CENTRO UNIVERSITÁRIO DE JOÃO PESSOA - UNIPÊ
PRÓ-REITORIA ACADÊMICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

LUCAS FIGUEIREDO PEREIRA

UTILIZAÇÃO DE APRENDIZAGEM DE MÁQUINA APLICADA AO MAPEAMENTO DA INCIDÊNCIA DE CRIMES

JOÃO PESSOA – PB

2017

LUCAS FIGUEIREDO PEREIRA

UTILIZAÇÃO DE APRENDIZAGEM DE MÁQUINA APLICADA AO MAPEAMENTO DA INCIDÊNCIA DE CRIMES

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação do Centro Universitário de João Pessoa - UNIPÊ, como pré-requisito para a obtenção do grau de Bacharel em Ciência da Computação, sob orientação do Prof. Ms. Fábio Falcão de França

JOÃO PESSOA - PB

2017

LUCAS FIGUEIREDO PEREIRA

UTILIZAÇÃO DE APRENDIZAGEM DE MÁQUINA APLICADA AO MAPEAMENTO DA INCIDÊNCIA DE CRIMES

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação do Centro Universitário de João Pessoa - UNIPÊ, como pré-requisito para a obtenção do grau de Bacharel em Ciência da Computação, apreciada pela Banca Examinadora composta pelos seguintes membros:

Aprovada em 29/11/2017.

BANCA EXAMINADORA

_____

Prof. Ms. Fábio Falcão de França (UNIPÊ)

_____

Prof. Ms. André Luis de Lucena Torres (UNIPÊ)

_____

Prof. Ms. Ricardo Roberto de Lima (UNIPÊ)

DECLARAÇÃO

A empresa Secretaria de Segurança Pública do estado da Paraíba representada neste documento pelo Sr.(a) Pantaleão, Capitão da Polícia Militar, autoriza a divulgação das informações e dados coletados em sua organização, na elaboração do Trabalho de Conclusão de Curso intitulado Utilização de Aprendizagem de Máquina Aplicada ao Mapeamento Da Incidência de Crimes, realizados pelo aluno Lucas Figueiredo Pereira, do Curso de Bacharelado em Ciência da Computação do Centro Universitário de João Pessoa - UNIPÊ, com o objetivo de publicação e/ ou divulgação em veículos acadêmicos.

João Pessoa/PB, 29 de Novembro de 2017.

Luís Carlos Pantaleão de Sena

Capitão da Polícia Militar

Polícia Militar da Paraíba

A Deus, pela graça da vida.

AGRADECIMENTOS

A Deus, por sempre me presentear com muita saúde, alegria e muita força de vontade;

Aos meus pais Cleto Júnior e Sheila Pereira, meus irmãos Igor e Mariana por sempre estarem juntos comigo nos vários momentos da vida;

A todos os meus amigos da faculdade, Rodrigo Mouzinho, Flávio Henrique, Welton Mattos, Anderson Eraldo, Matheus Leão, Raphael Ribeiro e Fernando Petros, que se manteram comigo durante todas as épocas de alegrias, resenhas e aperreios da faculdade e toda jornada;

A todos os professores da faculdade na qual os admiro bastante, com quem aprendi, aprendo e sempre irei aprender em especial ao meu orientador e professor Fábio Falcão, que me apoiou e me ajudou durante o ciclo de vida deste trabalho.

Em especial minha vó Valdemira Pereira, com que proporcionou minha estadia aqui em João Pessoa e sempre me ajudou nas dificuldades da vida, na qual tenho um carinho especial.

A minha namorada Déborah, pessoa que aguentou meus stress e alterações de humor no decorrer de 5 anos até o momento, inclusive deste trabalho, sempre me apoiou e é uma pessoal na qual eu sempre quero estar por perto.

E as outras pessoas que me ajudaram diretamente ou indiretamente para eu estar aqui, como a Fábrica de Software do Unipê, colegas de trabalho da Media4ALL, PBPREV, Indra e Conductor, na qual aprendi muito na prática a computação propriamente dita.

E a todas ás críticas de todas as pessoas que passei nesses quatro anos de curso, na qual me fizeram buscar ser uma pessoa melhor.

RESUMO

Ao longo dos últimos anos a taxa de criminalidade vem crescendo de forma cada vez mais acelerada. De acordo com um levantamento estatístico realizado pela ONG *Seguridad, Justicia y Paz*, a cidade de João Pessoa, que está localizada na Paraíba, Brasil, encontra-se entre as 30 cidades mais violentas do mundo, estando em vigésimo nono lugar, com uma taxa anual de aproximadamente 47 homicídios a cada cem mil habitantes. Com o intuito de identificar e prever possíveis locais de crimes, o objetivo deste estudo foi criar um modelo preditivo por utilizando inteligência artificial, que através de ferramentas computacionais possam auxiliar o monitoramento da Secretaria de Segurança Pública do Estado da Paraíba em relação aos crimes do tipo furto, usando coleta de dados e algoritmos capazes de extrair características relevantes através de mineração de dados, usando um classificador. A primeira tarefa que foi realizada foi obtenção dos dados a serem analisados e a extração dos pontos mais relevantes, como as características sobre crimes nessas áreas, e dessa forma foi construído um modelo usando técnicas de aprendizado de máquina, que utilizou essas informações referentes a crimes passados, a fim de realizar o treinamento e teste do algoritmo. Após a elaboração do modelo, o mesmo obteve uma precisão de aproximadamente 82.28%, uma porcentagem bem interessante para a predição de novos crimes. Em conjunto com a construção deste modelo, foi desenvolvida uma aplicação Web para visualizar as probabilidades que o modelo preditivo previu de cada bairro, localizados em um mapa interativo da cidade de João Pessoa.

Palavras Chave: Inteligência artificial. Aprendizado de máquina. Mineração de dados. Modelos preditivos.

ABSTRACT

Over the last few years, the crime rate has been growing more and more rapidly. According to a statistical survey carried out by the NGO Seguridad, Justicia y Paz, a city of João Pessoa, which is located in Paraíba, Brazil, is among the 30 most violent cities in the world, being in 20th non-local, with a annual rate of approximately 47 homicides per 100,000 inhabitants. In order to identify and predict possible crime sites, the objective of this study was to create a predictive model for using artificial intelligence, which through computer tools aboard the control of the Public Security Secretariat of the State of Paraíba in relation to the crimes of the type Theft, using data collection and algorithms, methods of extracting relevant characteristics through data mining, using a classifier. Once the data were collected and extracted from the most relevant points, such as characteristics about crimes in these areas, and this form was constructed using machine learning techniques, which uses this information regarding crimes past, an end of training achievement and algorithm testing. After a model elaboration, it obtained an accuracy of approximately 82.28%, a very interesting percentage for the prediction of new crimes. In conjunction with a construction of this model, a Web application was developed to visualize as probabilities the predictive model of each neighborhood, located in an interactive map of the city of João Pessoa.

.

Key Words: Artificial Intelligence. Machine Learning. Data Mining. Predictive models.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ABNT – Associação Brasileira de Normas Técnicas

API – *Application Programming Interface*

ARFF – *Attribute Relation File Format*

CSS – *Cascading Style Sheets*

CSV – *Comma Separated Values*

IA – Inteligência artificial

IBGE – Instituto Brasileiro de Geografia e Estatística

JSAT – Java Statistical Analysis Tool

KDD – *Knowledge -discovery in databases*

ML – *Machine learning*

MVC – *Model View Controller*

ROC – *Receiver Operating Characteristic*

UML – *Unified Modeling Language*

WEB – *World Wide Web*

SUMÁRIO

1 MOTIVAÇÃO

O aprendizado de máquina vem ajudando o ser humano em diversas questões de pesquisa, sociais e de segurança, utilizando-se geralmente de previsões de determinados incidentes ou acontecimentos, com o intuito de intervenção de possíveis calamidades (BERK, 2012).

1.1 RELEVÂNCIA DO ESTUDO

Após levantamentos de dados estatísticos sobre as cinquenta cidades, com mais de 300 mil habitantes, possuindo as taxas mais elevadas de homicídios do mundo, verifica-se que João Pessoa, capital do Estado da Paraíba, é a vigésima nona localidade do mundo mais violenta, com uma taxa de homicídios em torno de 47 para cada 100 mil habitantes (SEGURIDAD JUSTICIA Y PAZ, 2017), conforme os dados exibidos na Tabela 01.

Tabela 01 - Lista de cidades mais violentas do mundo em 2016

Posição	Cidade	País	Homicídios	Habitantes	Taxa
1	Caracas	Venezuela	4,308	3,305,204	130,35
2	Acapulco	México	918	810,669	113,24
3	San Pedro Sula	Honduras	845	753,864	112,09
4	Distrito Central	Honduras	1,027	1,206,897	85,09
5	Victoria	México	293	346,029	84,67
6	Maturín	Venezuela	499	592,574	84,21
7	San Salvador	El Salvador	1,483	1,778,476	83,39
8	Ciudad Guayana	Venezuela	727	877,547	82,84
9	Valencia	Venezuela	1,124	1,560,586	72,02
10	Natal	Brasil	1,097	1,577,072	69,56
29	João Pessoa	Brasil	530	1,114,039	47,57

Fonte: Adaptado de Seguridad Justicia y Paz (2017).

A prevenção de crimes, principalmente os da classe organizados, é uma das principais ameaças à segurança pública, havendo certa preocupação com diversas capitais ao redor do mundo, visto que, os governos, usufruindo-se de ferramentas de inteligência artificial, poderiam trabalhar junto com a população, consolidando uma melhor forma de combate ao crime (CRAWFORD; EVANS, 2016).

Atualmente, a quantidade de seguranças e policiais, é bem menor do que deveria ser devido ao custo e aos recursos disponibilizados pelos governos. Segundo um levantamento feito pelo Instituto Brasileiro de Geografia e Estatística (IBGE), a Paraíba possui apenas um policial militar a cada 423 habitantes, como demonstrado na Figura 07.

Tabela 02 - Relação de policiais por habitantes nos estados brasileiros[1]

Unidades da Federação	Total	Homens	Mulheres	Índice de policiais/ hab. (1)
Distrito Federal	14.345	13.176	1.169	1:194
Amapá	3.700	2.946	754	1:199
Acre	2.712	2.441	271	1:286
Roraima	1.669	1.426	243	1:292
Rondônia	5.200	4.700	500	1:332
Rio de Janeiro	46.135	42.147	3.988	1:355
Rio Grande do Norte	8.926	8.717	209	1:378
Tocantins	3.855	3.384	471	1:383
Amazonas	9.050	7.970	1.080	1:421
Paraiba	9.263	8.563	700	1:423

Com o objetivo de melhorar a distribuição do efetivo da polícia militar, a primórdio, em João Pessoa (PB), a construção de um modelo preditivo que auxilie a um melhor mapeamento dos crimes na região, é de suma importância para maximizar a eficiência da segurança pública do estado.

Dessa forma, com o intuito de melhorar o combate ao crime, ferramentas computacionais podem ser utilizadas para auxiliar o mapeamento de crimes, assim como

[1] Fonte: https://exame.abril.com.br/brasil/brasil-tem-deficit-de-20-mil-policiais-em-seu-efetivo/

permitir um maior monitoramento pelos órgãos de segurança pública, através da utilização de informação extraída de dados pretéritos. Para isso, necessita-se da identificação e análise dos dados de crimes locais, com o auxílio dos órgãos de segurança pública do estado da Paraíba, para a extração de características de diversos tipos de delitos, construindo um modelo que utilizará técnicas de aprendizado de máquina a fim de permitir um possível policiamento preditivo em geral, tendo em mãos, locais suspeitos que poderiam ser utilizados como base na captura de infratores.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Propor a criação de uma solução, utilizando técnicas de aprendizagem de máquina, capaz de informar a probabilidade de incidência de delitos nos bairros da cidade de João Pessoa na Paraíba, com vistas a auxiliar os diversos órgãos de segurança.

1.2.2 Objetivos Específicos

- Coletar dados de crimes pretéritos através de um convênio com a Polícia Militar da Paraíba;
- Utilizar um algoritmo que seja capaz de extrair características relevantes para o aprendizado;
- Utilizar um classificador com base nos dados extraídos;
- Utilizar técnicas de agrupamento, classificação e regras de associação permitindo um afinamento do aprendizado;
- Validar os resultados expressos pelo modelo utilizando avaliadores de classificação, permitindo que o classificador tenha uma precisão capaz de auxiliar os trabalhos dos órgãos de segurança pública do Estado;
- Construir uma aplicação para representar os resultados obtidos e validados;

1.3 INDICAÇÃO DA METODOLOGIA

Do ponto de vista da natureza, a pesquisa pode ser descrita como Pesquisa experimental, na qual consiste em determinar algum módulo de estudo, escolher as variáveis que poderiam ter alguma influência, assim definindo as formas de gestão e observação dos efeitos que essas variáveis produzem no módulo (GIL, 2001).

Do ponto de vista da forma de abordagem, foi utilizada uma análise qualitativa pelo fato de ter ocorrido etapas de redução, categorização e interpretação dos dados (GIL, 2001).

Do ponto de vista dos objetivos, foi realizada uma pesquisa explicativa na qual é possível identificar fatores que influenciem determinada ocorrência de fenômenos como, por exemplo, a predição de crimes (GIL, 2001).

Do ponto de vista aos procedimentos técnicos, o processo de construção deste trabalho será feito utilizando a metodologia de pesquisa documental, manuseando-se fontes como tabelas estatísticas, relatórios, documentos informativos entre outros que não recebem tratamento analítico, ou que podem ser recriados com os dados da pesquisa (GIL, 2001).

1.4 METODOLOGIA

Os dados foram obtidos através das informações de crimes ou delitos preexistentes, ocorridos em alguns bairros da cidade de João Pessoa na Paraíba, por meio de um convênio de cooperação técnica com a polícia militar e o Centro Universitário de João Pessoa (Unipê).

Próprio autor.

O fluxo criado para desenvolvimento do trabalho demonstrado na Figura 02 inicia pela coleta dos dados, mineração dos dados, utilização de um algoritmo de aprendizagem de máquina e validação do modelo preditivo criado.

Figura 01 – Fluxo do trabalho executado

Fonte: Próprio autor (2017).

Foram obtidas 41689 ocorrências de crimes, resultando em uma amostra de 39260 ocorrências após o pré-processamento. A natureza Inicial, nome utilizado para identificar o tipo do crime, escolhida foi do tipo Furto, todos os tipos de furto, sejam eles furtos seguidos

de morte, intenções entre outros, ocorridos entre Maio de 2009 até Janeiro de 2017, pelo fato de ser um dos crimes com maior ocorrência segundo a polícia militar da Paraíba.

