

**CENTRO UNIVERSITÁRIO ALVES FARIA (UNIALFA)**  
**GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO**

**Gabriel Gonçalves Medeiros**

**INDICATIVO DE TRANSTORNO DO ESPECTRO AUTISTA COM *MACHINE***  
***LEARNING***

**GOIÂNIA**  
**JUNHO DE 2021**

**CENTRO UNIVERSITÁRIO ALVES FARIA (UNIALFA)**  
**GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO**

**Gabriel Gonçalves Medeiros**

**INDICATIVO DE TRANSTORNO DO ESPECTRO AUTISTA COM *MACHINE***  
***LEARNIG***

Trabalho de conclusão de curso apresentado  
para conclusão do Curso de Engenharia da  
Computação do Centro Universitário Alves  
Faria, sob a orientação do Prof.<sup>a</sup> Ma. Fabiana  
Rocha de Andrade e Silva.

**Orientador (a):**

**Fabiana Rocha de Andrade e Silva**

**GOIÂNIA**  
**JUNHO DE 2021**

**CENTRO UNIVERSITÁRIO ALVES FARIA (UNIALFA)**  
**GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO**

**Gabriel Gonçalves Medeiros**

**INDICATIVO DE TRANSTORNO DO ESPECTRO AUTISTA COM *MACHINE***  
***LEARNIG***

**AVALIADORES:**

---

Prof.<sup>a</sup> Ma. Fabiana Rocha de Andrade e Silva – UNIALFA  
(Orientadora)

---

Prof. Me. Geraldo Valeriano Ribeiro – UNIALFA

---

Prof.<sup>a</sup> Esp. Nayane Moura Neris – UNIALFA

**GOIÂNIA**  
**JUNHO DE 2021**

*Aos meus pais e familiares, por sempre me apoiarem.*

*Aos meus amigos por toda essa jornada que passamos juntos.*

*Aos meus professores Fabiana, Thales, Evelyn e Richard.*

## RESUMO

MEDEIROS, Gabriel Gonçalves. **Indicativo do Transtorno do espectro autista com *machine learning***. Trabalho de conclusão de curso. 39 f. - Graduação em Engenharia da computação. Goiânia, 2021.

Este trabalho tem por finalidade apresentar um indicativo para Transtorno do Espectro Autista. Será aplicado algoritmos de *machine learning* a dados coletados de pessoas diagnosticadas com Transtorno do Espectro Autista e pessoas que não possuem, e assim treinar a rede neural para classificar novos dados em um provável diagnóstico de TEA. Após treinar a rede neural com os dados de pessoas com diagnóstico positivo e negativo e obter uma acurácia confiável é possível aplicar testes para provar a eficiência de classificação do modelo classificatório concluído, testes esses que são uma parte da base de dados disponível que serão submetidos ao modelo e após a predição comparados com o resultado real.

**PALAVRAS-CHAVE:** *Machine Learning*. Transtorno do Espectro Autista. Diagnóstico.

## **ABSTRACT**

MEDEIROS, Gabriel Gonçalves. **Indicative of Autism Spectrum Disorder with machine learning**. Undergraduate Final Project. 39 f - Graduation in Computer Engineering. Goiânia, 2021.

This work proposes to present an indicative for Autism Spectrum Disorder. Machine learning algorithms will be applied to data collected from people diagnosed with Autism Spectrum Disorder and people who do not have it, and this way train the neural network to classify new data in a probable ASD diagnosis. After training the neural network with the data of people with positive and negative diagnosis and obtaining a reliable accuracy, it is possible to apply tests to prove the classification efficiency of the classification model, tests that are part of the available database that will be submitted to the model and after the prediction compared to the actual result.

**KEYWORDS:** *Machine Learning*. Autism Spectrum Disorder. Diagnosis.

## LISTAS

### LISTA DE FIGURAS

Figura 1 – Quantidade de casos de TEA por nascimento.....	16
Figura 2 – Exemplo de uma árvore de decisão.....	18
Figura 3 – Representação de um neurônio biológico.....	19
Figura 4 – Representação de um neurônio artificial.....	20
Figura 5 – Bibliotecas e <i>Frameworks</i> importados.....	27
Figura 6 – Algumas funções de normalização.....	28
Figura 7 – Exemplo método <code>get_dummies</code> .....	29
Figura 8 – Separação dos dados de teste.....	30
Figura 9 – Criação da Rede Neural.....	31
Figura 10 – Representação da Rede Neural.....	31
Figura 11 – Configuração para gerar gráficos de accuracy e loss.....	33
Figura 12 – Histórico de accuracy e loss.....	34
Figura 13 – Obtendo previsões com os dados de teste.....	35
Figura 14 – Resultado da matriz de confusão.....	35

## LISTA DE QUADROS

Quadro 1 – Relação entre elementos da base de dados com seu tipo e descrição.....	23
---	----



## **LISTA DE SIGLAS**

APA - *American Psychological Association*

CDC - *Centers for Disease Control and Prevention*

DSM - *Diagnostic and Statistical Manual*

EUA - Estados Unidos da América

TEA - Transtorno do Espectro Autista

ML - *Machine Learning*

## SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>10</b>
<b>2 FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>12</b>
<b>2.1 O Transtorno do Espectro Autista.....</b>	<b>12</b>
<b>2.1.1 Histórico do Autismo .....</b>	<b>12</b>
<b>2.1.2 O que é o TEA? .....</b>	<b>13</b>
<b>2.1.3 Sintomas .....</b>	<b>13</b>
<b>2.1.4 Como é diagnosticado?.....</b>	<b>14</b>
<b>2.1.5 Tratamentos .....</b>	<b>16</b>
<b>2.2 Algoritmos de Machine Learning.....</b>	<b>16</b>
<b>2.2.1 Aprendizagem por Árvore de decisão.....</b>	<b>18</b>
<b>2.2.2 Redes Neurais Artificiais .....</b>	<b>19</b>
<b>2.2.3 Aprendizagem Estatística.....</b>	<b>21</b>
<b>3 METODOLOGIA.....</b>	<b>22</b>
<b>4 DISCUSSÃO, ANÁLISE E AVALIAÇÃO DOS DADOS COLETADOS.....</b>	<b>26</b>
<b>4.1 Normalizando os dados.....</b>	<b>26</b>
<b>4.2 Manipulação dos dados com bibliotecas.....</b>	<b>28</b>
<b>4.3 Divisão dos dados para teste.....</b>	<b>29</b>
<b>4.4 Construção da Rede Neural.....</b>	<b>30</b>
<b>4.5 Treinamento da Rede Neural.....</b>	<b>31</b>
<b>4.6 Resultados.....</b>	<b>32</b>
<b>4.6.1 Testando a rede com os dados de teste.....</b>	<b>34</b>
<b>5 CONCLUSÃO.....</b>	<b>36</b>
<b>REFERÊNCIAS .....</b>	<b>38</b>

## 1. INTRODUÇÃO

Como descrito no Manual Diagnóstico e Estatístico de Transtornos Mentais 5ª edição (DSM-5, 2013), os transtornos do neurodesenvolvimento são um grupo de determinadas características que se apresentam ainda no período de desenvolvimento, geralmente, antes mesmo da criança ingressar em uma instituição de ensino.

Dentro desse grupo de transtornos do neurodesenvolvimento tem-se o Transtorno do Espectro Autista (TEA), que tem como suas principais características as dificuldades de comunicação e interação social e padrões restritos e repetitivos de comportamentos, interesses e atividades (APA, 2020).

Mesmo com os avanços na área da medicina em relação a tecnologia, o diagnóstico para o TEA não é simples, tendo em vista que não existem exames laboratoriais que possam indicar tal diagnóstico. O diagnóstico é dado por psicólogos, psiquiatras e neurologistas baseados no DSM-5, no qual são feitas entrevistas com o próprio indivíduo ou com os responsáveis quando se trata de crianças, como também análise de comportamentos. Além dessas etapas, os profissionais utilizam escalas que são métodos desenvolvidos por especialistas no diagnóstico do TEA. Essas escalas são resultadas de questões objetivas que são respondidas pelo especialista com base nas etapas de entrevista e análise de comportamento do paciente (ARTONI et al., 2020).

Levando em consideração toda a dificuldade para chegar a um diagnóstico conclusivo do TEA, é possível aplicar *machine learning* aos dados coletados de pessoas autistas? Caso possível os resultados irão apresentar uma *accuracy* (precisão) confiável para auxiliar profissionais especialistas no diagnóstico? Esses resultados também podem ser usados por pais, responsáveis ou profissionais de saúde não especialistas no assunto a ter um pré-diagnóstico para que o paciente seja encaminhado a um especialista?

Para responder a esta questão tem-se como objetivos

### **Geral**

Desenvolver um algoritmo por meio de técnicas de *machine learning* que seja capaz de classificar pacientes portadores do transtorno do espectro autista.

### **Específicos**

- Entender o Transtorno do Espectro Autista;
- Estudar e entender algoritmos de *Machine Learning*;
- Aplicar a linguagem de programação Python na construção da rede neural;
- Definir o melhor algoritmo de *Machine Learning* para a análise do espectro autista.

