

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

Luiz Filipe Lopes Fialho

PROJETO DE DIPLOMAÇÃO

**SISTEMA DE PLANEJAMENTO DE IRRIGAÇÃO BASEADO
EM REDES NARX**

Porto Alegre
(2015)

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

SISTEMA DE PLANEJAMENTO DE IRRIGAÇÃO BASEADO EM REDES NARX

Projeto de Diplomação apresentado ao Departamento de Engenharia Elétrica da Universidade Federal do Rio Grande do Sul, como parte dos requisitos para Graduação em Engenharia Elétrica.

ORIENTADOR: Prof. Dr. Marcelo Götz

Porto Alegre
(2015)

LUIZ FILIPE LOPES FIALHO

SISTEMA DE PLANEJAMENTO DE IRRIGAÇÃO BASEADO EM REDES NARX

Este projeto de diplomação foi analisado e julgado adequado para a obtenção do título de Engenheiro Eletricista e aprovado em sua forma final pelo Orientador e pela Banca Examinadora designada pelo Departamento de Engenharia Elétrica da Universidade Federal do Rio Grande do Sul.

Prof. Dr. Marcelo Götz

Prof. Dr. Ály Flores Ferreira Filho

Aprovado em: __/__/____

Banca Examinadora:

Prof. Dr. Marcelo Götz, UFRGS _____

Prof. Dr. Renato Ventura, UFRGS _____

Eng Roberto Walter, UFRGS _____

Porto Alegre, 2015.

AGRADECIMENTOS

Agradeço a todos que fizeram parte desta maravilhosa e desafiante etapa da minha vida.

RESUMO

A irrigação representa 70% do uso de água no mundo. Apesar do custo elevado da instalação de sistemas de irrigação, o manejo pós-instalação é na maioria das vezes feito a partir da utilização de técnicas rudimentares e avaliações empíricas da lavoura. Ocorre que este fato acarreta uma perda de produtividade na ordem de 40% em relação à tomada de decisão baseada nas melhores práticas de manejo. Diversas metodologias efetivas de controle de umidade do solo entraram no mercado nos últimos anos, todavia a maioria delas é contra intuitiva para o agricultor e não permite um processo fácil de tomada de decisão.

Neste sentido, o presente trabalho visa obter uma solução deste impasse, através da criação de um programa, baseado em rede NARX (*Nonlinear Autoregressive Network with Exogenous Inputs*) que consiga efetivamente prever a umidade futura do solo. O resultado esperado deve ser primeiramente uma performance adequada para o algoritmo proposto e posteriormente o design de um software em programação orientada ao objeto. É esperado que o sistema aqui desenvolvido, possa tornar parte de um projeto de maior magnitude aplicando-o a um sistema de hardware e via de consequência contribuir para o aumento da produtividade na agricultura, bem como minorar o desperdício de água no mundo.

Palavras-chaves: Engenharia Elétrica. Aprendizado de Máquina. Controle. Irrigação e NARX.

ABSTRACT

Irrigation represents around 70% percent of the water usage in the world. Even though, the high development cost of irrigations systems, the watering decision is majorly based on empirical analysis and rudimentary techniques. This represents a spoil rate around 40% comparing to the state of the art techniques available. A number of soil moisture estimation techniques were developed in the last years, although many of them are not simple to use, keeping the irrigation management still difficult for the ordinary farmer.

This research aims to obtain a solution for this impasse, designing a program, using a NARX (*Nonlinear Autoregressive Network with Exogenous Inputs*), that allows to effectively estimate future soil moisture. The longed for result must be primarily obtaining an adequate performance for the proposed algorithm and later design a software using objected oriented programing. The expected outcome for the developed system will allow the development of a bigger project, which constitutes the design of a hardware system that may improve agriculture productivity and help to attenuate world's water spoil.

Keywords: Electrical Engineering. Machine Learning, Irrigation Control and NARX

SUMÁRIO

1	INTRODUÇÃO.....	14
1.1	Objetivos do trabalho.....	15
2	CONTEXTUALIZAÇÃO TEÓRICA	17
2.1	Sistemas de Irrigação	17
2.1.1	Sistema por aspersão simples.....	17
2.1.2	Sistema por gotejamento	18
2.1.3	Sistema de Pivô central	19
2.2	Metodologias agrônômicas para controle da umidade do solo	20
2.2.1	Sensoriamento remoto.....	21
2.2.2	Medição <i>in Situ</i>	22
2.3	Aprendizado de Máquina	22
2.3.1	Redes Neurais	23
2.3.2	Rede neural recorrente e rede NARX.....	29
3	METODOLOGIA EXPERIMENTAL.....	31
3.1	Escolha do campo de estudos e obtenção da base de dados.....	31
3.1.1	Campo de estudos.....	31
3.1.2	Escolha do modelo de previsão de precipitação.....	35
3.2	Obtenção das Bases de dados do estudo e manejo dos dados	36
3.3	Análise das variáveis de entrada para o modelo	40
3.4	Estruturação do Algoritmo de Aprendizado	43
3.4.1	Atrasos nas entradas e no laço de realimentação	43
3.4.2	Estruturação das variáveis no algoritmo	44
3.4.3	Número de neurônios na camada escondida	46
3.4.4	Software utilizado e linguagem de programação	46
3.5	Estruturação do Programa Java	47
3.5.1	Algoritmo Iterador para previsão do momento de irrigação	49
3.5.2	Bibliotecas Adicionais Utilizadas.....	50
4	RESULTADOS	51
4.1	Avaliação do desempenho do algoritmo não preditivo	51
4.2	Avaliação do resultado utilizando previsão do tempo	57
4.3	Avaliação do resultado do programa estruturado	62

5	CONCLUSÃO	64
---	-----------------	----

LISTA DE ILUSTRAÇÕES

Figura 2.1: Foto de uma fazenda com sistema de Aspersão simples.	18
Figura 2.2: Ilustração representativa de um sistema de gotejamento	19
Figura 2.3: Sistema de pivô central em funcionamento.....	20
Figura 2.4: Vista próxima do sistema de pivô central da empresa Lindsay.....	20
Figura 2.5: Representação gráfica das classificações de algoritmos de I.A.....	23
Figura 2.6: Modelo representativo de um perceptron utilizado em uma rede neural.	24
Figura 2.7: Modelo com quatro perceptrons associados.....	26
Figura 2.8: Representação de uma rede NARX.....	29
Figura 3.1: Imagem de satélite e mapa do experimento SMAPVEX12	33
Figura 3.2: Representação dos sensores da rede no estudo SMAPVEX 12.....	33
Figura 3.3: Foto representativa de satélite dos campos agrícolas do experimento ...	34
Figura 3.4: Representação do acúmulo de precipitação dos modelos GFS e NAM..	36
Figura 3.5: Esquemático do processo decisório para manejo de dados faltantes.....	39
Figura 3.6: Dados observados para os campos AAFC do SMAPVEX12.	42
Figura 3.7: Representação da rede neural com a camada escondida e atrasos.....	44
Figura 3.8: Estrutura do Algoritmo NARX com as variáveis de entrada analisadas. .	45
Figura 3.9: Representação do Algoritmo NARX desconsiderando o NDVI.	45
Figura 3.10: Representação da estrutura do Algoritmo NARX em cadeia.	46
Figura 3.11: Representação gráfica da utilização das classes no programa.	47
Figura 3.12: Representação gráfica (<i>Flow Chart</i>) do algoritmo Iterador.....	49
Figura 4.1: Performance para diferentes arquiteturas.	52
Figura 4.2: Arquitetura ótima para a o algoritmo proposto.	52
Figura 4.3: Histograma para os campos escolhidos com estimativa em 5cm.	54
Figura 4.4: Desempenho no domínio tempo e erros para previsão em 5cm.....	55
Figura 4.5: Performance do algoritmo calculado para 1000 iterações por campo.....	57

Figura 4.6: Resultados do desempenho no domínio tempo	58
Figura 4.7: Resultados do desempenho no domínio tempo para medição de 20 cm	59
Figura 4.8: Estimativa do sistema para a decisão ótima de irrigação à 5 cm.	61
Figura 4.9: Resposta do sistema permitindo dois momento de irrigação.	61
Figura 4.10: Interface do programa em Java.....	62
Figura 4.11: Interface para entrada de usuário	62
Figura 4.12: Diálogo de cadastro do campo.....	63
Figura 4.13: Visualização da integração com Google Maps.	63

LISTA DE TABELAS

Tabela 1: Lista dos experimentos relevantes de umidade do solo.	31
Tabela 2: Referência dos campos AAFC e MAFRI	37
Tabela 3: Composição de variáveis para o modelo preditivo	43
Tabela 4: Resultado das iterações de performance para camada escondida.	51
Tabela 5: Performance do algoritmo para os quatro campos à 5cm.	55
Tabela 6: Performance do algoritmo treinado com 25 dias de medições testado em 25 dias (caso completo) e 40 dias.....	56
Tabela 7: Resultados das análises para os casos bases	58
Tabela 8: Performance do algoritmo para previsão de 4 dias.	59

LISTA DE ABREVIATURAS

AAFC: Agriculture and Agri-Food Canada

ANN: (*Artificial Neural Network*) Redes Neurais Artificiais

ETc: Evapotranspiração

IA: Inteligência Artificial

GEFS: Global Ensemble Forecast System

GFS: Global Forecast System

LM: *Levenberg-Marquardt*

NOAA: National Oceanic and Atmospheric Administration

NARX: *Nonlinear Autoregressive Network with Exogenous Inputs*

NDVI: Universidade Federal do Rio Grande do Sul

NWP: Universidade Federal do Rio Grande do Sul

NAM: North American Mesoscale

NOMADS: National Operational Model Archive and Distribution System

SMC: *Soil Moisture Content (Conteúdo de Umidade do Solo)*

UFRGS: Universidade Federal do Rio Grande do Sul

TI: *Thermal Inertia (Inércia Termal)*

Ts: Temperatura de Superfície

VI: *Vegetation Index*

1 INTRODUÇÃO

“A finalidade básica da irrigação é proporcionar água às culturas de maneira a atender às exigências hídricas [...] a quantidade de água necessária às culturas é função da espécie cultivada, do local de cultivo, do estágio de desenvolvimento da cultura, do tipo de solo e da época de plantio” (BERNARDO, 2008).

Sabe-se que a agricultura irrigada é o que mais desvia água do meio ambiente para a produção de alimentos. Segundo Nunes, (2009, p. 1):

[...] a irrigação é praticada em 17% das áreas aráveis do planeta, produzindo 40% dos alimentos do mundo e utilizando ao redor de 70% de todas as águas retiradas dos corpos d'água do planeta. Estima-se que para garantir as demandas de alimentos, a área irrigada deve crescer entre 20% e 30% até o ano de 2025.

Nos últimos 25 anos, a produtividade do Brasil na agricultura dobrou, fato devido, em parte, ao aumento da utilização da irrigação. A irrigação ineficiente tem gerado salinização e problemas de drenagem em 15.000 ha, em especial no nordeste do país. De acordo com Aquastat (2000):

[...] a área irrigada no Brasil é responsável por mais de 16% do volume total da produção e 35% do valor econômico total da produção, enquanto que no mundo estes números são da ordem de 44% e 54% respectivamente. Em média, a produtividade nas áreas irrigadas é 2,5 a 3,0 vezes maior do que a das áreas não irrigadas”.

Brito (2015) comenta que a utilização de água pela agricultura pode influenciar o nível das barragens em um percentual significativo. O nível dos rios possui influência direta na quantidade da utilização de energia termoelétrica em países que possuem uma malha energética termo hídrica, o que afeta além do preço da energia, a quantidade de poluentes emitido. Portanto, é de interesse primordial a melhor utilização da água em sistemas de irrigação.

Um manejo de irrigação eficaz necessita de uma aplicação de água no tempo e na quantidade certa. Para uma aplicação mais eficiente, agricultores e agrônomos creditam a tomada de decisão na aplicação de água na medição da umidade do solo ou do inglês, “*Soil Moisture Content*” (SMC). Desta forma, um cenário onde esse tipo de medida possa ser descoberta de forma antecipada terá valor significativo. Os

equipamentos utilizados para estas medições, apesar de possuírem uma precisão ótima para a aplicação, consomem muito tempo na captura de dados. Além disso, os modelos atualmente utilizados requerem informações detalhadas sobre parâmetros físicos do solo, muitas vezes, difíceis de se obter. Essas barreiras encorajaram os cientistas a procurar métodos alternativos para estimar a SMC, como “*Data Mining*” (*Mineração de Dados*) e “*Machine Learning*” (*Aprendizado de Máquina*).

Um outro problema que se observa na obtenção de informações de umidade do solo é na ordem de escala. Via de regra, as medições são tomadas de forma manual fazendo com que seja muitas vezes impossível monitorar grandes áreas.

Algumas técnicas de obtenção de dados remotamente ajudam a solucionar esta questão. Em conformidade com Zaman (2011 p. 15), “os dados usados em modelos hidrológicos são obtidos de diferentes fontes, como: imagens de satélites, imagens adquiridas por veículos aéreos tripulados ou não, além de medidas por sensores remotos”. Zaman (2011) ainda comenta que a análise destes dados necessita um modelo que seja flexível o suficiente para acomodar diferentes resoluções espaciais e temporais, assim como diferentes disponibilidades e frequências.

O trabalho aqui reportado, utilizará técnicas de assimilação de dados, tais como modelos de inteligência artificial para relacionar a previsibilidade de variáveis meteorológicas com a umidade do solo (SMC).

O objetivo final é a modelagem de um sistema de previsão de umidade de solo na zona de raiz de planta que consiga, a partir da utilização de sensores remotos e variáveis meteorológicas, obter previsões confiáveis e de boa resolução para a recomendação de momento e quantidade de irrigação.

1.1 Objetivos do trabalho

- Estudar as variáveis significativas para a previsão de conteúdo de água no solo (SMC)
- Criar um sistema adaptativo de previsão de umidade de solo, investigando a efetividade em um sistema de inteligência artificial com auto aprendizado
- Construir um programa iterativo capaz de obter uma predição dos efeitos da irrigação, baseando-se no algoritmo previamente citado.

1.2 Estrutura do Trabalho

No Capítulo 2 é realizada uma introdução sobre os principais conceitos necessários para o acompanhamento do texto. São introduzidos os métodos de existentes de irrigação agrícola assim como as técnicas utilizadas no manejo e variáveis relevantes para a medição da umidade do solo. São abordados os conceitos de algoritmos de aprendizado, considerando o grande universo da inteligência artificial e aprendizado de máquina, redes neurais até a escolha do algoritmo NARX e suas implicações. Especificamente tratando de redes neurais é abordado o conceito de algoritmos de minimização de erros e de prevenção do sobre-ajuste.

O Capítulo 3 trata da metodologia utilizada no trabalho de pesquisa. No início do capítulo é discutida a obtenção das diferentes bases de dados necessárias como a escolha do campo de experimentos, modelos de precipitação e dados históricos climáticos. É realizada uma análise das variáveis de interesse para o algoritmo e propõe-se estruturas para sua aplicação. Ao final do capítulo são abordadas as metodologias para a estruturação de um programa em linguagem orientada ao objeto, assim como bibliotecas utilizadas e funções de interesse.

O Capítulo 4 analisa o desempenho do algoritmo em diferentes formas, para profundidades de cinco e vinte centímetros. Inicia-se estimando a performance sem considerar a previsão do tempo e sem realizar previsões temporais até a abordar a previsão futura de umidade do solo para até 84h de previsão. Ao final é demonstrado o funcionamento do programa orientado ao objeto a partir de sua interface gráfica e demonstrado resultados da estimativa de recomendação da irrigação.

É importante argumentar que não será possível realizar testes para medir a performance da recomendação solo, entretanto a estruturação do programa é motivada no sentido de realizar a aplicação real do sistema.

2 CONTEXTUALIZAÇÃO TEÓRICA

2.1 Sistemas de Irrigação

Sistemas de irrigação são o modo como o agricultor pode controlar sua produção. “Um manejo adequado de um sistema de irrigação deverá ser capaz de propiciar ao agricultor o uso eficiente da água, para aumentar a produtividade das culturas, reduzir os custos de produção e maximizar o retorno dos investimentos” (NUNES, 2001, p. 2).

O uso consciente da água na irrigação muitas vezes é considerado de baixa prioridade. “Se a irrigação fosse utilizada de forma racional, aproximadamente 40% da água e 30% da energia consumidas seriam economizadas; sendo 30% da energia economizada devido à aplicação desnecessária da água” (CEMIG, 1993, p. 22).

Os três sistemas de irrigação mais utilizados no Brasil são: aspersão simples, gotejamento e pivô central.

2.1.1 Sistema por aspersão simples

O sistema de aspersão simples consiste em um jato d’água que pode ser aplicado a diferentes distâncias e quantidades. Testezlaf (2004) comenta sobre as vantagens da utilização do sistema de aspersão, como por exemplo, a operação fácil e cômoda, a rápida adaptação dos operadores, adaptabilidade do sistema às condições topográficas do terreno e alta eficiência de aplicação.

Uma das desvantagens dos sistemas de aspersão simples estaria na falta de homogeneidade na aplicação quando comparado a sistemas como o de pivô central, entretanto, seu preço é muito mais acessível. A imagem da Figura 2.1 mostra a foto da utilização do sistema de aspersão simples.

Figura 2.1: Foto de uma fazenda com sistema de Aspersão simples.



Fonte: CANAL DO PRODUTOR,2015

2.1.2 Sistema por gotejamento

O sistema de irrigação por gotejamento é considerado um dos mais homogêneos na aplicação e consiste na aplicação de canos próximos às raízes das mudas a fim de possibilitar um solo devidamente úmido. O grande obstáculo desse tipo de sistema é sua limitação quanto a escala, inviabilizando sua utilização em fazendas de grande porte.

O sistema de gotejamento não requer técnicas sofisticadas de manejo de irrigação, somente manutenção, pois sua estrutura permite que a terra próxima as raízes das planta fiquem úmidas constantemente.

Figura 2.2: Ilustração representativa de um sistema de gotejamento



Fonte – CONSTRUINDO, 2011

2.1.3 Sistema de pivô central

“O pivô central de irrigação consiste em vários segmentos de tubo (aço ou alumínio normalmente galvanizado) unidas e apoiadas por treliças, montados em torres de rodas com aspersores posicionados ao longo do seu comprimento” (MADER, 2010). A imagem da Figura 2.3 mostra um pivô central em funcionamento. “A máquina se desloca em um círculo e seu abastecimento de água é feito a partir do ponto de articulação no centro. [...] um pivô central tem tipicamente 400 m comprimento (raio do círculo)”. (MRS, 2008). Na Figura 2.4 é possível perceber a bomba do sistema de pivô central.

Figura 2.3: Sistema de pivô central em funcionamento



Fonte - VORIES, 2010, p. 129

Figura 2.4: Vista próxima do sistema de pivô central da empresa Lindsay.