Como descrito na Figura 03, à coleta de dados foi feita junto com a equipe da Polícia Militar de João Pessoa, após a obtenção desses dados, foi feita uma análise, estruturação e pré-processamento dos dados.

Na etapa de pré-processamento dos dados obtidos, na qual foram removidos todos os atributos que não eram relevantes para o trabalho, como por exemplo: dados de outras cidades e dados em branco ou com erros de digitação e etc.

A extração de características nessa fase é extremamente importante para a validação do modelo, nessa parte é excluído a grande maioria de atributos desnecessários para a análise, entre eles, as características pessoais de criminosos, descrição das testemunhas entre outros, que podem causar certa inconsistência e "ruído" para o modelo, fazendo com que se distancie do foco do cenário desejado.

Os atributos disponíveis no banco de dados da Polícia Militar estão ilustrados na Figura 02.

Figura 02 – Atributos disponíveis pela Policia Militar da Paraíba

1) Registro
2) Data do Fato
3) Data do Registro
4) Natureza Inicial
5) Canal
6) AISP
7) QPP
8) OPM
9) Regional
10) Logradouro
11) Bairro
12) UF
13) Cidade
14) Localidade
15) Latitude
16) Longitude
17) Solicitante
18) Telefone
19) Nr. BO
20) Tempo Despacho
21) Tempo
22) Deslocamento
23) Tempo no Local
24) Tempo Total
25) Coordenado
26) Operador
27) Telefonista
28) Delegacia
29) Origem do Registro
30) Local do Fato
31) Natureza Fina
32) Ocorrido
33) Providencia
34) Status
35) QTD Envolvido
36) QTD Objeto
37) QTD Orgao
38) QTD Veiculo
39) QTD Arma Fogo Branca e Municao
40) QTD Viatura
41) QTD Registro Droga

Fonte: Próprio autor (2017).

A partir das informações adquiridas, uma análise foi feita com um intuito de refinar a massa de dados e colher padrões relevantes de crimes, aperfeiçoando-se os resultados para que estes possibilitem uma maior precisão na previsão de futuros delitos.

Foi utilizada a técnica feature *selection*, técnica de seleção de variáveis na qual procura melhorar o desempenho das previsões, utilizando menos capacidade computacional e

proporcionando uma melhor compreensão do modelo preditivo acerca dos dados (GUYON; ELISSEEFF, 2003).

Com o uso da técnica de seleção de variáveis, optou-se por duas variáveis principais:

- **Data do Registro (Data em que ocorreu o registro da ocorrência).**
- **Bairro (Bairro em que aconteceu a ocorrência).**

Foram escolhidas essas duas variáveis principais devido a grande quantidade de informação secundárias e importantes que poderiam ser originadas a partir destas duas variáveis como por exemplo, toda a questão temporal do atributo Data do Registro(Mês, Semana, Turno e etc), e várias possíveis características dos bairros, na variável Bairro(Comercial, Turístico, Residencial), que são informações relevantes na qual outros atributos da lista não teriam a princípio, como por exemplo o resto das variáveis como: Status, Número de Ocorrência, Solicitante, Cidade, Canal, Registro e etc.

Após a obtenção destas duas variáveis, aplicou-se outra técnica de seleção de variáveis, chamado de *feature extraction*, extração de recursos a partir de variáveis existentes, utilizando como principal objetivo descobrir um grupo de características mais eficazes para a classificação, removendo redundâncias e dados sem importância alguma, com isso o procedimento visa a maior quantidade de informações que minimizam as variabilidades ou mudanças repentinas de valores nas variáveis (MOHAMAD et al., 2015).

Com a técnica de extração de recursos, originaram-se mais quatro variáveis a partir da variável Data do registro, sendo elas:

- ✔ **Mês (Janeiro a Dezembro)**
- ✔ **Dia da Semana (Segunda a Domingo)**
- ✔ **Semana (Fim de Semana ou Semana)**
- ✔ **Turno (Manhã, Tarde, Noite, Madrugada).**

Como João Pessoa, possui muitos bairros e a quantidade de registros foi pouca em relação à quantidade de bairros, foram escolhidos vinte bairros com suas latitudes e longitudes respectivamente:

1. **Água Fria** [-7.1593344, -34.8555185]
2. **Altiplano** [-7.13649334, -34.83082294]
3. **Alto do Mateus** [-7.13784571, -34.90999525]
4. **Bairro dos Estados** [-7.11341284, -34.85496283]
5. **Bancários** [-7.1460895, -34.837581]

6. **Bessa** [-7.084319, -34.840162]
7. **Cabo Branco** [-7.1255451, -34.8251604]
8. **Centro** [-7.1195614, -34.8817282]
9. **Cristo Redentor** [-7.16404363, -34.87792253]
10. **Ernesto Geisel** [-7.179641, -34.8706935]
11. **Jaguaribe** [-7.1336742, -34.8762111]
12. **José Américo** [-7.1719679, -34.8541389]
13. **Manaíra** [-7.10438472, -34.83417034]
14. **Mandacaru** [-7.1042868, -34.861037]
15. **Mangabeira** [-7.1759102, -34.834821]
16. **Pedro Gondim** [-7.1145593, -34.8472401]
17. **Roger** [-7.110943, -34.8762111]
18. **Tambaú** [-7.11562725, -34.82653141]
19. **Torre** [-7.1260002, -34.8596574]
20. **Valentina** [-7.1997601, -34.8486199]

As latitudes e longitudes foram obtidas através da ferramenta da *Google*, o *Google Maps,* ferramenta em que permite a visualização de todas as coordenadas geográficas do planeta.

Após a escolha dos vinte bairros, foram cadastradas latitudes e longitudes dos bairros citados acima para rastreamento das coordenadas geográficas no mapa.

A estruturação e análise dos dados foram feitas através da ferramenta *Excel,* na qual se realizaram procedimentos para adequação dos dados e remoção de dados em branco e não correspondentes ao trabalho, resultando nos dados da Tabela 03.

Tabela 03 - Dados de crimes obtidos através da Policia Militar da Paraíba

Bairro	Mês	Dia da Semana	Semana	Turno	Residencial	Turistico	Comercial	Mês Festivo	Festa	Nivel de Evacuação	Região	Bairro Nobre	Densidade Demográfica	Zona
AGUA FRIA	Outubro	Sexta	Fim de Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Outubro	Domingo	Fim de Semana	Madrugada	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Outubro	Domingo	Fim de Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Outubro	Sexta	Fim de Semana	Manha	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Novembro	Sexta	Fim de Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Novembro	Segunda	Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Novembro	Quarta	Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Novembro	Quarta	Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Novembro	Quarta	Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Novembro	Sexta	Fim de Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Dezembro	Sexta	Fim de Semana	Tarde	Sim	Sim	Não	Não	Sim	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Abril	Sexta	Fim de Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Abril	Sabado	Fim de Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Maio	Sexta	Fim de Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Maio	Sexta	Fim de Semana	Manha	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Maio	Sexta	Fim de Semana	Manha	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Maio	Sabado	Fim de Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Maio	Domingo	Fim de Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Junho	Terca	Semana	Tarde	Sim	Sim	Não	Sim	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Junho	Domingo	Fim de Semana	Noite	Sim	Sim	Não	Sim	Sim	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Junho	Quarta	Semana	Noite	Sim	Sim	Não	Sim	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Junho	Sexta	Fim de Semana	Noite	Sim	Sim	Não	Sim	Sim	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Junho	Sabado	Fim de Semana	Noite	Sim	Sim	Não	Sim	Sim	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Julho	Domingo	Fim de Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Julho	Segunda	Semana	Manha	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Julho	Sabado	Fim de Semana	Manha	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Julho	Domingo	Fim de Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Julho	Domingo	Fim de Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Agosto	Quarta	Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Agosto	Quarta	Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Agosto	Sabado	Fim de Semana	Manha	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Agosto	Quarta	Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Setembro	Quarta	Semana	Madrugada	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Setembro	Quarta	Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Setembro	Quinta	Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Setembro	Segunda	Semana	Manha	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Setembro	Segunda	Semana	Noite	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Setembro	Sexta	Fim de Semana	Manha	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul
AGUA FRIA	Setembro	Terca	Semana	Tarde	Sim	Sim	Não	Não	Não	Alto	Centro	Não	Baixa	Sul

Fonte: Próprio autor (2017).

Além das quatro variáveis originadas a partir do procedimento de extração de recursos, foi utilizada também a técnica *N-grams*, técnica em que procura buscar correlação entre as variáveis através de variáveis de sequência, ou de variáveis que possam ser transformadas em correspondentes (MEIRELLES; FERNANDES; CASTRO, 2004) e também variáveis que também pudessem existir de alguma forma, uma forte relação.

A partir das variáveis selecionadas e extraídas anteriormente, foram originadas novas variáveis cuja intenção era aumentar a correlação das informações disponíveis até agora.

A partir disto, originaram-se mais dez variáveis, sendo elas:

- **Residencial (Representada por sendo um bairro é residencial ou não).**
- **Turístico (Representada por sendo um bairro é turístico ou não).**
- **Comercial (Representada por sendo um bairro é comercial ou não).**
- **Mês Festivo (Representada por sendo um mês em que ocorrem muitas festas ou não).**

- **Festa (Representada pela combinação das variáveis: Turno noite e madrugada em junção da variável Semana com o valor Final de semana em junção da variável Mês Festivo com valor Sim).**
- **Nível de Evacuação (Representada por sendo locais de acesso a vias e rodovias de fácil fuga ou difícil fuga).**
- **Região (Representada por sendo um bairro do sentido praia ou centro).**
- **Bairro Nobre (Representada por sendo um bairro é nobre ou não).**
- **Densidade Demográfica (Representada por sendo um bairro com um alto grau de moradores ou não).**
- **Zona (Representada pelos pontos cardeais: Norte, Sul, Leste e Oeste).**

Algumas variáveis acima como informações a respeito dos bairros, foram obtidas através de um balanço com o governo do Estado da Paraíba.

Com estas dez variáveis, somam-se quinze variáveis para serem utilizadas na construção do modelo preditivo utilizando os algoritmos de aprendizagem de máquina para criar correlações entre as variáveis.

1.5 ORGANIZAÇÃO DO TRABALHO

Após esse capítulo introdutório, o conteúdo deste trabalho organiza-se da seguinte forma:

- Capítulo 1 – MOTIVAÇÃO apresentará a contextualização do problema e os objetivos;
- Capítulo 2 – INTELIGÊNCIA apresentará quais teorias e respectivos autores mais contribuíram para a realização do estudo e as bases teóricas para a realização deste trabalho;
- Capítulo 3 – ARQUITETURA DA SOLUÇÃO apresentará o desenvolvimento da pesquisa juntamente com a solução proposta e sua aplicabilidade;
- Capítulo 4 – CONSIDERAÇÕES FINAIS apresentarão de forma conclusiva, respostas aos objetivos específicos propostos pelo trabalho, apresentando também limitações desta pesquisa e trabalhos futuros;

2 INTELIGÊNCIA

A inteligência pode ser considerada uma concepção fundamental no quesito psicologia moderna, na qual muitos se baseiam, embora seja escassa a quantidade de pessoas que conseguem ter uma definição concreta ou pelo menos amplamente convincente (SOBRAL, 2013 apud DALGALARRONDO, 2008).

A respeito disso, pode-se perceber que a palavra inteligência, derivada do latim *intelligare,* relativo a discernir, compreender, entender, não é algo simples, na qual existe uma definição global e definitiva.

Com o passar dos tempos, vários filósofos procuraram mecanizar a inteligência através de aprendizagem, visão e raciocínio, onde inúmeras tentativas ocorreram na ambição de se alcançar o entendimento do ser humano, em vigor das máquinas (GONZALES *et al.,* 2014).

De acordo com Nilsson (2010), George Boole demonstrou que exemplos do raciocínio lógico poderiam ser exemplificados a partir da manipulação de equações, consequentemente gerando proposições lógicas ou cálculos proposicionais, parte essencial no que diz respeito sobre a mecanização do raciocínio, principalmente na questão de sistemas equivalentes para manipulação e destas proposições.

Inteligência, em relação aos sistemas computacionais, poderia ser também descrita como a habilidade das máquinas de se alcançar metas em um determinado meio (MCCARTHY, 2007).

2.1 INTELIGÊNCIA ARTIFICIAL

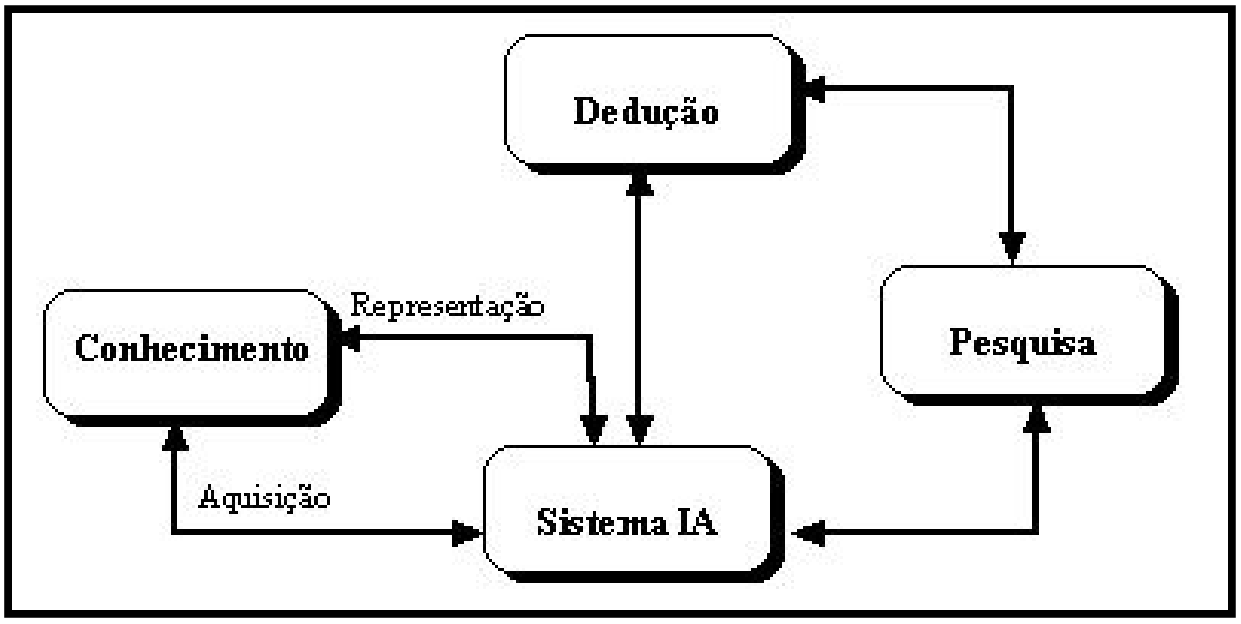
Esse processo de mecanização do conhecimento do ser humano deu-se o nome de Inteligência Artificial (IA), ou seja, permitir que sistemas, robôs ou seres irracionais, a princípio, pudessem ser capazes de analisarem possíveis situações, baseado em cenários na qual eram submetidos e chegarem a ter um parecer sobre cada uma, decidindo-se qual a próxima atitude a ser tomada.