Considerando a inviabilidade de algumas famílias com menos recursos a chegar a um profissional especialista para ter um diagnóstico, esse estudo é justificado pelos seguintes fatores, mostrar formas alternativas através da tecnologia para auxiliar profissionais especialistas no diagnóstico do TEA, mostrar que é possível ter de forma mais acessível um indicativo de um possível diagnóstico para o TEA, explorar a aplicabilidade dos algoritmos de *Machine Learning* aplicados a padrões de respostas relacionados ao TEA.

Este trabalho tem o intuito de aplicar algoritmos de *Machine Learning* a dados de pessoas diagnosticadas com TEA, visando gerar conhecimento, com isso essa pesquisa será aplicada. Quanto ao seu objetivo será exploratório e explicativo, levando em consideração a necessidade de conhecer mais profundamente o assunto e subsequentemente justificá-los. A abordagem do problema se dá de forma quantitativa, utilizando os dados já disponíveis que serão analisados e processados visando obter os resultados desejados. O mesmo utiliza pesquisa bibliográfica e experimental como técnicas, levando em consideração todo o compilado de obras de terceiros de modo a levantar toda a base teórica necessária para compreensão do tema, também se trata de uma investigação empírica no qual será manipulado e analisado todos os dados disponíveis.

Para desenvolver o assunto, analisar os dados e apresentar os resultados essa pesquisa se divide em introdução, referencial teórico; metodologia; discussão; análise e avaliação dos dados coletados previamente e conclusão.

## 2. FUNDAMENTAÇÃO TEÓRICA

### 2.1 O Transtorno do Espectro Autista

#### 2.1.1 Histórico do Autismo

De sua etimologia a palavra Autismo é derivado do grego AUTOS, que significa “por si mesmo”, o Autismo tem sua história um tanto quanto conturbada, foi usada em 1906 pela primeira vez por Plouller que possuía um estudo em andamento com pessoas diagnosticadas com demência. No entanto, só passou a ser difundido em 1911 quando o psiquiatra Eugen Bleuler ampliou essas relações, relacionando como um sintoma fundamental da esquizofrenia, o que chamou de “pensamento autista”, classificando como ruptura entre a pessoa e o mundo externo, perda do contato com a realidade, ausência ou dificuldade de afeto e ideias desconexas (DIAS, 2015).

Mesmo com as pesquisas de Plouller e Bleuler, nenhum trabalho científico havia sido publicado referindo-se a esse assunto, até que o psiquiatra austríaco Leo Kanner em 1943 publicou “*Autistic Disturbances of Affective Contact*” (distúrbio autístico do contato afetivo), que se tratava de um relatório clínico de onze crianças sendo oito meninos e três meninas. Kanner, não inventou o termo “autístico” ou “autismo” ele simplesmente adequou esse nome que era usado como descrito por Bleuler a pessoas acometidas com outro tipo de transtorno que era a esquizofrenia, essa ligação entre autismo e esquizofrenia traria grande confusão até serem distintas em 1950 (DONVAN; ZUCKER, 2017).

Foi Kanner, quem definiu certos padrões nessas crianças o que denominou como tendo manias e estereótipos, apego excessivo por rotinas e monotonia, dificuldade de lidar com certas mudanças, maneiras diferentes de se comunicar e algumas delas podendo apresentar e ecolalia imediata (repetir frases que acabou de ouvir) ou tardia (repetir frases que ouviu a dias atrás). Kanner foi infeliz em um ato que ele mesmo veio a pedir desculpas anos depois ao relacionar a causa das crianças que apresentavam tais características as mães, o que ficou conhecido como “mães geladeiras”, pois a culpa das crianças serem retraídas se dava por conta da indiferença apresentada pelos pais, não dando atenção e tratando os filhos de forma “fria” sem afeto (DONVAN; ZUCKER, 2017).

O DSM é um material destinado a profissionais que atuam na área da saúde mental, no qual são descritos os transtornos mentais e como chegar ao seu diagnóstico. Hoje em sua quinta edição desde a primeira, traz referência ao autismo. Na sua primeira edição em 1952, o DSM incluiu o autismo não como um diagnóstico, mas sim como um subgrupo da

esquizofrenia, essa relação só veio a ser desfeita com a publicação do DSM-III em 1980, e só em 1994 com o DSM-IV que trouxe a definição do autismo indicando as três principais características do TEA, sendo elas: a dificuldades de socialização e comunicação e a presença de interesses restritos e estereotipados, mas mesmo assim existia várias categorias de classificação, podendo ser classificado como Síndrome de Rett, autismo, Transtorno Desintegrativo da Infância, Transtorno Invasivo do Desenvolvimento e Síndrome de Asperger e não somente o autismo (LONNOR NETO; VALE; TEIXEIRA, 2016; BRETANI et al., 2013).

Só em 2013 com a publicação do DSM-V, o mais atual que trouxe várias mudanças dentro delas as principais foram a nova nomenclatura para o autismo sendo conhecido agora com o Transtorno do Espectro Autista ou TEA, eliminou a síndrome de Rett, exclusão do diagnóstico da síndrome de Asperger sendo diagnosticado como TEA. A categoria de transtornos invasivos do desenvolvimento será substituída por TEA, um novo sintoma foi incluído, acréscimo de um critério de gravidade sendo dividido em 3 níveis: leve moderado e severo (LONNOR NETO; VALE; TEIXEIRA, 2016; BRETANI et al., 2013).

### 2.1.2 O que é o TEA?

O TEA passou por diferentes definições desde que a palavra autismo foi utilizada pela primeira vez em 1906, chegando hoje a uma definição bem estruturada que caracteriza uma pessoa autista. De forma sucinta, o TEA é distúrbio neurobiológico que tem influência direta em áreas imprescindíveis para o desenvolvimento pessoal, sendo elas: emocional, comunicação social influenciando na socialização, atingindo tanto a comunicação verbal e a não verbal, comportamental apresentando padrões de interesses ou atividades particulares e movimentos característicos repetitivos. Tais características tem influência direta na capacidade da pessoa com TEA de se relacionar com outras pessoas. O TEA não apresenta apenas um, mas diferentes tipos de classificação tanto que é comumente usada a palavra espectro para se referir aos diferentes níveis que o TEA pode interferir na pessoa (LUBETSKY et al., 2011).

### 2.1.3 Sintomas

Os sintomas do TEA geralmente são identificados nos primeiros anos de vida pelos próprios pais ou responsáveis da criança, quando tais sintomas não são identificados

previamente o diagnóstico futuro provavelmente será afetado. Os sintomas podem ser diferentes em cada indivíduo com TEA, tanto que existem os níveis diagnósticos leve, médio e grave o que acaba dificultando a percepção quando se trata de um indivíduo com sintomas leves (VIEIRA; BALDIN, 2017).

Nos primeiros anos de vida a criança com TEA pode apresentar certa aversão por afeto, não manter contato visual, não demonstrar expressões faciais como resposta a interação, atraso no desenvolvimento da comunicação entre outros. Conforme a criança cresce tais sintomas podem desaparecer. É comum que indivíduos diagnosticados com TEA apresentem um déficit intelectual, tiques, ansiedade, epilepsia, alteração de humor, agressividade e hiperatividade (VIEIRA; BALDIN, 2017).

#### 2.1.4 Como é diagnosticado?

Vários especialistas estão aptos a dar um diagnóstico conclusivo de TEA, entre eles estão os psicólogos, psiquiatras, neurologista e pediatras, no entanto, é mais comum que um profissional mais ligado a psicologia que se especialize na relação do cérebro com o funcionamento cognitivo, emocional e comportamental humano esteja envolvido no processo de diagnóstico (APA, 2017).

Embora o TEA possa ser diagnosticado antes dos dois anos de idade a idade média é de quatro anos e seis meses, entretanto, existe pessoas que são diagnosticadas já na idade adulta. Como não existe atualmente exames médicos ou laboratoriais que possam dar um diagnóstico conclusivo para o TEA os especialistas para o processo de diagnóstico recorrem a uma sequência de fontes de informação do paciente, como entrevistas, observações comportamentais, testes de habilidade cognitiva e linguagem, exames médicos para outras condições possam ser descartadas e até mesmo entrevistas com pais, responsáveis, professores ou qualquer pessoa que tenha relação direta com o paciente (APA, 2017).

Para auxiliar no processo de diagnósticos especialistas podem seguir a edição mais atual do manual diagnóstico e estatístico de transtornos mentais, mais conhecido como DSM-V desenvolvido pela associação americana de psicologia (APA), que engloba apenas os transtornos mentais descrevendo cada um e a forma de diagnosticá-los.

No diagnóstico do transtorno do espectro autista, as características clínicas individuais são registradas por meio do uso de especificadores (com ou sem comprometimento intelectual concomitante; com ou sem comprometimento da linguagem concomitante; associado a alguma condição médica ou genética conhecida ou a fator ambiental), bem como especificadores que descrevem os

sintomas autistas (idade da primeira preocupação; com ou sem perda de habilidades estabelecidas; gravidade). Tais especificadores oportunizam aos clínicos a individualização do diagnóstico e a comunicação de uma descrição clínica mais rica dos indivíduos afetados. Por exemplo, muitos indivíduos anteriormente diagnosticados com transtorno de *Asperger* atualmente receberiam um diagnóstico de transtorno do espectro autista sem comprometimento linguístico ou intelectual (DSM-V, 2013, p. 32).

Os diagnósticos são dados levando em consideração apenas o paciente em questão, no entanto, existem algumas estatísticas que são levadas em consideração, como idade dos pais avançada e peso baixo no nascimento, são fatores que podem contribuir para o risco de TEA (DSM-V, 2013).

