que contêm valores discrepantes não serão aproveitados da amostra, porém isto só deve ocorrer quando o dado *outlier* representar um erro de observação, de medida ou algum outro problema similar. Deve-se observar cautelosamente o dado antes da exclusão, pois embora atípico, o valor pode representar um dado verdadeiro. *Outliers* podem representar, por exemplo, um comportamento não usual, uma tendência ou ainda transações fraudulentas (DINIZ, 2000).

2.4.3 TRANSFORMAÇÃO DOS DADOS

Após serem selecionados, limpos e pré-processados os dados necessitam ser armazenados e formatados adequadamente para que os algoritmos de aprendizado possam ser aplicados (FAYYAD, 1996).

Nesta fase, se necessário, é possível obter dados faltantes através da transformação ou combinação de outros (dados derivados). Um exemplo de um dado que pode ser calculado a partir de outro é a idade de um indivíduo, que pode ser encontrada a partir da sua data de nascimento. Outro exemplo é o valor total de um financiamento que pode ser calculado a partir da multiplicação do número de parcelas pelo valor da parcela.

2.4.4 MINERAÇÃO DE DADOS

A mineração de dados (*Data Mining*) é o processo de descobrir informações relevantes como padrões, associações, mudanças, anomalias e estruturas, em grandes quantidades de dados armazenados em banco de dados, depósitos de dados ou outros depósitos de informação. *Data Mining* fornece percepções dos dados, descobrindo padrões e relacionamentos ocultos em grandes bancos de dados e inferindo regras a partir deles, para prever comportamentos futuros (ZAKI; MINEIRA, 2014).

Segundo, HAN, KAMBER e PEI (2011) a mineração de dados é definida como o processo de descoberta de padrões que venham a ajudar os analistas na avaliação e otimização de processos de produção, negócios, prever o futuro comportamento dos dados, auxiliar na decisão estratégica, dentre outros.

De acordo com FAYYD (1997), as técnicas de *Data Mining* podem ser aplicadas a tarefas como:

- Associação: Determina quais fatos ou objetos tendem a ocorrerem juntos num mesmo evento;
- Classificação: Construção de um modelo que possa ser aplicado a dados não classificados visando categorizar os objetos em classes;
- Predição/Previsão: Usada para definir um provável valor para uma ou mais variáveis;
- Segmentação: Visa dividir uma população em subgrupos o mais heterogêneos possível entre si;
- Sumarização: Métodos para encontrar uma descrição compacta para um subconjunto de dados.

2.4.5 INTERPRETAÇÃO E AVALIAÇÃO

Esta é mais uma fase que deve ser feita em conjunto com um ou mais especialistas no assunto. O conhecimento adquirido através da técnica de *data mining* deve ser interpretado e avaliado para que o objetivo final seja alcançado.

Caso o resultado não seja satisfatório, o processo pode retornar a qualquer um dos estágios anteriores ou até mesmo recomeçado, conforme pode ser observado na Figura 3. As ações

mais comuns caso o resultado não seja satisfatório são: modificar o conjunto de dados inicial e/ou trocar o algoritmo de *data mining*, ou alterar suas configurações de entrada.

2.5 VISUALIZAÇÃO DE DADOS

A visualização de dados vem se tornando mais frequente, tanto do ponto de vista de abordagem acadêmica, quanto do ponto de vista de alargamento dos usos na mídia impressa e digital, tornando-se comuns como modelos que visam à representação visual de grandes volumes de dados. Na literatura, existem algumas terminologias e definições sobre visualização de dados (MEIRELES, 2010; VIÉGAS, 2013). De forma mais ampla, a visualização é o resultado de uma tecnologia plural que transforma dados complexos em informação semântica e facilita a interação por meio de ferramentas para que qualquer usuário complete o processo de modo autônomo.

Segundo MEIRELES (2010), a visualização de dados conceitua-se como representações de dados que pode assumir diferentes formas, tais como sistemas de notação, mapas, diagramas, explorações de dados interativos, e outras invenções gráficas. A visualização de dados é o processo de utilização de tecnologias mediadas por computador e digitais para exibir informações quantitativas e qualitativas. As visualizações de dados estão ficando cada vez mais complexas para narrativas sofisticadas que se utilizam de mapas com dados que permitem interação (VIÉGAS, 2013).

De acordo com VIÉGAS (2013) os dados podem ser aprofundados numa visualização, já que são complexas e tendem a ter uma malha informacional maior que não se limita a apenas apresentar, mas explorar e analisar. SEGEL e HEER (2010), diz que os dados, às vezes, não contam uma história convincente por si só, mas deve haver uma narrativa que relaciona as consequências reais e causar o impacto no usuário.

SEGEL e HEER (2010) definem dois parâmetros que auxiliam na decodificação dos dados complexos:

- Visualização assistida por informações: É fornecido ao usuário um segundo formato de visualização que normalmente exibe informações sobre um conjunto de dados, mas também pode apresentar atributos da visualização do processo, das propriedades dos resultados, ou das características dos comportamentos de percepção do usuário.

- Visualização assistida por conhecimento: O conhecimento do usuário é um aspecto indispensável, uma vez que pode-se atribuir cores dependendo do conhecimento.

Segundo MURRAY (2013), a visualização de dados também se configura num campo interdisciplinar, e que na era de dos grandes volumes de dados há uma sobrecarga que precisa ser decodificada de um modo compreensível.

2.6 SISTEMA DE APOIO À DECISÃO

Segundo TWEEDALE, PHILLIPS-WHEN e JAIN (2016) os sistemas de apoio à decisão são definidos como software que visam melhorar a tomada de decisão individual ou coletiva, combinando conhecimento dos tomadores de decisão dados relevantes de fontes confiáveis, nos quais são aplicados conceitos de *Data Science*, ou seja, métodos e modelos matemáticos, análise de dados, programação, para suportar a análise, comparação e escolha de alternativas no processo de decisão.

Os sistemas de apoio à decisão apoiam o entendimento de processos complexos, auxiliam na comparação dos fenômenos envolvidos e suportam a análise e escolha de alternativas no processo de decisão. A compreensão do domínio surge da combinação das habilidades e métodos dos especialistas à capacidade das máquinas de acessar dados, estruturá-los em modelos, interpretar, formular e avaliar alternativas e cenários diferentes (HEINZLE, GAUTHIER e FIALHO, 2010).

A arquitetura de um sistema de apoio à decisão pode ser representada de acordo com a Figura 4, no qual recebem uma entrada, fazem o processamento dela e retorna resultados que são analisadas pelo tomador de decisão (TWEEDALE, PHILLIPS-WHEN e JAIN, 2016).

Nota-se na Figura 4 que os componentes de um sistema de apoio à decisão são agrupados em:

- Entradas (*Inputs*): Corresponde às entradas do sistema, composta dos dados que serão processados e dos modelos de conhecimento dos especialistas.
- Processamento (*Processing*): Composto pelos modelos e métodos de organização e processamento de dados, que têm restrições para avaliar as alternativas de resposta.

- Saídas (*Outputs*): São os resultados do processamento dos *inputs* e permitem comparar as alternativas de decisão.

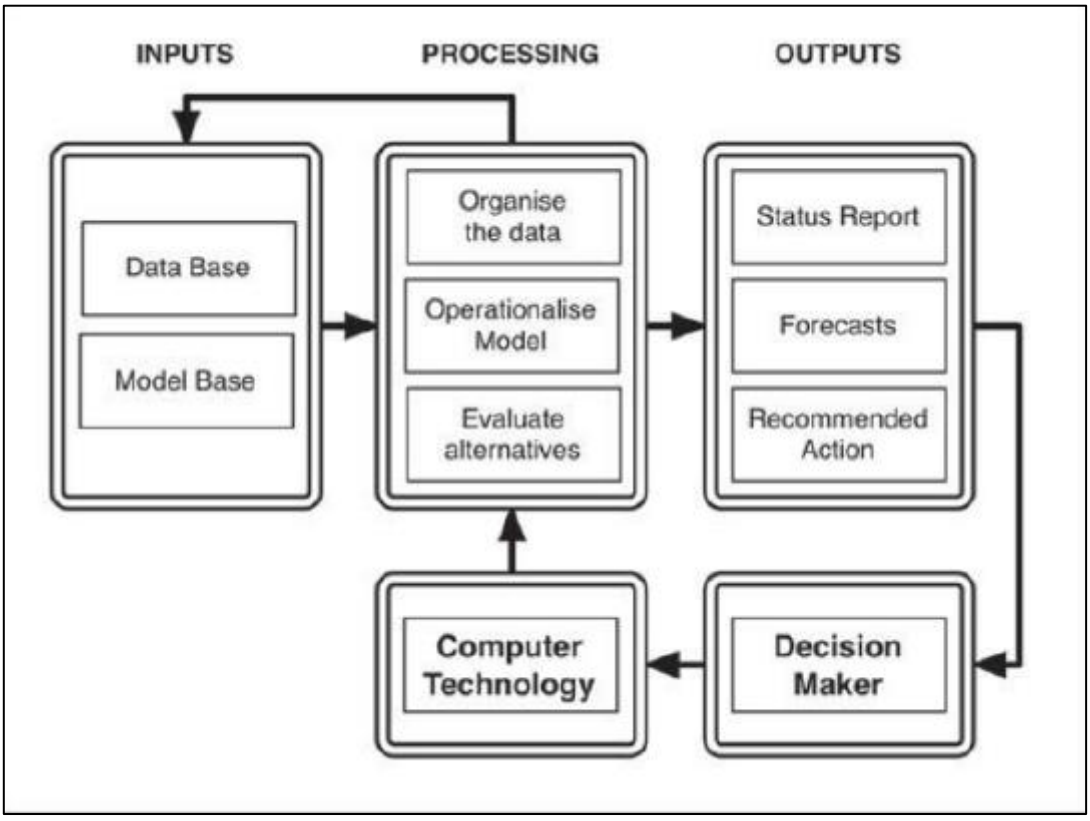

Figura 4: Componentes Sistema de apoio à decisão (Tweedale, 2016).