A inteligência artificial é considerada como a capacidade de coletar conhecimento e razão sobre esse conhecimento para resolver problemas complexos, e ela já está desempenhando um papel crescente na pesquisa de ciência e gestão e áreas de pesquisa operacional (PANNU e TECH, 2015).

Em um futuro próximo, máquinas inteligentes substituirão humanos com as mais diversas capacidades em várias áreas, com isso a inteligência artificial pode ser resumida em estudo e desenvolvimento de máquinas inteligentes e software que pode raciocinar aprender, reunir conhecimento, comunicar, manipular e perceber objetos (PANNU e TECH, 2015).

A Figura 03 contém um fluxo do processo de mecanização do conhecimento.

Figura 03 - Uma visão conceitual dos sistemas de inteligência artificial



Fonte: Direne (2017).

Segundo Russel e Norvig (2010) existiam vários conceitos, estudos e abordagens e aplicações diferentes sobre a IA, áreas como a Matemática, Biologia, Filosofia, Engenharia, Psicologia entre outras (Figura 02). Historicamente, as abordagens relacionadas a IA, foram seguidas por diversas pessoas em quatro diferentes tipos:

- **Pensamento Humano**: Ações ou atividades automatizadas, interesse em criar raciocínio para as máquinas.
- **Pensamento Racional**: Ambição dos estudos de autorreflexão das máquinas.
- **Ação Humana**: Estudo de como induzir as máquinas a fazerem tarefas melhores que o ser humano.
- **Ação Racional**: Estudo do processo de criação de agentes inteligentes.

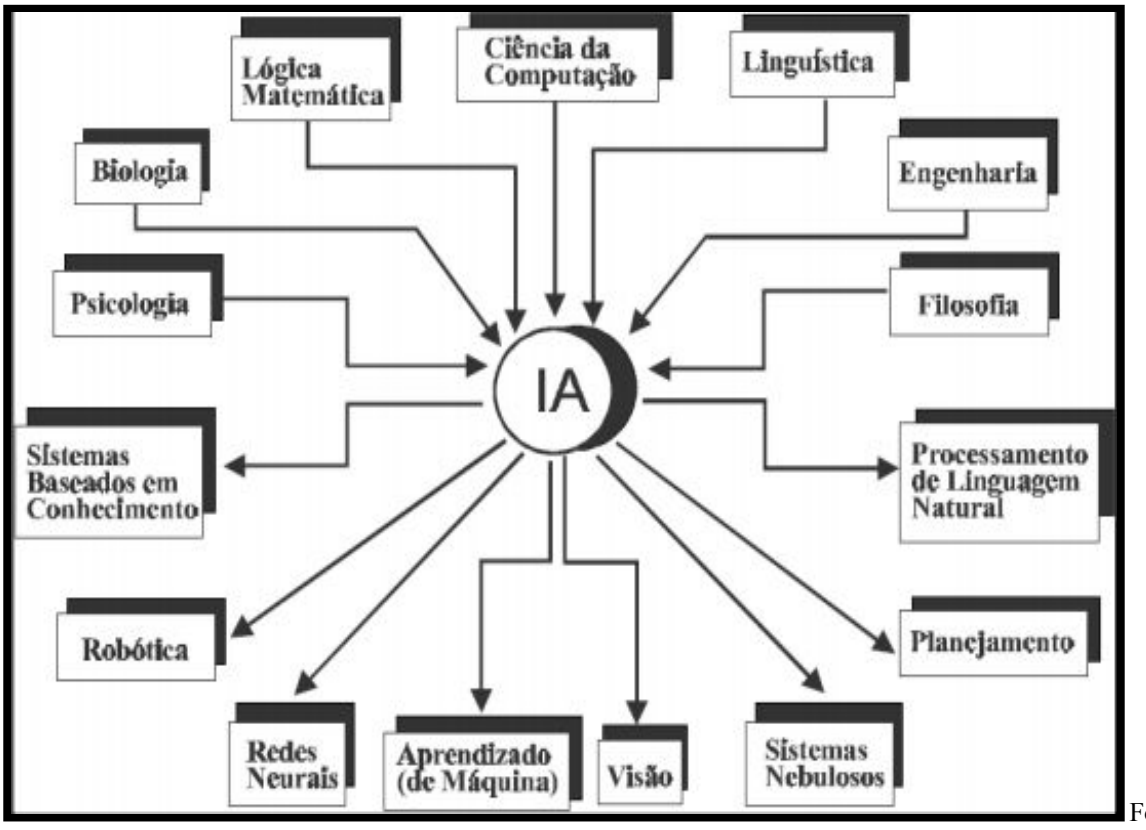
A inteligência é obtida através de transformações entre experiências e aquisições de novos conhecimentos, por outro lado o conhecimento pode ser interpretado como a informação baseada em experiências, habilidades e competências de cada pessoa (SANTOS e CARVALHO, 2008).

No que diz respeito ao conceito computacional, a definição satisfatória de inteligência, baseado pelo Teste de Turing, feito pelo famoso cientista renomado Alan Turing, acreditava que era muito mais importante estudar os princípios subjacentes da inteligência do duplicar

um exemplar (RUSSEL e NORVIG, 2010). Para isso, formulou que para uma máquina ser realmente inteligente, ela precisava das seguintes disciplinas:

- **Linguagem natural**: compreensão da língua dos humanos.
- **Representação do conhecimento**: armazenar e expressar conhecimento.
- **Raciocínio automatizado:** utilizar esse conhecimento respostas e próprias conclusões.
- **Aprendizagem de máquina:** adaptação às novas circunstâncias e predição de possíveis padrões
- **Visão computacional:** percepção acentuada de objetos.
- **Robótica** para a manipulação de objetos o que fazer a respeito.

Figura 04 – Áreas relacionadas com inteligência artificial



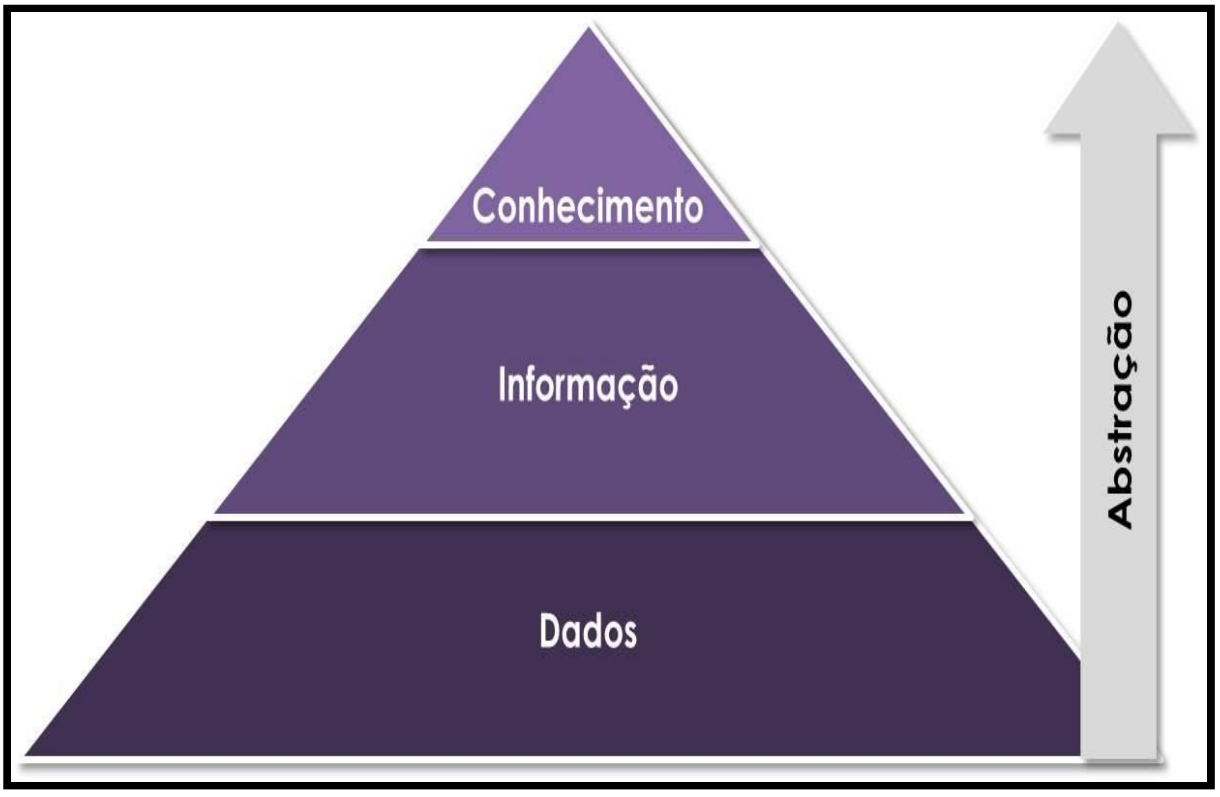
Fonte: Monard e Baranaukas (2000).

Na Figura 04, é possível visualizar a quantidade de áreas conectadas á IA.

2.2 MINERAÇÃO DE DADOS

O conhecimento pode ser descrito como o processo de entendimento e aplicabilidade das informações adquiridas pelo estudo ou pela experiência. Logo o conhecimento se remete ao conceito da palavra informação que pode ter o significado bastante subjetivo, mas simplesmente definido como um dado analisado e com algum significado (Figura 05). Dado seria a informação em seu estágio bruto sem nenhum processo de refinamento ou estruturação, na qual, pode estar em diferentes formatos como os presentes em redes de comunicação e sinais, ou seja, analógico e digital, os dados eletrônicos ou não eletrônicos, entre outros. Sobre o vasto mundo dos dados e sua grande diversificação, surgiu a expressão *Data Science*, a Ciência dos Dados, que procura analisar, organizar, normalizar e sistematizar esse conhecimento acerca dos dados e seu ciclo de vida (SILVEIRA, 2016).

Figura 05 – Processo de transformação dos dados[2]

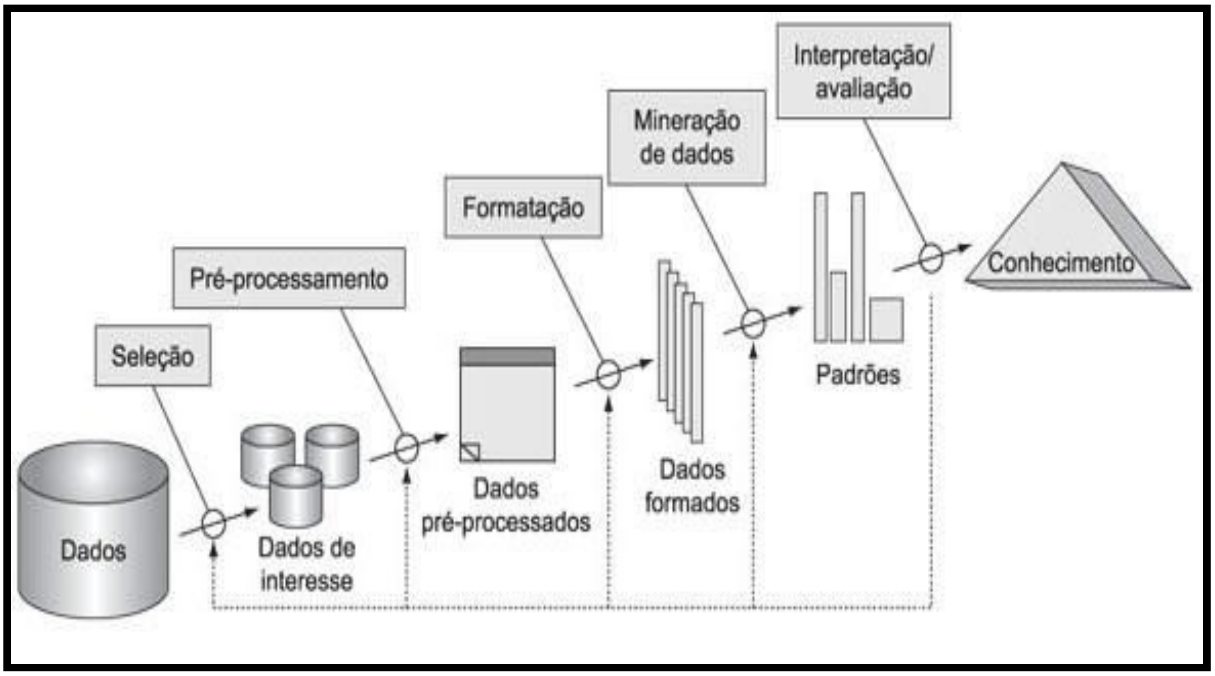


Atualmente, com o grande volume de dados e sua gama de informações sobre inúmeros campos de interesses do mundo contemporâneo, formou-se a *Data Mining*, Mineração de Dados que procura explorar e analisar por meio automático ou semiautomático esse enorme gama de dados com o propósito de descobrir padrões ou métricas significativas (BERRY e LINOFF, 2011).

[2] Fonte: http://www.vortice.inf.br/noticia/http-blog-prgbrasil-com-2015-06-19-o-que-e-bi-preview-id222

De acordo com Fayyad *et al*. (1996), o processo de descoberta de conhecimento em banco de dados, *knowledge-discovery in databases* (*KDD*), pode ser classificado como um processo não trivial para a descoberta de padrões relevantes e úteis para determinados nichos de estudo. O Processo *KDD* pode ser exemplificado como um conjunto de atividades contínuas, listados por: A obtenção dos dados, a seleção de características relevantes, o pré-processamento em que se aplicam várias técnicas para captação e preparação dos dados, a formatação e organização dos dados, a mineração dos dados propriamente dita, e seguindo com a avaliação dos resultados obtidos. Segue uma demonstração do processo na Figura 06.

Figura 06 – Etapas do processo *KDD*



Fonte: Fayyad et al. (1996).

Mineração de dados está relacionada com a solução de problemas, tanto seja na área de negócios, para aperfeiçoar padrões de venda utilizando organizações de prateleiras de produtos em lojas como supermercados e atacadões, como na área da saúde, na possível relação de incidências de doenças em um determinado grupo de pessoas (WITTEN *et al*., 2017).