Relativo a cultura não existe ligação direta para o risco e sim que a forma de interação social de determinada cultura pode influenciar no desenvolvimento, também relacionado a cultura e fatores socioeconômicos está ligado a idade em que o paciente será diagnosticado e com isso os prejuízos e benefícios no desenvolvimento caso o diagnóstico seja dado tardiamente ou na infância (DSM-V, 2013).

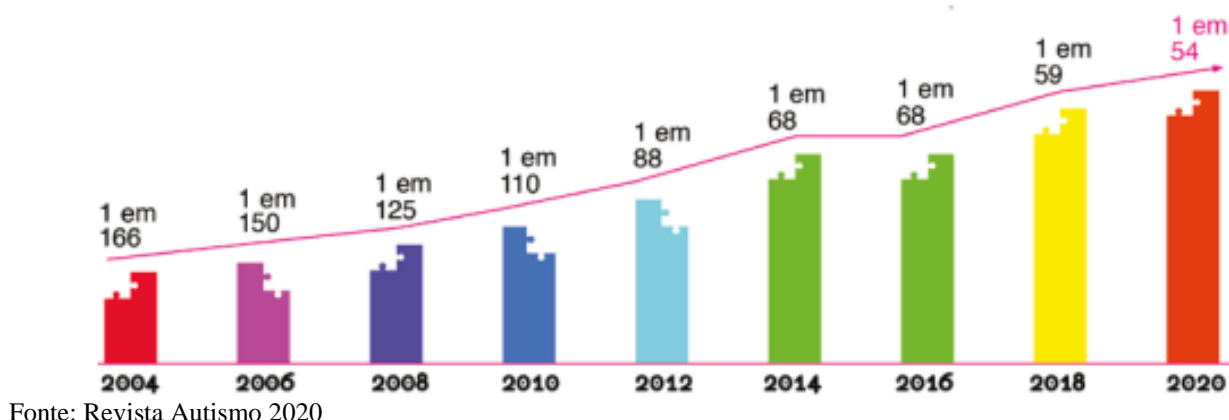
Relacionado a gênero tem-se a relação de diagnósticos de quatro pacientes do sexo masculino para uma do sexo feminino, e dados clínicos mostram que pacientes do sexo feminino têm mais probabilidade de ter déficit intelectual (DSM-V, 2013).

Segundo o último levantamento realizado em 2020 pelo Centro de Controle de Doenças e Prevenção do Governo dos EUA o CDC (*Centers for Disease Control and Prevention*), que é realizado a cada dois anos, mostra uma relação que para cada 54 crianças uma é diagnosticada com TEA.

Como observado na Figura 1, há um aumento na proporção de crianças diagnosticadas com TEA a cada novo levantamento, parte desse aumento está relacionado com a mobilização dos profissionais de saúde juntamente com a conscientização social sobre a importância do diagnóstico e tratamento do TEA. Especialistas nos Estados Unidos também observaram uma inclusão maior de diferentes grupos étnicos e raciais nos diagnósticos e tratamentos no país como outro ponto para o aumento dos diagnósticos de TEA (JUNIOR, 2020).



**Figura 1 – Quantidade de casos de TEA por nascimento**



### 2.1.5 Tratamentos

Como o TEA não apresenta características biológicas se torna inviável tratamentos definitivos ou que apresentem cura. Com um diagnóstico precoce e o acompanhamento de um profissional juntamente com o suporte familiar e educacional é possível desenvolver medidas que minimizam a gravidade do TEA (VIEIRA; BALDIN, 2019).

De início é necessário desenvolver um plano individual para cada paciente, pois cada um possui suas características e devem ser tratadas de forma específica, além disso um acompanhamento com psicólogo não apenas o paciente, mas sim com todos os familiares que convivem com ele diariamente pode trazer melhores resultados (VIEIRA; BALDIN, 2019).

Todos os tratamentos têm por objetivo trazer uma qualidade de vida melhor para o indivíduo diagnosticado com TEA, e para isso pode-se ter um conjunto de intervenções conjuntas para se obter bons resultados como terapia comportamental, acompanhamento com fonoaudiólogos, treinamentos e práticas de habilidades comportamentais entre outros (VIEIRA; BALDIN, 2019).

## 2.2 Algoritmos de *Machine Learning*

A maioria dos programas de Inteligência Artificial são baseados no ser humano, não diferente o *Machine Learning* ou Aprendizado de Máquina também é baseado na forma como os seres humanos aprendem. De acordo com Campos (1986, p.30) “A aprendizagem pode ser

definida como uma modificação sistemática do comportamento, por efeito da prática ou da experiência, com um sentido de progressiva adaptação ou ajustamento”.

O objetivo principal para o desenvolvimento de algoritmos de aprendizagem de máquina é realizar tarefas que uma pessoa não consegue ou teria muita dificuldade para realizar, como por exemplo reconhecer uma pessoa pela imagem do rosto aparentemente é uma atividade relativamente fácil, no entanto, o melhor dos programadores teria muita dificuldade para construir um algoritmo que reconhecesse cada pessoa e além de tudo deveria ter um algoritmo de reconhecimento específico para cada uma delas, já com o aprendizado de máquina com algumas imagens da pessoa o algoritmo de *machine learning* consegue analisar as imagens, definir e reconhecer padrões e com isso irá alterar seu comportamento, e da próxima vez que tiver acesso a imagem de uma pessoa já analisada conseguirá dizer quem é a pessoa, pois já aprendeu a reconhecer aquela pessoa em específico (RUSSEL; NORVIG, 2013).

Existem três tipos principais de *machine learning*, sendo eles: aprendizagem supervisionada, aprendizagem não supervisionada e aprendizagem por reforço.

Na aprendizagem supervisionada existem dados de entrada e dados de resposta, como por exemplo classificação de um animal, tem-se os dados como quatro patas, pelo longo, porte médio, etc. e correlacionado a isso tem-se a resposta cachorro, ou seja, relaciona os dados de classificação com a resposta, a máquina já tem acesso aos dados de características e os de classificação (RUSSEL; NORVIG, 2013).

Diferente da supervisionada a aprendizagem não supervisionada tenta criar padrões a novos dados, não tendo nenhum *feedback* (dados de correlação) dessas entradas como resposta. Como exemplo serviços de *streaming*, que irá sugerir outro produto quando o usuário terminar de consumir o atual, baseado em dados de outros usuários que também consumiram o conteúdo e com os dados do próprio usuário irá sugerir algo, ou seja, o algoritmo criou um perfil baseado em dados de entrada sem uma resposta para comparar (RUSSEL; NORVIG, 2013).

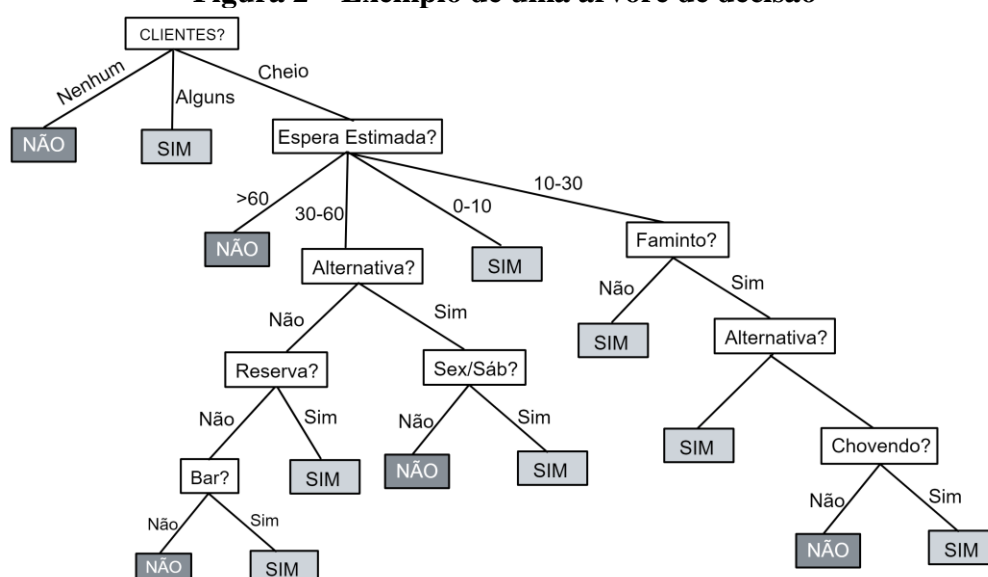
Na aprendizagem por reforço o algoritmo tenta decidir qual a melhor forma de agir, relaciona-se uma ação a um indicativo de sucesso ou fracasso, conforme a execução a máquina irá relacionar ações que levaram a um resultado positivo como sucesso e a resultados negativos a fracasso, e caso se repita irá evitar ações que levem a fracasso e preferir ações que levem a sucesso (RUSSEL; NORVIG, 2013).

### 2.2.1 Aprendizagem por Árvore de decisão

O modelo de árvore tem o objetivo de chegar a uma resposta passando por determinados impasses, a árvore tem sua estrutura dividida em níveis, em que o nível 1 está o nó raiz, no nível 2 estão os filhos do nó raiz e assim sucessivamente, existem dois tipos de nós: os nós folhas que são os que apresentam um estado final e os nós de decisão que contêm um impasse e possui outros nós como filhos, que dependendo da solução irá direcionar para o próximo nó (MONARD; BARANAUSKAS, 2003).