3. MACHINE LEARNING

3.1 INTRODUÇÃO

Aprendizado de máquina é um ramo da Inteligência Artificial cujo objetivo é o desenvolvimento de táticas computacionais sobre o aprendizado bem como a construção de sistemas capazes de adquirir conhecimento de forma automática. Um sistema de aprendizado é um programa de computador que toma decisões fundamentado em experiências acumuladas através da solução bem-sucedida de problemas anteriores (BARANAUSKAS e MONARD, 2006).

3.2 MODOS DE APRENDIZAGEM

De acordo com LORENA e CARVALHO (2007), para alguns sistemas de aprendizagem é necessário predizer se uma certa ação irá nos prover uma saída. Nesse contexto, é possível classificar os modos de aprendizagem em quatro principais categorias: Aprendizado supervisionado, aprendizado não supervisionado, aprendizado semi-supervisionado e aprendizado por reforço.

- Aprendizado supervisionado: É fornecido uma referência do objetivo a ser perseguido, isto é, um treinamento com o conhecimento de ambiente. Este treinamento são conjuntos de exemplos com entradas, rotuladas com a saída esperada. O algoritmo de aprendizagem obtém a representação do conhecimento por meio desses exemplos. O objetivo é que a representação gerada seja capaz de produzir saídas corretas para entradas novas que não foram apresentadas antes.
- Aprendizado não supervisionado: Nesta abordagem de aprendizado, os algoritmos procuram agrupar os exemplos históricos em função de características semelhantes que eles apresentam entre si. Desta forma, exemplos com características mais semelhantes tendem a ficar em um mesmo grupo, enquanto exemplos com características distintas tendem a ser organizados em grupos diferentes.
- Aprendizado semi-supervisionado: Dado um pequeno conjunto de observações ou exemplos rotulados e um conjunto de observações ou exemplos não rotulados, o objetivo da aprendizagem é utilizar os dois conjuntos para encontrar uma hipótese capaz de classificar novos exemplos entre as classes já existentes. O aprendizado

semi-supervisionado é um meio termo entre o aprendizado supervisionado e o aprendizado não supervisionado.

- Aprendizado por reforço: Esse modo de aprendizagem permite um agente aprender a partir da interação com o ambiente no qual ele está inserido. Esta aprendizagem se dá por meio do conhecimento sobre o estado do indivíduo no ambiente, das ações realizadas neste e das mudanças de estado consequentes das ações. Resume-se esse modo no aprendizado do mapeamento de estados em ações da forma que um valor numérico de retorno seja maximizado.

3.3 PARADIGMAS DE APRENDIZADO

Segundo GOLDSCHMIDT (2010), um paradigma de aprendizado diz respeito ao modo com que o espaço de busca por um modelo de conhecimento, que represente os dados históricos disponíveis, deve ser percorrido. Os principais paradigmas de aprendizado sobre os quais os algoritmos de *Machine Learning* se baseiam, que podem ser usados tanto para descrição quanto para predição do conjunto de dados, dividam-se em:

- Simbólico: Compreende a construção de representações simbólicas de um conceito por meio de análise de exemplos e contraexemplos desse conceito. As representações simbólicas estão tipicamente representadas na forma de alguma expressão lógica, regras de produção, árvores de decisão ou rede semântica.
- Estatístico: Neste paradigma, métodos estatísticos, são utilizados para encontrar aproximações do modelo de conhecimento que esteja sendo induzido. Tais modelos, comumente assumem que os valores de atributos estão normalmente distribuídos, e então utilizam os dados fornecidos para determinar média, variância e co-variância da distribuição.
- Baseado em exemplos: Baseiam-se na busca de casos existentes similares ao novo exemplo a ser analisado para deduzir a saída do sistema. Esse tipo de sistema necessita manter os exemplos na memória para classificar os novos exemplos.
- Conexionista: Utiliza modelos matemáticos simplificados inspirados no modelo biológico do sistema nervoso, redes neurais, para tentar adquirir mapeamentos de novos exemplos nas saídas desejadas ou agrupamentos de exemplos equivalentes. A representação de uma rede neural envolve unidades altamente interconectadas e,

por esse motivo, o nome conexionista é utilizado para esse paradigma. Redes neurais artificiais serão abordadas com mais ênfase no próximo capítulo.

- Genético: Esse modelo de classificação consiste de uma população de elementos de classificação que competem para fazer a predição. Elementos que possuem um desempenho fraco são descartados, enquanto os elementos mais fortes são absorvidos, produzindo variações de si mesmo.

3.4 REDES NEURAIS ARTIFICIAIS

De acordo com REZENDE (2003), são modelos matemáticos que se assemelham às estruturas neurais biológicas e que têm capacidade computacional alcançada por meio de aprendizado e generalização. Para GOLDSCHMIDT (2010), redes neurais são modelos computacionais não lineares, inspirados na estrutura e no funcionamento do cérebro, que buscam reproduzir características humanas, tais como: aprendizado, generalização, abstração e associação. Pela sua estrutura, as redes neurais são bastante efetivas no aprendizado de padrões a partir de dados históricos não lineares, incompletos, com ruído e até compostos de exemplos contraditórios.

3.4.1 HISTÓRICO

As primeiras informações sobre neurocomputação surgiram na década de 40, em artigos do neurofisiologista Warren McCulloch. Seu primeiro trabalho sobre neurônios formais, apresentou um modelo matemático de simulação do processo biológico que ocorre em células nervosas vivas. O modelo buscava simular artificialmente o comportamento de um neurônio natural. Em tal modelo o neurônio artificial possuía apenas uma saída, produzida em função da soma dos valores de suas diversas entradas (GOLDSCHMIDT, 2010).

O motivo pelo qual máquinas inspiradas na biologia diferem das máquinas atuais é pelo fato de que as máquinas atuais baseiam seu processamento explicitamente em modelos algorítmicos, em que o comportamento diante de todas as situações possíveis deve ser programado previamente. Mecanismos de controle que possuem sua base em mecanismos neurais, contudo, não são baseados em modelos algorítmicos. Fazem utilização de cálculo matemáticos para efetuar suas operações e podem coordenar diversos graus de liberdade durante a execução de tarefas manipulativas e ambientes desestruturados, sendo capazes

de resolver tarefas complicadas sem desenvolver um algoritmo específico para cada problema.

3.4.2 DENIFIÇÃO

De acordo com HECHT-NIELSEN (1990), uma rede neural artificial pode ser definida como uma estrutura que processa informação de forma paralela e distribuída e que consiste em unidades computacionais, as quais podem possuir memória local e executar operações locais, interligadas por canais unidirecionais chamados conexões. Cada unidade possui uma única conexão de saída, que pode ser dividida em quantas conexões forem necessárias.

Nas redes neurais artificiais, a ideia é realizar o processamento de informações tendo como princípio a estrutura de neurônios do cérebro. Como o cérebro humano é apto capaz de aprender e tomar decisões com base em aprendizagem, as redes neurais artificiais devem fazer o mesmo. Dessa forma, uma rede neural pode ser interpretada como um esquema de processamento capaz de armazenar conhecimento baseado em aprendizagem e tornar disponível este conhecimento para a aplicação em questão.

A Tabela 1 sintetiza a conexão entre características e comportamentos de redes neurais artificiais com elementos da natureza.

Tabela 1: Conexões características e comportamentos de RNA com elementos da natureza

Modelo Natural	**Modelo Artificial**
Cérebro	RNA
Neurônio Biológico	Neurônio artificial / Elementos processadores
Rede de Neurônios	Estrutura em camadas
10 bilhões de Neurônios	Centenas / Milhares de neurônios
Aprendizado	Aprendizado
Generalização	Generalização
Associação	Associação
Reconhecimento de Padrões	Reconhecimento de Padrões

Na Figura 5 pode-se ver uma ilustração comparativa entre o modelo biológico e o artificial adotado pelas redes neurais artificiais.

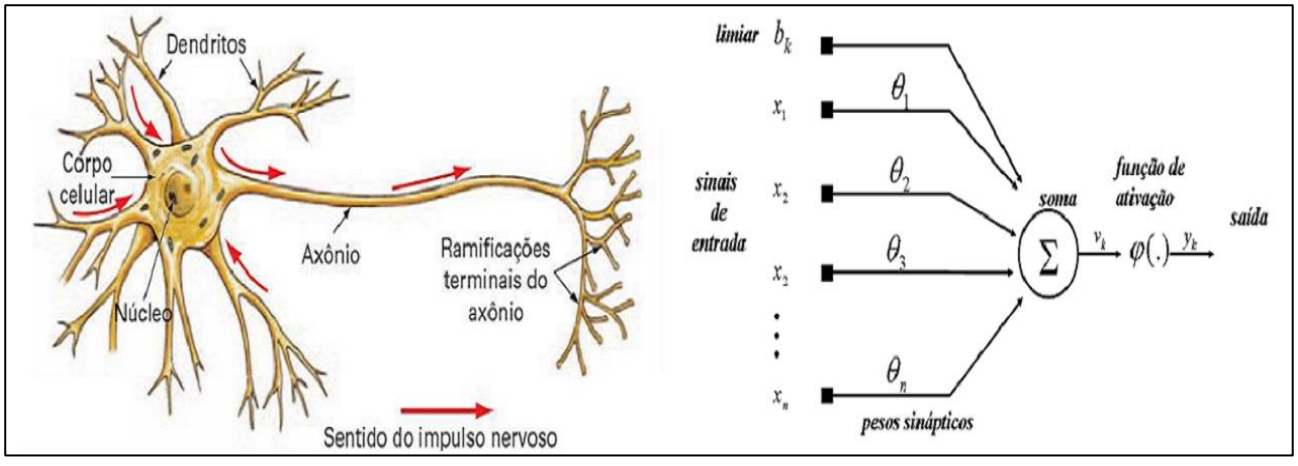

Figura 5: Comparação entre modelo biológico e artificial (GOLDSCHMIDT, 2010)

3.4.3 CARACTERÍSTICAS

Segundo GOLDSCHMIDT (2010), devido à similaridade com a estrutura do cérebro, as redes neurais artificiais possuem algumas características semelhantes às do comportamento humano, tais como:

- Busca paralela e endereçamento pelo conteúdo: Nas redes neurais o conhecimento fica distribuído pela estrutura das redes, de modo que a procura pela informação ocorra de forma paralela e não sequencial, analogamente com o cérebro que não possui endereço de memória.
- Aprendizado por experiência: As RNAs tentam aprender padrões a partir dos dados. Para isso, utilizam um processo de repetidas apresentações dos dados à rede que busca abstrair modelos de conhecimento de forma automática.
- Generalização: As redes neurais artificiais são capazes de generalizar seu conhecimento por meio de exemplos anteriores. A capacidade de generalização permite que as RNAs lidem com ruídos e distorções nos dados, respondendo positivamente a novos padrões. A Figura 6 mostra um exemplo de generalização.