2.3 APRENDIZAGEM DE MÁQUINA

A aplicação de técnicas computacionais em relação a padrões ocultos dos dados pode ser chamada de aprendizado de máquina ou *Machine Learning*, que é um subcampo ou área de conhecimento da inteligência artificial na qual procuram algoritmos que buscam

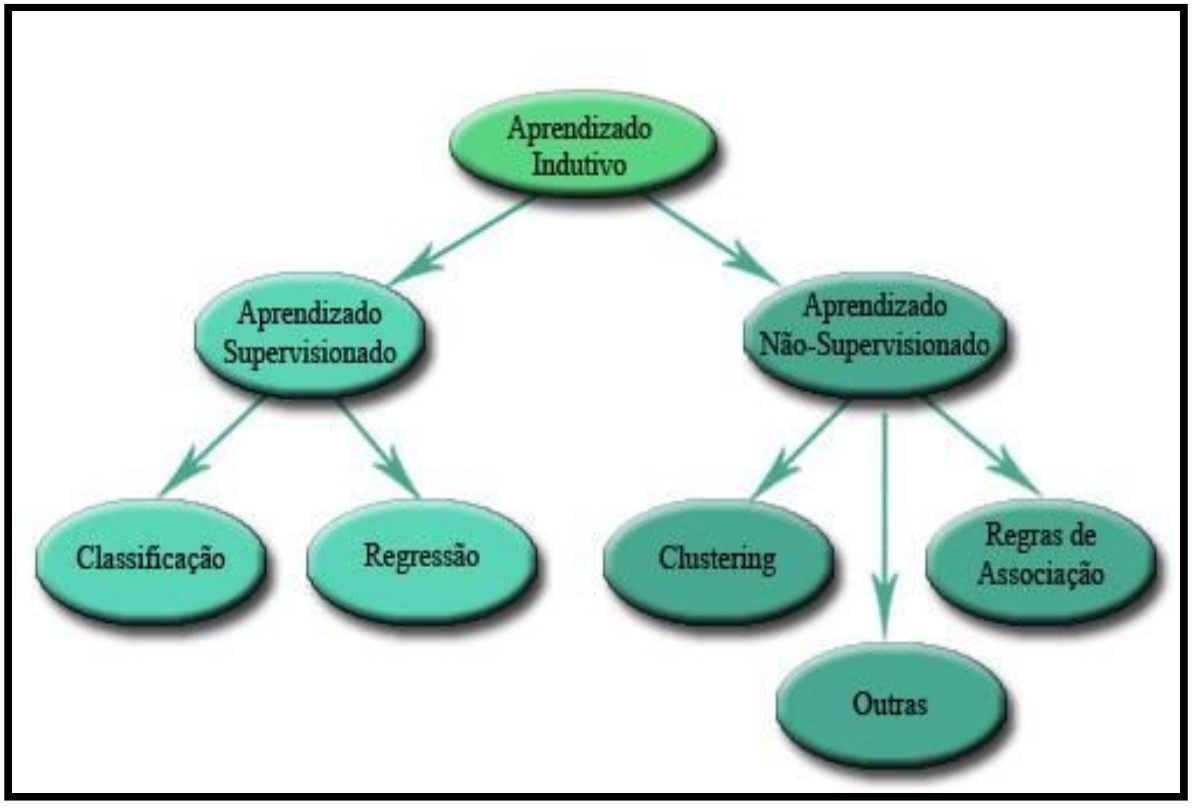
reconhecer semelhanças em dados. Oculto, pois não se observam explicitamente os padrões nos dados (SILVEIRA, 2016).

Tem como objetivo o desenvolvimento de técnicas capazes de lecionar ou ensinar a máquina a aprender ou desempenhar determinada atividade cada vez melhor baseado em suas experiências antecessoras (SANTOS, 2016).

Essas técnicas agregam um princípio de inferência, no qual, é denominado por indução, em que se possibilita a obtenção de conclusões genéricas em um determinado grupo de exemplos. Na indução, uma característica é aprendida utilizando-se inferência indutiva sobre determinadas amostras. Por isso, as hipóteses geradas a partir deste meio, podem ou não ser verídicas (REZENDE, 2013 *apud* SCHIMITT *et al.*, 2003).

O aprendizado indutivo pode ser subdividido em dois tipos: supervisionado e o não supervisionado, como demonstrado na Figura 07.

Figura 07 – Hierarquia do aprendizado indutivo



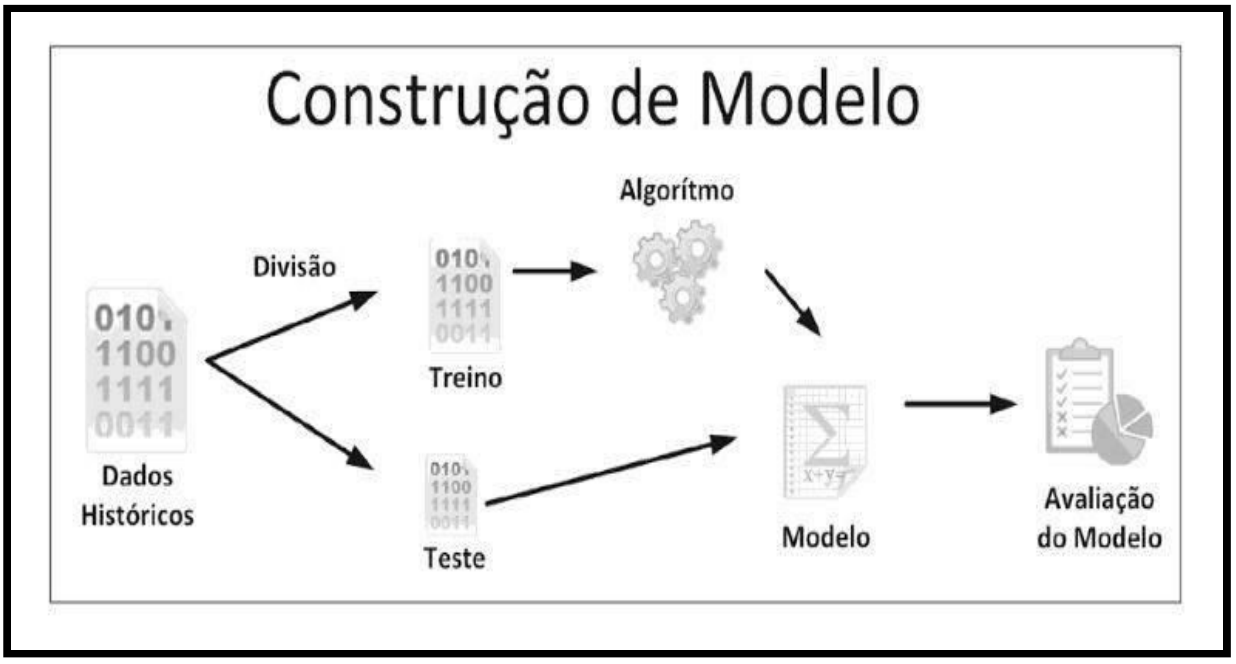
Fonte: Rezende et al., (2003).

O aprendizado supervisionado é aquele que compreende o relacionamento entre diferentes tipos de atributos e classes e com isso, ser capaz de fazer possíveis predições

baseadas na classe escolhida, e o aprendizado não supervisionado, ou seja, sem nenhum valor de conceito ou alvo associado. O atributo seria qualquer característica ou aspecto presente no exemplo, podendo ser qualitativo, como cores e quantitativo, como altura. Já a classe seria um atributo especial ou de interesse, no qual seria estudado ou analisado durante as predições (SCHIMITT, 2013).

Se a existência da classe ou atributo especial, cujo se pretende descrever ou prever, for qualitativa ou nominal, utiliza-se a classificação, caso seja quantitativa ou numérica, opta-se pela regressão. Na classificação, os dados a serem analisados são chamados de dados históricos, fatos ocorridos que já obtiveram uma classificação anterior. Quando o modelo, que é uma espécie de abstração do mundo real, estiver concluído, estes dados históricos já não serão mais necessários, pois este modelo será capaz de fazer predições acerca de futuros dados inseridos na Figura 08 (SILVEIRA, 2016).

Figura 08 – Construção do modelo de aprendizado de máquina



Fonte: Silveira (2016).

Os valores estimados de classificação e probabilidade do atributo principal ou de classe tentam prever para cada indivíduo na amostra de dados, que á um pequeno nicho de classes, este individuo pertence. Na maioria dos casos, as classes são reciprocamente exclusivas. Uma exemplificação de uma atividade do tipo de classificação seria a determinação se um indivíduo pertence ou não a uma classe específica, dada por uma

estimativa ou pontuação baseada na probabilidade do atributo principal (PROVOST; FAWCETT, 2013).

De acordo com os mesmos autores essa pontuação e a classificação estão intimamente relacionadas, pelo fato da pontuação representar a probabilidade de que esse indivíduo pertença a cada classe.

O algoritmo escolhido poderá ser a peça chave depois de ter ocorrido toda a análise e pré-processamento dos dados, ele irá aprender com os dados históricos, e a aplicação do algoritmo nos dados, resulta em um modelo preditivo, que pode ser chamado como uma espécie de fórmula criada para prever novos dados que venham a ser apresentados para o algoritmo, no qual ainda não tenham passado pelo mesmo (AMARAL, 2016).

Um dos algoritmos muito conhecido é o famoso algoritmo *Naive Bayes* muito utilizado para classificação, que é lembrado pelo teorema de *Bayes*, criado com o presumo que todas as variáveis X, ou seja, atributos a serem classificados, são mutualmente independentes, dada uma variável qualquer (LOWD e DOMINGOS, 2005).

As estatísticas também são muito utilizadas no âmbito dos algoritmos, principalmente nos modelos de regressão, sejam elas para descrição ou inferência de dados.

A regressão é semelhante à classificação, porém o atributo principal ou atributo classe na qual se procurar realizar a previsão é um valor numérico, para isso utiliza-se uma medida estatística chamada correlação, que indica a força de relação que existe entre as variáveis numéricas, sempre sendo um valor real de menos um a um, quanto mais próximo de um, mais forte será a ligação entre as variáveis (AMARAL, 2016).

Na regressão linear este resultado é contínuo, já na regressão logística o resultado é dicotômico, ou seja, binário. No modelo de regressão logística multivariável, aplicação da regressão logística seria em múltiplas variáveis com o objetivo de encontrar e avaliar a relação e correlação entre várias variáveis binárias (HIDALGO e GOODMAN, 2013).

Exemplificando através de um caso de uso, para verificação do clima em uma partida de tênis, utilizando os atributos: Aspecto do céu, temperatura, Humidade e vento.

A numeração dos dias está representada na primeira coluna, junto com as características dos aspectos do céu como: Sol, Nuvens ou Chuva, logo após a coluna

temperatura com os valores: Quente, Ameno e Fresco, em seguida o campo Humidade que leva em consideração o nível de Humidade: Fraco ou Forte, na sequência vem o campo Vento, na qual mostra o nível de intensidade do vento: Fraco ou Forte e na coluna final é mostrado se baseado nos valores deste conjunto de variáveis, se é possível jogar Tênis ou não. A Figura 09 representa os campos e dados mencionados em um formato de tabela, possuindo os campos dispostos lado a lado.

Figura 09 – Representação abstrata de atributos decisivos para jogar uma partida de tênis[3]

Exemplos de Treino

Dia	Aspecto	Temp.	Humidade	Vento	Jogar Ténis
D1	Sol	Quente	Elevada	Fraco	Não
D2	Sol	Quente	Elevada	Forte	Não
D3	Nuvens	Quente	Elevada	Fraco	Sim
D4	Chuva	Ameno	Elevada	Fraco	Sim
D5	Chuva	Fresco	Normal	Fraco	Sim
D6	Chuva	Fresco	Normal	Forte	Não
D7	Nuvens	Fresco	Normal	Fraco	Sim
D8	Sol	Ameno	Elevada	Fraco	Não
D9	Sol	Fresco	Normal	Fraco	Sim
D10	Chuva	Ameno	Normal	Forte	Sim
D11	Sol	Ameno	Normal	Forte	Sim
D12	Nuvens	Ameno	Elevada	Forte	Sim
D13	Nuvens	Quente	Normal	Fraco	Sim
D14	Chuva	Ameno	Elevada	Forte	Não

Outra maneira alternativa dos modelos estatísticos é o processo de construção de tabelas ou árvores de decisões a partir de dados históricos para serem utilizados tanto para a classificação quanto para regressão. A árvore é representada por um gráfico acíclico, com uma raiz fixa, no qual raiz seria o atributo principal ou classe a ser previsto (HAGENLOCHER, 2017).

Com os atributos da Figura 09 é possível construir a seguinte árvore de decisão, ilustrado na Figura 10:

[3] Fonte: http://web.tecnico.ulisboa.pt/ana.freitas/bioinformatics.ath.cx/bioinformatics.ath.cx/indexf23d.html?id

Figura 10 – Representação da árvore de decisão para jogar uma partida tênis [4]

O atributo principal ou variável classe neste caso seria o aspecto do céu, na qual ele está sendo representado como raiz da árvore na Figura 10, cujo valor pode ser: Sol, Nuvens ou Chuva. Baseado neste valor, cada nó proveniente de um destes valores, testa um atributo, caso a condição seja satisfeita, verifica-se o ramo correspondendo ao valor do atributo, a qual o nó (Humidade) foi testado anteriormente, e logo após atribui-se uma classificação as folhas, que seria o valor mais provável de ocorrer, o fluxo dito anteriormente é demonstrado na Figura 11.

Figura 11 – Representação do fluxo de operações de uma árvore de decisão para jogar a partida de tênis [5]

Entretanto, as árvores de decisão podem não generalizar de uma boa forma para o classificador, para isso existe as florestas aleatórias, que buscam reduzir este problema da generalização e possivelmente evitar o *overfitting* ou sobreajuste, que é quando o modelo tem

[4] Fonte: http://web.tecnico.ulisboa.pt/ana.freitas/bioinformatics.ath.cx/bioinformatics.ath.cx/indexf23d.html?id

[5] Fonte: http://web.tecnico.ulisboa.pt/ana.freitas/bioinformatics.ath.cx/bioinformatics.ath.cx/indexf23d.html?id

uma boa confiabilidade baseado nos dados históricos, mas falha quando tende a prever futuros resultados (BIAU, 2012).

2.4 AVALIADORES DE CLASSIFICAÇÃO

Os avaliadores de classificação são métricas ou técnicas que procuram atestar a confiabilidade dos modelos preditivos.

A *Receiver Operating Characteristic* Curve (*ROC*), ou seja, curva de característica de operação do receptor é uma técnica de visualização e organização para selecionar classificadores tendo em vista seu desempenho (FAWCEUT, 2005).

