CENTRO UNIVERSITÁRIO LUTERANO DE PALMAS

Recredenciado pela Portaria Ministerial nº 1.162, de 13/10/16, D.O.U nº 198, de 14/10/2016
ASSOCIAÇÃO EDUCACIONAL LUTERANA DO BRASIL

Lucas Ribeiro Reis de Sousa

UTILIZAÇÃO DE APRENDIZADO DE MÁQUINA PARA O DESENVOLVIMENTO DE UM MODELO COMPUTACIONAL PARA PREVISÃO DE RISCO DE DENGUE EM PALMAS - TO

Palmas – TO

2018

Lucas Ribeiro Reis de Sousa

UTILIZAÇÃO DE APRENDIZADO DE MÁQUINA PARA O DESENVOLVIMENTO DE UM MODELO COMPUTACIONAL PARA PREVISÃO DE RISCO DE DENGUE EM PALMAS - TO

Trabalho de Conclusão de Curso (TCC) II elaborado e apresentado como requisito parcial para obtenção do título de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. M.e Jackson Gomes de Souza.

Palmas – TO

2018

Lucas Ribeiro Reis de Sousa

UTILIZAÇÃO DE APRENDIZADO DE MÁQUINA PARA O DESENVOLVIMENTO DE UM MODELO COMPUTACIONAL PARA PREVISÃO DE RISCO DE DENGUE EM PALMAS - TO

Trabalho de Conclusão de Curso (TCC) II elaborado e apresentado como requisito parcial para obtenção do título de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. M.e Jackson Gomes de Souza

Aprovado em: _____/_____/_____

BANCA EXAMINADORA

_____

Prof. M.e Jackson Gomes de Souza

Orientador

Centro Universitário Luterano de Palmas – CEULP

_____

Prof. M.e Fernando Luiz de Oliveira

Centro Universitário Luterano de Palmas - CEULP

_____

Prof. Dra. Parcilene Fernandes de Brito

Centro Universitário Luterano de Palmas - CEULP

Palmas – TO

2018

RESUMO

SOUSA, Lucas Ribeiro Reis. **Utilização de aprendizagem de máquina para o desenvolvimento de um modelo computacional para previsão de risco de dengue em Palmas - TO**. 2018. 49 f. Trabalho de Conclusão de Curso (Graduação) – Curso de Sistemas de Informação, Centro Universitário Luterano de Palmas, Palmas/TO, 2018.

O presente trabalho tem como finalidade aplicar o algoritmo de aprendizagem de máquina, denominado como Regressão com Vetores de Suporte (SVR), a fim de realizar predição de dados de dengue. Os dados utilizados para este trabalho foram obtidos por meio do trabalho de Cavalcante (2013), com informações sobre dados de dengue da região de Palmas - TO. Com o intuito de encontrar a melhor aplicação do SVR para predição da dengue, foram aplicados conceitos de aprendizagem de máquina para estudo do funcionamento desse algoritmo, bem como os conceitos e técnicas de extração do conhecimento (KDD). Assim, foram realizadas análises entre as variáveis da base de dados a fim de identificar as que mais pudessem influenciar na predição de dengue. E, com isso, foi desenvolvido um modelo computacional para predição de dengue, bem como um software web para interagir com esse modelo.

Palavras-chave: Predição. Dengue. SVR

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

AM	Aprendizagem de Máquina
CEULP	Centro Universitário Luterano de Palmas
IA	Inteligência Artificial
KDD	Knowledge-Discovery in Databases
SVC	Support Vector Classification
SVM	Support Vector Machine
SVR	Support Vector Regression

SUMÁRIO

1 INTRODUÇÃO

A dengue é a doença viral transmitida por mosquitos com a mais rápida proliferação no mundo, que obteve nas últimas décadas bastante atenção por ter se tornado um problema de saúde pública mundial (WHO, 2012). Para Ribeiro et. al. (2006), a doença atinge com mais intensidade os países tropicais em resultado de suas características ambientais, climáticas e sociais.

No Brasil, as vitalidades de epidemias de dengue ocorrem desde a década de 1980, atingindo na atualidade todos seus 27 estados e sendo responsável por aproximadamente 60% dos casos notificados nas Américas (CAVALCANTE, 2013).

Conforme o estudo de Valadares, C. Filho e Peluzio (2013), o estado do Tocantins é apontado como área endêmica da doença, principalmente por estar localizado na região da Amazônia Legal. Com isso, as condições do estado favorecem para o surgimento de episódios de surtos e/ou epidemias. Por exemplo, apenas em 2010 foram notificados 17.294 casos de dengue, sendo que 55% destes foram resumidos unicamente em 5 cidades do estado: a capital tocantinense Palmas, Porto Nacional, Paraíso do Tocantins, Araguaína e Colinas do Tocantins (VALADARES; C. FILHO; PELUZIO, 2013).

Segundo os dados do Ministério da Saúde (BRASIL, 2007), no município de Palmas, o progresso de casos de dengue aconteceu de forma exorbitante entre os anos de 2000 e 2007, sendo 1396 casos registrados em 2000, e 9112, em 2007. Diante disso, diversas medidas governamentais e não governamentais têm sido tomadas para combater e prevenir a doença, entre elas, análise geoespacial da dengue e dos fatores que podem influenciar na sua transmissão (CAVALCANTE, 2013).

Entre outros métodos auxiliados por programas computacionais para prevenção do combate a dengue, pode ser citado o trabalho de Guo et. al (2017), que apresenta um modelo de aprendizagem de máquina para previsão precisa e robusta de incidência de Dengue na China.

A Aprendizagem de Máquina (AM), segundo Mitchell (1997), pode ser definida como "a capacidade de melhorar o desempenho na realização de alguma tarefa por meio de experiência", ou seja, a habilidade aprender computacionalmente sobre problemas que se desejar tratar por meio de experiências passadas (FACELI, 2011). Algoritmos de AM podem ser aplicados em diversos tipos de problemas, tais como predição de dados climáticos

(ARAÚJO, 2015), e previsão de demanda de energia elétrica a curto prazo (RUAS et al., 2000).

O processo de previsão de dados pode ser realizado por diferentes algoritmos dentro da área de AM, como Redes Neurais Artificiais (SANTOS, 2005), Árvores de Decisão (PITOMBO e COSTA, 2015) e Máquinas de Vectores de Suporte (*Support Vector Machine* - SVM) conforme apresentado em Oliveira (2017).

Segundo Guo et al. (2017), o modelo SVR (*Support Vector Regression*) é considerado o mais moderno e preciso para prever dados. Para Vapnik (1995), este algoritmo permite que a máquina aprenda com base em dados de entradas e saídas, e assim possa ser capaz de prever novas saídas considerando novas entradas, ou seja, prever dados com base em experiências passadas.

O presente trabalho busca desenvolver um modelo preditivo de dengue para o município de Palmas - TO. Para tanto é utilizado técnicas de AM, como o SVR (*Support Vector Regression*), para a definição de um modelo de previsão de dados que possa contribuir para a sociedade como ferramenta de combate a doenças endêmicas e epidêmicas, mais especificamente a dengue.

Desta forma, este trabalho está estruturado da seguinte forma: o capítulo 2 apresenta o referencial teórico, que aborda conceitos e exemplificações de aprendizagem de máquina, teoria do aprendizado estatístico, bem como conceitos, técnicas e aplicações de SVM. O capítulo 3 é apresentado o desenho de estudo, materiais e procedimentos para o desenvolvimento deste trabalho. Posteriormente, será apresentado os resultados e discussões do trabalho, e em seguida às conclusões. Por fim, é apresentado as referências utilizadas no projeto.

2 REFERENCIAL TEÓRICO

2.1 Aprendizagem de Máquina

O conceito por trás de aprendizagem pode ser entendido como a maneira de utilizar percepções para agir e melhorar habilidades de um autor para agir no futuro (RUSSEL e NORVIG, 2004).

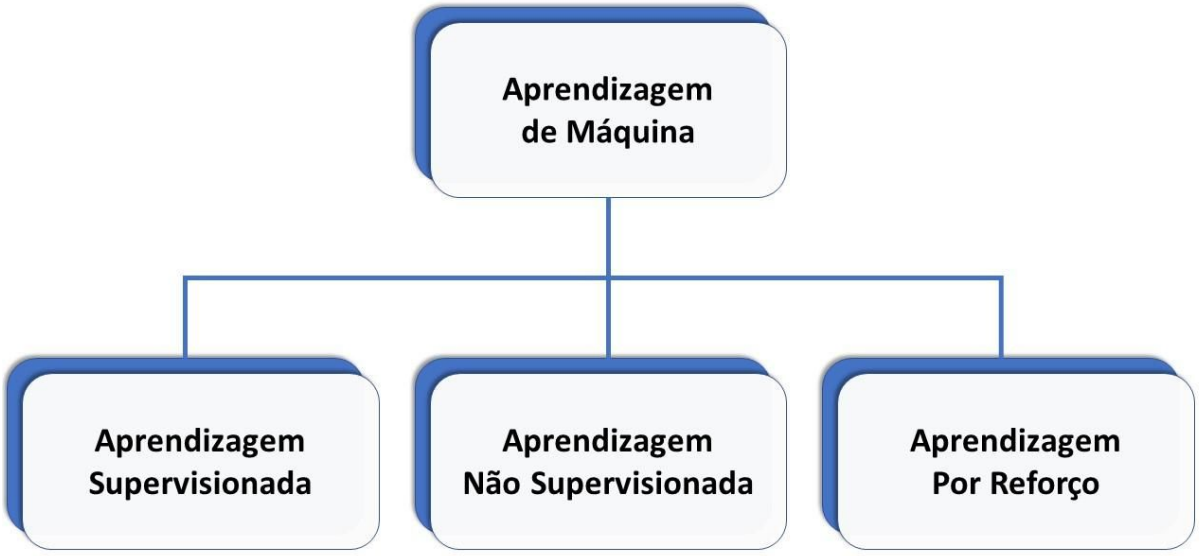
Segundo exposto por Alexandre (2010), para o homem a aprendizagem é considerada um método de mudança de comportamento provocado por meio da experiência constituída por fatores emotivos, neurológicos, conviveis e ambientais. Um exemplo de aprendizagem envolvendo um autor humano, consideramos esse autor como um aluno de auto escola. Toda vez que um instrutor falar "Reduz para primeira marcha", o autor poderá aprender uma prescrição de condição-ação sobre quando reduzir a marcha de um carro para primeira.

Para a máquina, a aprendizagem é a facilidade de aprender com experiências passadas, empregando um fundamento de inferência denominado indução, a fim de melhorar a performance na realização de tarefas ou solucionar problemas como (FACELI et. al, 2011): reconhecimento de palavras faladas, predição de taxas de cura de pacientes com diferentes doenças, detecção do uso fraudulento de cartões de crédito condução de automóveis de forma autônoma em rodovias, ferramentas que jogam gamão e xadrez de forma semelhante a humanos campeões, diagnóstico de câncer por meio da análise de dados de expressão gênica.

O Aprendizagem de Máquina (AM) está naturalmente ligado à Inteligência Artificial (IA), entretanto outras áreas possuem colaborações significativas em seu avanço, como Probabilidade e Estatística, Teoria da Computação, Neurociência, Teoria da Informação, entre outras (FACELI et. al, 2011). O foco de AM pode ser entendido como a capacidade de tomar decisões fundamentadas em conhecimentos adquiridos previamente (LANGLEY e SIMON, 1995, apud, ARAÚJO, 2015), e que seja capaz de aprimorar a base de conhecimento a cada tomada de decisão, para estar sempre adquirindo nova experiência (ARAÚJO, 2015).

A área de aprendizagem de máquina se divide em três categorias (RUSSEL e NORVIG, 2004): aprendizagem supervisionada; aprendizagem não supervisionada; e aprendizagem por reforço; conforme apresentado na Figura 1, a seguir.

Figura 1 - Categorias de Aprendizagem de Máquina (AM)



A aprendizagem supervisionada envolve um conjunto de dados de entradas e saídas que são utilizados como treinamento (ARAÚJO, 2015), para aprendizagem de uma função a partir destes dados (RUSSEL e NORVIG, 2004). A definição de aprendizagem supervisionada pode ser compreendida como uma máquina que possui dados de entrada e de saída para seu treinamento, possuindo perguntas e consequentemente respostas. A exemplo disso, pode ser citado uma máquina para identificação facial. Conforme apresentada a imagem de uma face, o sistema busca de quem é esta face, a partir de banco de faces que o sistema possui (MAIA, 2016).

Já na aprendizagem não supervisionada, não importa os dados de saída, sendo que são disponíveis apenas dados de entrada para o autor. A aprendizagem acontece por meio do processamento dos dados de entrada e observando os padrões estabelecendo representações sucessivamente para codificar características e compreendê-las automaticamente (FERNEDA, 2006). Um exemplo da aplicação desse tipo de aprendizagem é na classificação de comentários (SCHMITZ, 2015).

A aprendizagem por reforço segundo Russel e Norvig (2013) é compreendida como a mais extensa entre as três categorias devido o agente desse tipo de aprendizagem aprender por

meio do reforço ao invés de aprender o que ser feito, envolvendo a diversidade de entender o funcionamento do ambiente em questão.