Árvores de decisão é uma forma de distribuir os atributos para chegar a um resultado, a Figura 2 apresenta um exemplo cotidiano que pode ser representado por árvore de decisão, os nós que possuem impasses mais decisivos costumam vir primeiro pois eles têm um peso maior de decisão podendo chegar ao resultado ou a um nó de decisão que chegue ao resultado mais rápido.

**Figura 2 – Exemplo de uma árvore de decisão**



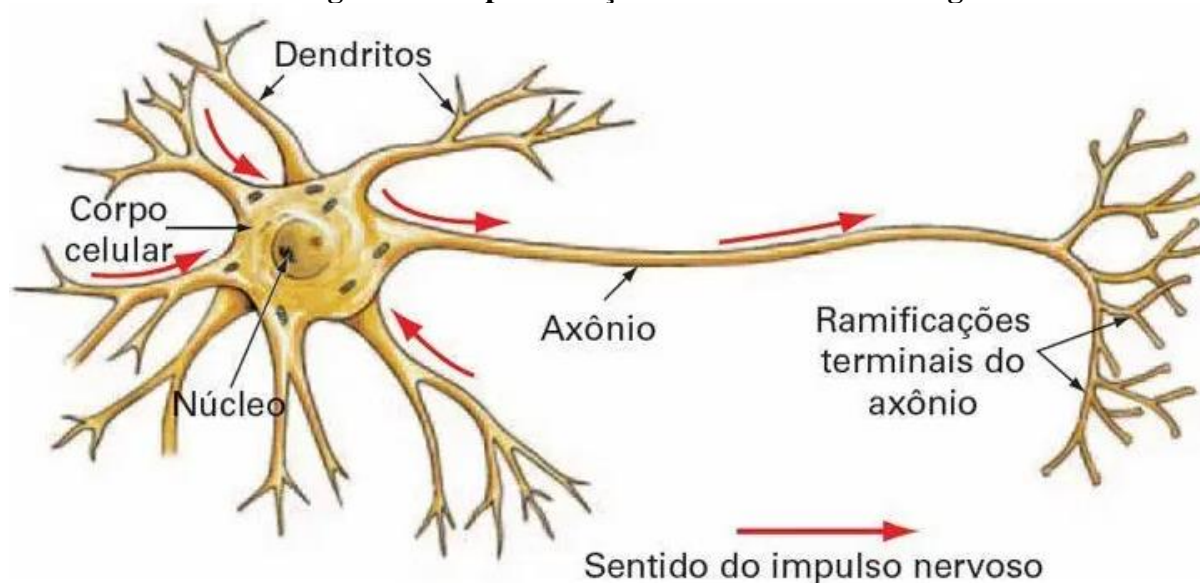
Fonte: Adaptado de Inteligência Artificial 3ª edição (2013)

Na Figura 2 é apresentado um exemplo em forma de árvore de decisão que corresponde a resposta se uma pessoa irá ou não esperar por uma vaga no restaurante, observa-se que existem dois estados finais sim ou não, e que cada nó de decisão possui um impasse que leva a outro nó de decisão ou a um nó folha que representa o estado final.

### 2.2.2 Redes Neurais Artificiais

Os modelos computacionais de Redes Neurais Artificiais são baseados no funcionamento cérebro humano apresentando capacidades e características como ser tolerante a falhas, capacidade de generalizar, adaptação, capacidade de aprender, não linearidade, processar informações que não se tem total certeza e paralelismo. O cérebro humano é formado por bilhões de neurônios e se por acaso algum não estiver funcionando isso não irá afetar a funcionalidade do todo, vários neurônios estão ativados simultaneamente trabalhando com informações diferentes ou trabalhando com a mesma informação de forma diferente. A Figura 3 representa uma ilustração de como é um neurônio biológico e a Figura 4 apresenta a ilustração de um neurônio artificial e da mesma forma que o cérebro é formado por diversos neurônios biológicos interligados, as redes neurais são formadas por diversos neurônios artificiais, divididos em camadas, onde cada um processa e passa a informação para o próximo neurônio (BRAGA; LUDEMIR; CARVALHO, 2000).

**Figura 3 – Representação de um neurônio biológico**



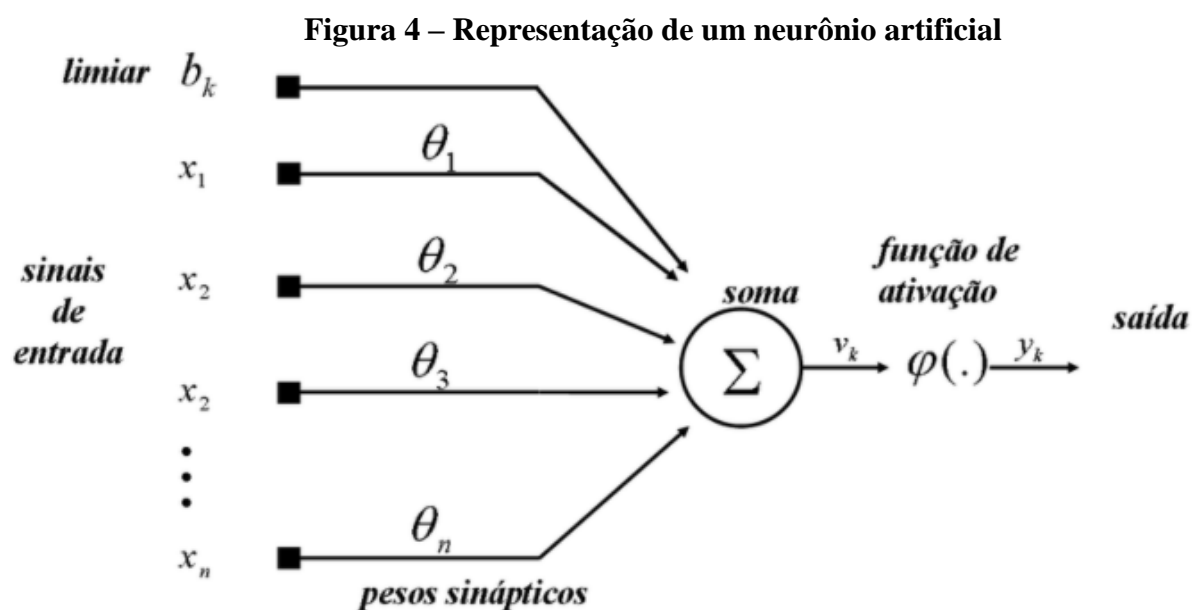
Fonte: Deep Learning Book, 2021

Os neurônios biológicos são divididos praticamente em três partes, sendo elas o corpo, axônio (juntamente com suas ramificações) e os dendritos, são essas partes responsáveis por receber informações em forma de sinais elétricos, processá-los e passar para o próximo neurônio.

O sistema nervoso seria uma rede de neurônios, compostos por um axônio e por soma (corpo celular). Existem sinapses entre o axônio de um neurônio e o corpo

celular de outro, e estas sinapses podem ser excitadas por impulsos. Quando o nível de excitação excede um limiar específico de cada neurônio, este inicia um impulso, a ser transmitido por seu axônio para as sinapses às quais está ligada, e a etapa mais lenta desse processo de transmissão é a passagem pelas sinapses (ALMADA, 2019, p. 32).

Baseado nos neurônios biológicos tem-se os neurônios artificiais que paralelamente aos biológicos que compõem o sistema nervoso, os artificiais compõem uma rede neural. Na Figura 4 observa-se a representação de um neurônio artificial, os sinais de entrada são os dados que provêm da informação direta ou da informação já processada por outro neurônio, relacionado a cada entrada é atribuído um peso, que é utilizado para definir a força do sinal da entrada específica, ou seja, o quanto aquela entrada é importante com relação as outras, todas essas entradas relacionadas com seus pesos passam por uma combinação e por uma função de ativação que irá gerar um valor de saída podendo ser uma resposta final de classificação ou a entrada para outro neurônio artificial. Essa ligação entre neurônios e camadas de neurônios da origem a uma rede neural (RUSSEL; NORVIG, 2013).



Fonte: Researchgate, 2021

### 2.2.3 Aprendizagem Estatística

Em aprendizagem estatística pode-se destacar dois principais pontos, dados e hipóteses, onde os dados representam as evidências, comparado a uma função os dados seriam o domínio, a parte variável necessária para chegar a uma imagem (resposta) e as hipóteses são tratadas como probabilidades de como os dados funcionam (RUSSEL; NORVIG, 2013).

A probabilidade é uma área matemática que interage com fenômenos aleatórios, que são eventos onde não se pode prever com total certeza. Trabalhar com aprendizagem baseada em probabilidades se torna uma alternativa quando o problema abrange situações reais onde não se tem toda a informação necessária. A aprendizagem *bayesiana* é uma forma de aprendizagem estatística baseada em probabilidades, ela irá calcular a probabilidade das hipóteses levantadas com base nos dados (GONÇALVES, 2008).

As redes *bayesianas* podem ser representadas como grafos, segundo Gonçalves (2008, p. 8) “Em redes Bayesianas a representação das variáveis e relações é feita utilizando Teoria dos Grafos. As variáveis são os nós e os arcos identificam as relações entre as variáveis, formando um grafo dirigido e sem ciclos”.