Fonte – VORIES, 2010, p. 130

2.2 Metodologias agrônômicas para controle da umidade do solo

Dos sistemas acima citados somente dois: pivô central e aspersão simples necessitam de uma metodologia apropriada para a decisão da irrigação.

Apesar de muitas técnicas de manejo de irrigação serem já conhecidas, a falta de praticidade as torna pouco utilizadas por muitos agricultores que acabam tomando de forma empírica a decisão irrigar ou não suas lavouras.

A seguir estão elencadas as principais metodologias de obtenção de umidade do solo praticadas hoje.

2.2.1 Sensoriamento remoto

A possibilidade de medição da banda infravermelha em imagens de satélite deu origem a utilização da prática da inércia termal, “*Thermal Inertia*” (TI) em inglês. “A TI relaciona SMC com as magnitudes das diferenças de temperaturas máximas e mínimas medidas no solo no período de um dia” (IDSO; JACKSON; REGINATO, 1976). Uma das dificuldades da utilização da TI está na necessidade de obter características do solo. Verstraeten, Veroustraete, Van Der Sande, Grootaers e J. Feyen (2006), comentam sobre utilização do método de TI:

[...] é simples porque usa o conhecimento das propriedades físicas do solo e do clima e pode produzir perfis de SMC na profundidade de 1 metro. A limitação da sua utilização é que a sensibilidade da incerteza das propriedades físicas do solo é complexa de determinar no campo e normalmente estas são obtidas com medidas pontuais.

Segundo Carlson, Gillies e Schmugge, (1995) uma das grandes causas de erro gerado no método de TI é a quantidade de vegetação na área. O *Vegetation Index* (VI) é uma medida que estima a quantidade de vegetação na área e pode ser utilizada para uma melhor estimativa a partir da sua relação temperatura de superfície (T_s). Um dos problemas da T_s / VI é que “[...] um número suficientemente grande de pixels precisa ser amostrado, essa limitação faz o método ser deficiente no que diz a aplicação em escalas menores, como o tamanho de uma fazenda típica” (KALEITA; TIAN; HIRSCHI, 2005).

Sistemas que utilizam a tecnologia de *Drones* e aviões tripulados para obtenção de imagens mais próximas, e logo cálculos mais precisos da umidade do solo, também tem sido alvo recorrente de diversos estudos. Estas técnicas, entretanto, segundo Capolupo (2015) possuem sua aplicabilidade reduzida por efeito da necessidade de um operador qualificado, custos envolvidos e performance deficiente, quando o sistema não é calibrado eficientemente.

A utilização de imagens de satélite também tem embasamento no que se refere a previsão de chuvas, sendo esse o fator que mais influencia a umidade do solo em zonas de sequeiro. A teoria de (LORENZ, 1973), demonstra que as componentes das forças internas do sistema atmosférico podem, potencialmente, ser previstas com

antecedência de até cerca de 14 dias, assim, não é possível fazer previsões de tempo para meses ou anos no futuro, no sentido de obter uma sequência correta de eventos atmosféricos, dia após dia. “Essa limitação é determinada pela taxa de crescimento dos inevitáveis erros na estimativa inicial do estado da atmosfera” (STERN e EASTERLING, 1999). Esta definição também influencia as diferenças de previsão de tempo e clima.

2.2.2 Medição *in situ*

A obtenção do SMC localmente é de extrema importância tanto para a calibração de técnicas remotas, como para a realização de medições em tempo real. Uma lista abrangente de metodologias de obtenção de umidade do solo *in situ* tais como suas aplicações e características pode ser obtida em De Souza (2006).

É perceptível a linha tênue entre confiabilidade dos resultados e complexidade de implementação dos sensores. Mesmo os métodos mais confiáveis como: atenuação gama, fluxo de calor no solo, refletância no espaço tempo e bomba de nêutrons são caros e necessitam de um processamento complexo devido à grande incidência do ruído existente.

O método utilizado no sistema proposto necessita da permanência do sensor no solo por um longo período. Entretanto, riscos de furto e a necessidade de um custo menor para viabilizar uma aplicabilidade prática, limitam a utilização de sensores mais onerosos. Dentro deste universo estão métodos como acetileno, capacitância elétrica, variação de temperatura e tensiométrico.

2.3 Aprendizado de máquina

O aprendizado de máquina também conhecido “*Machine Learning*” é o nome dado ao campo da inteligência artificial responsável por desenvolver algoritmos que possam melhorar seu desempenho especificamente para grupo de dados de interesse. O ato chamado de treinamento permite o algoritmo classificado como de aprendizado a moldar suas características para a base de dados em questão

O campo do aprendizado de máquina se divide em dois principais campos, sendo eles o aprendizado supervisionado e o aprendizado não supervisionado.

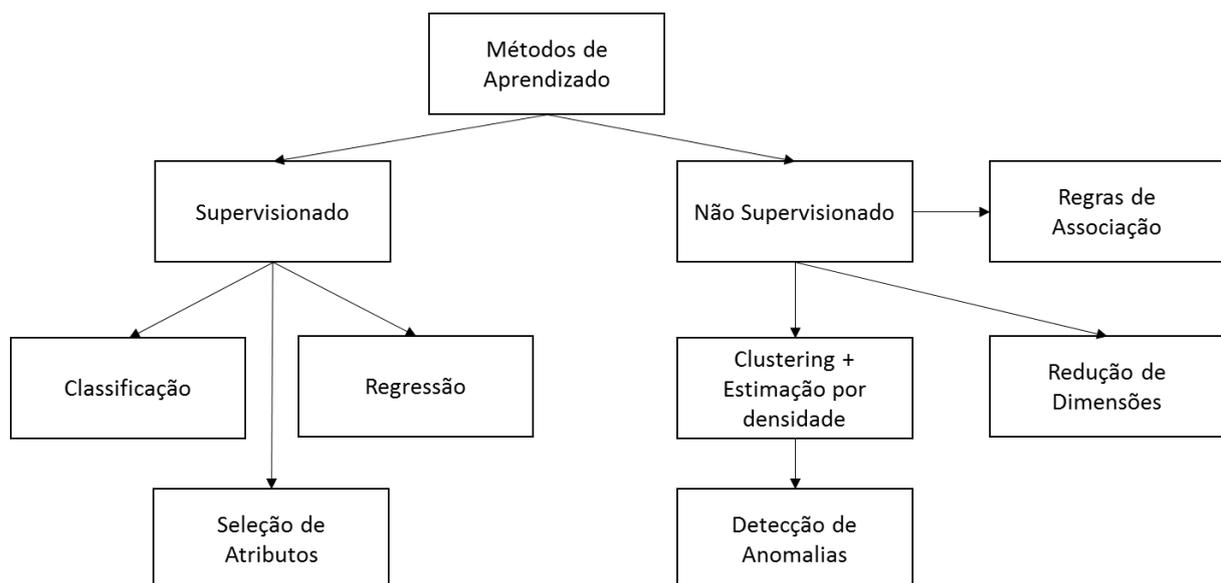
O aprendizado supervisionado é caracterizado pela existência de um conjunto de dados denominados “*Targets*” (Alvos). Cada dado de entrada do conjunto de

treinamento consistem um par com um objeto de entrada (tipicamente um vetor) e seus alvos de saída. Um algoritmo de aprendizagem supervisionada, analisa os dados de treino e produz uma função, que pode ser utilizado para o mapeamento de novos exemplos. Como pode ser observado na Figura 2.5 o aprendizado supervisionado permite resolver problemas como classificam de elementos, regressão de funções e seleção de atributos.

O aprendizado não supervisionado consiste em realizar o processo de treinamento a partir do conjunto de dados observado, não existindo dados alvo. Um exemplo que pode representar a utilização do aprendizado não supervisionado é a classificação de objetos somente por sua cor e tamanho, não contendo informações sobre classificações precedentes.

Aplicações clássicas de aprendizagem não supervisionada, consistem na separação de elementos a partir de agrupamento de conjuntos semelhantes.

Figura 2.5: Representação gráfica das classificações de algoritmos de I.A.



Fonte – Traduzido de (HOSPEDALES, 2013)

2.3.1 Redes neurais

Uma rede neural é caracterizada por estruturas que são inspiradas no funcionamento da capacidade adaptativa dos neurônios existentes em seres vivos.

Agatonovic-Kustrin e Beresford (2000) comentam que redes neurais artificiais (ANN) capturam seu conhecimento na detecção de padrões e relações nos dados, bem como no aprendizado a partir da experiência, ao invés de uma programação pré-existente. Inexiste um modelo formalmente definido e aceito para o conceito de uma rede neural, todavia uma classe de modelos estatísticos pode ser considerada neural uma vez que possua duas características:

1 - É constituída por series de parâmetros numéricos que são estimados por um algoritmo de aprendizado;

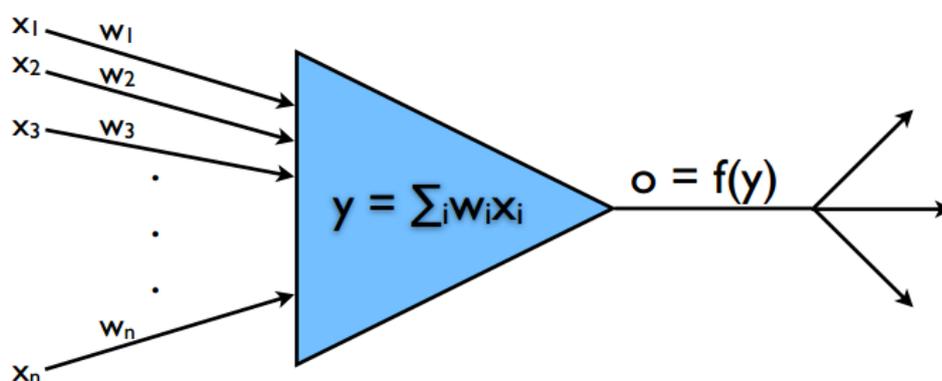
2 - São capazes de aproximar funções não lineares e não linearmente separáveis de suas entradas.

Um dos modelos mais difundidos e estudados na área é o “*perceptron*”. O “*perceptron*” é caracterizado pela aproximação matemática simplificada do funcionamento de um neurônio. Esta representação é demonstrada pela Figura 2.6.

Cada perceptron possui um número suficiente de entradas para sua aplicação e uma única saída $f(y)$ que segue a soma ponderada de cada um de seus elementos de entrada.

Cada elemento W_k representado na Figura 1.3.1 representa os pesos que cada entrada X_k em relação a única saída $f(y)$.

Figura 2.6: Modelo representativo de um perceptron utilizado em uma rede neural



Fonte – WIGGINS, 2013.

Matematicamente o perceptron da Figura 2.6 pode ser representado pela Equação 2.1 e Equação 2.2. Onde o representa a saída e $g(o)$ a função sigmoide que é definida pela Equação 2.3.

A função sigmoide, pode ou não estar presente na saída da rede neural definida. Para casos onde é necessária uma saída contínua, como em problemas de regressão, a função sigmoide não aparece na saída da rede neural. Já para problemas de classificação a função sigmoide deve ser utilizada.

$$o = f(X.W) \quad (2.1)$$

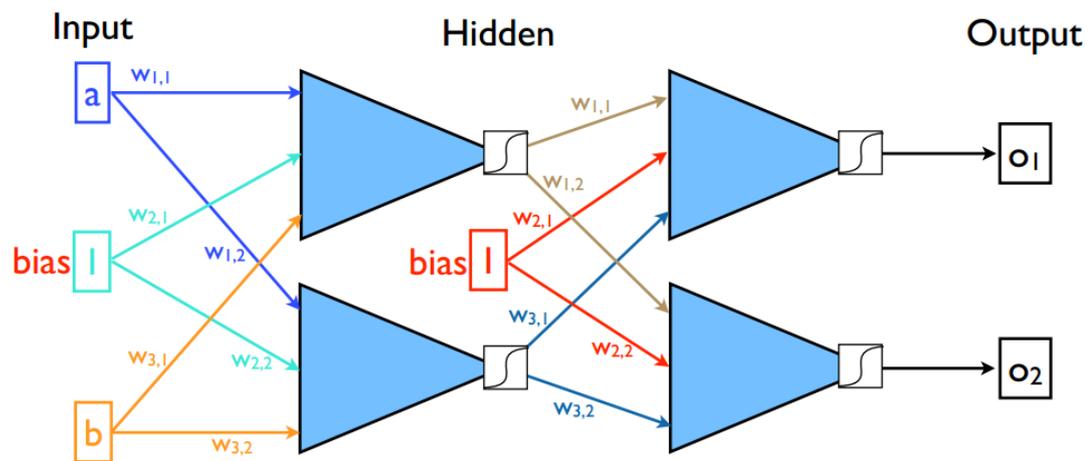
$$X.W = \sum_{i=0}^n w_i x_i = w_1 . x_1 + w_2 . x_2 + \dots + w_n . 1 \quad (2.2)$$

$$g(o) = \begin{cases} o \geq 0.5 = 1 \\ o < 0.5 = 0 \end{cases} \quad (2.3)$$

Um único “*perceptron*” pode ser utilizado na resolução de problemas matemáticos simples. O modelo *Perceptron* historicamente sofreu críticas, como na obra de Minsky e Seymour (1969) que mostraram que o “*perceptron*” não conseguiria representar a função “OU Exclusivo” (*XOR*). Em Rumelhart, Hinton e Williams (1985) publicaram o algoritmo *Backpropagation* possibilitando que redes de perceptrons conectados resolvessem o problema *XOR*, desde então as ANN são algoritmos conceituados mundialmente. A Figura 2.7 demonstra o exemplo de modelo de rede neural baseado na associação de quatro perceptrons, nota-se que este exemplo possui funções sigmoides em sua camada de saída, caracterizando assim um sistema de regressão.

Na opinião de Lee, Battle, Raina e Ng, redes neurais podem ser utilizadas em problemas que necessitem de soluções do tipo *Supervised Learning*, ou seja, sistemas que se baseiam em algum conhecimento prévio ou treinamento para obter a resposta de alguma entrada. Utilizando-se a relação de entradas e saída é possível treinar a rede neural a partir do algoritmo de *Backpropagation*.

Figura 2.7: Modelo com quatro perceptrons associados, duas entradas e duas saídas.



Fonte - WIGGINS, 2013.

O algoritmo de *Backpropagation* é utilizado para o treinamento da rede neural. Com os dados desejados dos *Targets* na saída, realiza-se a contra propagação afim de minimizar o erro em cada neurônio envolvido pela rede. Uma descrição abrangente do algoritmo *Backpropagation* não é o foco deste trabalho, tal algoritmo, entretanto, consiste em 4 passos.

- 1 - Obter a saída do sistema não treinado para uma entrada com resposta alvo (*Target*) conhecida;
- 2 - Obter a diferença entre a resposta alvo e a resposta obtida na saída da rede neural;
- 3 - Contra propagar a diferença e modificar os pesos até esta ser minimizada por um algoritmo de minimização;
- 4 - Repetir.

Tal procedimento é suficiente para redefinir os valores dos pesos W_k de cada “*perceptron*” definidos na Figura 2.7.

Levenberg-Marquardt como algoritmo de minimização de erros

A necessidade de minimização de erros no algoritmo de *Backpropagation* torna necessário o uso de algum algoritmo que faça tal. Nessa seção será realizada a dedução do método de minimização Levenberg-Marquardt.

O problema que o algoritmo LM soluciona se denomina Minimização não linear de mínimos quadrados. Isso implica a função ser minimizado no tipo da forma da Equação 2.4. Onde x é um vetor, e cada r_j uma função residual assumindo que $m \geq n$. f pode ser representado como o vetor residual $r: \mathfrak{R}^n \rightarrow \mathfrak{R}^m$.

$$f(x) = \frac{1}{2} \sum_{j=1}^m r_j^2(x) \quad (2.4)$$

Reescrevendo f encontra-se o Equação 2.5. As derivadas de f se igualam utilizando a matriz Jacobiana na Equação 2.6.

$$f(x) = \frac{1}{2} \|r(x)\|^2 \quad (2.5)$$

$$J(x) = \frac{\partial r_j}{\partial x_i}, 1 \leq j \leq m, 1 \leq i \leq n. \quad (2.6)$$

Considerando um caso onde toda função r_i é linear. A jacobiana é constante, partindo da Equação (2.5) $f(x)$ pode ser obtido pela Equação 2.7, adicionalmente, obtém-se as equações do caso geral que são representadas na Equação 2.8.

$$f(x) = \frac{1}{2} \|J_x + r(0)\|^2 \quad (2.7)$$

$$\nabla f(x) = J^T (J_x + r) \therefore \nabla^2 f(x) = J^T J \quad (2.8)$$

A então desejada função de minimização é obtida igualando $\nabla f(x)$ a 0. O valor mínimo da variável x resulta na Equação 2.9. Retornando ao caso geral obtém-se as Equações 2.10 e 2.11.

A parte que difere os problemas de mínimos quadrados é que dada uma matriz Jacobiana J é possível obter a matriz Hessiana ($\nabla^2 f(x)$) facilmente dada a circunstância onde os resíduos quadráticos $\nabla^2 r_j(x)$ são relativamente pequenos. A Hessiana neste caso torna-se da forma da Equação 2.12.

$$x_{min} = -(J^T J)^{-1} J^T r. \quad (2.9)$$

$$\nabla f(x) = \sum_{j=1}^m r_j(x) \nabla r_j(x) = J(x)^T r(x) \quad (2.10)$$

$$\nabla^2 f(x) = J(x)^T J(x) + \sum_{j=1}^m r_j(x) \nabla^2 r_j(x) \quad (2.11)$$

$$\nabla^2 f(x) = J(x)^T J(x) \quad (2.12)$$

Aplicação da metodologia de Validação Cruzada e prevenção de sobre-ajuste

Nas palavras de Devijver, Pierre, Kittler e Josef (1982, p. 357): “A Metodologia de validação cruzada é uma técnica de validação de modelos que permite generalizar modelos preditivos e estimar sua performance na prática”. A etapa de validação permite ao sistema de criar uma curva que consiga generalizar um grande conjunto de dados e represente somente seu conjunto de treino (*sobre-ajuste*).

O método *Holdout* divide a base de dados em dois subconjuntos mutuamente exclusivos, um para treinamento (estimação dos parâmetros) e outro para teste. “Uma proporção muito comum é considerar 2/3 dos dados para treinamento e o 1/3 restante para teste” (KOHAVI, 1995).

Enquanto o método *Holdout* permite realizar o teste de performance para uma rede treinada, O método de validação cruzada permite dividir a base de dados mais uma vez, criando o conjunto de validação.

O conjunto de validação permite realizar o treinamento com partes diferentes do conjunto de treino e validar sua performance para cada um desses subconjuntos. Desta forma o algoritmo com melhor performance deverá ser aquele que melhor desempenha para todos os casos validados.

