CENTRO UNIVERSITÁRIO LUTERANO DE PALMAS
Recredenciado pela Portaria Ministerial nº 3.607, de 17/10/05, D.O.U. nº 202, de 20/10/2005
ASSOCIAÇÃO EDUCACIONAL LUTERANA DO BRASIL

Kevin Martins Araújo

UTILIZAÇÃO DO ALGORITMO DE MÁQUINA DE VETORES DE SUPORTE (SVM) PARA PREDIÇÃO DE DADOS CLIMÁTICOS

Palmas - TO

2015

Kevin Martins Araújo

UTILIZAÇÃO DO ALGORITMO DE MÁQUINA DE VETORES DE SUPORTE (SVM) PARA PREDIÇÃO DE DADOS CLIMÁTICOS

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

Orientador: Prof. Dr. Edeilson Milhomem da Silva.

Palmas - TO

2015

Kevin Martins Araújo

UTILIZAÇÃO DO ALGORITMO DE MÁQUINA DE VETORES DE SUPORTE (SVM) PARA PREDIÇÃO DE DADOS CLIMÁTICOS

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

Orientador: Prof. Dr. Edeilson Milhomem da Silva.

Aprovada em: ___ de _____ de 2015.

BANCA EXAMINADORA

_____

Prof. Dr. Edeilson Milhomem da Silva
Centro Universitário Luterano de Palmas

_____

Prof. M.Sc. Jackson Gomes de Souza
Centro Universitário Luterano de Palmas

_____

Prof. M.Sc. Parcilene Fernandes Brito
Centro Universitário Luterano de Palmas

Palmas - TO

2015

RESUMO

O presente trabalho tem como objetivo aplicar o algoritmo de aprendizagem de máquina, denominado como, Máquina de Vetores de Suporte (SVM), para realizar a predição de dados climáticos. Os dados utilizados em todo o trabalho foram obtidos pelo site do inmet.gov.br, com informações climáticas da região de Palmas – TO. Foram aplicados conceitos de aprendizagem de máquina, para o estudo do funcionamento do algoritmo SVM, juntamente com os conceitos e técnicas de extração de conhecimento (KDD), buscando encontrar a melhor aplicação do SVM para a predição dos referidos dados.

PALAVRAS-CHAVE: Support Vector Machine, Regressão, Dados Climáticos.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

AM	Aprendizagem de Máquina
CEULP	CEntro Universitário Luterano de Palmas
EVP	Evaporação
IA	Inteligência Artificial
INS	Insolação
KDD	Knowledge-Discovery in Databases
KKT	Karush-Kuhn-Tucker
MMQ	Método dos Mínimos Quadrados
PREP	Precipitação
SV	Support Vector
SVC	Support Vector Classification
SVM	Support Vector Machine
SVR	Support Vector Regression
TCM	Temperatura Compensada Média
TMi	Temperatura Mínima
TMx	Temperatura Máxima
URM	Umidade Relativa Média

SUMÁRIO

1 INTRODUÇÃO

O presente trabalho propõe desenvolver o protótipo de um software que seja capaz de realizar a predição de dados climáticos, aplicando o algoritmo de Máquina de Vetores de Suporte (SVM).

A Máquina de Vetores de Suporte (SVM) é um algoritmo que aplica conceitos de aprendizagem de máquina e que a cada dia que passa vem recebendo uma maior atenção dos pesquisadores e estudantes que se interessam pela a área da Inteligência Artificial (IA) (LORENA e CARVALHO 2007), mais especificamente, os interessados pela subárea da IA chamada de Aprendizagem de Máquina (AM). A aplicação deste algoritmo proporciona resultados que são considerados melhores que os resultados obtidos por outros algoritmos de AM, como o algoritmo de Redes Neurais (FACELI, 2011).

A predição (ou previsão) climática é uma área de estudo pertencente à meteorologia (ciência que estuda atmosfera) e abrange conceitos sobre os eventos e acontecimentos que interferem nos valores climáticos de uma determinada região (BÍSCARO, 2007). Os valores climáticos mencionados anteriormente referem-se a valores de temperaturas, umidade do ar entre outros, em determinados períodos.

Para a implementação do protótipo do software, é utilizado o algoritmo da Máquina de Vetores de Suporte (SVM, do inglês *Support Vector Machine*) (RUSSELL e NORVIG, 2013) para realizar a predição dos valores climáticos. Este algoritmo proporciona a uma máquina aprender a partir de um conjunto de dados de entrada e saída, e desta maneira, consiga obter um modelo que possa realizar a predição de novos dados de saída para novas entradas (VAPNIK, 1995), e de análise de regressão. A mencionada análise é um método estatístico, em que é possível predizer o valor de uma determinada variável baseada em um conjunto de dados relacionados (LARSON e FARBER, 2010).

Desta forma, este trabalho está organizado da seguinte forma: a Seção 2 apresenta todo o referencial teórico utilizado para obter uma base de conhecimento para o desenvolvimento do trabalho. A Seção 3 apresenta os matérias e métodos utilizados no desenvolvimento do trabalho, a Seção 4 apresenta os resultados obtidos na execução do trabalho, na Seção 5 apresenta as considerações finais do trabalho, por fim finda-se o presente trabalho com o referencial bibliográfico.

2 REFERENCIAL TEÓRICO

Nesta seção são apresentados os conceitos que serão aplicados neste presente trabalho, tais como aprendizado de máquina, aprendizagem supervisionada, bem como o algoritmo de Máquina de Vetores de Suporte, procurando obter um melhor entendimento dos mesmos. Sendo assim, a Seção 2.1 apresenta os conceitos sobre aprendizagem de máquina e os seus tipos, seguindo pela Seção 2.2 que trata sobre o tipo de aprendizagem que será empregado neste trabalho. Ao final, os conceitos que envolvem o algoritmo de aprendizagem de máquina que será aplicado, sendo este, conhecido como Máquina de Vetores de Suporte (SVM) (Seção 2.3).

2.1. Aprendizagem de Máquina

Na definição do dicionário Michaelis, tem-se por aprendizagem, a ação, processo, efeito ou consequência de aprender qualquer ofício, arte ou ciência. Para os seres humanos a aprendizagem é um processo de mudança de comportamento obtido por meio da experiência construída por fatores emocionais, neurológicos, relacionais e ambientais (ALEXANDRE, 2010). Para a máquina a aprendizagem é a capacidade de melhorar o desempenho na realização de alguma tarefa por meio da experiência (MITCHELL, 1997), mas ela não pode ser influenciada por fatores emocionais, neurológicos, relacionais e ambientais como o ser humano. Experiência é o conhecimento ou aprendizado obtido pela máquina a partir dos conhecimentos já possuídos e das decisões tomadas.

Um exemplo de uma máquina que possui aprendizagem seria: uma máquina que possui sensores que leem as linhas centrais e as bordas de uma pista, e lê também os movimentos realizados por um humano em cada momento. A partir disso, a máquina irá aprender que quando a linha central possuir um deslocamento para a direita é o momento de virar o volante para direita, quando a linha central possuir um deslocamento para a esquerda é o momento de virar o volante para a esquerda. E a cada tarefa executada, sendo certa ou errada, a máquina irá aprender em qual momento deve-se realizar determinada tarefa. Isso é a aprendizagem por experiência, devido ao fato que a máquina adquiri conhecimento e os utiliza para tomar novas decisões, e a cada nova decisão ela memoriza esta decisão em sua "*base de conhecimento*" (experiência). Para que sistemas sejam capazes de realizar este tipo de tarefa é necessário aplicar os conceitos relacionados a aprendizagem de máquina (AM).

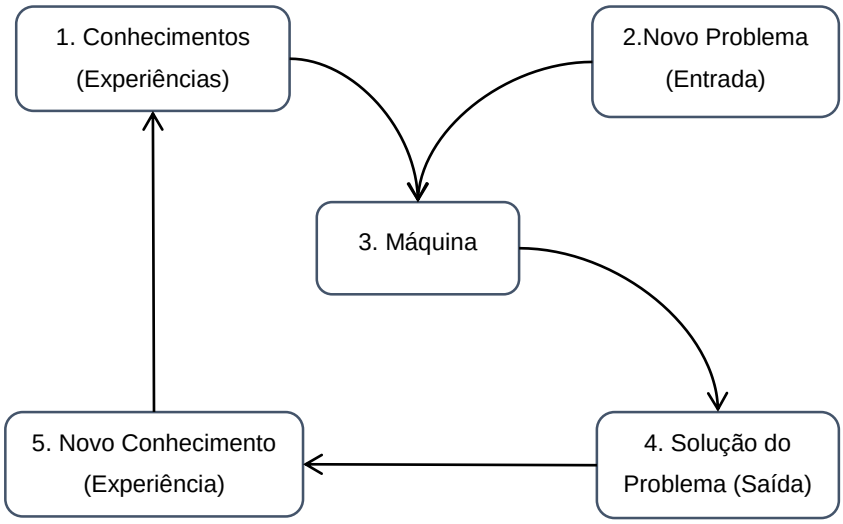
A aprendizagem de máquina é uma das áreas da Inteligência Artificial que tem por foco desenvolver sistemas capazes de tomar decisões baseadas no conhecimento acumulado

(LANGLEY e SIMON, 1995), e que consegue estar sempre aprimorando a base de conhecimentos a cada tomada de decisão, estar sempre adquirindo experiência.

Aprimorar a base de conhecimentos significa dizer que após a máquina ter gerada uma resposta (decisão), essa resposta se torna um novo conhecimento para ela, e se junta a base de conhecimentos (experiências) que já possuía, isso faz com que a máquina aumente (aprimore) sua base de conhecimentos (experiências), adquirindo mais experiência a cada resposta gerada (tomada de decisão).

A Figura 1 mostra, de forma resumida, como funciona o processo de aprendizagem de máquina.

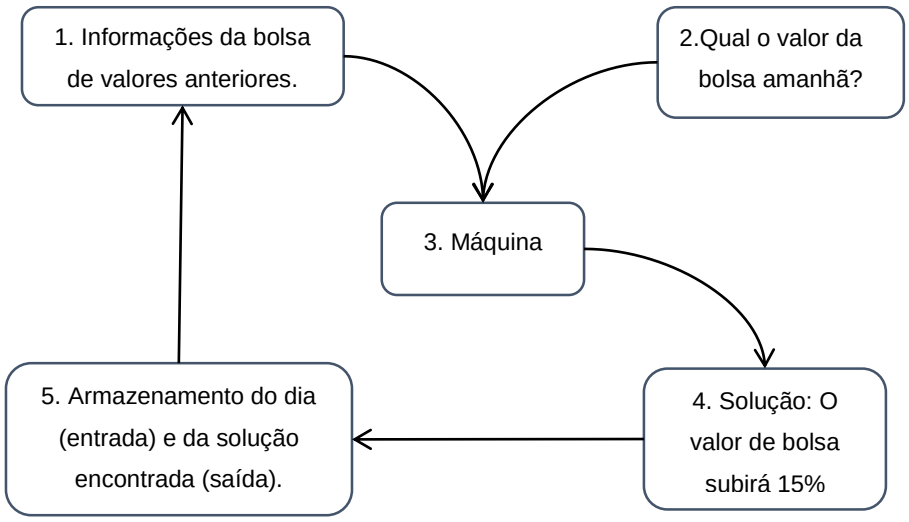
Figura 1 - Processo Simples de Aprendizagem



Fonte: adaptado de PRUDÊNCIO (2008)

De acordo com a Figura 1, a aprendizagem por uma máquina se inicia quando a máquina (Figura 1-3) recebe um novo problema (ENTRADA) (Figura 1-2), e a partir do conhecimento que ela possui (Figura 1-1) gera uma solução para o problema (SAÍDA) (Figura 1-4). Após ter gerado a saída, a máquina salva os dados (Figura 1-5) de entrada e saída no seu banco de conhecimentos (experiências), realizando o que chamado de aprendizado.

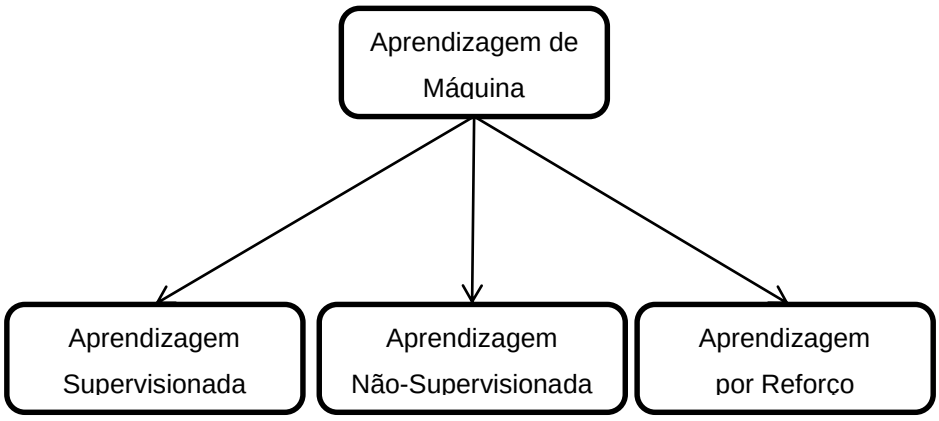
Figura 2 - Processo de aprendizagem de uma máquina que prediz o valor da bolsa de valores em um determinado dia



A Figura 2 exemplifica o processo de aprendizagem de uma máquina que prediz o valor da bolsa de valores. A partir de dados de valores de dias (ou messes) anteriores (Figura 2-1), a máquina prediz qual será o valor da bolsa (Figura 2-4) para um determinado dia (Figura 2-2). Após ter encontrado a solução para o problema proposto realiza o armazenamento do dia (entrada) e da solução encontrada (saída) (Figura 2-5) no seu banco de informações (Figura 2-1).

A área de aprendizagem de máquina é dividida em três tipos (RUSSELL e NORVIG, 2013): aprendizagem supervisionada, aprendizagem não supervisionada e aprendizagem por reforço.

Figura 3 - Tipos de Aprendizagem de Máquina (AM)



Na aprendizagem supervisionada a máquina recebe um conjunto de dados com pares de entradas e saídas corretas (que serão utilizadas como treinamento). Ela procura obter padrões e informações desses dados para que consiga entender como eles foram encontrados, e possa

fornecer a saída correta para entradas desconhecidas (LORENA e CARVALHO, 2007). Entradas desconhecidas são as novas entradas que não pertencem ao conjunto de experiências (conhecimentos) da máquina, que não se sabe qual a saída correspondente. O que define este tipo de aprendizagem como sendo aprendizagem supervisionada é devido ao fato de a máquina possuir para o seu treinamento tanto os dados de entrada quanto os de saída, isto é, pode-se dizer que ela possui as perguntas e também possui as respostas (CONDUTA e MAGRIN, 2010). Por exemplo, uma máquina que realiza o reconhecimento de faces. Dada a imagem de uma face (nova entrada), o sistema procura descobrir de quem é esta face (nova saída), dentro de um número finito de faces que o sistema possui (dentro do banco de conhecimento da máquina) (KOERICH, 2008).

Já na aprendizagem não supervisionada não ocorre o treinamento (por isso é chamada de não supervisionada), a máquina recebe dados como sendo as entradas, e aprende a representar (ou agrupar) os padrões e/ou tendências destes dados. Esse modelo costuma ser utilizado quando o objetivo é encontrar padrões ou tendências que ajudem no entendimento dos dados (SOUTO et al, 2003). Um exemplo da aplicação deste tipo de aprendizagem é em sistema de classificação de texto ou comentários. Neste sistema recebem-se determinados textos (dados de entrada), e procura os padrões e tendência que os definem e/ou distinguem. No final pode ser executado um agrupamento dos textos que tratam sobre o mesmo assunto e/ou separar os que possuem assuntos diferentes. Para os comentários o sistema poderia agrupa-los como comentários positivos e comentários negativos ou neutros.

A aprendizagem por reforço é baseada no conceito da tentativa e erro, determina-se um peso ou recompensa cada vez que o algoritmo acerta e uma punição a cada vez que o algoritmo erra. Em Russell e Norvig (2013) é descrito como exemplo para este tipo de aprendizagem a máquina que consegue jogar xadrez. A cada jogada correta ela recebe uma pontuação positiva, e a cada jogada incorreta (por exemplo, uma jogada que faça perder uma peça) recebe uma pontuação negativa. Ao final de cada jogo atribui-se uma pontuação positiva maior para máquina caso ela ganhe, informando-a que ela realizou a tarefa de forma correta, e caso ela perca, não se atribui nenhuma pontuação, ou atribui-se uma pontuação negativa.

A finalidade deste trabalho é realizar a predição de dados climáticos, para isso, serão utilizados conceitos e técnicas que estão relacionadas a aprendizagem supervisionada. Devido a isso a Seção 2.2 apresenta mais detalhes sobre este tipo de aprendizagem.

2.2. Aprendizagem Supervisionada

Na aprendizagem supervisionada, a máquina realiza a aprendizagem inicial a partir de um conjunto de dados de treinamento (ou conjunto de treinamento) (RUSSELL e NORVIG, 2013). Esse conjunto possui pares de dados de entradas e saídas (x_1, y_1), (x_1, y_2), (x_1, y_3), ..., (x_n, y_n), em que x representa as entradas (podendo ser um ou vários atributos) e y (um valor) representa as saídas.

Cada saída (y) é gerada a partir de uma função real (que é desconhecida) $y = f(x)$ (em que x corresponde aos valores das entradas) (RUSSELL e NORVIG, 2013). A máquina procura descobrir um modelo matemático que melhor represente o $f(x)$ real. Faceli et al. (2011) define uma função real como:

$$D = \{(x_i, f(x_i)), i = 1,2,\dots,n\} \quad (1)$$

onde:

D: um conjunto de observações de pares;

i: entrada dentre o conjunto de pares;

n: número total de pares de entradas e saídas;

f: função desconhecida;

x_i: representa a entrada de numero i.

Dentre o conjunto de entradas e saídas conhecidas, a máquina faz uma análise para encontrar uma função (que será chamada de $h(x)$) que mais se aproxime da função real ($f(x)$) e, diante disso, utiliza essa função $h(x)$ para que seja capaz de estimar novas saídas a partir de uma determinada entrada.

Este tipo de aprendizagem necessita que o analista auxilie a máquina no seu treinamento e nos testes (BARREIRA, 2013). O analista é responsável pela divisão do conjunto de dados em duas partes, sendo uma utilizada para o sistema treinar (tentar encontrar uma função $h(x)$ que mais se aproxime da função real $f(x)$), e a outra utilizada como teste.

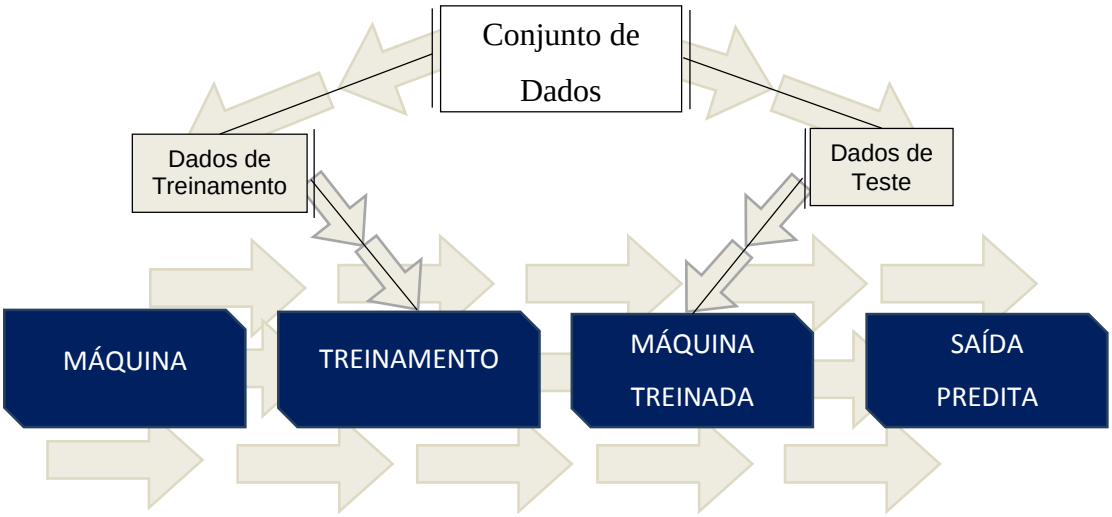
O conjunto utilizado para treinamento é o conjunto que contém tanto os dados de entradas como os dados de saída. Esses dados são passados para a máquina, a fim de que ela consiga encontrar um modelo matemático (uma função $h(x)$) que poderia ser utilizado para a predição de novas entradas submetidas para a máquina.

O conjunto de teste também possui os dados de entrada e saída, mas a máquina utiliza apenas os valores das entradas, e a partir desses valores ela prediz a saída de cada um deles. Após, a máquina verifica as saídas geradas por ela e as saídas corretas (que estão no conjunto

de teste), com o objetivo de verificar se a máquina foi capaz de produzir saídas corretas para as entradas recebidas.

A forma de divisão do conjunto dados é definida pelo analista, sabendo que quanto maior for o conjunto de treinamento melhor será o aprendizado (RUSSELL e NORVIG, 2013).

Figura 4 - Processo da Aprendizagem Supervisionada



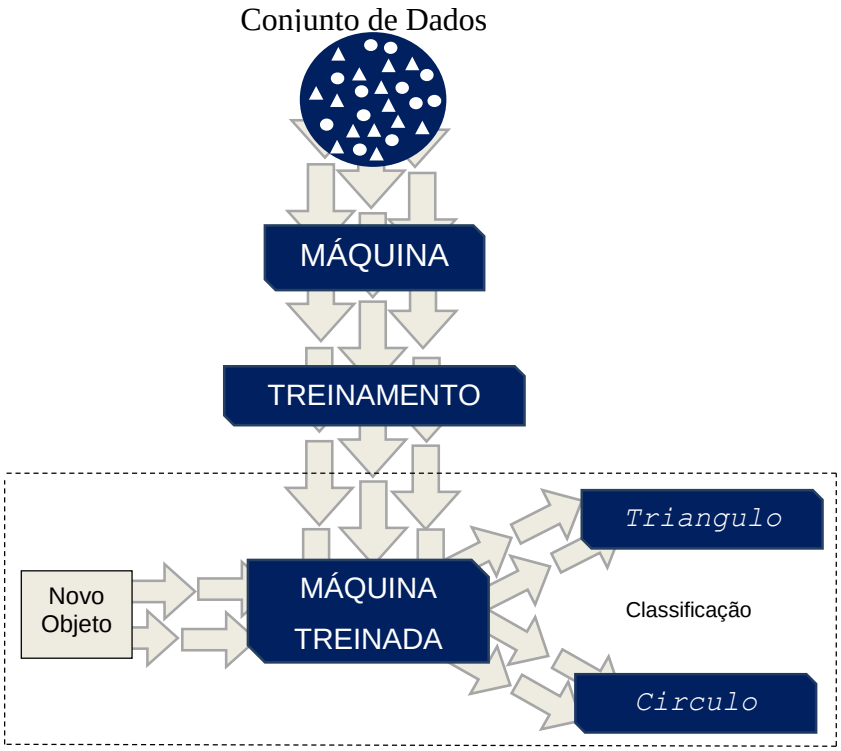
A Figura 4 mostra o processo da aprendizagem supervisionada. O conjunto de dados é dividido em duas partes, sendo elas chamadas de dados de treinamento (conjunto que contém os dados com pares de entradas e saídas) e de dados de teste (conjunto que contém apenas os dados de entradas). A máquina recebe os dados de treinamentos e, procura identificar os padrões e propriedades em comum entre os dados, gerando assim, modelos matemáticos (funções) que permita a ela criar uma regra de classificação que pode ser utilizada para a descoberta de novas soluções (saídas) para entradas desconhecidas (GSI, 1998). Após o treinamento, a máquina recebe o conjunto de teste e gera o valor de saída para cada entrada. Verifica-se a semelhança entre as saídas geradas pela máquina e as saídas desejadas, a fim de verificar se a máquina conseguiu prever os valores corretos.

A aprendizagem supervisionada pode ser dividida em dois tipos: classificação e regressão. Quando a saída (y) for de um conjunto finito de valores (como o clima de um dia, que poderia ser *ensolarado*, *nublado*, *chuvoso;* ou se o preço do dólar, em um determinado dia, irá *aumentar* ou *diminuir*), o problema da aprendizagem será chamado de classificação. Quando o y for um conjunto infinito de valores (como temperatura média de amanhã, ou o valor do dólar), o problema de aprendizagem é chamado de regressão (RUSSELL e NORVIG, 2013).

Nota-se que no exemplo do dólar apresentado na classificação, a resposta só poderia ser se o dólar irá *aumentar* ou *diminuir*, o valor dele não é o foco, o foco é o evento que poderia

acontecer, no caso *aumentar* ou *diminuir*. Já na regressão o que se procura predizer é o valor do dólar em um determinado dia, se vai aumentar ou não, não importa, simplesmente tenta-se predizer qual seria o possível valor do dólar em um determinado dia. A Figura 5 apresenta um exemplo do processo de classificação.

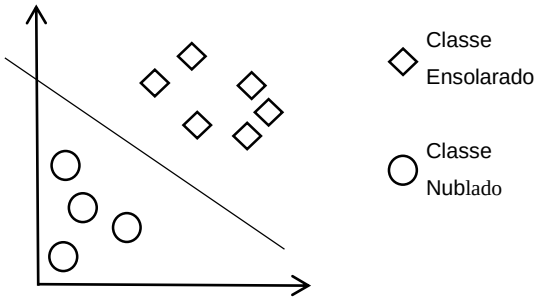
Figura 5 - Processo de classificação na Aprendizagem Supervisionada



No problema de classificação, o objetivo é fazer com que a máquina consiga distinguir e classificar, objetos de classes distintas, a partir de exemplos disponíveis, classificando um novo objeto de entrada a classe que ela pertença (GUNN, 1998). Um exemplo é apresentado na Figura 5, em que, existe um conjunto de dados que contém objetos que podem pertencer a dois tipos de classes, a classe *Triangulo* e a classe *Circulo*. A partir deste conjunto de dados a máquina é treinada, e consegue classificar novos objetos de entrada à classe que eles pertencem. Como no caso da Figura 5, que classifica o novo objeto como sendo da classe *Triangulo* ou da classe *Circulo*.

Para realizar a distinção das classes, a máquina procura descobrir os padrões e características das classes, e encontrar um classificador (função) que consiga separá-las (GUNN, 1998). A Figura 6 apresenta um exemplo de um problema de classificação de objetos, em que estes objetos devem ser classificados como sendo da classe *Ensolarado* ou da classe *Nublado*.

Figura 6 - Figura ilustrativa de um problema de classificação



Fonte: adaptado de FACELI et al. (2011, p. 55)

A Figura 6 apresenta um exemplo de classificação, em que se pretende classificar um determinado dia como sendo *Nublado* ou *Ensolarado* a partir de suas características (como temperatura, umidade relativa do ar, velocidade do ar, entre outros). Essas características apresentadas são analisadas pela máquina, e a partir do classificador encontrado, é realizada a classificação deste novo dia (objeto), a classe que ela pertença.

Como foi apresentada anteriormente, a aprendizagem supervisionada trabalha com solução de problemas de classificação (em que os conceitos e exemplos foram apresentados) e problemas de regressão. Este último será abordado com mais detalhes na Seção 2.2.1, que aborda o uso da aprendizagem supervisionada em problemas de regressão.

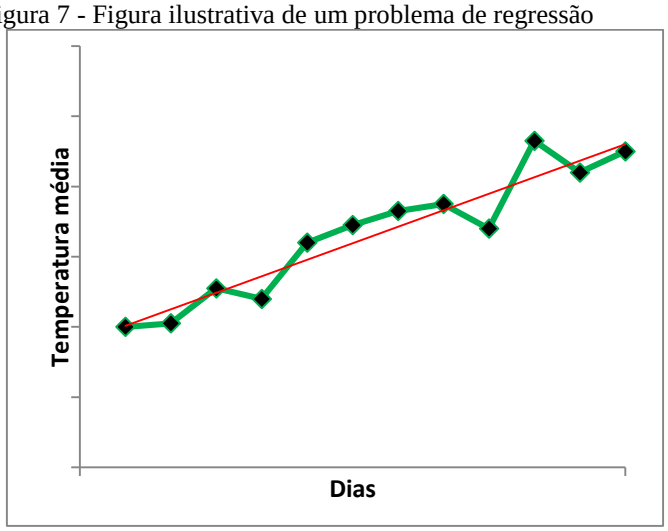
2.2.1. Aprendizagem Supervisionada em problema de Regressão

A regressão é um método matemático utilizado para realizar a predição de uma variável dependente (y) em função de uma ou mais variáveis independentes (x) (LARSON e FARBER, 2010). Utilizam-se as relações de causa-efeito que foram observadas no passado para poder prever as ações que irão ocorrer no futuro (TORRES e D'OTTAVIANO, 2015). Por exemplo, prever a temperatura média de um determinado dia. A partir dos dados colhidos anteriormente (como temperaturas de dias anteriores), é realizada a regressão para prever valores de temperaturas futuras.

Na regressão o objetivo é encontrar uma função $\hat{h}(x)$ que mais se aproxime da função real $f(x)$, para que esta função $\hat{h}(x)$ consiga prever o valor da variável dependente (y) através das variáveis independentes (x) (variável independente é a variável que o seu valor não depende de nenhuma outra variável para existir, e variável dependente é a variável que o seu valor depende de outras variáveis). A ligação dos pontos de x real (é o valor de x coletado do conjunto de dados) e y predito (que é o valor de y gerado pela função $\hat{h}(x)$, e será representado por $\hat{y}$) gera uma linha e, esta linha deve ser semelhante (ou o mais próximo possível) da linha que é gerada pela ligação dos pontos (x e y reais) (FACELI et al., 2011).