### 3. METODOLOGIA

Com intuito de gerar o conhecimento necessário para aplicar algoritmos de *Machine Learning* a dados de pessoas diagnosticadas com TEA, essa pesquisa será aplicada. Quanto ao seu objetivo será exploratório e explicativo, levando em consideração a necessidade de conhecer mais profundamente o assunto e subsequentemente justificá-los. A abordagem do problema se dá de forma quantitativa, utilizando os dados já disponíveis que serão analisados e processados visando obter os resultados desejados. Essa pesquisa utiliza pesquisa bibliográfica e experimental como técnicas, pois utiliza um compilado de obras de terceiros de modo a levantar toda a base teórica necessária para compreensão do tema, também se trata de uma investigação empírica no qual será manipulado e analisado todos os dados disponíveis.

O principal objetivo dessa pesquisa está no fato de apresentar uma triagem para um possível diagnóstico de Transtorno do Espectro Autista. Pode-se dividir em quatro etapas principais. Desde o levantamento do conhecimento necessário até a aplicação dos algoritmos de *Machine Learning* aos dados disponíveis.

A primeira etapa consiste em gerar o conhecimento necessário sobre o Transtorno do Espectro Autista, passando pelo seu histórico, sintomas, formas de diagnósticos e a porcentagem de crianças que são diagnosticadas com TEA. O intuito de levantar tais informações é correlacionar com a base de dados e entender quais informações são mais relevantes e o quanto podem influenciar no resultado.

Subsequente tem-se o conteúdo referente aos algoritmos de *Machine Learning*, trazendo os tipos de algoritmos de aprendizagem de máquina que podem ser aplicados aos dados disponíveis, com objetivo de mostrar em quais condições cada tipo de algoritmo se encaixa melhor e qual irá se adequar melhor ao tipo de informação disponível.

Após explorar e explicar as bases teóricas da pesquisa, inicia-se a parte de análise e manipulação dos dados. A base de dados foi obtida através do site “*UC Irvine Machine Learning Repository*”, que mantém conjuntos de dados como forma de serviço para a comunidade de *Machine Learning*. Esses dados foram coletados através do aplicativo *ASDTests* desenvolvido em 2018. Esses dados contêm um compilado de 3 grupos sendo eles, o de crianças que contêm 104 instâncias, adolescentes contendo 292 e adultos que contêm 704, totalizando os 3 conjuntos tem-se 1100 instâncias. Todas as 1100 instâncias são compostas por 21 dados, dentre eles 20 são dados de entrada e 1 de saída, o dado de saída representa a classificação entre autista ou não autista que está no formato string podendo ser “yes” ou “no”,

os 20 dados de entrada são distribuídos da seguinte forma: os 10 primeiros dados são referentes a perguntas de comportamento do indivíduo ao qual está respondendo, esses dados estão disponíveis de forma binária 1 ou 0 dependendo da resposta para cada uma das 10 perguntas.

Na sequência dos dados tem-se: idade no formato inteiro, gênero no formato de *string* sendo “f” ou “F” para feminino e “m” ou “M” para masculino, etnia no formato de *string*, *jundice* (Icterícia) no formato *string* “yes” ou “Yes” e “no” ou “No” que indica se o indivíduo possui Icterícia, *austim* no formato *string* “yes” ou “Yes” e “no” ou “No” que indica se algum membro família possui Transtorno Invasivo do Desenvolvimento, país no formato de *string* que é preenchido com o país onde o indivíduo reside, o próximo dado é referente se a pessoa já usou o aplicativo antes, também está no formato de *string* “yes” ou “no”, resultado no formato inteiro que representa o tipo de método de triagem escolhido. Intervalo de idade no formato *string* sendo três intervalos “4-11 years”, “12-16 years” e “18 and more”. Relação no formato *string* que indica a relação entre a pessoa que preencheu o formulário e a pessoa ao qual as respostas se referenciam, podendo ser a própria pessoa, profissional de saúde, pais, entre outros.

O Quadro 1 mostra a descrição da base de dados, contendo o nome do dado (**Dado**), o tipo da variável (**Tipo**), e uma descrição do que significa aquele dado (**Descrição**).

**Quadro 1 – Relação elementos da base de dados com seu tipo e descrição.**

<b>Dado</b>	<b>Tipo</b>	<b>Descrição</b>
<i>A1_Score</i> até <i>A10_Score</i>	Binário 0 ou 1	Baseado na resposta de um determinado comportamento.
<i>age</i>	Inteiro	Referente a idade
<i>gender</i>	<i>String</i>	Referente ao sexo, “m” ou “f”, masculino ou feminino, respectivamente
<i>ethnicity</i>	<i>String</i>	Referente a etnia
<i>jundice</i>	<i>String</i>	Referente se tem ou não Icterícia, podendo se “yes” e “no”, caso tenha ou não, respectivamente.
<i>austim</i>	<i>String</i>	Referente se existe alguém na família diagnosticado com Transtorno Invasivo do Desenvolvimento
<i>contry_of_res</i>	<i>String</i>	Referente ao país onde está residindo.
<i>used_app_before</i>	<i>String</i>	Referente se já utilizou ou não o aplicativo antes, “yes” ou “no” que representa se já utilizou antes ou nunca utilizou, respectivamente.
<i>result</i>	Inteiro	Referente ao método de triagem utilizado.



<i>age_desc</i>	<i>String</i>	Referente ao intervalo de idade que o indivíduo se encontra “4-11 years”, “12-16 years” ou “18+ years”
<i>relation</i>	<i>String</i>	Referente a ligação entre a pessoa que preencheu as informações com a qual as respostas se referem, podendo ser a própria pessoa, profissional de saúde, pais entre outros.
<i>Class/ASD</i>	<i>String</i>	Referente ao diagnóstico para TEA, podendo ser “yes” ou “no”, caso diagnóstico positivo e diagnóstico negativo, respectivamente.

Fonte: Próprio Autor

Esses dados são de extrema relevância levando em consideração o referencial teórico sobre TEA, principalmente os 10 primeiros dados em que apresenta respostas para determinados tipos de comportamentos, como também gênero no qual existe uma relação com um diferencial expressivo que a cada pessoa do sexo feminino diagnosticada com TEA tem-se três do sexo masculino com diagnósticos positivos. No entanto, alguns dados não são interessantes manter na base de dados, dentre eles “*contry\_of\_res*” e “*used\_app\_before*”, que como descrito no referencial teórico sobre TEA não tem ligação alguma com o diagnóstico final. Ao retirar esses dois dados tem-se os dados relevantes para aplicar ao algoritmo de *Machine Learning*, no entanto, antes é necessário passar por uma normalização para que o algoritmo reconheça os dados corretamente.

A base de dados contém dados iguais que aparecem de forma diferente, como em alguns campos de dados em que além das classificações de *string* “yes” ou “no”, existia outros formatos como “YES”, “NO”, “Yes” e “No”, que caso não fossem normalizados ao invés de reconhecer como duas variáveis diferentes iria reconhecer como seis variáveis diferentes, para isso, todos os dados que se referem a um mesmo tipo foram alterados para ficarem em um mesmo formato, trocando todos os tipos de “yes” por 1 e todos os tipos de “no” por 0. Todos os dados que possuíam seu formato em *string* e como possibilidade “yes” ou “no” foram alterados dessa forma. A normalização de sexo também foi realizada dessa forma evitando diferenciação entre maiúscula e minúscula, todos com sexo masculino (m ou M) foram substituídos por 1 e todos com sexo feminino (f ou F) foram substituídos por 0.

Alguns dos dados não foram preenchidos no momento em que o indivíduo respondeu ao questionário e onde deveria ficar sua resposta ficou apenas um “?”, e por se tratar de um caractere especial é necessário substituir, em todas essas ocorrências o “?” foi substituído pela *string* “Unknown”.

Após feita a normalização dos dados no próprio ambiente do código é necessário dividir uma parte para ser os dados de treinamento e outra para ser os dados de teste, a fim de checar a assertividade da aprendizagem na classificação, sendo essa divisão 60% para dados de treinamento e 40% para dados de teste, o que permite gerar uma matriz de confusão que apresenta os dados que foram classificados corretamente e os que foram classificados incorretamente.

Após a normalização e manipulação dos dados tem-se a etapa de criar e configurar a rede neural para treiná-la baseando-se nos dados de entrada relacionados aos dados de classificação. O modelo de *Machine Learning* utilizado foi o *multilayer perceptron*, utilizando o modelo disponível pela biblioteca *Keras* versão 2.4.3 que é executada no *framework TensorFlow* versão 2.4.3 que é baseado em *python*. Toda parte de manipulação e normalização dos dados a nível de código foram escritas utilizando a linguagem *python* na versão 3.9.2 juntamente com a IDE *spyder* versão 4.1.5.

## 4 DISCUSSÃO, ANÁLISE E AVALIAÇÃO DOS DADOS COLETADOS

O TEA pode ser definido como um distúrbio neurobiológico que tem influência direta em áreas imprescindíveis para o desenvolvimento pessoal, sendo elas: emocional, comunicação social influenciando na socialização, atingindo tanto a comunicação verbal e a não verbal, comportamental apresentando padrões de interesses ou atividades particulares e movimentos característicos repetitivos. Tais características tem influência direta na capacidade da pessoa com TEA de se relacionar com outras pessoas (LUBETSKY et al., 2011).