2.3.2 Rede neural recorrente e rede NARX

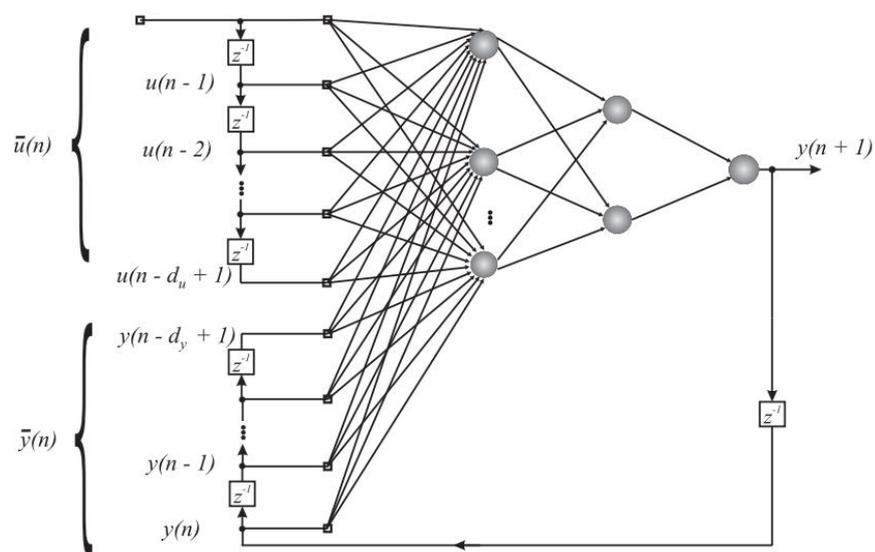
Uma rede neural recorrente é de interesse quando medições passadas possuem correlação com os resultados futuros de uma amostra. No caso da umidade do solo, a variabilidade no tempo e a retenção cumulativa de água tornam necessário um método que considere a posição temporal das variáveis de entrada.

Redes neurais recorrentes utilizam o conceito de atraso. Quando uma variável é exposta ao atraso, na prática o que se realiza é uma referência a N posições anteriores em seu conjunto de saída. Siegelmann, Horne e Giles (1997) comentam sobre a capacidade das redes neurais recorrentes que são consideradas máquinas computacionalmente ricas e podem ser consideradas pelo menos tão poderosas quanto máquinas de Turing.

ANN's recorrentes podem possuir realimentação em todos seus neurônios independentemente da camada. A complexidade de um sistema com realimentação em diversos elementos, torna este excessivamente complexo, uma solução possível são as redes NARX.

A rede Neural baseada em modelos não lineares auto regressivos com entradas exógenas se denomina Rede NARX. Essas redes, ao contrário das recorrentes, são limitadas a um único *feedback* de seu neurônio de saída ao invés de realimentação dos neurônios da sua camada escondida.

Figura 2.8: Representação de uma rede NARX



Fonte - MENEZES, p. 3336, 2008

Em teoria modelos NARX podem ser utilizados ao invés de redes neurais recorrentes sem perda significativa de desempenho mesmo com sua retroalimentação sendo limitada” (SIEGELMANN; HORNE; GILES; 1997, p. 208). Matematicamente a uma rede NARX pode ser representada pela Equação 2.13. Aplicando-se um deslocamento tempo, é possível obter um comportamento preditivo para o algoritmo. A Equação 2.14 demonstra o comportamento observado na Figura 2.8

$$y(t) = f(u(t - n_u), \dots, u(t - 1), u(t), y(t - n_y) \dots, y(t - 1)) \quad (2.13)$$

$$y(t + 1) = f(u(t - n_u + 1), \dots, u(t), u(t + 1), y(t - n_y + 1) \dots, y(t)) \quad (2.14)$$

A rede NARX considera a utilização de novas entradas simultaneamente à utilização da retroalimentação da saída tornando o algoritmo, tornando-a assim ideal para a aplicação deste trabalho.

3 METODOLOGIA EXPERIMENTAL

A metodologia de pesquisa para este trabalho tem como meta estruturar o algoritmo proposto. Neste sentido, é necessário a obtenção de dados confiáveis e históricos sobre medidas de umidade do solo e previsão do tempo.

Para a realização do estudo de previsão de umidade do solo deve-se obter uma base de dados passada com medições de variáveis do campo afim de simular o desempenho deste em tempo real. A escolha e manejo destas bases são abordados nas seções 3.1 e 3.2 respectivamente. Na Seção 3.3 e 3.4 se realizará a estruturação de um algoritmo e a escolha das variáveis de interesse. No final da seção se abordará a construção de um programa em linguagem orientada ao objeto.

3.1 Escolha do campo de estudos e da base de dados climáticos

3.1.1 Campo de estudos

Devido a necessidade da obtenção variáveis de previsão controladas para a realização do experimento é fundamental a seleção de um campo de testes que permita o estudo simultâneo das variáveis selecionadas. Experimentos sobre a umidade de solo foram realizados em diversas localidades nas últimas décadas. Os experimentos mais relevantes são expostos na Tabela 1

Tabela 1: Lista dos experimentos relevantes de umidade do solo

Washita '94	SMEX05
SGP97	CLASIC - Land
SGP99	SMAPVEX08
SMEX02	CanEx-SM10
SMEX03	SJV10
SMAPVEX12	

Fonte- HYDROLOGY AND REMOTE SENSING LABORATORY, 2013

Na maioria dos estudos listados na Tabela 1 o objetivo principal foi a criação de parâmetros de calibração para o satélite SMAP da NASA. Este experimento consistia em obter os parâmetros do solo para a utilização posterior pelo satélite.

SMAPVEX12

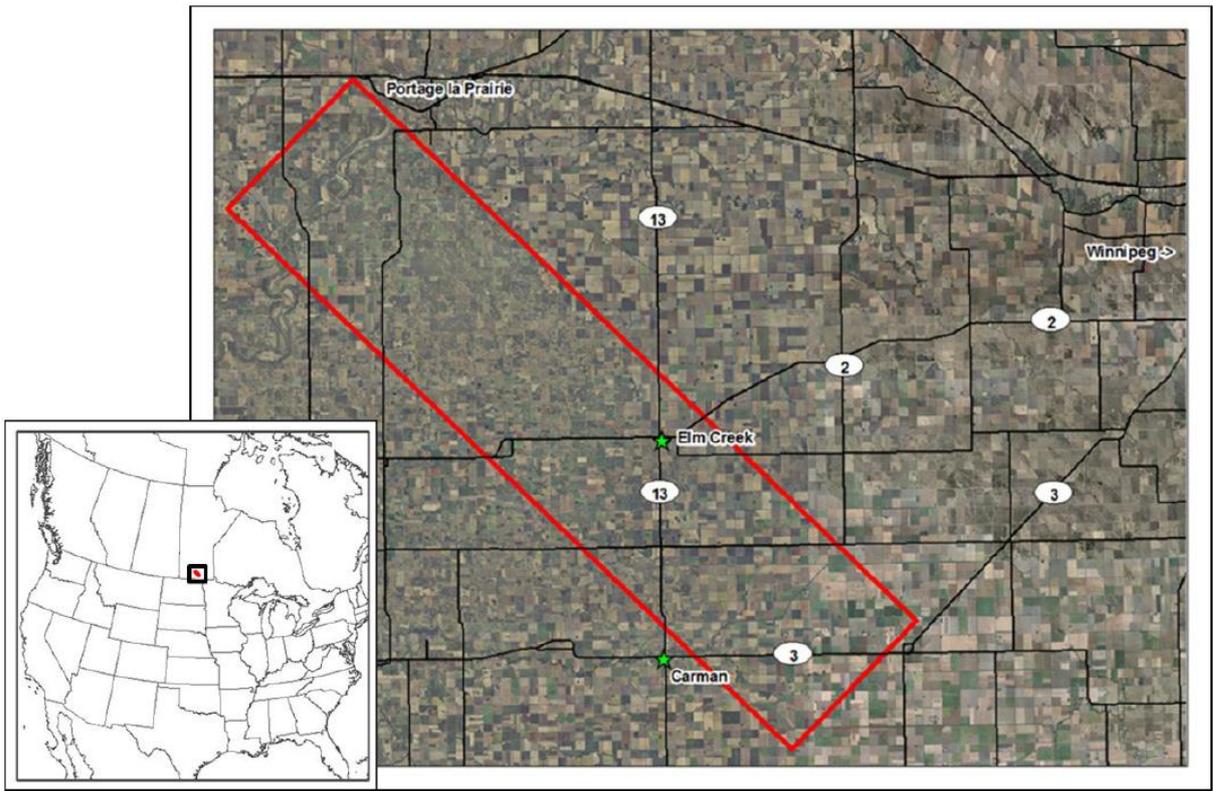
A área de estudo do SMAPVEX12 é localizada na região centro-sul de Manitoba, Canadá, na cidade de Elm Creek (98° 0' 23" O, 49° 40' 48" N). O estudo foi realizado em 2012 como parte do programa de lançamento do satélite SMAP, promovido pela Agência espacial Norte Americana (NASA) em 2015.

O estudo compreendeu áreas de plantação agrícola e Florestas, estas últimas não sendo relevantes ao estudo deste trabalho.

Dentre as razões para a escolha do SMAPVEX12 como base de dados estão a disponibilidade da base de dados e a obtenção de diferentes variáveis sendo estas:

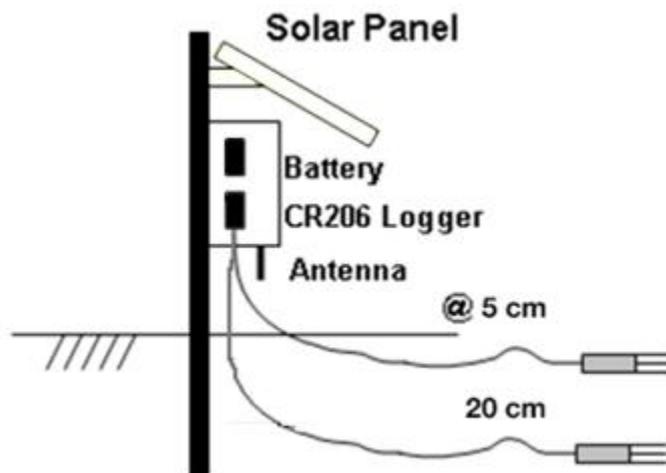
- Existência de estações meteorológicas próximas aumentando a confiabilidade de previsões climáticas.
- Rede de sensores de umidade do solo instalados por tempo maior (reduz a possibilidade erros devido a movimentos na terra) com medições de umidade do solo em diferentes profundidades.
- Registro dos estágios de desenvolvimento de planta nos campos agrícolas estudados.
- Imagens aéreas e de satélite fazendo-se possível obter o índice VI.

Figura 3.1: Imagem de satélite e mapa de localização do experimento SMAPVEX12



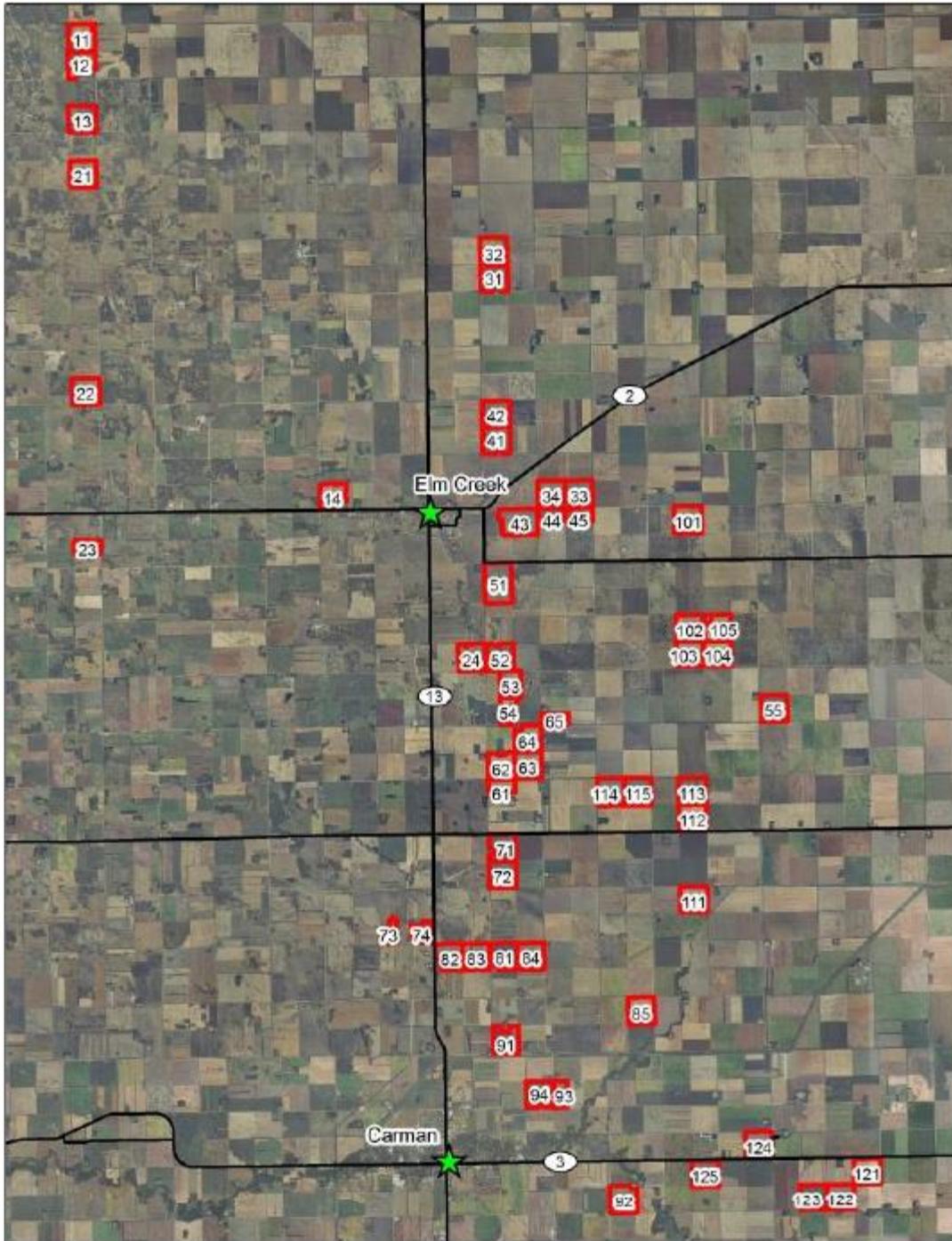
Fonte- SMAPVEX12, 2012, p. 5

Figura 3.2: Representação dos sensores da rede de sensores no estudo SMAPVEX 12.



Fonte- Modificado pelo autor (SMAPVEX12,2012, p. 14).

Figura 3.3: Foto representativa de satélite dos campos agrícolas do experimento SMAPVEX 12



Fonte- SMAPVEX12, 2012, p. 7

3.1.2 Escolha do modelo de previsão do tempo

A fim de obter um modelo assertivo para a previsão de precipitação foi feita uma pesquisa na busca pela base de dados com a melhor resolução e com um confiável sistema de previsão do tempo. A partir do centro de dados climáticos nacionais dos estados unidos disponibilizado pela *National Oceanic and Atmospheric Administration* (NOAA) é possível obter acesso a base de dados históricas de modelos numéricos de previsão meteorológicas (NWP). Nesse tipo de previsão a observação do tempo serve de entrada para cálculos que proveem saídas numéricas com uma gama de horizontes de previsão. Os modelos têm variabilidade no que se refere a sua resolução em área, número de medições diárias, tempo máximo de previsão além de áreas que cobrem.

O acesso aos bancos de dados também deve ser avaliado, uma vez que estes se utilizam de protocolos variados de acesso. Dentre os modelos selecionados para o estudo, estão:

Global Ensemble Forecast System (GEFS)

O GEFS é um modelo de abrangência global realizado a partir de 21 sistemas de previsão diferentes, é produzido quatro vezes por dia e que pode prever até 16 dias na frente. É dividido entre modelos (curto prazo) de até 18 horas de previsão e médio prazo de 18h até 16 dias. A resolução é de 1 grau de longitude e 1 grau de latitude. Ele é usado também para medir a quantidade de incerteza de modelos de previsão

Global Forecast System(GFS)

É um modelo de abrangência global que utiliza 4 modelos para sua previsão. Consta com uma resolução de até 28km. Também é medido 4 vezes ao dia e pode prever até 16 dias. É comumente utilizado também para previsão de clima a partir do chamado *reforecast* que chega a prever 9 meses no futuro para médias mensais de clima.

North American Mesoscale (NAM)

Considerado um dos mais confiáveis sistemas de previsão do tempo. Funciona somente na América do Norte e é medido 4 vezes ao dia. Um dos grandes diferenciais é a alta resolução de suas previsões com intervalos de 12km. Um dos lados negativos do NAM é sua capacidade de prever somente 84 horas.

3.2 Obtenção das bases de dados do estudo e manejo dos dados

As bases de estudo para o modelo foram obtidas a partir dos sites responsáveis pelo armazenamento. Uma vez que não existe um banco de dados consolidado para o experimento, foi necessária uma busca extensiva pelas diferentes fontes de informação e metodologias para o manejo dos diferentes tipos de dados.

Previsão de precipitação e Temperatura

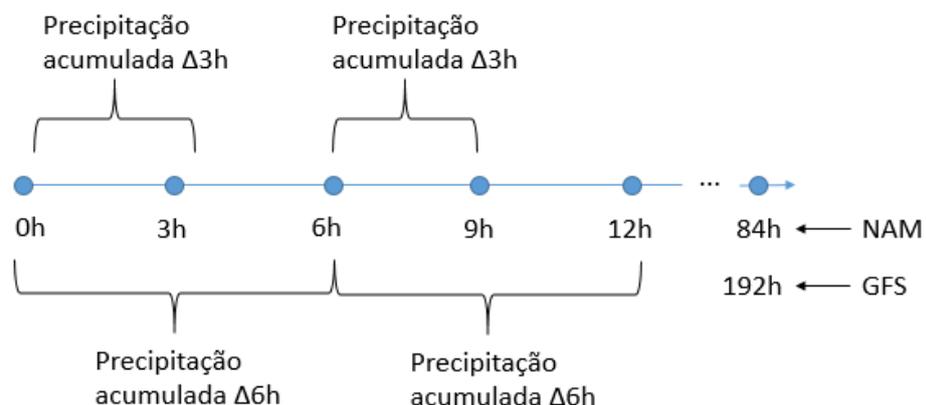
A obtenção de dados para períodos antes de 2013 deve ser requisitado por e-mail a partir do *website* NOMADS (RUTLEDGE,ALPERT e EBUISAKI , 2006) da agência americana NOAA.

A partir da requisição, foi obtida a base de dados contendo a previsão do tempo medidas respectivamente às 0, 6, 12 e 18 horas dos dias 10/06/2012 até 13/07/2012 para a área de interesse no campo de estudos.

Uma vez obtido o acesso para o período de interesse, arquivos no formato de base dos dados podem ser acessados pelo software Matlab Math Works versão 2013b a partir da função NcRead. Foi realizado a programação de forma iterativa de uma função para o manejo do banco de dados. Esta função é demonstrada de forma explícita no **Anexo A**.