Um exemplo destas linhas é apresentado na Figura 7, em que a ligação entre os pontos (x, y) (que são os dados observados – "dados de treinamento") é representada pela linha na cor verde. A ligação entre os pontos $(x, \hat{y})$ (em que $\hat{y}$ é o y predito) é representada pela linha na cor vermelha. Os pontos (x, y) (dados observados) são representados pelos losangos (FACELI et al., 2011).

Figura 7 - Figura ilustrativa de um problema de regressão



O tipo de regressão apresentada pela Figura 7 é a regressão linear. Neste tipo de regressão a linha gerada pela função $\hat{h}(x)$ não, necessariamente, será idêntica à linha da função real $f(x)$, é aplicada em dados que podem ser linearmente representados (LARSON e FARBER, 2010).

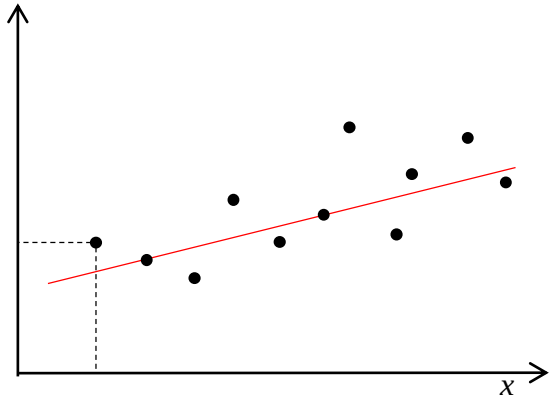
Entretanto, a linha gerada por $\hat{h}(x)$ deve ser uma linha reta, e seguir o sentido dos dados reais. A linha na cor verde representa os dados de treinamento (dados reais), já a linha na cor vermelha representa os dados preditos (os dados gerados por $\hat{h}(x)$). Mas, se observá-las, nota-se que não são idênticas e logo há uma queda no segundo ponto, que deixa a curva da linha verde em relação à linha na cor vermelha (reta) um pouco distante.

Desta maneira é possível notar que a linha vermelha não prevê valores exatamente iguais aos dados reais, mas os valores gerados por ela são próximos dos dados corretos para o determinado x. Isso ocorre devido que, a variável dependente (y) que se deseja predizer o valor não é 100% dependente da variável independente (x). Em outras palavras, valor da variável dependente não dependente apenas do valor da variável independente, pode depender de outros fatores, fenômenos ou até mesmo de outras variáveis.

Os valores resultantes da função $\hat{h}(x)$ (para cada valor de x) com o valor correspondente de x (quando agrupados em pares ordenados em um plano cartesiano) são representados por

uma linha, e esta linha é chamada de linha de regressão (LARSON e FARBER, 2010). Na Figura 7 e na Figura 8, a linha de regressão é representada pela linha na cor vermelha.

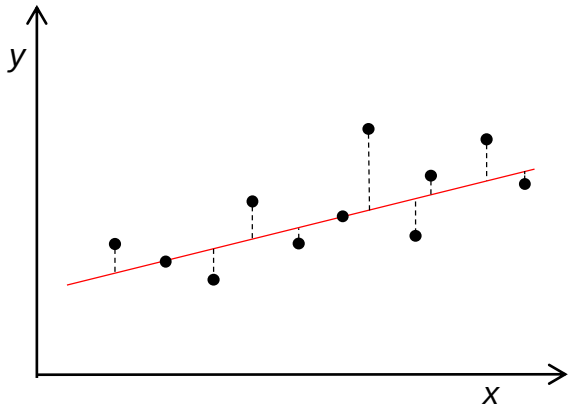
Figura 8 - Gráfico com a linha de regressão (x e $\hat{y}$).



Nota-se que a linha construída pelos valores de x e $\hat{y}$ ($\hat{h}(x)$) não se ajusta perfeitamente aos dados reais (pontos pretos na Figura 8), sendo assim, percebe-se que é possível obter diversas linhas com a mesma característica desta linha na cor vermelha apresentada na Figura 8, que é não se ajustar perfeitamente entre os dados. Desta maneira, é necessário encontrar a linha regressão que melhor se ajuste aos dados.

Freund (2006) define que a linha de regressão que possui o melhor ajuste aos dados é a linha 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 $\hat{y}$). As distâncias são definidas através da diferença do ponto real até a reta gerada pela função $\hat{h}(x)$, em outras palavras, a distância é a diferença entre o valor de y e o valor de $\hat{y}$ (y predito), para um determinado x. Essas distâncias são mostradas na Figura 9, e estão representadas pelas linhas tracejadas.

Figura 9 - Distância entre os pontos (x e y) da linha de regressão (x e $\hat{y}$).



O método utilizado para encontrar a função $\hat{h}(x)$ que melhor se ajuste aos dados, é denominado como Método dos Mínimos Quadrados (MMQ). Este método encontra o valor

mínimo da soma dos quadrados das distâncias entre os pontos, e a linha de regressão (FREUND, 2006), fazendo com que a função encontrada seja a que possuir a menor distância entre os pontos, obtendo assim o menor erro possível da predição de y.

Assim, tem-se que, para realizar o cálculo da distância e eleva-lo ao quadrado, é utilizada a Equação 2 (HOFFMANN, 1998). Essa equação deve ser aplicada para cada valor de y, a fim de encontra a diferença entre o valor de y e a reta, e elevá-la ao quadrado.

$$e_i^2 = [y_i - a - bx_i]^2 \tag{2}$$

onde:

y_i: é o valor de y no i-ésimo nível da variável (é a variável dependente);

a: é o valor de y quando x for igual a zero, também chamado de corte no eixo y;

b: é a variação de y em relação ao aumento de uma unidade em x;

x_i: é o valor de x no i-ésimo nível da variável (é a variável independente);

e_i^2: é o erro da distancia entre o valor y_i (valor de y real) e o $\hat{y}_i$ (valor de y predito), para o x_i.

Aplicando o somatório nos quadrados das distâncias da Equação 2, obtém-se, como resultado, a Equação 3 (LARSON e FARBER, 2010).

$$\sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} [y_i - a - bx_i]^2 \tag{3}$$

Assim, o método dos mínimos quadrados necessita que os valores dos estimadores de a e b possibilitem que o valor da função de reta ($\hat{h}(x)$) seja mínima (HOEL, 1981), obtendo assim a minimização da soma dos quadrados das distâncias.

Os valores dos estimadores a e b podem ser encontrados a partir das Equações 4 e 5 (GOMES, 2000).

$$b = \frac{n \sum xy - (\sum x)(\sum y)}{n \sum x^2 - (\sum x)^2} \tag{4}$$

onde:

n: é o número de amostras de x e y.

$$a = \bar{y} - b\bar{x} = \frac{\sum y}{n} - b\frac{\sum x}{n} \quad (5)$$

onde:

n: é o número de amostras de x e y;

$\bar{y}$: é a média dos valores de y no conjunto de dados;

$\bar{x}$: é a média dos valores de x no conjunto de dados.

b: é a variação de y em relação ao aumento de uma unidade em x.

A partir dessas equações é obtido o valor mínimo de a e b para ser aplicado na equação da regressão (Equação 6). Deste modo, é gerado a minimização da soma dos quadrados das distâncias (FREUND, 2006). Assim, o problema de gerar uma reta que melhor se ajuste aos dados se resume em calcular o valor dos estimadores a e b da equação da regressão (Equação 6).

Obtidos o valor dos estimadores a e b, é possível encontrar a reta que melhor se ajuste aos dados, aplicando esses estimadores na equação de regressão (Equação 6) (LARSON e FARBER, 2010).

$$\hat{h}(x) = \hat{y} = a + bx \quad (6)$$

onde:

$\hat{y}$: é o valor que se deseja predizer, é o y predito;

a: é o valor de y quando x for igual a zero, também chamado de corte no eixo y;

b: é a variação de y em relação ao aumento de uma unidade em x;

x: variável independente.

A partir da Equação 6 é possível realizar a predição do valor de $\hat{y}$ para um determinado x. Sendo assim a função $\hat{h}(x)$ que se procurava encontrar se resume em: encontrar o valor dos estimadores a e b, devido que a partir destes, a reta encontrada será a reta que se ajusta melhor aos dados (HOEL, 1981).

Os conceitos apresentados até o momento são referentes à regressão entre duas variáveis lineares, sendo assim chamada de regressão linear simples (GOMES, 2000).

Existem diferentes formas para se realizar a regressão, dentre elas existem: a regressão linear multivariada: utilizada para cenários em que os dados são lineares, mas que possuam

mais de uma variável independente (LARSON e FARBER, 2010); e a regressão não linear: para cenários em que uma linha reta não consegue se ajustar aos dados, devido que esses são não lineares (FREUND, 2006).

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.

Dentre estes, este trabalho tem como propósito utilizar a regressão linear simples através do algoritmo de Máquina de Vetores de Suporte (SVM, do inglês *Support Vector Machine*), assunto abordado com mais detalhes na Seção 2.3.

Buscando entendimento melhor sobre o conceito de regressão que foi apresentado, a seção a seguir (Seção 2.2.1.1) apresentará o exemplo de um problema de predição de valores, utilizando o método de regressão linear.

2.2.1.1. Exemplo de cálculo de regressão

Para que haja um entendimento sobre a aplicação do método de regressão, é apresentado um cenário hipotético para exemplificar o passo-a-passo dos cálculos para predição.

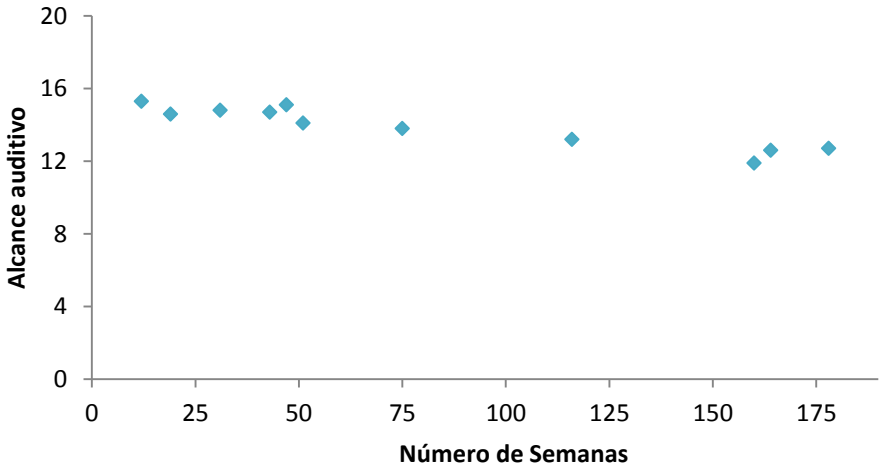
Freund (2006) utiliza o seguinte exemplo de aplicação do método de regressão para que se possa obter um melhor entendimento sobre a regressão. Os dados da Tabela 1 mostram a relação entre o tempo o qual uma pessoa esteve exposta a um alto nível de ruído (como o barulho das turbinas de aviões) (esta é a variável independente x), e o alcance auditivo (esta é a variável dependente y). Alcance auditivo é a amplitude da frequência sonora pelo qual os ouvidos respondem. Desta maneira procura-se predizer qual será o possível alcance auditivo de uma pessoa que tenha ficado 74 semanas exposta a um alto nível de ruído.

Tabela 1 - Dados do problema proposto (adaptado de FREUND, 2006).

Número de semanas (x)	*Alcance auditivo (y)*
47	15,1
51	14,1
116	13,2
178	12,7
19	14,6
75	13,8
160	11,9
31	14,8
12	15,3
164	12,6
43	14,7
74	?

Passando os dados fornecidos para um plano cartesiano é o obtido o gráfico da Figura 10, em que os losangos representam os pares de x e y.

Figura 10 - Gráfico dos dados auditivos



Inicialmente 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 ao aumento de uma unidade em x.

Foram calculados os somatórios que serão necessários para a resolução das equações, e foi criada a Tabela 2 para proporcionar um melhor entendimento dos valores desses somatórios e facilitar o uso e consulta desses valores.

Tabela 2 - Somatórios utilizados para o cálculo da regressão.

x	y	xy	x^2
47	15,1	709,7	2209
51	14,1	719,1	2601
116	13,2	1531,2	13456
178	12,7	2260,6	31684
19	14,6	277,4	361
75	13,8	1035	5625
160	11,9	1904	25600
31	14,8	458,8	961
12	15,3	147,6	144
164	12,6	2066,4	26896
43	14,7	632,1	1849
$\sum x = 896$	$\sum y = 152{,}8$	$\sum xy = 11741{,}9$	$\sum x^2 = 111386$

O valor de n será 11, que é o número de dados coletados.

Uma vez que os valores dos somatórios foram calculados, é possível calcular os valores de a e b, utilizando suas respectivas fórmulas (Equação 5 e Equação 4). Deve-se iniciar pelo calculado de b.

Substituindo os valores calculados na equação de b (Equação 4), tem-se:

$$b = \frac{n\sum xy - (\sum x)(\sum y)}{n\sum x^2 - (\sum x)^2} = \frac{11 * 11741{,}9 - 896 * 152{,}8}{11 * 111386 - (896)^2} = \frac{129160{,}9 - 136908{,}8}{1225246 - 802816}$$

$$= \frac{-7747{,}9}{422430} = -0{,}01834 \approx -0{,}0183$$

Após ser encontrado o valor de b é possível calcular o valor a, utilizando a Equação 5, substituindo os devidos valores.

$$a = \bar{y} - b\ddot{x} = \frac{\sum y}{n} - b\frac{\sum x}{n} = \frac{152{,}8}{11} - (-0{,}0183)\frac{896}{11} = 13{,}8909 - (-0{,}0183) * 81{,}4545$$

$$= 13{,}8909 + 1{,}4906 = 15{,}3815$$

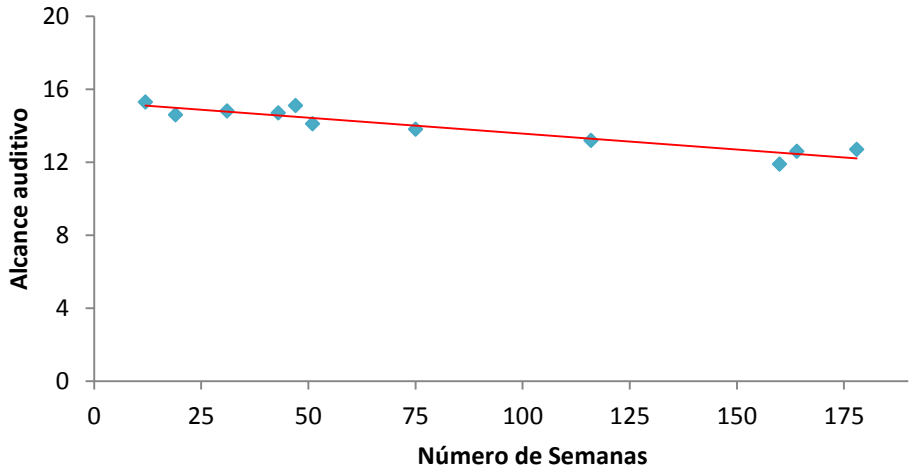
Com os valores de a e b já calculados é obtida a linha de regressão, calculando o valor de $\hat{y}$ para cada x existente. Como é mostrado na Tabela 3.

Tabela 3 - Pontos na linha de regressão.

x	$\hat{y} = a + bx$	$\hat{y}$	$Pontos(x; \hat{y})$
7	$\hat{y} = 15{,}3815 + (-0{,}0183)47$	14,52	$(47;\ 14{,}52)$
1	$\hat{y} = 15{,}3815 + (-0{,}0183)51$	14,44	$(51;\ 14{,}44)$
116	$\hat{y} = 15{,}3815 + (-0{,}0183)116$	13,25	$(116;\ 13{,}25)$
178	$\hat{y} = 15{,}3815 + (-0{,}0183)178$	12,12	$(178;\ 12{,}12)$
19	$\hat{y} = 15{,}3815 + (-0{,}0183)19$	15,03	$(19;\ 15{,}03)$
75	$\hat{y} = 15{,}3815 + (-0{,}0183)75$	14,01	$(75;\ 14{,}01)$
160	$\hat{y} = 15{,}3815 + (-0{,}0183)160$	12,45	$(160; 12{,}45)$
31	$\hat{y} = 15{,}3815 + (-0{,}0183)31$	14,81	$(31;\ 14{,}81)$
12	$\hat{y} = 15{,}3815 + (-0{,}0183)12$	15,16	$(12;\ 15{,}16)$
164	$\hat{y} = 15{,}3815 + (-0{,}0183)164$	12,38	$(164;\ 12{,}38)$
43	$\hat{y} = 15{,}3815 + (-0{,}0183)43$	14,59	$(43;\ 14{,}59)$

Inserindo estes pontos $(x, \hat{y})$ encontrados no gráfico, e ligando-os por uma linha, é possível visualizar a linha de regressão. Como é mostrado na Figura 11, em que a linha na cor vermelha representa a linha de regressão.

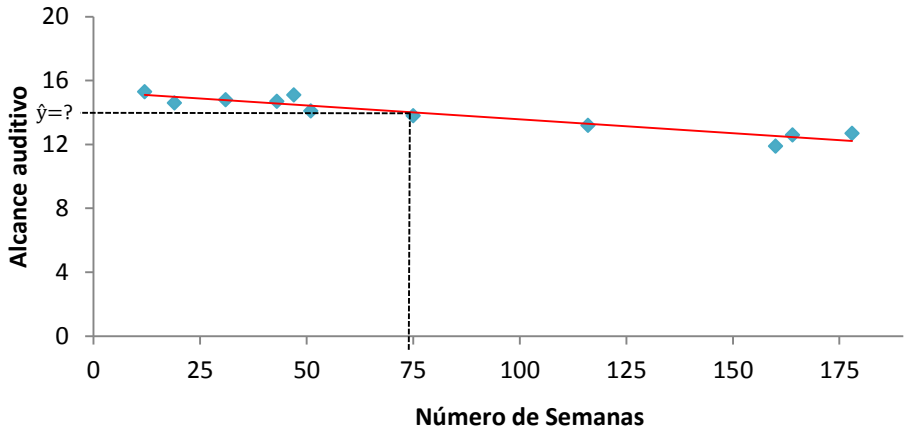
Figura 11 - Gráfico dos dados auditivos e a linha de Regressão



Nota-se que esta linha foi gerada a partir de valores de a e b. Estes valores permitem encontrar uma linha que minimizem as distâncias dos pontos até a reta. Deste modo, tem-se que esta linha é a que possui o melhor ajuste entre os dados, e é chamada de linha de regressão.

Dado que os valores de a e b foram calculados, é possível resolver o problema apresentado, em que se deseja predizer qual será o possível alcance auditivo de uma pessoa que tenha ficado 74 semanas exposta a um alto nível de ruído.

Figura 12 - Gráfico dos dados auditivos, com valor de y que se deseja predizer (ŷ)



Para isso é utilizado a equação de regressão (Equação 6).

$$\hat{y} = a + bx$$

Em que, o valor de x (variável independente) é 74, que é o número de semana em que se deseja predizer o possível alcance auditivo de uma pessoa que esteve exposta a ruído por este período.

Até o momento, têm-se os seguintes valores mostrados na Tabela 4.

Tabela 4 - Valores calculados até o momento.

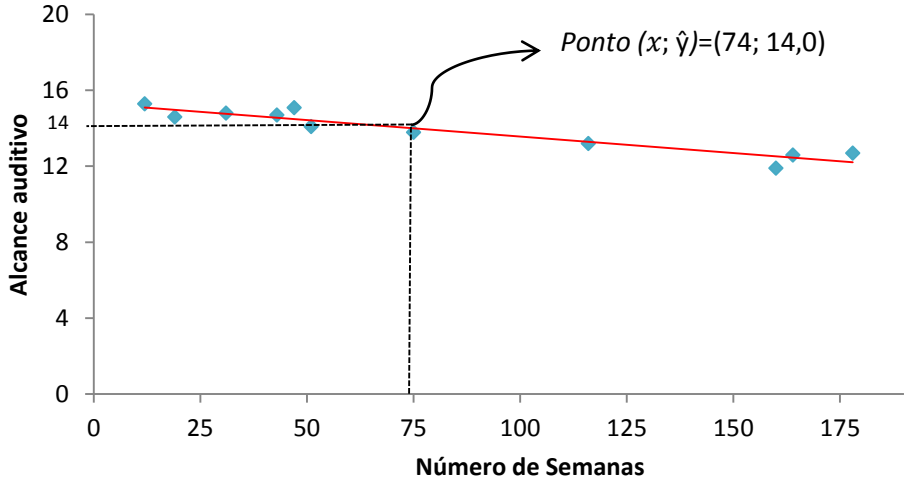
$\boldsymbol{a}$	$\boldsymbol{b}$	$\boldsymbol{x}$
15,3815	−0,0183	75

Substituindo as variáveis por seus respectivos valores, tem-se:

$$\hat{y} = a + bx = 15{,}3815 + (-0{,}0183)74 = 15{,}3815 - 1{,}3542 = 14{,}0273 \approx 14{,}0$$

Desta maneira, conclui-se que, se uma pessoa ficar 74 semanas expostas a um alto nível de ruído terá um alcance auditivo de, aproximadamente, 14,0.

Figura 13 - Gráfico com o ponto predito



A Figura 13 mostra o ponto que possui as coordenadas de x (74) e o $\hat{y}$ (14,0) (que é o y predito).

2.3. Máquina de Vetores de Suporte (Support Vector Machine – SVM)

A Máquina de Vetores de Suporte (SVM, do inglês *Support Vector Machine*) foi desenvolvida pelo russo Vladmir Vapnik a partir de estudos iniciados em Vapnik e Chevonenkis (1971 apud ALBUQUERQUE, 2012, p. 30). A SVM aborda os conceitos de aprendizagem supervisionada, possui conjunto de treinamento e teste. Inicialmente foi criada para resolver problemas de classificação, mas há algum tempo vem sendo aplicada para resolver problemas de regressão também (CHAMASEMANI e SINGH, 2011).

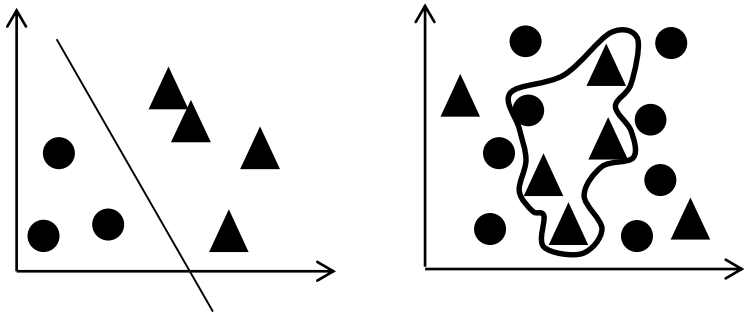
As SVMs podem ser aplicadas para realização de tarefas de categorização de textos, por exemplo, atribuir de forma automática uma ou mais categorias predefinidas a documentos textuais (SUN e LIM, 2001 apud MORAES e LIMA, 2008); reconhecimento de caracteres, como sistemas de reconhecimento de assinaturas (CARVALHO et al., 2009); bioinformática, auxiliando na compreensão de estruturas genéticas (REZENDE e SILVA, 2008); de categorização de spam, classificando se um e-mail é spam ou não; entre outros (VAPNIK, BOSER e GUYON, 1992).

Na Seção 2.3.1 são apresentados os conceitos e características da SVM para problema de classificação, que é referida como Classificação de Vetores de Suporte (SVC, do inglês *Support Vector Classification*), e Regressão de Vetores de Suporte (SVR, do inglês *Support Vector Regression*) para problemas de regressão (GUNN, 1998).

2.3.1. Classificação de Vetores de Suporte (SVC)

A Classificação de Vetores de Suporte (SVC, do inglês *Support Vector Classification*), parte dos conceitos de classificação apresentados na Seção 2.2 sobre aprendizagem supervisionada. Objetivo da SVC é fazer com que a máquina seja capaz de classificar objetos de classes distintas, a partir de exemplos anteriores, podendo, a partir disso, classificar uma nova entrada (objeto) a classe que ela pertença (GUNN, 1998).

Figura 14 - Exemplos de hiperplanos. a) hiperplano de separação para dados lineares; b) hiperplanos de separação para dados não lineares

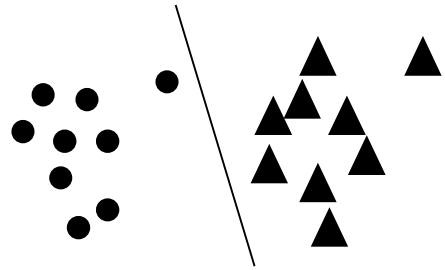


Existem duas formas de aplicação do SVC, são elas: Classificação de Vetores de Suporte (SVC) Linear, utilizada para dados lineares (que podem ser separados por uma reta) (Figura 14 (a)), e Classificação de Vetores de Suporte (SVC) Não-Linear, utilizada para dados não lineares (que não podem ser separados por uma reta) (Figura 14 (b)). A Seção 2.3.1.1 apresenta os conceitos relacionados a utilização da SVC em dados lineares, que são os dados que podem ser separados por uma reta. Já na Seção 2.3.1.2 é apresentada os conceitos da utilização da SVC em dados não lineares, que são os dados que não podem ser separados por uma linha reta.

2.3.1.1. SVC Linear

A SVC objetiva encontrar um classificador (com base nas características e padrões de cada classe) que consiga separar as classes de forma correta (GUNN, 1998). Isto é, não permite que nenhum objeto das classes seja classificado de forma incorreta. Este classificador é chamado de hiperplano. A Figura 15 apresenta um exemplo de um hiperplano. Este hiperplano separa os objetos da classe "*circulo"* dos objetos da classe *"triângulo"* de forma linear.

Figura 15 - Exemplo de hiperplano de separação



Fonte: adaptado de GUNN (1998)

Este classificador (hiperplano) pode ser encontrado através da Equação 7 (CHAMASEMANI E SINGH, 2011):

$$f(x) = w \,.\, x + b = 0 \tag{7}$$

onde:

w: é o vetor que representa o conjunto de entradas, conhecido como vetor de peso (weight vector);

x: são os pontos sobre o hiperplano;

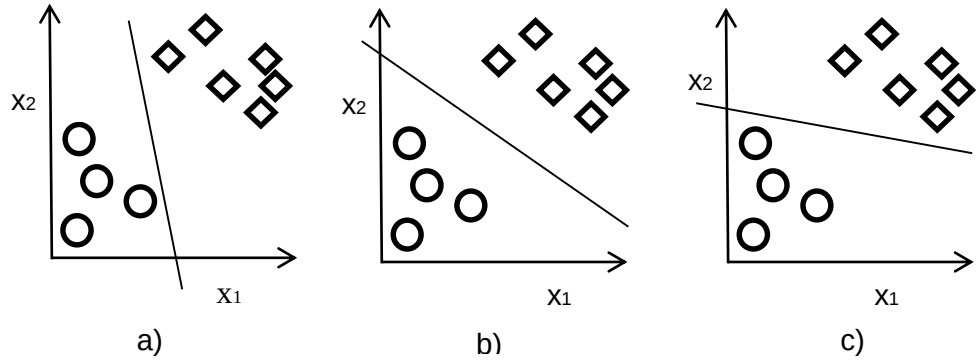
b: é a distancia entre o hiperplano e o ponto de origem.

A partir disso, o hiperplano irá separar os dados em duas regiões, sendo elas: $f(x) > 0$ ou $f(x) < 0$ (LORENA e CARVALHO, 2007). Para obter a classificação é utilizada a função de sinal, conforme é mostrado na Equação 8.

$$h\,(x) = sgn\,\big(\,f(x)\big) = \begin{cases} -1 \; se \; w \,.\, x + b < 0 \\ +1 \; se \; w.x + b \; > 0 \end{cases} \tag{8}$$

Sendo que o -1 corresponde a objetos de uma classe (por exemplo, no caso da Figura 15, corresponde a classe "*circulo")*, e o +1 corresponde a objetos de outra classe (por exemplo, no caso da Figura 15, classe "triângulo*"*). A Figura 16 apresenta três possíveis hiperplanos que separam as classes de forma correta.

Figura 16 - Três hiperplanos possíveis de separação das classes



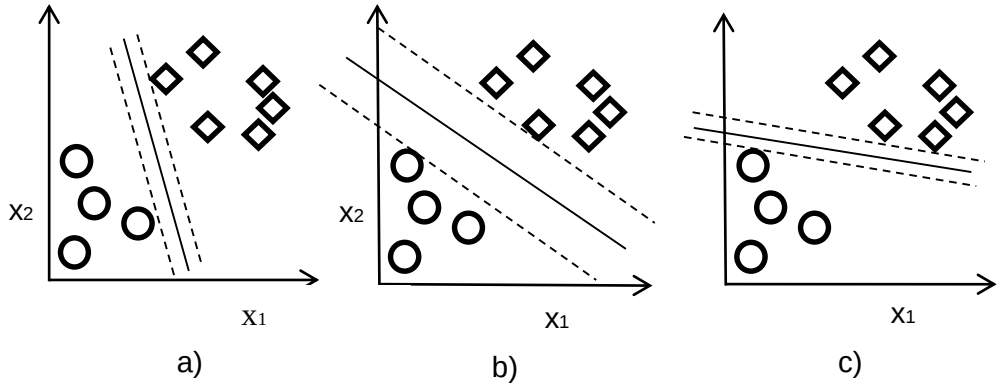
Fonte: adaptado de COSTA e SIMÕES (2008)

Vários hiperplanos (que classifiquem de forma correta os dados) podem ser encontrados a partir da Equação 7. Logo, deve-se escolher um hiperplano que melhor classifique os dados (FACELI et al., 2011). A Figura 16 mostra um exemplo de três hiperplanos (representados por uma reta) que conseguem realizar a separação entre as classes de forma correta.