Como as pessoas que são diagnosticadas com TEA geralmente possuem algumas características que são praticamente um padrão comportamental, principalmente se baseando no Manual Diagnóstico e Estatístico de Transtornos Mentais o DSM-5, este trabalho tem como objetivo aplicar aos dados coletados de pessoas classificadas com diagnóstico positivo e negativo para TEA algoritmos de *Machine Learning* com propósito de chegar a um pré-diagnóstico.

### 4.1 Normalizando os dados

Para treinar a rede neural se fez necessário ter uma base de dados composta por dados de entrada e dados de saída, os dados de entrada irão passar pela rede que ao final irá gerar uma resposta e comparar com os dados de saída e quando necessário irá fazer as modificações na rede. No entanto, antes de submeter os dados a rede neural é necessário passar por uma etapa de normalização, que consiste em não deixar que dados que tenham o mesmo significado sejam interpretados de forma diferente, como uma *string* ‘yes’ e outra ‘YES’ sejam interpretadas como uma resposta diferente, e com isso se faz necessário normalizar todos os dados de entrada e saída.

Primeiramente foi necessário importar todas as bibliotecas e *frameworks* que são necessários para ler os dados, para modificar dados para um formato compatível com as funções de manipulação da linguagem python, para construção da rede neural, para gerar gráficos, entre outras.

Na Figura 5 é possível observar todas as bibliotecas e *frameworks* que foram importados para o desenvolvimento do trabalho.

**Figura 5 – Bibliotecas e *Frameworks* importados**

```
7
8  import numpy as np
9  import pandas as pd
10 import tensorflow as tf
11 import keras as kr
12 from keras.models import Sequential
13 from keras.layers import Dense
14 from keras.utils import np_utils
15 from sklearn.preprocessing import LabelEncoder
16 from keras.optimizers import Adam
17 import matplotlib.pyplot as plt
18 from sklearn.model_selection import train_test_split
19 from sklearn.metrics import classification_report, confusion_matrix
20
```

Fonte: Próprio autor (2021)

Antes de iniciar a normalização fez-se necessário a leitura dos dados. Os dados estavam divididos em 3 arquivos diferentes um para crianças (4 a 11 anos) contendo 104 instâncias, outro para adolescentes (12 a 18 anos) contendo 292 instâncias e outro para adultos (maiores de 18 anos) contendo 704 instâncias, esses 3 arquivos possuíam as mesmas instâncias de entrada e saída e por isso foi possível unir em um único arquivo e ler através da biblioteca pandas.

Com os dados disponíveis para serem manipulados no algoritmo a próxima fase consiste em normalizar os dados.

A Figura 6 demonstra 3 funções que tem como objetivo normalizar a coluna correspondente de acordo com o necessário, a primeira função *normalizeGender* recebe a coluna referente as informações de gênero e troca “m” seja ele maiúsculo ou minúsculo por 1 e “f” por 0 seja ele maiúsculo ou minúsculo. Já a função *normalizeAge* tem o objetivo de trocar os caracteres especiais ‘?’ por ‘Unknown’ pois o framework *Tensor Flow* não aceita caracteres especiais.

**Figura 6 – Algumas funções de normalização**

```

20
21 def normalizeGender(genderColumn):
22     return genderColumn.str.replace('M', '1').str.replace('m', '1')
23     .str.replace('F', '0').str.replace('f', '0').astype('category')
24
25 def normalizeAge(ageColumn):
26     return ageColumn.str.replace('?', 'Unknown').astype('category')
27
28 def normalizeRelativesWithAutism(autismColumn):
29     return autismColumn.str.replace('yes', '1').str.replace('Yes', '1')
30     .str.replace('no', '0').str.replace('No', '0').astype('category')
31

```

Fonte: Próprio autor

Após a normalização dos dados foi realizada a separação entre os dados de entrada dos dados de saída, com isso são 18 colunas com dados de entrada relacionadas a 1 coluna representando o dado de saída ('yes' que indica diagnóstico positivo para TEA representado por 1 após a normalização e 'no' indicando diagnóstico negativo para TEA representado por 0 após a normalização).

## 4.2 Manipulação dos dados com bibliotecas

Para manipular os dados e montar a rede neural com suas devidas dependências é necessário utilizar algumas bibliotecas e *frameworks*, tanto para montar a rede neural em si quanto para deixar os dados na forma correta antes de submetê-los a rede.

Dentre as bibliotecas utilizadas estão o *TensorFlow* e o *Keras* como as principais. O *TensorFlow* é uma plataforma completa *open source* para *Machine Learning* (ML). Ele tem um ecossistema abrangente e flexível de ferramentas, bibliotecas e recursos da comunidade que permite aos pesquisadores levar adiante ML de última geração e aos desenvolvedores criar e implantar aplicativos com tecnologia de ML (TENSORFLOW, 2021).

O Keras é uma API de aprendizado profundo escrita em Python, executada na plataforma de aprendizado de máquina *TensorFlow*. Foi desenvolvido com foco em possibilitar experimentação rápida com a ideia de que alcançar o resultado o mais rápido possível é a chave para fazer uma boa pesquisa (KERAS TEAM, 2021).

Antes de submeter os dados ao algoritmo de ML é preciso deixá-los na melhor forma para que o algoritmo consiga apresentar uma performance melhor. A biblioteca Pandas possui

um método chamado “*get\_dummies*” que permite converter as variáveis das instâncias em indicadores.

Na Figura 7, a esquerda tem-se a tabela com os dados antes de passarem pelo método *get\_dummies* e a tabela da direita representa os dados após passarem pelo método, esse método consiste em criar uma nova coluna para cada tipo de dado. “A1\_Score” era composto por dois tipos de dados 1 e 0, ao passar pelo método será criado uma coluna para cada tipo de dado daquela coluna, ficando com “A1\_Score\_0” e “A1\_Score\_1” onde irá preencher com 1 no local onde estava preenchido com o valor que a coluna está representando, ou seja, a coluna “A1\_Score\_0” tem valor 1 na posição que antes era 0. O método foi aplicado em todos os dados de entrada resultando em uma tabela binária com dados do tipo 1 e 0, antes de passar pelo *get\_dummies* tinha-se 18 colunas com dados de entrada após o método tem-se 117 colunas representando os dados de entrada composta por 1 e 0.

**Figura 7 – Exemplo método *get\_dummies***

	A1_Score	A2_Score			A1_Score_0	A1_Score_1	A2_Score_0	A2_Score_1
I1	1	0		I1	0	1	1	0
I2	0	1		I2	1	0	0	1
I3	1	0		I3	0	1	1	0
I4	1	1		I4	0	1	0	1
I5	0	1		I5	1	0	0	1

Fonte: Próprio autor (2021).

### 4.3 Divisão dos dados para teste

Para testar o modelo de predição criado após executado o algoritmo de ML é necessário separar uma parte dos dados para treinamento e outra parte para teste. Para fazer essa separação dos dados utilizou-se a biblioteca de ML *sklearn* que possui o método “*train\_test\_split*” que permite separar uma porcentagem dos dados para dados de teste que não irão passar pela rede neural.

Na Figura 8 é possível observar a utilização do método “*train\_test\_split*”, que recebe 3 informações como parâmetros, na primeira posição recebe os dados de entrada, na segunda recebe os dados de saída que servem para comparar as respostas, e por último recebe à

proporção que terá os dados de teste, que nesse caso utilizou-se 0.4 ou 40% dos dados para teste e 60% para treinamento da rede neural. Esse método retorna 4 valores, sendo o primeiro e o terceiro os dados de entrada e saída respectivamente que serão utilizados para treinar a rede neural, e o segundo e quarto sendo os dados de entrada e saída respectivamente que serão utilizados para testes.

**Figura 8 – Separação dos dados de teste**

```
104  
105 X_train, X_test, y_train, y_test = train_test_split(dataSet, classificationEncoded, test_size = 0.4)  
106
```

Fonte: Próprio autor (2021)

#### 4.4 Construção da Rede Neural

Para construção do modelo da rede neural utilizou-se a função “*Sequential*” que anteriormente foi importada da biblioteca *Keras* e é responsável por fornecer os recursos para o treinamento dos dados.

Na Figura 9, é possível observar toda a construção da Rede Neural. Após ter a estrutura (*model*, representado pela linha 107 Figura 9) deve-se adicionar a camada de entrada, camadas ocultas e a camada de saída da Rede Neural. Através do “*Sequential*” representado agora pela variável “*model*” é possível adicionar as camadas através do método “*add*” que recebe uma camada como parâmetro. Para criar uma camada utilizou-se o método “*Dense*” também importado da biblioteca *Keras*, ele é responsável por criar uma camada de neurônios artificiais. Esse método pode receber várias informações como parâmetro, nesse caso recebeu 3 parâmetros, o primeiro representa a quantidade de neurônios que irá compor aquela camada, o segundo parâmetro (*input\_dim*) indica a quantidade de entradas que aquela camada irá receber, e o terceiro parâmetro (*activation*) representa a função de ativação.