A previsão de dados dos modelos de NWP utilizados são disponibilizados em intervalos futuros de 3 em 3 horas, onde intervalos múltiplos de 6 correspondem ao cenário de previsão de precipitação acumulada nas últimas 6 horas até o horário de referência e múltiplos de 3 e não de 6 são a precipitação acumulada nas últimas 3 horas.

Figura 3.4: Representação do acúmulo de precipitação dos modelos GFS e NAM



Todo intervalo de previsão de precipitação deve ser considerado e logo, para manter a coesão com os dados de previsão do tempo todas às variáveis de entrada deverão ser agrupadas em intervalos de seis horas. De forma a utilizar o máximo a confiabilidade do sistema NAM quanto do GFS será considerado um período de previsão de até 84 horas para o sistema, possibilitando o acesso as duas bases de dados.

Umidade do solo dos sensores permanentes

A AAFC (Agriculture and Agrifood Canada) instalou em 2011 sensores de solo dentro e próximo da região do experimento (SMAPVEX12, 2012). O sistema consiste em nove estações de monitoramento distribuidas para capturar a variação em diferentes classes de solo e logo a sua umidade relativa. A Tabela 3.3.1.1. mostra a referência do número dos sensores na imagem da Figura 3.1.1.1

Tabela 2: Referência dos campos AAFC e MAFRI

AAFC Estações <i>in Situ</i>	Código Campo SMAPVEX12
AAFC -1	92
AAFC -2	93
AAFC -3	73
AAFC -4	65
AAFC -5	24
AAFC -7	44
AAFC -9	32

Fonte- SMAPVEX12, 2012, p. 11

Para cada estação três sensores de umidade do tipo Stevens Hydra foram colocados em profundidades 5cm e 20cm, além disso, para obter uma melhor precisão na região de 5cm mais três sensores foram enterrados. Cada sensor capta umidade relativa do solo em valores percentuais (utilizando o modelo padrão de conversão dielétrica do equipamento), temperatura do solo em graus célcios e permissividade dielétrica (variável não utilizada neste trabalho). As estações também são equipadas com pluviômetros do tipo Campbell Scientific 700. Os dados são

logados, utilizando uma estação telemétrica adcon A755 que transmite os dados remotamente para uma base a cada 30 minutos. .

O site da AAFC disponibiliza os dados dos sensores permanentes para o período do experimento no formato de odm, foi necessária a conversão dos arquivos para o formato XLS interpretável no software Excel 365 da empresa Microsoft.

Somente os campos: 92, 93, 24 e 32 apresentaram bases de dados intactas para todo o período.

Precipitação medida in situ

Para uma melhor integração dos dados de precipitação existentes com os dados de previsão de precipitação será necessário obter dados de pluviômetros, localizados próximos às medições de temperatura e umidade do solo.

Foram observadas medições de 14 pluviômetros nas áreas próximas aos campos observados escolhidos a fim de minimizar o impacto da incerteza de medição de precipitação.

Lidando com perdas de dados

Para Hospadales (2013) a maioria dos algoritmos de aprendizado necessita o mesmo tamanho de variáveis de entrada e *targets* para a realização do treinamento.

Endereçamos este problema de forma simplista para a base de dados dos campos de estudo ao considerar apenas dados dos campos que não tenham sofrido perda de dados para o período de interesse. Entretanto, como informam Rutledge, Alpert e Ebuisaki (2006), a base histórica dos modelos de previsão do tempo possuem dados faltantes. Para Smith (2007, p. 117) “Se um sistema é não linear e realimentado, o erro deve crescer exponencialmente com o tempo, pois corrigir em tempo de programa aumenta exponencialmente a ordem do algoritmo” .

Soluções são abordadas por Hospadales (2013):

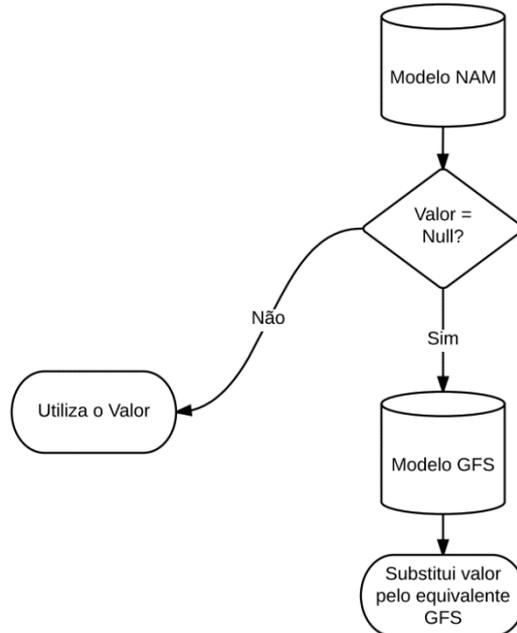
1. Excluir a linha de dados faltantes.
2. Tratar os dados como uma categoria especial. Exemplo: Substituir valores nulos por um número de valor muito inferior ao valor mínimo da amostra.
3. Substituir dados faltantes pela média da amostra.

4. Utilizar uma árvore decisória para estipular valores adequados, baseados em variáveis correlatas.

5. Estruturar um algoritmo classificatório não supervisionado para classificar os dados faltantes

O método 1 não pode ser utilizado devido a dependência temporal do modelo. O método 2 poderá ser utilizado somente em casos onde não existirem outros métodos disponíveis. Alternativas 3 e 5 não podem ser utilizadas pela falta de dimensões de interesse adicionais. A alternativa 4, todavia, torna-se possível pois existe a disponibilidade de outros modelos de previsão do tempo. Caso haja necessidade será possível realizar a substituição dos dados faltantes de um modelo climático por dados medidos no mesmo período temporal para a mesma localização espacial. A Figura 3.5 ilustra o processo decisório para o modelo de previsão NAM.

Figura 3.5: Esquemático do processo decisório para manejo de dados faltantes.



3.3 Análise das variáveis de entrada para o modelo

Para encontrar um nível ótimo de performance de qualquer máquina de aprendizado é necessário escolher entre diferentes estruturas. A arquitetura ideal para o sistema de SMC deve utilizar premissas precedentes de que as variáveis de treinamento possuem correlação com a variável de objetivo.

Modelagens diferentes podem ser deduzidas a partir de diferentes conjuntos de treinamento, porém, para uma construção eficaz e em linha com a realidade, deve-se considerar as variáveis que mais influenciem o resultado.

É relevante destacar que o trabalho em questão está adstrito à engenharia elétrica. Assim a seleção de variáveis para o modelo deixará de ser baseada a partir do estudo de modelos agrônômicos. Serão utilizadas variáveis mencionadas em trabalhos precedentes e que já possuem suas correlações provadas.

De acordo com (ZAMANN, 2010) “[...] variáveis climáticas tais como precipitação (deslocada no tempo), temperatura do ar próximo ao solo e o índice NDVI possuem uma grande correlação com o índice de umidade do solo. “

Das variáveis acima citadas pode-se separar entre as que possuem correlação positivas e negativas com o SMC a ser medido no modelo proposto.

Tabela 3.2.1: Variáveis de interesse para o modelo

Variável	Correlação
Precipitação Acumulada	+
Nível de Umidade do Solo	+
Temperatura do Solo	-
NDVI	-

NDVI

O método T_s/V_i referido na Seção 2.2.1 necessita de estudos pré estabelecidos das variáveis do solo, que não serão de interesse a estimativa para a modelagem. O NDVI, todavia, não requer a estimativa destes parâmetros e baseia-se apenas nas bandas eletromagnéticas medidas de forma remota. “O NDVI é calculado utilizando as porções da energia eletromagnética refletida pela vegetação nas bandas vermelho

(comprimento de onda = 0,6 micrômetros) e Infravermelho próximo (comprimento de onda = 0,8 micrômetros)” (INSA, 2013).

O NDVI é um índice que retorna o estágio de crescimento da planta. Em uma rede neural, variáveis com menor variabilidade temporal, possibilitam um maior peso de novos treinamentos. Para o exemplo do crescimento da planta em sí, dados que foram capturados quando o tamanho da planta era menor, resultarão em um menor efeito na experimentação realizada para a planta mais desenvolvida, possibilitando ao modelo adaptar-se novamente a esta realidade.

O fato do campo de testes disponível possuir um período reduzido de dados, possibilita a menor consideração do NDVI como um fator relevante para o resultados dos testes que aqui serão realizados, todavia, é necessário considerar sua presença para a aplicação do algoritmo em sistemas onde a captura de dados dure mais tempo.

A consideração do NDVI como um fator negativo de correlação com a umidade de solo é correto devido ao fato do sistema não permitir que a planta sofra o estresse hídrico, assim a variação do NDVI representará somente o estágio de crescimento da planta e logo o quanto esta consome de água do solo.

Precipitação

A precipitação é uma variável atrasada em relação a umidade do solo, o acúmulo de água reflete em uma maior umidade, logo para uma modelagem correta do sistema deve-se considerar a influência histórica da precipitação.

Para uma modelagem acertiva, consideraremos 25 dias de histórico de chuvas separados em 8 períodos de 6 horas prévias.

Umidade do solo precedente

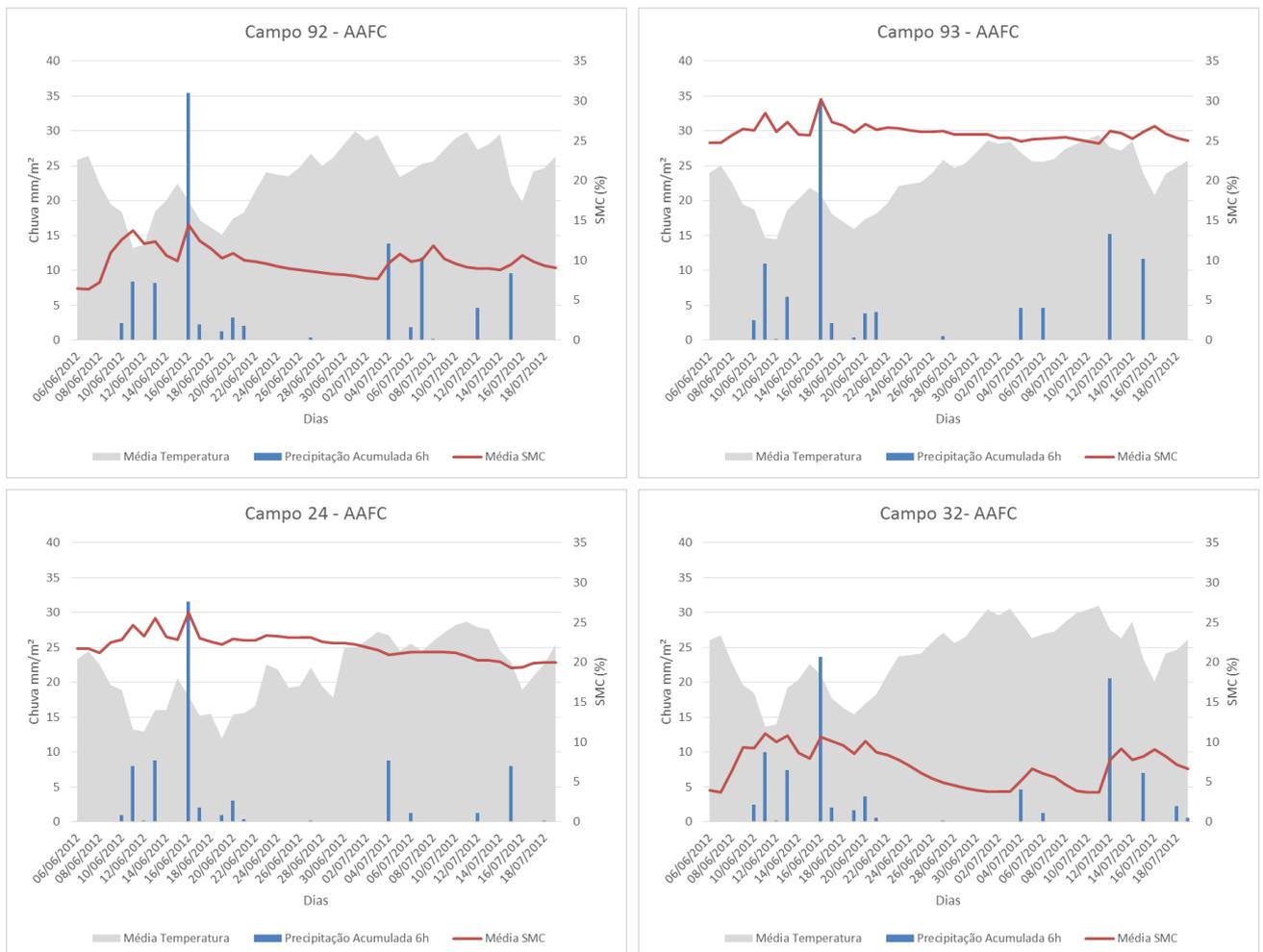
O SMC pode ser comparado a uma bateria, onde as variáveis com correlação positiva aumentam seu armazenamento e as com correlação negativa diminuem, devido a disponibilidade a partir dos sensores dos valores reais de umidade do solo é possível considerar o último valor medido como base da modelagem.

Temperatura

Penman (1948) Comenta como a temperatura do solo possui relação direta com o nível de evaporação deste e logo a variação do SMC. A evapotranspiração da planta (ETc), ou em outras palavras, utilização da água pela planta, é a soma da água removida do solo por evaporação na superfície do solo e por transpiração. De acordo com (SWR , 2009) aproximadamente de setenta a oitenta por cento da utilização da água pela cultura resulta da transpiração da planta. A variação da transpiração da planta varia principalmente com temperatura, umidade e radiação solar.

Para uma melhor obtenção de dados foram escolhidos campos dadas as seguintes condições: 1) Possuíssem diferentes culturas, 2) Não mostrassem perdas de dados no período do estudo, 3) Estivessem distantes entre si de solo e 4) Possuíssem níveis de medições em 5 cm e 20 cm.

Figura 3.6: Dados observados para os campos AAFC do SMAPVEX12.



Quatro campos com sensores da AAFC correspondiam a todas as condições e serão utilizados para a aplicação dos algoritmos. São estes os Campos 92, 93,24 e 32 respectivamente. Os gráficos da Figura 3.6 demonstram a variabilidade da SMC nos campos citados, também, nota-se que existe uma certa diferença entre as chuvas, mesmo de campos próximos, a temperatura, entretanto, varia menos. Em vermelho a Umidade do solo percentual, azul: mm/m³ de água precipitada acumulada a cada 6 horas e cinza: Média de temperatura a cada 6 horas em C^o.

3.4 Estruturação do algoritmo de aprendizado

Após feita a análise dos dados deve-se realizar a estruturação do algoritmo NARX.

Para realização de um modelo de previsão deve-se considerar a disponibilidade da previsão das variáveis futuras. Utilizando-se ambas bases dos modelo GSF e NAM de previsão de precipitação e temperatura, consegue-se estruturar o modelo NARX de laço fechado. Relacionando as variáveis citadas anteriormente e adicionando sua previsão obtem-se as variáveis listadas na Tabela 3.

Tabela 3: Composição de variáveis para o modelo preditivo

Variável	Intervalo de Medição
Previsão de Precipitação Acumulada	D até D + 84h
Nível de Umidade do Solo	D
NDVI	D
Previsão de Temperatura do Solo	D até D + 84h
Saída	D até D+ 84h

3.4.1 Atrasos nas entradas e no laço de realimentação

A Figura 3.4 demonstra a necessidade da separação das variáveis em intervalos de seis horas no tempo. Para a utilização de quatro dias no passado, deve-se utilizar dezesseis intervalos de seis horas. A estrutura com atrasos de dezesseis períodos para a entrada e para a saída é representada na Figura 3.7.

3.4.2 Estruturação das variáveis no algoritmo

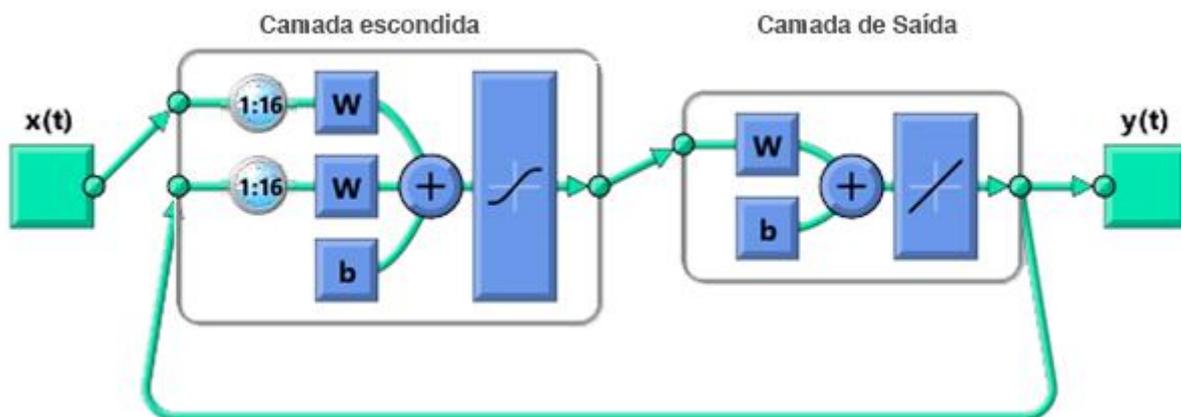
Após a seleção das variáveis o primeiro passo para estruturação de um algoritmo de aprendizado supervisionado é estabelecer a relação com seus *targets*.

Antes da implementação de um algoritmo preditivo, deve-se confirmar a eficiência das variáveis para o tempo presente. A representação da rede na

Figura 3.8 exemplifica a criação de um algoritmo que utiliza da precipitação acumulada dos dias passados para obtenção do SMC em tempo presente.

Conforme citado na seção, o NDVI não possui grande variabilidade para 30 dias de experimento por tratar-se de uma variável correlacionada com o crescimento e hidratação da planta. Como mencionam May, Dandy e Maier (2011), informações redundantes e com pouco correlacionadas podem diminuir o desempenho da rede neural, devido este fato, para na estrutura representada na Figura 3.9. realiza-se a modelagem do algoritmo NARX, sem utilizar o NDVI.

Figura 3.7: Representação da estrutura da rede neural com a camada escondida e atrasos nas entradas e saídas.



Utilizando algoritmo representado na Figura 3.9 para estimar a umidade do solo em 5cm, pode-se considerar a saída deste como valor de entrada para a previsão da precipitação à 20 cm. Na Figura 3.10 obtemos uma arquitetura possível para um algoritmo NARX recursivo em cadeia. Uma vez que a umidade do solo em profundidades mais rasas pode ser considerada um dos fatores mais relevantes para a estimativa de umidade do solo em terras mais profundas.

Figura 3.8: Representação da estrutura do algoritmo NARX com as variáveis de entrada analisadas.