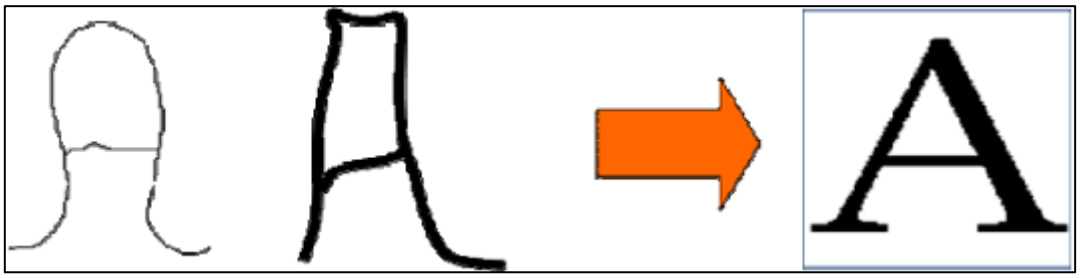

Figura 6: Exemplo de generalização (GOLDSCHMIDT, 2010)

- Associação: As RNAs são capazes de estabelecer relações entre padrões de naturezas distintas. Como por exemplo, identificar pessoas a partir de características da voz destas pessoas.
- Abstração: Abstração é a capacidade das redes neurais em identificar a essência de um grupo de dados de entrada. Isto é o que faz as redes neurais serem capazes de deduzir quais as características relevantes em um conjunto de entrada. É dessa forma que as RNAs extraem informações a partir de padrões ruidosos para padrões sem ruído. A Figura 7 ilustra uma exemplificação de abstração.

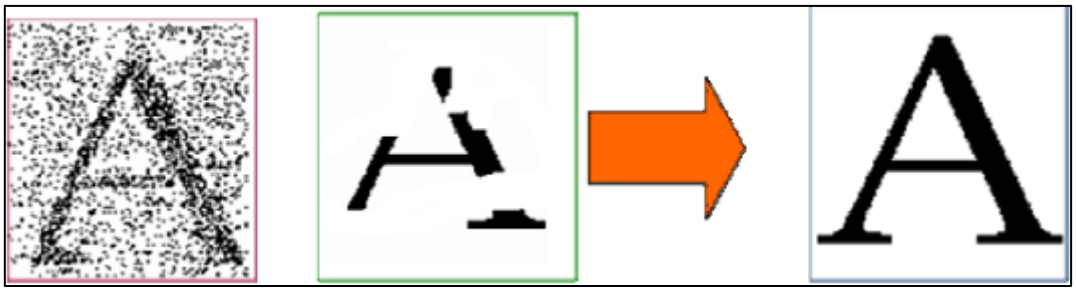

Figura 7: Exemplo de abstração (GOLDSCHMIDT, 2010)

- Robustez e Degradação gradual: Tendo em vista que a informação fica distribuída em uma rede neural artificial, a perda de um conjunto de neurônios artificiais não causa necessariamente o mau funcionamento da rede. O desempenho de uma RNA tem a tendência de diminuir gradativamente na medida em que a quantidade de neurônios artificiais inoperantes aumenta.
- Não programáveis: As redes neurais não necessitam de programação, elas são construídas. A rede deve ser modelada de acordo com as entradas e saídas envolvidas e um algoritmo de aprendizado, programado anteriormente é aplicado sobre o modelo e sobre os dados históricos, visando mapear corretamente as entradas da rede nas saídas correspondentes.
- Soluções aproximadas: As redes neurais nem sempre produzem a melhor solução para um problema, porém, geram soluções aproximadas e aceitáveis. Algumas estruturas de redes trabalham com o intuito de minimização de erros, mas nem sempre reduzido a zero.

Igualmente ao sistema biológico, uma rede neural artificial possui, de maneira simplificada, um sistema de neurônios e conexões ponderadas por pesos. Em uma rede neural os neurônios são arrumados em camadas, com ligações entre elas. A figura mostra a arquitetura de uma RNA simples, onde os círculos representam os neurônios e as linhas representam os pesos das ligações. A camada que recebe os dados é chamada camada de entrada e a camada que mostra o resultado é chamada de camada de saída. A camada interna, onde acontece o processamento interno é chamada de camada escondida. Dependendo da complexidade do problema, a rede neural pode ter uma ou várias camadas escondidas.

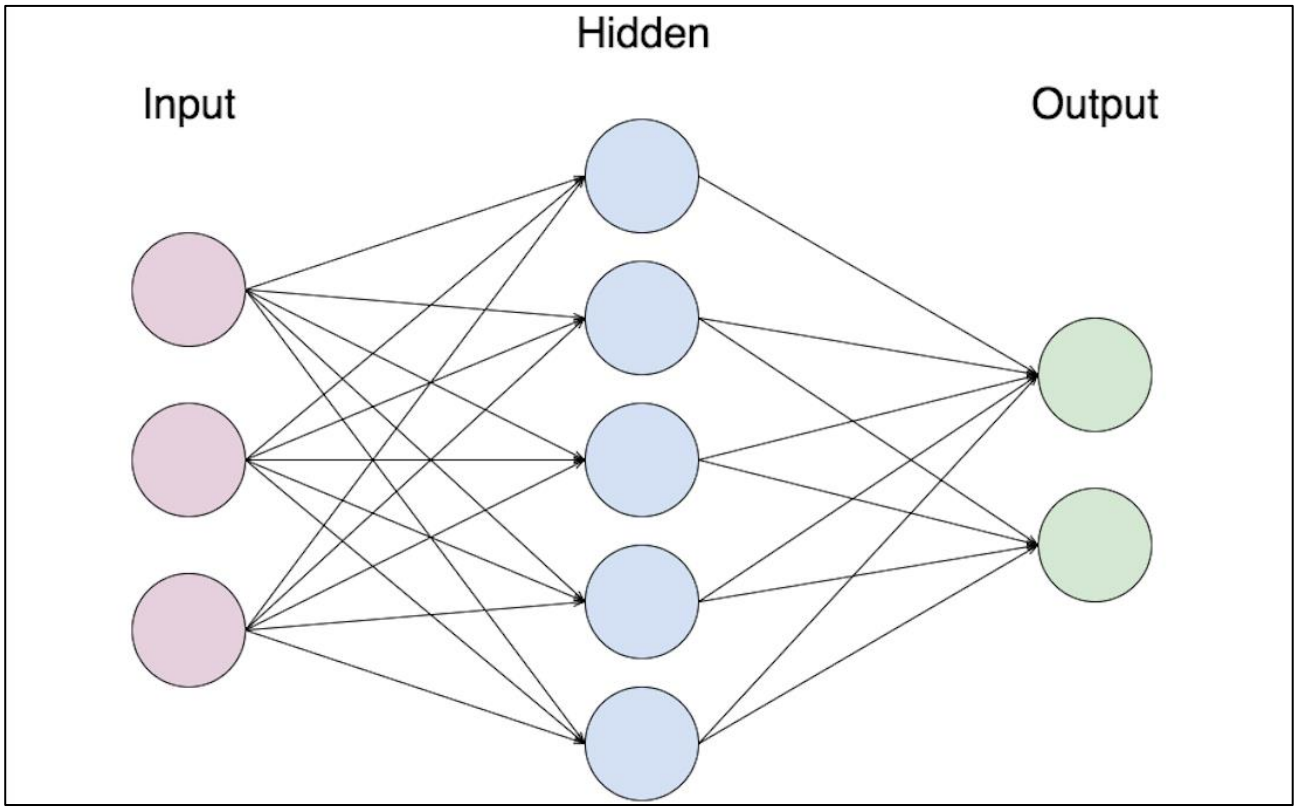

Figura 7: Arquitetura de uma RNA simples (GOLDSCHMIDT, 2010)

O processamento das redes neurais, em geral, ocorre da esquerda para a direita e, para fins computacionais os neurônios são rotulados com uma numeração sequencial de cima para baixo, da esquerda para direita.

4. INICIATIVAS OPEN SOURCE EM DATA SCIENCE

4.1 HISTÓRIA DA INICIATIVA OPEN SOURCE

Segundo SILVEIRA (2004), em 1983 Richard Stallman deu início ao projeto GNU, acrônimo recursivo de GNU *is not Unix*, ou seja, o projeto GNU tinha como objetivo produzir um sistema operacional livre que pudesse fazer o mesmo que o sistema Unix. Em 1983 Stallman explicou que GNU seria capaz de rodar programas do Unix, porém não seria idêntico ao Unix. A intenção era de ocorrer aperfeiçoamentos periódicos de acordo com a experiência com outros sistemas operacionais. A ideia de constituir um sistema operacional livre foi ganhando adeptos e se consolidou na formação da *Free Software Foundation*, em 1984.

No ano de 1991, Linus Torvalds, anunciou que havia desenvolvido o kernel para um sistema operacional do tipo Unix e disponibilizou de maneira livre. Com os esforços da comunidade de desenvolvedores em torno da *Free Software Foundation*¸ as primeiras versões do Software, Linux, já se mostravam mais flexíveis e robustas que o MS-DOS e o Windows.