**Figura 9 – Criação da Rede Neural**

```

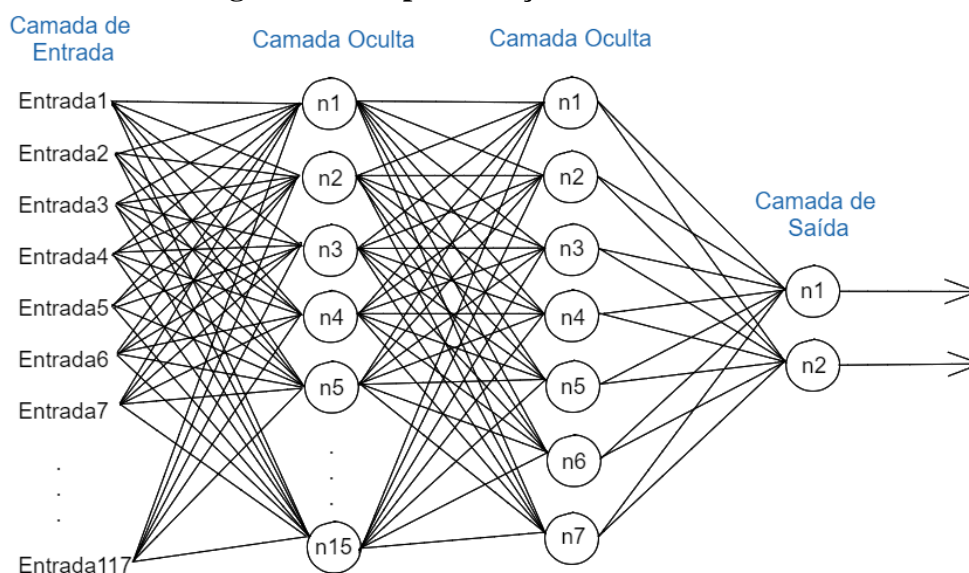
106
107 model = Sequential()
108 model.add(Dense(15, input_dim=117, activation='relu'))
109 model.add(Dense(7, input_dim=15, activation='relu'))
110 model.add(Dense(2, input_dim=7, activation='sigmoid'))
111 adam=Adam(lr=0.001)
112 model.compile(loss='categorical_crossentropy', optimizer=adam, metrics=['accuracy'])
113 historico = model.fit(X_train, y_train, epochs=40)
114

```

Fonte: Próprio autor

Como é possível observar na Figura 9 a primeira camada criada é composta por 15 neurônios artificiais, ela recebe 117 entradas e possui “*relu*” como função de ativação. Já a segunda camada possui 7 neurônios artificiais e irá receber 15 entradas pois a segunda camada recebe a saída da primeira que é composta por 15 neurônios onde cada 1 irá produzir uma saída.

A Figura 10 mostra a representação gráfica de como é a rede neural, a camada de entrada composta pelas 117 colunas de dados, a primeira camada oculta composta por 15 neurônios artificiais, a segunda camada oculta contendo 7 neurônios artificiais e por último a camada de saída que possui dois neurônios artificiais.

**Figura 10 – Representação da Rede Neural**

Fonte: Próprio autor (2021)



## 4.5 Treinamento da Rede Neural

Após a construção do modelo da rede neural, é o momento de treiná-la a partir dos dados de entrada e saída, ou seja, compilar o modelo e depois inserir os dados para treiná-lo. A biblioteca *Keras* possui o método *compile* que é possível observar na linha 112 da Figura 9, esse método é responsável por configurar a rede neural, nele é definido qual será a forma de calcular a perda (*loss*) e qual será o tipo e o valor para otimizar (*optimizer*) os balanceamentos.

Com o modelo da rede compilado é possível iniciar o treinamento com os dados, que é feito através do método *fit* também disponibilizado pela biblioteca *Keras*.

Como observado na linha 113 da Figura 9, o método *fit* recebe três parâmetros, o primeiro são os dados de treino referente as entradas (os 117 valores por instância), o segundo parâmetro são os dados de treino referente as saídas que serão usados como comparativo ao final do cálculo da rede neural, e o terceiro é referente a quantidade de épocas que a rede neural será submetida, ou seja, a quantidade de vezes que o processo se repetirá.

Para visualizar todo o processo de treino posteriormente como a precisão (*accuracy*) e a perda (*loss*) é necessário salvar o processo em uma variável para que seja possível visualizar graficamente quando necessário, esse é o motivo pela variável “*histórico*” na linha 113 da Figura 9 estar recebendo o retorno do método *fit*.

## 4.6 Resultados

Após guardar o resultado no método *fit* na variável “*historico*” é possível ter acesso ao desempenho da rede neural, o retorno salvo na variável “*historico*” possui os resultados referente a precisão da rede (*accuracy*) e também a perda (*loss*).

Para ver graficamente esses resultados utilizou-se o *Matplotlib* que é uma biblioteca de *software* para criação de gráficos e visualizações de dados em geral, feita para linguagem de programação *Python*.

Como descrito na própria documentação do *keras*, a *accuracy* é a frequência com que as previsões são iguais aos rótulos, ou seja, ao ser calculada a saída para a instância que contém as 117 entradas ela deve ser igual a saída esperada, ao final de cada época é calculada a *accuracy* que é a porcentagem em relação aos acertos e erros.

Já a *loss* diferente da *accuracy* não é uma porcentagem dos erros da rede, e sim uma soma de todos os erros para que a rede possa se basear para balancear novamente seu modelo

a documentação do *keras* define que o objetivo das funções de perda é calcular a quantidade que um modelo deve procurar minimizar durante o treinamento (KERAS TEAM, 2021).

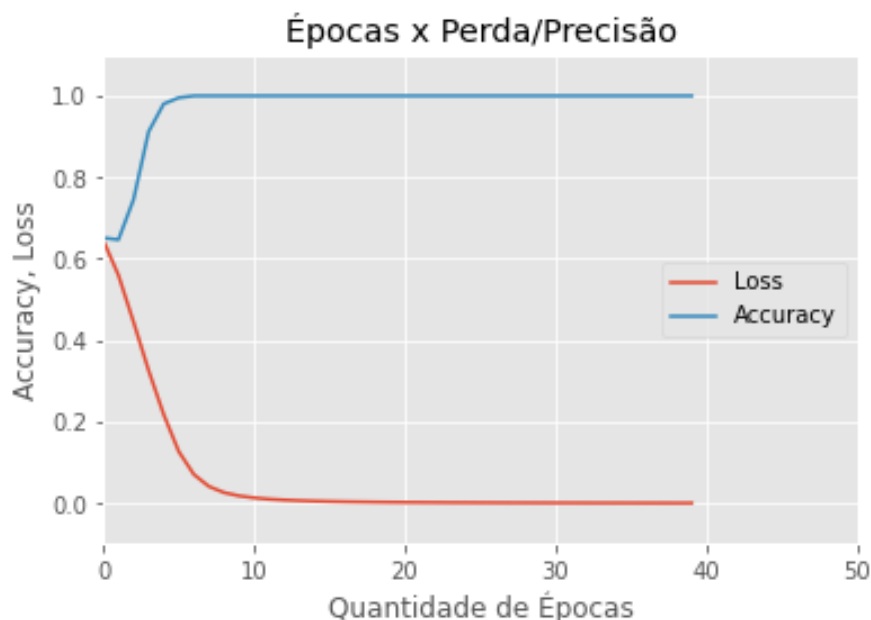
A Figura 11 mostra a configuração com a biblioteca *Matplotlib*, para mostrar a evolução da *accuracy* e de *loss* ao decorrer das épocas, com destaque nas linhas 131 e 132 da Figura 11 onde são passados através da variável *historico* que salva um histórico do treinamento (*accuracy* e *loss*).

**Figura 11 – Configuração para gerar gráficos de *accuracy* e *loss***

```
128
129 plt.style.use("ggplot")
130 plt.figure()
131 plt.plot(historico.history['loss'], label = 'Loss')
132 plt.plot(historico.history['accuracy'], label = 'Accuracy')
133
134 plt.title('Épocas x Perda/Precisão')
135 plt.xlabel('Quantidade de Épocas')
136 plt.ylabel('Accuracy, Loss')
137 plt.legend()
138 plt.axis([0, 50, -0.1, 1.1])
139 plt.show()
140
```

Fonte: Próprio Autor (2021)

A Figura 12 mostra os resultados de *accuracy* e *loss*, como é possível observar a *accuracy* chegou a 100% rapidamente ao mesmo tempo que a *loss* caiu para 0, ou seja, ao final das 40 épocas a rede neural acertou 100% das predições. No entanto, isso não diz que a rede neural acertará 100% dos dados a que for submetida e sim que ela acertou todos os dados de treino. Para ter uma precisão melhor dos resultados se faz necessário gerar uma matriz de confusão.

**Figura 12 – Histórico de *accuracy* e *loss***

Fonte: Próprio Autor (2021)

#### 4.6.1 Testando a rede com os dados de teste

Após treinar a rede neural e chegar a *accuracy* de 100% e *loss* de 0, é o momento de utilizar os dados que foram separados para teste anteriormente para realizar a predição e comparar com as saídas. Como é possível visualizar na Figura 8 foram separados 40% dos dados para teste, como são 1100 instâncias, 440 instâncias são para testes e 660 para treinamento.

Para verificar o quanto a rede neural está acertando deve-se submeter os dados de teste de entrada à rede neural e armazenar suas saídas que são as predições feitas pela rede neural.