Diante das diversas possibilidades de hiperplanos, tem-se que o melhor hiperplano de separação é o que possuir a maior margem entre as classes, e é chamado de hiperplano ótimo. A margem é definida como duas vezes a distância do hiperplano (reta) até o objeto mais próximo (RUSSELL e NORVIG, 2013). Quanto maior for a margem, menor é o erro associado à classificação de novos objetos (novas entradas) (COSTA e SIMÕES, 2008).

A Figura 17 mostra a margem de cada hiperplano apresentado na Figura 16. A margem é delimitada pelas linhas tracejadas da Figura 17. Na Figura 17 (a) e (b), são dois os objetos mais próximos do hiperplano: um *círculo* e um *losango*; e na Figura 17 (c) o objeto mais próximo é da classe *círculo*.

Figura 17 - Exemplo de margem de cada hiperplano

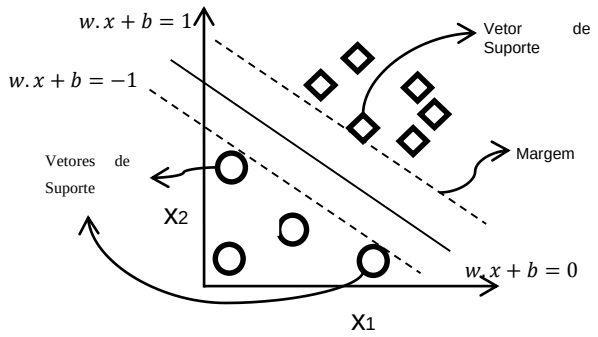


Fonte: adaptado de COSTA e SIMÕES (2008)

Para encontrar quais são os hiperplanos que possuem as maiores margens é preciso definir os Vetores de Suporte (SV, do inglês *Support Vector*). Vetores de Suporte (SV) são os

objetos, de ambas as classes, que estão mais próximos do separador de classes (hiperplano) (ALBUQUERQUE, 2012). A partir desses conceitos, é possível verificar que entre os três hiperplanos mostrados na Figura 17, o que realiza a melhor separação das classes está sendo mostrado na Figura 17 (b), pois ele apresenta a maior distância (margem) possível, entre a função de separação (hiperplano) e os vetores de suporte (LORENA e CARVALHO, 2007).

Figura 18 - Margem e Vetores de Suporte



Fonte: adaptado de COSTA e SIMÕES (2008)

A Figura 18 mostra os vetores de suporte existentes e as margens. Os vetores de suporte (objetos mais próximos do hiperplano) são aqueles que satisfazem a Equação 9 (FACELI et al., 2011).

$$| w . x_i + b| = 1; \qquad i = 1,2, \dots, n. \tag{9}$$

onde:

i: posição do objeto;

n: quantidade de objetos existentes no conjunto;

x_i: corresponde a um determinado objeto;

Satisfazendo as seguintes restrições (BURGES, 1998):

$$\begin{cases} w.x_i + b \geq +1 \; se \; y_i = +1 \\ w.x_i + b \leq -1 \; se \; y_i = -1; \end{cases} \; i = 1,2, \dots, n. \tag{10}$$

onde:

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

A Inequação 10 é resumida na Inequação 11 (EL-NAQA et al., 2002):

$$y_i(w.x_i + b) - 1 \geq 0; \qquad i = 1,2,\dots,n. \tag{11}$$

Para encontrar o hiperplano ótimo é necessário obter a maximização da margem. Essa maximização da margem pode ser obtida pela minimização (redução) de $\left|\left|w\right|\right|$ (Equação 12) (GUNN, 1997). Respeitando a Inequação 13.

$$Minimizar_{w,b}\ \frac{1}{2}\left|\left|w\right|\right|^2 \tag{12}$$

$$\text{Respeitando as restrições } y_i(w.x_i + b) - 1\ \geq 0, \forall i = 1,\dots,n. \tag{13}$$

onde:

$||w||$: é a norma euclidiana de w, $\left|\left|w\right|\right| = \sqrt{w.w}$.

Esta restrição garante que não haverá nenhum objeto entre as margens. Devido a não permitir que haja dados entre as margens, esse tipo de SVC é conhecido como SVC (ou SVM) com margens rígidas.

O problema de minimização de $||w||$, respeitando restrições, pode ser resolvido utilizando o método de multiplicadores de Lagrange (Equação 14) (ASANO, 2005). Este método fornece o máximo e mínimo de uma função sujeita a uma restrição.

$$L(w,b,\lambda) = \frac{1}{2}\left|\left|w\right|\right|^2 - \sum_i \lambda_i\,(y_i(w.x_i + b) - 1) \tag{14}$$

onde:

λ_i: são os multiplicadores de Lagrange.

Buscando encontrar a solução ótima, é necessário minimizar a função de Lagrange, realizando a maximização dos multiplicadores de Lagrange (λ), e minimização de w e b (CHAMASEMANI E SINGH, 2011) resultando no problema de otimização:

$$Maximizar_{\lambda_i} \sum_{i=1}^{n} \lambda_i - \frac{1}{2}(w.x_i + b) \tag{15}$$

$$Com\ as\ restrições: \begin{cases} \lambda_i \geq 0, \forall i = 1, \dots, n \\ b = 0 \end{cases} \quad (16)$$

onde:

$$w = \sum_{i=1}^{n} \lambda_i y_i x_i \qquad e \qquad b = \sum_{i=1}^{n} \lambda_i \, y_i \quad (17)$$

O problema de otimização apresentado acima possui um único máximo global que pode ser encontrado (EL-NAQA et al., 2002). Sendo assim tem-se:

$$h(x) = sgn(\sum \lambda_i y_i (x.x_i) + b) \quad (18)$$

A função terá λ_i diferente de zero para os objetos que são chamados de vetores de suportes (SV) (que são os pontos mais próximos do hiperplano) e λ_i igual a zero para os demais objetos. Assim tem-se que a função de classificação é:

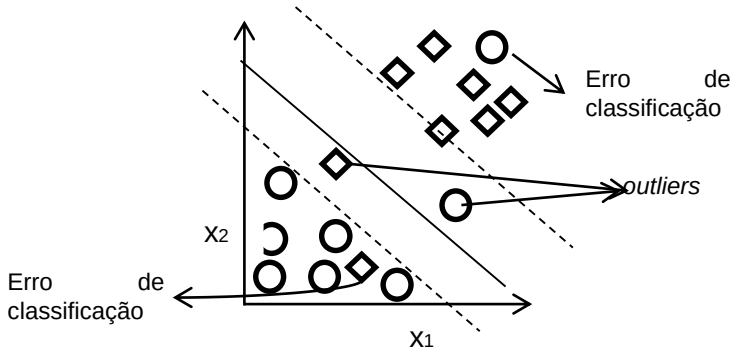
$$f(x) = \sum \lambda_i y_i x_i x + b \quad (19)$$

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

2.3.1.1.1. SVC com margens suaves

Existem problemas em que o classificador linear (hiperplano) não consegue separar os dados de forma perfeita (sem realizar nenhum erro de classificação). Normalmente devido à presença de objetos com características fora dos padrões de sua classe (chamados de *outliers*) (GUNN, 1998). A Figura 19 mostra um exemplo de um conjunto de dados com *outliers*, neste tipo de problema não é possível separar os dados por uma reta.

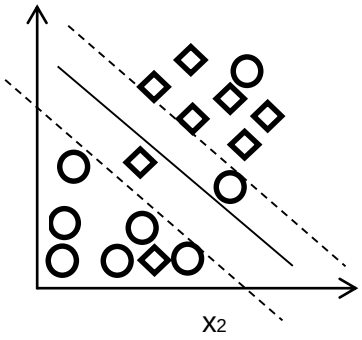
Figura 19 - Exemplo de dados com outliers.



Observa-se na Figura 19 que dois objetos estão classificados de forma errada (considerados de erro de classificação) e dois objetos estão entre as margens do hiperplano, o que não é permitido na SVC de margens rígidas. Para este tipo de problema é usado o mesmo conceito de margens rígidas, mas neste caso, permite-se que alguns dados possam ficar entre as margens do hiperplano, e que ocorram alguns erros de classificação (LORENA e CARVALHO, 2007). Dessa maneira as margens encontradas nesse modelo são chamadas de margens suaves.

A Figura 20 mostra um exemplo da aplicação de margens suaves, com um objeto da classe círculo e losango entre as margens.

Figura 20 - Margens de Suaves



As margens suaves são encontradas adicionando, nas restrições impostas na Equação 12, variáveis de folga (ξ), resultando na inequação:

$$y_i(w.x_i + b) \geq 1 - \xi_i, \qquad \xi_i \geq 0 \; \forall i = 1, \ldots, n \tag{20}$$

Estas variáveis de folga são variáveis positivas que indicam a tolerância a erros de classificação (ASANO, 2005).

A partir disso, as Equações 12 e 13 são reescritas da seguinte forma:

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

Com as restrições: $y_i(w.x_i + b) \geq 1 - \ \xi_i,\ \xi_i \geq 0\ \forall i = 1, \dots, n.$ (22)

Onde, C é um termo de regularização, que pode ser visto como uma forma de controlar o ajuste entre a importância de maximizar a margem e ajustar os dados (CHAMASEMANI E SINGH, 2011). $\sum_{i=1}^{n}\xi_i$: representa o limite no número de erros no conjunto de dados.

A solução para este problema de minimização aplicada nas margens rígidas pode ser aplicada nas margens suaves. Se busca encontrar a solução ótima, e para isso é necessário aplicar o método de Lagrange, realizar a maximização dos multiplicadores de Lagrange (λ), e a minimização de w e b (CHAMASEMANI E SINGH, 2011), resultando, desta forma, no problema de otimização:

$$Maximizar_{\lambda_i}\ \sum_{i=1}^{n}\lambda_i - \frac{1}{2}(w.x_i + b) \tag{23}$$

Com as restrições: $\begin{cases} C\ \geq \lambda_i \geq 0, \forall i = 1, \dots, n \\ b = 0 \end{cases}$ (24)

onde:

$$w = \sum_{i=1}^{n}\lambda_i y_i x_i \qquad \text{e} \qquad b = \sum_{i=1}^{n}\lambda_i\, y_i. \tag{25}$$

Da mesma forma que nas margens rígidas, os pontos x_i que possuem o $\lambda_i > 0$ são chamados de vetores de suportes (SV). E a função que define o classificador (Equação 26) é escrita da mesma maneira da Equação 19, se diferindo apenas pela forma que é determinada o valor de λ_i, que é encontrado pela Equação 23 com suas devidas restrições.

$$f(x) = \sum \lambda_i y_i x_i x + b \tag{26}$$

Ainda assim existem problemas, em que mesmo utilizando as SVC de margens suaves, não é possível encontrar um hiperplano que consiga classificar de forma correta todos os dados (como pode ser visto na Figura 21 (b)), esses problemas envolvem dados não lineares (que não podem ser separados por uma linha). Sendo assim se introduziu o conceito de SVC não linear,

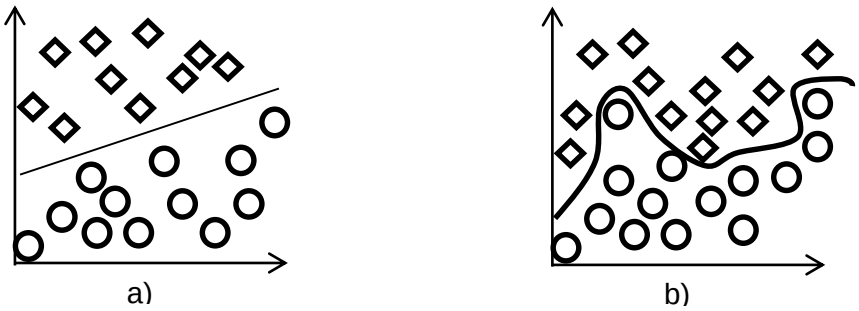
que trabalha com a classificação desses dados. Os conceitos de SVCs não lineares são apresentados na Seção 2.3.1.2.

2.3.1.2. SVC Não Linear

Existem problemas em que os dados não são linearmente separáveis, (não podem ser separados por uma reta), logo o classificador ótimo (que consegue classificar a maior parte dos dados de forma correta e possui a maior margem) não consegue obter uma classificação considerada ótima dos dados (CHAMASEAMANI e SINGH, 2011).

Desta maneira, os Classificadores de Vetores de Suporte (SVC, do inglês *Support Vector Classification)* não linear trabalham com a classificação destes dados (dados que não podem ser linearmente separados). A Figura 21 mostra um problema linearmente separável (Figura 21 (a)) e outro não linearmente separável (Figura 21 (b)).

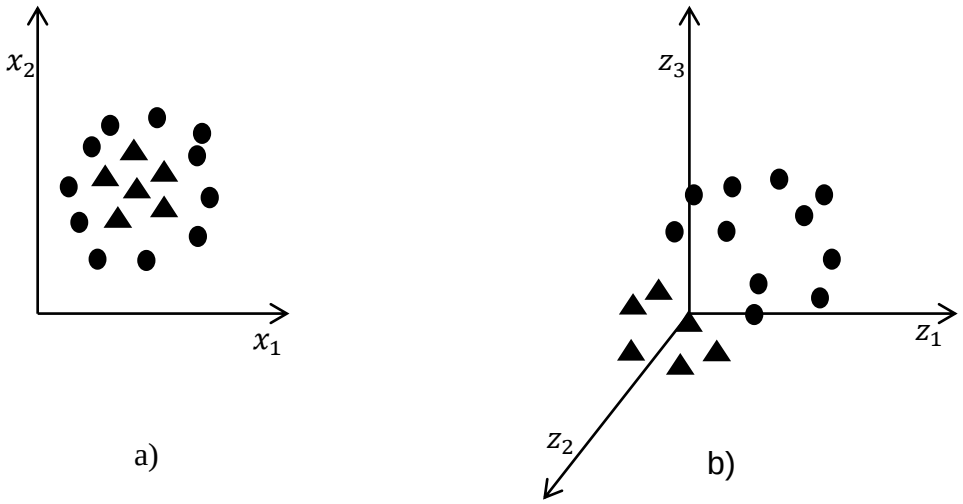
Figura 21 - Exemplo de dados linearmente separáveis (a) e não linearmente separáveis (b)



Segundo Haykin (1999, apud SANTOS, 2002), um problema que é dito como um problema complexo de classificação de objetos, possui uma maior probabilidade de ser linearmente separável se estiver em um espaço de alta dimensão. Esta afirmação tem como base o teorema de Cover, em que mostra que um conjunto de dados não lineares pode ser linearmente separável quando transformado em um espaço de alta dimensão (LORENA e CARVALHO, 2007).

Sendo assim, o SVC não linear mapeia o conjunto de treinamento (que está em seu espaço original) para um espaço de dimensões maiores, chamado de espaço de características (*features space*) (ALBUQUERQUE, 2012). Um exemplo é mostrado na Figura 22, em que a Figura 22 (a) representa o conjunto de treinamento (que possui dados não lineares) em seu espaço original, e a Figura 22 (b) mostra o conjunto de treinamento em um espaço de dimensões maiores.

Figura 22 - Exemplo de um espaço original (a) e espaço com dimensões maiores (b)



Fonte: adaptado FACELI et al. (2011)

O mapeamento dos dados para espaços de dimensões maiores (espaço de características) é obtido através da equação (HEARST, 1998):

$$\Phi(Obj) = \Phi(x, y) = (x^2, \sqrt{2}xy, y^2) \tag{27}$$

onde:

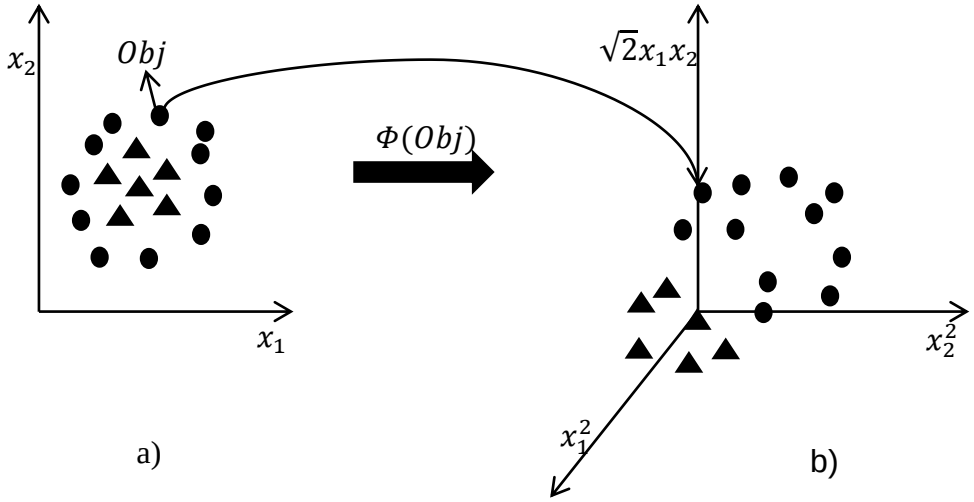
Obj: é o objeto que se deseja mapear para um plano de dimensão maior;

x, y: são as coordenadas do Obj no espaço original.

Aplicando a Equação 27 no exemplo apresentado na Figura 22, tem-se:

$$\Phi(Obj) = \Phi(x_1, x_2) = (x_1^2, \sqrt{2}x_1x_2, x_2^2) \tag{28}$$

Figura 23 - Exemplo de um espaço original (a) e espaço com dimensões maiores (b)



Fonte: adaptado FACELI et al. (2011)

Para que se consiga encontrar um classificador para este espaço de características são utilizadas funções conhecidas como funções Kernel (CHAMASEMANI e SINGH, 2011). Estas funções têm como objetivo projetar dados do espaço de entrada para um espaço de características de alta dimensão, fazendo com que seja possível classificar estes dados de forma linear. Uma função Kernel é definida como:

$$K(x_i, x_j) = \Phi(x_i).\Phi(x_j) = (x_i \,.\, x_j)^2 \tag{29}$$

onde:

K: é a função Kernel;

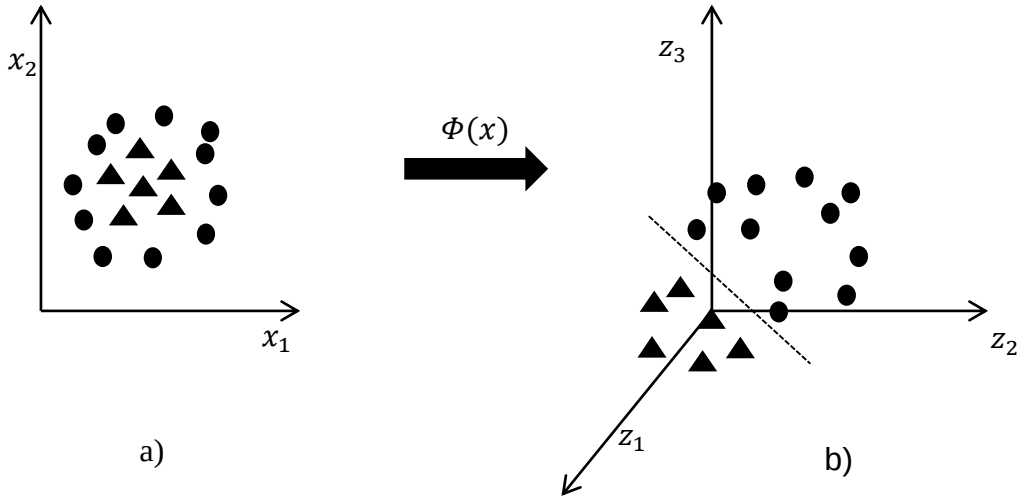
x_i, x_j: são pontos (objetos) do espaço original;

Segundo Lorena e Carvalho (2007) o uso da função Kernel possui a vantagem de que não é necessário conhecer o mapeamento Φ, que é gerado implicitamente, desta maneira a vantagem de se utilizar estas funções está na simplicidade do seu cálculo e na sua capacidade de representar espaços abstratos.

Após os dados terem sido mapeados para um espaço de características, é aplicado o SVC linear sobre este espaço. A partir disso, é encontrado o hiperplano de margem máxima, para se garantir uma boa generalização dos dados, e utilizado as margens suaves para lidar com os *outliers* e erros de classificação pertencentes aos dados (FACELI et al., 2011). Aplicando a função Kernel na Equação 26 (equação do classificador ótimo com margens suaves, devido que este permite que existam erros de classificação) tem-se então (BURGES, 1998):

$$f(x) = \sum \lambda_i y_i K(x_i, x_j) + b \tag{30}$$

Figura 24 - Dados no espaço original (a) e no espaço com dimensões maiores (b)



Fonte: adaptado FACELI et al. (2011)

A Figura 23 ilustra um exemplo de dados não separáveis linearmente (em seu espaço original) (Figura 24 (a)), sendo mapeado para um espaço de características (espaço de dimensões maiores) (Figura 24 (b)), e o seu hiperplano de separação (representado pelos pontos tracejados da Figura 24 (b)), separando corretamente os dados de cada classe.

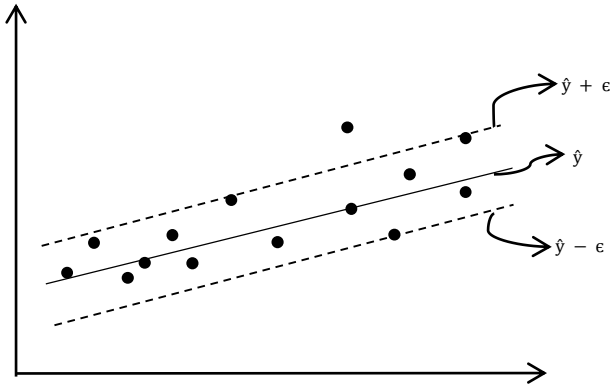
Percebe-se que as funções Kernel são de grande importância para que seja possível encontrar um classificador que consiga separar de forma correta os dados. Contudo, existem várias funções Kernel que podem ser utilizadas para realizar a separação de dados, e entre elas, as mais utilizadas são: Lineares (sendo este o modelo apresentado acima), Polinomiais, Sigmoidais e os RBF (*Radial-Basic Function*) (ERASTO, 2001; CHAMASEMANI e SINGH, 2011).

As SVMs inicialmente foram criadas para resolver problemas de classificação, e durante algum tempo elas eram utilizadas apenas para a solução deste tipo de problema (GUNN, 1998). Em 1997 foi proposto por Vladmir Vapnik, Steven Golowich e Alex Smola o uso de SVM para resolver problemas de regressão (BASAK et al., 2007). Este modelo de SVM foi chamado de vetores de suporte de regressão (SVR, do inglês *Support Vector Regression*). A Seção 2.3.2 a seguir traz uma abordagem sobre este modelo de SVM.

2.3.2. Regressão de Vetores de Suporte (SVR)

A SVR é baseada na teoria do método de regressão, aplicada na metodologia da aprendizagem supervisionada, descrita na Seção 2.2. Em que, a partir de um conjunto de dados de treinamento $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, (onde x representa os dados de entrada, e y representa os dados de saída), o algoritmo procura obter uma função $f(x)$, que permita prevê o valor de y. Entretanto, a função deve possuir no máximo um desvio ϵ sobre os dados. Esse desvio ϵ representa a distância da margem até a linha regressão, essa margem é chamada de margem de erro. Logo, procura-se uma função $f(x)$ que possua a menor margem de erro, que é caracterizado pelo intervalor de $[\hat{y} - \epsilon, \hat{y} + \epsilon]$, e que seja o mais plano possível (SMOLA e SCHÖLKOPF, 1998).

Figura 25 - Representação da linha de regressão e suas margens.



Na Figura 25 é possível visualizar esta margem de erro (que está representada pela linha tracejada) e a linha de regressão (representada pela linha contínua entre as margens). Nota-se que a maioria dos dados está entre as margens, isso é possível devido que, a margem de erro utiliza os conceitos da margem suave apresentada na SVC (Seção 2.3.1.1.1). Desta maneira, permite que dados fiquem entre as margens, criando assim o que é chamado de tubo de regressão (SMOLA e SCHÖLKOPF, 1998).

Para encontrar a linha de regressão é utilizada a mesma equação (Equação 7) apresentada na SVM para classificação (SVC) (Seção 2.3.1). Assim tem-se que a função de aproximação (GUNN, 1998):

$$f(x) = w \,.\, x + b \tag{31}$$

onde:

w: é o vetor que representa o conjunto de entradas, conhecido como vetor de peso (*weight vector*);

x: são os pontos sobre o hiperplano;

b: é a distancia entre o hiperplano e o ponto de origem;

$w.x$: é o produto cartesiano de w e x.

Buscando obter a função mais paralela aos dados, deve-se, como no método de classificação, realizar a minimização de w (problema de otimização). Assim, tem-se (BASAK et al., 2007):

$$Minimizar_{w,b}\ \frac{1}{2}||w||^2 \tag{32}$$

$$Sujeito\ as\ restrições, \quad \begin{cases} y_i - (w.x_i) - b \leq \epsilon, & i = 1, \dots, n \\ (w.x_i) + b - y_i \leq \epsilon, & i = 1, \dots, n \end{cases} \tag{33}$$

onde:

$||w||$: é a norma euclidiana de w, $||w|| = \sqrt{w.w}$;

y_i: é o valor de y correspondente a x_i

Pode-se notar, que a Equação 32 é semelhante a Equação 12 da classificação, diferenciando-se apenas nas restrições impostas pela Equação 33. O objetivo é encontrar valores de w que proporcione a função se aproximar dos objetos com precisão ϵ, e assim obter a solução do problema otimização.

A função encontrada não consegue atingir todos os dados (como é mostrado na Figura 25), visto que alguns pontos (objetos) violam as restrições. Deste modo, utilizam-se os conceitos semelhantes aos das margens suaves, apresentada na SVC (Seção 2.3.1.1.1), devido que este conceito permite que dados fiquem entre a linha da margem e a linha da função $f(x)$ e, possa haver alguns erros.

São introduzidas variáveis de folga (ξ_i, ξ_i^*) nas restrições, a fim de permitir que alguns erros, para que seja possível solucionar o problema de otimização. Introduzindo as variáveis de folga na Equação 32 e nas suas restrições (Equação 33) (SMOLA e SCHÖLKOPF, 1998), tem-se:

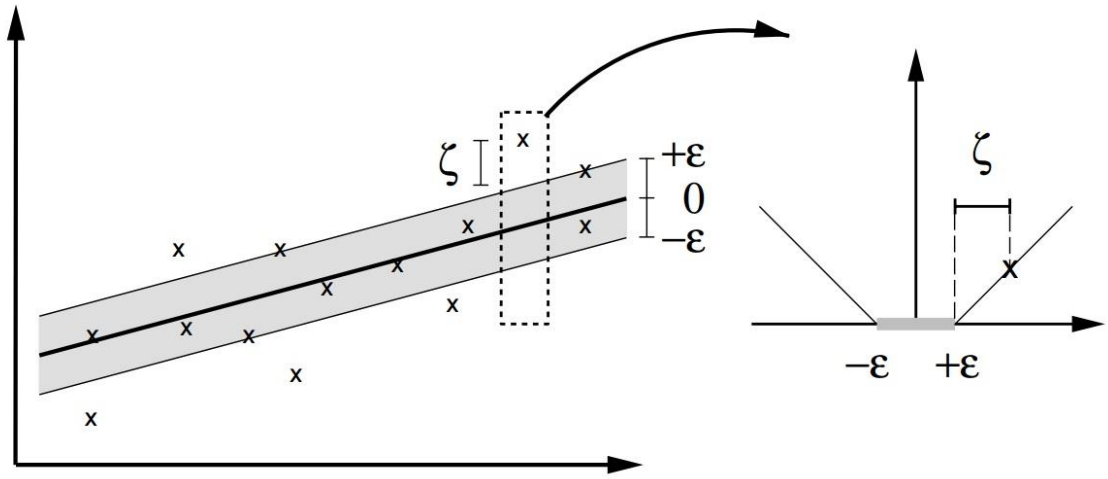
$$Minimizar_{w,b}\ \frac{1}{2}||w||^2 + C(\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{34}$$

$$Sujeito\ as\ restrições, \quad \begin{cases} y_i - (w.x_i) - b \leq \epsilon + \xi_i, & i = 1, \dots, n \\ (w.x_i) + b - y_i \leq \epsilon + \xi_i^*, & i = 1, \dots, n \\ \xi_i, \xi_i^*, & i = 1, \dots, n \end{cases} \tag{35}$$

Onde, C termo de regularização (penalização), que faz o balanço entre a função e a margem (valor dos desvios), a qual os desvios maiores que ϵ são tolerados, e desta forma, sendo tolerados erros. Os pontos (objetos) entre as margens (-ϵ e +ϵ) não sofre a penalização, apenas os valores fora das margens são penalizados, conforme a função de perda *ε-insensitive* (SMOLA e SCHÖLKOPF, 1998):

$$|\xi|_\epsilon := \begin{cases} 0 & , \quad se \quad |\xi| \leq \epsilon \\ |\xi| - \epsilon & , \quad caso\ contrário \end{cases} \tag{36}$$

Figura 26 - Função de perda *ε-insensitive*



Fonte: SMOLA e SCHÖLKOPF (1998)

A Figura 26 representa a situação da Equação 36 graficamente. É possível ver os desvios -ϵ e +ϵ, e assim as margens obtidas por eles, junto com a linha de regressão (linha entre os desvios) e os pontos de erro (pontos fora da região das margens). Apenas os pontos fora da região das margens irão contribuir para o valor custo (C) da Equação 34 (RIBEIRO, 2012).

Procurando resolver o problema de otimização apresentado na Equação 34, devido às restrições da Equação 35, é apresentado esse modelo na forma dual, por meio da função de Lagrange proporcionando eficiência flexibilidade ao algoritmo (BELTRAMI et al., 2010). Transformando a Equação 34 (forma primal) para a formulação dual, são introduzidos multiplicadores de Lagrange não negativos, resultando na Equação 37 (SMOLA e SCHÖLKOPF, 1998):

$$L \coloneqq \frac{1}{2}\left|\left|w\right|\right|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) - \sum_{i=1}^{n}(\alpha_i\xi_i + \alpha_i^*\xi_i^*) - \sum_{i=1}^{n}\lambda_i(\epsilon + \xi_i - y_i + w.x_i + b) - \sum_{i=1}^{n}\lambda_i^*(\epsilon + \xi_i^* + y_i - w.x_i - b) \quad (37)$$

Em que L é a função de Lagrange e $\lambda_i, \lambda_i^*, \alpha_i, \alpha_i^*$ são os multiplicadores de Lagrange (SMOLA e SCHÖLKOPF, 1998).