Surgia assim uma alternativa ao software proprietário, o sistema operacional GNU/Linux. Diversos outros softwares livres vieram a ser criados, como o Apache, para rodar páginas web nos servidores de rede, o Gimp, para tratamento de desenhos e imagens, o OpenOffice, que contém editor de texto folha de cálculo e editor de apresentações, dentre outros. Hoje são milhares de softwares livres que utilizam a Licença Pública Geral (*General Public Licence*), criada pela *Free Software Foundation.*

Em 1998, Eric Raymmond e Linus Torvalds foram os protagonistas na criação da *Open Source Iniciative* (OSI), defendendo a adoção do software livre por razões técnicas e sugerindo o uso da expressão *Open Source* ao invés de *Free Software*. O principal motivo para o termo *Open Source* ser adotado foi introduzir o software livre no mundo dos negócios de uma forma mais aceitável para empresas mais conservadoras, evitando equívoco com o tempo *free* (OSI, 2012)*.*

4.2 FERRAMENTAS OPEN SOURCE NO CONTEXTO DE DATA SCIENCE

Em uma pesquisa feita por REINSTEIN (2018) foram listados os 20 maiores projetos *Open Source* de acordo com o número de contribuidores no GitHub, repositório de código fonte, conforme ilustra a Figura 9. O tamanho é proporcional ao número de contribuidores, a cor

representa a mudança no número de contribuidores em relação a essa mesma pesquisa feita no ano de 2017, vermelho significa que o número aumentou e azul que diminuiu. Os asteriscos representam os projetos sobre *Deep Learning.*

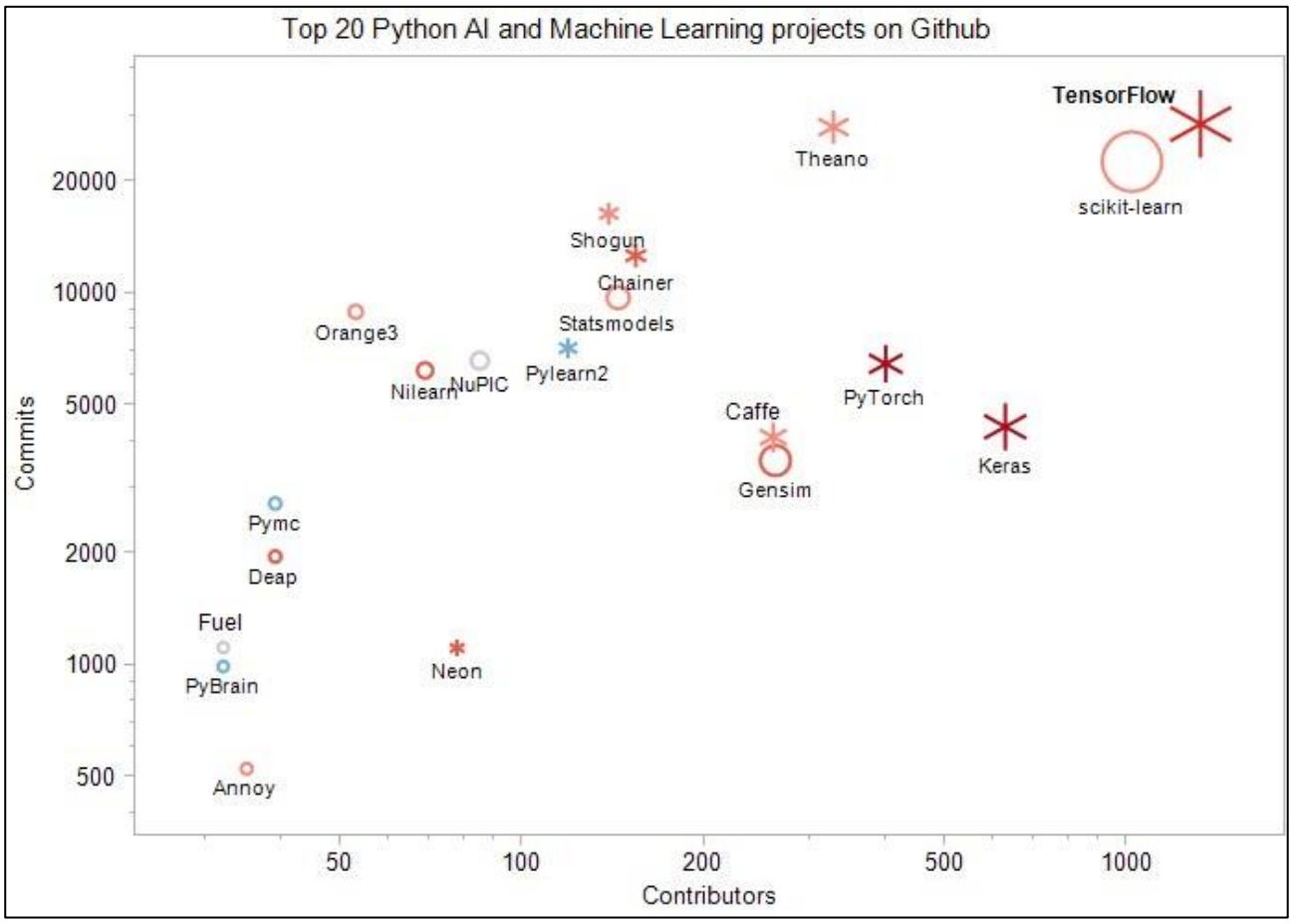

Figura 8: Maiores Projetos Machine Learning no Github (REINSTEIN, 2018)

TensorFlow, o primeiro lugar da pesquisa, foi originalmente desenvolvido por pesquisadores e engenheiros que trabalhavam no *Google Brain Team* dentro da Organização da Google de pesquisa sobre *Machine Learning.* O projeto foi criado com o intuito de facilitar a pesquisa sobre Aprendizado de Máquina, e tornar mais fácil a transição de um protótipo de pesquisa para um sistema de produção (TENSORFLOW, 2018).

Scikit-learn, o segundo lugar, é uma biblioteca simples e eficiente para mineração de dados e análise de dados, pode ser usada em vários contextos, construída em cima de Numpy, Scipy e Matplotlib (PEDREGOSA, 2011).

Keras, o terceiro lugar, uma API de redes neurais de alto nível, feita em Python e capaz de rodar em cima do TensorFlow, CNTK ou Theano. Keras foi desenvolvido com o foco em possibilitar a experimentação rápida. Sendo capaz de ir da ideia para o resultado com o menor tempo possível (KERAS, 2018).

4.3 TENSORFLOW

O TensorFlow é a segunda geração de um sistema projetado pelo Google Brain, equipe de pesquisa sobre *Deep Learning* do Google que teve seu início em 2010, as pesquisas da equipe do Google Brain giram em torno de aprendizagem de máquina, computação em escala e engenharia de sistemas. (GOOGLE BRAIN, 2018).

A primeira versão do TensorFlow foi lançada em 15 de fevereiro de 2015, podendo ser executado em dispositivos individuais, como também em múltiplas CPUs, do inglês *Central Processing Unit,* e GPUs, do inglês *Graphics Processing Unit*. Ele está disponível em versões de 64 bits Windows, Linux, MacOS, e plataformas de computação móveis. (TENSORFLOW, 2018)

TensorFlow reúne uma série de modelos e algoritmos de *Machine Learning* e *Deep Learning* e utiliza a linguem Python para fornecer uma API de front-end, enquanto os aplicativos são executados na linguagem C++. Por meio do TensorFlow os desenvolvedores criam gráficos de fluxo de dados, estruturas que representam como os dados em uma série de nós de processamento, sendo cada nó uma representação de uma operação matemática e cada conexão entre nós é um tensor multidimensional.

O TensorFlow além de toda a sua eficiência em *Deep Learning e Machine Learning,* ele possui algumas facilidades adicionais para os desenvolvedores, como o TensorBoard, um suíte de visualização que permite examinar e criar o perfil da forma como os gráficos são executados por meio de um painel interativo.

4.4 SCIKIT-LEARN

A biblioteca de aprendizado de máquina Scikit-Learn de código aberto, é baseada na linguagem de programação Python, e possui nativamente diversos algoritmos de *Machine*

Learning como, regressão, classificação, agrupamento, máquinas de vetores de suporte, k-means, florestas aleatórias, e foi projetada para relacionar-se com as bibliotecas Python numéricas científicas NumPy e SciPy. (PEDREGOSA, 2011).

O Scikit-Learn foi desenvolvido originalmente por David Cournapeau no Google Summer of Code, programa global com o objetivo de trazer estudantes desenvolvedores para o desenvolvimento de software de código aberto, onde os alunos trabalham com uma organização de código aberto em um projeto de programação de 3 meses durante as férias escolares. (GOOGLE SUMMER OF CODE, 2018).

O código base original foi posteriormente reescritos por outros desenvolvedores, e ainda continua em estado de desenvolvimento ativo, sendo patrocinado pela INRIA, organização francesa de caráter científico e tecnológico, Telecom ParisTech, escola de ensino superior pública localizada em Paris, e eventualmente pelo Google, por meio do Google Summer of Code.

4.5 KERAS

O framework de *Deep Learning* Keras, foi desenvolvido como parte de uma pesquisa do projeto ONEIROS, do inglês *Open-ended Eletronic Intelligent Robot Operating System,* e o seu primeiro e principal autor é François Chollet, um engenheiro do Google.

Escrita em Python, é capaz de rodar em cima do TensorFlow, Microsoft Cognitive Toolkit ou Theano, desenvolvida para ser capaz de proporcionar uma experimentação rápida com os algoritmos de *Deep Learning*, e prioriza a experiência do desenvolvedor. Sendo uma API projetada para humanos e não máquinas, Keras segue as melhores práticas para reduzir a carga cognitiva, isso o torna fácil de aprender e fácil de usar, aumentando a produtividade dos usuários e permitindo que isso seja um diferencial no mercado. (KERAS, 2018).

Além de contar com uma extensa comunidade, Keras possui também uma documentação bem estruturada e completa, fazendo com que seja a segunda maior biblioteca *Open Source* de *Deep Learning,* conta também com diversos dataset nativos da biblioteca, para a prática e experimentação dos algoritmos disponíveis pela plataforma.