A destinação deste trabalho é realizar a predição de dados de dengue utilizando SVR, por isso, as técnicas de aprendizagem de máquina não serão discutidas com mais detalhes.

2.2 Conceitos Fundamentais

Nesta seção serão apresentados os principais conceitos para entendimento da técnica SVM. Para tanto, a Seção 2.2.1 apresenta a Teoria do Aprendizado Estatístico (TAE) e sua aplicação em algoritmos de aprendizagem de máquina.

2.2.1 Teoria do Aprendizado Estatístico

Desenvolvida por Vapnik (2005) com base em estudos iniciados em Vapnik e Chervonenkis (1971), a Teoria do Aprendizado Estatístico (TAE), segundo Faceli et al. (2011), "estabelece uma série de princípios que devem ser seguidos na obtenção de classificadores com boa capacidade de generalização".

Faceli et al. (2011) descreve que o processo de aprendizado de um algoritmo de AM, que utiliza um conjunto de treinamento X, composto de *n* pares (x_i, y_i), é capaz de gerar um classificador específico $\hat{h} \in H$, sendo h um classificador e H o conjunto de todos os classificadores que um determinado algoritmo de AM pode estabelecer. De forma geral, a TAE determina premissas matemáticas que auxiliam na escolha de um classificador específico $\hat{h}$ baseado em um conjunto de dados de treinamento (FACELI et al., 2011).

2.2.1.1 Considerações sobre a Escolha do Classificador

Segundo Burges (1998), quando se aplica a TAE, inicialmente atribui-se que os dados no qual o aprendizado está acontecendo são formados de forma independente e identicamente distribuída (i.i.d) de acordo com uma distribuição de probabilidade $P(x,y)$ que representa a relação entre os objetos e os rótulos. O erro (também conhecido como risco) esperado de um classificador *h* para todos os dados desse domínio pode ser calculado pela Equação 1 (Muller et al., 2001), dessa forma, compreendendo a capacidade de generalização de *h*. Na Equação 1, $c(h(x),y)$ é uma função de custo ligando a previsão $h(x)$ à saída desejada *y,* em que $y_i \in \{-1,+1\}$. Em problemas de classificação a função 0-1, estabelecida pela Equação 2, geralmente é a mais utilizada entre os tipos de funções de custo. Essa função retorna o valor 0 se x é classificado corretamente e 1 em caso contrário.

$$R(h) = \int c(h(x),y)\, dP(x,y) \qquad (1)$$

$$c(h(x), y) = \frac{1}{2}|y - h(x)| \quad (2)$$

O risco esperado exposto na Equação 1 não pode ser minimizado diretamente, visto que em geral a classificação da probabilidade $P(x, y)$ é desconhecida, tendo somente informação dos dados de treinamento. Usualmente, a maioria das técnicas de AM supervisionadas, por exemplo, utilizam o princípio de indução para inferir uma função $\hat{h}$ que minimize o erro sobre os dados de treinamento, esperando um menor erro possível sobre os dados (FACELI et al., 2011). Ainda conforme descrito por Faceli et al. (2011), o risco empírico de h, conforme apresentado na Equação 3, apresenta o desempenho do classificador nos dados de treinamento, através da taxa de classificações incorretas obtidas em **X**.

$$R_{emp}(h) = \frac{1}{2}\sum_{i=1}^{n} c(h(x_i), y_i) \quad (3)$$

Ainda que a minimização do risco possa levar a um menor risco, em alguns casos isso pode não acontecer, por exemplo, em um conjunto de dados menores isso nem sempre pode ser garantido.

Tendo um conjunto H, é sempre possível encontrar uma função h com com risco empírico pequeno. A TAE fornece diversos limites no risco esperado de uma função h, pois os exemplos de treinamento podem se tornar pouco informativos para a tarefa de aprendizado, podendo ser aplicados na escolha do melhor classificador.

2.2.1.2 Limites no Risco Esperado

Entre os limites fornecidos pela TAE, existe um que relaciona o risco esperado de uma função ao seu risco empírico e a um termo de capacidade. Segundo Burges (1998), esse limite é garantido com probabilidade $1 - \theta$, em que $\theta \in [0, 1]$. A Inequação 4 apresenta o limite em questão.

$$R(h) \leq R_{emp}(h) + \sqrt{\frac{VC\left(ln\left(\frac{2n}{VC}\right) + 1\right) - ln\left(\frac{\theta}{4}\right)}{n}} \quad (4)$$

Essa Inequação comprova a importância de se regular a classe de funções h, a fim de adaptar a quantidade de dados de treinamento disponível. Isto é, o risco esperado pode ser minimizado adequadamente para determinada escolha, por meio do algoritmo de aprendizado, de um classificador $\hat{h}$ pertencente a uma classe de funções H com baixa dimensão VC, e que esse classificador minimize o risco empírico (FACELI, 2011). Através desses objetivos, Vapnik (1998) estabeleceu um princípio de indução denominado minimização do risco

estrutural, que busca a função de menor complexidade possível que tenha um baixo erro para os dados de treinamento.

Alguns problemas devem ser considerados no procedimento de minimização do risco estrutural, apresentado na Inequação 4. A princípio, conforme descrito por Muller et al. (2001), avaliar a dimensão VC de uma classe de funções em geral não é um trabalho simples, uma vez que seu valor pode ser desconhecido ou infinito.

Deste modo, o conceito de margem (Smola et al., 1999) pode ser aplicado ao risco estrutural, como forma alternativa, para funções lineares do tipo $h(x) = w \cdot x$ (em que **w** é o vetor normal a *h*). A margem de um exemplo será uma medida de confiança da previsão do classificador (FACELI, et al., 2011). Conforme descrito em Faceli (2011), para um problema em que $y_i \in \{-1, +1\}$, dados uma função *h* e uma amostra y_i, a margem $\varrho(h(x_i), y_i)$ com que esse objeto é classificado por *h* pode ser calculada pela Equação 5.

$$\varrho(h(x_i), y_i) = y_i h(x_i) \tag{5}$$

Segundo Smola e Scholkopf (2002), para obter a margem geométrica de um objeto x_i é necessário dividir o termo da direita da Equação 5 pela norma de w, ou seja, por $||w||$.

Com base no conceito posto, é possível definir o risco ou erro marginal de uma função $h(R_p(h))$ sobre um conjunto de dados de treinamento (FACELI, 2011). Para Smola et. al. (1999), esse erro oferece a proporção de exemplos de treinamentos que possuem margem de confiança é menor que uma constante $\rho > 0$ escolhida. A Equação 6 apresenta o risco marginal, onde $I(q) = 1$ se *q* é verdadeiro e $I(q) = 0$ se *q* é falso.

$$R_p(h) = \frac{1}{n} \sum_{i=1}^{n} I(y_i h(x_i) < \rho) \tag{6}$$

Para Smola et al. (1999), o limite apresentado na Equação 7, se aplica a uma constante *c* tal que, tenha uma probabilidade $1 - \theta \in [0, 1]$, para todo $\rho > 0$ e *H* referindo-se à classe de funções lineares $h(x) = w \cdot x$ com $||x|| \leq R$ e $||w|| \leq 1$.

$$R(h) \leq R_p(h) + \sqrt{\frac{c}{n}\left(\frac{R^2}{\rho^2} log^2\left(\frac{n}{p}\right) + log\left(\frac{1}{\theta}\right)\right)} \tag{7}$$

Da mesma forma que na Equação 4, tem-se na Expressão 7 o erro limitado pela soma de uma medida de erro dentro do conjunto de treinamento, nesse caso o erro marginal, a um termo de capacidade. Este limite expõe que uma maior margem ρ implica um menor termo de capacidade (FACELI et al., 2011).

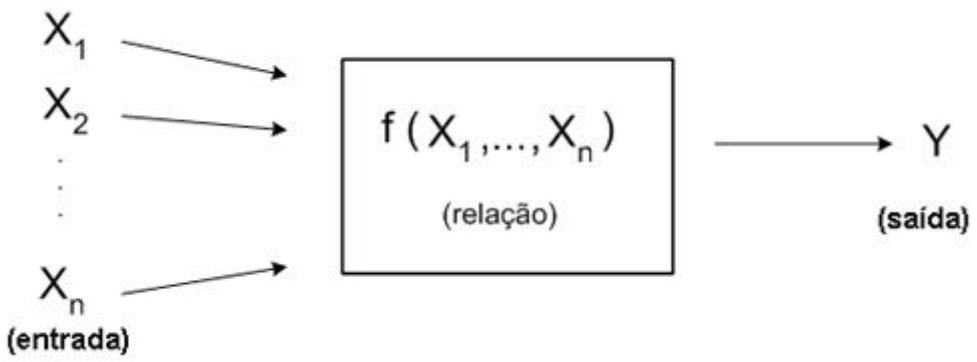
Faceli et al. (2011) afirma que na geração de um classificador linear, deve-se buscar um hiperplano que tenha margem ρ alta e ofereça poucos erros marginais. Consequentemente, minimiza o erro sobre os dados de treinamento bem como sobre novos dados de entrada. Esse hiperplano é consideravelmente ótimo.

Embora exista outros limites apresentados na literatura, os limites apresentados nesta seção oferece uma base teórica considerada suficiente para compreensão das Máquinas de Vetores de Suporte (*Support Vector Machines* - SVMs) apresentadas na seção seguinte.

2.2.2 Método de Regressão

O método de regressão possibilita realizar a predição de uma variável dependente (y) em função de uma ou mais variáveis independentes (x) (LARSON e FARBER, 2010 apud ARAÚJO, 2015), ou seja, verifica como determinada(s) variável(i)s pode alterar o comportamento de outra. Esse tipo de processo é apresentado na Figura 2, onde os valores $x_1, x_2 ... x_n$ são chamados de variáveis de entrada regressora e y de variável de saída(PORTAL ACTION, 2017).

Figura 2 - Representação do processo de regressão.



Fonte: (PORTAL ACTION, 2017)

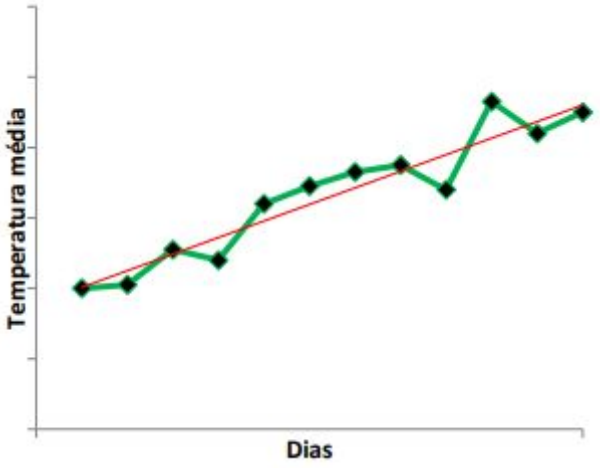
A técnica de regressão que envolve apenas uma variável independente é chamado de regressão simples, enquanto a técnica que envolve duas ou mais variáveis independentes é nomeada regressão múltipla (HAIR Jr. et al., 2005).

Em sua formulação matemática, o processo de regressão objetiva encontrar uma função h(x) que mais se aproxime da função real $f(x)$, com intuito desta função ser capaz de prever o valor da variável dependente (y) por meio das variáveis independentes (x) (ARAUJO, 2015). Conforme exposto por Faceli et al. (2011), a ligação dos pontos de x real (valor coletado do conjunto de dados) e y previsto (valor de y fornecido pela função h(x),

demonstrado por $\hat{y}$) gera uma linha semelhante ou mais próximo possível da linha que é gerada pela ligação dos pontos x e y reais.

A Figura 3 apresenta ligação entre os pontos x e y (neste caso são dados de treinamento), representada pela linha verde, bem como a ligação entre os pontos x e $\hat{y}$ representada pela linha vermelha. Os pontos x e y (dados de treinamento) são representados pelos losangos (FACELI et al., 2011).

Figura 3 - Ilustração de um problema de regressão

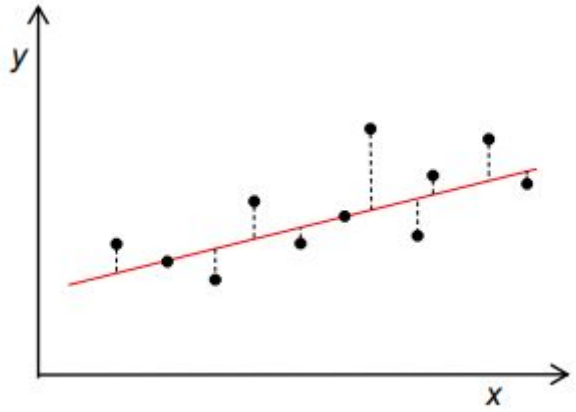


Fonte: Araujo (2015)

A técnica de regressão aplicada na ilustração da Figura 3 é denominada regressão linear (ARAUJO, 2015).

Segundo Freund (2006) a linha de regressão que apresenta o ajuste mais preciso ao dados é a que possuir o menor valor da soma das distâncias ao quadrado entre os pontos reais x e y e a linha de regressão x e $\hat{y}$. A distância é encontrada pela diferença entre o valor de y e o valor de $\hat{y}$ (valor previsto), para um determinado x (ARAUJO, 2015). A Figura 4 apresenta essas distâncias representadas pelas linhas tracejadas.

Figura 4 - Distância entre os pontos x e y da linha de regressão.