Para encontrar a solução ótima, é necessário minimizar a função de Lagrange, realizando a maximização dos multiplicadores de Lagrange ($\lambda_i, \lambda_i^*, \alpha_i, \alpha_i^*$), e minimização de w e b (CHAMASEMANI E SINGH, 2011) resultando no problema de otimização. Assim, tem-se a formulação de Equação 38 na sua forma dual (GUNN, 1998):

$$Maximizar \begin{cases} -\frac{1}{2}\sum_{i=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)(x_i.x_j) \\ -\epsilon\sum_{i=1}^{n}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{n} y_i(\alpha_i - \alpha_i^*) \end{cases} \quad (38)$$

$$sujeito\ a \sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0 \qquad e \qquad \alpha_i, \alpha_i^* \in [0, C] \quad (39)$$

Desta maneira a variável w da forma primal (Equação 25) é definida como (SMOLA e SCHÖLKOPF, 1998):

$$w = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)x_i \quad (40)$$

e a variável b da Equação 25 é definida como (GUNN, 1998):

$$b = -\frac{1}{2}(w.(x_i + x_s)) \quad (41)$$

Satisfazendo as condições de Karush-Kuhn-Tucker (KKT) (BASAK et al., 2007):

$$a_i(\epsilon + \xi_i - y_i + (w.x_i) + b) = 0, \qquad \forall i = 1, \dots, n$$
$$a_i^*(\epsilon + \xi_i + y_i - (w.x_i) - b) = 0, \qquad \forall i = 1, \dots, n \tag{42}$$

$$(C - a_i) = 0, \qquad \forall i = 1, \dots, n$$
$$(C - a_i^*) = 0, \qquad \forall i = 1, \dots, n \tag{43}$$

$$a_i a_i^* = 0, \qquad \forall i = 1, \dots, n \tag{44}$$

As condições mostram que apenas para exemplos fora das margens os multiplicadores de Lagrange serão diferentes de zero. Sendo assim os pontos localizados entre as margens ($-\epsilon$, $+\epsilon$), os multiplicadores de Lagrange podem ser iguais a zero e logo não utilizados no cálculo de *w* (BELTRAMI et al., 2010). Desta maneira, apenas os dados fora das margens são utilizados no cálculo de *w* e por isso são chamados de vetores de suporte (BASAK et al. , 2007).

Sendo assim, a função de regressão (Equação 31) pode ser reescrita da seguinte maneira (SMOLA e SCHÖLKOPF, 1998):

$$f(x) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*)(x_i.x) + b \tag{45}$$

A Equação 44 é chamada de expansão dos vetores de suporte *(*em inglês *Support Vector expansion*). Desta forma, demonstra que a complexidade da função de aproximação (função de SV) é independente da dimensionalidade do conjunto de treinamento, mas depende apenas do número de vetores de suporte SV (RIBEIRO, 2012)

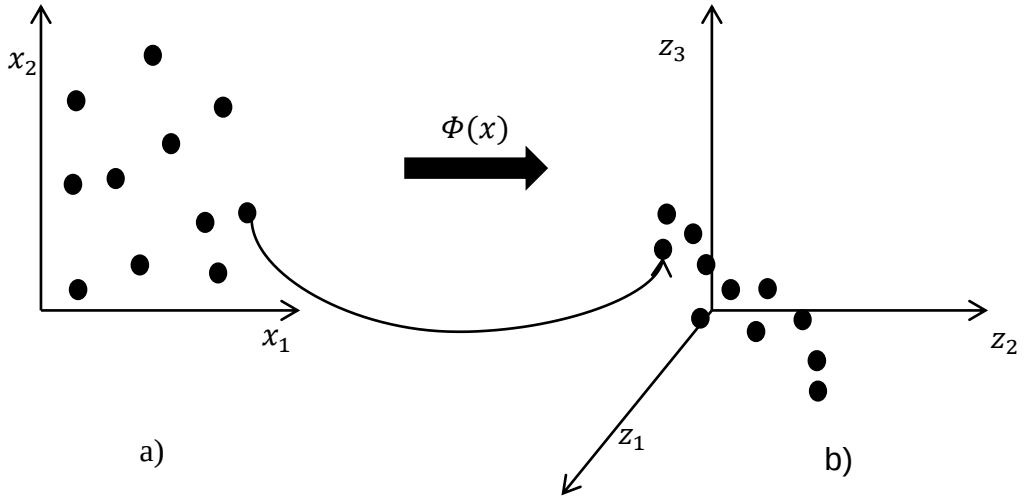
As SVRs também podem ser aplicadas na predição de dados não lineares. Sendo assim, a Seção 2.3.2.1 apresenta a teoria sobre este tipo de aplicação da SVR.

2.3.2.1. SVR não linear

Da mesma forma que a SVC não linear, a SVR também pode ser utilizada para predizer valores de dados não lineares (dados que não podem ser separados linearmente). Na SVR não linear é necessário o mapeamento (Φ) dos dados (conjunto de treinamento) não lineares do seu espaço original para o espaço de características (espaço com dimensões maiores) (BASAK et al., 2007), da mesma forma como é feito na SVC não linear (Seção 2.3.1.2). A

Figura 27 apresenta um exemplo do mapeamento dos dados, do espaço original para o espaço de características (*features spaces).*

Figura 27 - Dados em seu espaço original (a), para o espaço de características (b)



Na

Figura 27 é possível notar que todos os objetos, do conjunto de dados, quando estão no espaço original (

Figura 27 (a)) não podem ser representados por uma linha reta. Entretanto, quando são mapeados para o espaço de características é possível os dados se tornam dados lineares (

Figura 27), e desta forma é possível encontrar uma linha, reta, que se ajuste bem aos dados (SMOLA e SCHÖLKOPF, 1998).

Contudo, para a SVR, a transformação desses dados não é importante, a SVR necessita apenas do produto escalar dos pontos (objetos) do conjunto de dados (LORENA e CARVALHO, 2007). Para obter o produto escalar dos pontos é utilizado funções que permitam calcular este procedimento. Essas funções são chamadas de funções Kernel (GUNN, 1998). As funções Kernel recebem dois pontos (objetos) (x, x') e calculam o produto escalar destes pontos (objetos) no espaço de características.

Com a aplicação da função Kernel no problema de otimização demonstrado na Equação 38, tem-se (BASAK et al., 2007):

$$Maximizar \begin{cases} -\frac{1}{2}\sum_{i=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)K(x_i.x_j) \\ -\epsilon\sum_{i=1}^{n}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{n} y_i(\alpha_i - \alpha_i^*) \end{cases} \quad (46)$$

$$sujeito\ a \sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0 \qquad e \quad (47)$$

$$0 \leq \alpha_i, \alpha_i^* \leq C, \quad i = 1, \ldots, n.$$

Deste modo o cálculo de w (Equação 40) é redefinido como (BASAK et al., 2007):

$$w = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\Phi(x_i) \quad (48)$$

E a equação de b (Equação 40) é definida como (GUNN, 1998):

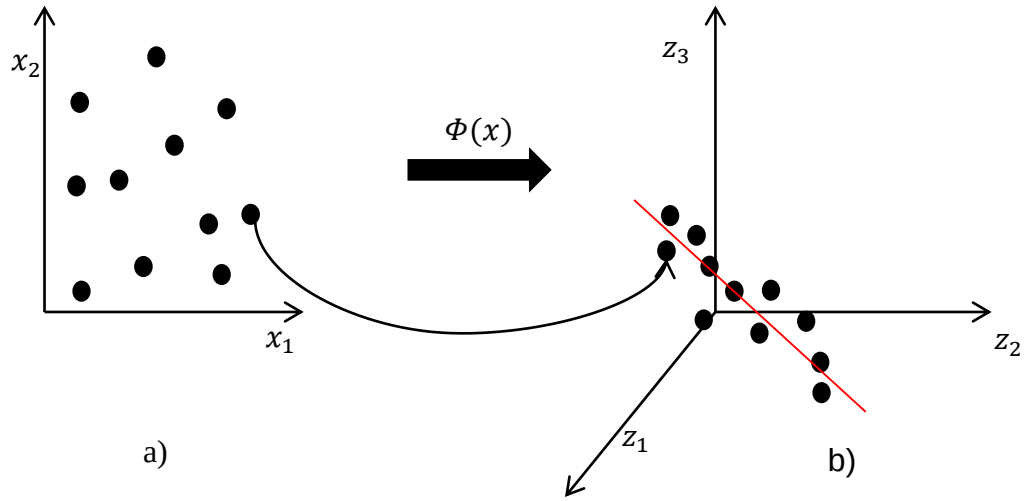
$$b = -\frac{1}{2}(\alpha_i - \alpha_i^*).K(x_i, x_s) \quad (49)$$

Resultando na função de regressão da SVR para dados não lineares como sendo (SMOLA e SCHÖLKOPF, 1998):

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)K(x_i.x) + b \quad (50)$$

A partir da Equação 50 é possível encontrar a linha de regressão que consiga se ajustar bem aos dados, e realizar a predição de novos valores de saída, a partir de entradas desconhecidas. Como é apresentado na Figura 28, em que a linha de regressão é representada pela linha na cor vermelha da Figura 28 (b) (espaço de características).

Figura 28 - Dados em seu espaço original (a), para o espaço de características e a linha de regressão (b)



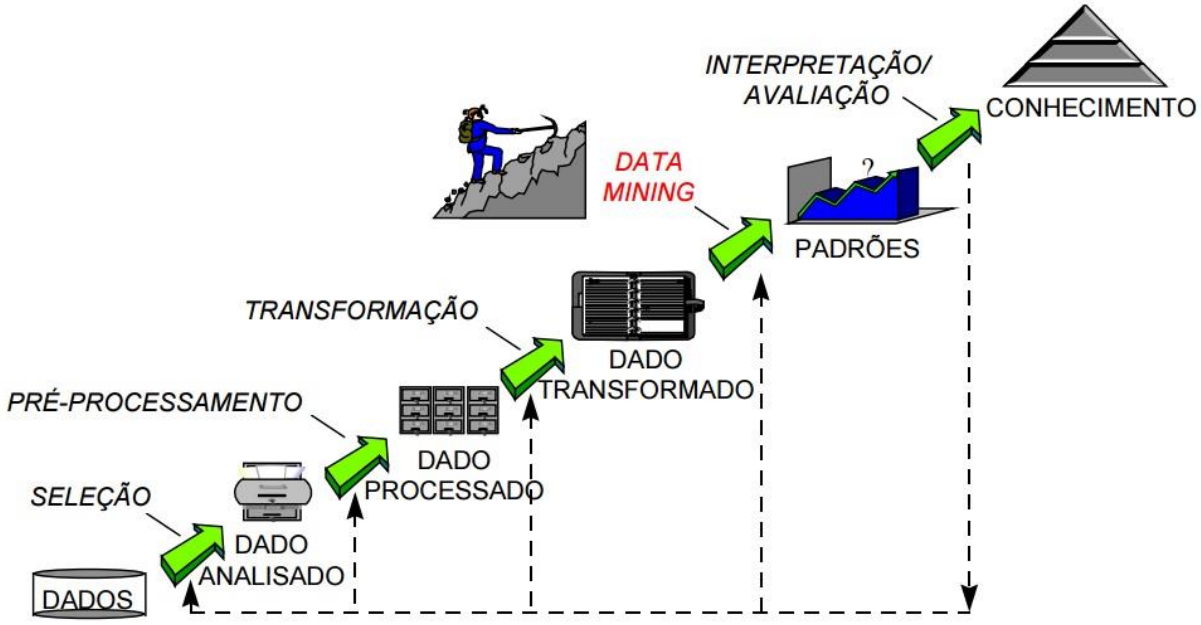
Logo, a diferença entre a SVR linear e a SVR não linear está na inclusão da função Kernel, que permite que problema de otimização seja resolvido no espaço de características e não no espaço original (como é realizado na SVR linear).

Diversas funções Kernel podem ser utilizadas para o cálculo do produto interno. No entanto, para que essas funções sejam aceitas, elas devem respeitar as condições de Mercer (LORENA e CARVALHO, 2007; SMOLA e SCHÖLKOPF, 1998). A partir dessas condições é possível garantir que a função Kernel calcule o produto escalar dos dados, no espaço de características (BASAK et al., 2007). As funções Kernel que podem ser empregadas são as mesmas que podem ser utilizadas no SVC, se destacando, entre elas, as funções Polinomiais, Sigmoidais e os RBF (*Radial-Basic Function*) (FACELI et al., 2011; GUNN, 1998).

2.4. Extração de Conhecimento

O processo de Extração de Conhecimento *(Knowledge Discovery in Database – KDD)* é definido como um processo, interativo e iterativo, de extração de informações, desconhecidas (devido que não se sabe os resultados que podem ser encontrados) e úteis a partir de uma base de dados (FAYYAD et al., 1996). É um processo interativo que, no momento da execução do processo é necessária a interação do usuário para a realização algumas tarefas, por exemplo, para realizar a seleção dos dados ou um pré-processamento dos dados (seleção dos dados e pré-processamento dos dados são conceitos que são apresentando no decorrer desta seção). A iteração refere-se à possibilidade de os processos serem executados diversas vezes. O processo KDD possui várias etapas (passos) que devem ser executados para a busca do conhecimento, as quais são apresentadas na Figura 29.

Figura 29 - Etapas do KDD



Fonte: (ALMEIDA et al., 2004)

A Figura 29 apresenta as cinco etapas do KDD, sendo elas: *Seleção*, *Pré-processamento*, *Transformação*, *Data mining*, *Intepretação e Avaliação*. A etapa *Seleção* é a etapa em que se realiza a escolha do conjunto de dados que será utilizado, e quais serão as possíveis variáveis que serão analisadas (PRASS, 2012). Este conjunto de dados pode possuir diversos formatos, como arquivos de textos, banco de dados, planilhas, imagens entre outros.

O *Pré-processamento* é a etapa onde é realizada a eliminação, tratamento, ou recuperação dos dados considerados ruídos ou ausentes (ALMEIDA et al., 2004). Para a realização desta etapa é necessário ter um conhecimento avançado do domínio para que se possa analisar se um dado é realmente um ruído ou não.

A *Transformação* é a etapa em que os dados são organizados e formatados para o formato que possa ser aplicado ao algoritmo.

Na etapa *Data mining* (Mineração de Dados) é realizado o processo de procurar padrões, associações e correlações nos dados colhidos. Para este processo são utilizados algoritmos de Aprendizado de Máquina, com objetivo de extrair informações e conhecimentos considerados úteis para a aplicação.

Na *Interpretação e Avalição* o conhecimento obtido pelo *Data mining* é analisado (MACEDO e MATOS, 2010; ALMEIDA et al., 2004). Se os resultados não foram satisfatórios, o processo de extração de conhecimento pode ser reiniciado, alterando as etapas apresentadas anteriores.

3 MATERIAIS E MÉTODOS

O presente trabalho possuiu como finalidade metodológica a pesquisa aplicada, visto que o trabalho procurou aplicar o algoritmo de Máquina de Vetores de Suporte (SVM) para a predição de dados climáticos, e para isso são utilizados uma base de dados, algoritmos de aprendizagem de máquina, ferramentas de desenvolvimento, e conhecimentos de outros trabalhos relacionados ao tema. Esta seção apresenta os materiais utilizados e a metodologia adotada para o desenvolvimento deste trabalho.

3.1. Materiais

A base de dados utilizadas foi obtida pelo site inmet.gov.br, contendo as informações climáticas referentes a região de Palmas – TO, do dia de 01 de janeiro de 2010, a 01 de junho de 2015.

Foi utilizada a implementação do SVM fornecida pela biblioteca Scikit-learn (PEDREGOSA, 2011). Esta implementação é desenvolvida em linguagem Python, e devido a isso, está linguagem foi definida com a linguagem de programação que foi utilizada em todo o trabalho.

Para o desenvolvimento do software foi utilizado o framework Django, que é desenvolvido em Python também. Este framework permite a criação e gerenciamento de páginas web, com a aplicação de funções e métodos em escrito em Python.

3.2. Métodos

Procurando atender os objetos do trabalho foi realizado a aplicação do processo KDD, se iniciando a partir da etapa de *Seleção*, com a obtenção de um histórico de dados climáticos de uma determinada região. Após, foi executado as etapas de *Pré-processamento* e de *Tratamento dos dados*. Concluídas estas etapas, foi iniciada a etapa de *Data mining* e logo em seguida a etapa de *Intepretação e Avaliação*, com o objetivo de verificar se os resultados obtidos possuem relação direta com o objetivo geral do trabalho.

A partir disso, verificar se a aplicação do algoritmo de Máquina de Vetores de Suporte pode ser utilizada para a predição de dados climáticos. Detalhes sobre a forma como foi executado cada uma destas etapas é apresentado na Seção 3.2.1.

3.2.1. Execução dos métodos

Iniciando a aplicação do processo de KDD, foi realizada a etapa de *Seleção*, em que foram obtidos, de forma manual, dados meteorológicos através do site do Instituto Nacional de Meteorologia (INMET)(inmet.gov.br) sendo selecionados os dados da região Norte, mais precisamente da cidade de Palmas no estado do Tocantins. Os dados colhidos foram do período do dia 01 de janeiro de 2010 a 01 de junho de 2015. Estes dados contêm informações de *data*, *hora*, *temperatura máxima* (medida em grau Celsius) (TMx), *temperatura mínima* (medida em grau Celsius) (TMi), *precipitação* (medida em milímetros) (PREP), *insolação* (medida em horas) (INS), *evaporação piche* (medida em milímetros) (EVP), *temperatura compensada média* (medida em grau Celsius) (TCM), *velocidade do vento* (medida em metros por segundos), e *umidade relativa média* (medida em porcentagem) (URM).

Na etapa de *Pré-processamento* são removidas todas as informações que não serão utilizadas na aplicação, como cabeçalho que contenham a fonte dos dados e outras informações. Os dados da *velocidade do vento* foram descartados devido que os dados estavam inconsistentes, havendo muitos valores nulos ou próximos de zero. Concluída a etapa de *Pré-processamento*, foi iniciado a etapa de *Transformação*, transformando a base de dados no formato que é aceito pelo algoritmo, que é o formato *csv*.

Na etapa do *Data mining* foram realizados procedimentos para verificar a correlação de cada variável em relação à variável de umidade relativa média (que é a que se deseja prever ao final do trabalho). Foram gerados gráficos e cálculos de correlação, para verificar quais variáveis possuem a maior correlação com a variável de umidade relativa média, em outras palavras, procura-se verificar quais as variáveis possuem uma relação com a variável de umidade relativa média. Para que duas variáveis possuam uma correlação considerada forte, ela deve se aproximar o máximo possível de 1 ou de -1, sendo que quanto mais próximo de 0 (zero) for o coeficiente de correlação entre as variáveis, menor é a correlação entre elas.

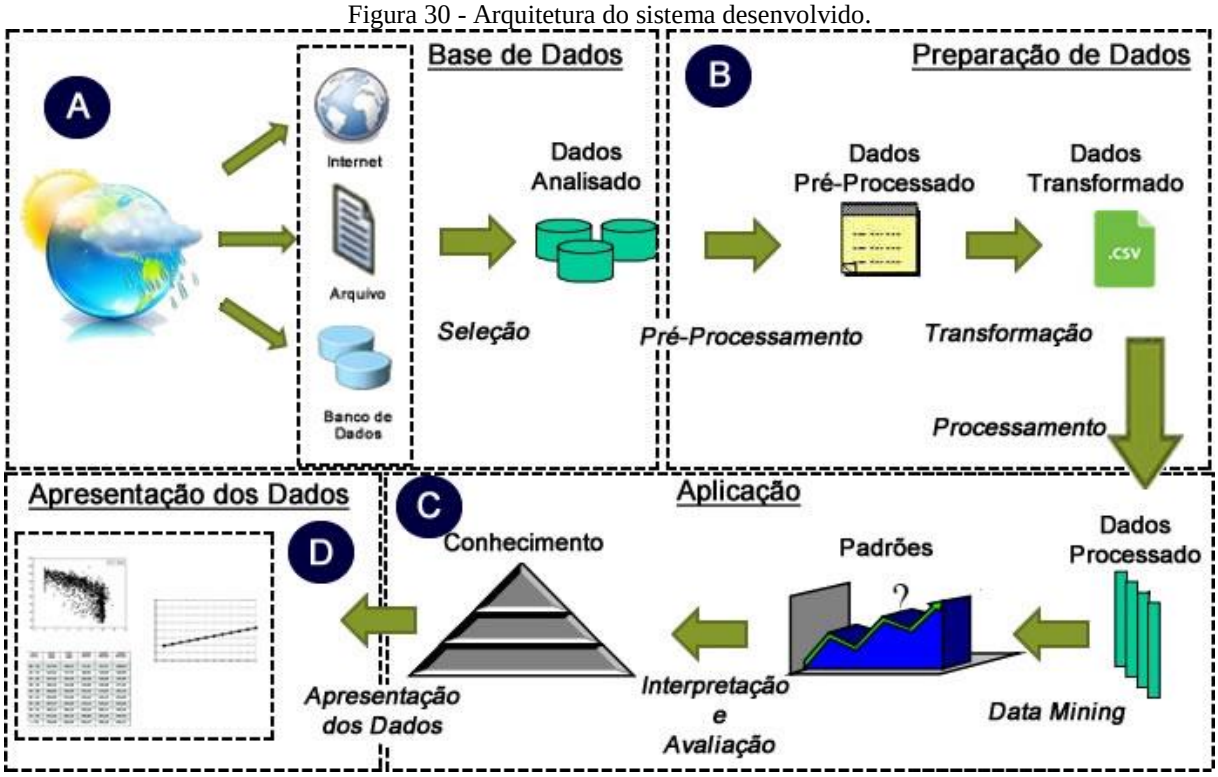
Ao termino do *Data mining* foi iniciada a etapa de *Intepretação e Avaliação*, e estes resultados são apresentados na Seção 0, referentes aos resultados e discussões.

As etapas de *Data mining* e de *Intepretação e Avaliação* foram realizadas diversas vezes, com diversas combinações de variáveis, buscando encontrar o melhor modelo, ou o modelo mais ideal, que consiga realizar a predição dos valores da umidade relativa média. Estas etapas, e suas repetições são apresentadas na Seção 0, que apresenta os resultados obtidos no trabalho.

4 RESULTADOS E DISCUSSÃO

Uma vez que o presente trabalho tem como objetivo aplicar o algoritmo de Máquina de Vetor de Suporte (SVM) para a predição de dados climáticos, foi definida uma arquitetura para o sistema desenvolvido, baseando-se no processo KDD, que foi apresentado na Seção **3.2.1,** com o intuito de oferecer uma melhor visualização das partes que compõe o sistema, e uma explicação mais detalhada de cada uma das partes. Esta arquitetura é apresentada na Figura 30.

Figura 30 - Arquitetura do sistema desenvolvido.



A Figura 30 apresenta a arquitetura do sistema desenvolvido, com cada umas das partes existentes no sistema. Para explicar melhor as etapas apresentadas na Figura 30, elas foram divididas em quatro módulos, sendo eles, o módulo *Base de Dados* (Figura 30-A), módulo *Preparação dos Dados* (Figura 30-B), módulo *Aplicação* (Figura 30-C), e módulo *Apresentação dos Dados* (Figura 30-D).

Conforme apresentado na Figura 30, a fonte de dados utilizada como dados históricos (módulo A - *Base de Dados)* para se realizar a predição pode vir de diferentes fontes, por exemplo, sites da internet, arquivos, bancos de dados, o importante é que tenham os atributos necessários para que possam ser avaliados. Para este contexto, os atributos são os valores de temperatura máxima, temperatura mínima, temperatura compensada média, precipitação, insolação, evaporação e umidade relativa média.

Uma vez que a fonte de dados (módulo A - *Base de Dados*) tenha sido definida, é necessário preparar os dados (módulo B - *Preparação dos Dados* - Figura 30-B) para adequar

ao formato que é aceito pelo sistema (esta preparação é realizada de forma manual), gerando um arquivo no formato *csv* para ser utilizado na aplicação.

Com o arquivo *csv* gerado (módulo B - *Preparação dos Dados)* a aplicação recebe os dados e inicia a tarefa de busca de padrões entre os dados (módulo C - *Aplicação*). A partir destas etapas, é encontrado o "*conhecimento*", e os resultados obtidos pela aplicação são apresentados ao usuário (módulo D - *Apresentação dos Resultados)*. Cada uma das etapas apresentadas na Seção **3.2.1**, que foram descritas resumidamente nos parágrafos anteriores, e são explicadas com mais detalhes na Seção 4.1.

4.1. Detalhes sobre a arquitetura do sistema

Esta seção apresentas os módulos, desenvolvidos na arquitetura, como mais detalhes, explicando as etapas e os resultados de cada módulo. Iniciando-se pelo módulo *Base de Dados* e módulo *Preparação dos dados*.

4.1.1. Base de Dados e Preparação dos Dados

O módulo *Base de Dados* consiste na etapa de obtenção de dados climáticos anteriores, para que se possa aplicar estes dados na aplicação. A obtenção destes dados climáticos foi realizada através da etapa de *Seleção*, que resultou no conjunto de dados com as seguintes variáveis: *data*, *hora*, *temperatura máxima*, *temperatura mínima*, *precipitação*, *insolação*, *evaporação piche*, *temperatura compensada média* e *umidade relativa média*.

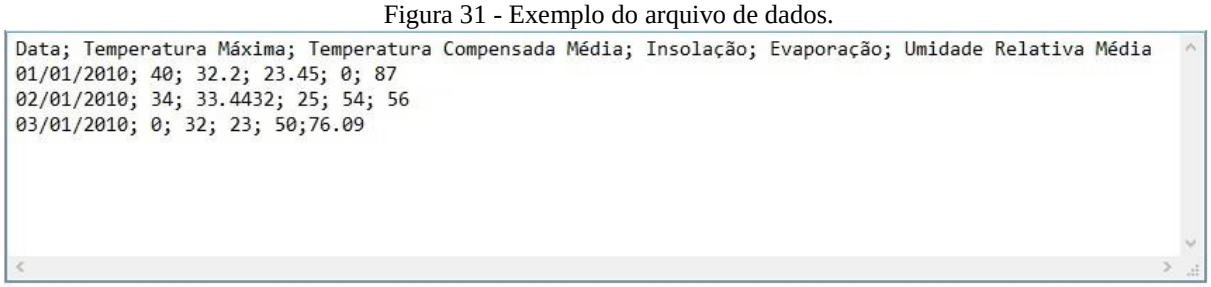
A Figura 30 demonstra que os dados podem ser obtidos por qualquer um dos meios apresentados, contanto que possuam os valores que são necessários para a utilização do sistema. A etapa de *Seleção* gera como resultado um artefato denominado como *Dado Analisado*. Após a criação deste artefato o módulo *Base de Dados* é finalizado, e é iniciado o módulo de *Preparação dos dados*.

Este módulo (módulo *Preparação de Dados*) se inicia com a etapa de *Pré-processamento*, em que é realizada a remoção e/ou tratamento dos dados nulos. Gerando ao final desta etapa um artefato denominado como *Dado Pré-processado*.

Uma vez que a etapa de *Pré-processamento* esteja concluída, é iniciada etapa de *Transformação.* Esta etapa realiza a transformação do artefato *Dado Pré-processado* em um arquivo (*Dados Transformado),* no formato *csv*, com os valores da cada variável. A primeira linha deste arquivo deve conter o nome de cada uma das variáveis utilizadas. Os dados são apresentados por dia, e os dados de cada dia são colocados em uma única linha, separando-os

por ponto e vírgula (“;”). A Figura 31 apresenta exemplos da visualização deste arquivo (*Dados Transformados).*

Figura 31 - Exemplo do arquivo de dados.



A Figura 31 apresenta exemplos da visualização do arquivo gerado na preparação dos dados. Na Figura 31é apresentada a visualização do arquivo em um leitor de texto simples, como bloco de notas. Com a conclusão da etapa *Transformação,* o arquivo (*Dado Transformado*) obtido é fornecido para a aplicação, onde é extraído o conhecimento contido nos dados. Os detalhes das etapas realizadas pela aplicação (módulo *Aplicação*) são apresentados na Seção 4.1.2.

4.1.2. Aplicação e Apresentação dos Dados

A partir do arquivo gerado pelo módulo *Preparação dos Dados*, a aplicação executa a etapa *Processamento*. Nesta etapa é realizada a remoção, e organização dados. São removidos dados com valores nulos das variáveis que serão utilizadas. Essa remoção ocorre da seguinte forma: entre as variáveis que serão utilizadas para o *Data mining* (etapa seguinte - Figura 30) é realizada uma verificação em cada uma delas, procurando encontrar valores nulos. Se um dado nulo for encontrado, todos os dados daquele dia são descartados. Essa remoção é feita de forma automática pela aplicação, gerando ao final um artefato denominado como *Dado Processado*.

Com a etapa de *Processamento* concluída, inicia-se a etapa de *Data mining*. É neste momento que se inicia a aplicação do algoritmo SVR fornecido pela biblioteca escolhida. O SVR recebe como parâmetros os valores das variáveis, independente e dependente, e os resultados desta aplicação são armazenados em um artefato chamado de *Padrões*.