4.6 PYTORCH

PyTorch é uma biblioteca *Open Source* de *Machine Learning* para a linguagem de programação Python, baseada no Torch, framework computação científica baseado na linguagem de programação Lua. PyTorch foi desenvolvida originalmente pelo grupo de pesquisa de Inteligência Artificial do Facebook. Desde o seu lançamento no início de janeiro de 2016, muitos pesquisadores o adotaram como uma biblioteca para leitura devido à sua facilidade de construir gráficos e até mesmo os de alta complexidade. (PYTORCH, 2018).

Segundo os criadores do PyTorch, eles possuem uma filosofia de ser imperativos. O que resulta nos cálculos que são executados imediatamente, não precisando esperar todo código ser escrito para saber se ele funciona corretamente, encaixando-se na metodologia de programação Python que funciona da mesma forma. Cada linha de código necessária para construir um gráfico define um componente do gráfico, os cálculos podem ser feitos de forma independente sobre os componentes, antes mesmo do gráfico ser construído completamente.

PyTorch é baseada em Python e foi criada com objetivo de fornecer flexibilidade como uma plataforma de *Deep Learning*. O fluxo de trabalho do PyTorch é o mais próximo dentre as bibliotecas de Machine Learning, da biblioteca de computação científica do Python, Numpy.

4.7 THEANO

Theano é uma biblioteca de computação numérica *Open Source* para Python, foi desenvolvida originalmente por um grupo de pesquisa de Aprendizagem de Máquina da Universidade de Montreal em 2007. Em Theano, os cálculos são expressos usando a sintaxe do NumPy e após compilados, podem ser executados com eficiência em arquiteturas de CPU ou GPU.

A biblioteca permite definir, otimizar e avaliar expressões matemáticas envolvendo matrizes multidimensionais de maneira eficiente, possui integração forte com NumPy, uso transparente de GPU, realizando cálculos com o uso intensivo de dados mais rápido do que em uma CPU, otimização de velocidade e estabilidade, geração dinâmica de código C, extenso teste unitário e auto verificação.

A linguagem de interface para Theano é Python, o que fornece uma prototipagem rápida e uma maneira fácil de usar e interagir com os dados, em contraponto a desvantagem do Python é seu intérprete, que em alguns casos é um mecanismo fraco para executar cálculos matemáticos em termos de uso de memória e velocidade, porém Theano supera essa limitação, explorando mecanismos de otimização. Por basear-se na biblioteca Numpy, Theano fornece um tipo de dados de matriz n-dimensional e diversas funções para indexar, remodelar e executar cálculos elementares em matrizes inteiras de uma só vez. Theano também foi projetado para facilitar e agilizar a extensibilidade através da definição expressões gráficas personalizadas escritas em C++, CUDA ou Python. (THEANO, 2018).

4.8 DISPOSITIVOS MÓVEIS E EMBARCADOS

Com a crescente dos dispositivos móveis e embarcados, com expectativa da existência de 50 bilhões de dispositivos conectados em 2020, sendo 27 bilhões conexões de máquina para máquina, representando um acréscimo de 15 trilhões de dólares ao PIB mundial nos próximos anos, de acordo com a DATAPREV (2017), isso ressalta a necessidade de manter tais dispositivos no centro dos investimentos do mercado de tecnologia. Devido à alta demanda, tem surgido diversos estudos para tornar os dispositivos inteligentes por meio da Aprendizagem de Máquina.

Atualmente, esses tipos de dispositivos funcionam principalmente como sensores que coletam e enviam dados para modelos de aprendizado de máquina executados na nuvem, todo o processamento requer muita computação e armazenamento, colocar todo esse hardware em um dispositivo embutido de baixo custo é o grande paradigma, segundo ERFANI (2017). Existe um grupo de pesquisadores da Microsoft, estudam a possibilidade de encolher e tornar o aprendizado da máquina muito mais eficiente para que realmente seja possível executá-los nos dispositivos.

O grupo está trabalhando com duas abordagens, uma de cima para baixo e outra de baixo para cima, no desafio de implantar modelos de Machine Learning em dispositivos de recursos limitados. A primeira abordagem baseia-se no desenvolvimento de algoritmos que comprimam modelos de aprendizado de máquina treinados para a nuvem que funcionem de forma eficiente em dispositivos como o Raspberry Pi 3, para isso estão fazendo uso da técnica chamada quantização de peso, que representa cada parâmetro de rede

neural com apenas alguns bits, em algumas vezes um único bit, ao invés do padrão de 32. A segunda abordagem, parte do princípio que não há como fazer uma rede neural profunda, mantê-la tão precisa quanto hoje e consumir bem menos recursos, dessa forma nessa abordagem os pesquisadores estão focados na construção de uma biblioteca com algoritmos de treinamento, cada um ajustado para alto desempenho em seu nicho. (MICROSOFT, 2017).

Um protótipo de dispositivo que está sendo desenvolvido para mostrar o potencial da pesquisa é uma bengala inteligente, que pode detectar quedas e fazer um pedido de assistência. Outro protótipo é o de uma luva inteligente capaz de interpretar a linguagem de sinais e reproduzir as palavras sinalizadas por meio de um alto falante.

5. PROPOSTA E DESENVOLVIMENTO DO TRABALHO

Neste capítulo será apresentada a proposta e o desenvolvimento do trabalho, enfatizando-se o desenvolvimento de algoritmos de *Machine Learning*, com o objetivo de explorar os principais e mais utilizados meios de aprendizado no contexto de *Data Science*, fazendo uso de ferramentas *Open Source.*

Serão utilizadas as seguintes ferramentas para o desenvolvimento dos algoritmos:

- Jupyter Notebook – Ambiente Integrado de Desenvolvimento (do inglês, Integrated Development Environment, IDE), uma aplicação web que permite criar e compartilhar documentos que contém código, equações, visualizações e textos.
- Numpy / Pandas – Utilizadas para a análise dos dados com mais performance e de maneira mais intuitiva.
- Matplotlib / Seaborn – Utilizada para a visualização dos dados.
- Scikit-Learn – Utilizada para o treinamento dos algoritmos.

5.1 BASES DE DADOS

A proposta será de utilizar três algoritmos distintos de IA, um para cada base de dados que será apresentada. As bases de dados a serem utilizadas serão:

- Base de dados de uma empresa de vendas online, a base possui as seguintes informações dos seus usuários: e-mail, endereço, avatar, tempo médio de seção, tempo no aplicativo, tempo no site, tempo de adesão, quantia anual gasta; totalizando 1000 registros, a estrutura de dados pode ser vista na Figura 9. Com a finalidade de saber qual é o melhor canal de vendas, aplicativo ou site, de acordo com os dados, será utilizado o algoritmo de regressão linear, algoritmo pertencente à aprendizagem supervisionada, se trata de uma função capaz de estimar um valor de uma variável, de acordo com os valores de outras variáveis. Uma empresa que possui uma loja online e realiza as suas vendas tanto por site, quanto por aplicativo, gostaria de saber onde focar seus esforços, na experiência do usuário no aplicativo, ou no site. Para a realização do trabalho, a empresa disponibilizou os dados com a estrutura de acordo com a Figura 9.

Email	Endereco	Avatar	Tempo medio secao	Tempo no App	Tempo no Site	Tempo de adesao	Quantia anual gasta

Figura 9: Estrutura de dados – Problema 1

- Base de dados dos passageiros do Titanic, navio que naufragou em 1912. A base possui as seguintes informações: identidade do passageiro, se sobreviveu ou não, classe, nome, sexo, idade, parentes a bordo, ticket, valor da passagem, cabine e em qual porto embarcou, de um total de 418 passageiros, a estrutura da base de dados pode ser vista na Figura 10. Para descobrir se determinado passageiro iria sobreviver ao naufrágio ou não, de acordo com os dados, será utilizado o algoritmo de regressão logística.

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Figura 10: Estrutura de dados - Problema 2

- Base de dados gerada com dados aleatórios, com o objetivo de agrupar esses dados aleatórios em conjuntos de dados menores, mas com alguma similaridade, será utilizado o algoritmo de K Means Clusterização.

5.2 ALGORITMOS

Para a exploração das bases de dados serão utilizados três algoritmos distintos de *Machine Learning,* que são eles:

- Regressão Linear: Algoritmo pertencente à aprendizagem supervisionada, se trata de uma função capaz de estimar um valor de uma variável, de acordo com os valores de outras variáveis. Estimar o tamanho da mão de uma pessoa, de acordo com a sua altura, por exemplo. É por meio da regressão linear que se obtém o resultado.
- Regressão Logística: Algoritmo pertencente à aprendizagem supervisionada, o algoritmo possui comportamento semelhante com o de regressão linear, porém no

modelo logístico, a saída é binária. Ou seja, apenas dois valores são possíveis, 0 e 1.

- K Means: Algoritmo pertencente à aprendizagem não supervisionada, ou seja, não se precisa informar rótulos de entrada, ele é capaz de realizar o seu propósito sem qualquer rotulação de dados, ele é um algoritmo de agrupamento que tem por objetivo dividir os dados em n grupos, onde cada dado fica pertencendo ao grupo mais próximo da média.

5.3 DESENVOLVIMENTO DOS ALGORITMOS

5.3.1 ALGORITMO DE REGRESSÃO LINEAR

O desenvolvimento deste algoritmo teve como objetivo solucionar o problema citado no capítulo 5, saber qual é o melhor canal de vendas de uma loja de vendas online, aplicativo ou site, de acordo com a base de dados.