Fonte: Araujo (2015)

A aplicação de regressão linear nem sempre é aplicável para qualquer problema que envolva regressão, como por exemplo, estudos sobre o impacto da poluição atmosférica na saúde populacional, devido ao caráter não linear da variável resposta (TADANO et al., 2009).

Até o momento, os conceitos apresentados são referentes à regressão entre duas variáveis, denominada regressão linear simples (GOMES, 2000). O método de regressão também pode ser realizado em cenários em que uma linha reta não consegue se regular aos dados por não serem lineares, denominado regressão não linear (FREUND, 2006).

Araujo (2015) afirma que "a regressão não linear possui a mesma variação da regressão linear, sendo elas: a regressão não linear simples: para dados com apenas duas variáveis; e a regressão não linear multivariada: para dados com três ou mais variáveis".

2.2.2.1 Exemplo de Cáculo de Regressão

Com o objetivo de obter um melhor entendimento sobre aplicação do método de regressão, é apresentado uma situação hipotética para exemplificar o passo-a-passo dos cálculos para predição de dados.

Para tal exemplo serão utilizados os dados da Tabela 1, que mostram a relação entre a renda mensal de determinada família (variável independente *x*), e o consumo de energia elétrica dessa família (variável dependente *y*). Assim pretende-se predizer qual será o consumo de energia elétrica de determinada família a partir de uma renda conhecida.

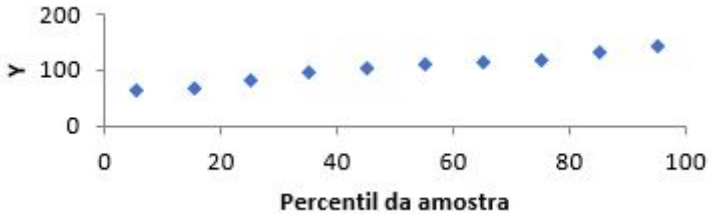
Tabela 1 - Dados de um problema de regressão

Consumo (y)	Renda (x)

122	139
114	126
86	90
134	144
146	163
107	136
68	61
117	62
71	41
98	120
?	150

Transpondo os dados fornecidos para um plano cartesiano é obtido o gráfico da Figura 5, em que os losangos representam os pares de x e y.

Figura 5 - Gráfico dos dados de exemplo



Primeiramente é necessário calcular os valores de *a* e *b*, uma vez que *a* é o valor de *y* quando *x* for igual a zero, e b é a variação de y em relação a x.

Para calcular os valores de *a* e *b* é necessário utilizar suas respectivas formas (Equação 8 e Equação 9) (FREUND, 2006).

$$a = \underline{y} - b\underline{x} \tag{8}$$

$$b = \frac{\sum_{i=1}^{n} x_i(y_i - \underline{y})}{\sum_{i=1}^{n} x_i(x_i - \underline{x})} \tag{9}$$

Com a aplicação das fórmulas acima na tabela de dados de exemplo, chega-se a seguinte Equação (Derivada da Equação 10 da linha de regressão):

$$Consumo = 52,69 + 0,4954 * Renda + e$$

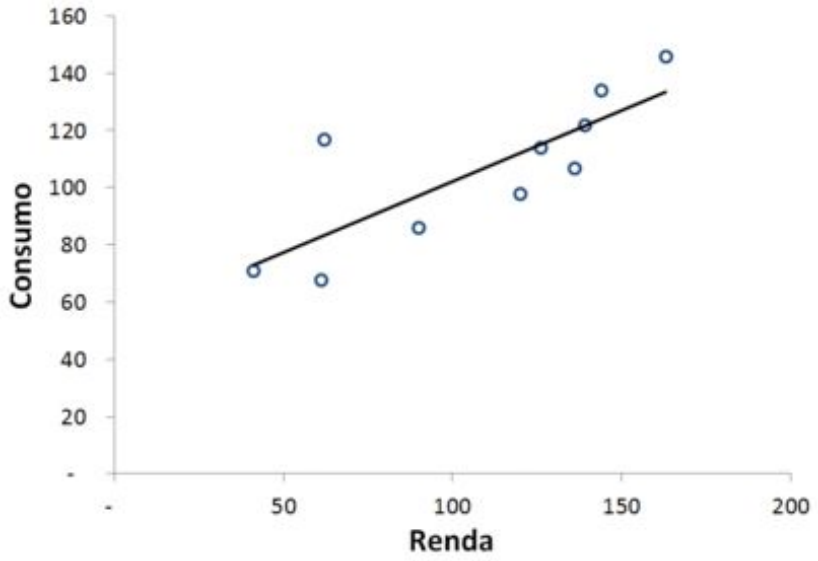
$$y = \alpha + \beta x + \varepsilon \quad (10)$$

onde,

e representa a variação de *y* que não é explicada pelo modelo.

A linha de regressão representada pela Equação 10 é representada pelo gráfico apresentado na Figura 6.

Figura 6 - Gráfico da linha de Regressão



Visto que os valores de *a* e *b* foram calculados, é possível resolver o problema em que se deseja predizer qual será o possível consumo de uma família com a renda de 150 aplicando a Equação 10 no problema proposto. Assim, conclui-se que, uma família com a renda 150 possuirá, aproximadamente, um consumo de 128.

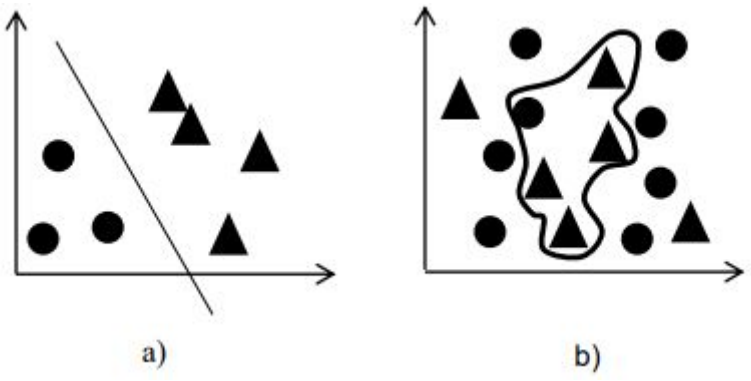
2.3 Máquinas de Vetores de Suporte (SVMs)

As Máquinas de Vetores de Suporte (*Support Vector Machines* - SVMs) integram em uma técnica que vem ganhando bastante reconhecimento da comunidade de AM nos últimos anos (MITCHELL, 1997). Segundo Faceli et al. (2011) “os resultados da aplicação dessa técnica são comparáveis e muitas vezes superiores aos obtidos por outros algoritmos populares de aprendizado tal como as RNAs”.

As SVMs são fundamentadas na teoria do aprendizado estatístico (apresentada na Seção 2.2.1). Inicialmente, a SVM foi desenvolvida para solucionar problemas de classificação, entretanto também é utilizada para solucionar problemas de regressão (CHAMASEMANI e SINGH, 2011). Exemplos de aplicações das SVMs podem ser encontrados em diversos domínios, tais como categorização de textos (JOACHINS, 2002), bioinformática (REZENDE e SILVA, 2009) e reconhecimento de caracteres (CARVALHO et al., 2009).

O Classificação com Vetores de Suporte (*Support Vector Classification* - SVC), um caso específico de SVM para tratar problemas de classificação, fundamenta-se do conceito de aprendizagem supervisionada apresenta na Seção 2.1. A SVC objetiva classificar objetos de classes diferentes por meio de amostras anteriores e, a partir disso, ser capaz de classificar uma nova entrada a classes que ela pertença (GUNN, 1998). A Figura 7 apresenta os tipos de classificações da SVC.

Figura 7 - Exemplos de hiperplanos. a) hiperplano de separação para dados lineares; b) hiperplanos de separação para dados não lineares



Fonte: Araújo (2015)

Uma SVC encontra o melhor hiperplano de separação entre duas classes de amostras de treinamento, dentro de um espaço de atributos, com margem máxima. Conforme apresentado na Figura 7, existem duas formas de aplicação do SVC, tais como: classificação

linear, utilizada para dados lineares, ou seja, que podem ser separados por uma reta; e classificação não linear, utilizada para dados não lineares, que não podem ser separados por uma reta.

2.3.1 SVMs Lineares

Esta subseção apresenta a utilização de SVMs no prosseguimento de fronteiras lineares para a separação de objetos pertencentes a duas classes. Para tanto, esta subseção ainda se divide em duas formulações: a primeira, considerada mais simples, trata de problemas linearmente separáveis (BOSER et al., 1992); considerada uma extensão da primeira, a segunda formulação lida com a definição de fronteiras lineares sobre conjuntos de dados mais gerais (CORTES e VAPNIK, 1995).

2.3.1.1 SVMs com Margens Rígidas

As SVMs lineares com margens rígidas determinam fronteiras lineares a partir de dados linearmente separáveis (FACELI et al., 2011). Ainda segundo este autor, seja **X** uma coleção de treinamento com *n* objetos $x_i \in X$ e seus específicos rótulos $y_i \in Y$, no qual X constitui o espaço de entrada e $Y = \{-1, +1\}$ são as classes possíveis, então se, for possível separar os objetos da classe $+1$ e -1 por um hiperplano, **X** é considerado linearmente separável. A equação do hiperplano (Equação 8), onde $\mathbf{w} \cdot \mathbf{x}$ é o produto escalar entre os vetores **w** e **x**, **w** $\in X$ é o vetor normal ao hiperplano descrito e $\frac{b}{\|w\|}$ corresponde à distância do hiperplano em relação à origem, com $b \in R$ (FACELI et al., 2011).

$$h(x) = \mathbf{w} \cdot \mathbf{x} + \mathrm{b} \tag{8}$$

A equação apresentada acima pode ser utilizada para dividir o espaço de entrada X em duas espaços $\mathbf{w} \cdot \mathbf{x} + b > 0$ e $\mathbf{w} \cdot \mathbf{x} + b < 0$. Assim, uma função sinal $g(\mathbf{x}) = \mathrm{sgn}(h(\mathbf{x}))$ pode ser utilizada para obter classificadores, conforme apresentado na Equação 9 (FACELI et al., 2011).

$$g(x) = sgn(h(x)) = \begin{cases} +1 \; se \; w \cdot x + b > 0 \\ -1 \; se \; w \cdot x + b < 0 \end{cases} \tag{9}$$

A partir de $h(x)$ é possível obter um número infinito de hiperplanos correspondentes, pela multiplicação de w e *b* por uma mesma constante. Segundo Muller et al. (2001) o hiperplano canônico em relação ao conjunto X é determinado como aquele em que w e *b* são escalados de forma que os exemplos mais próximos ao hiperplano $\mathrm{w} \cdot \mathrm{x} + b = 0$ satisfaçam a Equação 10.

$$\left| w \cdot x_i + b \right| = 1 \tag{10}$$

Satisfazendo as restrições a seguir (BURGES, 1998):

$$\begin{cases} w.x_i + b \geq +1 \; se \; y_i = +1 \\ w.x_i + b \leq -1 \; se \; y_i = -1 \end{cases} i = 1, 2, ..., n. \quad (11)$$

onde:

$y_i : y_i \in \{-1, +1\}$ condiz ao valor de classificação de x_i e determina se o objeto está na parte maior que zero ou menor que zero do classificador.

Para encontrar o hiperplano ótimo (o que realiza a melhor separação das classes) é preciso atingir a maximização da margem. Essa maximização pode ser alcançada pela minimização de $||w||$ (CAMPBELL, 2000). Assim, respeita-se o seguinte problema de otimização:

$$Minimizar_{w,b} \frac{1}{2} ||w||^2 \quad (12)$$

Com as restrições: $y_i(w \cdot x_i + b) - 1 \geq 0, \; \forall_i = 1, \; ..., \; n \quad (13)$

As restrições são necessárias para garantir que não haja dados de treinamento entre as margens de separação das classes. Devido a isso, esse tipo de SVM é conhecido como SVM com margens rígidas (FACELI et al., 2011).

