

METODOLOGIA PARA CLASSIFICAÇÃO DE PILOTOS DE VEÍCULO BAJA SAE VIA REDES NEURAIAS

Email:

Abstract— Immersing in the motorsports context, the systems identification becomes a great ally to the racing team efficiency because it can extract important features from car/driver system e then provides feedbacks to a better performance. From that principle, this work aims to identificate drivers of an off-road Baja SAE vehicle. Through the partnership with the Car-Kará Baja SAE UFRN team, 4 different drivers have been selected in 7 different test tracks. The data has been collected, through an industrial data logger, and analyzed in an offline manner. From the data collection, it has been done a division of the variable vector in 3 and 5 sections and then it has been computed the statistical analysis for each part, creating the feature vectors. That vector was inserted in an artificial neural architecture with two hidden layer, obtaining a sucess rate of 97% for the division in 3 parts and 93% for 5 parts.

Keywords— System Identification, Statistical Data Analysis, Baja SAE, Artificial Neural Networks

Resumo— Imergindo-se no contexto dos esportes motorizados, a identificação de sistemas se torna uma grande aliada para o bom rendimento de uma equipe de corrida, pois pode extrair características importantes do sistema carro/piloto e assim fornecer *feedbacks* para melhor performance. Partindo deste princípio, este trabalho objetiva a classificação de pilotos de um veículo *off-road* Baja SAE. Através da parceria com a equipe Car-Kará Baja SAE UFRN, foram selecionados 4 pilotos diferentes em 7 pistas de testes. Os dados foram colhidos através de um data logger industrial e analisados de maneira *offline*. A partir da coleta dos dados, fez-se a divisão do vetor de variáveis em 3 e 5 partições e então foi computada análise estatística de cada parte, criando o vetor de características. Ele, por sua vez, foi inserido em uma arquitetura neural artificial de duas camadas ocultas, obtendo uma taxa de sucesso de 97% para o vetor de 3 divisões e 93% para o vetor de 5 divisões.

Palavras-chave— Identificação de Sistemas, Análise Estatística de Dados, Baja SAE, Redes Neurais Artificiais.

1 Introdução

A evolução da eletrônica automotiva tem chegado a um patamar onde os automóveis tornaram-se sistemas dotados de inteligência. O sensoramento automobilístico foi introduzido na década de 70, com o propósito de atingir níveis satisfatórios para emissões de gases poluentes e se deu ao longo de três “ondas”: a primeira se deu com o crescimento dos sensores aplicados ao sistema de *powertrain*, focando em reduções de emissões; a segunda foi pela demanda do mercado por performance e segurança; e a terceira foi introduzida com personalização para conforto de sistemas obrigatórios de segurança (GPS, air bags laterais, etc) (Fleming, 2001).

Diante desse cenário, diversos dispositivos estão espalhados pelo carro de tal forma que, no ano de 2008, existiam cerca de 100 sensores em um carro de luxo (Fleming, 2008). As unidades de controle eletrônico (*ECU - Eletronic Control Unit*) devem ler seus respectivos grupo de sensores e tomar decisões (troca de marcha, anti-travamento de rodas) dependendo de características externas.

Na principal categoria do automobilismo mundial, a Fórmula 1, o engenheiro de dados deve ficar atento as variáveis de cerca de 200 sensores do carro, que geram em torno de 300GB de dados (Goldwin, 2017). Dessa forma, a análise de dados e Big Data estão se tornando abordagens mais críticas nessa categoria. Equipes como Mercedes-Benz e Renault já estão com suas

apostas em projetos inteligentes, utilizando aprendizado de máquina, para a temporada de 2018 (Goldwin, 2017; Spencer, 2017). Assim, a determinação da melhor estratégia para a equipe passa a ser automatizado pela inteligência artificial, seja ela entrada no *pitstop*, ritmo de corrida, etc. Esse conceito de sistemas inteligentes também pode ser aplicado em outras disputas automobilísticas, como a Baja SAE.

A competição de âmbito acadêmico Baja SAE (*Society of Automotive Engineers*) ocorre tanto no escopo nacional quanto internacional e requer que as equipes participantes sigam algumas regras pré-estabelecidas quanto a dimensionamento e desempenho. O carro deve ser robusto para superar condições severas da competição tais como lama, trepidação, chuva, etc (SAE Brasil, 2018).

O projeto Baja SAE na Universidade Federal do Rio Grande do Norte (UFRN) teve início em 1995, sendo campeã mundial em 1998 (EUA) com a equipe Car-Kará, competindo com 86 países, conquistando um feito inédito para uma equipe estrangeira. No ano de 2018, a equipe Car-Kará conseguiu diversos prêmios individuais, obtendo a terceira melhor colocação do Brasil (SAE Brasil, 2018). O caráter competitivo torna os *feedbacks* ao piloto de grande importância, seja informando sobre comportamentos desnecessários ou identificando falhas do veículo. Com isso, este trabalho objetiva a classificação, através do perfil de dirigibilidade, de quatro pilotos de Baja SAE, fazendo uso de análise estatística para treinamento de Redes Neurais Artificiais. Dessa forma, o sis-

tema inteligente pode aprimorar o modo de condução do piloto para uma maneira mais competitiva e segura.

Assim, o restante deste trabalho será dividido como segue: seção 2 faz uma breve revisão teórica sobre redes neurais e análise estatística; a seção 3 aborda os trabalhos relacionados a este tema; a seção 4 destrincha o veículo Baja, estudo de caso; a seção 5 explica as materiais e métodos utilizados; a seção 6 mostra os resultados e, por fim, na seção 7 são descritas as conclusões.