A partir disso, a etapa de *Interpretação e Avaliação* realiza a análise dos valores contidos no artefato *Padrões*. Os valores verificados são: o erro médio, erro padrão, erro médio absoluto, coeficiente de determinação e coeficiente de correlação, e a partir disso é dito se o modelo é satisfatório ou não, gerando assim o artefato de *Conhecimento.* Os dados obtidos no conhecimento são apresentados (módulo *Apresentação dos Dados*) utilizando gráficos e tabelas, com o intuito de possibilitar a visualização e análise do modelo obtido, permitindo-se que seja possível concluir se o modelo pode ou não realizar a predição destes dados.

4.2. Implementação

Nesta seção são apresentados trechos de código do software desenvolvido (na linguagem de programação *Python*). O primeiro trecho do código apresentado é referente ao método `pre_Processamento`, que realiza a tarefa de processamento dos dados.

```
1. def pre_Processamento(csv_PreProcessado):
2. # Arrays
3.       lst_Data, lst_Hora = [], []
4.       lst_Precip, lst_TMx, lst_TMi, lst_Ins = [],[],[],[]
5.       lst_Evap, lst_TCM, lst_URM, lst_Vento = [],[],[],[]
6.       cont = 0
7.       for [data, precip, temp_Max, temp_Min, insolacao, evaporacao,
   TCM, umidade, velocidade_Vento] in csv_PreProcessado:
8.             if (cont == 0):
9.                   cont+=1
10.                  continue
11.            lst_Data.append(data)
12.            lst_Precip.append(float(precip))
13.            lst_TMx.append(float(temp_Max))
14.            lst_TMi.append(float(temp_Min))
15.            lst_Ins.append(float(insolacao))
16.            lst_Evap.append(float(evaporacao))
17.            lst_TCM.append(float(TCM))
18.            lst_URM.append(float(umidade))
19.            lst_Vento.append(velocidade_Vento)
20.            cont+=1
21.      cont-=1
22.      i = 0
23.      lst_Data2, lst_Precip2 = [], []
24.      lst_TMx2,    lst_TMi2, lst_Ins2, = [],[],[]
25.      lst_Evap2, lst_TCM2, lst_URM2, lst_Vento2 = [],[],[],[]
26.      while (i<cont):
27.          if(lst_Data[i] == lst_Data[i+1]):
28.             lst_Data2.append(lst_Data[i])
29.             if(lst_TMx[i] >= lst_TMx[i+1] ):
30.                  lst_TMx2.append([lst_TMx[i]])
31.             else:
32.                  if ( lst_TMx[i+1] >= lst_TMx[i]):
33.                        lst_TMx2.append([lst_TMx[i+1]])
34.            if(lst_TMi[i] >= lst_TMi[i+1] ):
35.                  lst_TMi2.append([lst_TMi[i]])
36.            else:
37.                  if ( lst_TMi[i+1] >= lst_TMi[i]):
38.                        lst_TMi2.append([lst_TMi[i+1]])
39.            if (lst_Precip[i] >= lst_Precip[i+1]):
40.                  lst_Precip2.append([lst_Precip[i]])
41.            else:
42.                  if (lst_Precip[i+1] >= lst_Precip[i]):
43.                        lst_Precip2.append([lst_Precip[i+1]])
44.            if(lst_Ins[i] >= lst_Ins[i+1]):
45.                  lst_Ins2.append([lst_Ins[i]])
46.            else:
47.                  if(lst_Ins[i+1] >= lst_Ins[i]):
48.                        lst_Ins2.append([lst_Ins[i+1]])
49.            if (lst_Evap[i]>= lst_Evap[i+1]):
50.                  lst_Evap2.append([lst_Evap[i]])
51.            else:
```

```
52.                   if(lst_Evap[i+1] >= lst_Evap[i]):
53.                         lst_Evap2.append([lst_Evap[i+1]])
54.             if (lst_TCM[i] >= lst_TCM[i+1]):
55.                   lst_TCM2.append([lst_TCM[i]])
56.             else:
57.                   if(lst_TCM[i+1] >= lst_TCM[i]):
58.                         lst_TCM2.append([lst_TCM[i+1]])
59.             if ( lst_URM[i] >= lst_URM[i+1]):
60.                   lst_URM2.append(lst_URM[i])
61.             else:
62.                   if(lst_URM[i+1] >= lst_URM[i]):
63.                         lst_URM2.append(lst_URM[i+1])
64.             if( lst_Vento[i] >= lst_Vento[i+1]):
65.                   lst_Vento2.append(float(lst_Vento[i]))
66.             else:
67.                   if (lst_Vento[i+1] >= lst_Vento[i]):
68.                   lst_Vento2.append(float(lst_Vento[i+1]))
69.             i+=2
70.             continue
71.           else:
72.             i+=1
73.             i =0
74.       return [lst_TMx2, lst_TMi2, lst_Precip2, lst_Ins2,
   lst_Evap2,lst_TCM2, lst_URM2, lst_Vento2]
```

Este código apresenta o método de processamento dos dados, em que o método recebe como parâmetro o caminho completo do arquivo, e os dados desse arquivo são salvos nos *arrays* referentes a cada variável. A partir desses *arrays* é realizada a união dos dados referentes ao um mesmo dia. Ao término do processamento são retornados os *arrays* com os valores referentes a cada variável, por exemplo, o *array* dos valores de temperatura máxima (*lst_TMx2*). O método que realiza a criação do modelo SVR é apresentado a seguir.

```
75.def cria_Modelo(lst_Independente, lst_Dependente,custo):
76.       svr_lin = SVR(kernel='linear', C=custo)
77.       svr_lin.fit(lst_Independente, lst_Dependente)
78.       return [svr_lin, infoModelo(svr_lin, lst_Independente,
   lst_Dependente)]
79.
80.def infoModelo(svr_lin, lst_Independente, lst_Dependente)
81.       svr_lin.fit(lst_Independente, lst_Dependente)
82.       y_Predito = svr_lin.predict(lst_Independente)
83.       er =median_absolute_error(lst_Dependente, y_Predito)
84.       y_medio = np.mean(lst_Dependente)
85.       r2 = svr_lin.score(lst_Independente, lst_Dependente)
86.       if(r2<0):
87.             r2=r2*(-1)
88.       RR = math.sqrt(r2)
89.       resposta =  ("Erro Medio Absoluto;  Vl. Medio Y; Coef.
   Determinacao; Coef. Correlacao; total; acertos; %; erros; %\n")
90.       resposta = resposta + (str(er).replace(".", ",")+";
   "+str(y_medio).replace(".", ",")+"; "+str(r2).replace(".", ",")+";
   "+str(RR).replace(".", ",")+";")
91.       resposta = resposta + verifica_Acertos(y_Predito,
   lst_Dependente)+"\n"
92.       return resposta
93.
94.
95.def verifica_Acertos(Ypredito, Yreal):
```

```
96.    acertos ,erros = 0,0
97.    erroMedioAbsoluto = median_absolute_error(Yreal, Ypredito)
98.    if(erroMedioAbsoluto<0):
99.        erroMedioAbsoluto = erroMedioAbsoluto*(-1)
100.    intervalo = erroMedioAbsoluto*1.960
101.    contador =0
102.    while contador < len(Ypredito):
103.        if(float(Yreal[contador]) <= (float(Ypredito[contador]) +
   float(intervalo) )):
104.            if( float(Yreal[contador]) >= (
   float(Ypredito[contador]) - float(intervalo))):
105.                acertos += 1
106.            else:
107.                erros += 1
108.        else:
109.            erros += 1
110.        contador+=1
111.    resp = (str(len(Ypredito))+"; "+str(acertos)+"; ")
112.    if(acertos > 0):
113.        resp +=
   (str((acertos*100)/len(Ypredito)).replace(".",",")+"; ")
114.    else:
115.        resp += ("0% ;")
116.    if(erros > 0):
117.            resp+= (str(erros)+";
   "+str((erros*100)/len(Ypredito)).replace(".",",")+";")
118.    else:
119.        resp+= (str(erros)+"; 0%;")
120.    return resp
```

O método que realiza a criação do modelo é denominado como *cria_Modelo()*. Este método recebe como parâmetro uma lista com as variáveis independentes, uma lista com os valores da variável dependente e o valor da função de custo. O modelo SVR é criado na linha 76, o seu treinamento ocorre pelo código apresentado na linha 77. Os resultados desse modelo são obtidos através do método *infoModelo()*.

O método *infoModelo()* recebe como parâmetro o modelo SVR, a lista de variáveis independentes e a lista de variáveis dependentes, e retorna os valores obtidos pelo modelo passado como parâmetro. Os resultados retornados são: o erro médio absoluto (linha 83), o valor médio da variável dependente (linha 84), o coeficiente de determinação (linha 85), coeficiente de correlação (linha 88), quantidade de dados, quantidade predições corretas e quantidade de predições erradas (por meio do método de *verifica_Acertos()*).

O retorno do método *infoModelo()* é o SVR criado, e uma *string* com os resultados do modelo criado. Os códigos apresentados são referentes apenas ao módulo *Aplicação*, e são apresentados com intuito de possibilitar um melhor entendimento dos processos realizados nesta etapa. Este processo foi executado diversas vezes, à procura do modelo que obtivesse os melhores resultados para a predição de valores climáticos. Este processo de busca do melhor modelo é apresentado na Seção 4.3.

4.3. Busca pelo melhor modelo para predição

Com toda a arquitetura e seus processos definidos, foi iniciada uma busca pelo melhor modelo que realize a predição destes dados climáticos, com o objetivo de predizer o valor de *umidade relativa média*. O módulo *Aplicação* e o módulo *Apresentação dos Dado*s foram executados repetidas vezes, iniciando-se com cada uma das variáveis em relação à variável da *umidade relativa média*, para que assim fosse possível verificar o erro médio da predição e o grau de correlação entre elas entre os resultados, de cada modelo.

Em cada modelo gerado foi verificado os resultados de cada variável com a *umidade relativa média* (que é a que se deseja prever ao final do trabalho) a partir da etapa de *Apresentação dos Dados*. Feito isso, foi verificado o coeficiente de correlação entre as variáveis independentes e a variável dependente (*umidade relativa média*), através de gráficos, cálculo de correlação e valores de erros, possibilitando verificar quais variáveis possuíam os melhores resultados em relação a predição da *umidade relativa média*. Em outras palavras, procurou-se verificar quais as variáveis possuem as maiores relações com a variável umidade relativa média com o menor valor de erro. Para que as variáveis possuam uma correlação (relação) considerada forte, o coeficiente de correlação dela deve se aproximar o máximo possível de 1 ou -1, sendo que quanto mais próximo de 0 (zero) for o coeficiente de correlação entre as variáveis, menor é a correlação entre elas, e quanto mais próximo de um maior será a correlação entre elas. Os resultados que serão apresentados foram obtidos por meio da utilização do algoritmo SVR, disponibilizado pela biblioteca Scikit-learn.

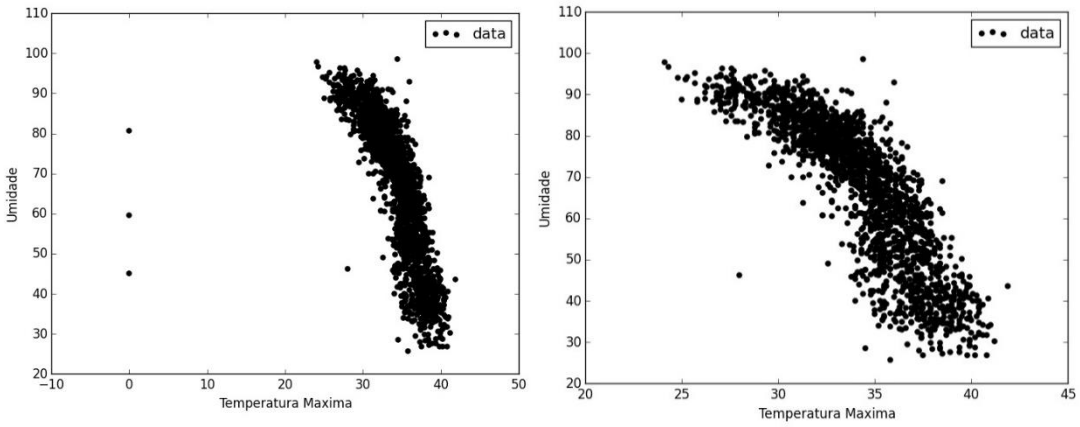
4.3.1. Aplicação do SVR com regressão linear simples

A busca pelo melhor modelo iniciou-se com a aplicação de todas as variáveis da base de dados (*temperatura máxima, temperatura mínima, precipitação, temperatura compensada média, insolação e evaporação*) em relação a variável de *umidade relativa média*. Iniciando-se pela aplicação da variável de *temperatura máxima* (TMx) vs *umidade relativa média* (URM).

4.3.1.1. Temperatura Máxima vs Umidade Relativa Média

A aplicação das variáveis de *temperatura máxima* e *umidade relativa média* ao algoritmo SVR, iniciou-se pela geração dos dados em um gráfico de dispersão, que é apresentado na Figura 32.

Figura 32 – Gráficos com os valores de Temperatura Máxima (TMx) vs Umidade Relativa Média (URM)

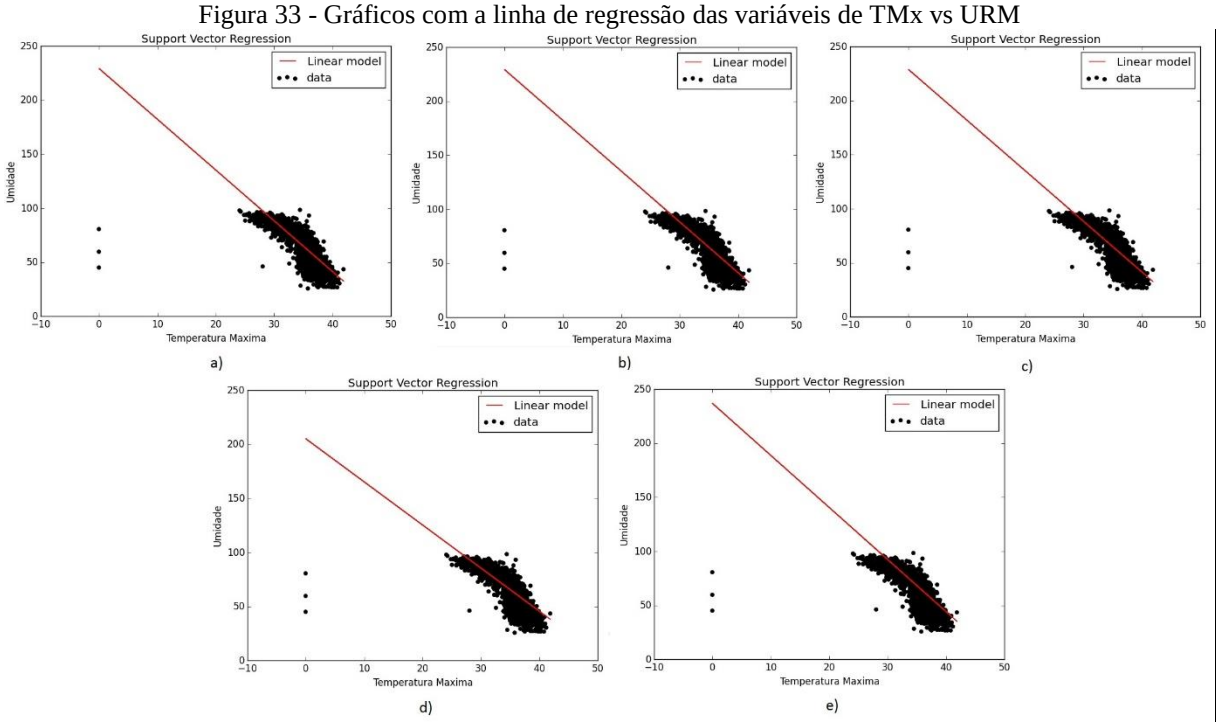


a) b)

Com a geração do gráfico apresentado na Figura 32-a) foi verificado que existiam dados que possuíam valores de *temperatura máxima* iguais a zero. Analisando as características da região de onde foram coletados os dados (Palmas, no estado do Tocantins), foi definido que dados que possuíssem o valor de temperatura máxima igual a zero devem ser considerados erros, e desta forma serem descartados (o processo de remoção de dados com valores iguais a zero, denominado como Tratamento-1, foi aplicado a todas as variáveis disponíveis, e é realizado na etapa de *Processamento*). O gráfico da Figura 32-b) é gráfico gerado após a remoção dos dados nulos. Com a visualização do gráfico da Figura 32-b) é possível notar que estas duas variáveis possuem uma correlação, porém somente com o cálculo do coeficiente de correlação foi possível determinar qual é o grau de correlação entre as variáveis.

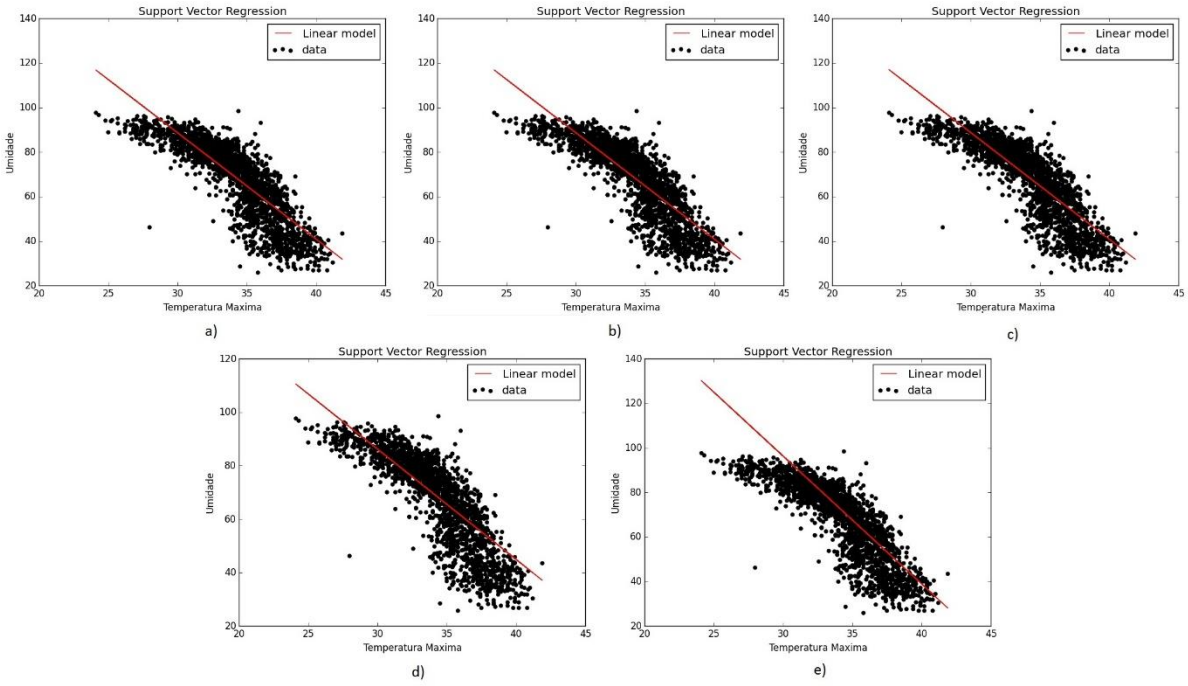
Para a realização do cálculo do coeficiente de correlação, foi aplicado as variáveis TMx e URM ao algoritmo SVR, variando o valor da função de custo (a tolerância a erros do algoritmo). Para cada variação da função de custo (C), foi gerado um gráfico de dispersão com as variáveis utilizadas e a linha de regressão encontrada pelo algoritmo.

Figura 33 - Gráficos com a linha de regressão das variáveis de TMx vs URM



Cada gráfico da Figura 33 apresenta a linha de regressão (linha na cor vermelha) gerada pelo algoritmo SVR, variando o valor da variável de custo, e o pontos referentes aos dados de *temperatura máxima* (TMx) e *umidade relativa média* (URM). Na Figura 33-a) a variável de custo possuía o valor igual a 1, na Figura 33-b) o valor de custo foi de 10, na Figura 33-c) o custo possuía o valor de 100, na Figura 33-d) o valor de custo foi de 1000, e na Figura 33-e) a variável de custo possuía o valor de 10000. O mesmo foi realizado com os dados sem valores nulos, como é apresentado na Figura 34.

Figura 34 - Gráficos com a linha de regressão das variáveis de TMx vs URM submetidos a etapa remoção de valores nulos



A Figura 34 apresenta os gráficos gerados com os dados que foram submetidos a remoção dos dados nulos. Cada gráfico possui a mesma variação da variável de custo apresentado na Figura 33. Os resultados da aplicação dos dados originais e dos dados sem valores nulos são apresentados na Tabela 5.

Tabela 5 - Dados obtidos pelas variáveis TMx vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Não	1	**7,57567**	0,25727	**0,56388**	0,75092
	10	7,57568	0,25730	0,56378	0,75085
	100	7,57573	0,25723	0,56402	0,75101
	1000	7,73844	0,25235	0,58041	0,76185
	10000	8,21740	0,28265	0,47361	0,68819
Sim	1	**7,33057**	0,21117	**0,70671**	0,84066
	10	7,33058	0,21120	0,70661	0,84060
	100	7,33068	0,21125	0,70648	0,84052
	1000	7,46432	0,21671	0,69110	0,83132
	10000	8,34187	0,24666	0,59985	0,77450

A Tabela 5 mostra os resultados obtidos com a utilização do algoritmo SVR para realizar a predição. Iniciando pelos dados originais (que não foram submetidos ao Tratamento-1) podemos verificar que, independente da função de custo, o valor do erro médio e do erro padrão (desvio médio entre os valores reais da variável dependente, neste caso a *umidade relativa média* e os valores obtidos pelo algoritmo, valor da linha de regressão) não sofrem muitas

alterações, e o coeficiente de determinação entre as variáveis não sofrem muitas alterações também, variam entre 0,473 e 0,580. No caso menos tolerante a erros (quando o valor C é igual 1) o coeficiente de determinação entre as variáveis é de aproximadamente 0,5638, logo, pode-se afirmar que 56,38% das variações da umidade relativa média podem ser explicadas pela variação da temperatura máxima. E com o caso mais tolerante a erro apenas 47,36% das variações da umidade relativa média pode ser explicada pela variação da temperatura máxima. Os valores do coeficiente de correlação afirmam isso, quanto mais próximo de -1 ou 1 for o valor do coeficiente de correlação, maior será a correlação entre as variáveis.

Analisando apenas o valor do coeficiente de determinação tem-se que, o melhor caso seria utilizar o algoritmo SVR com parâmetro C=1000, visto que o mesmo possui o coeficiente de correlação de aproximadamente 0,5804 (58,04%). Entretanto, o valor do erro médio com C=1000 é o segundo maior, sendo de aproximadamente 7,7384, desta forma tem-se que a melhor configuração para aplicar o SVR será com o parâmetro C=1, mais tolerante a erros.

Partindo para a análise dos dados submetidos ao Tratamento-1, nota-se que os valores de erro médio e erro padrão diminuíram em relação aos dados originais, e mesmo que essa diminuição seja em décimos ela é importante. No coeficiente de determinação houve um aumento significativo, estando entre 0,599 e 0,706, aproximadamente, em que com o SVR menos tolerante a erros obteve um coeficiente de determinação de aproximadamente 0,7067, logo 70,67% das variações ocorridas na umidade relativa média podem ser explicadas pelas variações da temperatura máxima. E analisando o coeficiente de correlação é possível verificar que existe uma boa correlação entre as variáveis.

Vemos que os dados originais não seriam uma utilização interessante para aplicar o SVR, visto que os dados possuem uma correlação inferior aos dados submetidos ao Tratamento-1, e desta forma, as aplicações seguintes utilizando a variável de *temperatura máxima* foram realizadas apenas com os dados submetidos ao Tratamento-1.

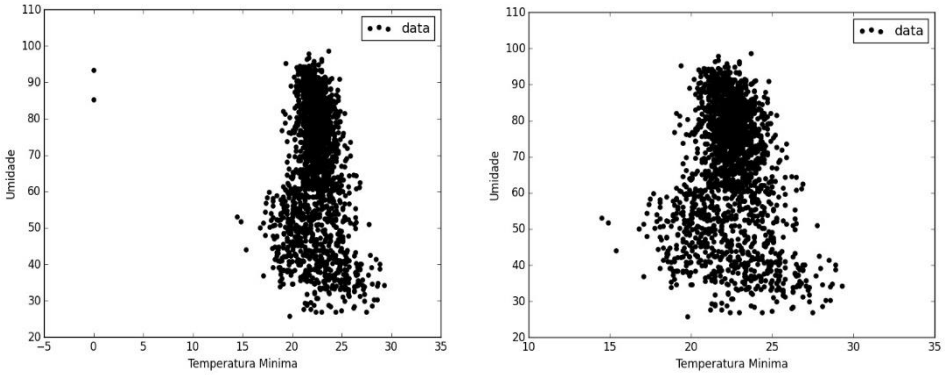
Todo o processo apresentado anteriormente foi realizado com todas as variáveis combinadas com a variável de *umidade relativa média*, e a seção seguinte (Seção 4.3.1.2) apresenta as informações encontradas com a utilização da variável de *temperatura mínima* (TMi) e a *umidade relativa média* (URM).

4.3.1.2. Temperatura Mínima vs Umidade Relativa Média

Os procedimentos realizados com as variáveis de *temperatura mínima* e *umidade relativa média*, foram os mesmos realizados na seção Temperatura Máxima vs Umidade Relativa Média (Seção 4.3.1.1), iniciando-se pela geração de gráficos de dispersão que

apresentem os dados, para iniciar verificação da correlação entre eles e de pontos com valores zerados.

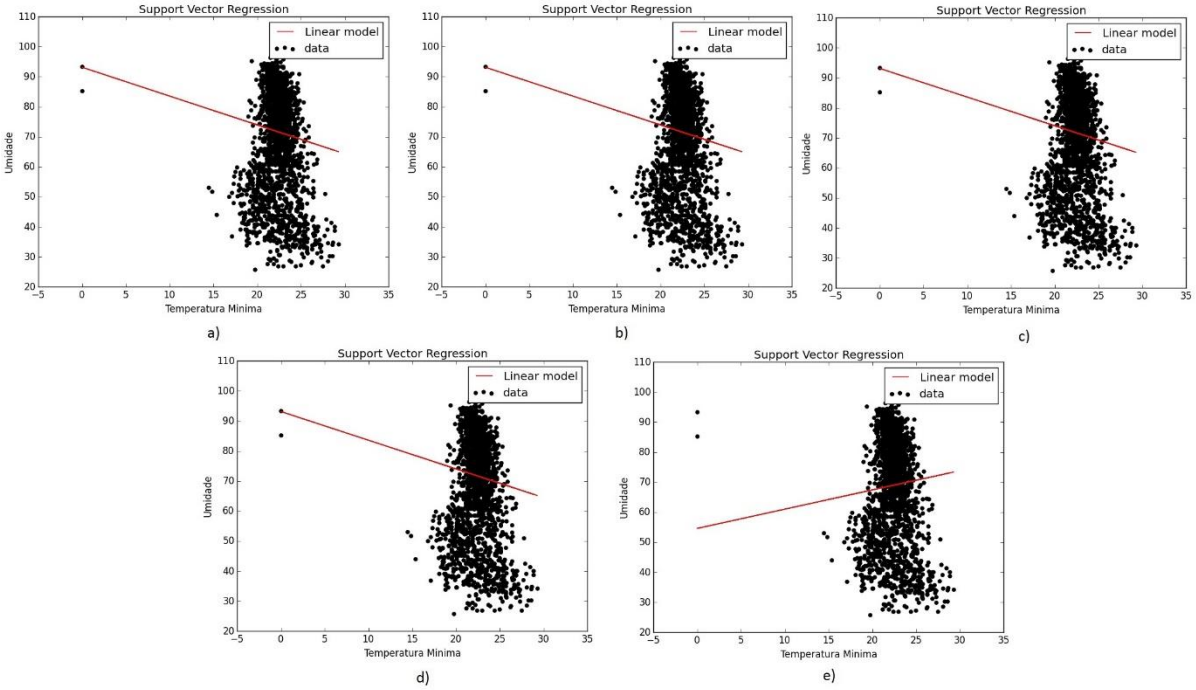
Figura 35 - Gráficos Gráfico com os valores de TMi vs URM



a) b)

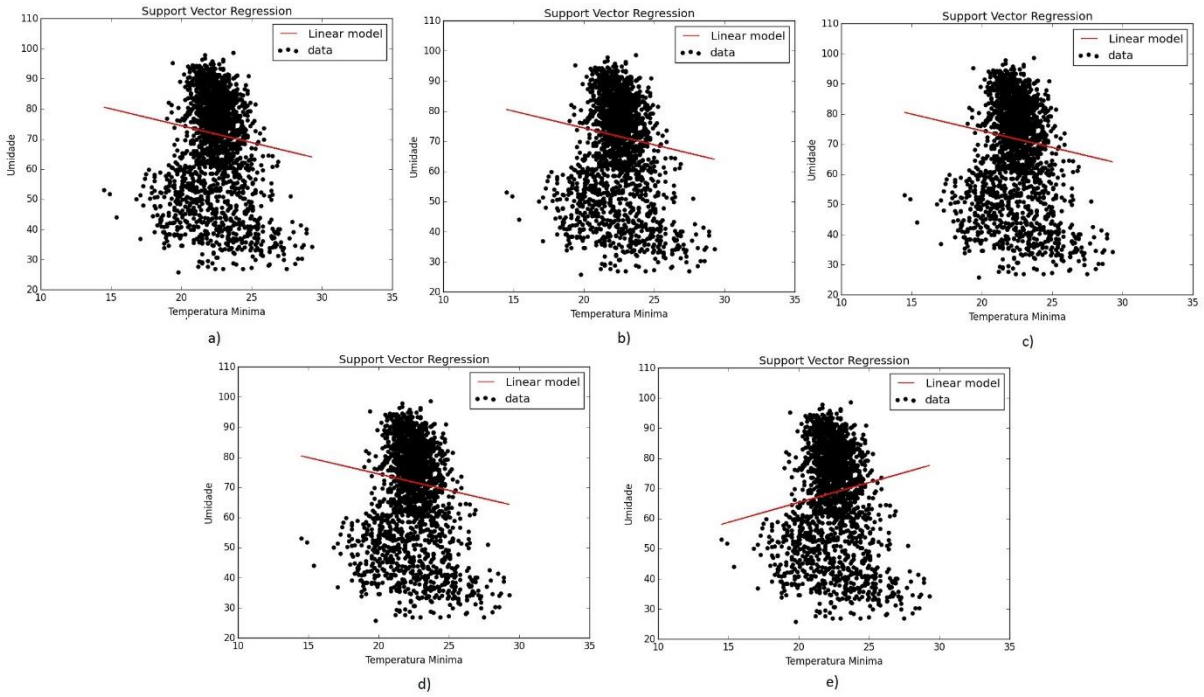
A Figura 35 apresenta dois gráficos, sendo que o gráfico da Figura 35-a) apresenta os valores originais das referidas variáveis, sem nenhum tratamento, já o gráfico apresentado na Figura 35-b) apresenta os dados após terem sidos removidos os valores com erros (através do Tratamento-1). É possível notar de forma clara, apenas olhando para os gráficos, que a correlação entre estas variáveis é bem pequena. Porém, mesmo percebendo que a correlação entre as variáveis é pequena, foi realizado o cálculo do coeficiente de correlação entre as variáveis utilizando o algoritmo SVR, e variando o seu parâmetro de custo da mesma forma variada na *temperatura máxima* e *umidade relativa média*. A partir destas aplicações foram obtidos os seguintes gráficos:

Figura 36 - Gráficos com a linha de regressão das variáveis de TMi vs URM originais



A Figura 36 apresenta cinco gráficos com a linha de regressão (linha na cor vermelha) obtidos por meio da variação do valor da variável de custo na aplicação do SVR. A Figura 36-a), Figura 36-b), Figura 36-c), Figura 36-d), e Figura 36-e) foram geradas com a variável de custo possuindo os seguintes valores, respectivamente, 1, 10, 100, 1000, 10000.