Para gerar as predições para os dados de teste referentes as entradas (*X\_test*) utilizou-se o método *predict* que é um método disponível através do modelo compilado (*model*). Como é possível observar na Figura 13 é necessário passar essas predições para o tipo *array NumPy* realizado na linha 126 para que seja possível gerar a matriz de confusão que é uma matriz que mostra a classificação dos resultados, nesse caso será uma matriz 2x2, na posição 1-1 se refere aos verdadeiros que foram classificados com verdadeiros, ou seja, a posição 1-1 da matriz indica os TEA positivos que foram classificados como positivos, na posição 1-2 são os

positivos que foram classificados como negativos, na posição 2-1 são os negativos que foram classificados como positivos e na posição 2-2 são os negativos que foram classificados como negativos, de uma forma melhor as posições 1-1 e 2-2 são os acertos e nas posições 1-2 e 2-1 são os erros.

**Figura 13 – Obtendo previsões com os dados de teste**

```

124
125     predictions = model.predict(X_test)
126     predictions = np.array(predictions)
127
128     print(confusion_matrix(y_test.argmax(axis=1), predictions.argmax(axis=1)))
129

```

Fonte: Próprio Autor (2021)

A Figura 14 mostra o resultado da matriz de confusão gerado pela linha 128 da Figura 13, é possível observar que 276 resultados positivos foram classificados corretamente e 4 erroneamente (resultados positivos classificados como negativos). Referente aos resultados negativos 2 deles erroneamente foram classificados como positivos e 158 corretamente como negativos. Mesmo a acurácia no treino apresentando 100% é possível observar que a rede neural está propensa a erros nesse caso aproximadamente 98,64% dos dados foram classificados corretamente e 1,36% foram classificados de forma errada.

**Figura 14 – Resultado da matriz de confusão**

	276	4	
	2	158	

Fonte: Próprio Autor (2021)

## 5 CONCLUSÃO

Este trabalho iniciou-se com o objetivo geral de desenvolver um algoritmo por meio de técnicas de *machine learning* que fosse capaz de classificar pacientes portadores do transtorno do espectro autista.

Em relação ao objetivo considera-se que foi atingido com sucesso levando em consideração todos os resultados apresentados. Através dos dados disponíveis de pessoas com diagnóstico positivo e negativo para transtorno do espectro autista utilizou-se parte dos dados para treinar uma rede neural para que essa fosse capaz de criar previsões para novos dados classificando-os em indicativo de TEA positivo ou não. Para testar a precisão de acerto dessa rede utilizou-se a parte restante dos dados e chegou-se à precisão de acerto de 98,64%.

Com o estudo da seção 2.1 foi possível estudar e entender mais precisamente o que é o TEA, quais os principais sintomas, como é realizado o diagnóstico e o mais importante as principais características e padrões de uma pessoa que possui TEA. Através do entendimento do TEA foi possível saber quais os dados mais relevantes e quais dados não fariam sentido manter para treinar a rede neural, caso não fosse feita essa diferenciação entre dados que não ajudariam em nada na classificação, esses poderiam prejudicar a assertividade das previsões da rede neural.

O estudo sobre Algoritmos de *Machine Learning* possibilitou entender como funciona uma rede neural, como ela é construída e qual sua estrutura, além disso também mostrou o que é um neurônio artificial e qual o seu objetivo dentro de uma rede neural, e o principal: a seção 2.2 possibilitou entender como funciona a aprendizagem através dos dados que iria-se submeter a essa rede e qual seria a saída (previsão), ou seja, com o estudo e entendimento do TEA juntamente com Algoritmos de *Machine Learning* seria possível utilizá-los para criar o algoritmo que daria início a rede neural de uma forma mais eficiente uma vez que sabe-se quais os dados mais relevantes, a melhor forma de submetê-los a rede e o que esperar como resposta ao final da execução da rede neural (previsões).

Para criar e executar a rede neural, tanto em seu treinamento quanto na parte de testes foi necessário criar um modelo através de algoritmos e para essa construção escolheu-se a linguagem *Python*, todo o processo de normalização e manipulação de dados, importação e utilização de bibliotecas de terceiros, como também a criação e execução da rede neural foi realizada através da linguagem *Python* e o principal motivo de utilizar-se essa linguagem está relacionado a popularidade da linguagem que está diretamente ligada ao suporte recebido e a quantidade de bibliotecas disponíveis e consequentemente testadas amplamente pela

comunidade, o que permitiu utilizar códigos prontos disponibilizados por terceiros através das bibliotecas e frameworks e não construí-los do zero.

Utilizar uma biblioteca ou um *framework* permite aumentar a confiabilidade e performance do algoritmo uma vez que esse código disponibilizado já foi utilizado diversas vezes por várias pessoas diferentes em diversos contextos diferentes, ou seja, vários erros já foram encontrados e arrumados, o que não acontece quando se cria um algoritmo do zero em que a única forma de teste será na própria execução não sendo possível saber se existe um possível erro, o qual será descoberto posteriormente já na parte de execução.

A parte de definir o melhor algoritmo de *Machine Learning* para a análise do espectro autista foi alcançada parcialmente, mesmo a rede neural apresentando um ótimo desempenho na performance e nos resultados faltou testes com outras formas de construção para um comparativo melhor.

Como sugestões para futuras pesquisas seria o fato de executar de formas diferentes a parte de treinamento. Na parte de construção seria utilizar outros tipos de algoritmos além do *Multilayer Perceptron* para fins comparativos, outra forma seria testar separadamente cada base de dados, tendo assim 3 redes de aprendizado um para os dados de crianças, outro para adolescentes e outro para adultos que é a forma como a base de dados vem separada.

## REFERÊNCIAS

APA, American Psychological Association. **Diagnosing and managing autismo spectrum disorder**. Disponível em: <<https://www.apa.org/topics/autism-spectrum-disorder>>. Acesso em: 14 nov. 2020.

APA, American Psychological Association. **Autism Spectrum Disorders**. Disponível em: <<https://www.apa.org/topics/autism>>. Acesso em: 15 nov. 2020.

ARTONI, Arthur; OLIVEIRA, Bruno; DE BARBOSA, Cinthyan Sachs. **Auxílio à Decisão no Diagnóstico do Transtorno do Espectro Autista com M-Health**. In: SEMINÁRIO INTEGRADO DE SOFTWARE E HARDWARE (SEMISH). Cuiabá, 2020.

BRAGA, Antônio de Pádua; LUDEMIR, Teresa Bernarda; CARVALHO, André P. L. Ferreira. **Redes Neurais Artificiais: Teoria e Aplicações**. Rio de Janeiro: LTC, 2000.

BRETANI, Helena Paula. **Autism spectrum disorders: an overview on diagnosis and treatment**. Revista Brasileira de Psiquiatria vol.35 supl.1, São Paulo, 2013.

CAMPOS, D. M. S. – **Psicologia da Aprendizagem**, 19º ed., Petrópolis: Vozes, 1986.

DEEP LEARNING BOOK - <http://deeplearningbook.com.br/o-neuronio-biologico-e-matematico/> acesso em: 25 nov. 2020

DIAS, Sandra. **Asperger e sua síndrome em 1944 e na atualidade**. Revista Latinoamericana de Psicopatologia Fundamental, [S. I.]: 2015.

DONVAN, John; ZUCKER, Caren. **Outra Sintonia: A história do autismo**. [S. I.]: Broadway Books. 2016.

DSM, **Diagnostic and Statistical Manual of Mental Disorders**, Manual Diagnóstico e Estatístico de Transtornos Mentais, American Psychological Association, 5 ed. Arlington, VA: American Psychiatric Association, 2013.

GONÇALVES, André Ricardo. **Redes Bayesianas**. Universidade Estadual de Londrina, 2008.

JUNIOR, Francisco Paiva. **Novos números dos EUA: 1 para 54**. Revista Autismo. São Paulo, ano VI, n. 4, p. 32, jun. 2020.

KERAS TEAM. Keras documentation: **About Keras**. **Keras.io**. Disponível em: <<https://keras.io/about/>>. Acesso em: 22 de Abril. 2021.

LONNOR NETO, Doriack Armando Mann; VALE, Silvana Franca; TEIXEIRA, Silvana Lobato Gomes; **Transtorno do Espectro Autista: Uma Contribuição Para Enfermagem na Atenção Básica** - Faculdade Paraense de Ensino, Belém, 2016.

LUBETSKY, Martin J et al. **Autism Spectrum Disorder**. Pittsburgh: Oxford University Press, 2011.

MONARD, Maria Carolina, BARANAUSKAS, José Augusto. Conceitos Sobre Aprendizado de Máquina. **Sistemas Inteligentes Fundamentos e Aplicações**. 1 ed. Cap. 5 Barueri-SP: Manole, 2003.

RESEARCHGATE, [https://www.researchgate.net/figure/Representacao-de-um-neuronio-artificial\\_fig3\\_264735169](https://www.researchgate.net/figure/Representacao-de-um-neuronio-artificial_fig3_264735169) acesso em: 25 nov. 2020

RUSSEL, Jonathan Russell; NORVING, Peter. **Inteligência Artificial**. 3. ed. Rio de Janeiro: Elsevier, 2013.

TENSORFLOW. **TensorFlow**. Disponível em: <<https://www.tensorflow.org/>>. Acesso em: 22 Abril. 2021.

VIEIRA, Neuza Maria; BALDIN, Sandra Rosa. **Diagnóstico e Intervenção de Indivíduos Com Transtorno do Espectro Autista**. Encontro Internacional de Formação de Professores e Fórum Permanente de Inovação Educacional. [S. I.]: 2017.