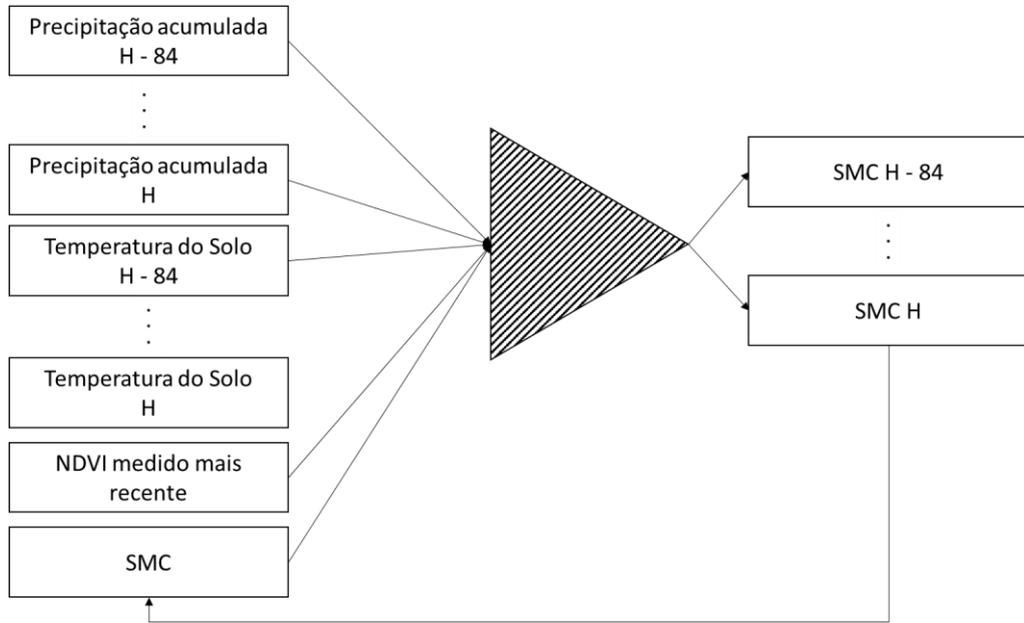


Figura 3.9: Representação da estrutura do Algoritmo NARX com as variáveis de entrada e desconsiderando a variabilidade do NDVI

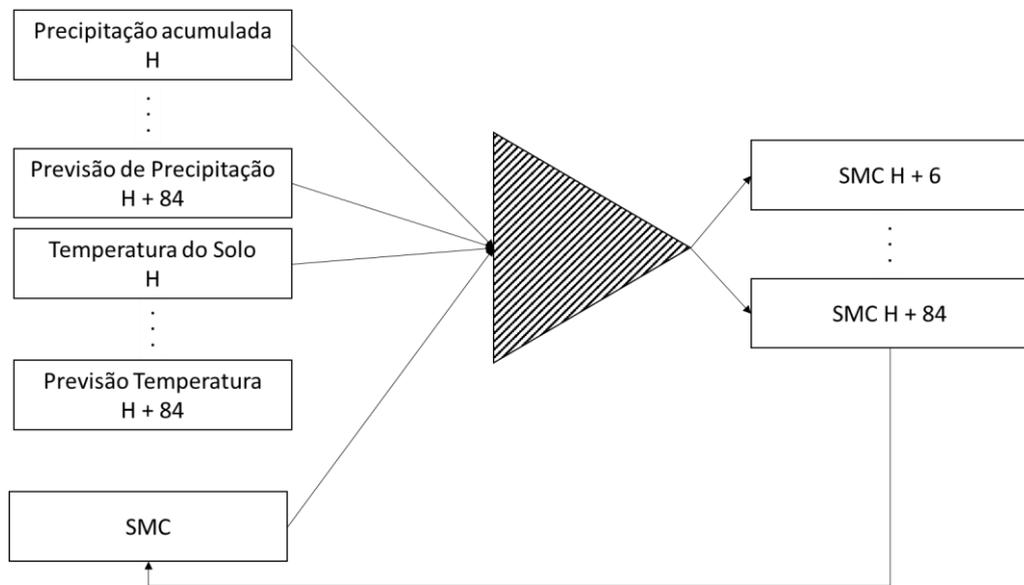
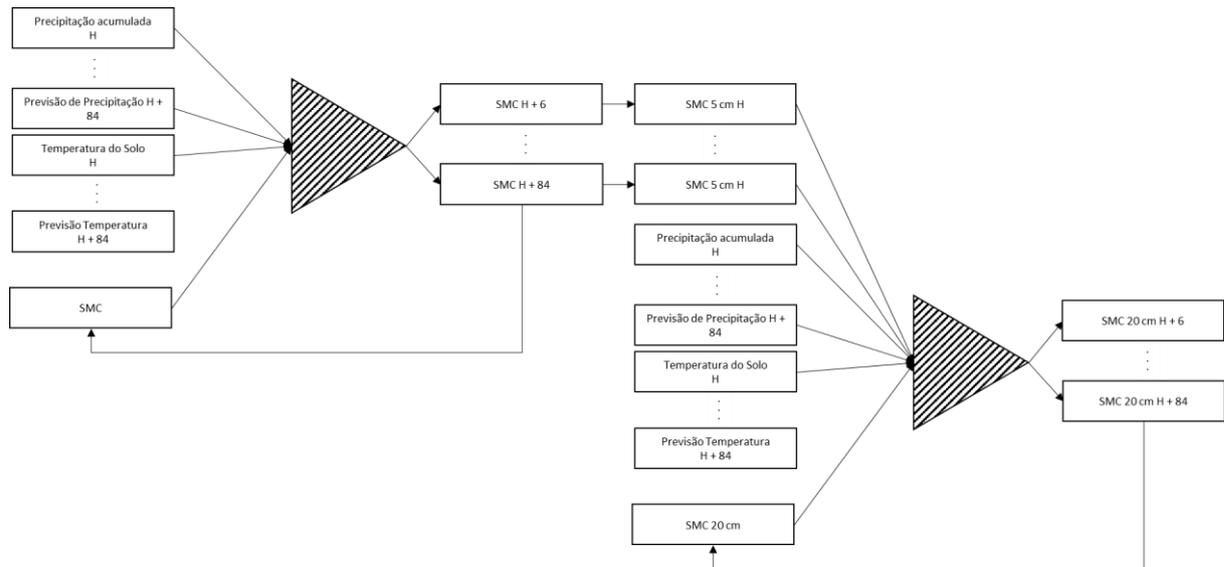


Figura 3.10: Representação da estrutura do Algoritmo NARX em cadeia para a previsão da umidade do solo à 20 cm de profundidade.



3.4.3 Número de neurônios na camada escondida

Muita discussão paira em torno do número de neurônios que devem existir na camada escondida de uma rede neural. Geman, Bienenstock, and Doursat (1992) comenta como diversas regras informais consideradas não possuem embasamento científico e o meio mais assertivo para estipular o número de elementos na camada escondida é simular diversas opções. Se realizará estimativas baseada na performance do algoritmo para diferentes arquiteturas.

3.4.4 Software utilizado e linguagem de programação

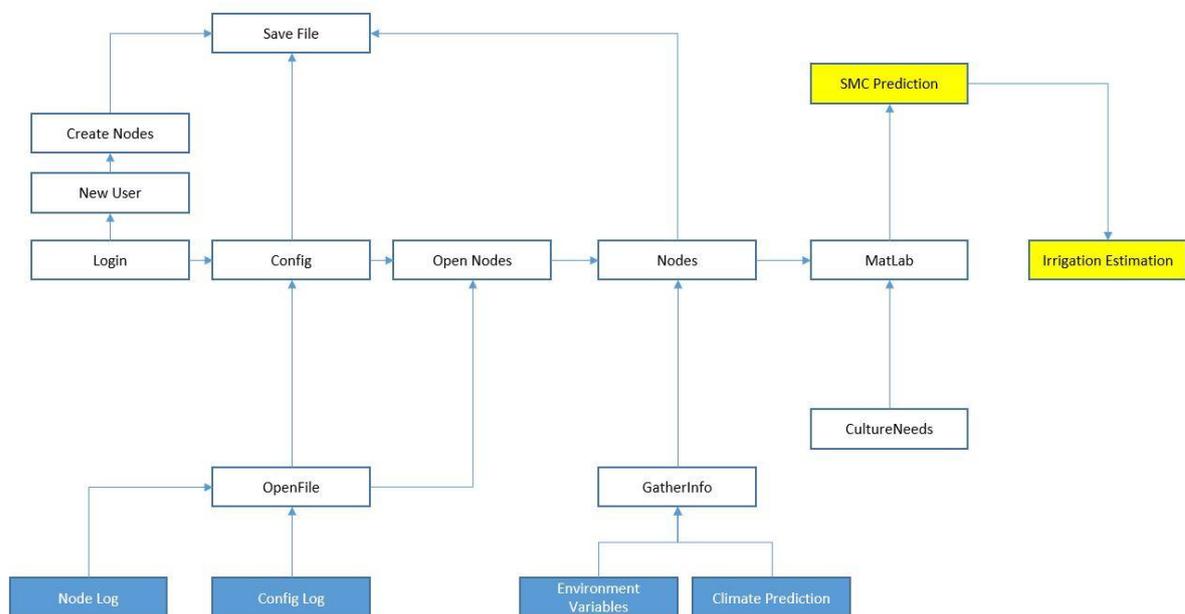
O software Matlab R2013b da Mathworks permite a implementação de diferentes redes neurais. Sua arquitetura baseada em C++ permite um desempenho melhor do que outras linguagens comerciais para o desempenho de cálculos e buscas de dados. Além disso a memória consumida pelo programa é relativamente menor.

“O programa que é baseado na linguagem de programação java. consome o dobro de memória que programas em C++. Para a fase da busca de dados, um programa C e C++ leva metade do tempo que em Java. “ (PRECHELT, 2000, p. 24).

3.5 Estruturação do programa java

Um algoritmo capaz de realizar uma previsão ótima de umidade do solo tem sua utilidade como ferramenta de auxílio para o agricultor na tomada de decisão da irrigação ou não de sua lavoura. Cabe ressaltar que o processo decisório também pode ser automatizado de forma ideal. A Figura 3.11 representa graficamente o processo de fluxo de dados do algoritmo proposto, a parte de estruturação, leitura e salvamento de arquivos será realizada em Java. No que trata de algoritmos numéricos e redes neurais foram estruturadas completamente em Matlab.

Figura 3.11: Representação gráfica da utilização das classes no programa. Em branco, classes do programa. Azul: agentes externos. Amarelo: algoritmos programados em Matlab



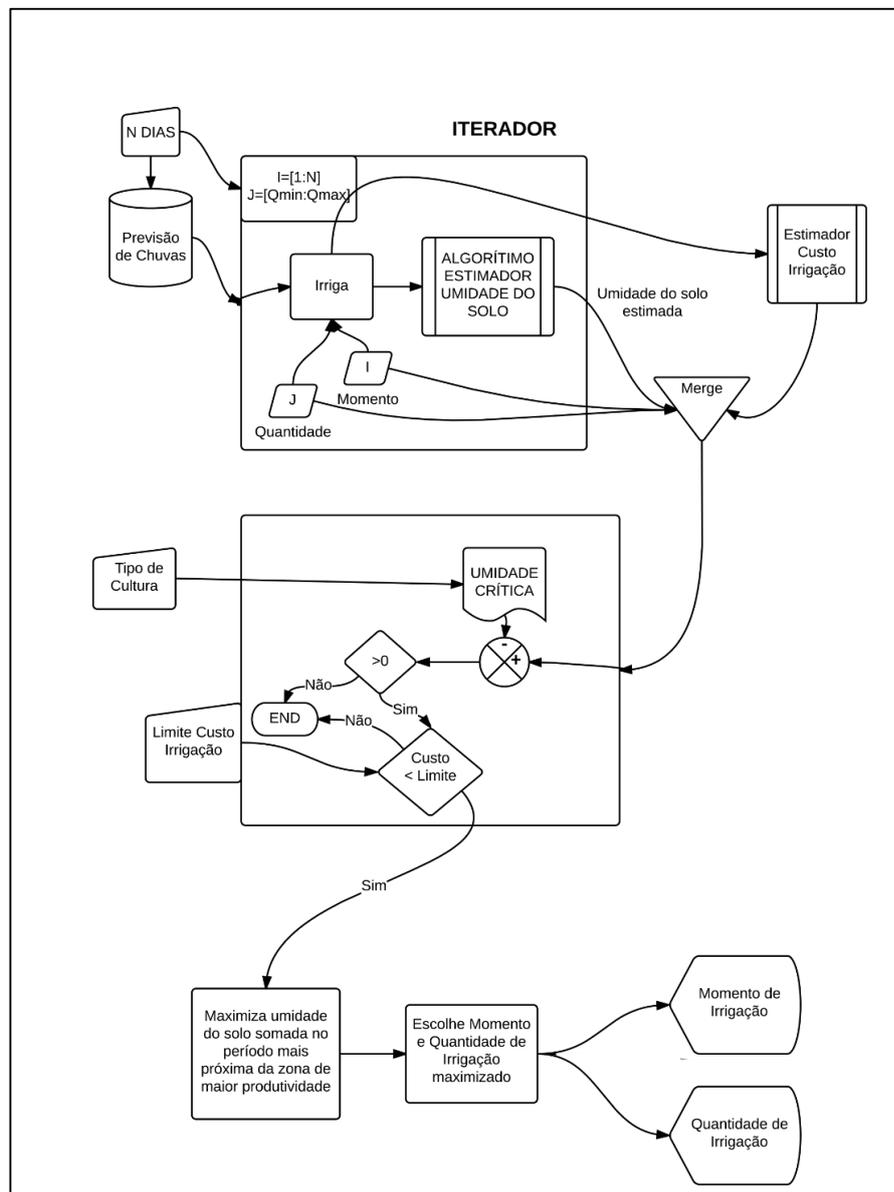
Funcionalidades do programa:

- 1) *Login* de usuário – Permitir o usuário criar contas e realizar a entrada no sistema de forma segura.
- 2) Salvamento de estados – Permitir que o último estado do programa seja salvo em uma base de dados até o novo acesso.
- 3) Comunicação com o software de matemática – Permitir envio de dados e funções afim de realizar os cálculos da rede NARX através do software.
- 4) Comunicação com uma base de dados – Abertura de arquivos de lote a partir de requisições.
- 5) Comunicação com bases de dados externas – Requisitar dados climáticos e mapas do software google maps.
- 6) Utilização de uma interface gráfica (*GUI*) para interação com o usuário.

3.5.1 Algoritmo iterador para previsão do momento de irrigação

Uma vez obtido os resultados para a previsão ótima da umidade do solo futura será utilizado um algoritmo capaz de utilizar os dados afim de criar estimativas para os efeitos do momento e quantidade de irrigação na lavoura a fim de facilitar a utilização pelo agricultor. A Figura 3.12 ilustra o funcionamento do algoritmo no ponto de vista da utilização conjunta com o algoritmo de estimativa da umidade do solo. A lógica demonstrada na imagem representará a estrutura base do desenvolvimento do algoritmo que não será divulgado por questões de sigilo.

Figura 3.12: Representação gráfica (*Flow Chart*) do algoritmo Iterador.



3.5.2 Bibliotecas adicionais utilizadas

Além da programação própria do algoritmo, algumas partes do código serão utilizadas “*Of the shelf*” a partir de bibliotecas conhecidas da linguagem Java.

Matlab Control 4.1.0

Escrito originalmente pela universidade de Regents California permite a integração e envio de comandos ao software Matlab 2013b. Será utilizado devido a necessidade de realizar cálculos custosos para o programa em java e a estruturação de redes neurais permitidas pelo software Matlab.

Opencsv-3.1

Permite o manejo de arquivos do tipo CSV por programas em Java.

Forecast IO Lib 1.5.2

Permite a conexão com a base de dados de previsão do tempo da plataforma Forecast IO. A plataforma reúne modelos de previsão do tempo para previsões futura.(Sem histórico) baseando-se nos modelos NAM e GFS.

Java SWT 4.4.1

Biblioteca disponibilizada pelo eclipse.org que permite a construção de interfaces gráficas a partir da programação Java.

4 RESULTADOS

A avaliação dos resultados da pesquisa consiste em estimar quatro dias no futuro da umidade do solo para as profundidades de 5 cm e 20 cm a partir de uma modelagem realizada com um algoritmo do tipo NARX. Essa seção discute a avaliação de parâmetros e a performance do modelo, assim como os resultados do programa estruturado.

4.1 Avaliação do desempenho do algoritmo não preditivo

No intuito de avaliar uma análise assertiva do desempenho do algoritmo NARX, os conjuntos de dados selecionados foram testados.

Uma das premissas de valor a serem abordadas reside no fato de que a modelagem do sistema prevê a realização de novos treinos sempre que os dados forem atualizados. Sendo assim, é necessária uma avaliação da performance dos resultados para um treinamento arbitrário, não somente considerando a performance de um treinamento singular.

Uma arquitetura ótima para o modelo deve ser escolhida de forma que tenha otimizado dois parâmetros: 1) a média dos erros das saídas; 2) O desvio padrão dos erros de suas saídas.

Realizando a análise iterativa de performance para o modelo com 1000 cenários por arquitetura da camada escondida, obtém-se a distribuição da Figura 4.1, onde foram considerados a média (linha contínua) e o desvio padrão (linha tracejada) totalizando 120.000 cenários.