Figura 37 - Gráficos com a linha de regressão das variáveis de TMi vs URM com os dados submetidos ao Tratamento-1



A Figura 37 apresenta gráfico com a linha de regressão com dados que foram submetidos ao Tratamento-1. Cada gráfico apresentado foi gerado com uma variação da função de custo, da mesma forma apresentada na figura anterior (Figura 36), sendo esta variação de 1, 10, 100, 1000, 10000 correspondente a, respectivamente, Figura 37-a), Figura 37-b), Figura 37-c), Figura 37-d), Figura 37-e).

Aplicação do SVR com as variáveis de *temperatura mínima* e *umidade relativa média* obteve os resultados apresentados na Tabela 6.

Tabela 6 - Resultados do SVR com as variáveis de TMi vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Não	1	14,42866	0,39940	0,05106	0,22596
	10	14,42866	0,39938	0,05092	0,22565
	100	**14,42878**	0,40010	**0,05471**	0,23390
	1000	14,42871	0,39998	0,05411	0,23262
	10000	14,61833	0,39543	0,03027	0,17400
Sim	1	14,43847	0,40014	0,05459	0,23364
	10	14,43847	0,40014	0,05461	0,23370
	100	14,43838	0,40040	0,05594	0,23652
	1000	**14,43857**	0,40071	**0,05760**	0,24001
	10000	14,68344	0,39946	0,05101	0,22586

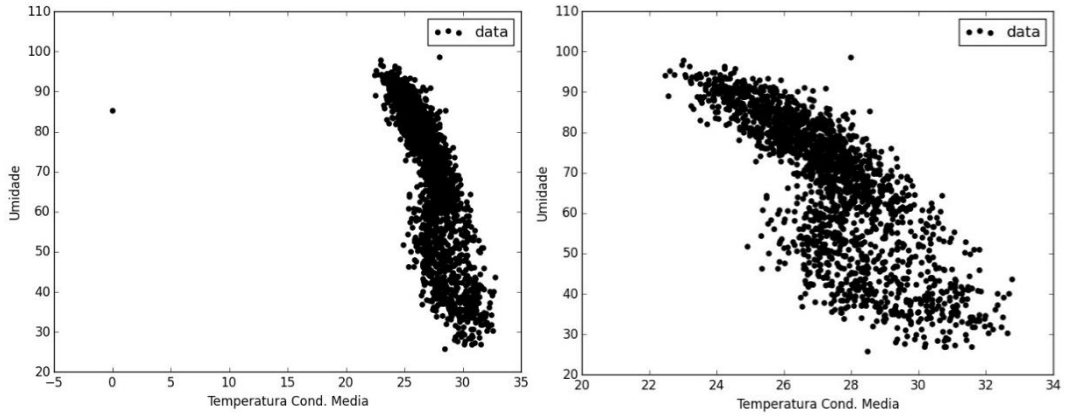
Como é apresentado nos gráficos (Figura 36 e Figura 37), os valores de *temperatura mínima* possuem pouco ou nenhuma correlação com os valores de *umidade relativa média*,

mesmo com os dados submetidos ao Tratamento-1. Dentre os valores do coeficiente de determinação dos resultados obtidos, o maior valor do coeficiente de determinação chega a aproximadamente 0,0576, logo, apenas 5,76% das variações dos valores da umidade relativa média podem ser explicadas pela variação da temperatura mínima. E o seu coeficiente de correlação está próximo de zero, com valor aproximado de 0,24. Portanto, conclui-se que existe pouca ou nenhuma relação linear entre as variáveis. Após está etapa, foi realizado a análise da utilização da *temperatura compensada média* com a *umidade relativa média*.

4.3.1.3. Temperatura Compensada Média vs Umidade Relativa Média

Para realizar a verificação do grau de correlação entre as variáveis de *temperatura compensada média* (TCM) e a *variável de umidade relativa média* (URM), e verificar, também, se esta variável é interessante para a aplicação. Para isso, foram gerados gráficos e realizados cálculos que proporcionaram a apresentação e análise dos resultados da aplicação destas variáveis. Os primeiros gráficos gerados são apresentados na Figura 38.

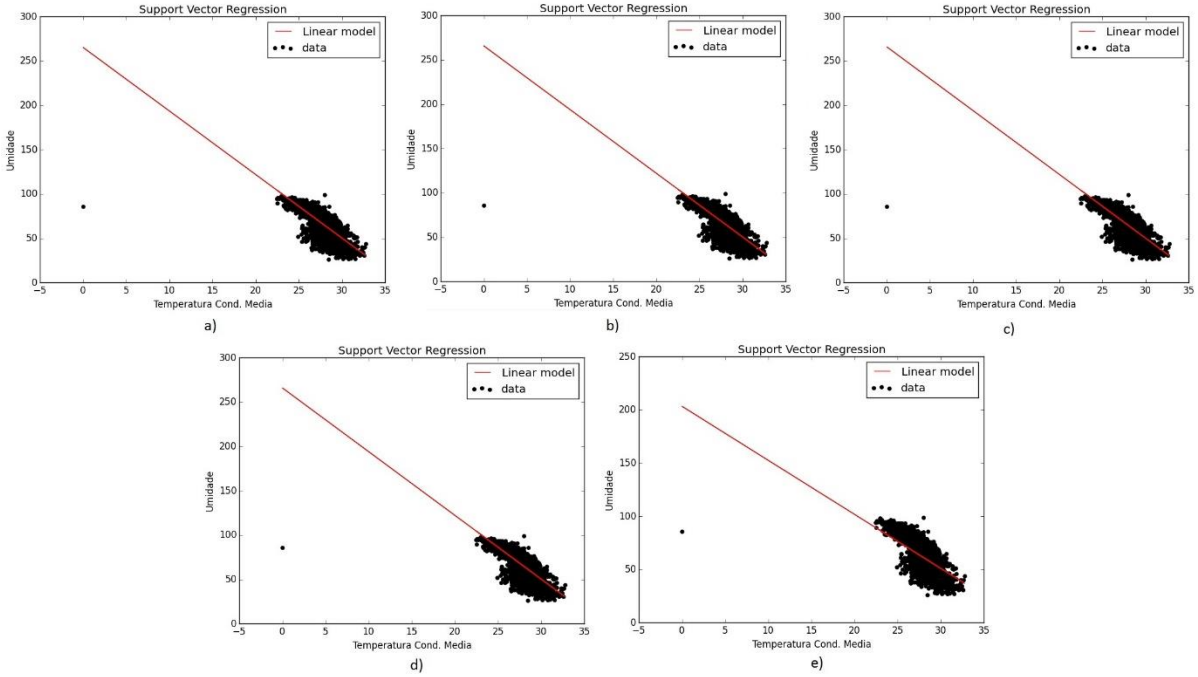
Figura 38 - Gráficos de TCM vs URM



a) b)

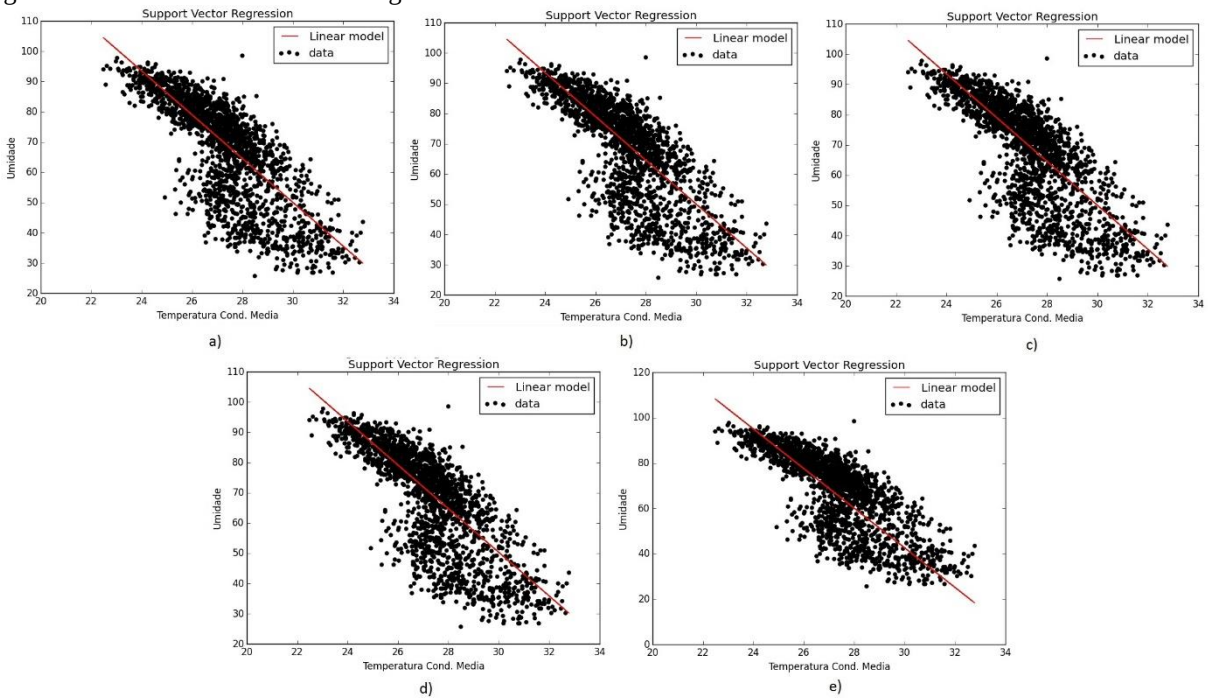
A Figura 38 apresenta os gráficos gerados a partir dos valores das variáveis *temperatura compensada média* e *umidade relativa média*, sendo que na Figura 38-a) o gráfico foi gerado com os dados originais (sem realizar nenhum tratamento neles), já na Figura 38-b) o gráfico foi gerado a partir dos dados que foram submetidos ao Tratamento-1. Nota-se que as variáveis não possuem uma grande correlação linear, e somente por meio do cálculo do coeficiente de correlação, é possível saber qual é o nível exato de correlação entre as variáveis. Mas antes de calcular o coeficiente de correlação, foi gerado a linha de regressão obtida pelo SVR em cada uma das bases de dados (base de dados originais e base de dados submetidos ao Tratamento-1). Esta linha é apresentada na Figura 39 e Figura 40.

Figura 39 - Gráficos com a linha de regressão das variáveis de TCM vs URM originais



Os gráficos apresentados na Figura 39 mostram a linha de regressão gerada pelo algoritmo SVR, onde foi variado o parâmetro C da função, gerando cada gráfico para se realizar a verificação de linha em cada um deles. O valor do parâmetro C possui o valor 1 na Figura 39-a), 10 na Figura 39-b), 100 na Figura 39-c), 1000 na Figura 39-d), e 10000 na Figura 39-e). O mesmo procedimento foi realizado com os dados que foram submetidos ao Tratamento-1, variando o parâmetro C da mesma forma, em 1 (Figura 40-a), 10 (Figura 40-b), 100 (Figura 40-c), 1000 (Figura 40-d), 10000 (Figura 40-e).

Figura 40 - Gráficos com a linha de regressão das variáveis de TCM vs URM submetidas ao Tratamento-1



Em ambas figuras (Figura 39 e Figura 40) é possível notar que a linha de regressão não se ajusta bem aos pontos, isso reforça a afirmação feita anteriormente que o nível de correlação entre estas variáveis não é grande, e a partir dos resultados obtidos está afirmação foi reforçada. A Tabela 7 apresenta os resultados encontrados.

Tabela 7 - Resultados do SVR com as variáveis de TCM vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Não	1	8,43074	0,26520	0,53660	0,73252
	10	**8,43063**	0,26518	**0,53667**	0,73257
	100	8,43058	0,26519	0,53661	0,73254
	1000	8,43137	0,26590	0,53415	0,73085
	10000	10,51998	0,27472	0,50273	0,70903
Sim	1	8,34294	0,24887	0,59210	0,76948
	10	8,34297	0,24882	0,59224	0,76957
	100	**8,34299**	0,24882	**0,59225**	0,76958
	1000	8,34405	0,24968	0,58943	0,76774
	10000	8,99097	0,25168	0,58282	0,76343

Verifica-se, nos resultados apresentados na Tabela 7, que as variáveis possuem uma pequena correlação linear, apresentando o coeficiente de determinação, com os dados originais, entre 0,5027 e 0,5366, ou seja apenas 50,27% a 53,66% da variação dos valores da *umidade relativa média* podem ser explicadas pela variação da *temperatura compensada média*. Possuem uma correlação entre 0,709 e 0,732, e desta forma, pode-se afirmar que as variáveis possuem uma correlação, porém, a correlação não é uma correlação muito forte.

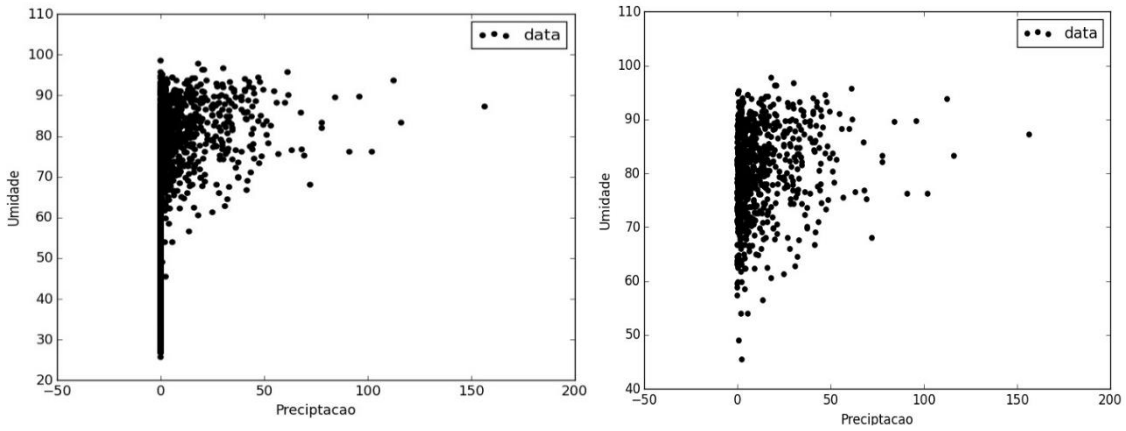
Com os dados submetidos ao Tratamento-1 os resultados melhoram um pouco. Variando o parâmetro C do algoritmo, as variáveis obtiveram o coeficiente de determinação aproximadamente entre 0,5828 e 0,5922. Portanto, 58,28% a 59,22% das variações da variável de *umidade relativa média* podem ser explicadas pelas variações dos valores da *temperatura compensada média*. Possuindo uma correlação entre 0,7634 e 0,7695, e da mesma forma dos dados originais, as variáveis possuem uma correlação, porem a correlação não é muito forte.

Terminado a análise entre os valores de *temperatura compensada média* e *umidade relativa média*, foi realizado a análise dos valores de *precipitação* (PREP) e *umidade relativa média* (URM).

4.3.1.4. Precipitação vs Umidade Relativa Média

A *precipitação* é relacionada ao nível de chuva, neve ou granizo ou qualquer outro fenômeno que provoque queda de água do céu. Nesta etapa, foram gerados dois gráficos com os pontos referentes as variáveis (da mesma forma realizada nas etapas anteriores), sendo o primeiro gráfico feito com os dados iniciais e o segundo gráfico com dados sem valores que sejam iguais a zero.

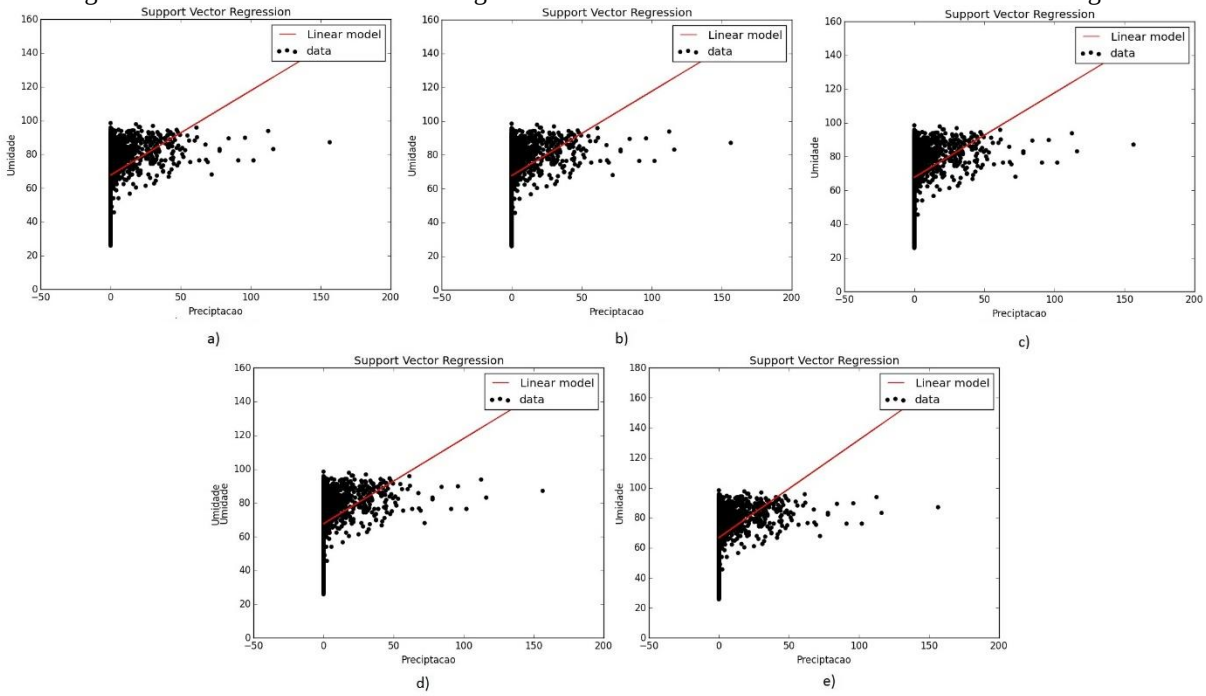
Figura 41 - Gráficos de PREP vs URM



a) b)

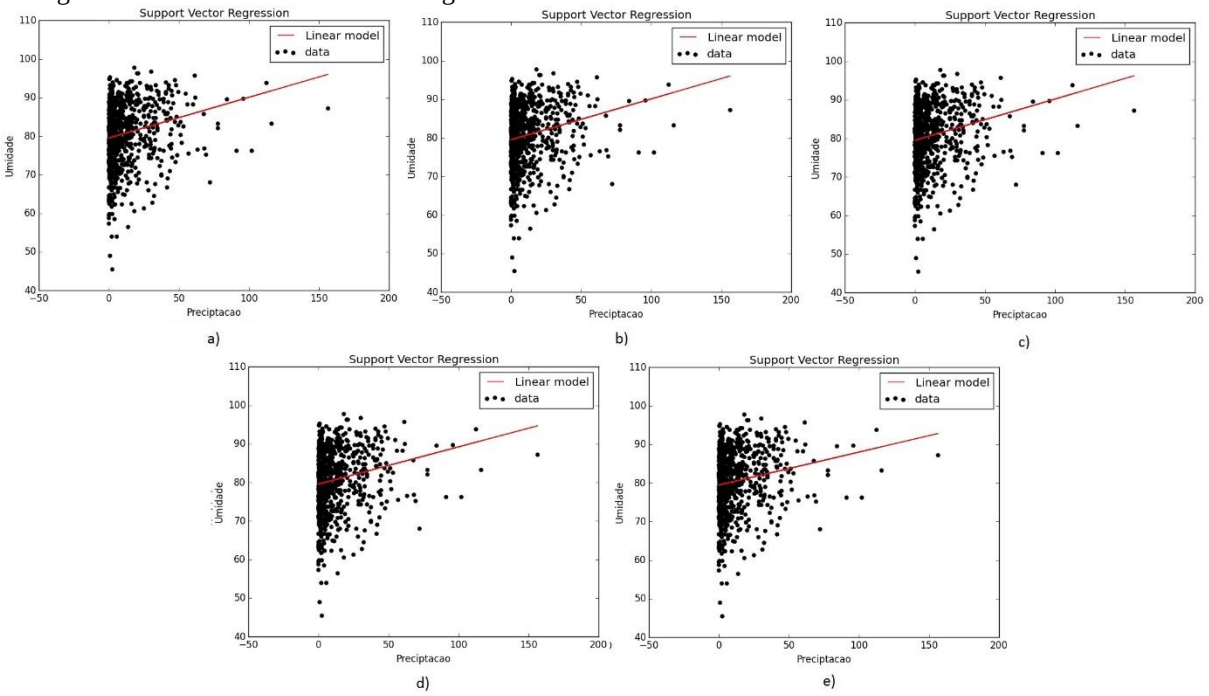
A Figura 41 apresenta dois gráficos com as variáveis *precipitação* e *umidade relativa média*, em que a Figura 41-a) apresenta o gráfico gerado a partir dos valores originais das variáveis, e a Figura 41-b) apresenta o gráfico com os dados submetidos ao Tratamento-1, que consiste na remoção dos valores da variável independente iguais a zero. A partir desses dados nota-se que a precipitação não possui uma grande correlação com os valores da *umidade relativa média*. Buscando obter o nível de correlação entre as variáveis, os dados foram aplicados ao algoritmo SVR, da mesma forma aplicada nas variáveis anteriores, e a partir disso foi gerado os gráficos para cada aplicação, apresentando em cada um deles a linha de regressão.

Figura 42 – Gráficos com a linha de regressão das variáveis de PREP vs URM com os dados originais



É apresentado na Figura 42 os gráficos gerados a partir dos dados originais de *precipitação* e *umidade relativa média*, em que cada gráfico possui a linha de regressão (linha na cor vermelha) obtida através da aplicação dos dados ao algoritmo SVR, variando o parâmetro de C em cada caso. O valor do parâmetro de custo foi variado em 1 (Figura 42-a), 10 (Figura 42-b), 100 (Figura 42-c), 1000 (Figura 42-d), 10000 (Figura 42-e).

Figura 43 - Gráficos com a linha de regressão das variáveis de PREP vs URM submetidos ao Tratamento-1



A Figura 43 apresenta os gráficos com a linha de regressão obtidos com a variação do valor do parâmetro C, com os dados submetidos ao Tratamento-1. Em que a Figura 43-a) o valor do parâmetro de C foi igual a 1, Figura 43-b) o C=10, Figura 43-c) o parâmetro C possuiu o valor 100, Figura 43-d) o valor de C foi igual a 1000, e a Figura 43-e) a linha de regressão foi gerada com o valor do parâmetro C igual a 10000.

Os resultados obtidos da aplicação dos dados de *precipitação* e *umidade relativa média* obtidos ao algoritmo SVR, foram obtidos os valores apresentados na Tabela 8.

Tabela 8 - Resultados do SVR com as variáveis de PREP vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Não	1	**13,47795**	0,37032	**0,09643**	0,31054
	10	13,47795	0,37032	0,09643	0,31053
	100	13,47795	0,37033	0,09637	0,31044
	1000	13,47797	0,37041	0,09597	0,30979
	10000	13,54766	0,37249	0,08581	0,29294
Sim	1	**6,205814**	0,28190	**0,02521**	0,15880
	10	6,205810	0,28190	0,02516	0,15864
	100	6,205843	0,28195	0,02485	0,15764
	1000	6,206610	0,28157	0,02748	0,16578
	10000	6,212573	0,28111	0,03068	0,17515

A partir da Tabela 8 podemos afirmar que as variáveis possuem pouca ou nenhuma correlação. Este caso foi único em que os dados originais apresentaram todos os valores superiores aos valores da aplicação com os dados que foram submetidos ao Tratamento-1.

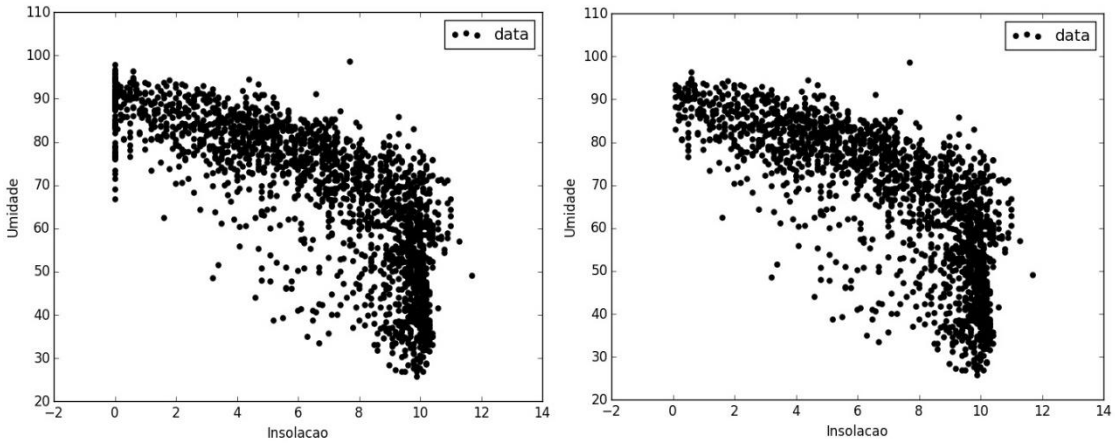
Sendo na aplicação com os dados originais, variando os valores do parâmetro C, o coeficiente de determinação esteve entre 0,085 e 0,096, e desta forma apenas 8,5% a 9,6% das variações dos valores da *umidade relativa média* podem ser explicados pelas variações dos valores da *precipitação*. Enquanto com os dados submetidos ao Tratamento-1 o coeficiente de determinação esteve entre 0,024 e 0,036, então apenas de 2,4% a 3,6% das variações dos valores da *umidade relativa média* podem ser explicadas pelas variações dos valores de *precipitação*.

O valor do coeficiente de correlação para os dados originais esteve entre 0,29 e 0,31, já com os dados submetidos ao Tratamento-1 os valores estiveram entre 0,15 e 0,17. Em ambos os casos o grau de correlação entre as variáveis é considera baixa ou nula.

4.3.1.5. Insolação vs Umidade Relativa Média

A insolação, como o nome já diz, corresponde a incidência do sol em determinado momento. Buscando visualizar como se relaciona os valores de *insolação* (INS) com os valores de *umidade relativa média* (URM) foram gerados dois gráficos, sendo o primeiro com os dados originais e o segundo com os dados submetidos ao Tratamento-1.