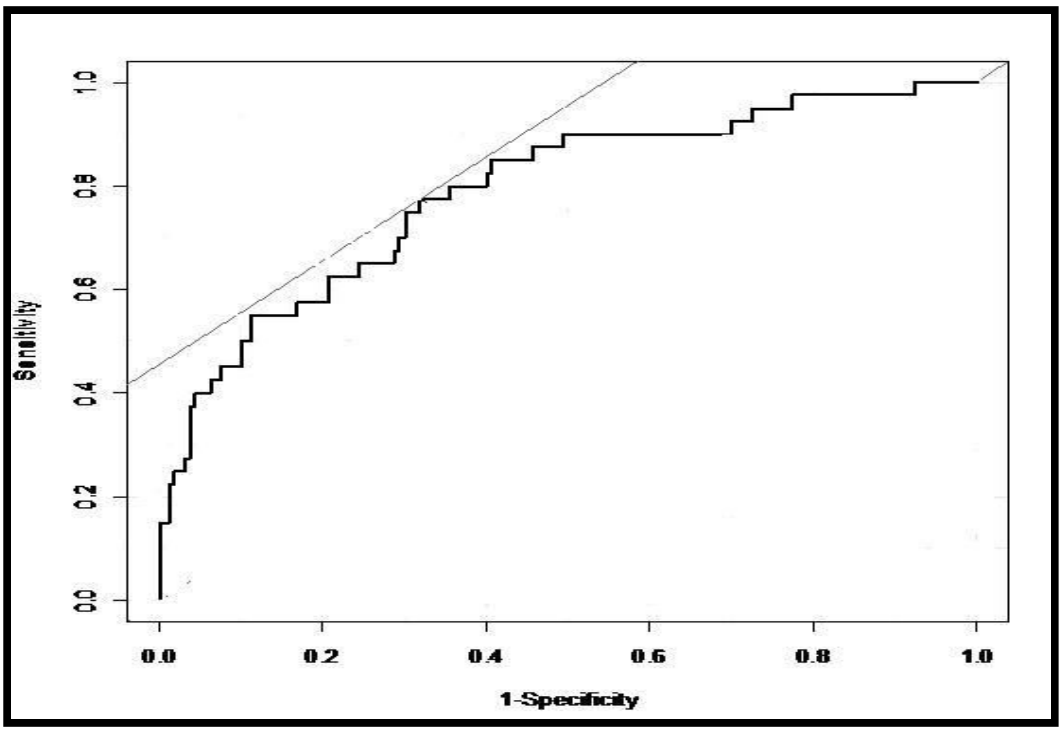
A curva *ROC* pode ser representada por uma ilustração de uma curva em um gráfico de sensibilidade por especificidade, baseado na escolha de um ponto de corte que será avaliado nos valores entre zero e um.

O ponto de corte deve ser feito com a relação sensibilidade e 1-especificidade, quanto mais se aproximar do canto superior esquerdo, melhor a taxa de probabilidades de acerto do modelo preditivo avaliado (FAWCEUT, 2005).

A Figura 12 mostra um exemplo de gráfico em relação à sensibilidade e 1-especificidade para cálculo do ponto de corte.

Figura 12 – Gráfico de Sensibilidade por 1-Especificidade[6]

[6] Fonte: http://www.portalaction.com.br/analise-de-regressao/45-predicao



No eixo Y do gráfico, representa os índices de verdadeiros positivos, índice alvo, e no eixo X, representa os índices de falsos positivos, índice descartável.

- Verdadeiros Positivos: Crimes que aconteceram e o classificador também previu que iriam acontecer.
- Verdadeiros Negativos: Crimes que aconteceram e o classificador não previu que iriam acontecer.
- Falsos Positivos: Crimes que não aconteceram e o classificador previu que iriam acontecer.
- Falsos Negativos: Crimes que não aconteceram e o classificador também previu que não iriam acontecer.

Dos índices listados acima, os mais interessantes para calcular a ponte de corte da curva *ROC*, são os Verdadeiros Positivos e Verdadeiros Negativos, ou seja, dados em que o classificador acertou nos palpites de crimes que aconteceram e de crimes que não aconteceram (VISA, RAMSAY, et al., 2011).

A precisão também é uma métrica calculada utilizando os valores positivos e negativos dividindo pela proporção de eventos que ocorreram e não ocorreram.

Aplicando os valores ditos acima em fórmulas, ficaria da seguinte forma:

- **Sensibilidade = Verdadeiros Positivos / (Verdadeiros Positivos + Falsos Negativos).**

- **Especificidade = Verdadeiros Negativos / (Verdadeiros Negativos + Falsos Positivos).**
- **Precisão = (Valores Positivos + Verdadeiros Negativos) / (Eventos que ocorreram + Eventos que não ocorreram).**

A matriz de confusão (Figura 13) é utilizada para a verificação do desempenho do classificador ou regressor utilizado, baseado nos valores preditos (SOKOLOVA; LAPALME, 2009).

Figura 13 – Ilustração da Matriz de Confusão[7]

Todos os palpites corretos estarão localizados na diagonal principal da matriz, ou seja, Valores Verdadeiros positivos, e valores verdadeiros negativos e os palpites errados estão localizados na diagonal secundária da matriz, ou seja, Valores Falsos Positivos e Valores Falsos Negativos.

Outro avaliador usado frequentemente para testar a confiabilidade e precisão de modelos preditivos é o índice *Cohen's Kappa*, índice de análise de concordância entre as variáveis possuindo valores de -1 a 1, baseado no grau de precisão do modelo (MCHUGH, 2012).

A Tabela abaixo demonstra o valor do índice *Kappa* (K) em relação ao grau de concordância do modelo preditivo.

[7] http://developerdeveloper.blogspot.com.br/2013/11/matriz-confusao.html

Tabela 04 - Relação entre o índice *Kappa* e a concordância dos dados[8]

Valor Índice *Kappa* (*K*)	Concordância
0	Pobre
0 a 0,20	Ligeira
0,21 a 0,40	Considerável
0,41 a 0,60	Moderada
0,61 a 0,80	Substancial
0,81 a 1	Excelente

Qualquer valor do índice K menor que 0.6, indica uma concordância inadequada para a grande maioria dos problemas do mundo real, como análise de dados críticas, como de pacientes com algum problema de saúde (MCHUGH, 2012).

A união dos de valores abaixo da curva *ROC*, junto com os valores da diagonal principal da matriz de confusão em conjunto com o índice *Kappa*, formam uma poderosa métrica na avaliação de um modelo preditivo.

2.5 TÉCNICAS PARA TREINO DE ALGORITMOS

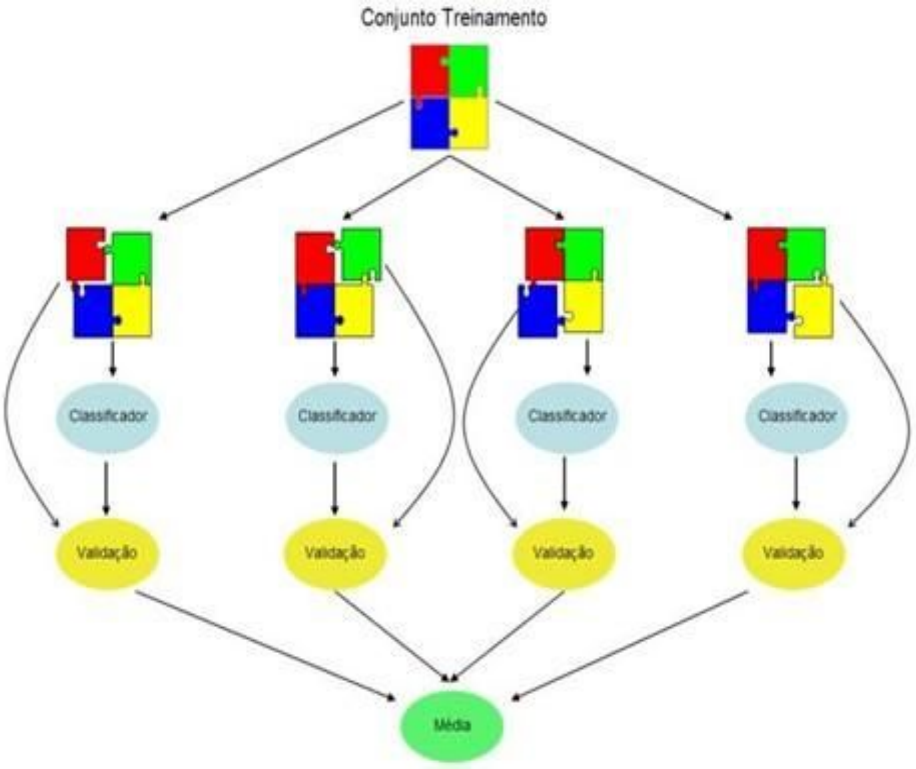
Grande parte da maioria das técnicas de classificação que existem procuram tentar aprender como uma única unidade similar, apesar da grande quantidade de atributos complexos que se diferenciam um dos outros, mas envolvidos. Um modo muito comum é a divisão dos dados em porcentagens de dados de treinos e testes (BATTULA; PRASAD, 2013).

Uma estratégia muito popular para o treino de algoritmos de aprendizagem de máquina é o *Cross-Validation*, validação cruzada, cujo principal objetivo, é dividir os dados em múltiplas vezes para calcular o risco de cada possível evento. Parte dos dados é utilizada para treinar o algoritmo e a parte que resta é utilizada para testar e estimar o risco do algoritmo. (ARLOT; CELISSE, 2009).

A execução da validação cruzada, ele divide em várias partes, que são chamadas de *folds,* páginas, na qual essas páginas representam uma porção dos dados.

[8] Fonte: http://principo.org/estatstica-computacional-uso-do-spss-statistical-package-for-t.html?page=4

Figura 14 - Demonstração de uma validação cruzada utilizando quatro páginas



Após essa divisão, é submetido ao classificador cada uma dessas divisões para a validação individual de cada página dividida anteriormente. No final é feita uma média de todos os valores que foram obtidos através da validação das porções de dados. O fluxo é demonstrado na Figura 14 na qual o conjunto de dados foi dividido em quatro páginas.

3 ARQUITETURA DA SOLUÇÃO

Esta seção apresenta a arquitetura do sistema que foi desenvolvido com objetivo de auxiliar o monitoramento da Secretaria de Segurança Pública do estado (SSP) na cidade de João Pessoa, tentando fornecer informações de probabilidades dos crimes de Furto nos bairros da região. Sendo apresentado a aplicação e o modelo preditivo criado que foi usado como "combustível" para a construção da interface.

3.1 TECNOLOGIAS UTILIZADAS

A seleção das variáveis foi realizada, através da extração das classes desejadas na base de dados utilizando o formato *Comma Separated Value*s (*CSV*), formato de agrupamento de informações separado por vírgulas.

Após a obtenção do arquivo *CSV*, o mesmo foi repassado para um conversor (*Parser)*, no qual é uma espécie de Analisador de estrutura para a conversão desse tipo de arquivo em um formato desejado, e também refinações de tipos dos atributos recebidos.

O formato desejado seria o *Attribute-Relation File Format* (*ARFF*), que é um formato de arquivo mais conhecido por conter regras de associações e muito utilizado para a classificação ou regressão.

Para a classificação dos dados obtidos, adaptados e convertidos para o tipo de arquivo *ARFF*, obtenção da precisão e índice *Kappa* K, foi feita a análise e classificação desse arquivo, por intermédio de uma *library,* que é um conjunto de recursos disponibilizados para um determinado propósito, que nesse caso é o aprendizado de máquina.

O nome da *library* é o *Java Statistical Analysis Tool* (*JSAT*) atualmente na versão 0.0.8, escrito puramente em Java, a linguagem de programação mais utilizada nos últimos tempos (TIOBE, 2017).

O JSAT contém inúmeras implementações de algoritmos de análise e estatística. Os algoritmos são uma sequência de passos para solucionar um determinado problema, que nesse caso, seria a criação do modelo preditivo.

O desempenho do *JSAT* é bem superior comparado a seus concorrentes, como o mais famoso *Waikato Environment for Knowledge Analysis (WEKA), BudgetedSVM, LIBLINEAR,* enquanto utiliza o padrão de programação orientado a objetos, além de permitir a computação concorrente e paralela, que utilizam melhor o poder de processamento do processador ou

GPU (*Graphics Processing Unit*) que é a unidade de processamento gráfica da máquina para usufruir de performance e paralelismo (RAFF, 2017).

A Figura 15 ilustra o desempenho da ferramenta *JSAT* em comparação a seus concorrentes baseado no tempo de processamento de alguns dos principais algoritmos.

Figura 15 – Desempenho do JSAT em segundos, na comparação com outras ferramentas

	Weka						*LIBLINEAR*		*BudgetedSVM*	
	Platt SMO	C45	RF	1-NN	LR	Lloyd's k-means	SVM by DCD	newGLMNET	AMM	BSGD
Other Time	7904	303	143	2537	3301	1011	161	14.5	59.5	160
JSAT Time	973	139	125	691	914	36	71	29.2	9.7	64
Other Error	1.55%	11.1%	3.26%	3.09%	8.21%	—	8.30%	8.35%	37.2%	21.8%
JSAT Error	1.56%	11.5%	4.19%	3.09%	7.76%	—	9.21%	8.53%	5.02%	10.5%

Fonte: Raff (2017).

Para obtenção da área da curva *ROC*, área que representada pela relação das taxas de verdadeiros positivos e falsos positivos, foi utilizado o *Weka,* versão 3.8.1, ferramenta para mineração de dados possuindo uma interface amigável e simples.

Como framework de aplicação web em Java, utilizou-se o *Spring Boot* versão 1.5.3 *RELEASE*, framework do ecossistema Spring, que são um conjunto de *API's,* que são interfaces de programação de aplicações, na qual procuram facilitar o desenvolvimento provendo, métodos e objetos já prontos e dispostos para utilizar junto com o *JDK*, *Java Development Kit*, na qual foi utilizada a versão 8.0.

Alguns itens da interface foram utilizados dois *frameworks,* ferramentas de desenvolvimento, famosos para desenvolvimento de interfaces como o *Bootstrap* 3.3.5 e o *Cascading Style Sheets* (*CSS*) versão 3.0.3, linguagem de comunicação visual como alteração de cor, fontes e etc nas interfaces.

Como SGBD, Sistema gerenciador de banco de dados, software onde são persistidos os dados ou objetos desejados, utilizou-se o *PostgreSQL* versão 9.4-1201, programa gratuito fornecendo como principal interface o *PGADMIN* versão 3, para visualização e manipulação dos dados persistidos.

Para o gerenciamento de dependências e ferramenta de compilação, utilizou-se o *Maven* versão 3.5.2, ferramenta da *Apache* na qual organiza, faz o *download* automático de todas as dependências necessárias apenas colocando o nome da dependência e versão no seu arquivo principal de configuração, o *pom.xml*.

Para servidor de aplicação, para disponibilizar um ambiente para aplicação web, e gerenciamento da aplicação, utilizou-se o Tomcat versão 8.5.6, ferramenta também da empresa *Apache*.