Para Faceli et al. (2011) o problema de minimização de $||w||$ pode ser resolvido com a introdução de uma função lagrangiana, respeitando as restrições relacionadas a parâmetros denominados multiplicadores de Lagrange α_i (Equação 14).

$$L(w, b, \alpha) = \frac{1}{2}||w||^2 - \sum_{i=1}^{n} \alpha_i (y_i(w \cdot x_i + b) - 1) \quad (14)$$

onde:

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i \text{ e } b = \sum_{i=1}^{n} \alpha_i y_i \quad (15)$$

A função lagrangiana deve ser minimizada para encontrar a solução ótima, o que implica na maximização das variáveis α_i e minimização de *w e b* (CHAMASEMANI e SINGH, 2011), resultando no problema de otimização (FACELI et al., 2011):

$$Maximizar \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (16)$$

Com as restrições: $\begin{cases} \alpha_i \geq 0, \; \forall_i = 1, ..., n \\ \sum_{i=i}^{n} \alpha_i y_i = 0 \end{cases}$

(17)

O problema de otimização apresentado acima possui apenas um máximo global que pode ser encontrado (EL-NAQA et al., 2002 apud ARAÚJO, 2015). Desta forma, tem-se:

$$h(x) = sgn\,(\Sigma\alpha_i y_i (x_i \,.\, x) \;+\; b) \tag{18}$$

A função terá α_i diferente de zero para os objetos mais próximos do hiperplano, denominados de Vetores de Suporte (SV), e α_i igual a zero para outros objetos (ARAÚJO, 2015). Desta forma tem-se a função de classificação:

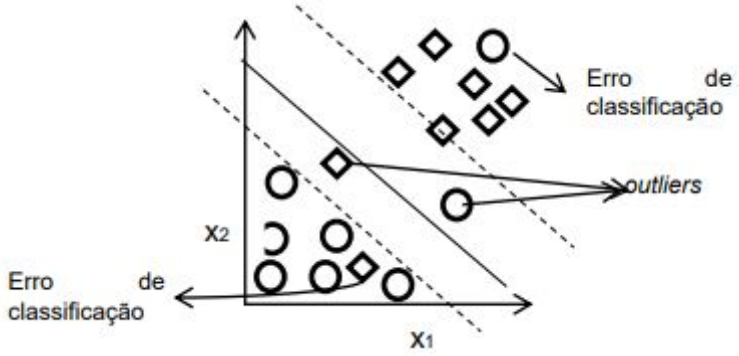
$$f(x) = \Sigma\alpha_i y_i x_i x \;+ \mathrm{b} \tag{19}$$

Consequentemente, o hiperplano obtido pela resolução deste problema é capaz de classificar dados que sejam linearmente separáveis (SOUTO et al., 2003).

2.3.1.2 SVMs com Margens Suaves

Existem casos em que o hiperplano (classificador linear) não consegue separar dados de forma perfeita (ARAÚJO, 2015). Isso ocorre devido a diversos fatores, entre eles a presença de ruídos e *outliers* nos objetos ou à própria condição do problema, que não pode ser linear (FACELI et al., 2011). Na Figura 8 é apresentado um conjunto de dados com *outliers*, representando um tipo de problema que não é linearmente separável.

Figura 8 - Exemplo de dados com outliers



Fonte: Araújo (2015)

Para solucionar este tipo de de problema deve-se permitir que alguns objetos possam desrespeitar a restrição da Equação 13. Isso é proporcionado com a introdução de variáveis de folga ξ_i, para todo $i = 1, ..., n$. Essas variáveis suavizam as restrições necessárias ao problema de otimização, que tornam (SOMOLA e SCHOLKOPF, 2002):

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i,\; \xi_i \geq 0,\; \forall_i = 1, ..., n \tag{20}$$

A aplicação do procedimento apresentado acima suaviza as margens do hiperplano e permite que alguns objetos continuem entre os hiperplanos H_1 *e* H_2 e também a ocorrência de alguns erros de classificação (FACELI et al., 2011). Devido a isso, as SVMs obtidas nesse tipo de problema são referenciadas como SVMs com margens suaves.

É considerado um erro no conjunto de treinamento quando o valor de ξ_i é maior que 1 (FACELI et al., 2011). Por isso, a soma dos ξ_i significa um limite no número de erros de treinamento. A partir disso, a Equação 12 é reformulada como (BURGES, 1998):

$$Minimizar_{w,b,\xi} \frac{1}{2}||w||^2 + C(\sum_{i=1}^{n} \xi_i) \tag{21}$$

A constante *C* é um termo de regularização que pode ser entendido com uma forma de determinar o controle sob o ajuste entre a importância de maximizar a margem e ajustar os dados. O termo $\sum_{i=1}^{n} \xi_i$ descreve o limite existente no número de erros no conjunto de dados (ARAUJO, 2015).

Para encontrar a solução ótima para este tipo de problema, da mesma forma que para SVMs de margens rígidas, é necessário aplicar o método de Lagrange, realizar a maximização dos multiplicadores de lagrange, e a minimização de *w* e *b* (CHAMASEMANI e SINGH, 2011). Assim, tem-se como resultado o seguinte problema de otimização:

$$Maximizar_{\alpha} \sum_{i-1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \tag{22}$$

Com as restrições $\begin{cases} 0 \leq \alpha_i \leq C, \ \forall_i = 1,\ldots,n \\ \sum_{i=1}^{\alpha} \alpha_i y_i = 0 \end{cases}$

(23)

onde:

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i \qquad \text{e} \qquad b = \sum_{i=1}^{n} \alpha_i y_i \,. \tag{24}$$

Igualmente nas margens rígidas, os pontos x_i que apresentam o $\alpha_i > 0$ são denominados de vetores de suporte (SV). A função apresentada na Equação 23, define o classificador e é escrita da mesma forma que a Equação 19, divergindo apenas pela forma que é determinado o valor de α_i (ARAUJO, 2015).

$$f(x) = \Sigma \alpha_i y_i x_i x + b \tag{25}$$

Contudo, ainda existem problemas em que, mesmo utilizando as SVC com margens suaves, não é possível encontrar um hiperplano que consiga classificar corretamente todos dos dados. Esses problemas lidam com dados não lineares, que não podem ser separados por uma linha, conforme apresentado na Figura 2. Desta forma, fez-se necessário introduzir o conceito de SVM não linear, exposto na Seção 2.3.2.

2.3.2 SVMs Não Lineares

As SVMs não lineares trabalham com problemas não lineares mapeando o conjunto de treinamento do seu espaço original, referido como de entradas, para um novo espaço de maior dimensão, chamado espaço de características (*feature space*) (HEARST et al., 1998, apud FACELI et al., 2011).

O procedimento utilizado pelas SVMs não lineares é originado pelo teorema de Cover (HAYKIN et al., 1999): dado um conjunto de dados não linear no espaço de entrada *X*, esse teorema garante que *X* pode ser convertido em um espaço de características I no qual com alta probabilidade os objetos são linearmente separáveis. Para tanto, duas condições devem ser satisfeitas: a primeira é que a transformação seja não linear; e a segunda é que a dimensão do espaço de características seja satisfatoriamente alta (FACELI et al., 2011).

O mapeamento dos dados para espaço de características é realizado por meio da Equação 26 (FACELI et al., 2011):

$$\Phi(x) = \Phi(x_1, x_2) = (x_1^2, \sqrt{2x_1}x_2, x_2^2) \tag{26}$$

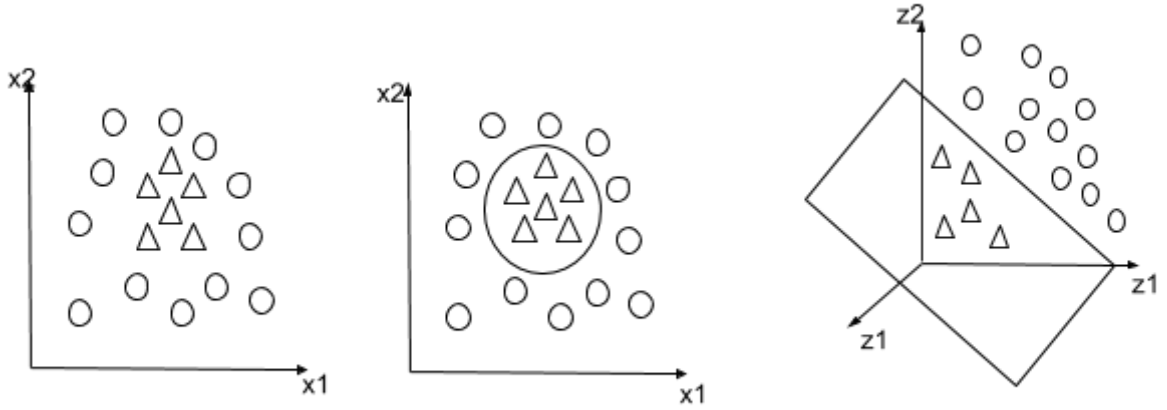
onde:

x: é o objeto que se deseja mapear para um espaço de características;

x_1, x_2 : são as coordenadas do x no espaço original.

A Figura 9 ilustra o mapeamento para um conjunto de dados não linear para um espaço de características.

Figura 9 - Exemplo de transformação realizada em conjunto de dados não linear para espaço de características.



(a) Conjunto de dados (b) Fronteira para duas dimensões (c) Fronteira para três dimensões

Fonte: adaptado Faceli et al. (2011)

Para que seja possível encontrar um classificador para o espaço de características apresentado na Figura 9 é necessário o uso de funções denominadas Kernel (CHAMASEMANE e SINGH). Um Kernel *K* é uma função que comporta dois pontos x_1 e x_2 do espaço de entradas e admite o produto escalar desses objetos no espaço de características (HERBRICH, 2001 apud FACELI et al., 2011). Desta forma, tem-se:

$$K(x_i,\ x_i) = \Phi(x_i) \cdot \Phi(x_j) \tag{27}$$

Para o mapeamento apontado na Equação 24 e dois objetos x_i e x_j em R^2, por exemplo, o kernel é dado por (FACELI et al., 2011):

$$K(x_i,\ x_i) = (x_i \cdot x_i)^2 \tag{28}$$

Aplicando a função Kernel na Equação 25 (equação do classificador ótimo com margens suaves) tem-se então (BURGES, 1998):

$$f(x) = \Sigma\alpha_i y_i K(x_i, x_j)\ + b \tag{29}$$

Usualmente, a função Kernel é aplicada sem que se conheça o mapeamento Φ, que é gerado implicitamente (FACELI et al., 2011). Por isso, segundo Lorena e Carvalho (2007), a utilidade dos Kernels está na simplicidade de seu cálculo e em sua capacidade de representar espaços abstratos.

Existem várias funções Kernel que podem ser empregadas para efetuar a separação de dados, como: os polinomiais, os de função com base radial (*radial basis function* - RBF) e os sigmoidais (Tabela 2).

Tabela 2 - Funções Kernel mais comuns

Tipo de Kernel	**Função** $K(x_i, x_j)$	**Parâmetros**
Polinomial	$(\delta(x_i \cdot x_j) + \kappa)^d$	δ, κ *e* d
RBF	$\exp\left(-\sigma\left\|x_i - x_j\right\|^2\right)$	σ
Sigmoidal	$\tanh(\delta(x_i \cdot x_j) + k)$	δ e κ

Para Faceli et al. (2011) a conquista de um classificador através do uso de SVMs compreende então a escolha de uma função kernel, além de parâmetros dessa função e do valor da constante de regularização *C*. Assim, a escolha do kernel e dos parâmetros refletem no desempenho do classificador obtido, uma vez que eles definem a fronteira de decisão induzida.

O algoritmo apresentado na Tabela 3 retrata a formulação final seguida pelas SVMs em seu treinamento (FACELI et al., 2011).

Tabela 3 - Algoritmo de treinamento de SVM

Algoritmo de treinamento de SVM	
	Entrada: Um conjunto de *n* objetos de treinamento
	Saída: SVM treinada
1	Seja $\alpha^* = (\alpha_1^*, \ldots, \alpha_n^*)$ a solução de:
2	Maximizar: $\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j K(x_i, x_j)$
3	Sob as restrições: $\begin{cases} \sum_{i=1}^{n} y_i \alpha_i = 0 \\ 0 \le \alpha_i \le C, i = 1, \ldots, n \end{cases}$
4	O classificador é dado por: $g(\mathbf{x}) = \text{sgn}(h(\mathbf{x})) = sgn\left(\sum_{x_i \in SV} \alpha_i^* y_i K(x_i, x) + b^*\right)$

5 Em que: $b^* = \frac{1}{n_{SV:\alpha^*<C}} \sum_{x_j \in SV:\alpha_j^*<C} \left(\frac{1}{y_j} - \sum_{x_i \in SV} \alpha_i^* y_i K(x_i, x_j) \right)$

Fonte: Faceli et al. (2011)

Para resolução de problemas de regressão, é aplicado o modelo de SVM chamado de vetores de suporte de regressão (*Support Vector Regression* - SVR) (VAPNIK, 1995). A Seção 2.3.3 apresenta uma abordagem deste modelo de SVM.

2.3.3 Regressão com Vetores de Suporte (SVR)

A SVR é fundamentada na teoria do método de regressão, apresentada na Seção 2.2.2. O algoritmo ε-SVR (*Support Vector Regression*) (VAPNIK, 1995) tem como finalidade encontrar uma função *h*(x) que crie saídas constantes para os dados de treinamento que possuem desvio de ε de seu rótulo desejado. Essa função deve ser o mais uniforme e regular possível (FACELI et al., 2011).

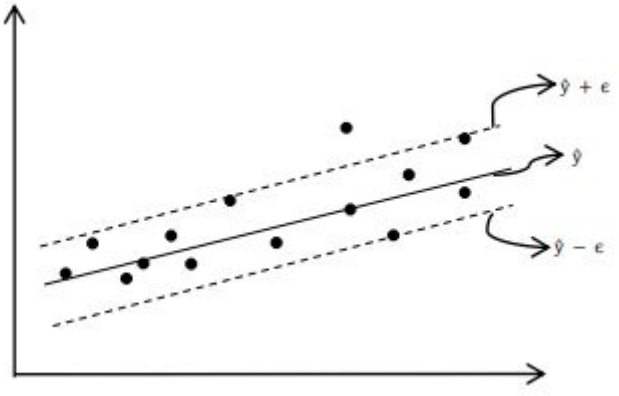
Observando a estrutura apresentada para o caso de classificação, considera-se primeiramente o uso de funções lineares *h* (Equação 8). Assim sendo, a regularidade significa procurar uma função com **w** pequeno, o que pode ser alcançado com a minimização da regra $||w||$ (FACELI et al., 2011). Com isso, tem-se o seguinte problema de otimização:

$$Minimizar_{w,b} \frac{1}{2}||w||^2 \quad (30)$$

Com as restrições: $\begin{cases} y_i - w \cdot x_i - b \leq \varepsilon_i \\ w \cdot x_i + b - y_i \leq \varepsilon_i \end{cases}$ (31)

À vista disso, procura-se encontrar a função linear que aproxima os pares de treinamento (x_i, y_i) com uma precisão de ε (FACELI et al., 2011). A Figura 10 ilustra esse procedimento.