Pode-se simplificar o desenvolvimento de algoritmos de *Machine Learning* em algumas etapas, a primeira é a importação das ferramentas a serem utilizadas, conforme Figura 11, que para o desenvolvimento deste trabalho serão as mesmas para todos os algoritmos.

```
In [2]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns
        %matplotlib inline
```

Figura 11: Importações

O próximo passo é o trabalho com os dados, primeiro tem-se a leitura dos dados, que pode ser feita por meio da biblioteca pandas com o comando pd.read_cvs (“base de dados_csv”), para dados no formato csv. Posteriormente, foram realizadas algumas análises mediante as bibliotecas de visualizações, Seaborn e Matplotlib para verificar relações entre os dados. Conforme a Figura 12, foi encontrado uma relação forte entre “Tempo de adesão” com a “Quantia anual gasta”, tal relação pode ser notada ao visualizar que a medida que o “Tempo de adesão” aumenta, a “Quantia anual gasta” também aumenta.

```
In [19]: sns.lmplot(x='Tempo de adesao',y='Quantia anual gasta',data=customers)
Out[19]: <seaborn.axisgrid.FacetGrid at 0x287f08969b0>
```

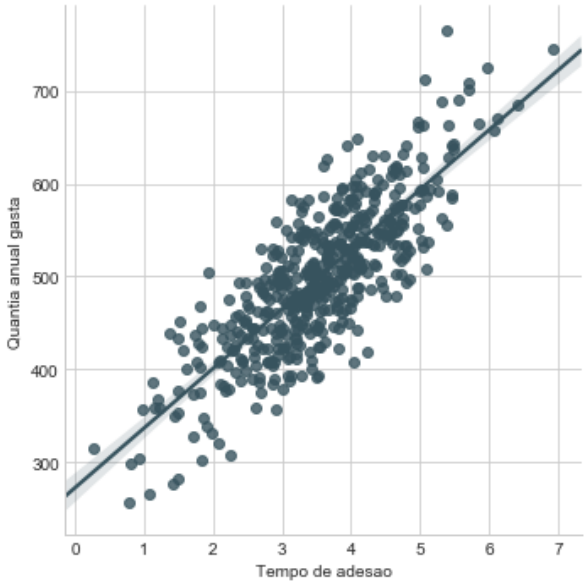

Figura 12: Tempo Adesão x Quantia Anual Gasta

Após análise dos dados, é necessário a realizar o treinamento do algoritmo. Nessa parte é definido qual é o dado que o algoritmo irá prever, e quais são os dados que ele tomará como base para realizar a previsão. No problema apresentado, levando em consideração que a empresa quer saber em qual investir para aumentar o seu lucro, os dados foram separados conforma a Figura 13, onde a “Quantia anual gasta” é o que será previsto pelo algoritmo e, “Tempo médio seção, Tempo no App, Tempo no Site, Tempo de adesão” são os dados que o algoritmo tomará como base para realizar as previsões.

```
In [20]: y = customers['Quantia anual gasta']

In [21]: X = customers[['Tempo medio secao', 'Tempo no App','Tempo no Site', 'Tempo de adesao']]
```

Figura 13: X e Y - Problema 1

Posteriormente, foi-se necessário utilizar a biblioteca do Scikit-Learn para treinar a rede, conforme a Figura 14, ao treinar a rede, foi separado 70% dos dados para treinar e o restante para testar a validar o algoritmo.

```
In [15]: from sklearn.model_selection import train_test_split

In [16]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=101)

In [18]: from sklearn.linear_model import LinearRegression

In [19]: lm = LinearRegression()

In [20]: lm.fit(X_train,y_train)
Out[20]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)
```

Figura 14: Treino - Regressão Linear

Com a rede treinada, foi feito uso da biblioteca Matplotlib para plotar um gráfico relacionando os valores das predições do algoritmo, com os reais valores, como pode ser visto na Figura 15.

```
In [23]: predictions = lm.predict( X_test)

In [24]: plt.scatter(y_test,predictions)
         plt.xlabel('Y Test')
         plt.ylabel('Predicted Y')
Out[24]: Text(0,0.5,'Predicted Y')
```

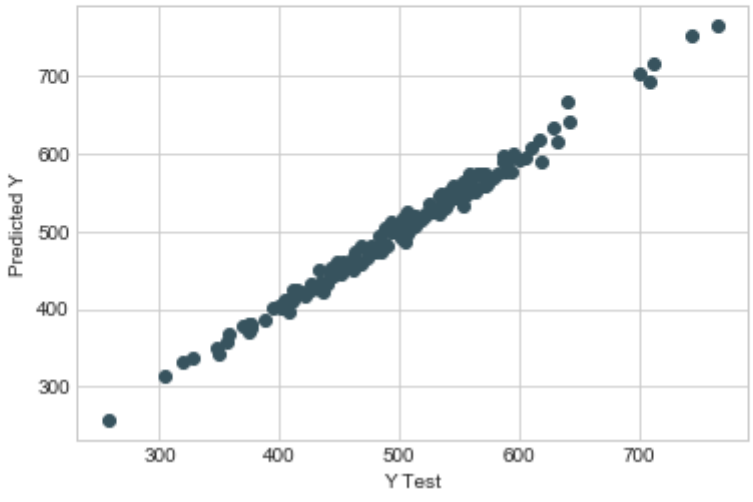

Figura 15: Predições - Regressão Linear

Para a conclusão do problema, foi necessário um passo adicional, encontrar os coeficientes dos dados com relação à quantia gasta. O resultado final pode ser observado na Figura 16.

Com o coeficiente, pode-se concluir que, mantendo todas as outras variáveis e aumentando o tempo no site em uma unidade, a quantidade gasta no site aumentará em 0,19. Ao realizar o mesmo cálculo com o tempo no App, o resultado será de 38,59 a mais nos gastos dos usuários. Dessa forma a empresa, deverá se empenhar mais em melhorar a experiência do usuário no aplicativo.

```
In [25]: coeficientes = pd.DataFrame(lm.coef_,X.columns)
         coeficientes.columns = ['Coeficiente']
         coeficientes
```

Out[25]:

	Coeficiente
Tempo medio secao	25.981550
Tempo no App	38.590159
Tempo no Site	0.190405
Tempo de adesao	61.279097

Figura 16: Coeficientes

5.3.2 ALGORITMO DE REGRESSÃO LOGÍSTICA

O desenvolvimento deste algoritmo teve como objetivo solucionar o problema citado no capítulo 5. Onde o algoritmo seria capaz de predizer se determinado passageiro sobreviveria ou não, ao naufrágio, caso estivesse a bordo do navio.

Para o desenvolvimento deste algoritmo foi utilizado as ferramentas do algoritmo de regressão linear, dessa forma as importações foram as mesmas, de acordo com a Figura 12.

Posteriormente foi realizado algumas análises dos dados para um melhor entendimento do caso, por meio das análises foi possível concluir que a maioria dos passageiros que

sobreviveram ao naufrágio eram mulheres da primeira classe, e o grupo de passageiros que mais obtiveram mortes foi o de homens, como observa-se nas Figuras 18 e 19.

```
In [5]: sns.set_style('whitegrid')
        sns.countplot(x='Survived',hue='Sex',data=train,palette='RdBu_r')

Out[5]: <matplotlib.axes._subplots.AxesSubplot at 0x1ee0cfd7c88>
```

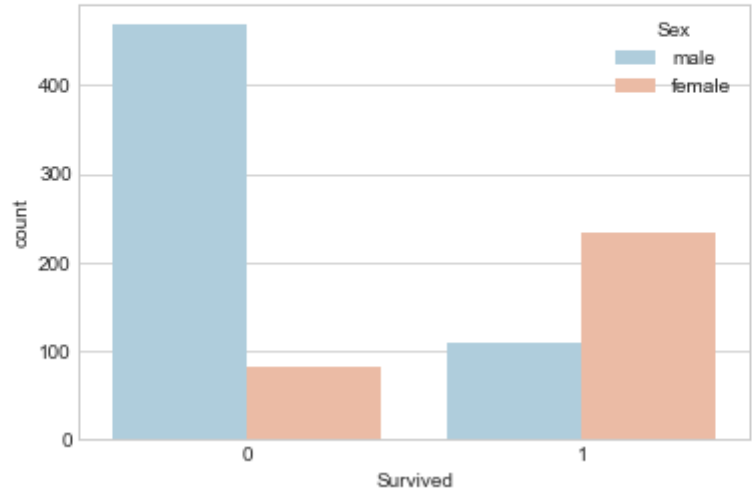

Figura 17: Sobreviventes por sexo

```
In [6]: sns.set_style('whitegrid')
        sns.countplot(x='Survived',hue='Pclass',data=train,palette='rainbow')

Out[6]: <matplotlib.axes._subplots.AxesSubplot at 0x1ee0d038588>
```

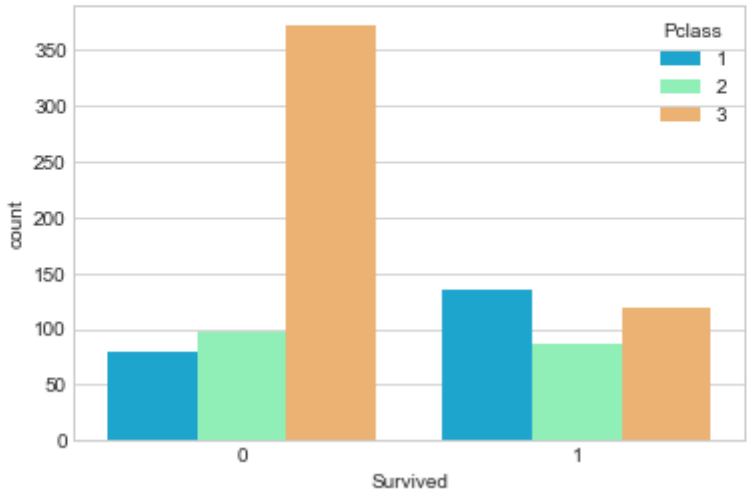

Figura 18: Sobreviventes por classe

Ao terminar a análise dos dados, foi necessário realizar a limpeza, tirando dados irrelevantes para o algoritmo, e transformando dados relevantes não numéricos, como sexo, em dados numéricos, adiante foi efetuado o treinamento da rede e, assim como no algoritmo de regressão linear, também foram separados 70% dos dados para o treinamento e o restante para a validação do algoritmo, conforme a Figura 19.

```
In [17]: from sklearn.model_selection import train_test_split

In [18]: X_train, X_test, y_train, y_test = train_test_split(train.drop('Survived',axis=1),
                                                    train['Survived'], test_size=0.30,
                                                    random_state=101)

In [20]: from sklearn.linear_model import LogisticRegression

In [21]: logmodel = LogisticRegression()
         logmodel.fit(X_train,y_train)

Out[21]: LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, max_iter=100, multi_class='ovr', n_jobs=1,
                   penalty='l2', random_state=None, solver='liblinear', tol=0.0001,
                   verbose=0, warm_start=False)

In [23]: predictions = logmodel.predict(X_test)
```

Figura 19: Treino - Regressão Logística

Com a rede treinada, partiu-se para a validação da rede, realizando predições e comparando com o real resultado, como pode ser visto na Figura 20. Percebe-se que o algoritmo teve um total de 82% de acertos, ou seja, dado um tripulante aleatório o algoritmo de regressão lógica treinado, será capaz de precisar corretamente 82% dos casos.

```
In [24]: from sklearn.metrics import classification_report

In [25]: print(classification_report(y_test,predictions))
                 precision    recall  f1-score   support

              0       0.81      0.93      0.86       163
              1       0.85      0.65      0.74       104

    avg / total       0.82      0.82      0.81       267
```

Figura 20: Resultado - Regressão Logística

5.3.3 ALGORITMO DE K MEANS CLUSTERIZAÇÃO

O desenvolvimento deste algoritmo teve como objetivo solucionar o problema citado no capítulo 5. Com a estrutura de dados que será gerada no desenvolvimento do algoritmo, realizar a Clusterização de forma a dividir os dados em conjuntos de dados menores com certa similaridade.