A aplicação foi *versionada*, ou seja, armazenar diferentes versões em um determinado local ou repositório, utilizando o GitLab ferramenta para versionamento nas nuvens, assim tornando seguro e fácil a integração contínua. Os *commits*, processo para realizar as mudanças no código-fonte e artefatos, foram realizados através do *Git* que é um sistema de controle de versão para gerenciamento do código-fonte e outros artefatos.

O nome utilizado para o projeto foi o *CrimePrediction*, que significa predição de crimes e atualmente está na versão 1.2.

Na Figura 16, mostra uma demonstração do *Changelog,* arquivo no qual contém as alterações realizadas no projeto e qual versão e projeto se encontra.

Figura 16 – Demonstração de uma parte do arquivo de mudanças da aplicação

```
#CrimePrediction changelog information :)

##Versao 1.2
Corrigido bug na geolocalizacao dos bairros;
Adicionado AlgoritmoService;
Adicionado probabilidade da classe alvo e taxa de acertos nos classificadores;
Ajustado classes e pacotes;
Ajustado AlgoritmosBatch para utilizar taxa de acerto para escolher melhor Algoritmo;
Removido algumas classes de apoio;

##Versao 1.1
Adicionado geolocalizacao para os principais bairros de Joao Pessoa;
Modificado CrimeController;
Adicionado atributo cor na classe Bairro;

##Versao 1.0
Adicionado conexao com SGBD PostgresSQL;
Adicionado classe EntidadeAbstrata;
Adicionado novo algoritmo FlorestaRandomica(batch,testes);
Corrigido alguns bugs com o Controle de Sessao, css e javascript;
Refatorado e revisado classes dos pacotes de modelo e algoritmos;

##Versao 0.9
Adicionado pacotes: Service, Repository, Config, Validators;
Adicionado SpringSecurity e SpringData no projeto;
Adicionado controle de Autenticacao, permissao, Usuario e criacao de conta;
Adicionado Controlador para o controle de Autenticacao;
Adicionado Arquivo properties de validacao;
Modificado alguns nomes de classes para melhor entendimento;

##Versao 0.8
Atualizado arquivo de dados(DadosFormatados.csv);
Adiconado classes para serializar e deserializar modelos preditivos(Jsat e Weka);
Adicionado classe de teste para AlgoritmoBatch;
Adcionado campo diaSemana na classe Bairro;c

##Versao 0.7
Adicionado UploadController para submissão de arquivos;
Adicionado páginas index,crime e uploadStatus;
Centralizado painel na pagina index;
Adicionado GlobalExceptionHandler para tratamento de excecoes;

##Versao 0.6.1
Bug fixado na classe AlgoritmoBatch;
Modificado executor de threads;
Adicionado getters and setters da classe Classificador;
Adicionado classe de Teste para AlgoritmosBatch;

##Versao 0.6
Adicionado pool de threads para execução dos algoritmos;
```

Fonte: Próprio autor (2017).

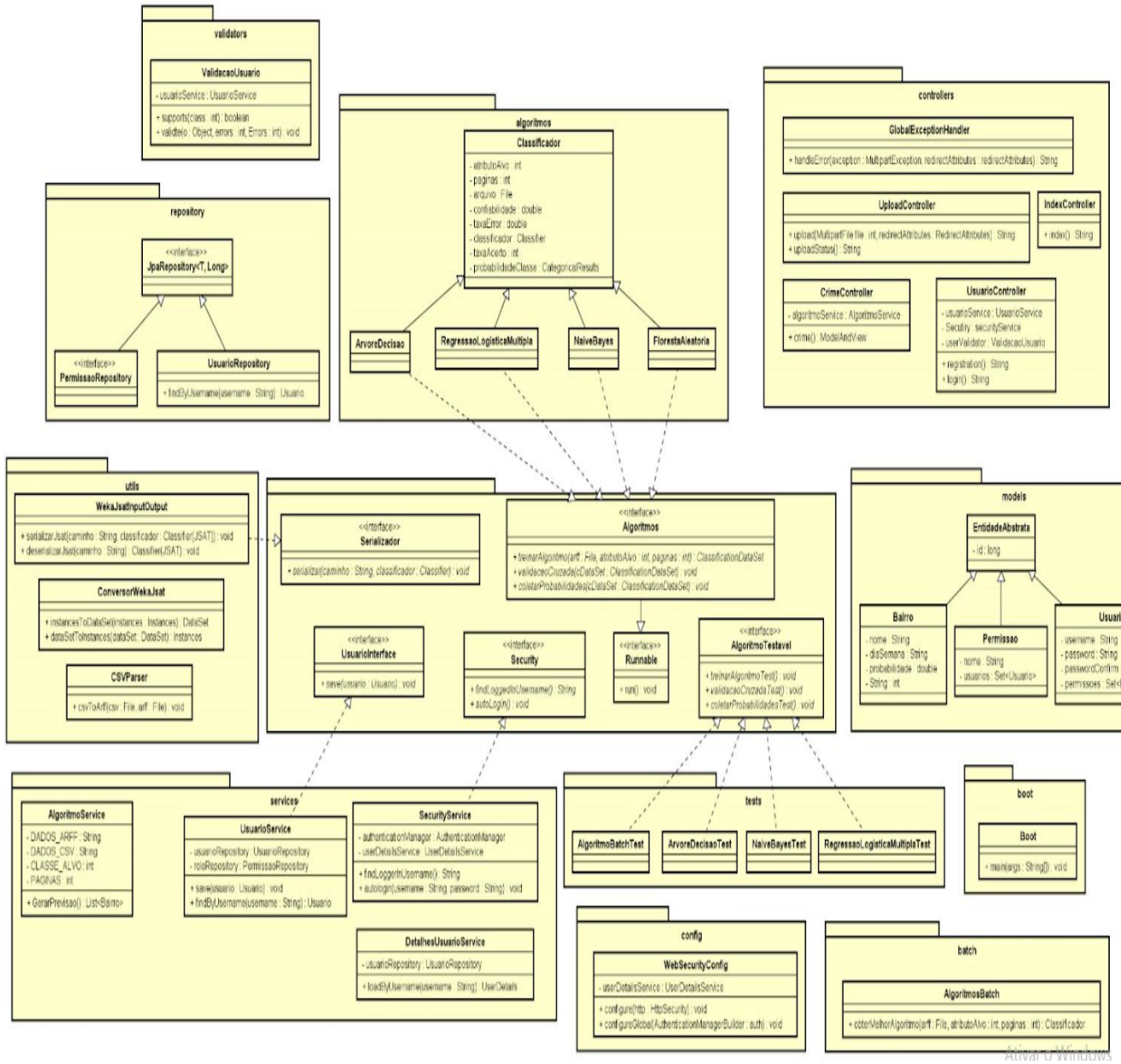
As mudanças vão desde criação de novas funcionalidades, mudanças na estrutura, disponibilidade de novos algoritmos como correção de bugs e refatoração, otimização do código fonte, de algumas classes para tentar deixar o sistema o mais desacoplado e alto coeso possível.

Com isto a expansibilidade da aplicação se torna muito mais fácil para integrar novas funcionalidades e futuras melhorias em diversos aspectos do sistema.

3.2 ESTRUTURA DO SISTEMA

A Figura 17 demonstra o artefato para descrever a estrutura do sistema representado, o o diagrama de classes, através de classes, atributos, operações e as diversas relações entre os objetos, (SILVA, 2007).

Figura 17– Diagrama de classes do sistema



Fonte: Próprio autor (2017).

Utilizou-se o padrão MVC, *Model View Controller*, na qual procura modularizar o projeto dividindo em camadas suas respectivas responsabilidades bem definidas acerca do projeto. O *controller*, ou controlador, é uma espécie de intermediador de requisições entre a interface do usuário, chamada de *view,* ou visão*,* e a camada de modelo que nada mais é do que a camada das classes que poderão ser persistidas em um banco de dados (KRASNER; POPE, 1988).

Classificando em pacotes, estrutura que agrupam as classes representando sentidos comuns, o diagrama possui:

- Algoritmos: Algoritmos disponíveis para a classificação ou regressão dos dados (Naive Bayes, Regressão Logística Múltipla, Àrvore de decisão, Florestas Aleatórias).
- *Batch*: Toda parte de processamento mais complexo (método Obter Melhor Algoritmo que testa todos os algoritmos utilizando computação paralela e *Threads* e obtem o melhor algoritmo baseado na taxa de acertos do classificador criado por cada um).
- *Boot*: Inicialização do sistema (classe *Boot*).
- *Config*: Configurações de segurança e do sistema no geral (classe *WebSecurityConfig*)
- *Controllers*: Camada onde está disponibilizado todos os controladores que tem o principal papel unir a camada de aplicação com a interface do cliente (classes *UploadController*, *CrimeController*, *UsuarioController* e *IndexController*,)
- Interfaces: Camada onde estão localizadas as interfaces que permitem um nível de abstração ainda maior fornecendo assinaturas de métodos para futuras melhorias. (interfaces: Algoritmos, *Serializador*, *Security*, *AlgoritmoTestavel*, *Runnable*, *UsuarioInterface*).
- *Models*: Camada que representa as classes do modelo, ou seja, classes que representam entidades do mundo real (classes Bairro, Usuário, Permissao).
- *Repository*: Conjunto de interfaces que representam métodos para comunicação com um SGBD para persistir, recuperar, remover e alterar dados (*PermissaoRepository* e *UsuarioRepository*).
- Services: Camada responsável pela regra de negócio da aplicação, as classes presentes nesta camada programam as interfaces disponibilizadas na camada *Repository* (*AlgoritmoService*, *UsuarioService*, *SecurityService*, *DetalhesUsuarioService*)
- *Tests*: Camada onde estão presentes todas as classes de testes da aplicação que foram realizadas nos pacotes de Algoritmos e *Batch* (*AlgoritmosBatchTest*, *ArvoreDecisaoTest*, *NaiveBayesTest*, *RegressaoLogisticaMultiplaTest*, *FlorestaAleatoriaTest*).
- *Utils*: Camada onde estão presentes todas as classes utilitárias do sistema (*ConversorWekaJsat*, *CSVParser*, *WekaJsatInputOutput*).
- *Validators*: Pacote onde estão presentes todas as classes de validação em termos de interface com o usuário. (*ValidacaoUsuario*).

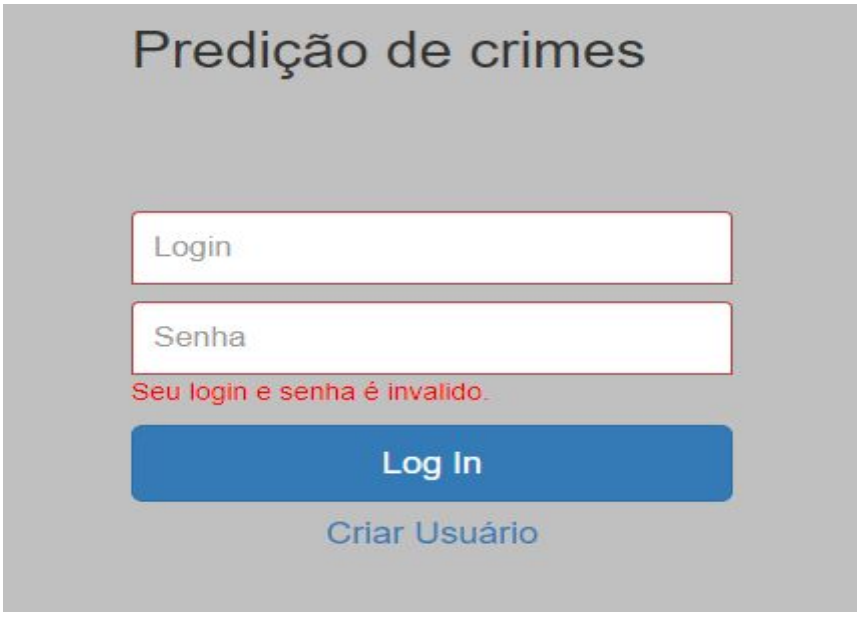
3.3 INTERFACE DA SOLUÇÃO

A interface a princípio, foi á disponibilidade de uma ferramenta, em que a secretaria de segurança pública do estado, pudesse visualizar as predições do modelo preditivo em um mapa interativo, com os índices e probabilidades a respeito dos bairros em João Pessoa, localizadas em um mapa.

Foi desenvolvido um controle de autenticação e permissão, com uma missão de tornar um ambiente mais seguro, para os usuários conseguirem acessar o sistema.

Na Figura 18 demonstra a tela de entrada para a aplicação:

Figura 18 – Tela de *Login* da aplicação.



Fonte: Próprio autor (2017).

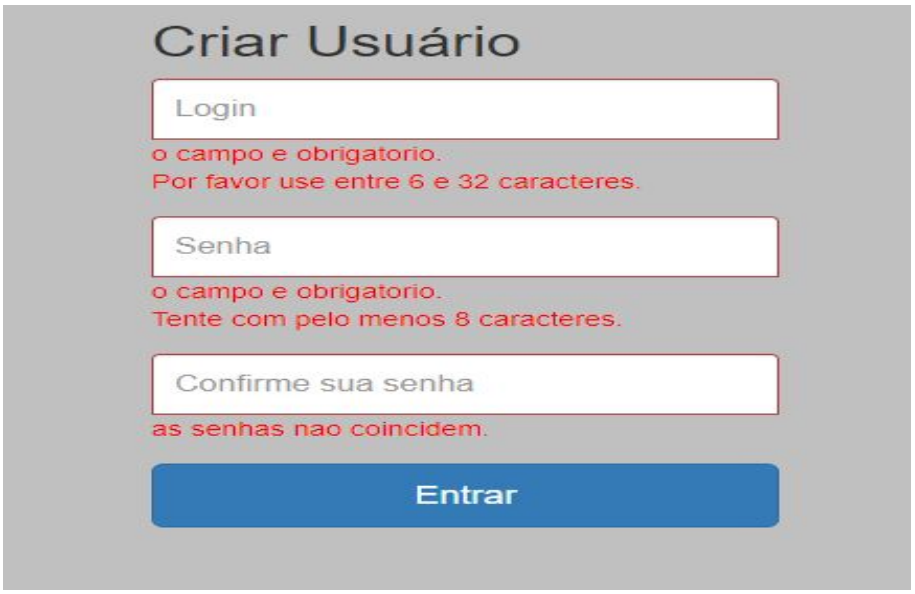
Só é permitido acessar o sistema, que tenha as credencias necessárias para isto. O botão de Criar conta está temporariamente disponível a visão do usuário comum meramente por prototipagem e demonstração

O controle de múltiplas permissões e tipos de usuário foi desenvolvido, porém não foi disponibilizado na interface.