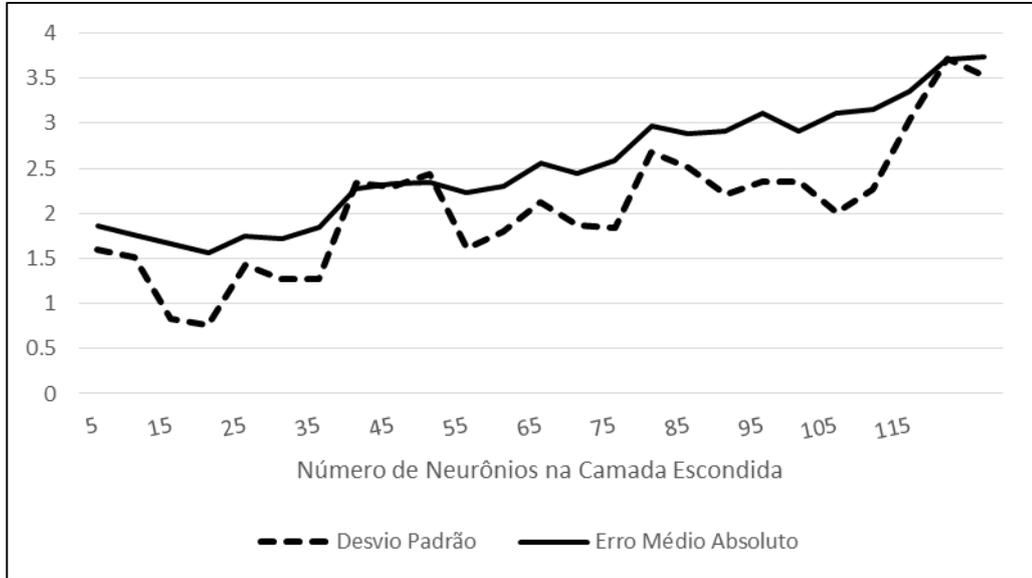
Tabela 4: Resultado das iterações de performance para camada escondida

Nº Neurônios	5	10	15	20	25	30	35	40	45	50	55	60
Desvio Padrão	1.60%	1.52%	0.83%	0.76%	1.43%	1.28%	1.28%	2.33%	2.30%	2.44%	1.61%	1.80%
Erro Médio	1.86%	1.76%	1.66%	1.56%	1.75%	1.72%	1.85%	2.28%	2.32%	2.35%	2.23%	2.30%
Nº Neurônios	65	70	75	80	85	90	95	100	105	110	115	120
Desvio Padrão	2.13%	1.86%	1.84%	2.68%	2.51%	2.21%	2.36%	2.35%	2.01%	2.27%	3.05%	3.71%
Erro Médio	2.56%	2.45%	2.59%	2.97%	2.88%	2.91%	3.11%	2.91%	3.12%	3.15%	3.35%	3.71%

É notável que tanto a média para números entre 15 e 20 perceptrons na camada escondida quanto o desvio padrão atingiram os menores valores da distribuição. A

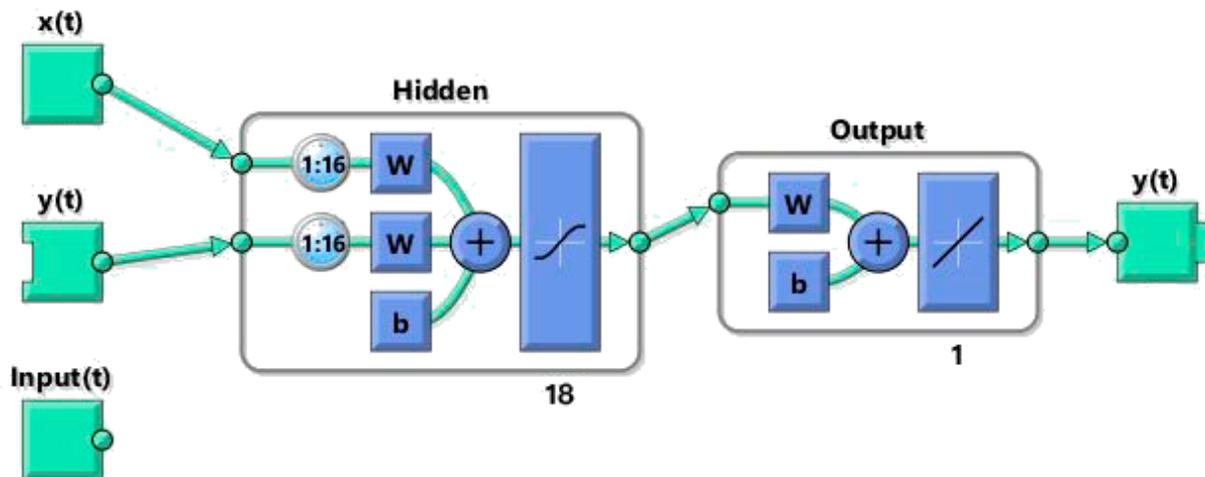
Tabela 4 mostra os valores dos resultados médios obtidos em relação ao número de perceptrons na camada escondida.

Figura 4.1: Performance para diferentes arquiteturas



Baseando-se no resultado obtido foi escolhida a arquitetura exposta na Figura 4.2 que inclui dezoito neurônios na camada escondida e dezesseis atrasos em suas entradas, como o discutido na Seção 3.2.1.

Figura 4.2: Arquitetura ótima para a o algoritmo proposto



Como observado em Zaman (2010), quanto mais próximo da superfície, maior a incerteza sobre previsões de SMC quando estas se baseiam em sensores instalados

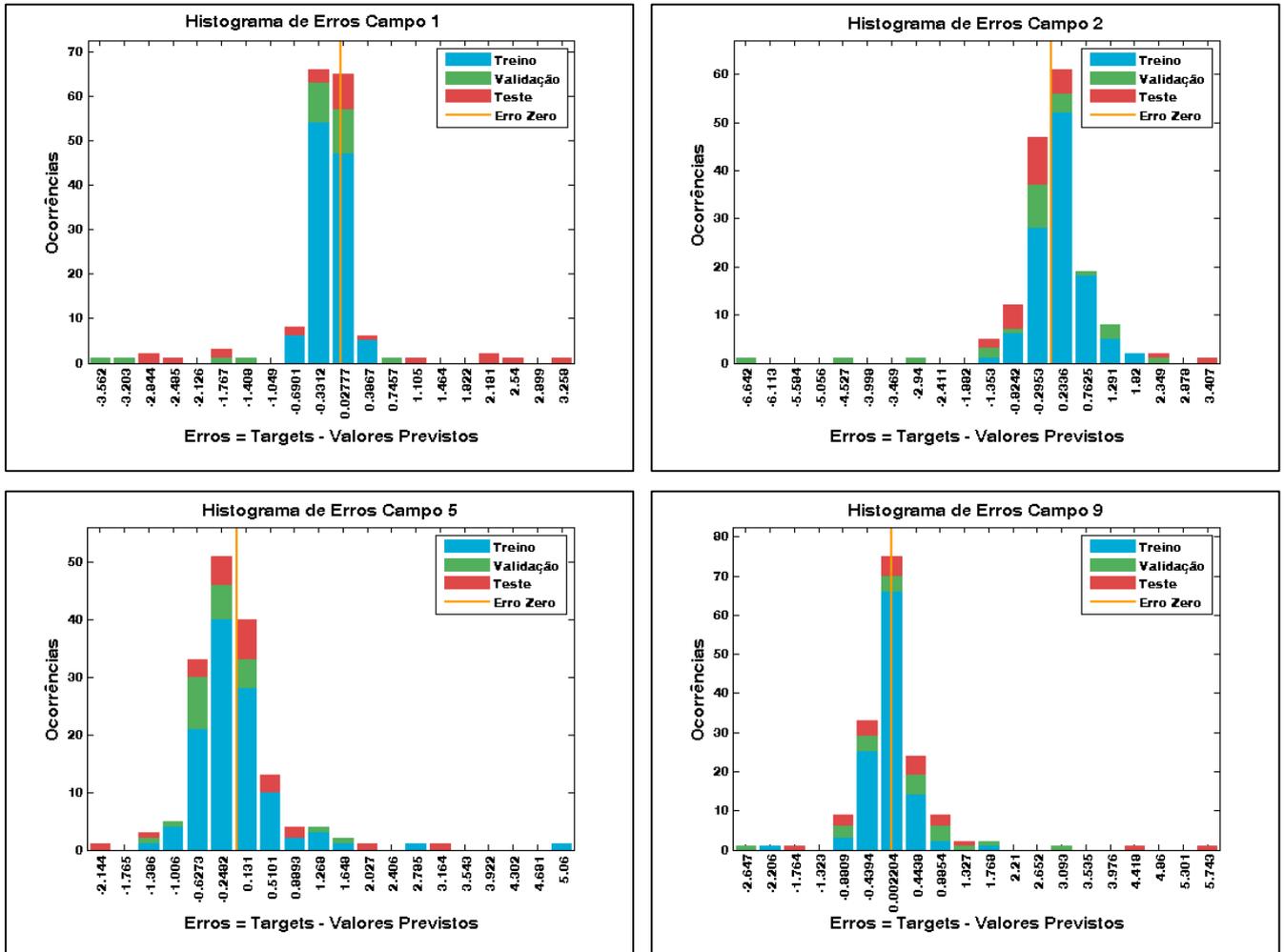
no solo. Partindo deste pressuposto é de interesse primordial estimar a umidade para a profundidade de 5 cm com um valor de erro aceitável.

O histograma de erros demonstra a capacidade de regressão de um algoritmo de aprendizado: separando a avaliação dos conjuntos de teste, validação e treino é possível perceber a efetividade do algoritmo *Cross Validation*. A Figura 4.1 representa um histograma de erros da estimativa de umidade do solo em 5 centímetros para os quatro campos, divididos respectivamente em erros para o conjunto de treino (em azul), erros para o conjunto de teste (em vermelho), erros do conjunto de validação (em verde). A linha laranja representa o erro zero.

É notória a capacidade de generalização do algoritmo e a prevenção do sobreajuste, uma vez que para todos os casos observa-se uma distribuição, envolvendo os três conjuntos citados. Os campos 2 e 5, ambos respectivamente com as maiores médias de umidade do solo medidas demonstraram um histograma de erros mais esparso, quando em comparação com os casos do Campo 1 e Campo 9. Tal fato pode ser explicado em razão do erro ser medido de forma absoluta, sendo que o peso percentual da medida não apresentaria tanta relevância.

Neste tipo de treinamento foram utilizados somente os dados obtidos no campo de treinamento e sem que sejam consideradas as previsões climáticas. Nota-se pela imagem da Figura 4.4, que representa a resposta do algoritmo treinado no domínio tempo, que a regressão conseguiu representar a curva dos campos mencionados na Seção 3.1 fidedignamente.

Figura 4.3: Histograma representando o algoritmo treinado para os campos escolhidos com estimativa em 5cm



A Tabela 5 mostra o erro em cada um dos campos. Para a medição da performance foi selecionado o método dos mínimos quadráticos R^2 . A variável erro absoluto médio também foi levada em conta, uma vez que sua aplicação prática é de fácil percepção. Para todos os casos obteve-se um erro médio menor que 0,6%.

Figura 4.4: Resultados do desempenho no domínio tempo e erros do algoritmo para previsão em 5cm

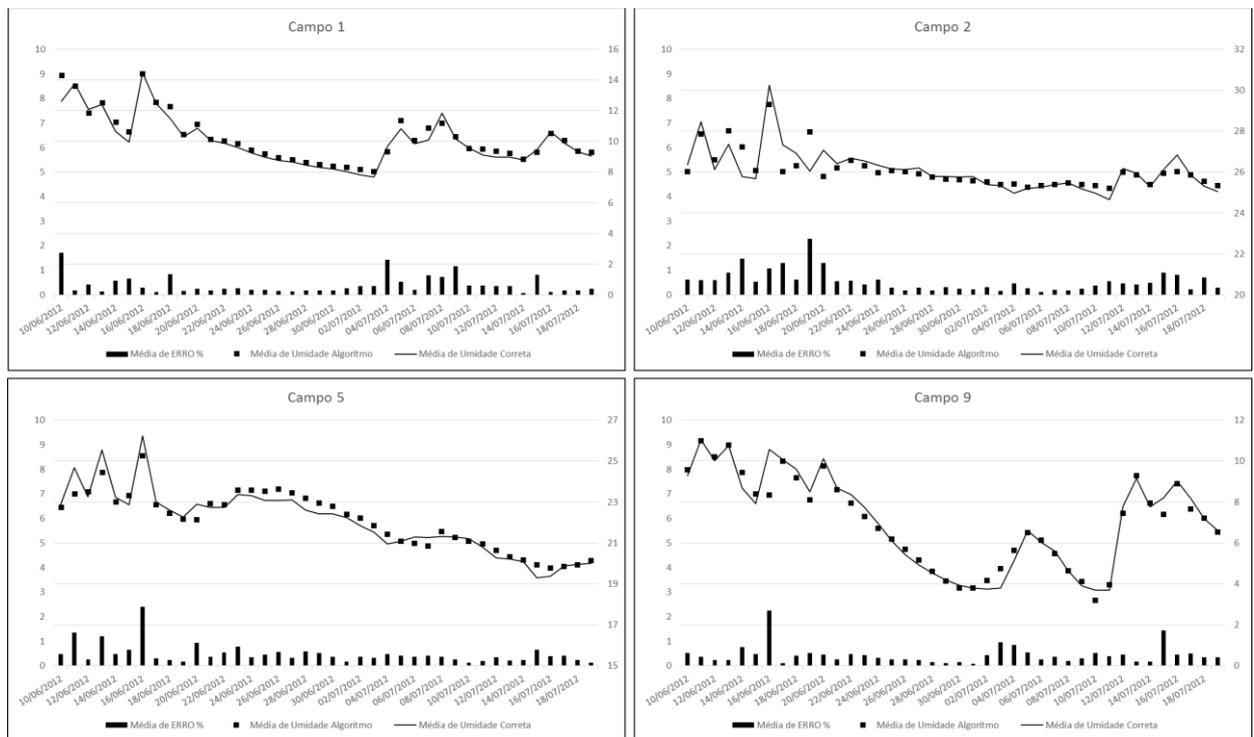


Tabela 5: Performance do algoritmo para os quatro campos à 5cm

Campo	Performance (R^2)	Erro Absoluto médio (%)
1	0.6078	0.41
2	0.9526	0.56
5	0.5963	0.49
9	0.7097	0.44

Nota-se que simulando o algoritmo com o treinamento realizado apenas para 25 dias e avaliando seu desempenho para os dias restantes, não há perda significativa de performance. Os resultados podem ser observados para as representações gráficas da Figura 4.4, assim como na Tabela 6.

Uma aplicação prática que pode ser levada em conta é a consideração da performance do algoritmo dado um conjunto de treinamento reduzido e um conjunto de testes significativamente maior que o treinado. Este tipo de abordagem também

permite considerar a performance do algoritmo desconsiderando a incerteza da previsão do tempo. Uma analogia a este tipo de teste poderia ser o caso onde a previsão do tempo fosse 100% correta.

Na Tabela 5 estão representados os resultados da performance máxima do algoritmo para 100 casos treinados medidos para o caso onde o conjunto de testes é o mesmo que o conjunto treinado e para aplicando este mesmo algoritmo em 15 dias no Futuro.

Tabela 6: Performance do algoritmo treinado com 25 dias de medições testado em 25 dias (caso completo) e 40 dias

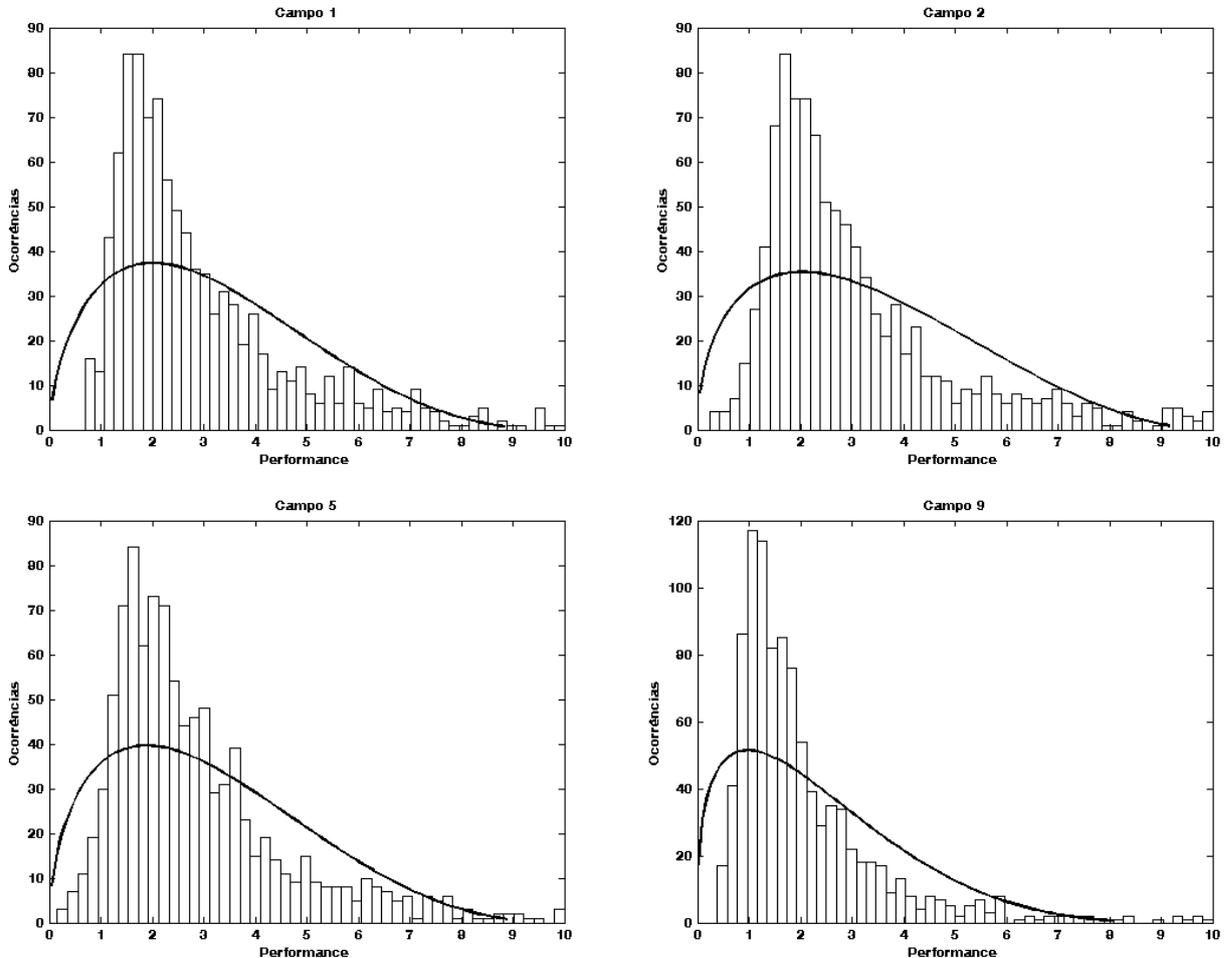
Campo	Performance teste 25 dias (melhor cenário)	Performance teste 40 dias (melhor cenário)
1	0,3203	0,8102
2	0,7748	1,0500
5	0,5917	0,7481
9	0,3761	1,1651

Como esperado, a performance para os testes com exatamente os mesmos dias de treino foi superior em todos os casos. A estimativa para 40 dias nos permite comparar os resultados com os da próxima seção onde será incluído os resultados considerando as previsões climáticas.

4.2 Avaliação do resultado utilizando previsão do tempo

Conforme o exposto na Seção 3.4, para a correta aplicação no programa o algoritmo necessitará novos treinamentos quando novos dados forem adicionados aumentando assim seu conjunto de treino. Para tal, torna-se de vital importância realizar o estudo da performance do algoritmo para diferentes treinamentos. Os gráficos da Figura 4.5 permitem uma análise do erro médio esperado para o algoritmo, assim como sua distribuição de probabilidades. A distribuição beta foi a curva de melhor generalização dos dados testados.