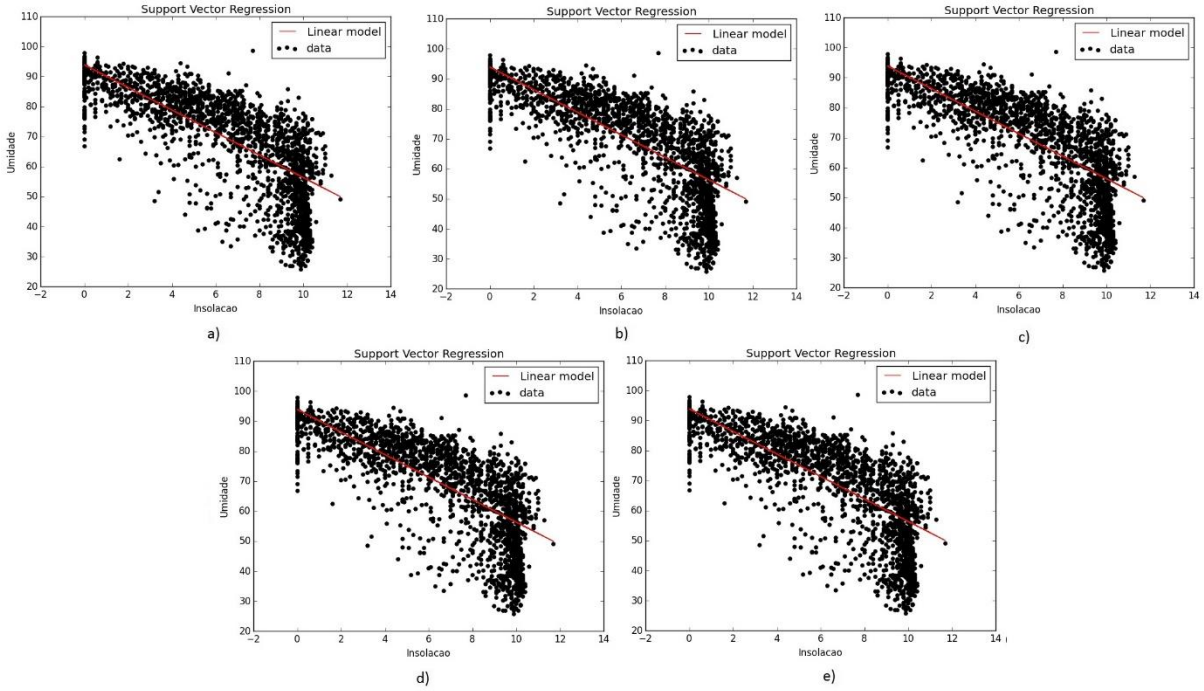
Figura 44 - Gráficos de INS vs URM



a) b)

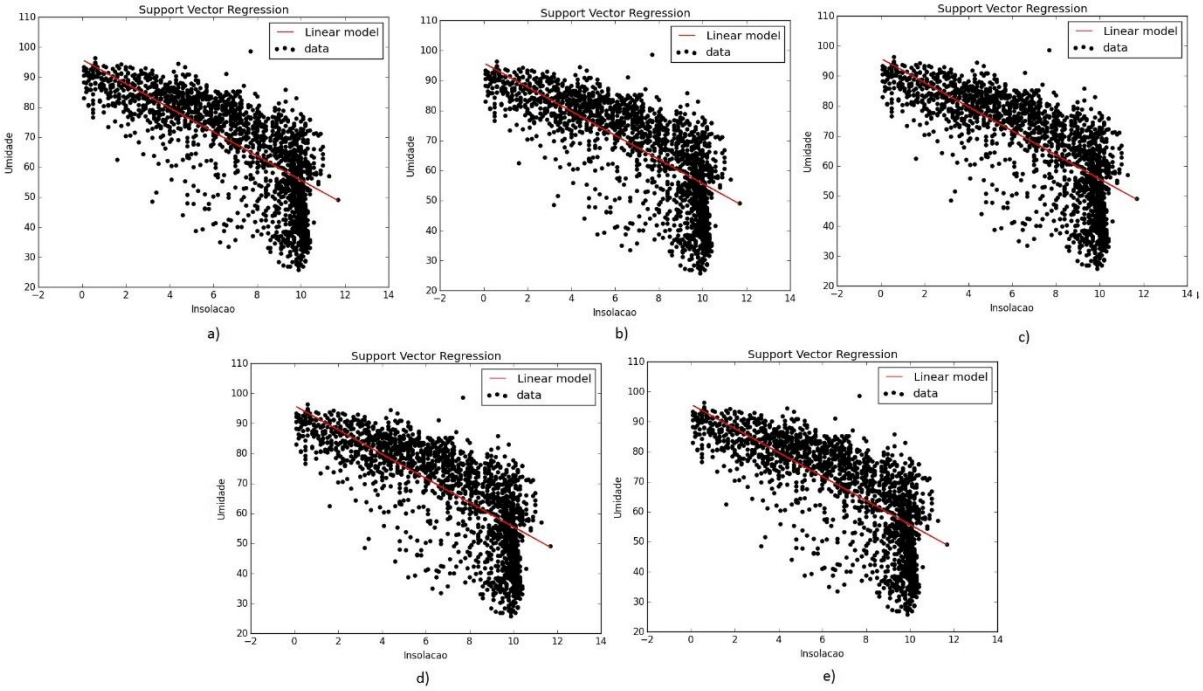
A Figura 44 apresenta os gráficos com os valores de *insolação* e *umidade relativa média*, sendo que na Figura 44-a) os dados utilizados foram os dados originais, sem nenhum tratamento, e na Figura 44-b) o gráfico foi gerado a partir dos dados submetidos ao Tratamento-1. Pode-se notar que as variáveis não possuem uma correlação linear grande, mas para que se possa afirmar qual o nível ou grau de correlação entre as variáveis, foi realizado a aplicação dos dados no algoritmo SVR, da mesma forma realizada com as variáveis anteriores gerando assim os gráficos apresentados nas Figura 45 e Figura 46.

Figura 45 – Gráficos com a linha de regressão das variáveis de INS vs URM originais



Na Figura 45 os gráficos apresentados foram obtidos a partir dos dados originais, e cada um dos gráficos apresentam uma linha de regressão gerada pelo algoritmo SVR, variando apenas o valor do parâmetro de tolerância a erros, o parâmetro C. A Figura 45-a) apresenta a linha de regressão obtida com o parâmetro de custo igual a 1, a Figura 45-b) mostra a linha de regressão gerado com o parâmetro C=10, na Figura 45-c) o valor da variável C é de 100, e na Figura 45-d) e Figura 45-e) mostram a linha de regressão gerada com o valor da variável C sendo, respectivamente, 1000 e 10000.

Figura 46 - Gráficos com a linha de regressão das variáveis INS vs URM submetidos ao Tratamento-1



A Figura 46 apresenta gráficos com a linha de regressão gerada a partir dos dados submetidos ao Tratamento-1, com a variação do valor do parâmetro C. Em que na Figura 46-a), Figura 46-b), Figura 46-c), Figura 46-d), e Figura 46-e) a linha de regressão apresentada foram calculadas com o valor do parâmetro C sendo, respectivamente, 1, 10, 100, 1000,10000. Os resultados da aplicação dos dados originais e dos dados submetidos ao Tratamento-1 são apresentados na Tabela 9.

Tabela 9 - Resultados do SVR com as variáveis de INS vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Não	1	**9,05706**	0,25731	**0,56374**	0,75083
	10	9,05706	0,25732	0,56373	0,75082
	100	9,05705	0,25733	0,56368	0,75078
	1000	9,05705	0,25733	0,56368	0,75078
	10000	9,05721	0,25747	0,56321	0,75047
Sim	1	9,09741	0,26332	0,55364	0,74407
	10	9,09741	0,26333	0,55363	0,74406
	100	9,09740	0,26336	0,55353	0,74399
	1000	9,09752	0,26306	0,55452	0,74466
	10000	9,09747	0,26346	0,55317	0,74375

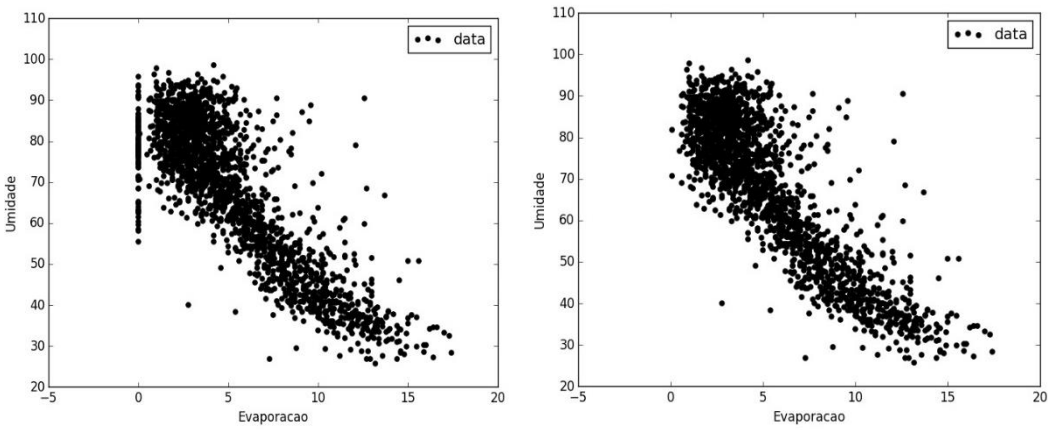
A partir dos resultados apresentados na Tabela 9**,** verificou-se que o coeficiente de determinação, o erro padrão e o coeficiente de correlação, entre os dados originais e os dados submetidos ao Tratamento-1, não obtiveram muita variação, com uma diferença de no máximo 0,01. Com o coeficiente de determinação estando entre 0,55 e 0,56, aproximadamente, vemos

que apenas 55% a 56% das variações dos valores de *umidade relativa média* podem ser explicadas pelas variações dos valores de *insolação*. E com o coeficiente de correlação variando entre 0,74 e 0,75, é possível concluir que as variáveis possuem uma correlação, porem ela não é muito forte.

4.3.1.6. Evaporação vs Umidade Relativa Média

Foram realizados os mesmos procedimentos feitos nas variáveis apresentadas anteriormente, iniciando-se pela apresentação das variáveis, neste caso, a *evaporação* (EVP) e *umidade relativa média* (URM), em dois gráficos, sendo o primeiro com os dados originais e o segundo para os dados submetidos ao Tratamento-1.

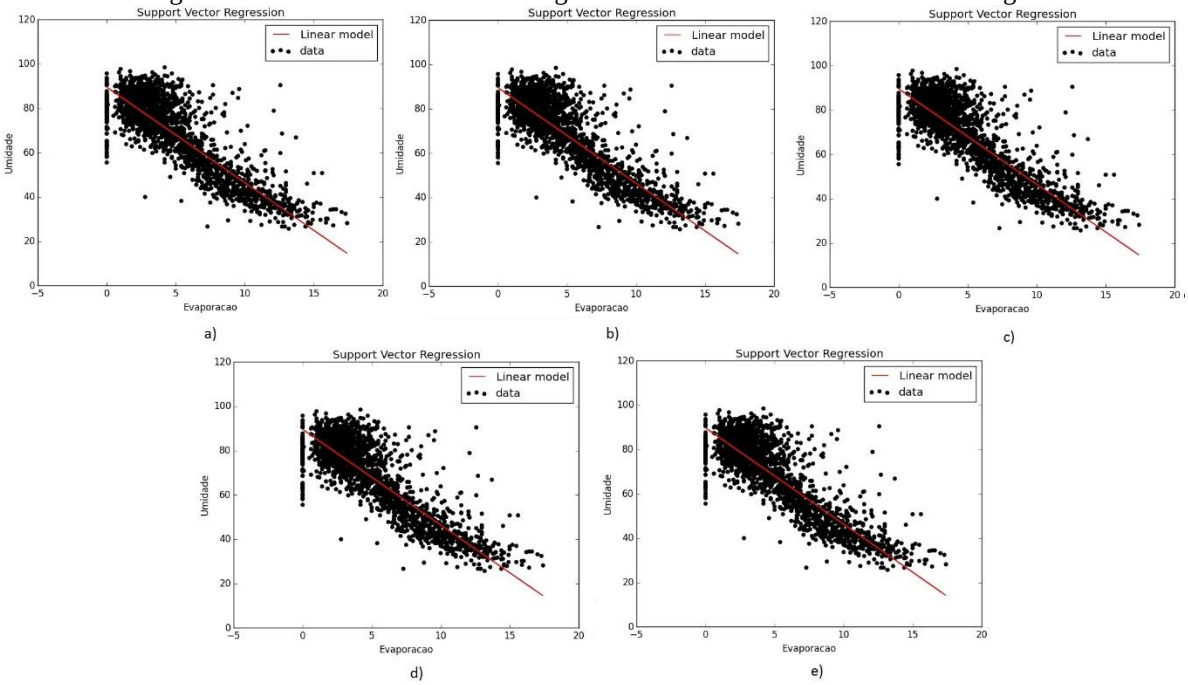
Figura 47 - Gráficos de EVP vs URM



a) b)

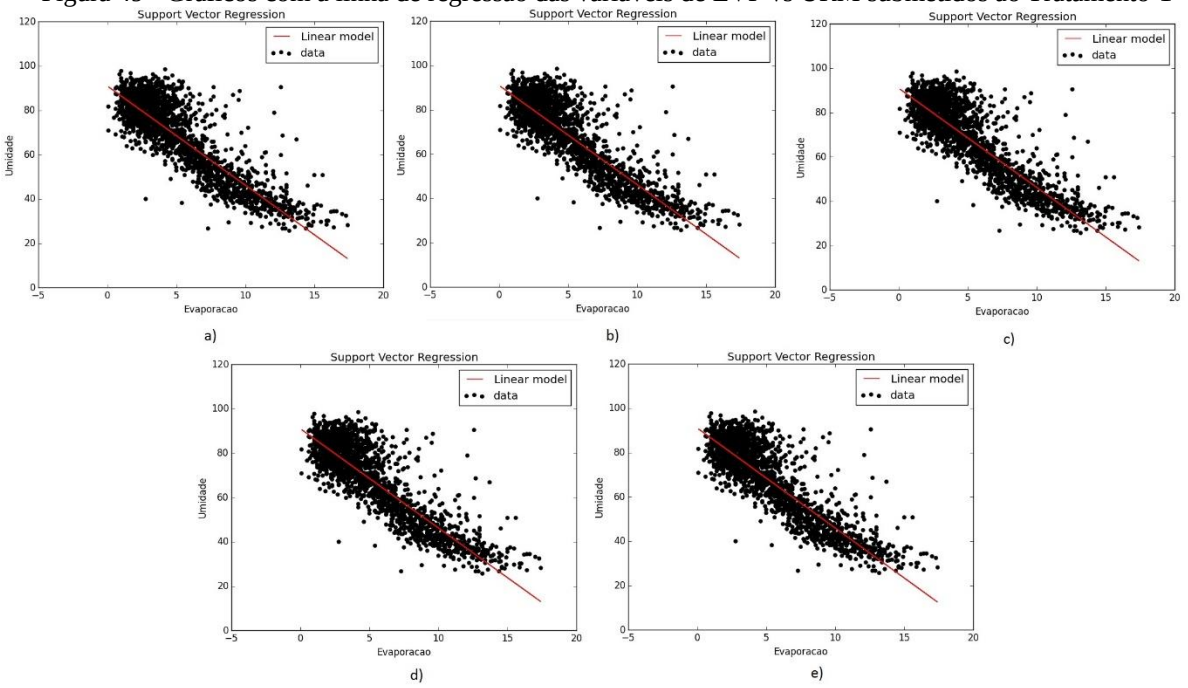
A Figura 47-a) apresenta o gráfico gerado a partir dos dados originais e a Figura 47-b) apresenta o gráfico com os dados submetidos ao Tratamento-1. Nota-se que em ambos os gráficos que os dados possuem uma correlação, mas não é possível determinar qual o grau de correlação entre elas. Para isto, os dados foram aplicados ao algoritmo SVR, e seus resultados foram apresentados e analisados. Foi realizado as cinco variações do valor do parâmetro C em cada conjunto de dados, e o resultado desta aplicação é apresentado através de gráficos, contendo a linha de regressão obtida pelo algoritmo SVR para cada variação de C.

Figura 48 - Gráficos com a linha de regressão das variáveis de EVP vs URM originais



A Figura 48-a), Figura 48-b), Figura 48-c), Figura 48-d), Figura 48-e) apresenta a linha de regressão gerada com o valor de C sendo, respectivamente 1, 10, 100, 1000, 10000. Os dados utilizados aqui foram os dados originais.

Figura 49 - Gráficos com a linha de regressão das variáveis de EVP vs URM submetidos ao Tratamento-1



Na Figura 49 os gráficos apresentados foram gerados com os dados submetidos ao Tratamento-1. Cada gráfico possui a linha de regressão gerada pelo algoritmo SVR, se diferenciando pelo valor do parâmetro C, que é referente a função de C. Na Figura 49-a) C=1,

na Figura 49-b) C=10, Figura 49-c) C=100, Figura 49-d) C=1000, Figura 49-e) C=10000. Os demais resultados obtidos com esta aplicação são apresentados na Tabela 10.

Tabela 10 - Resultados do SVR com as variáveis de EVP vs URM.

Dados com Tratamento-1?	*C*	***Erro Médio***	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Não	1	**7,13475**	0,21126	**0,70593**	0,84020
	10	7,13474	0,21127	0,70591	0,84018
	100	7,13476	0,21125	0,70594	0,84020
	1000	7,13471	0,21131	0,70578	0,84010
	10000	7,13562	0,21149	0,70529	0,83981
Sim	1	6,84775	0,20678	0,73601	0,85791
	10	6,84775	0,20678	0,73601	0,85791
	100	6,84776	0,20678	0,73601	0,85791
	1000	6,84778	0,20678	0,73600	0,85790
	10000	6,84920	0,20720	0,73493	0,85728

Com os dados originais o coeficiente de determinação é aproximadamente de 0,70, e desta forma aproximadamente 70% das variações dos valores de *umidade relativa média* podem ser explicadas pela variação dos valores de *evaporação*. O coeficiente de correlação varia entre 0,83 (no caso mais tolerante a erros, em que o parâmetro C recebe o valor 10000) e 0,84 (no caso menos tolerante a erros, em que o parâmetro C recebe o valor 1).

Nos dados submetidos ao Tratamento-1, os valores do coeficiente de determinação foram de aproximadamente 0,73, com 73% das variações da variável dependente (*umidade relativa média*) podendo ser explicada pela variação da variável independente (*evaporação*). Obtendo uma correlação aproximada de 0,85, possibilitando concluir que as variáveis possuem uma correlação forte.

4.3.2. Aplicação do SVR com regressão linear múltipla

A partir dos resultados obtidos, nota-se que apenas uma variável independente não é suficiente para prever o valor da *umidade relativa média*. Desta forma, parte-se para aplicação dos dados utilizando o conceito de regressão linear múltipla (a regressão linear múltipla, permite-se usar mais de uma variável independente) (RLM). Porém, este método não foi aplicado a todas a variáveis, como foi realizado na regressão linear simples, será realizado apenas com as variáveis de *temperatura máxima* (TMx), *temperatura compensada média* (TCM), *insolação* (INS) e *evaporação* (EVP). A escolha de somente estas variáveis é devido que na regressão linear simples elas foram as que obtiveram o maior coeficiente de determinação em relação a variável dependente (*umidade relativa média*), e o dados utilizados

serão submetidos ao Tratamento-1, visto que na maioria dos casos, os dados submetidos ao Tratamento-1 obtiveram resultados superiores aos dados originais.

A aplicação destas variáveis na regressão linear foi dívida em três partes: RLM com duas variáveis independentes; RLM com três variáveis independentes; RLM com quatro variáveis independentes. Realizando todas a combinações possíveis entre as variáveis.

4.3.2.1. Aplicação com duas variáveis independentes

A aplicação do SVR em duas variáveis foi realizada com todas as combinações possíveis entre as quatro variáveis selecionadas. As combinações das variáveis independentes possíveis são: *temperatura máxima* e *temperatura média compensada*, *temperatura máxima* e *insolação*, *temperatura máxima e evaporação*, *temperatura compensada média* e *insolação*, *temperatura compensada média* e *evaporação*, *insolação* e *evaporação*. As aplicações são apresentadas nas seções seguintes (Seção 4.3.2.1.1, Seção 4.3.2.1.2, Seção 4.3.2.1.3, Seção 4.3.2.1.4, Seção 4.3.2.1.5 e Seção 4.3.2.1.6), iniciando pela aplicação de *temperatura máxima* e *temperatura compensada média* e *umidade relativa média*.

4.3.2.1.1. Temperatura Máxima e Temperatura Compensada Média vs Umidade Relativa Média

A utilização das variáveis de *temperatura máxima* (TMx) e *temperatura compensada média* (TCM) como sendo variáveis independentes aplicadas ao algoritmo SVR gerou os resultados que são apresentados na Tabela 11.

Tabela 11 - TMx e TCM vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	7,28519	0,21027	0,70933	0,84222
	10	7,28517	0,21030	0,70925	0,84217
	100	**7,28516**	0,21025	**0,70938**	0,84225
	1000	7,28880	0,20974	0,71078	0,84308
	10000	7,44751	0,20802	0,71553	0,84589

Já com a primeira utilização de mais de uma variável independente é possível notar uma melhora significativa dos resultados. A Tabela 11 apresenta os valores obtidos com a aplicação das variáveis de *temperatura máxima* e *temperatura compensada média*. Nota-se que há uma diminuição do valor do erro médio, uma melhora no coeficiente de determinação e correlação. Esta melhora é pequena se for comparada com a aplicação da *temperatura máxima* apresentada na regressão linear simples, porém é uma melhora bastante significativa se comparada a

aplicação da *temperatura compensada média*. Tem-se que o coeficiente de correlação, no caso menos tolerante a erros um valor de aproximadamente de 0,7093, o que significa que 70,93% das variações da umidade relativa média pode ser explicada pelas variações da *temperatura máxima* e da *temperatura compensada média*, com um erro médio de 7,28, para mais ou para menos, e com uma correlação considerada forte.

Os resultados obtidos pela aplicação com mais tolerância a erros, apresenta o melhor coeficiente de determinação e correlação, porém o valor de erro é maior, e como a diferença entre estes casos não é grande, o caso considerado como a melhor aplicação neste teste será o que possui a menor tolerância a erros, em que o valor de C é igual a 1. Seguido desta aplicação, foi realizado a aplicação da *temperatura máxima* e *insolação* como variáveis independentes, e os resultados desta aplicação é apresentado na seção a seguir (Seção 4.3.2.1.2).

4.3.2.1.2. Temperatura Máxima e Insolação vs Umidade Relativa Média

Nesta seção é apresentado os resultados da aplicação das variáveis *temperatura máxima* (TMx) e *insolação* (INS) como variáveis independentes. Estes resultados são disponibilizados na Tabela 12. A análise feita aqui será a mesma realizada na seção anterior (Seção 4.3.2.1.1), possuindo o objetivo de verificar qual a melhor aplicação.

Tabela 12 - TMx e INS vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	7,16576	0,20827	0,72129	0,84929
	10	7,16575	0,20830	0,72122	0,84924
	100	**7,16576**	0,20811	**0,72171**	0,84953
	1000	7,16852	0,20843	0,72085	0,84903
	10000	10,64877	0,28728	0,46972	0,68536

Os resultados encontrados até a quarta aplicação (em que o parâmetro C possui o valor de 1000) não possui uma variação notável, estando com o erro médio entre 7,1657 e 7,1682, e o coeficiente de determinação e coeficiente de correlação estão entre, respectivamente, 0,7208 a 0,7217, e 0,8490 a 0,8495. E na aplicação com mais tolerância a erros, o erro médio está em 10,6487, coeficiente de determinação em 0,4697 e o coeficiente de correlação em 0,6853. Valores bem diferentes das outras aplicações, o que fazem este ser a pior aplicação do modelo. Neste caso, foi determinado como a melhor aplicação a que possuiu o valor do parâmetro C igual a 1, devido que este possui a menor tolerância a erros. A aplicação seguinte foi realizada com a variável de *temperatura máxima* e *evaporação* como sendo as variáveis independentes.

4.3.2.1.3. Temperatura Máxima e Evaporação vs Umidade Relativa Média

A partir da aplicação das variáveis de *evaporação* (EVP) e *temperatura máxima* (TMx) como sendo variáveis independentes, foram obtidos os resultados apresentados na Tabela 13.

Tabela 13 - TMx e EVP vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	**4,90042**	0,14890	**0,86337**	0,92918
	10	4,90045	0,14891	0,86336	0,92917
	100	4,90059	0,14894	0,86330	0,92914
	1000	4,90047	0,14890	0,86338	0,92918
	10000	6,94302	0,19353	0,76919	0,87703

Com os resultados apresentados na Tabela 13 desta aplicação tem-se que o valor do erro médio está em aproximadamente 4,9004, e o coeficiente de determinação em aproximadamente 0,8633 (86,33%) e o coeficiente de correlação em 0,9291, para o parâmetro C possuindo o valor igual a um. A aplicação com a menor tolerância a erros apresenta os melhores resultados, porem a diferença entre os outros resultados está em milésimos, exceto para a aplicação mais tolerante a erros, (onde o C possui o maior valor) com uma variação de duas unidades no erro médio. Com isso a melhor aplicação neste modelo é a menos tolerante a erros, em que o valor de C é igual a um, e está aplicação é a aplicação que apresenta, até o momento, os melhores resultados.

Seguido desta, foi utilizado as variáveis de *temperatura compensada média* e *insolação* com variáveis independentes e os resultados obtidos são apresentados na Seção 4.3.2.1.4.

4.3.2.1.4. Temperatura Compensada Média e Insolação vs Umidade Relativa Média

Aplicação das variáveis de *temperatura compensada média* (TCM) e a *insolação* (INS), após serem aplicadas ao algoritmo SVR, produziram os resultados apresentados na Tabela 14.

Tabela 14 - TCM e INS vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	7,69261	0,21893	0,69159	0,83161
	10	7,69259	0,21893	0,69157	0,83160
	100	**7,69256**	0,21888	**0,69172**	0,83170
	1000	7,69309	0,21912	0,69105	0,83129
	10000	7,70737	0,21689	0,69729	0,83504

Dentre as aplicações apresentadas, com duas variáveis independentes, está é a que possui os piores resultados. O erro médio varia, aproximadamente, entre 7,69 e 7,70, o coeficiente de determinação com o valor de aproximadamente de 0,69, o que significa que

apenas 69% das variações da *umidade relativa média* podem ser explicadas pelas variações da *temperatura compensada média* e as variações da *insolação*. O coeficiente de correlação está em 0.83, no melhor caso, e mesmo com esta correlação sendo forte, esta aplicação é a que possui os piores resultados, porém é melhor que a maioria das aplicações realizadas com apenas uma variável independente. A próxima aplicação foi realizada com as variáveis de *temperatura compensada média* e *evaporação*, e os resultados são apresentados logo a seguir.

4.3.2.1.5. Temperatura Compensada Média e Evaporação vs Umidade Relativa Média

A aplicação da *temperatura compensada média* (TCM) e *evaporação* (EVP), como variáveis independentes, ao algoritmo SVR, resultou nos valores apresentados na Tabela 15.

Tabela 15 - TCM e EVP vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	**5,86258**	0,17407	**0,81292**	0,90162
	10	5,86256	0,17407	0,81291	0,90161
	100	5,86256	0,17407	0,81291	0,90161
	1000	5,86280	0,17413	0,81279	0,90155
	10000	6,12078	0,17873	0,80277	0,89597

Esta aplicação resultou nos valores apresentados na Tabela 15, em que o erro médio está em, aproximadamente, 5,86, variando milésimos na maioria dos casos. O coeficiente de determinação possui o valor de 0,81, variando também em milésimos na maioria dos casos, e consequentemente o coeficiente de correlação em 0,90, um valor forte e muito bom para a aplicação. Esta aplicação está entre as aplicações com os melhores resultados. Em seguida temos a última aplicação realizada com apenas duas variáveis independentes, sendo elas a *evaporação* e a *insolação*.

4.3.2.1.6. Insolação e Evaporação vs Umidade Relativa Média

Foram aplicados ao algoritmo SVR as variáveis de *insolação* (INS) e *evaporação* (EVP) como sendo variáveis independentes, e os resultados obtidos desta aplicação são apresentados na Tabela 16.

Tabela 16 - INS e EVP vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	**4,87127**	0,15007	**0,86407**	0,92955
	10	4,87128	0,15007	0,86406	0,92955
	100	4,87129	0,15008	0,86405	0,92954
	1000	4,87142	0,15010	0,86400	0,92952
	10000	4,87160	0,14998	0,86422	0,92963

Esta aplicação gerou os dados apresentados na Tabela 16, em que o valor de erro médio está 4,871, o valor do coeficiente de determinação está em 0,864 (86,40%), e o coeficiente de correlação em 0,929, em que em todos os casos a uma variação em milionésimos em cada aplicação. Estas variáveis possuem uma correlação com a variável *umidade relativa média* forte, e a sua aplicação proporcionou resultados muito bons, tanto para o valor do erro médio quanto para o coeficiente de determinação para o modelo. Após esta aplicação, foi concluída a aplicação realizada com duas variáveis e foi iniciada a aplicação com três variáveis independentes. Os resultados de cada aplicação realizada são apresentados na Seção 4.3.2.2.

4.3.2.2. Aplicação com três variáveis independentes

As aplicações realizadas aqui foram feitas da mesma forma apresentada na aplicação das duas variáveis independentes, realizando todas a combinações possíveis com as quatro variáveis com maior correlação apresentada na Seção 4.3.1, com aplicações simples. Sendo estas combinações: *temperatura máxima, temperatura compensada média* e *insolação*; *temperatura máxima, temperatura compensada média* e *evaporação*; e *temperatura compensada média, insolação* e *evaporação*. Cada uma destas aplicações é apresentada nas seções seguintes, iniciando pela aplicação das variáveis de *temperatura máxima, temperatura compensada média e insolação.*

4.3.2.2.1. Temperatura Máxima e Temperatura Compensada Média e Insolação vs Umidade Relativa Média

As variáveis de *temperatura máxima* (TMx), *temperatura compensada média* (TCM) e *Insolação* (INS) foram aplicadas ao algoritmo SVR, e os resultados desta aplicação são apresentados na Tabela 17.

Tabela 17 - TMx e TCM e INS vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	**7,21655**	0,20309	**0,72883**	0,85371
	10	7,21656	0,20316	0,72865	0,85361
	100	7,21650	0,20318	0,72860	0,85358
	1000	7,21768	0,20317	0,72864	0,85360
	10000	7,28789	0,20607	0,72083	0,84901

Com resultados apresentados na Tabela 17, verifica-se que o erro médio obtido foi de aproximadamente 7,21, e em relação as aplicações realizadas anteriormente, este resultado não é muito satisfatório. O mesmo acontece com o coeficiente de determinação que possui o valor de 0,72, aproximadamente, e consequentemente o coeficiente de correlação possui o valor de 0,85. Os resultados obtidos não são ruins, porem existem modelos anteriores que apresentaram resultados bem melhores que este.

A próxima aplicação realizada foi com as variáveis de *temperatura máxima*, *temperatura compensada média* e *evaporação*.