Todas as telas do sistema estão com algumas validações para erros comuns de digitação de *login* e senhas erradas ou que não coincidem e também no cadastro de um novo usuário no sistema, possuindo indicações de quantidade mínima de caracteres permitidos para a criação de um novo usuário no sistema.

A Figura 19 demonstra a tela de registro para novos usuários e os possíveis erros caso, as condições de validação não se satisfaçam.

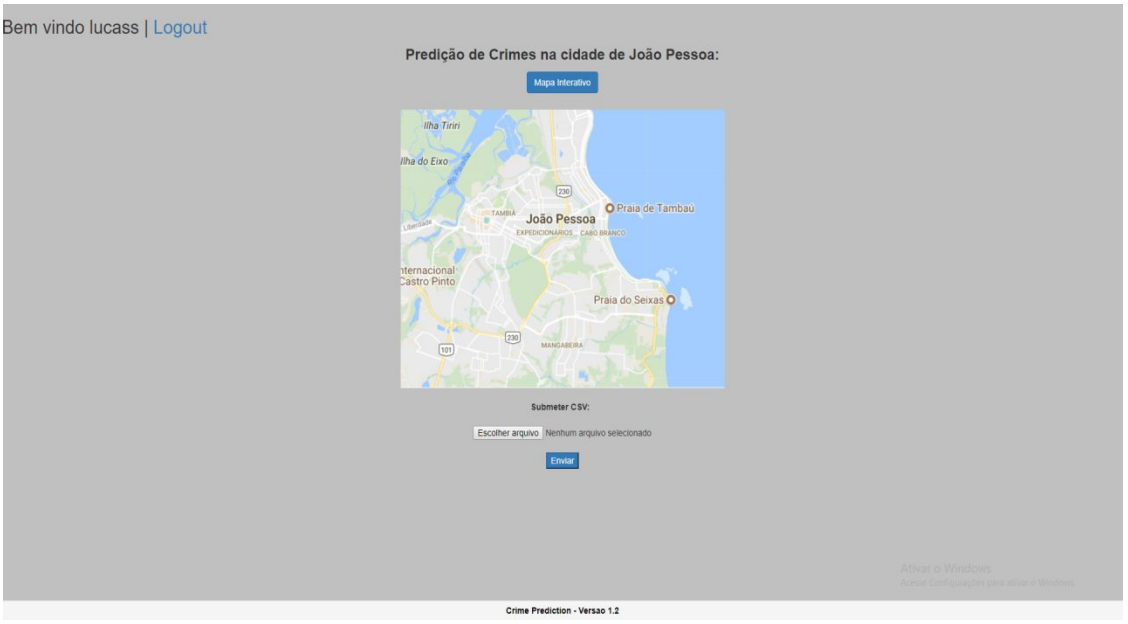
Figura 19 – Tela de cadastro de usuário da aplicação



Fonte: Próprio autor (2017).

A Figura 20 demonstra a tela *index*, tela inicial após a tela de *login* da aplicação e a autenticação do usuário ter ocorrido com sucesso.

Figura 20 – Tela *index* da aplicação



Fonte: Próprio autor (2017).

Na tela *index*, o sistema demonstra a opção para sair da sessão do usuário.

Esse botão de *Logout* significa possibilita o usuário, o qual fez a autenticação, e está permitido naquela sessão, sair da sessão e do sistema, retornando para página de *login*.

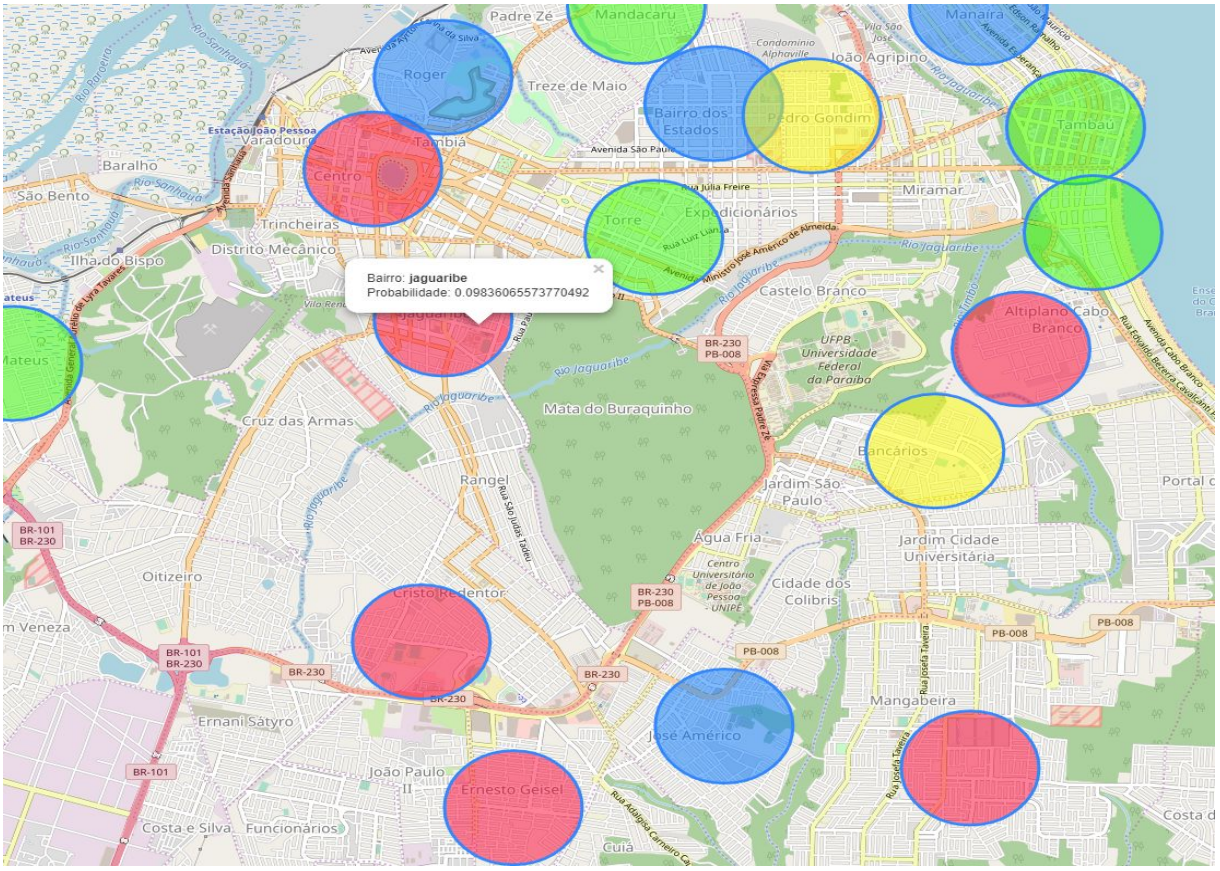
Também mostra um *footer, rodapé* contendo a versão atual do sistema as suas principais funcionalidades visíveis.

- **Mapa Interativo (Funcionalidade em que permite a visualização das possíveis probabilidades em que o classificador foi capaz de prever).**
- **Envio de arquivo CSV (Submissão de arquivos no formato CSV na qual obedecem ao padrão dos atributos descritos na metodologia deste trabalho).**

A interface da solução foi criada utilizando uma *library* em *JavaScript,* uma linguagem de programação interpretada muito utilizada na codificação das lógicas nas interfaces com o usuários. A *library* foi o *LeafletJS*, que é *open-source*, ou seja, disponível á todos de graça, para criação de mapas interativos.

Essa interface utiliza chamada a *Web Services*, que são chamadas feitas á serviços disponibilizados, que nesse caso, será ao modelo preditivo.

Figura 21 – Tela do mapa interativo



Fonte: Próprio autor (2017).

Na Figura 21, é demonstrada a visualização das probabilidades dos vinte bairros de João Pessoa utilizados para a predição dos crimes do tipo Furto. Cada bairro está representado por um círculo possuindo uma cor representativa, bairro alvo e a probabilidade em que o

modelo preditivo foi capaz de prever. As cores significam o nível de atenção em relação ás probabilidades. Como por exemplo:

- **Verde (probabilidade muito baixa, representa um nível seguro).**
- **Azul (probabilidade baixa, representa um nível estável).**
- **Amarelo (probabilidade mediana, representa um nível de atenção).**
- **Vermelho (probabilidade alta, representa um nível de perigo).**

As informações de predição de cada bairro são disponibilizadas para usuário através de um clique com o botão direito do mouse nos círculos visualizados.

3.4 RESULTADOS

Os algoritmos testados com o *dataset,* conjuntos dos dados prontos para o treino e os testes, foram os algoritmos de *Naive Bayes*, Regressão Logística Múltipla, Árvore de Decisão e Floresta Aleatória.

Foram escolhidos estes algoritmos devidos a sua popularidade e a segregação dos tipos de algoritmo em:

- Naive Bayes (Aprendizado supervisionado / Classificação).
- Regressão Logística Múltipla (Aprendizado supervisionado / Regressão).
- Árvore de Decisão (Aprendizado supervisionado / Classificação).
- Floresta Aleatória (Aprendizado supervisionado / Classificação).

A estrutura de Naive Bayes e Regressão Logística Múltipla foram utilizadas predominantemente listas e vetores de dados, já nas Árvores de decisões de Florestas Aleatórias foram utilizados a estrutura de árvores, e o algoritmo de Floresta Aleatória, é um algoritmo que procura solucionar o problema de *overfitting* ou sobreajuste, que é quando o modelo preditivo se ajusta muito bem a situação dos dados, mas quase sempre se torna ineficaz para prever novas situações que venham a ocorrer, esse problema ocorre com certa frequência no algoritmo de Árvore de Decisão.

Os resultados de classificação e regressão utilizados pelo modelo foram expressos pelo console da aplicação, quais foram comparados à taxa de acertos do classificador ou *regressor* e a consideração do índice de análise de concordância *Kappa*.

Os valores são dados em porcentagens e significa a representação da submissão dos dados ao algoritmo utilizado, por exemplo, com uma taxa de acertos de cem por cento,

significa que o classificador ou *regressor* com base nos dados de treino, acertou todas as previsões utilizando os dados de teste, representando um possível cenário perfeito.

Figura 22 – Representação dos índices dos modelos preditivos no console da aplicação

```
Algoritmo Naive Bayes
A precisão do modelo preditivo foi de: 81.83902190524707%
A taxa de acertos do modelo preditivo foram de: 81.83902190524707%
O índice de análise de concordância Kappa foi de: 80.21050324805235%

Algoritmo Àrvore de Decisão
A precisão do modelo preditivo foi de: 82.28476821192054%
A taxa de acertos do modelo preditivo foram de: 82.28476821192052%
O índice de análise de concordância Kappa foi de: 80.71635699172258%

Algoritmo Floresta Randomica
A precisão do modelo preditivo foi de: 80.71319409067753%
A taxa de acertos do modelo preditivo foram de: 80.71319409067753%
O índice de análise de concordância Kappa foi de: 78.96535284953504%

Algoritmo Regressão Logística Múltipla
A precisão do modelo preditivo foi de: 1.4696892511462047%
A taxa de acertos do modelo preditivo foram de: 1.4696892511462067%
O índice de análise de concordância Kappa foi de: 0.0%

Melhor Algoritmo: Àrvore de Decisão
```

Fonte: Próprio autor (2017).

Os resultados obtidos após a classificação do modelo preditivo foram bastante satisfatórios, com uma *accuracy,* ou seja, uma precisão de 82.28%, representando uma boa quantidade na taxa de acertos do modelo preditivo e um índice *Kappa* de 80.71%, no qual representa uma alta relação de concordância em relação às variáveis do modelo, utilizando o melhor algoritmo escolhido pela aplicação, foi o algoritmo de Arvores de Decisão como ilustrado na Figura 22.

Os resultados do algoritmo de *Naive Bayes* e Floresta aleatória foram bem próximos 81.83% de confiabilidade e 80.21% no índice *Kappa* no algoritmo *Naive Bayes* e 80.71% de confiabilidade e 78.96% no índice *Kappa* no algoritmo Floresta Aleatória.

Já no resultado do algoritmo de Regressão Logística Múltipla, os valores foram muito baixos sendo passiveis de total desconsideração, sendo 1.46% de confiabilidade e um índice *Kappa* nulo.

A área da curva *ROC* dos bairros foram bem satisfatórias também, isso mostra uma boa proporção entre os valores verdadeiros positivos em relação aos valores falsos positivos, a os valores da curva *ROC* dos vinte bairros testados está ilustrado na Figura 23.

Figura 23 – Área da curva *ROC* dos bairros utilizados no modelo

ROC Area	Class
0,972	AGUA FRIA
1,000	ALTIPLANO
1,000	ALTO DO MATEUS
0,981	BAIRRO DOS ESTADOS
1,000	BANCARIOS
1,000	BESSA
0,984	CABO BRANCO
0,973	CENTRO
1,000	CRISTO REDENTOR
0,993	GEISEL
1,000	JAGUARIBE
1,000	JOSE AMERICO
1,000	MANAIRA
0,969	MANDACARU
0,971	MANGABEIRA
0,965	PEDRO GONDIM
0,964	ROGER
0,991	TAMBAU
0,939	TORRE
0,948	VALENTINA

Fonte: Próprio autor (2017).

Todos os algoritmos foram submetidos à técnica de validação cruzada com dez páginas, ou seja, dez partições do total de registros, testados e validados particularmente para uma obtenção de uma média dos mesmos.