Para o desenvolvimento deste algoritmo foi utilizado as ferramentas do algoritmo de regressão linear, dessa forma as importações foram as mesmas, de acordo com a Figura 12.

Os dados foram gerados utilizando a função make-blobs da biblioteca Scikit-learn, que possui como principais parâmetros, n_sample – número de pontos divididos igualmente entre os clusters, n_features – número de características de cada amostra, centers – número de clusters, conforme a Figura 21.

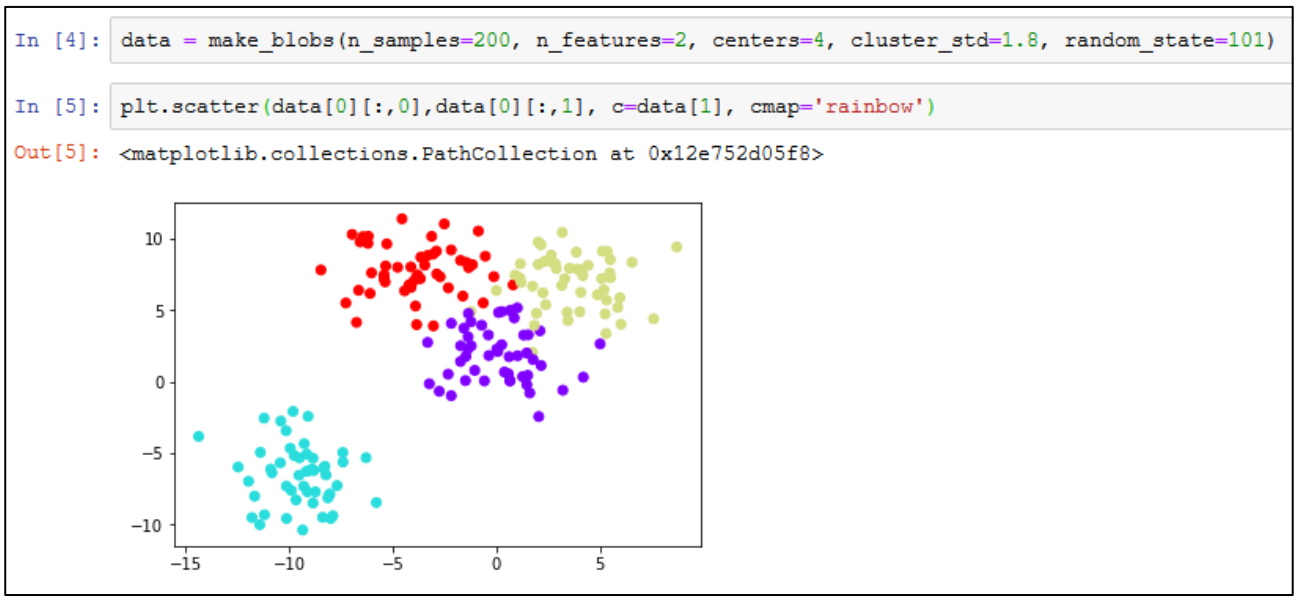

Figura 21: Dados gerados

Foi utilizado posteriormente o Scikit-Learn também para treinar a rede. Para realizar o treinamento da rede utilizando o algoritmo de K Means é necessário passar como parâmetro a quantidade de clusters, como os dados gerados foram utilizados 4 clusters, o

treinamento do algoritmo também foi baseado em 4 grupos, como pode ser visto na Figura 22.

```
In [6]: from sklearn.cluster import KMeans

In [7]: kmeans = KMeans(n_clusters=4)

In [8]: kmeans.fit(data[0])

Out[8]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
            n_clusters=4, n_init=10, n_jobs=1, precompute_distances='auto',
            random_state=None, tol=0.0001, verbose=0)
```

Figura 22: Treino - K Means Clusterização

Após o treinamento, foi realizado uma comparação com a separação que o algoritmo realizou com a separação real que foi gerada. Ao se tratar de um problema fictício em que os dados foram gerados e a resposta é de conhecimento do desenvolvedor, é possível fazer tal comparação, porém em problemas reais, não é o natural ter esses dados para comparação, uma vez que se trata de um algoritmo não-supervisionado, ou seja, não possui rótulos.

O resultado final da comparação pode ser visto na Figura 23.

```
In [9]: fig , (ax1, ax2) = plt.subplots(1,2, sharey=True, figsize=(10,6))

        ax1.set_title('K Means')
        ax1.scatter(data[0][:,0], data[0][:,1], c=kmeans.labels_, cmap='rainbow')

        ax2.set_title('Original')
        ax2.scatter(data[0][:,0], data[0][:,1], c=data[1], cmap='rainbow')

Out[9]: <matplotlib.collections.PathCollection at 0x12e76bdaa90>
```

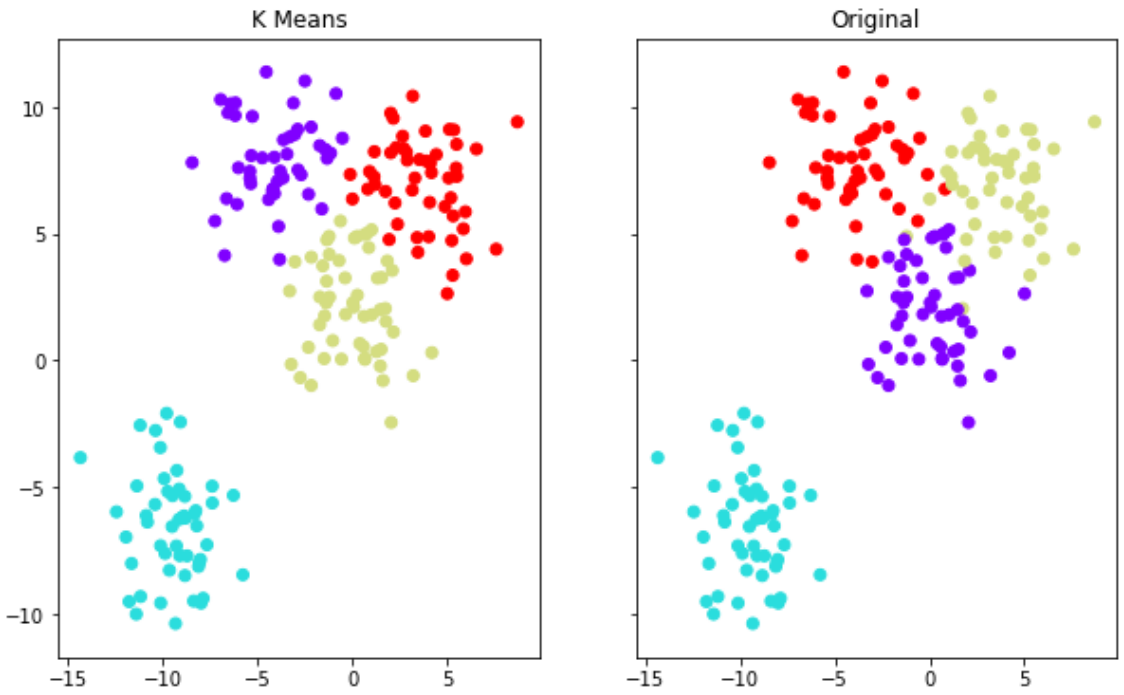

Figura 23: Resultado - K Means

Nota-se que o algoritmo se saiu muito bem ao comparar o resultado obtido com a separação original. Pode-se visualizar algumas divergências em dados que se encontram na extremidade de cada grupo e próximo ao grupo vizinho, porém os dados situados em posições que não sejam essas, foram todos classificados corretamente.

6. CONCLUSÃO

Observa-se uma multidisciplinaridade no âmbito de *Data Science*, sendo habilidade de programação, conhecimentos em matemática e estatística, e conhecimento de negócio, os mais abordados na literatura. Dentre as disciplinas, Aprendizado de Máquina – junção entre habilidades de programação e conhecimentos em matemática e estatística – está sendo trabalhada com mais ênfase.

O Aprendizado de Máquina, uma das ramificações da Inteligência Artificial, trabalha em reconhecimento de padrões, seja por meio de suas redes neurais, que tem por objetivo agir de forma semelhante às redes neurais biológicas, ou pelos diferentes modos de aprendizados, sendo eles: Aprendizado supervisionado, não supervisionado ou por reforço.

Com o desenvolvimento da parte teórica deste trabalho pude adquirir um maior conhecimento dos conceitos explanados, principalmente na área de *Data Science*, que por ser um assunto que engloba diversas outras disciplinas, foi o mais complexo. Com o desenvolvimento da parte prática, espero adquirir maior conhecimento nas ferramentas *Open Source* sobre *Machine Learning* e os seus diversos algoritmos de aprendizado.