Figura 4.5: Performance do algoritmo com previsão do tempo calculado para 1000 iterações por campo



Utilizando performances condizentes com a maior probabilidade obtida a partir das distribuições gama nos casos acima para cada um dos campos foi representado a distribuição dos erros dos dias dez de junho até trinta de junho, intervalo para o qual

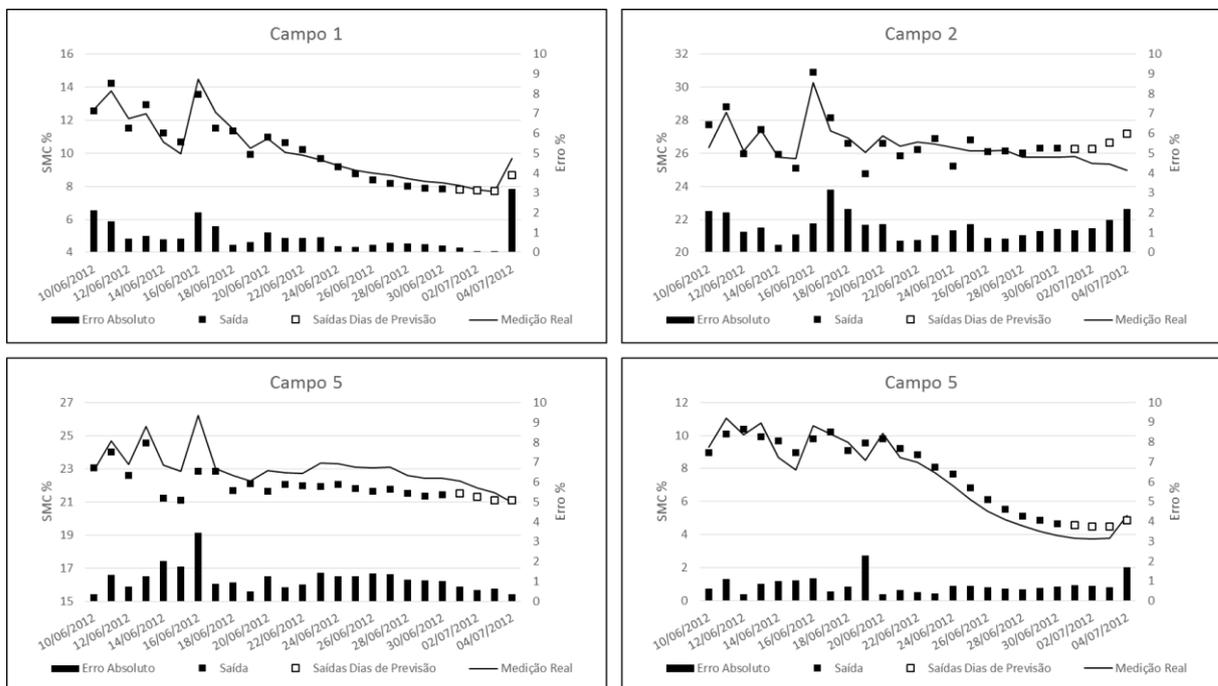
a rede foi treinada, e de primeiro a quatro de julho (linha tracejada), intervalo em que a rede realiza a previsão de umidade do solo.

Diferentemente do experimento realizado na Tabela 5, considera-se agora o sistema alimentado com a previsão das 84 horas de umidade do solo, da meia noite do dia primeiro de junho de 2012 até a às dezoito horas do dia quatro de julho de 2012.

Tabela 7: Resultados da distribuição beta

	Campo 1	Campo 2	Campo 5	Campo 9
R ² Médio	2.90	3.04	2.91	2.87
Desvio Padrão	1.72	1.84	1.77	2.04

Figura 4.6: Resultados do desempenho no domínio tempo para campo em 5 cm com treinamento limitado a 25 dias.

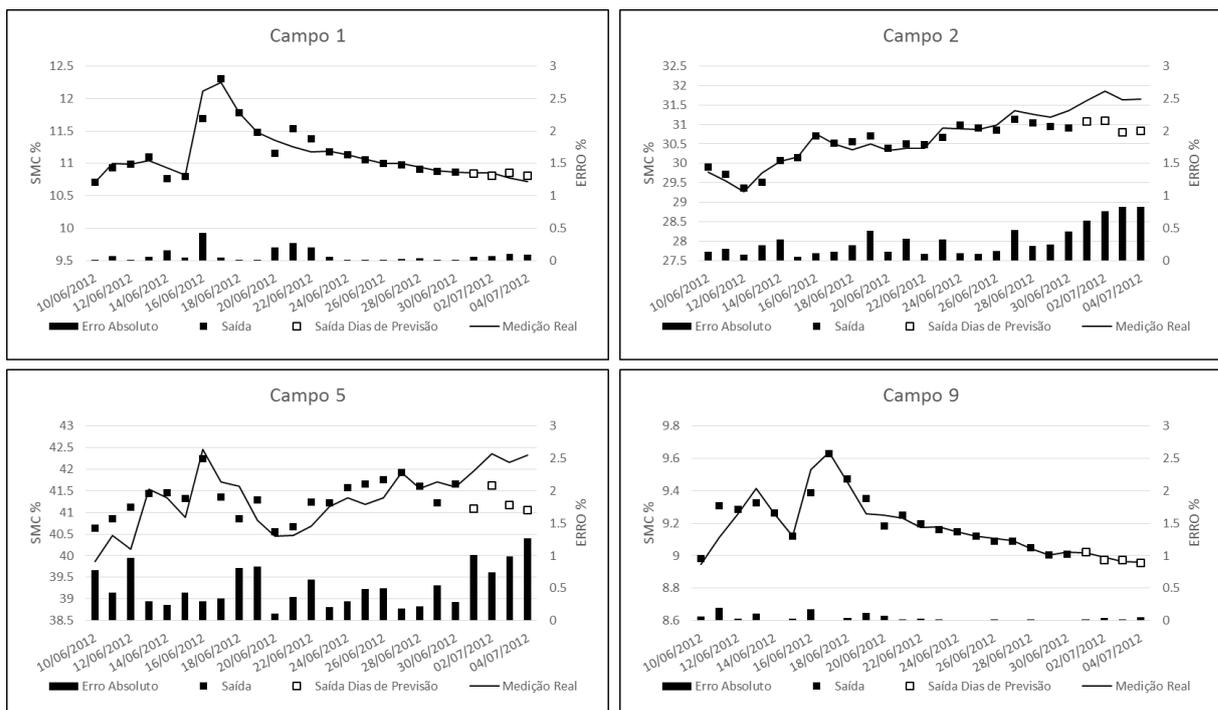


Os resultados da Figura 4.6 estão apresentados na Tabela 7. Adicionalmente foram simuladas dentro de 100 casos o treinamento com melhor performance R². Nota-se que diferentemente para a simulação

Tabela 8: Performance do algoritmo para previsão de 4 dias

Campo	Cenário	Performance (R ²)
1	Base	1.67
	Melhor	0.67
2	Base	2.72
	Melhor	0.40
5	Base	2.72
	Melhor	0.25
9	Base	1.02
	Melhor	0.75

Realizou-se o mesmo teste para o desempenho do algoritmo de previsão à 20cm. Nota-se um erro absoluto de menor magnitude para todos os casos, todavia observa-se uma diferença significativa do erro associado novamente aos Campos 2 e 5. Os erros obtidos em todos resultados são aceitáveis para aplicação.

Figura 4.7: Resultados do desempenho no domínio tempo para medição de 20 cm

O desempenho para a simulação de um sistema de irrigação é demonstrado na Figura 4.8. Neste caso, utilizou-se a rede treinada na Figura 4.6 e foi testado

cenários para a obtenção de uma SMC% acima de 4,5%. O resultado do algoritmo foi a recomendação de uma aplicação de 6,00 mm/m³ de água à meia noite do dia dois de julho de 2012. O sistema simulado, propositalmente foi o do campo 9, campo mais estressado. Nota-se que os gráficos estão em menor escala, o suficiente para representar períodos de seis horas.

Baseado na saída do sistema este resultado reflete o máximo aproveitamento da água dadas as previsões climáticas futuras e o desempenho histórico do sistema dadas as premissas previamente utilizadas, considerando uma aplicação máxima de água de 7mm/m³ e no máximo um ciclo de irrigação. Nota-se que para os dias testados não existiu incidência de chuvas até o penúltimo horário amostrado, deixando claro o estresse do solo se encontrava.

Nota-se que a aplicação foi feita de forma a demonstrar a decisão do sistema, não se atendo a estimativas agronômicas para a umidade exata.

O desempenho de um sistema adaptativo torna-se mais assertivo a medida em que sua base de treinamento cresce. Serve de motivação a possibilidade de realizar treinamentos para uma base de dados maior.

A Figura 4.9 evidencia a recomendação do sistema para a mesma rede de testes, desta vez permitindo a escolha de dois momentos para a irrigação ainda respeitando o teto de 7mm/m³ estipulado arbitrariamente. O sistema manteve-se em um patamar acima de 5% de umidade

Figura 4.8: Estimativa do sistema para a decisão ótima de irrigação à 5 cm

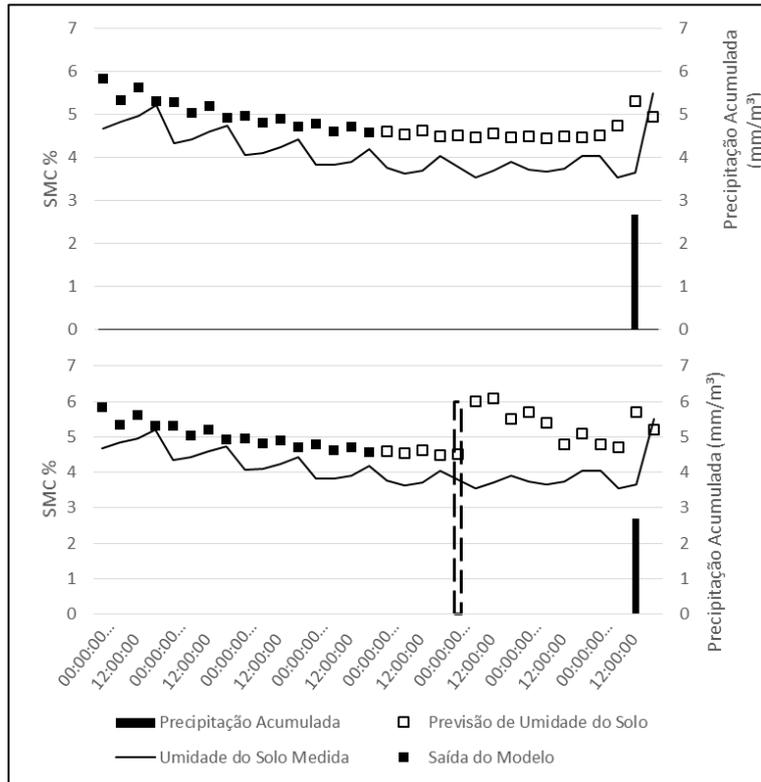
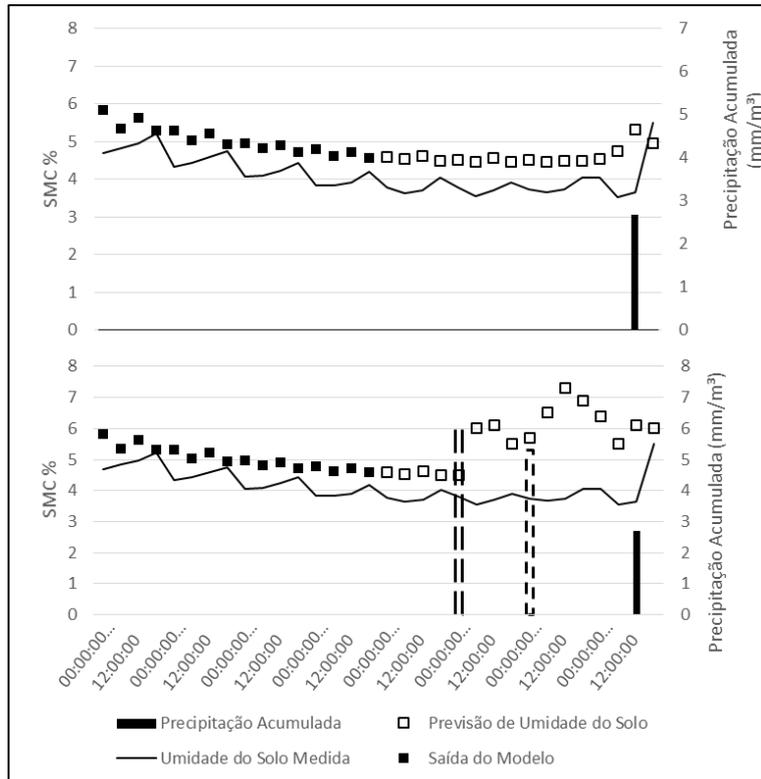


Figura 4.9: Resposta do sistema permitindo dois momento de irrigação



4.3 Avaliação do resultado do programa estruturado

O programa em Java permitiu realizar a partir de uma interface amigável os experimentos de predição de umidade do solo e possui como saída a recomendação da quantidade de irrigação.

A Figura 4.10 demonstra as funções do sistema em sua janela principal. Nesta janela é informado ao usuário as condições climáticas atuais. O programa estruturado contém todas as classes planejadas e permite um controle de criação de conta para o usuário como na Figura 4.11.

Figura 4.10: Interface do programa em Java

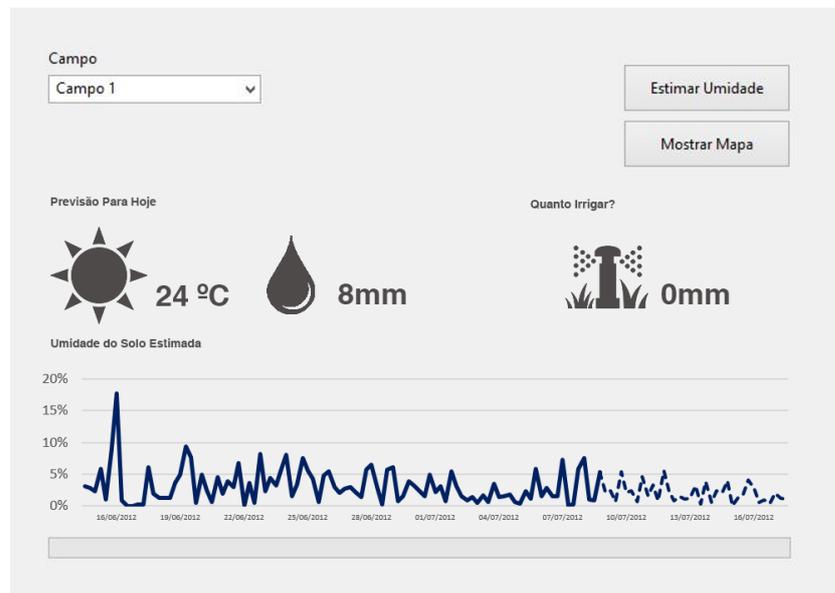
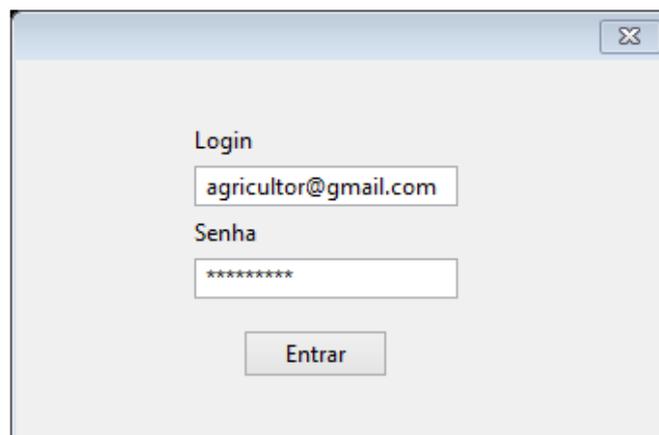
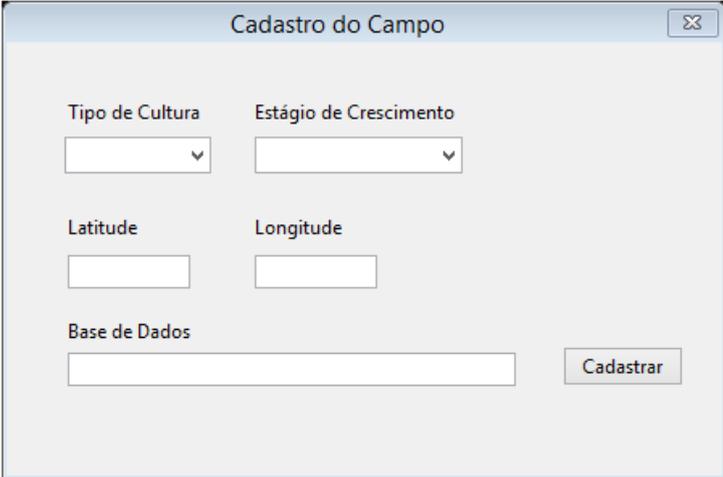


Figura 4.11: Interface para entrada de usuário



A Figura 4.12 mostra a interface de cadastramento de novos sensores no campo, dado que os resultados devem ser simulados, escolhe-se a localização do arquivo com dados históricos do sistema.

Figura 4.12: Diálogo de cadastro do campo

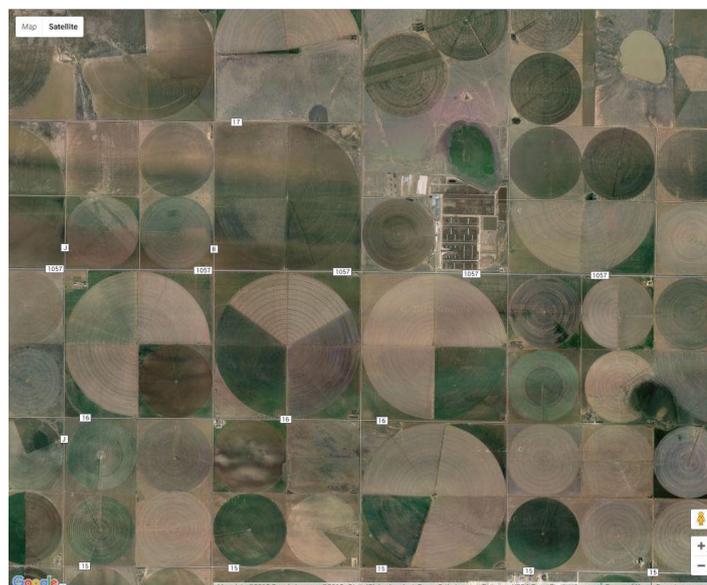


The image shows a software dialog box titled "Cadastro do Campo". It features a close button (X) in the top right corner. The form contains the following elements:

- Tipo de Cultura:** A dropdown menu.
- Estágio de Crescimento:** A dropdown menu.
- Latitude:** A text input field.
- Longitude:** A text input field.
- Base de Dados:** A larger text input field.
- Cadastrar:** A button located at the bottom right of the dialog.

A Figura 4.13 ilustra a representação da janela de diálogo de visualização do Google Maps. A integração com a plataforma foi realizada no intuito de possibilitar uma escolha mais assertiva dos dados cartográficos (latitude e longitude) pelo usuário.