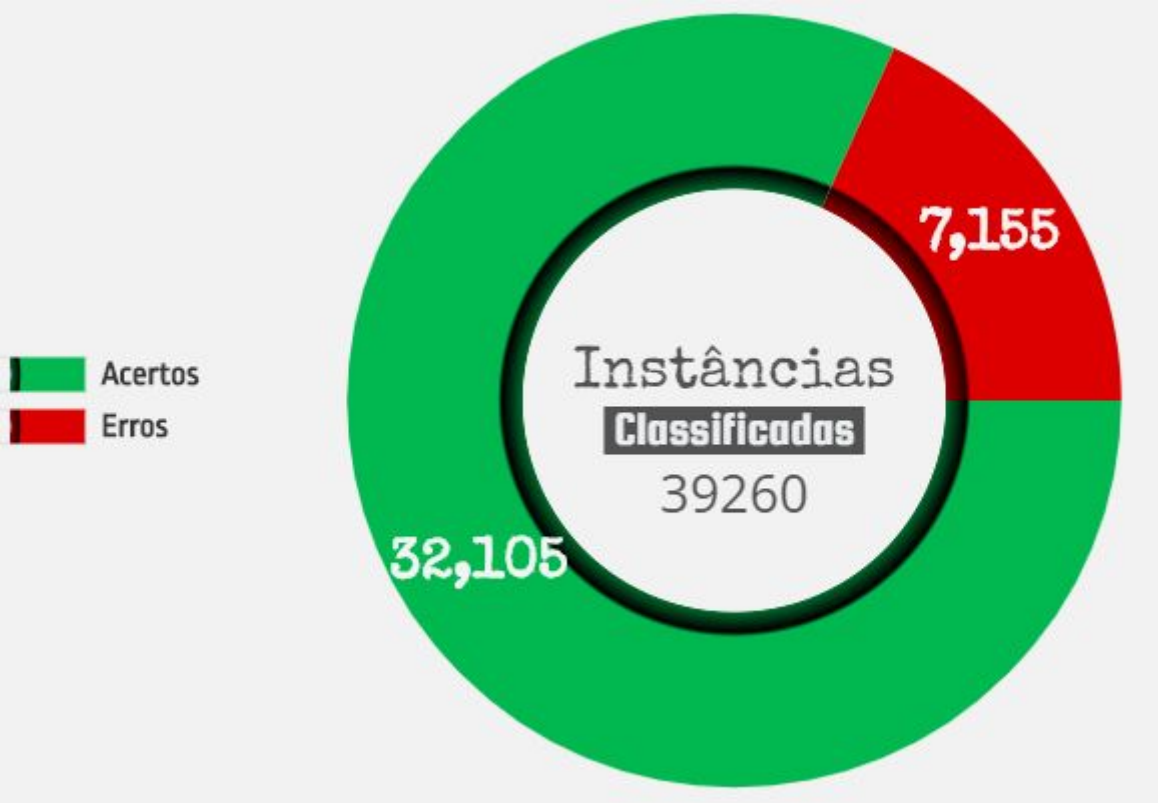
Todos os dados foram submetidos aos algoritmos com uma configuração de 50% dos dados utilizados foram separados para treino e os 50% restantes, foram utilizados para teste.

A probabilidade de cada bairro foi obtida testando vários conjuntos de *datapoint*, objeto que representa uma unidade singular de dados do *dataset,* entre si somando e se auto ajustando, conforme os dados eram consumidos, isso garantiu a margem das probabilidades descritas.

Foram utilizadas 39260 registros de crimes do tipo Furto, para os vinte bairros disponíveis na aplicação, para a classificação e avaliação do modelo preditivo.

Após a validação cruzada foram obtidos de 39260 tentativas de comparação entre os *datapoints,* 32105 acertos e 7155 erros de classificação no modelo como demonstrado na Figura 24.

Figura 24 – Representação da taxa de acertos e erros do classificador utilizado



Fonte: Próprio autor (2017).

As instâncias foram os registros classificados. O classificador obteve aproximadamente 81.77% de taxa de acertos, e 18.22% de taxa de erros.

A aplicação forneceu as probabilidades de predição para cada bairro dos vinte mencionados.

1. **Água Fria** [0.00000000071716889770140 %].
2. **Altiplano** [0.00000000000431553213453 %].
3. **Alto do Mateus** [0.00024889042428768819170 %].
4. **Bairro dos Estados** [0.00000005079386409153200 %].
5. **Bancários** [0.00000000001153710603740 %].
6. **Bessa** [0.00000000000000000000048 %].
7. **Cabo Branco** [0.000000000000000000000050 %].
8. **Centro** [0.00000000000000058295831 %].
9. **Cristo Redentor** [0.00000004110574686704945 %].
10. **Ernesto Geisel** [0.00000000003366798395543 %].
11. **Jaguaribe** [0.00000005046779248533146 %].

12. José Américo [0.0001845694183123192870 3 %].

13. Manaíra [0.000000000000000000033751 %].

14. Mandacaru [0.00000014746450978427637 %].

15. Mangabeira [0.77002951298461819185092 %].

16. Pedro Gondim [0.00000023354948155782452 %].

17. Roger [0.00000027917135738321896 %].

18. Tambaú [0.000000000000000000000001 %].

19. Torre [0.000000000000001652009668 %].

20. Valentina [0.22953622385332300703808 %].

Destaques para os bairros do Bessa, Tambaú e Cabo Branco, bairros com uma baixíssima probabilidade e os bairros de Mangabeira e Valentina com a maior probabilidade dos estudados, em ocorrer um crime do tipo Furto em João Pessoa, baseado nos registros históricos dos dados obtidos.

Foi desenvolvido na aplicação métodos para conversão de dados do *WEKA* para o *JSAT* e vice-versa, e métodos para serialização e *deserialização*, métodos para importar e exportar, dos classificadores obtidos, gerando um arquivo *.model* no qual contém o classificador para futuras classificações.

4 CONSIDERAÇÕES FINAIS

O trabalho teve como principal contribuição uma solução capaz de auxiliar o monitoramento de crimes de todos os tipos de Furto, na secretaria de segurança pública do estado. Com todo o apoio da Polícia Militar da Paraíba em relação à disponibilidade dados, foi possível desenvolver uma aplicação, que possa prever e demonstrar as probabilidades de vinte bairros da cidade de João Pessoa.

Nas seções a seguir serão abordadas respectivamente, as contribuições alcançadas até o momento, às dificuldades encontradas e as propostas como trabalhos futuros.

4.1 CONTRIBUIÇÕES ALCANÇADAS

Construção de uma aplicação *WEB* desacoplada e com alta coesão assim facilitando sua expansão e de fácil manutenção, com segurança e computação *multi-thread*, ou seja, utilizando vários núcleos do processador visando entregar uma melhor interação e um menor tempo de espera para o treinamento do modelo, com a inclusão de inteligência artificial e algoritmos de aprendizagem de máquina capazes de construir um modelo preditivo com uma boa taxa de acertos, sendo capaz de avaliar qual melhor algoritmo para aquele determinado conjunto de dados. Cerca de: 82.28% no *accuracy,* índice de precisão, e 80.71% de índice *Kappa* K, índice de concordância.

4.2 DIFICULDADES ENCONTRADAS

As principais dificuldades encontradas foram à obtenção dos dados, pelo fato de ser dados com um alto grau de sigilo, então se precisou de um termo de compromisso, assinado pelo orientador e pela coordenação do curso de Ciência da Computação e o acordo de cooperação técnica entre o Centro Universitário de João Pessoa, o pré-processamento dos dados, visto que a grande maioria estava com valores inválidos, vários atributos e valores em branco, valores fora do comum em termos de números e datas, campos com informações duplicadas, as limitações do *JSAT*, em situações de obter circunstâncias em tempo real dos atributos, validação de alguns algoritmos, como por exemplo, validação-cruzada para o algoritmo *Naive Bayes*, a obtenção da área da curva *ROC* só podia ser obtida a partir de

variáveis com valores binários, a difícil correlação dos *datapoints* nos métodos que utilizam árvore como estrutura de dados, Arvore de decisão e Floresta Aleatória, para estes algoritmos o classificador coloca probabilidade zero em alguns bairros com baixíssimas probabilidades, então para melhor visualização das predições utilizou-se o algoritmo *Naive Bayes* na interface. Outra dificuldade foi a leitura dos arquivos já no formato original *CSV*, foi preciso realizar a criação de um *parser* para o tipo de arquivo *ARFF* para conversão para um *dataset* do *JSAT*.

4.3 TRABALHOS FUTUROS

Como proposta de trabalhos futuros, o aumento da precisão do modelo preditivo, melhora dos valores das curvas *ROC* nos bairros e dos índices *Kappa,* o aumento da quantidade de algoritmos disponíveis para o treinamento e validação, maior quantidade de bairros e tipos de crimes, refinamento da localização no mapa geográfico como contornos em tornos de possíveis ruas e avenidas nos bairros, a adição de mais atributos estáticos para melhores taxas do classificador, expansão do sistema de permissões para a aplicação, adicionar a retroalimentação ao classificador, podendo assim fazer que o mesmo possa evoluir com a inserção de novos dados de registros de crimes e a validação de circunstâncias dos atributos estáticos em tempo real.

REFERÊNCIAS

AMARAL,F. **Aprenda Mineração de Dados:** Teoria e Prática.1. Ed. Alta Books, 2016

BATTULA.P.B; PRASAD.S.R.Dr. **An Overview of Recent Machine Learning Strategies in Data Mining.** International Journal of Advanced Computer Science and Applications. Vol.4. No.3, 2013.

BERK, R. **Criminal Justice Forecasts of Risk:** A machine learning approach**.** USA: University of Pennsylvania, Philadelphia, 2012.

BIAU.G. **Analysis of a Random Forests Model.** Université Pierre et Marie Curie – Paris VI. 2012

CRAWFORD, T.A.M; EVANS, K. **Crime prevention and Community Safety.** Oxford University Press, 2016.

DALGARRONDO.P. **Psicologia e semiologia dos transtornos mentais.** 2. Ed. Porto Alegre: Artmed, 2008.

DIRENE.A. **Visão geral sobre inteligência artificial.** Disponível em: < http://www.nce.ufrj.br/GINAPE/VIDA/ia.htm> Data acesso: 09/04/2017

DOS SANTOS.A.C.M. **Aprendizado de máquina aplicado ao diagnóstico de dengue.** XIII Encontro Nacional de Inteligência Artificial e Computacional, 2016.

FAWCETT.T. **A introduction to ROC analysis.** Institute for the Study of Learning and Expertise, USA. 2005.

FAYYAD, U.M; PIATETSKY-SHAPIRO.G; SMYTH.P; UTHURUSAMY.R. **Advances in knowledge discovery and data mining.** Massachusetts: AAAI Press, 1996.

FERRARI.M. **O que é BI?** Disponível em: < http://www.vortice.inf.br/noticia/http-blog-prgbrasil-com-2015-06-19-o-que-e-bi-preview-id222 > Data acesso: 10/04/2017

GIL.A.C; **Como Elaborar Projetos de Pesquisa.** 4 Ed. Atlas S.A, 2002

GONZALES.G.C; FILHO.J.M.Q; FRONCHETTI.F.L; MORAIS.V.R. **Uso da Inteligência artificial no cotidiano.** Artigo Científico e Seminário de Introdução a Ciências da Computação, Universidade Tecnológica Federal do Paraná, 2014.

GUYON.I; ELISSEEFF.A. **An Introduction to Variable and Feature Selection**. Journal of Machine Learning Research. 2003.

HAGENLOCHER.P. **Decision Tree Learning.** Technische Universitat Munchen. 2016

HIDALGO.B; GOODMAN.M. **Multivariate or Mutivariable Regression?** Am J Public Health. 2013.

IBGE. **Perfil dos Estados e Municípios Brasileiros 2014.** Instituto Brasileiro de Geografia e Estatística. Disponível em: < http://www.ibge.gov.br/home/ > Data acesso: 31/05/2017

JURAFSKY.D; MARTIN.H.J. **Speech and Language Processing.** 2014.

KRASNER.E.G; POPE.T.S. **A Cookbook for Using View-Controller User the ModelInterface Paradigm in Smalltalk-80.** Smalltalk-80 is a trademark of ParcPlace Systems. 1988.

LINOFF.G.S; BERRY.M. **Data Mining Techniques:** For maketing, sales, and customer relationship management. 3.Ed. Indianapolis, 2011.

LOWD.D; DOMINGOS.P. **Naive Bayes Models for Probability Estimation.** Departament of Computer Science and Engineering, University of Washington, Seattle, 2005.

MEIRELLES.A; FERNANDES.C.E; CASTRO.D.R. **Tradução de Textos Baseado em Estatística.** Universidade Estadual de Campinas. 2014.

MCCARTHY.J. **Branches of AI:** What is Artificial Intelligence. 2007. Disponível em: < http://www-formal.stanford.edu/jmc/whatisai/node1.html > Data acesso: 04/06/2017

MOHAMAD.M; HASSAN.H; NASIEN.D; HARON.H. **A Review on Feature Extraction and Feature Selection for Handwritten Character Recognition.** International Journal of Advanced Computer Science and Applications. Vol.6, No.2, 2015.

NILSSON.N.J. **The quest for artificial intelligence**: A history of ideas and achievements. Cambridge University Press, 2010.

PANNU.A; TECH.M. **Artificial Intelligence and its Application in Different Areas.** International Journal of Engineering and Innovative Technology (IJEIT). Vol.4, 2015

PROVOST.F; FAWCETT,T. **Data Science for Business: What you need to know about data mining and data-analytic thinking**. " O'Reilly Media, Inc.", 2013.

REZENDE.S.O; PUGLIESI.J.B; MELANDA.E.A; PAULA.M.F. **Mineração de dados:** Rezende, S.O. Sistemas inteligentes**.** Ed Manole Ltda, 2003.

RAFF.E. **JSAT**: Java Statistical Analysis Tool, a Library for Machine Learning. JMLR:v18:16-131. Journal of Machine Learning Research.

RUSSEL.S; NORVIG.P. **Artificial Intelligence:** A modern approach. 3. Ed Pearson, 2010.

SANTOS.F.C; CARVALHO.C.L. **Aplicação da Inteligência Artificial em Sistemas de Gerenciamento de Conteúdo**. 3. Instituto de Informática(Universidade Federal de Goiás), 2008.

SCHIMITT.V.F. **Uma análise comparativa de técnicas de aprendizagem de máquina para prever a popularidade de postagens no Facebook.** Universidade Federal do Rio Grande do Sul, 2013.

SEGURIDADJUSTICIAYPAZ: **Consejo Ciudadano para la Seguridad Pública y Justicia Penal A.C.** Disponível em: http://www.seguridadjusticiaypaz.org.mx/biblioteca/prensa/send/6-prensa/239-las-50-ciudades-mas-violentas-del-mundo-2016-metodologia >. Data acesso: 12/03/2017.

SILVEIRA.F.A. **Introdução a ciência de dados**: Mineração e Big Data. 1.Ed. Rio de Janeiro: Altabooks, 2016.

SILVA, R. P. e. UML 2 em Modelagem Orientada a Objetos. Florian´ polis: Visual Books, 2007.

SOBRAL.O.J. **Inteligência Humana**: Concepções e possibilidades. Revista Científica FacMais, Volume. III, Número 1, 2013.

SOKOLOVA.M; LAPALME.G. **A Systematic Analysis of Performance Measures for Classification Tasks.** Département d'informatique et de recherché opérationnelle. Université de Montréal, Canada. 2009.

TIOBE. The Importance Of Being Earnest. Disponível em: < https://www.tiobe.com/tiobe-index/ >. Data Acesso: 07/05/2017

VISA.S; RAMSAY.B; RALESCU.A; KNAAP.D.V.E; **Confusion Matrix-based Feature Selection.** Ceur-ws.org. 2011.

WITTEN.I.H; FRANK.E; HALL.M.A; PAL.C.J. **Data Mining**: Pratical Machine Learning Tools and Techniques. 4. Ed San Francisco, Morgan Kaufmann; 2017.