Figura 10 - Representação da linha de regressão e suas margens



Fonte: Araujo (2015).

Conforme apresentado na Figura 10, procura-se encontrar a função linear bem como agrupar os dados de treinamento dentro de uma região *h*, representada pelo espaço entre as linhas tracejadas (FACELI et al., 2011). Este espaço é chamado de tubo de regressão (SMOLA e SCHOLKOPF, 1998).

Semelhante ao caso das SVMs de margens suaves, o problema de regressão pode ser disposto com inserção de variáveis de folga, tornando possível trabalhar com ruídos e *outliers* nos objetos (FACELI et al., 2011). Assim, tem-se:

$$Minimizar_{w,b,\xi,\underline{\xi}} \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{n} \xi_i + \underline{\xi}_i\right) \tag{32}$$

$$\text{Com as restrições: } \begin{cases} y_i - w \cdot x_i - b \leq \varepsilon + \xi \\ w \cdot x_i + b - y_i \leq \varepsilon + \xi \end{cases} \tag{33}$$

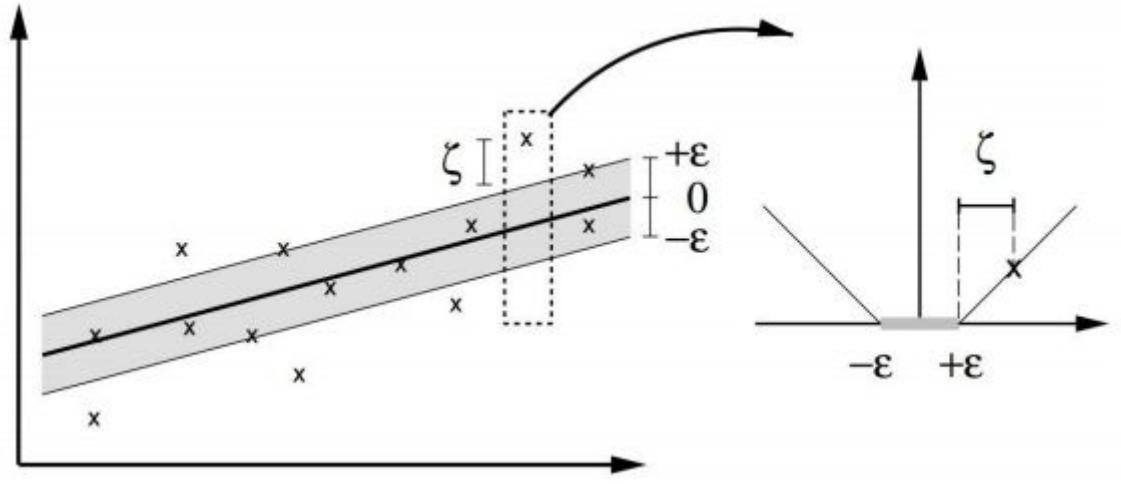
$$\xi_i, \underline{\xi}_i \geq 0$$

Nas equações apresentadas, *C* é uma constante que realiza a tarefa de regularização entre *h* e a margem, e representa o quanto os desvios são tolerados (ARAUJO, 2015). Os objetos que sofrem regularização entre as margens $-\epsilon$ e $+\epsilon$ são somente os que se localizam fora das margens, em conformidade com a função de perda ε-*intensive* (SMOLA e SCHOLKOPF, 1998):

$$|\xi|_\varepsilon := \begin{cases} 0 & , se \quad |\xi| \leq \epsilon \\ |\xi| - \epsilon & , caso\ contrário \end{cases} \tag{34}$$

A Figura 11 reproduz a Equação 34 de forma gráfica:

Figura 11 - Função de perda ε-intensive



Fonte: Smola e Scholkopf (1998)

Da mesma forma que nas SVMs para classificação, neste caso também monta-se o problema dual equivalente ao anterior por meio da função de Lagrange, anulando assim o

resultado das derivações parciais e substituindo as sentenças resultantes na equação lagrangiana inicial (FACELI et al. 2011). Este autor ainda afirma que para realizar regressões não lineares pode-se recorrer ao uso de Kernels, que possibilitam o mapeamento dos objetos para um espaço de características, no qual a função linear mais regular e com baixo erro de treinamento é encontrada.

$$Maximizar_{\alpha,\underline{\alpha}} -\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i-\underline{\alpha_i})(\alpha_j-\underline{\alpha_j})K(x_i,\,x_j)-\varepsilon\sum_{i=1}^{n}(\alpha_i+\underline{\alpha_i})+\sum_{i=1}^{n}y_i(\alpha_i-\alpha_i) \quad (35)$$

$$\text{Com as restrições: } \begin{cases}\sum_{i=1}^{n}(\alpha_i-\underline{\alpha_i})=0\\ \alpha_i,\,\underline{\alpha_i}\in[0,\,C]\end{cases} \quad (36)$$

Nas equações expostas, α_i e $\underline{\alpha_i}$ caracterizam as variáveis de Lagrange e K é a função kernel, que deve atender as condições de Mercer, ou seja, atender aos tipos de kernel aplicados à SVR apresentados na Tabela 1. As variáveis de Lagrange relacionadas aos objetos dentro da margem entre $-\varepsilon$ e $+\varepsilon$ são nulas enquanto os outros casos representam os SVs (FACELI et al., 2011).

2.5 Trabalho Relacionado

Ao longo do levantamento bibliográfico foi identificada uma pesquisa que contribuiu com a hipótese abordada neste trabalho. Essa pesquisa explana sobre o uso de algoritmos de aprendizagem de máquina de última geração para desenvolver um modelo preditivo preciso e robusto da dengue. A Seção 2.5.1 discute sobre o desenvolvimento deste trabalho relacionado.

2.5.1 Developing a dengue forecast model using machine learning: A case study in China

Neste trabalho, os autores Guo et al. (2017) desenvolveram um modelo preditivo de dengue para a província de Guangdong, na China. Os dados utilizados para este modelo foram coletados durante 2011-2014, sendo eles: casos semanais da dengue, consultas de pesquisa do Baidu e fatores climáticos (temperatura média, umidade relativa do ar e precipitação).

Para encontrar um modelo de previsão robusta e preciso, os autores do trabalho avaliaram diferentes algoritmos de aprendizagem de máquina para identificar um modelo ideal para previsão de dengue. Os algoritmos utilizados como modelos candidatos para prever incidência de dengue foram: regressão vetorial de suporte (SVR), modelo de regressão linear descendente, algoritmo de regressão forçada por gradiente (GBM), modelo de regressão

binomial negativo (NBM), menor encolhimento absoluto e regressão linear do operador de seleção (LASSO) e modelo aditivo generalizado (GAM).

Os modelos candidatos foram comparados e validados utilizando quatro cenários (GUO et al., 2017):

- Inicialmente, foram utilizados como dados de treinamento os dados de casos de dengue semanais, obtidos pela vigilância da dengue na china a partir da 1ª semana de 2011 até a 41ª semana de 2014, em Guangdong. Com isso, foi analisado a precisão preditiva de cada modelo ao longo de um horizonte temporal de 12 semanas e comparou seu desempenho;
- Em segundo lugar, com objetivo de avaliar o desempenho dos modelos de previsão do surto de dengue, em 2014, foi utilizado o RMSE (Raiz do Erro Quadrático Médio) para avaliar as diferenças entre os valores previstos por um modelo e os valores reais. Essa estratégia possibilitou avaliar o desempenho de precisão dos modelos com os dados entre a 35ª e 46ª semana (compreendendo o pico de incidência de dengue referente ao surto de 2014);
- Logo após, a fim de avaliar a capacidade dos modelos em rastrear monitorar a situação da dengue, os autores aplicaram uma abordagem de previsão fora da amostra para realizar previsões com uma semana de antecedência para obter estimativas quase em tempo real para as cidades estudadas em Guangdong;
- Por último, os modelos estudados foram validados utilizando dados da vigilância da dengue e consulta de busca na internet de cinco outras grandes províncias.

Ao fim dos procedimentos de comparação e validação entre os modelos candidatos, verificou-se que o modelo SVR apresentou os menores valores de RMSE, independentemente da cidade. Além disso, a precisão da previsão do modelo SVR aumentou com o aumento do valor do parâmetro *C,* e em seguida, rapidamente convergiu para um nível consistente, apresentando a boa capacidade preditiva de estabilidade do modelo. A Figura 12 apresenta a precisão preditiva relativa à incidência de dengue e da avaliação de adequação do ajuste para cada modelo.

Figura 12 - Comparação do desempenho do modelo e qualidade de ajuste para modelos de regressão

Measure	Prediction period	City	Model					
			SVR	Linear	LASSO	GAM	GBM	NBM
RMSE	The last 12 weeks	Guangzhou	16.2576*	109.9521	150.9228	218.0674	413.2917	182.2022
		Foshan	1.0483*	42.6509	25.9806	21.7453	47.6923	88.4364
		Zhongshan	0.3537*	3.7104	3.7638	4.7373	7.0461	4.3282
		Zhuhai	0.5717*	3.9115	3.9045	2.7538	6.7354	3.9376
		Shenzhen	0.8045*	6.1420	6.4693	12.0565	8.6777	5.0949
		Other cities studied	0.2681*	2.3806	2.0621	4.4973	3.4527	2.3305
	Outbreak period	Guangzhou	95.9668*	2204.7680	1378.6220	3215.8340	2691.7620	1764.1030
		Foshan	16.0143*	173.7577	181.8552	293.1263	223.1956	411.1545
		Zhongshan	1.1039*	89.4721	19.1110	78.5326	46.2534	39.9386
		Zhuhai	1.3978*	24.2412	25.9709	32.9678	38.1410	13.8852
		Shenzhen	1.6497*	26.8269	29.0679	43.6315	43.9624	18.4250
		Other cities studied	0.7876*	16.3118	14.9820	18.3629	15.3275	26.4680
R-squared	The last 12 weeks	Guangzhou	0.9990§	0.8513	0.9602	0.9315	0.5796	0.9411
		Foshan	0.9992§	0.7413	0.7142	0.7066	0.6054	0.6402
		Zhongshan	0.9948§	0.7932	0.9659	0.7416	0.9665	0.7704
		Zhuhai	0.9996§	0.7457	0.9699	0.9416	0.8287	0.7232
		Shenzhen	0.9983§	0.8307	0.8296	0.6099	0.7137	0.8423
		Other cities studied	0.9963§	0.5709	0.6463	0.6796	0.5620	0.5315
	Outbreak period	Guangzhou	0.9438§	0.8121	0.8170	0.8109	0.7736	0.9765
		Foshan	0.9441§	0.6670	0.6481	0.6794	0.5084	0.5748
		Zhongshan	0.9730§	0.6888	0.9039	0.6989	0.7929	0.9277
		Zhuhai	0.9804§	0.7074	0.8159	0.7329	0.7167	0.8946
		Shenzhen	0.9735§	0.6789	0.6278	0.6691	0.4033	0.7937
		Other cities studied	0.8865§	0.3081	0.2106	0.3361	0.1924	0.4224

* This indicates the values of RMSE of the SVR model were smallest.

§ This indicates the values of R-squared of the SVR model were largest.

https://doi.org/10.1371/journal.pntd.0005973.t001

Fonte: Guo et al. (2017)

Assim, o algoritmo SVR foi escolhido neste estudo para desenvolver a ferramenta para prever surtos de dengue na China (GUO et al., 2017). Os resultados deste estudo relacionado são importantes para dois âmbitos: **social**, por auxiliar o governo na identificação prévia de iniciativas necessárias para fortalecer o controle da dengue; e **acadêmico**, por disponibilizar uma pesquisa de comparação de desempenho entre diversos algoritmos de previsão de dados.

3 MATERIAIS E MÉTODOS

Em conformidade com os objetivos do trabalho, esta pesquisa possui como objeto metodológico a pesquisa aplicada de natureza qualitativa, objetivo exploratório e procedimento experimental, uma vez que este trabalho buscou utilizar o algoritmo de Regressão de Vetores de Suporte (*Support Vector Regression* - SVR) para criar um modelo computacional para predição do risco de dengue em Palmas - TO. Esta seção apresenta os materiais utilizados e a metodologia adotada para o desenvolvimento deste trabalho.

3.1 Local e Período de Realização da Pesquisa

A pesquisa foi realizada na cidade de Palmas - TO, mais especificamente no Centro Universitário Luterano de Palmas, no período de fevereiro de 2018 a dezembro de 2018.