6.1 TRABALHOS FUTUROS

Várias extensões podem ser exploradas relacionadas ao trabalho realizado. Uma das possíveis extensões é explorar de forma prática outras tecnologias de Aprendizado de Máquina Open Source. Seria interessante, por exemplo, implementar algoritmos de *Deep Learning* utilizando as principais ferramentas como, TensorFlow e Keras.

Um outro trabalho interessante é comparar o desempenho de algoritmos com o mesmo propósito, porém com bibliotecas livres diferentes, sendo capaz de explanar os prós e contras das bibliotecas trabalhadas. Também seria interessante realizar um trabalho que incorpore todas as fases da Ciência de Dados, desde a coleta de dados até o treinamento dos algoritmos.

REFERÊNCIAS

BARANAUSKAS, J. A; MONARD, M. C. **Sistemas inteligentes: Conceitos Sobre Aprendizado de Máquina**. Cap 4. Ed. Manole, 2006.

BRINK, H.; RICHARDS, J. **Real World Machine Learning**. [S.1.]: Manning Publications C.O, 2014.

CINTRA, M. **Os custos dos congestionamentos na cidade de São Paulo.** 2014. 38 p. Disponível em: <http://biblioteca.fgv.br/dspace/bitstream/handle/10438/11576/TD%20356%20-%20Marcos%20Cintra.pdf?sequence=1>. Acesso em: 20 nov. 2017.

CONWAY, D. 2010. **The Data Science Venn Diagram.** Disponível em: <http:drewconway.com/zia/2013/3/26/the-data-science-venn-diagram>. Acesso em: 22 nov. 2017.

DAMA INTERNACIONAL. **The DAMA Guide to the Data Management Body of Knowledge.** First edition ed. Bradley Beach, N.J.: Technics Publications, LLC, 2009.

DAVENPORT, T. H.; COHEN, D.; JACOBSON, A. **Competing on Analytics** In: Harvard Business Review, p.98-107. 2006.

DHAR, V. **Data Science and prediction**. Communications of the ACM, v.56, n.12, 2013.

DINIZ, Carlos Alberto; LOUZADA NETO, Francisco. **Data Mining: uma introdução.** São Paulo: ABE, 2000.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P.; **From Data Mining to Knowledge Discovery: An Overview**. AAAI The MIT Press, England. 1996.

FREIRE, J. et al. **Reproducibility using vistrails**. In: Implementing Reproducible Research. 2014.

GOLDSCHMIDT, R. R. **Uma Introdução à Inteligência Computacional: fundamentos, ferramentas e aplicações**. Rio de Janeiro: IST-Rio, 2010.

GONÇALVES, C. R.; CERVANTES, B. M. **Data Science: Ciência orientada a dados**. Inf. Londrina, v.21, n.2, 2016.

GOOGLE BRAIN. **About Google Brain.** Disponível em: <https://ai.google/research/teams/brain>. Acesso em: jun, 2018.

GOOGLE SUMMER OF CODE. **About Google Summer of Code**. Disponível em: <https://summerofcode.withgoogle.com/about/>. Acesso em: jun, 2018.

HADDAH, E. A.; VIEIRA, R. S. **Mobilidade, acessibildade e produtividade: nota sobre a valoração econômica do tempo de viagem na região metropolitana de São Paulo.** 2015. 26 p. Disponível em: <http://www.usp.br/nereus/wp-content/uploads/TD_Nereus_08_2015.pdf>. Acesso em: 20 nov. 2017.

HAN, J.; KAMBER, M.; PEI, J. **Data Mining: Concepts and Techniques**, Third edition (the morgan kaufmann series in data management systems). Morgan Kaufmann, 2011.

HECHT-NIELSEN, R. **Applications of Conterpropagation Networks, Neural Networks**, v1. Neurocomputing, New York, Addison-Wesley, 1990.

HEINZLE, R.; GAUTHIER, F. A. O.; FIALHO, F. A. P. **Semântica nos Sistemas de Apoio à decisão: O Estado da Arte**. Revista da Unifebe. Disponível em: <http://periodicos.unifebe.edu.br/index.php/revistaeletronicaunifebe/article/view/551>. Acesso em: jan, 2018.

HEY, T.; TANSLEY, S.; TOLLE, K. (EDS.). **The Fourth Paradigm: Data-Intensive Scientific Discovery.** 1 edition ed. Redmond, Wash.: Microsoft Research, 2009.

KERAS. **Keras: The Python Deep Learning Library.** Disponível em: <https://keras.io>. Acesso em: fev, 2018.

KOBSA, A.; KOENEMAN, J.; POHL, W. **Personalised hypermedia presentation techniques for improving online customer relationships. The Knowledge Engineering Review,** 2001.

LORENA, A, C.; CARVALHO, A. C. P. L. F. **Uma introdução às Support Vector Machines**. Revista de Informática Teórica e Aplicada, vol 14, no2, 2007. Disponível em: <http://seer.ufrgs.br/index.php/rita/article/viewPDFInterstitial/rita_v14_n2_p43-67/3543>. Acesso em: jan, 2018

LOUKIDES, M.; **What Is Data Science?** 2010 O'Reilly, edição digital disponível em <https://www.amazon.com/What-Data-Science-Mike-Loukidesebook/dp/B007R8BHAK/ref=sr_1_1?s=books&ie=UTF8&qid=1399984583&sr=1-1&keywords=data+scientist>. Acesso em Fevereiro, 2018.

MADDEN, S. **From Databases to Big Data**. Internet Computing IEEE, v.16, n.3. 2012

MAYER-SCHONB, V.; CUKIER, K. **Big Data: A Revolution That Will transform How We Live, Work, and Think. 2014.** Ed. Eamon Dolan/Mariner Books, 2014.

MCKINSEY & COMPANY. **Where Machines Could Replace Humans – and Where They Can't (Yet).** 2016.

MEIRELES, I. **Visualizing data: new pedagogical challenges.** Spinillo, Farias e Padovani Ed. São Paulo: SBDI. Brazilian Society of Information Design, 2010.

MICROSOFT. **Grande Salto da IA para pequenos dispositivos abre um mundo de possibilidades**. Disponível em: <https://news.microsoft.com/pt-br/salto-da-ia-para-pequenos-dispositivos-abre-um-mundo-de-possibilidades/>. Acesso em: jun, 2018.

MURRAY, S. **Interactive Data Visualization for the Web**. Sebastopol: O'Reilly Media, 2013.

OSI, Open Source Iniciative. **History of The OSI**. 2012. Disponível em <https://opensource.org/history>. Acesso em fev/2018.

PEDREGOSA, F ET AL. **Scikit-Learn: Machine Learning in Python**, JMLR 12, 2011.

PORTO, F.; ZIVIANI, A. **Ciência de Dados**. Laboratório Nacional de Computação Científica (LNCC), 2014.

PYTORCH. **About PyTorch.** Disponível em: <https://pytorch.org/about/>. Acesso em: jun, 2018.

REINSTEIN, I. **Top 20 Python AI and Machine Learning Open Source Projects**. Disponível em: https://www.kdnuggets.com/2018/02/top-20-python-ai-machine-learning-open-source-projects.html. Acesso em: fev, 2018.

REZENDE, S. **Sistemas Inteligentes: Fundamentos e Aplicações.** Barueri: Manole, 2003.

SEGEL, E; HEER, J. **Narrative visualization: Tellingstories with data.** Visualization and Computer Graphics, IEEE Transactions 2010.

SHALEV-SHWARTZ, S.; BEN-DAVID, S. **Understanding Machine Learning: From Theory to Algorithms**. [S.1.]: Cambridge University Press, 2014.

SILVA, V.A. **Determinação da estrutura organizacional das vias MAP KINASES em sorgo, Arabidopsis lyrata e cana-de-açucar por meio de análise de Bioinformática.** Tese de Doutorado. UENF Darcy Ribeiro, Agosto 2010.

SMOLA, A.; VISHWANATHAN, S. **Introduction to machine learning.** Cambridge University, UK, v. 32, 2008.

SILVEIRA, SÉRGIO AMADEU. **Software Livre: a luta pela Liberdade do conhecimento** – São Paulo: Editora Fundação Perseu Abramo, 2004.

SMITH, I. **The internet of Things 2012: New Horinzons.** CASAGRAS, 2012.

STANTON, J. **An introduction data science.** Syracuse University School of Information Studies, 2012.

STRASSER, C. et al. **Primer on Data Management: What you always wanted to know**. 2012.

TENSORFLOW. **An open-source machine learning framework for everyone**. Disponível em: https://www.tensorflow.org. Acesso em: fev, 2018.

THEANO. **Theano: A Python framework for fast computation of mathematical expressions**. Disponível em: <https://arxiv.org/pdf/1605.02688.pdf>. Acesso em: jun, 2018.

TIERNEY, B. **Data Science is multidisciplinary.** 2016. Disponível em: <http://migre.me/vlsaS.> Acesso em: fev, 2018.

TWEEDALE, J. W.; PHILLIPS-WREN, G.; JIAN, L. C. **Advances in Intelligent Decision-Making Technology Support**. Cham: Springer International Publishing 2016

VIÉGAS, F. **Designer explica como a visualização de dados pode ser atraente.** Rio de Janeiro, TV Globo, 2013. Disponível em: htpp://goo.gl/c35Ztn. Acesso em fev, 2018.

WANG, D.; JEFFREY F. HARPER; GRIBSKOV M. **Systematic Trans-Genomic Comparison of Protein Kinases between Arabidopsis and Saccharomyces cerevisiae.** Plant Physiology, Vol. 132, Agosto 2003.

ZAKI, M. J.; MEIRA, W. J. **Data mining and Analysis: Fundamental Concepts and Algorithms**. 1.ed. Cambridge University Press, 2014

ZHOU, Z. **Three perspectives of data mining**. Artif. Intell. Elsevier Science Publishers Ltd., Essex, UK, v. 143, n.1, jan. 2003.

ZHU, Y. Y; XIONG, Y. **Dataology and Data Science: Up to now.** Fundan University Press, 2011.