Figura 4.13: Visualização da integração com *Google Maps*



5 CONCLUSÃO

Ao longo desse trabalho foi verificada a importância de um sistema que melhor estime a umidade do solo, bem como determine ao agricultor tomar a decisão mais adequada para o manejo da irrigação. Tanto pelo benefício em relação aos sistemas já existentes, quanto por sua aplicabilidade prática, verificada pelo aumento de poder de processamento, decréscimo do custo de implementação de tecnologias remotas e possibilidade de implementação de algoritmos de aprendizado.

As especificações propostas contemplam a funcionalidade de um sistema desde a concepção do algoritmo, até sua aplicabilidade através de um programa em linguagem comercial orientada ao objeto.

Com base nas análises feitas acredita-se que a implementação de um sistema de monitoração e previsão seria viável do ponto de vista algorítmico e de aplicação em um servidor.

A concepção do algoritmo baseada em profundo embasamento teórico permitiu a criação de uma estrutura considerada ótima, no ponto de vista do autor, e provou ser assertiva não somente para os melhores casos de treinamento, mas para a generalização de 120.000 treinamentos, utilizando 120 cenários.

A demonstração dos resultados no domínio tempo, assim como a previsão da umidade do solo para ambos os casos para 5 cm e 20 cm mostrou-se assertiva, no sentido que possui um erro significativamente menor que outros métodos.

Na motivação da aplicabilidade prática o sistema preditivo e de monitoramento da umidade do solo permite tomar a decisão da irrigação antes da planta apresentar sintomas de stress e também permite a decisão de poupar água uma vez que acusa um conteúdo de umidade do solo ideal para o desenvolvimento da cultura. Outras metodologias citadas de manejo de irrigação baseiam-se, por exemplo, em características de stress hídrico da planta e só permitem tomar uma decisão após a detecção deste.

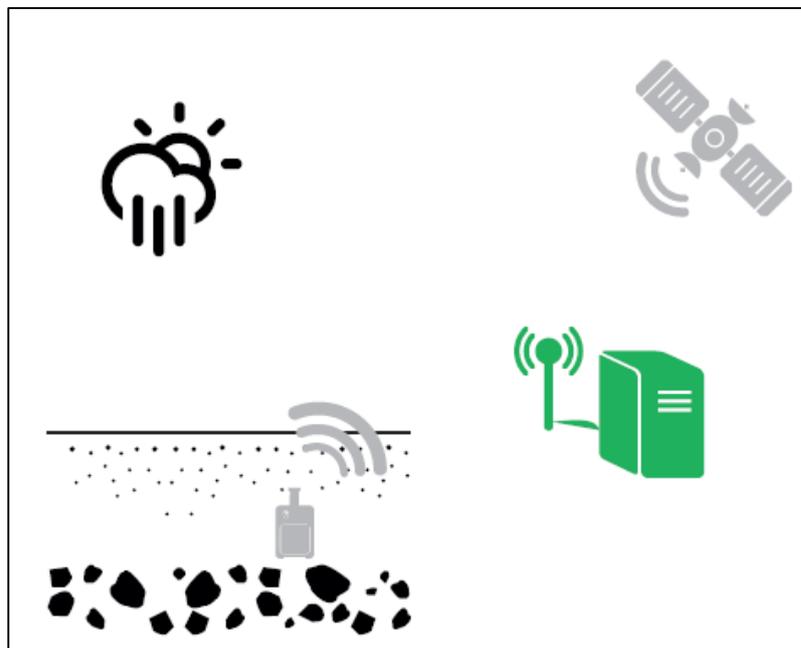
Conforme o previsto, não foi possível obter uma avaliação para a taxa de assertividade do sistema de recomendação de irrigação devido a indisponibilidade de dados em tempo real. Entretanto, a estruturação do programa em linguagem comercial assegura a aplicabilidade e motiva a continuidade do trabalho.

PESQUISA FUTURA

O próximo passo da pesquisa deverá ser a aplicação prática do programa em uma situação de medição de umidade do solo em tempo real.

Para realizar medições do solo e para aplicar um algoritmo de forma prática utiliza-se um sistema de sensores sem fio instalados no campo que captam dados reais sobre características do solo. Os dados deverão ser enviados ao servidor onde são comparados às previsões do tempo e a base histórica de dados captados por aquele sensor. A Figura 2.9 ilustra o funcionamento de um sistema de medições remotas de variáveis do solo e o envio para um servidor onde ocorrem os cálculos.

Figura: Representação do funcionamento do sistema proposto



A necessidade de aplicação do sistema em um campo onde exista a possibilidade de controlar o sistema de irrigação, preferencialmente do tipo pivô central devido a homogeneidade da sua aplicação possibilitaria o teste do sistema de recomendação de irrigação e assim a comprovação dos resultados da última parte deste trabalho de pesquisa.

Um dos principais desafios é a utilização dos dados de forma fácil e intuitiva para qualquer agricultor. Ao utilizar um sistema de aprendizado e quanto mais o sistema é utilizado, mais aprende sobre a lavoura, conseguindo após a utilização

contínua levar a economia de luz e água pela irrigação. A Figura 2.10 ilustra uma possível interface para o programa vendido a um nível comercial.

Figura: Possível interface para o sistema



Uma outra possibilidade de pesquisa seria comparação de diferentes tipos de algoritmos para predição da umidade do solo, tais como Redes Bayesianas e *Relevance Vector Machines*.

Qualquer iniciativa em relação a obter práticas mais sustentáveis que aumentem a produtividade do campo e diminuam seu uso de água são de valia para a sociedade como um todo. A utilização dos sistemas de preditivos para melhorar a qualidade de vida e deve sim ser uma prática buscada pelos pesquisadores. A pesquisa deve ser continuada.

REFERENCIAS BLIBIOGRÁFICAS

AGATONOVIC-KUSTRIN, S.; BERESFORD, R. **Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research.** Journal of pharmaceutical and biomedical analysis, vol. 22, no. 5, p. 717-727, 2000.

AQUASTAT. **Country profiles: Brazil.** Food and Agricultural Organization, São Paulo, 2000.

BERNARDO, Salassier. **Impacto ambiental da irrigação no Brasil.** Recursos hídricos e desenvolvimento sustentável da agricultura. Viçosa: MMA, SRH, ABEAS, UFV, 1997.

BRITO, LT de L. **Avaliação de impactos das atividades antrópicas sobre os recursos hídricos da bacia hidrográfica do rio Salitre-Bahia e classificação das fontes hídricas.** Embrapa Semiárido-Teses/dissertações, 2003.

CANAL DO PRODUTOR, **Aspersores Sobre Lavoura.** Disponível em: <http://www.canaldoprodutor.com.br>, Acesso em: 5 de outubro de 2015

CAPOLUPO, Alessandra. Photogrammetry for environmental monitoring: **The use of drones and hydrological models for detection of soil contaminated by copper.** Science of the Total Environment, vol. 514, p. 298-306, 2015.

CARLSON, T.; R. GILLIES; T. SCHMUGGE. **An interpretation of methodologies for indirect measurement of soil water content.** Agricultural and Forest Meteorology vol. 77(3-4), p. 191-205, 1995.

CEMIG. **Estudo de Otimização Energética.** Belo Horizonte, 1993.

CONSTRUINDO **Esquema De Funcionamento Do Sistema De Irrigação Por Gotejamento.** Disponível em: <http://construindo.org/irrigacao-por-gotejamento/> Acesso em: 5 de outubro de 2015.

DE SOUZA MENDES, Paulo Cesar. **Caracterização De Um Sensor Para Medição De Umidade Do Solo Com Termo-Resistor a Temperatura Constante.** Tese (Doutorado) - Universidade Federal da Bahia, Salvador, 2006.

DEVIJVER, P. A.; KITTLER, J. **Pattern recognition: a statistical approach**. GB: Prentice-Hall, London, 1982.

GEMAN, S., BIENENSTOCK, E. and DOURSAT, R. **Neural Networks and the Bias/Variance Dilemma**, Neural Computation, v4, p. 1-58. 1992.

HOSPEDALES, T. **Data Mining - Missing Data, Outliers & Ensembles**. ECS 607/766, Notas de Aula, p. 51, London, 2013.

IDSO, S.; R. JACKSON; R. REGINATO. **Compensating for environmental variability in the thermal inertia approach to remote sensing of soil moisture**. Journal of Applied Meteorology, 1976.

INSA, Instituto Nacional Do Semiárido. **NDVI – PARCERIA INSA e LAPIS/UFAL**. 27 Disponível em: <http://www.insa.gov.br/ndvi/#.VkeU73arSiM>. Acessado em 10 de Novembro de 2015.

HYDROLOGY AND REMOTE SENSING LABORATORY **Related Topics: Soil Moisture Experiments**. . 2015.

KALEITA, A.; L. TIAN; M. HIRSCHI. **Relationship between soil moisture content and soil surface reflectance**. Transactions of the ASAE 48: 1979-1986, 2005.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: **International joint Conference on artificial intelligence**. [S.l.: s.n.], v. 14, p. 1137–1145, 1995.

LEE, H.; BATTLE, A.; RAINA, R.; NG, A. Y. **Efficient sparse coding algorithms**. In: **Advances in neural information processing systems**. Michigan, p. 801-808, 2006.

LORENZ, E.N. **On the Existence of Extended Range Predictability**. Journal of Applied Meteorology, vol.12, p. 543-546, 1973.

MADER, S. **Center pivot irrigation revolutionizes agriculture**. The Fence Post Magazine. v. 6, 2010.

MAY, Robert; DANDY, Graeme; MAIER, Holger. **Review of input variable selection methods for artificial neural networks**. INTECH, 2011.

MCNAIRN, H. et al **The Soil Moisture Active Passive Validation Experiment 2012 (SMAPVEX12): Prelaunch calibration and validation of the SMAP soil moisture algorithms**. IEEE Trans. Geosci. Remote Sens, v. 53, p. 2784-2801, 2005

MENEZES, José Maria P.; BARRETO, Guilherme A. **Long-term time series prediction with the NARX network: an empirical evaluation**. *Neurocomputing*, v. 71, n. 16, p. 3335-3343, 2008.

MINSKY, Marvin; SEYMOUR, Papert. **Perceptrons**. 1969.

MISSOURI R. P. **Growing rice where it has never grown before**. College of Agriculture, Food and Natural Resources, University of Missouri, Missouri, 2008.

NUNES, V.S. **Agricultura irrigada x Saúde ambiental: existe um conflito**. Artigo de Divulgação na Mídia, Embrapa Pantanal, Corumbá -MS, no. 11, p.1-2, 2001

PENMAN, Howard Latimer. **Natural evaporation from open water, bare soil and grass**. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences. Vol. 193. No. 1032. The Royal Society, 1948

PRECHELT, Lutz. **An empirical comparison of seven programming languages**. *Computer*, vol. 33, no. 10, p. 23-29, 2000.

RUMELHART, David E.; HINTON, Geoffrey E.; WILLIAMS, Ronald J. **Learning internal representations by error propagation**. CALIFORNIA UNIV SAN DIEGO LA JOLLA INST FOR COGNITIVE SCIENCE, 1985.

RUTLEDGE, G.K., J. ALPERT, and W. EBUISAKI. **NOMADS: A Climate and Weather Model Archive at the National Oceanic and Atmospheric Administration**. *Bull. Amer. Meteor. Soc.*, vol. **87**, p. 327-341, 2006.

SIEGELMANN, Hava T.; HORNE, Bill G.; GILES, C. Lee. **Computational capabilities of recurrent NARX neural networks**. *Systems, Man, and Cybernetics, Part B: Cybernetics*, IEEE Transactions on, vol. 27, no. 2, p. 208-215, 1997.

SMAPVEX12. **SMAPVEX12 Database Report**. Base de dados do Experimento, Toronto, 2012.

SMITH, Leonard. **Chaos: a very short introduction**. Oxford University Press, p.117, Oxford, 2007

STERN, P.C.; EASTERLING, W.E. **Making Climate Forecasts Matter**. National Academic Press: National Research Council, Panel on the Human Dimensions of Seasonal-to-Interannual Climate Variability, EUA. 175p. 1999.

SWR, Soil Water Relations. **Root zone soil water extraction and plant root development patterns**, p.1, Toronto, 2014

TESTEZLAF, R. **Irrigação por Aspersão**. Campinas, 2004. Notas de Aula. Disponível em: <http://webensino.unicamp.br/disciplinas/FA876-055506/apoio/12/aspersao.pdf>
Acesso em: 5 de outubro de 2015.

VERSTRAETEN, W. W.; F. Veroustraete; C. J. Van Der Sande; I. Grootaers; J. Feyen. **Soil moisture retrieval using thermal inertia, determined with visible and thermal spaceborne data, validated for European forests**. Remote Sensing of Environment, no. 101, p. 299-314. 2006.

VORIES, E. D. **Spatial Variability of Yield for Sprinkler Irrigated Rice**. In: Rice Technical Working Group Meeting Proceedings. 2010.

WIGGINS, Geraint A. **Artificial Neural Networks v. 2013**. p. 17-19, London, 2013.
Disponível em: <http://2013.qmplus.qmul.ac.uk/mod/resource/view.php?id=305193>
Acessado em : 3 de agosto de 2015

ZAMAN, B. **Remotely sensed data assimilation technique to develop machine learning models for use in water management**. Tese (Doutorado), Utah State University, Logan, 2010.

ANEXO A

```

% Obtenção dos dados de Entrada para previsão do tempo NAM
% Função desenvolvida para o manejo dos dados das bases de dados no formato
Netcdf

function [saida]=iterar_nam()
medicao=[];
me=[];
for k=10:1:30
    for g=0:6:18
        for i = 1:1:30
            medicao=[ChamaBasedeDados(2012,6,k,g,i)];
            me=[me;k,6,i,g,medicao]
        end
    end
end
saida=[me,medicao]

function data = ChamaBasedeDados(ano_,mes_,dia_,hora_,horasdeprevisao)

    if horasdeprevisao<10
        previsao=(cat(2,'00',num2str(horasdeprevisao)));
    elseif horasdeprevisao<100
        previsao=(cat(2,'0',num2str(horasdeprevisao)));
    else
        previsao=num2str(horasdeprevisao);
    end

    if dia_<10
        dia=(cat(2,'0',num2str(dia_)));
    else
        dia=num2str(dia_);
    end

    if mes_<10
        mes=(cat(2,'0',num2str(mes_)));
    else
        mes=num2str(mes_);
    end

    if hora_<10
        hora=(cat(2,'0',num2str(hora_)));
    else
        hora=num2str(hora_);
    end

    ano=num2str(ano_);

url=sprintf('/s%s/s%s%s/s/nam_218_%s%s%s_%s00_%s.grb2',ano,mes,ano,mes,dia,ano,
mes,dia,hora,previsao)

data = ncread(url,'Total_precipitation',[550 378 1],[1 1 1])
data = ncread(url,'Temperature_below',[550 378 1],[1 1 1],0.4)
end

```

ANEXO B

Nesta seção serão descritas as classes principais do programa orientado ao objeto. Funções secundárias não serão descritas.

File Handling

Classes com objetivo de lidar com abertura, modificação e salvamento de arquivos externos do programa em Java.

Nome da Classe -> Openfile

Função: Abrir arquivos no formato CSV dada uma referida pasta. Para cada linha do arquivo ele transforma salva os arquivos no formato ArrayList . O nome escrito na primeira coluna do arquivo salvo será o nome da ArrayList

Entradas:

String NomedoArquivo
String NomedoPasta

Saídas:

HashMap<String, ArrayList<Double>>
HashMap<String, ArrayList<String>>

Nome da Classe -> Savefile

Função: Salva arquivos no formato CSV.

Entradas:

ArrayList<String []> Dados
String NomedoArquivo
String NomedoPasta

Saídas:

“Arquivo CSV”

Nome da Classe -> GatherData

Função: Conecta ao servidor e recebe dados das variáveis desejadas

Entradas:

Int numerodoNó
String Funçãodesejada
String [] Dados

Saídas:

Arraylist<Node> ArraydeNodes

Estruturação de Dados

Classes com funções de armazenagem e atualização de variáveis internas ao sistema. Também servem como estruturas que armazenam dados de arquivos externos na memória interna do programa.

Nome da Classe -> Node

Função: Representa cada uma das unidades de leitura de variáveis relativas aos solo. Guarda a referência para a rede neural do sistema, características do campo. Chama a classes Matlab para calcular informações. Salva alterações com a classe save file.

Adiciona novos dados ao histórico salvo no matlab.

Entradas:

HashMap< String,ArrayList<String>> data_

Saídas:

Nome da Classe -> Config

Função: Salva as informações gerais do campo e login. Chama classe OpenFileDialog para receber dados salvos.

Entradas:

UserName

Saídas:

```
String farmName;
Double numberofNodes;
String culture;
Double farmLatitude;
Double farmLongitude;
```

Nome da Classe -> CultureNeeds

Função: Registra informações sobre a cultura do campo. Permite modelar a recomendação de irrigação baseando-se nos limites necessários para a planta.

Entradas:

UserName

Saídas:

```
String farmName;
Double numberofNodes;
String culture;
Double farmLatitude;
```

Inicialização do Programa

Classes utilizadas quando o programa é aberto tem como principais funções criar uma experiência intuitiva para o usuário, além de conseguir efetivamente retomar a etapa do programa de onde foi parado a última vez.

Nome da Classe -> NewAccount

Função: Recebe dados de entrada pelo usuário ou da GUI e chama a classe savefile para criar um novo csv.

Entradas:

String Login;
String farmName;
Double numberOfNodes;
String culture;
Double farmLatitude;
Double farmLongitude;

Nome da Classe -> IsUserNew

Função: Verifica se a entrada de login do usuário já existe. Se sim, chama a função config, se não, chama a função NewAccount.

Entradas:

String Login;
String Password

Nome da Classe -> CreateNodes

Função: Chamada caso o usuário seja novo, cria arquivos referentes a cada um dos nós no sistema .

Entradas:

Config Config;

Nome da Classe -> StartNodes

Função: Inicializa todos os Nós para a o login realizado

Entradas:

Config Config;

Saídas:

ArrayList<Node> ArraydeNodes

Classes de Execução

Classes caracterizadas pela atuação no sistema modelado pelos objetos anteriormente descritos. Atuam dinamicamente no sistema.

Nome da Classe -> Matlab

Função: Inicializa o matlab, abre o workspace desejado e roda as funções desejadas. Obs: o matlab é inicializado na sua forma escondida para possibilitar uma inicialização mais rápida do programa.

Entradas:

Int numerodoNó
String Funçãodesejada
String [] Dados

Saídas:

Arraylist<Node> ArraydeNodes

Nome da Classe -> RefreshNodes

Função: Recebe uma array de Nodes e adiciona aos dados de cada um as variáveis mais atualizadas correspondentes aos nós.

Conecta ao servidor e recebe dados das variáveis desejadas

Entradas:

ArrayList<Node> Node

Saídas:

Nome da Classe -> Iterate

Função: Realiza a iteração de dados a partir de função chamada no matlab. Registra os limites e requisitos do sistema e permite estimar a recomendação da irrigação

Entradas:

ArrayList<Node> Node

Saídas:

Date momentolirrigação
Double quantidadeIrrigação