3.2 Objeto de Estudo

A base de dados que foi utilizada neste projeto foi obtida a partir de um trabalho realizado em Cavalcante (2013). Essa base de dados compreende 2160 registros de pacientes notificados/diagnosticados com dengue no município de Palmas - TO. Cada registro desta base de dados é composto pelas seguintes variáveis:

- relacionados à pessoa
    - sexo
    - grau de escolaridade
- relacionados ao ambiente público:
    - qualidade do asfalto
    - presença de terrenos baldios
    - condições dos terrenos baldios
    - presença de bueiros
    - tipo de abastecimento de água
    - tipo de sistema de esgoto
    - presença de coleta pública do lixo e sua frequência
- relacionados ao ambiente domiciliar
    - frequência da limpeza nos locais de armazenamento de água
    - presença de lixeira
    - presença de focos de Aedes aegypti com notificação pela vigilância epidemiológica

    - frequência de limpeza de quintal
    - frequência de observação de calhas
- relacionados aos serviços de saúde
    - presença de mobilização da sociedade na quadra
    - ocorrência de casos de dengue
    - ocorrência de notificação e/ou confirmação dos casos
    - procura pelos serviços de saúde após a notificação
    - motivo da não procura pelos serviços de saúde
    - tempo de demora no diagnóstico

3.3 Materiais

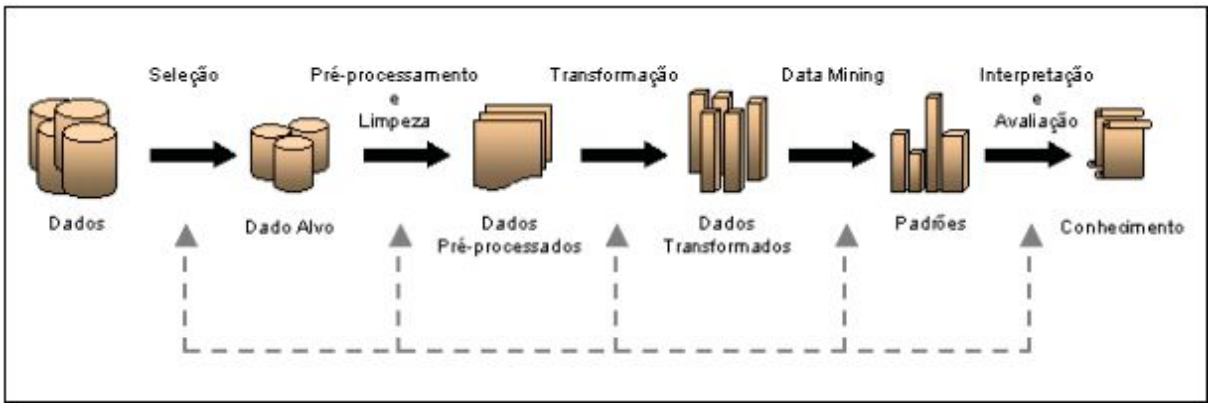
Para o desenvolvimento do modelo computacional para predição, foi utilizada a implementação do SVR disponibilizada pela biblioteca Scikit-learn, que segundo Pedregosa et al. (2011) fornece implementações de algoritmos de aprendizagem de máquina puramente em linguagem python. Foi utilizado também, o microframework Flask (RONACHER, 2018), responsável neste trabalho por fornecer um aplicativo para comunicação entre o modelo e o *frontend*.

Com a finalidade de apresentar os dados em uma interface gráfica, ou seja, desenvolver um *frontend*, foi utilizado o framework Angular (ANGULAR, 2018), que é uma plataforma e estrutura com diversas bibliotecas, sendo algumas básicas e outras opcionais, para criar aplicativos *frontend* utilizando tecnologias como HTML e TypeScript.

3.4 Procedimentos

Buscando atingir os objetivos do trabalho foi executado o processo de extração do conhecimento - KDD, proposto por Fayyad et al. (1996). O processo KDD, conforme apresentado na Figura 13, possui 5 etapas sendo elas: Seleção, Pré-processamento, Transformação, Data Mining, Interpretação e Avaliação.

Figura 13 - O ciclo do processo KDD



Fonte: (PRASS, 2012)

Inicialmente na etapa de Seleção, foram escolhidos os dados apenas das quadras do plano diretor sul e plano diretor norte da cidade de Palmas - TO, a fim de minimizar os dados que serão utilizados no modelo computacional.

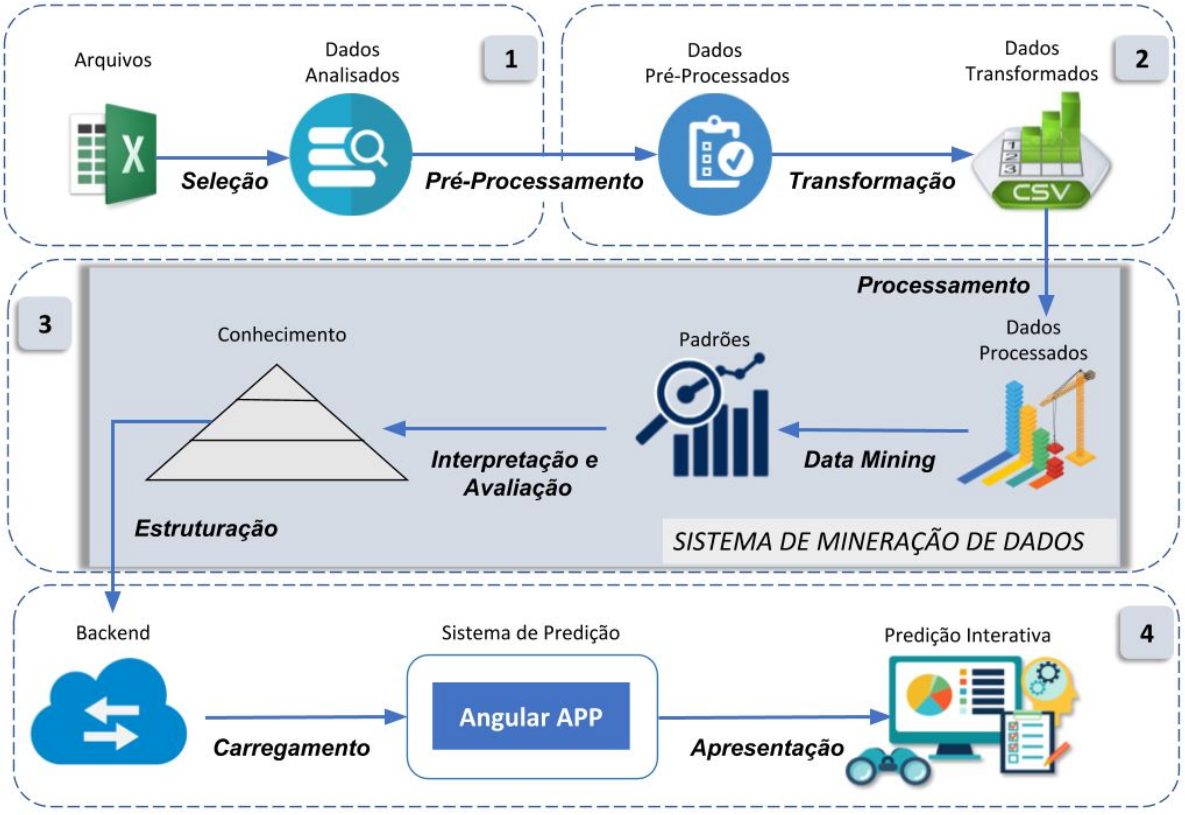
Na etapa de Pré-processamento foram removidas todas as informações que não foram necessárias para o modelo computacional. Após ser finalizada a etapa de Pré-processamento, será iniciada a etapa de Transformação, onde a base de dados foi transformada para o formato que é aceito pelo algoritmo SVR, que é o formato csv.

Na etapa de Data Mining são realizados os procedimentos para verificar a correlação de cada variável em relação ao risco de dengue de determinada localidade. Ao término do Data Mining será iniciada a última etapa: Interpretação e Avaliação. Esta etapa compreende os resultados deste trabalho, que serão organizados para carregamento em uma plataforma de predição e visualização de dados.

4 RESULTADOS E DISCUSSÃO

Em virtude do objetivo de aplicar o algoritmo SVR para predição de dados de dengue neste trabalho, foi definida um diagrama de visão geral do processo de predição. Este diagrama foi baseado no processo KDD (apresentado na Seção **3.4**), com a finalidade de dispor uma melhor visualização dos processos realizados para desenvolvimento do trabalho, bem como uma explicação mais detalhada destes processos. A visão geral do processo é apresentado na Figura 14.

Figura 14 -Visão Geral do Processo



A Figura 14 apresenta uma visão geral do sistema desenvolvido, demonstrando a comunicação entre as partes deste sistema. Buscando uma melhor explicação sobre as etapas apresentadas na Figura 14, estas foram divididas em quatro módulos, sendo eles: **módulo 1** (Figura 14-1) que representa a base de dados; **módulo 2** (Figura 14-2) representando a preparação dos dados; **módulo 3** (Figura 14-3) incluindo as etapas da aplicação, que são a mineração dos dados, aplicação do algoritmo SVR, interpretação e avaliação dos resultados dessa mineração e estruturação desses resultados; e por fim, o **módulo 4** (Figura 14-4) apresentando as etapas para apresentação dos dados.

Conforme apresentado na Figura 14, a fonte dos dados utilizados para realizar predição é de planilhas do excel, mais especificamente dos resultados do trabalho de Cavalcante (2013). No contexto deste trabalho, os valores a serem considerados no modelo estão descritos na Seção **3.2**.

Com a fonte dos dados definida, foi necessário realizar o procedimento de preparação dos dados (módulo 2 - Figura 14-2) para adaptar estes ao formato aceito pelo sistema, criando um arquivo no formato *csv* para ser utilizado na aplicação. A preparação dos dados foi realizada de forma manual por meio de funcionalidades do excel.

A partir do arquivo *csv* criado, a aplicação realizou a leitura dos dados e iniciou a tarefa de busca por padrões entre os dados (módulo 3 - Figura 14-3). Em seguida é encontrado o "conhecimento" por meio de interpretação e avaliação entre os padrões encontrados. Os resultados são estruturados em um *backend* para serem acessíveis por uma aplicação web de predição de dengue.

4.1 Base de Dados e Preparação dos Dados

O módulo Base de Dados traduz-se na etapa de obtenção de dados de dengue, realizado por meio da etapa de Seleção**,** que resultou nas variáveis descritas na Seção **3.2,** totalizando 2160 registros**.** Essa etapa possui como resultado um artefato denominado Dado Analisado, finalizando assim o módulo Base de Dados e iniciado o módulo Preparação de Dados.

O Módulo Preparação de Dados inicia-se com a etapa de Pré-processamento, na qual foi realizada a categorização dos dados para valores numéricos, uma vez que estes estavam em formato de texto, bem como a remoção e tratamento dos dados nulos. A remoção e tratamento dos dados nulos consistiu na eliminação dos registros em que existiam pelo menos uma coluna (variável) sem registro. Ao final desta etapa é gerado um artefato denominado Dado Pré-processado.

Após a conclusão da etapa de Pré-processamento, foi iniciada a etapa de Transformação, que consiste na transformação do artefato Dado Pré-processado em um arquivo denominado Dados Transformado, no formato csv com os dados separados por ponto e vírgula (";"), com os valores de cada variável. A primeira linha deste arquivo (cabeçalho do arquivo) contém o nome de cada uma das variáveis utilizadas. A Figura 15 demonstra a estrutura de visualização do arquivo Dados Transformados.

Figura 15 - Exemplo do arquivo de dados.

	A	B	C	D	E	F	G	H	I	J
1	sexo	escolaridade	asfalto	terrenos baldios condicoes	bueiros	agua	armazena	coleta lixo frequencia	lixeira	esgoto
2	1	3	0	1	1	0	0	1	0	1
3	1	2	0	1	1	0	0	1	1	1
4	0	2	1	2	1	0	0	1	1	1
5	1	2	1	2	1	0	0	1	1	1
6	0	3	2	1	1	1	0	1	0	1
7	0	3	2	0	1	0	0	1	0	1
8	0	3	2	0	1	0	0	1	0	1
9	0	2	2	1	1	0	0	1	0	1
10	0	3	1	1	1	0	1	1	1	1
11	1	2	1	1	0	0	2	1	1	1
12	1	3	1	1	0	0	2	1	1	1
13	0	3	2	0	0	0	0	1	1	1
14	0	3	2	1	1	0	0	1	1	1
15	0	3	2	0	0	0	0	1	0	1
16	0	3	1	1	1	0	2	1	0	1
17	0	3	1	0	0	0	3	1	0	1
18	0	3	0	1	0	0	2	1	1	1
19	0	3	1	1	1	0	2	1	0	1
20	0	3	2	1	1	0	2	1	1	1
21	0	1	2	1	1	0	0	1	0	1
22	0	1	1	1	1	0	2	0	1	1

A Figura 15 apresenta um exemplo de visualização do arquivo gerado no programa Excel. Após a conclusão da etapa Transformação, o arquivo Dado Transformado obtido é fornecido para a aplicação, onde é gerado o modelo de predição de dados. Os detalhes realizados pela aplicação são demonstrados na Seção 4.2.