4.3.2.2.2. Temperatura Máxima e Temperatura Compensada Média e Evaporação vs Umidade Relativa Média

Os resultados obtidos a partir da aplicação das variáveis de *temperatura máxima* (TMx), *temperatura compensada média (*TCM) e *evaporação* (EVP), como sendo as variáveis independentes, no algoritmo SVR são apresentados na Tabela 18.

Tabela 18 - TMx e TCM e EVP vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	**5,01195**	0,14881	**0,85441**	0,92434
	10	5,01191	0,14882	0,85439	0,92433
	100	5,01199	0,14892	0,85420	0,92423
	1000	5,02053	0,14838	0,85525	0,92480
	10000	5,04690	0,14845	0,85513	0,92473

A Tabela 18 apresenta os resultados obtidos com aplicação das mencionadas variáveis. O erro médio desta aplicação está entre, aproximadamente, 5,01 e 5,04, com o coeficiente de determinação variando entre 0,8542 e 0,8552, e consequentemente com o coeficiente de correlação entre 0,9242 e 0,9248, bem próximo de 1. Os resultados obtidos com esta aplicação são bastantes satisfatórios, visto que 85% das variações da variável dependente podem ser explicadas pela variação das variáveis independentes, com uma correlação forte, porem com o

erro médio um pouco acima de alguns modelos já apresentados. Isso faz com este modelo seja um modelo considerado razoável.

Em seguida, é apresentada a aplicação das variáveis de *temperatura máxima, insolação* e *evaporação,* com sendo as variáveis independentes. E os resultados são apresentadas na seção seguinte.

4.3.2.2.3. Temperatura Máxima e Insolação e Evaporação vs Umidade Relativa Média

Aplicação realizada aqui envolve a utilização das variáveis de *temperatura máxima* (TMx), *insolação* (INS) e *evaporação* (EVP) como variáveis independentes, e os resultados obtidos a partir desta aplicação são apresentados na Tabela 19.

Tabela 19 - TMx e INS e EVP vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	4,66865	0,13938	0,87518	0,93551
	10	4,66868	0,13937	0,87519	0,93551
	100	**4,66867**	0,13940	**0,87514**	0,93549
	1000	4,66914	0,13956	0,87485	0,93533
	10000	4,80173	0,14576	0,86349	0,92924

Com os resultados apresentados na Tabela 19, nota-se que está aplicação apresentou resultados muito bons, com o valor do erro médio em aproximadamente 4,66. O coeficiente de determinação em aproximadamente 0,87 e o coeficiente de correlação em 0,93, correlação considerada uma correlação forte. Estes resultados fazem com esta aplicação esteja entre as aplicações que apresentaram os melhores resultados, mesmo com a variação do parâmetro C.

4.3.2.2.4. Temperatura Compensada Média e Insolação e Evaporação vs Umidade Relativa Média

Esta seção apresenta os resultados obtidos com a aplicação da *temperatura compensada média* (TCM), *insolação* (INS) e *evaporação* (EVP) como sendo variáveis independentes, e estes resultados são apresentados na Tabela 20.

Tabela 20 - TCM e INS e EVP vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	**4,88399**	0,14397	**0,86662**	0,93092
	10	4,88400	0,14399	0,86658	0,93090
	100	4,88405	0,14404	0,86648	0,93085
	1000	4,88465	0,14398	0,86660	0,93091
	10000	4,99228	0,14475	0,86516	0,93014

A Tabela 20 apresenta os resultados obtidos na aplicação, com o erro médio encontrado de aproximadamente 4,88, com 86% (coeficiente de determinação em aproximadamente 0,86) das variações da variável dependente podendo ser explicada pela variação das variáveis independentes, e uma correlação forte entre as variáveis (coeficiente de correlação em aproximadamente 0,93).

Esta aplicação encerra as aplicações realizadas com três variáveis independentes, e o melhor resultado encontrado foi com a aplicação das variáveis de *temperatura máxima*, *insolação* e *evaporação*, porém não se diferenciando muitos das outras aplicações que foram realizadas com três variáveis. Ainda procurando encontrar a aplicação com melhor resultado, foi realizado a aplicação das quatro variáveis juntas, como variáveis independentes, e os resultados obtidos desta aplicação são apresentadas na Seção 4.3.2.3.

4.3.2.3. Aplicação com quatro variáveis independentes

Foi realizado a aplicação do SVR com as quatro variáveis que possuíram as maiores correlações na aplicação simples, sendo elas: a *temperatura máxima* (TMx), *temperatura compensada média* (TCM), *insolação* (INS) *e evaporação* (EVP). E os resultados obtidos da utilização destas quatro variáveis são apresentados na Tabela 21.

Tabela 21 - TMx e TCM e INS e EVP vs URM.

Dados com Tratamento-1?	*C*	*Erro Médio*	*Erro Padrão*	*Coef. De Determinação*	*Coef. De Correlação*
Sim	1	4,71348	0,13743	0,87583	0,93586
	10	**4,71347**	0,13743	0,87584	**0,93586**
	100	4,71368	0,13741	0,87586	0,93587
	1000	4,73142	0,13795	0,87489	0,93536
	10000	4,88346	0,14189	0,86764	0,93147

Os resultados apresentados pela Tabela 21 mostram que a aplicação das quatros variáveis gerou resultados satisfatórios, devido que o erro médio foi pequeno em relação as outras aplicações, possuindo o valor de 4,71. O coeficiente de determinação obteve o valor de 0,87 e o coeficiente de correlação obteve o valor de 0,93, sendo estes os melhores valores

obtidos para estas variáveis (coeficiente de determinação e coeficiente de correlação) até o momento. Está aplicação mostrou que 87% das variações da variável dependente podem ser explicadas pela variação das variáveis independentes, e a correlação entre elas é considerada forte, apresentando uma correlação de aproximadamente 0.93, que bem próxima de 1 (correlação perfeita ou ótima). Entretanto, ainda não é possível afirmar que está aplicação é a que possui os melhores resultados, mas ela está entre as melhores aplicações, se diferenciando em décimos das demais.

Após está etapa de verificação de qual o método apresenta os melhores resultados foi iniciado o processo de treinamento do modelo, e comparação dos resultados (acertos e erros) entre os melhores modelos. Esta tarefa é apresentada no Seção 4.3.3.

4.3.3. Escolha do melhor modelo

Mesmo com os modelos apresentados na regressão linear múltipla, não foi possível determinar o qual seria o melhor modelo para a realizar a predição dos dados climáticos, devido que muitos apresentaram resultados semelhantes. Sendo assim, foram realizados novos e diferentes testes, com objetivo de analisar outros resultados obtidos por cada modelo. Porém, este novo teste foi realizado apenas com os modelos que apresentaram os melhores resultados. Foi realizado uma seleção entre os modelos procurando selecionar os modelos com os melhores resultados.

Nos modelos aplicados com duas variáveis independente, os modelos com melhores resultados foram os de *temperatura máxima* (TMx) e *evaporação* (EVP), e *insolação* (INS) e *evaporação* (EVP). Estes modelos foram escolhidos como os melhores modelos com duas variáveis, analisando os resultados de erro médio e coeficiente de correlação e coeficiente de determinação de cada aplicação.

Passando para análise do melhor modelo com três variáveis independente, conclui-se que o modelo que apresentou os melhores resultados foi o modelo de *temperatura máxima* (TMx) e *insolação* (INS) e *evaporação* (EVP). Este modelo possui o melhor resultado de erro médio e dos coeficientes de correlação e determinação, porém, não possui grande diferença entre os resultados do modelo com a utilização das variáveis de *temperatura média compensada* (TMC) e *insolação* (INS) e *evaporação* (EVP), se diferenciando em alguns resultados por milésimos.

Como só foi criado um modelo com quatro variáveis independentes, este é o melhor modelo com quatro variáveis. A partir da definição dos quatro melhores modelos foi realizada uma análise entre estes quatro modelos, com objetivo de definir qual será o modelo definido

como melhor modelo para a aplicação. Iniciando-se pela verificação dos valores de erro médio de cada modelo.

Tabela 22 - Resultados dos melhores modelos

Modelo	*Erro Médio*	*Coef. De Determinação*	*Coef. De Correlação*
TMx e EVP	4,900	0,863	0,929
INS e EVP	4,871	0,864	0,929
TMx e INS e EVP	4,668	**0,875**	0,935
TMx e TCM e INS e EVP	4,713	**0,875**	0,935

A Tabela 22 apresenta os resultados de cada um dos modelos que apresentaram os melhores resultados. O valor de erro médio dos modelos TMx e EVP, INS e EVP, TMx e INS e EVP, e TMx e TCM e INS e EVP, no melhor caso são, respectivamente: 4,9, 4,871, 4,668, 4,713. Verifica-se que o modelo com o menor erro médio é o modelo com de três variáveis (TMx e INS e EVP).

Realizando a comparação dos valores de coeficiente de determinação e coeficiente de correlação, tem-se que os valores do coeficiente de determinação e correlação de cada modelo são: 0.863 e 0.929 (TMx e INS), 0.864 e 0,929 (INS e EVP), 0,875 e 0,935 (TMx e INS e EVP), 0,875 e 0,935 (TMx e TCM e INS e EVP). Nesta comparação nota-se que os melhores modelos foram os modelos com três (TMx e INS e EVP) e quatro (TMx e TCM e INS e EVP) variáveis, com resultados iguais para esta avalição.

Com isso fica determinado como os melhores modelos a aplicação com três e quatro variáveis, porem o objetivo é encontrar um modelo que possua o melhor resultado para aplicação. Desta forma foi calculado para os dois modelos o erro médio absoluto, e realizado a verificação da quantidade de acerto dos valores predito por cada um dos modelos, utilizando um grau de confiança de 95%. Os resultados desta aplicação são apresentados na Tabela 23.

Tabela 23 - Comparação entre modelos

Modelo	*C*	*Erro Médio Absoluto*	*Margem de Erro*	*Nº de dados*	*Acertos*	*% de Acertos*
TMx e INS e EVAP	1	3,6642	**7,1819**	1797	**1453**	80
TMx e TCM e INS e EVAP	1	3,6123	**7,0802**	1797	**1444**	80

A Tabela 23 apresenta os valores de erro médio absoluto e margem de erro de cada um dos modelos, juntamente com apresentação do número de predições realizadas corretamente pelo modelo, respeitando a margem de erro encontrada. Os resultados dos dois modelos são bastantes semelhantes, em que o modelo com quatro variáveis possui uma margem de erro menor, porem realiza a uma quantidade menor de acertos que o modelo com três variáveis. Como a diferença entre os resultados é muita pequena, foi realizado mais um teste com estes dois modelos procurando analisar o desempenho de cada um deles para realizar a predição,

variando o tamanho do conjunto de dados de treinamento e do conjunto de dados de teste. Os resultados desta aplicação são apresentados nas Tabela 24 e Tabela 25.

Tabela 24 - Resultado da predição do modelo variando dados de treino e teste.

Modelo	*TMx e INS e EVAP*								
Nº de dados de Treinamento	1687	1437	1257	1078	898	718	539	359	179
Nº de dados de Teste	180	360	540	719	899	1079	1258	1438	1618
Erro Médio Absoluto	3,7902	**3,6520**	3,8808	3,8838	3,8946	3,8608	4,0115	3,9208	**4,1734**
Predições Certas	141 (78,33%)	294 (81,66%)	444 (82,22%)	599 (83,31%)	734 (81,64%)	884 (81,92%)	1020 (81,08%)	1183 (82,26%)	1332 (82,32%)
Predições Erradas	39 (21,66%)	66 (18,33%)	96 (17,77%)	120 (16,68%)	165 (18,35%)	195 (18,07%)	238 (18,92%)	255 (17,74%)	286 (17,68%)

A Tabela 24 apresenta os resultados obtido pelo modelo TMx e INS e EVP, foram realizados nove testes, variando o tamanho dos conjuntos de treinamento e de teste e apresentados a quantidade de acertos e de erro em cada um dos testes. Na Tabela 25 são apresentados os resultados dos nove testes realizados com o modelo TMx e TCM e INS e EVP, variando da mesma forma que o modelo de três variáveis.

Tabela 25 - Resultado da predição do modelo variando dados de treino e teste.

Modelo	*TMx e TCM e INS e EVAP*								
Nº de dados de Treinamento	1687	1437	1257	1078	898	718	539	359	179
Nº de dados de Teste	180	360	540	719	899	1079	1258	1438	1618
Erro Médio Absoluto	3,8678	**3,6541**	3,8823	3,8725	3,8350	3,8737	3,9977	3,9607	**4,1237**
Predições Certas	142 (78,89%)	294 (81,67%)	444 (82,22%)	597 (83,03%)	728 (80,98%)	883 (81,84%)	1020 (81,08%)	1181 (82,13%)	1308 (80,84%)
Predições Erradas	38 (21,11%)	66 (18,33%)	96 (17,78%)	122 (16,97%)	171 (19,02%)	196 (18,16%)	238 (18,92%)	257 (17,87%)	310 (19,16%)

A partir dos dados apresentados nas tabelas Tabela 24 e Tabela 25 nota-se que os desempenhos de ambos modelos são similares, com valores quase idênticos, variando na maioria dos casos em uma unidade. Desta forma, conclui-se que, ambos os modelos podem ser utilizados para a predição de valores climáticos, devido que os resultados são semelhantes, ocorrendo variações em nível de milésimos de diferença, o que neste caso não possui muita significância. Freund (2006) afirma que se for possível diminuir o número de variáveis independentes, deve ser feito. A partir desta afirmação, decidiu-se adotar como o melhor modelo para a aplicação o modelo com três variáveis (TMx e INS e EVP), visto que o mesmo possui um número menor de variáveis independente em relação ao outro modelo.

5 CONSIDERAÇÕES FINAIS

Os resultados obtidos com a aplicação da Máquina de Vetores de Suporte para a realização da predição de dados climáticos mostraram que, as variáveis climáticas que mais podem influenciar no valor da umidade relativa média são as variáveis de temperatura máxima e evaporação piche. Entretanto, aplicação de cada uma individualmente, ou em pares, não é suficiente para obter os resultados bons para a predição dos valores da umidade relativa média.

Foi acrescentado a variável de insolação juntamente com as variáveis de temperatura máxima e evaporação, e a partir disso, foi obtido um modelo que realiza a predição dos dados climáticos com resultados bons para esta tarefa.

Porém, os resultados obtidos somente são validos para o conjunto de dados utilizados, que foram os dados climáticos da região de Palmas – TO, obtido através do site do inmete.gov.br. Visto que esta região possui particularidades que são exclusivas da região. Como o valor da temperatura, tanto a máxima, quanto a mínima, nunca forem registradas como sendo menor ou igual a zero.

Outra característica destes resultados, é que eles foram obtidos pelo algoritmo SVM fornecido pela biblioteca Scikit-learn, e podem variar se for utilizado outras implementações do SVM.

O protótipo do software foi desenvolvido como uma página web, que permite ao usuario fazer o upload do arquivo que deseja utilizar como base de dados. E permite ao usuario selecionar quais os valores ele que aplicar como sendo variáveis independentes e variável dependentes. Possibilitando uma grande quantidade de aplicações que podem ser realizadas e analisadas, em busca de uma melhor aplicação.

E a partir dos estudos e análises dos resultados obtidos é possível concluir que o algoritmo de Máquina de Vetores de Suporte pode ser utilizado para a predição de dados climáticos, mais especificamente para prever o valor da umidade relativa média.

6 REFERÊNCIAS BIBLIOGRÁFICAS

ALBUQUERQUE, Rafael Walter de. **Monitoramento da cobertura do solo no entorno de hidrelétricas utilizando o classificador SVM (Support Vector Machines).** 2012. 95 p. Dissertação (Mestrado em Engenharia de Transportes) – Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Transportes. São Paulo, 2012.

ALEXANDRE, Sueli de Fátima. **Aprendizagem e suas Implicações no Processo Educativo**. In: Revista de Letras. v. 6. Universidade Estadual de Goiás (UEG), julho 2010, Goiás. UnU de São Luís de Montes Belos. p 51-60.

ALMEIDA, Leandro M. et al. **Uma ferramenta para extração de padrões**. Palmas, 2004. 13 f. Centro Universitário Luterano de Palmas, Palmas. Disponível em: http://www.cin.ufpe.br/~lma3/UmaFerramentaParaExtracaoDePadroes.pdf. Acesso em: 16/10/2015.

ARAÚJO, Ricardo Matsumura de. **Aprendizado de máquina em sistemas complexos multiagentes:** estudo de caso em um ambiente sob racionalidade limitada. 2004. 83 p. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal do Rio Grande do Sul (UFRGS). Programa de Pós-Graduação em Computação (PPGC), Porto Alegre, RS, 2004.

ASANO, A. **Support vector machine and kernel method.** Pattern information processing (2004 Autumn Semester), Session 12, 2005.

ATHAUDA, R.; TISSERA, M.; FERNANDO, C. **Data Mining Applications: Promise and Challenges**. Data Mining and Knowledge Discovery in Real Life Applications, Julio Ponce and Adem Karahoca (Ed.), ISBN: 978-3-902613-53-0, InTech, Disponível em: http://www.intechopen.com/books/data_mining_and_knowledge_discovery_in_real_life_applications/data_mining_applications__promise_and_challenges. Acesso em: 16/10/2015.

BARREIRA, R. G. **Análise de sentimentos com rapidminer.** 2013, 74 f. Trabalho de Conclusão de Curso em Sistemas de Informação - Centro Universitário Luterano de Palmas, Palmas, TO, 2013.

BARRENO, Marco; NELSON, Blaine; SEARS, Russell; JOSEPH, Anthony D.; TYGAR, J. D.; **Can Machine Learning Be Secure?** Computer Science Division, University of California, Berkeley, 2006. In: Proceedings of the ACM Symposium on Information, Computer and Communications Security (ASIACCS), pages 16–25, 2006

BASAK, Debasish et al. **Support Vector Regression.** Neural Information Processing: Letters and Reviews, 11(10), 2007.

BELTRAMI, M. et al. **Comparação das técnicas de support vector regression e redes neurais na precificação de opções**. In: XLII Simpósio Brasileiro de Pesquisa Operacional (SBPO), Bento Gonçalves, 30 de agosto de 2010. p.572-583.

BÍSCARO, Guilherme Augusto. **Meteorologia agrícola básica**. 1ª ed. Cassilândia, MS: Unigraf, 2007.

BISOGNIN, Gustavo. **Utilização de Máquinas de Suporte Vetorial para Predição de Estruturas Terciárias de Proteínas**. 2007. 102 p. Dissertação (Mestrado em Computação Aplicada) - Universidade do Vale do Rio dos Sinos (UNISINOS). São Leopoldo, 2007.

BURGES, Christopher J. C. **A tutorial on support vector machines for pattern recognition.** In: Data Mining an Knowledge Discovery 2, 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.

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.

CONDUTA, Bruno Custódio; MAGRIN, Diego Henrique. **Aprendizagem de Máquina.** 2010. 19p. Dissertação (Mestrado em Inteligência Artificial) – Universidade Estadual de Campinas (UNICAMP). Faculdade de Tecnologia (FT). Limeira, SP, 2010.

COSTA, Ernesto; SIMÕES, Anabela. **Inteligência Artificial**: Fundamentos e Aplicações. Lisboa, Portugal: FCA, 2008.

CUAYÁHUITIL, Heriberto; DETHLEFS, Nina; OTTERLO, Martijn van; FROMMBERGER, Lutz. **Machine Learning for Interactive Systems and Robots**: A Brief Introduction. In MLIS, 2013, p. 19-28.

D'HAINAUT, Louis. **Conceitos e métodos da estatística**: volume II – Duas ou três variáveis segundo duas ou três dimensões. Tradução de Antônio Rodrigues Lopes e Maria da Conceição Carreiras Lopes. Fundação Calouste Gulbenkian, 1992. 365 p. Título original: Concepts et methodes de la statistique, tome 2.

DJAIR. **Diagrama de Dispersão.** Lugli.com.br. Disponível em: www.lugli.com.br/2008/02/diagrama-de-dispersao/. Acesso em: 13/03/2015.

DRUCKER, Harris; BURGES, Chris .J.C.; KAUFMAN, Linda; SMOLA, Alex; VAPNIK, Vladimir. **Support vector regression machines** In: M. Mozer, M. Jordan, and T. Petsche, editors, Advances in Neural Information Processing Systems 9, pages 155-161, Cambridge, MA, MIT Press.

EL-NAQA, Issam et al. **A support vector machine approach for detection of microcalcifications.** IEEE Transactions on Medical Imaging, 2(12): 1562 - 1563, December 2002.

ERÄSTÖ, Panu. **Support Vector Machines - Backgrounds and Practice.** Licentiate Thesis, University of Helsinki, Rolf Nevanlinna Institute, Faculty of Science. 2001.

FACELI, Katti. et al. **Inteligência Artificial:** uma abordagem de aprendizagem de máquina. Rio de Janeiro: LTC, 2011.

FAYYAD, U. M. et al.. **Advances in Knowledge Discovery and Data Mining**. The MIT Press. 1996. Disponível em: http://www.worldcat.org/isbn/0262560976. Acesso: 16/10/2015.

FERNANDES, Anita Maria da Rocha. **Inteligência Artificial:** Noções Gerais. Florianópolis. Visual Books, 2003.

FERRO, Mariza; LEE, Huei Diana. **O Processo de KDD Knowledge Discovery in Database para Aplicações na Medicina**. In: Seminc 2001, 2001, Cascavel. Anais da Seminc 2001, 2001.

FONSECA, Jairo Simon da; MARTINS, Gilberto de Andrade; TOLEDO, Geraldo Luciano. **Estatística aplicada**. 2. ed. São Paulo: Atlas, 1985.

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.

GSI - Grupo de Sistemas Inteligentes. **Mineração de Dados**. Departamento de Informática (DIM). Universidade Estadual de Maringá (UEM). Maringá, 1998. Disponível em: <http://www.din.uem.br/ia/mineracao/tecnologia/relacionadas.html>. Acesso em: 20/03/2015.

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

HEARST, Marti A.; SCHÖLKOPF, Bernhard; DUMAIS, Susan T.; OSUNA, Edgar; PLATT, John. **Trends and controversies - support vector machines**. IEEE Intelligent Systems, 13(4): p. 18–28, 1998.

HOEL, Paul G. **Estatística Elementar**. São Paulo, Atlas, 1981.

HOFFMANN, Rodolfo. **Estatística para Economistas**, 3ª ed. rev. e amp. São Paulo, Biblioteca Pioneira de Ciências Sociais, 1998.

HOFMANN, Thomas. **Unsupervised learning by probabilistic latent semantic analysis.** Machine Learning, 42: 177 – 196, 2001.

HSU, Chih-Wei; LIN, Chin-Jen. **A comparison of methods for multi-class support vector machines.** IEEE Transactions on Neural Networks, 13(2):415–425, March 2002.

KOERICH, Alessandro L. **Aprendizagem de Máquina**. Pontifícia Universidade Católica do Paraná (PUCPR) - Paraná, 2008. 75 slides, color. Acompanha Texto.

LANGLEY, Pat; SIMON, Herbert A.; **Applications of machine learning and rule induction**. Communications of the ACM, v.38 n.11, p.54-64, Nov. 1995.

LARSON, Ron; FARBER, Betsy. **Estatística aplicada**. 2 ed. São Paulo: Prentice Hall, 2010. 476 p.

LIMA, Allan Reffson Granja. **Máquina de Vetores de Suporte na Classificação de Impressões Digitais.** 2002. 81 p. Tese de Mestrado – Universidade Federal do Ceará. Fortaleza, Ceará, 2002.

LORENA, Ana Carolina; CARVALHO, André C. P. L. F. de. **Uma Introdução às Support Vector Machines**. Revista de Informática Teórica e Aplicada (RITA) V.14, n.2, p.43-67, 2007.

LUGER, George F. **Inteligência artificial**: estruturas e estratégias para resolução de problemas complexos/ George F. Luger; trad. Paulo Engel. 4.ed. – Porto Alegre: Bookmann, 2004.

MEDEIROS, Ericles Alves de. **Técnica de Aprendizagem de Máquina para Categorização de Textos**. Recife: Universidade de Pernambuco, 2004.

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

MORAES, Silvia Maria Wanderley; LIMA, Vera Lúcia Strube de. **Categorização de textos baseada em conceitos**. Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS) – Faculdade de Informática (FACIN), 2008. 15 Slides: color. Acompanha Texto.

MOREIRA, Adriana Aparecida; FERNANDES, Fernando Hiago Souza; NERY, César Vinícius Mendes. **Aplicação do algoritmo Support Vector Machine na Analise espaço-temporal do uso e ocupação do solo na bacia do Rio Viera**. Instituto de Geografia (UFU), Programa de Pós-graduação em Geografia. CAMINHOS DE GEOGRAFIA – revista online 2014, Uberlândia, V.15, n.50, p152-153.
http://www.seer.ufu.br/index.php/caminhosdegeografia/

NASCIMENTO, Renata Fernandes Figueira. et al. **O algoritmo *Support Vector Machines (SVM)*:** a avaliação da separação ótima de classes em imagens CCD-CBERS-2. In: Simpósio Brasileiro de Sensoriamento Remoto (SBSR), 25-30 abril 2009, Natal. Anais... XIV Simpósio Brasileiro de Sensoriamento Remoto, Natal, Brasil, 25-30 abril 2009, Instituto Nacional de Pesquisas Espaciais (INPE), p 2079-2086.

NILSSON, Nils J. **Introduction to machine learning**.1998 Stanford, CA, USA. Disponível em: http://robotics.stanford.edu/people/nilsson/mlbook.html

OGURI, Pedro. **Aprendizado de Máquina para o Problema de Sentiment Classification**. 2006. 54p. Dissertação de Mestrado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro. Rio de Janeiro.

OSUNA, Edgar; FREUND, Robert; GIROSI, Federico. **Training support vector machines: An application to face detection.** In: Proc. IEEE Workshop on Neural Networks and Signal Processing, IEEE Press, Piscataway, N.J., 1997, p. 130 – 136.

PRASS, Fernando Sarturi. **Uma visão geral sobre as fases do Knowledge Discovery in Databases (KDD).** 2012. Disponível em: http://fp2.com.br/blog/index.php/2012/um-visao-geral-sobre-fases-kdd/. Acesso em: 16/10/2015.

PETERNELLI, Luiz Alexandre. **Estatística I**. Vila Velha, 2004. Disponível em: www.dpi.ufv.br/~peternelli/inf162.www.16032004/materiais/CAPITULO9.pdf. Acesso em: 23/03/2015.

PRUDÊNCIO, Ricardo. **Aprendizado de Máquina**: Introdução. Centro de Informática – Universidade Federal de Pernambuco (UFPE), 2008. 31 slides, color. Acompanha texto.

RASHID, Ekbal; PATNAYAK, Srikanta; BHATTACHERJEE, Vandana. **A Survey in the Area of Machine Learning and Its Application for Software Quality Prediction**. ACM SIGSOFT Software Engineering Notes page 1 September 2012.37(5).

RIBEIRO, Daniel José Silva. **Support Vector Machines na previsão do comportamento de uma ETAR.** 2012. 160p. Dissertação (Mestrado em Engenharia de Informática) – Universidade do Minho. Gualtar, Braga, PT, 2012.

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

RUFINO, Hugo Leonardo Pereira. **Algoritmo de Aprendizado Supervisionado Baseado em Máquina de Vetores de Suporte**: Uma Contribuição Para o Reconhecimento de Dados Desbalanceados. 2011. 107p. Dissertação de Doutorado – Universidade Federal de Uberlândia. Uberlândia, MG.

RUSSELL, Stuart. NORVIG, Peter. **Inteligência Artificial**. Tradução de Regina Célia Simille. Rio de Janeiro: Elsevier, 2013. Título original: Artificial Intelligence, 3rd. Ed.

SANTOS, Eulanda Miranda. **Teoria e Aplicação de Support Vector Machines à Aprendizagem e Reconhecimento de Objetos baseados na Aparência**. 2002. 121 p. Dissertação (Mestrado em Informática) – Universidade Federal da Paraíba. Campina Grande, Paraíba, 2002.

SCARINCI, R. G. Extração de informação como base para descoberta de conhecimento em dados não estruturados. In: Workshop interno sobre descoberta de conhecimento em bases de dados, 1., 2000, Porto Alegre. Porto Alegre: Instituto de Informática da UFRGS, 2001. v. 1, p. 15-20.

SFERRA, Heloisa Helena; CORREA, Ângela M. C. Jorge. **Conceitos e Aplicações de Data Mining**. Jul/Dez de 2003, Revista Ciência & Tecnologia, PP. 19-34

SILVA, Ivan de Souza. et al. **A importância da Inteligência Artificial e dos sistemas especialistas**. In: Congresso Brasileiro de Ensino de Engenharia (COBENGE). Setembro de 2004, Brasília.

SMOLA, Alex J.; SCHÖLKOPF, Bernhard. **A tutorial on support vector regression**. Technical Report NC2-TR-1998-030, NeuroCOLT2, 1998.

SOARES, Fabio de Azevedo. **Aprendizado de Máquina**. Rio de Janeiro: 2008. 18. color. desenho, texto.

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.

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

TORRES, Nilton Ricoy; D'OTTAVIANO, Camila. **Estatística Aplicada**. Universidade de São Paulo. São Paulo, 2015.

VAPNIK, V. N.; CHERVONENKIS, A. Y. apud ALBUQUERQUE, Rafael Walter. **Monitoramento da Cobertura do Solo no Entorno de Hidrelétricas Utilizando o Classificador SVM (Support Vector Machines)**. 2012, 107p. Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo. São Paulo.

VAPNIK, Vladmir. **The Nature of Statistical Learning Theory**. New York: Springer-Verlag, 1995.

VAPNIK, V.; BOSER, B.; GUYON, I. **A training algorithm for optimal margin classifiers.** Conference on Computational Learning Theory 15th (COLT 1992), 1992, pp. 144-152.

WELLING, Max. **Support Vector Regression**. Disponível em: http://www.ics.uci.edu/~welling/teaching/KernelsICS273B/Svregression.pdf. Acesso em: 01/04/2015.