4.2 Aplicação

A partir do arquivo gerado pelo módulo Preparação dos Dados, a aplicação efetua a etapa de Processamento. Esta etapa é realizada de forma automática (por meio de um algoritmo) antes dos dados serem minerados. Este algoritmo é fornecido pela biblioteca pandas do python, possibilitando realizar a leitura dos dados e manipulação sobre estes. Esta seção apresenta a aplicação da mineração dos dados, análise de correlação entre as variáveis, construção do modelo computacional.

4.2.1 Mineração dos dados

Esta seção apresenta a aplicação da análise de correlação entre as variáveis, bem como aplicação do algoritmo de predição (SVR) a partir da descoberta de padrões sobre os dados.

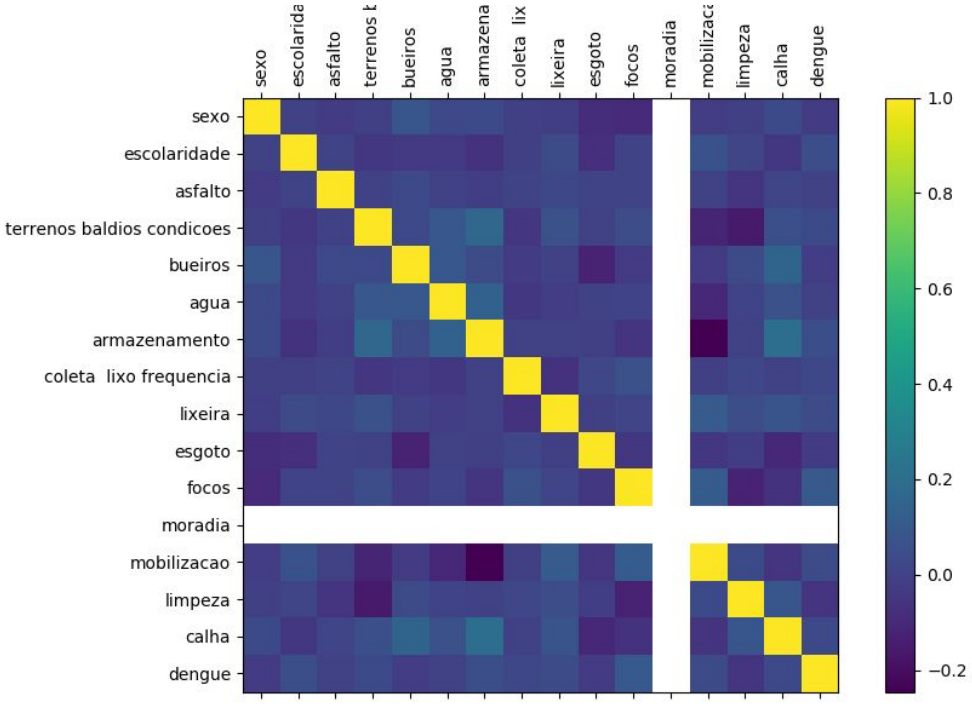
4.2.1.1 Análise de Correlação

Após o processamento dos dados, foi realizado a análise de correlação entre as variáveis, a fim de identificar as variáveis mais relevantes para o modelo computacional. Para tanto, foi utilizado o método **corr()** fornecido pela biblioteca **pandas** do python. Este método calcula o nível de correlação linear entre duas variáveis, delimitando este nível com valores situados entre -1,0 e 1,0. Neste trabalho, foi adotado o coeficiente de correlação denominado *Pearson,* que caracteriza os valores entre -1,0 e 1,0, como:

- 1,0: representa uma correlação perfeita positiva entre duas variáveis;
- -1,0: significa uma correlação perfeita negativa entre duas variáveis. Ou seja, se uma aumenta, a outra diminui; e
- 0: significa que as duas variáveis não dependem linearmente uma da outra. Entretanto, pode existir uma dependência “não linear” entre elas.

Ao se obter o resultado da aplicação do método **corr()**, é possível gerar um gráfico para visualização deste resultado, por meio da biblioteca **matplotlib** da linguagem python. A Figura 16 apresenta a matriz de correlação de todas as variáveis (dependente e independentes, onde a primeira representa a variável dengue) utilizadas neste trabalho.

Figura 16 - Matriz de correlação.



De acordo com a Figura 16, a matriz de correlação apresenta a correlação entre todas as variáveis, dispostas de forma vertical e horizontal na matriz. O ponto de cruzamento entre duas variáveis é caracterizado por um quadrado representado por uma cor que pode variar entre amarelo e azul. Conforme o gráfico de gradiente localizado no lado direito da Figura 16, é possível notar que os valores de correlação tendem entre 1.0 (representado pelo tom mais amarelo) e -0.2 (representado pelo tom mais azul escuro). Desta forma, obteve-se a seguinte tabela (Tabela 4) com os valores de correlação entre as variáveis dependentes e variável independente.

Tabela 4 - Correlação entre variáveis independentes e dengue

ordem	variável independente	Correlação variável independente X dengue	Valores Positivos
1	focos	0,101974	0,01039869668
2	limpeza	-0,058712	0,003447098944
3	armazenamento	0,054917	0,003015876889
4	escolaridade	0,052477	0,002753835529
5	mobilização	0,042861	0,001837065321
6	lixeira	0,038859	0,001510021881
7	condições de terrenos baldios	0,032116	0,001031437456
8	calha	0,030778	0,000947285284
9	esgoto	-0,027633	0,000763582689
10	sexo	-0,022521	0,000507195441

11	frequência de coleta lixo	0,022248	0,000494973504
12	bueiros	-0,020173	0,000406949929
13	asfalto	0,001346	0,000001811716
14	água	-0,000808	0,000000652864

A Tabela 4, apresentada acima, lista os resultados em ordem decrescente na coluna "Valores Positivos" (representado o quadrado do resultado da correlação de cada variável em relação a dengue), a fim de se obter as variáveis com melhor correlação em relação a variável dengue. Neste caso, foi escolhido aproximadamente 50% das variáveis visando escolher as variáveis com os maiores níveis de correlação dentro o conjunto de variáveis. Desta forma, resultou-se no seguinte conjunto de variáveis para aplicação no modelo de predição de dengue: *focos, limpeza, armazenamento, escolaridade, mobilização, lixeira, condições de terrenos baldios e calha*. Visando explicar com mais detalhes o processo de construção do modelo de predição de dengue, a **Subseção 4.2.1.2** apresenta a aplicação do algoritmo SVR para desenvolvimento deste modelo.

4.2.1.2 Aplicação do SVR

A execução da etapa de Data Mining iniciou-se com aplicação do algoritmo SVR, fornecido pela biblioteca *scikit learn*. O tipo de aprendizagem aplicada para este algoritmo é a supervisionada. Isto devido ser fornecido ao algoritmo um conjunto de dados de entrada (variáveis independentes) que possuem uma relação com uma variável de saída (variável dependente dengue).

O algoritmo SVR é aplicado neste trabalho com a função *kernel* denominada RBF, uma vez que esta teve uma menor taxa de erro comparada às demais (apresentadas na **Seção 2.3.2**). Os dados da base de dados foram divididos em 70% para treino e 30% para teste visando treinar dados do modelo com os dados de treino, e realizar a predição com os dados de teste. O trecho de código apresentado, Tabela 4 a seguir, é referente a implementação do algoritmo SVR, que realiza tarefa de aprendizagem supervisionada a partir de um conjunto de dados fornecidos.

```
1. svr = SVR(kernel='rbf')
2. x_train, x_test, y_train, y_test = train_test_split(array_x,
array_y,test_size=0.30,random_state=0)
3. clf = svr.fit(x_train, y_train)
4. pred_svr_y = svr.predict(x_test)
5. scores = cross_val_score(clf, array_x, array_y, cv=5)
6. print('{};{}'.format(scores.mean(), clf.score(x_test,y_test)))
```

Conforme este trecho de código, o SVR possui o parâmetro a função kernel do tipo RBF, e os dados passados para o algoritmo são tratados por meio da função *train_test_split(),* fornecida também pela biblioteca scikit learn. Esta função recebe como parâmetros os dados da base de dados utilizada no trabalho, e os divide em 30% para teste e 70% para treinamento. Para ajustar os dados de treinamento, e consequentemente fazer com que o algoritmo aprenda sobre estes dados, foi utilizado o método *fit()* do SVR.

Para testar o desempenho do algoritmo, foi utilizado o conjunto de dados de teste, o qual é um conjunto separado do conjunto de treinamento. Por exemplo, para realizar a predição de a partir de uma nova entrada, é utilizada a função *predict()* do SVR recebendo os dados do conjunto de teste. A fim de avaliar o desempenho de um modelo treinado pelo algoritmo, utilizou-se o método de validação cruzada denominado *K-Fold* por meio da função *cross_val_score()* e o método *score()*, ambos fornecidos pela biblioteca scikit learn.

A partir destes métodos, foi possível analisar o desempenho de um modelo, e então realizar a busca pelo melhor modelo de predição com base nas variáveis da base de dados. A subseção a seguir apresenta os passos para busca pelo melhor modelo de predição para este trabalho.

4.2.1.3 Busca pelo melhor modelo de predição

Com a base de dados transformada e o algoritmo SVR estruturado, foi iniciado o processo de busca pelo melhor modelo de predição de dengue na cidade de Palmas. Os processos de mineração de dados e interpretação e a avaliação foram repetidos várias vezes, realizando todas combinações possíveis entre todas as variáveis da base de dados (*escolaridade, condições de terrenos baldios, armazenamento, lixeira, focos, mobilização, limpeza e calha*) em relação a variável de *dengue*.

Tais combinações foram realizadas pelo método *combinations()* do módulo itertools do python. Em cada conjunto de combinações resultantes - dentro de um conjunto de oito variáveis - foi verificado o score de erro, para que assim fosse identificado o melhor modelo de predição. A Tabela 5 apresenta os modelos de cada conjunto de combinações com as menores taxas de score.

Tabela 5 - Modelos de predição de dengue com menores score

Modelos	Score

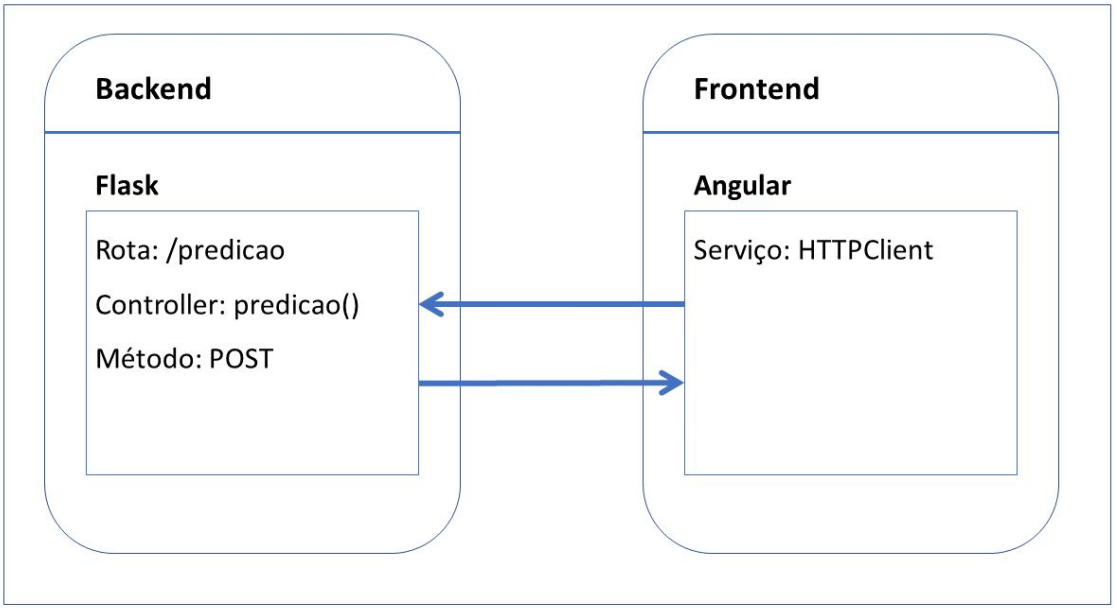
escolaridade-armazenamento-lixeira-focos-mobilizacao-limpeza-calha	-0,09342547946
escolaridade-terrenos baldios condicoes-armazenamento-lixeira-focos-mobilizacao-limpeza-calha	-0,1123440207
escolaridade-terrenos baldios-armazenamento-mobilizacao-limpeza-calha	-0,1215834574
escolaridade-terrenos baldios-armazenamento-limpeza-calha	-0,1670830127
escolaridade-terrenos baldios-armazenamento-lixeira	-0,2138566125
escolaridade-terrenos baldios-armazenamento	-0,2350399826
escolaridade-terrenos baldios	-0,288

Tendo em vista a utilização de um modelo de predição de dados de dengue em Palmas, foi selecionado o modelo com sete variáveis (*escolaridade, armazenamento, lixeira, focos, mobilização, limpeza e calha*) devido possuir um menor score em relação aos demais modelos apresentados na Tabela 5. Com o modelo selecionado, foi aplicado o algoritmo SVR apenas para o mesmo, a fim de encontrar erro médio do algoritmo bem como a função de predição para novas entradas.

4.3 Software de Predição

Tendo em vista a disponibilização de uma interface gráfica, na qual um usuário possa inserir dados de uma pessoa para predição de dengue, foi desenvolvido um software de predição. A Figura 17 a seguir, apresenta uma visão geral sobre esse software.

Figura 17 - Visão geral do software de predição.

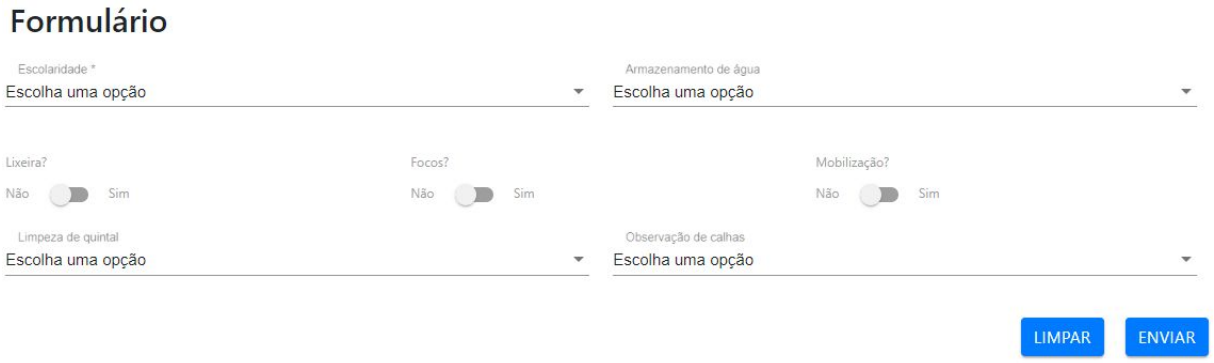


O software de predição possui um *backend* desenvolvido com o framework Flask, e um *frontend* desenvolvido com o framework Angular. No *backend* encontra-se o modelo de predição em um controller denominado *predicao()* que pode ser requisitado pela rota */predicao* por meio de requisições do tipo POST. E no *frontend* existe um serviço do tipo HTTPClient que possibilita a comunicação com o *backend*.

Quando o *frontend* realiza uma requisição para o *backend*, este último retorna os dados no formato JSON (JavaScript Object Notation). Esses dados, são referentes a predição realizada para os valores passados pelo *frontend*, neste caso, um representação numérica para cada resposta do formulário contido no *frontend*.

O usuário do software de predição pode inserir informações de um determinado indivíduo a fim de verificar o valor predição de dengue para esse indivíduo. A Figura 18 apresenta o formulário no qual o usuário pode inserir as informações e realizar predição.

Figura 18 - Formulário de informações sobre um indivíduo.



Os campos do formulário apresentado são referentes às variáveis do modelo de predição selecionado. Após o usuário enviar as informações inseridas no formulário, o sistema apresenta uma área contendo o resultado da predição para as informações enviadas, bem como a taxa de erro do algoritmo. A Figura 19 apresenta o resultado da predição determinados valores inseridos no formulário.

Figura 19 - Exemplo de resultado da predição

O resultado da predição de novas entradas realizada pelo modelo computacional é apresentado para o usuário como um número. Esse valor é coerente com a escala de dengue definida na transformação da base de dados (dengue sim: 1; dengue não: 0). A taxa de erro do algoritmo, também apresentada na área “Resultado da Predição”, é referente ao erro médio de predição dos dados de teste do modelo computacional.

5 CONSIDERAÇÕES FINAIS

Neste trabalho foi apresentado um modelo computacional para predição de dados de dengue e um software web para realizar predições dessa doença em tempo real. Os resultados alcançados com a aplicação do algoritmo SVR para predição de dados de dengue apresentam

um modelo de predição de dengue composto pelas variáveis independentes: escolaridade, armazenamento de água, lixeira, focos. mobilização, limpeza de quintal e observação de calhas.

Contudo, os resultados obtidos são pertinentes somente para o conjunto de dados utilizado, que foram dados de casos de dengue da cidade de Palmas - TO, adquiridos do trabalho de Cavalcante (2013). Esses resultados podem variar caso seja utilizado outras implementações do SVR, fornecido pela biblioteca Scikit-learn.

O software de predição que foi desenvolvido possui o formato de uma página web, que possibilita ao usuário enviar informações sobre uma pessoa a qual pretende predizer o nível do risco de dengue. Essa predição é possível devido ao modelo preditivo desenvolvido, que utilizou a base de dados de Cavalcante (2013).

Como trabalhos futuros considera-se importante a melhoria do processamento da base de dados, por exemplo, realizar o tratamento dos dados com o método de substituição de um valor nulo pela média da sua coluna. No entanto, para isso é necessário um estudo de categorização dos dados de Cavalcante (2013) a fim de se criar uma escala para identificar a qual categoria pertence determinada média. Isso para que se possa ter uma base de dados com uma acurácia melhor.

Outro tópico importante para trabalhos futuros é a utilização de mais variáveis que possam influenciar na variável dengue, visto que os modelos com menores taxas de erros possuíam maiores números de variáveis.

A partir dos estudos e observações dos resultados obtidos é possível concluir que o algoritmo SVR pode ser utilizado para a predição do risco de dengue para uma pessoa de uma determinada região.

REFERÊNCIAS

ARAÚJO, Kevin Martins. **UTILIZAÇÃO DO ALGORITMO DE MÁQUINA DE VETORES DE SUPORTE (SVM) PARA PREDIÇÃO DE DADOS CLIMÁTICOS.** 2015. 96 f. Monografia (Especialização) - Curso de Ciência da Computação, Centro Universitário Luterano de Palmas, Palmas, 2015.

ALEXANDRE, Sueli de Fátima. Aprendizagem e suas implicações no processo educativo. **Ícone**: Revista de Letras, Goiás, v. 6, n. 1, p.51-60, jul. 2010. Disponível em: <http://www.revista.ueg.br/index.php/icone/article/view/5100>. Acesso em: 26 fev. 2018.

BURGES, Christopher J.c.. A Tutorial on Support Vector Machines for Pattern Recognition. **Data Mining And Knowledge Discovery**, Boston, p.121-167, 1998.

CARVALHO, J.V. et al. **Utilização de técnicas de "Data Mining" para o reconhecimento de caracteres manuscritos**. In: 14º Simpósio Brasileiro de Banco de Dados, Ceará, 2009. p. 235-249.

CAVALCANTE, Micheline Pimentel Ribeiro. **DISTRIBUIÇÃO ESPACIAL DA DENGUE NAS ÁREAS URBANAS E PERIURBANAS DE PALMAS DE 2008-2010, SEGUNDO ÓTICA GEOMÉDICA.** 210 f. Tese (Doutorado) - Curso de Doutorado em Ciências da Saúde, Universidade de Brasília, Brasília, 2013. Disponível em: <http://repositorio.unb.br/handle/10482/15101>. Acesso em: 02 out. 2017.

CHAMASEMANI, Fereshteh Falah; SINGH, Yashwant Prasad. **Multi-class Support Vector Machine (SVM) classifiers-An Application in Hypothyroid detection and Classification**. In: Sixth International Conference on Bio-Inspired Computing: Theories and Applications, 2011.

FACELI, Katti et al. **Inteligência Artificial:** Uma Abordagem de Aprendizado de Máquina. Rio de Janeiro: Ltc, 2011. 378 p.

FREUND, John E. **Estatística aplicada**: economia, administração e contabilidade. Tradução Claus Ivo Doering. 11.ed., Porto Alegre: Bookman, 2006, 535 p. Título original: Modern Elementary Statistics.

GOMES, Frederico Pimentel. **Curso de Estatística Experimental**. 14ª ed. rev. e amp. Piracicaba - SP, Editora F. Pimentel-Gomes, 2000.

GUNN, Steve. Support vector machines for classification and regression. Technical report, Image Speech & Intelligent Systems Group, University of Southampton, 1998.

Guo P, Liu T, Zhang Q, Wang L, Xiao J, Zhang Q, et al. 2017. **Developing a dengue forecast model using machine learning: A case study in China**. PLoS Negl Trop Dis 11(10): e0005973 Acesso em: 19 fev. 2018. Disponível em: <https://doi.org/10.1371/journal.pntd.0005973>.

HAIR Jr., J. F. et al. **Análise multivariada de dados**. São Paulo: Bookman, 2005

MAIA, Hugo Leite Florenço. **DETECÇÃO E RECONHECIMENTO FACIAL POR MEIO DE APRENDIZADO DE MÁQUINA.**2016. 50 f. TCC (Graduação) - Curso de Engenheiro de Redes de Comunicação, Universidade de BrasÍlia, Brasília, 2016.

MAVROFORAKIS, M.e.; THEODORIDIS, S.. A geometric approach to Support Vector Machine (SVM) classification. **Ieee Transactions On Neural Networks**, [s.l.], v. 17, n. 3, p.671-682, maio 2006. Institute of Electrical and Electronics Engineers (IEEE). http://dx.doi.org/10.1109/tnn.2006.873281. Disponível em: <https://ieeexplore.ieee.org/document/1629090/>. Acesso em: 26 fev. 2018.

Ministério da Saúde (Brasil). **Situação da dengue no Brasil**. 2007. Acesso em: 19 fev. 2018.

MITCHELL, T. **Machine Learning**. McGraw Hill. 1997. New York, USA.

OLIVERA, André Rodrigues et al. **Comparação de algoritmos de aprendizagem de máquina para construir um modelo preditivo para detecção de diabetes não diagnosticada - ELSA-Brasil: estudo de acurácia.** *Sao Paulo Med. J.* [online]. 2017, vol.135, n.3, pp.234-246. ISSN 1516-3180. Disponível em: <http://dx.doi.org/10.1590/1516-3180.2016.0309010217>. Acesso em: 20 fev. 2018.

PEDREGOSA et al.. S**cikit-learn: Machine Learning in Python**. JMLR 12, pp. 2825-2830, 2011.

PITOMBO, Cira Souza et al. Aplicação conjunta de modelos não paramétricos e paramétricos para previsão de escolha modal. **Journal Of Transport Literature**, Manaus, v. 9, n. 1, p.30-34, jan. 2015. FapUNIFESP (SciELO). http://dx.doi.org/10.1590/2238-1031.jtl.v9n1a6. Disponível em: <http://dx.doi.org/10.1590/2238-1031.jtl.v9n1a6>. Acesso em: 20 fev. 2018.

REZENDE, Bruno Ferreira; SILVA, Diogo Santos da. **Bioinformática**. Universidade Federal de Mato Grosso. Rondonópolis, MT. 2009

RIBEIRO, Andressa F et al. Associação entre incidência de dengue e variáveis climáticas. **Revista de Saúde Pública**, São Paulo, v. 40, n. 4, p.671-676, ago. 2006. Disponível em: <http://dx.doi.org/10.1590/S0034-89102006000500017>. Acesso em: 19 fev. 2018.

RUAS, Gabriel I. S. et al. **Previsão de Demanda de Energia Elétrica Utilizando Redes Neurais Artificiais e Support Vector Regression**. Curitiba. 2004.

SANTOS, Alcione Miranda dos et al. Usando redes neurais artificiais e regressão logística na predição da Hepatite A. **Revista Brasileira de Epidemiologia**, [s.l.], v. 8, n. 2, p.117-126, jun. 2005. FapUNIFESP (SciELO). http://dx.doi.org/10.1590/s1415-790x2005000200004.

Disponível em: <http://dx.doi.org/10.1590/S1415-790X2005000200004>. Acesso em: 20 fev. 2018.

SCHMITZ, Felipe Eduardo Bechert. **APLICAÇÃO DA TÉCNICA DE TEXT MINING PARA COMENTÁRIOS RELACIONADOS AO CONTEXTO DO TURISMO.** 2015. 47 f. TCC (Graduação) - Curso de Sistemas de Informação, Centro Universitário Luterano de Palmas, Palmas, 2015.

SMOLA, Alex J.; SCHÖLKOPF, Bernhard. **Support Vector Machines and Kernel Algorithms**. 20 mar. 2002.

SOUTO, Marcilio Carlos Pereira de. et al. **Técnicas de Aprendizado de Máquina para problemas de Biologia Molecular**, p 103–152. In: Minicursos de Inteligência Artificial, Jornada de Atualização Científica em Inteligência Artificial, XXIII Congresso da Sociedade Brasileira de Computação, 2003.

TADANO, Yara de Souza; UGA, Cássia Maria Lie; FRANCO, Admilson Teixeira. Método de regressão de Poisson: metodologia para avaliação do impacto da poluição atmosférica na saúde populacional. **Ambiente & Sociedade**, Campinas, v. 12, n. 2, p.241-255, dez. 2009.

VALADARES, Adriane Feitosa; C. FILHO, José Rodrigues; PELUZIO, Joênes Mucci. Impacto da dengue em duas principais cidades do Estado do Tocantins: infestação e fator ambiental (2000 a 2010). **Epidemiologia e Serviços de Saúde**, Brasília, v. 22, n. 1, p.59-66, mar. 2013. Instituto Evandro Chagas. http://dx.doi.org/10.5123/s1679-49742013000100006.

VAPNIK, Vladmir. **The Nature of Statistical Learning Theory**. New York: SpringerVerlag, 1995.

World Health Organization. **Dengue: guidelines for diagnosis, treatment, prevention and control**. 2009. Disponível em: <http://whqlibdoc.who.int/publications/2009/9789241547871_eng.pdf>. Acesso em: 19 fev. 2018.

APÊNDICES

ANEXOS