2 Redes Neurais Artificiais e Análise Estatística de Dados

As Redes Neurais Artificiais têm como objetivo tornar-se semelhante ao cérebro humano em termos de funcionalidades. O conhecimento que a rede adquire com o tempo se dá com o processo de treinamento da mesma e a informação do aprendizado é armazenada nas conexões entre unidades neurais, ou pesos sinápticos (Haykin, 2001).

2.1 O Neurônio Artificial

O modelo do neurônio artificial proposto por McCulloch-Pitts é composto por diversos sinais de entradas e uma saída. Entre as entrada (Xs) e a saída (Y), existem outros elementos básicos que simulam a funcionalidade do neurônio biológico - o conjunto de sinapses ou pesos sinápticos ($W's$); o somador que faz a operação de soma do produto entre os sinais de entrada e os respectivos pesos sinápticos tem o nome de campo local induzido; a função de ativação $\Phi()$ que limita o valor do sinal de saída; e o bias, que consiste em uma entrada de valor constante, podendo ser positivo ou negativo, a fim de ajustar a entrada da função de ativação.

A relação entre as entrada e a saída do modelo neural é representada pela Eq. 1, na qual o termo $\sum_{i=1}^n x_i w_i$ é operador somador e b_i é o bias. Essas estruturas podem estar organizadas na forma de uma única ou várias camadas, como propõe o modelo Perceptron de Múltiplas Camadas de Rosenblatt.

$$y(t) = \Phi\left(\sum_{i=1}^n x_i(t)w_i + b_i\right) \quad (1)$$

2.2 Modelo Perceptron de Múltiplas Camadas

Em uma distribuição neural de camada única, encontra-se a mais simples arquitetura possível, descrita através do modelo perceptron proposto por Rosenblatt que incorporava algoritmos de treinamentos supervisionados para atualização dos pesos sinápticos (*backpropagation*) e que foram aplicados em problemas de classificação de padrões (Bishop, 1995).

Neste caso os neurônios se organizam em uma única coluna de nós que se projeta diretamente para a saída da rede. É válido ressaltar que a informação se propaga dos nós de entrada até a camada de saída dos elementos neurais. O caminho inverso jamais é possível (Haykin, 2001).

Um único neurônio consegue gerar, como sinal de saída, apenas valores entre 0 e 1. Dependendo da forma de organização da rede, um aumento da performance pode ser identificado, quando dispostas em uma ou mais camadas ocultas, aqui denotado por “H”.

Uma ou mais camadas ocultas nas redes neurais podem ser encontradas e tem como propósito atuar entre os nós de entrada e os de saída eficazmente. Com a presença da camada oculta numa rede neural, o sistema torna-se capaz de extrair características de alta complexidade. No entanto, quanto maior a complexidade do problema, maior será a quantidade de unidades processadoras de informação na camada oculta e consequentemente o aumento do tempo necessário para treinamento da rede (Bishop, 1995). A figura 2 mostra uma arquitetura neural genérica.

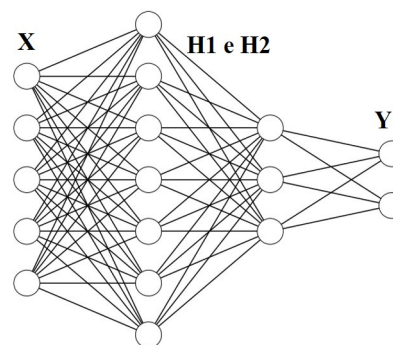


Figura 1: Arquitetura Neural de Múltiplas Camadas

O processo treinamento de uma rede neural, por sua vez, se dá a partir da atualização dos pesos sinápticos. A figura 1 descreve o procedimento. A partir de uma entrada X , a informação é passada pelas camadas da rede neural que gera um valor Y estimado em sua saída. Esse valor é então comparado com o valor real, definido pelo usuário, e essa comparação gera a função custo, que nada mais é do que um *feedback* para um sinal de erro. O campo otimizador utiliza-se desse sinal de erro para atualizar os pesos sinápticos (Chollet, 2018).

2.3 Análise Estatística de Dados

Um dos conceitos mais básicos da análise estatística são as *Variáveis Aleatórias*. Elas referem-se à números reais cujos valores descrevem um experimento aleatório seja ele discreto, quando assume valores finitos (e.g. jogar de uma moeda), ou contínuo - assumindo infinitas possibilidades.

A partir dela pode-se traçar algumas funções estatísticas. (Pishro-Nik, 2014).

O conjunto das probabilidades acumuladas de possíveis valores de uma variável aleatória contínua (X) é definido pela *Função de Distribuição Acumulada* e sua derivada é a *Função Densidade de Probabilidade*. Algumas dessas funções de distribuições são usadas com bastante frequência, pois modelam diversos sistemas físicos reais, e assim recebem nomes específicos (Pishro-Nik, 2014): uma dessas distribuições é a *Normal*.

Distribuições normais (ou normais gaussianas) são denotadas matematicamente por $X \sim N(\mu, \sigma^2)$, sendo X a variável aleatória que possui tal distribuição; μ e σ são a média e variância, respectivamente, da distribuição normal dada. Quando em um experimento com um número de amostras n , suficientemente grande, esses valores podem ser calculados através das Eq. 2 e Eq. 3. Além disso, se $\mu = 0$; $\sigma^2 = 1$ tal distribuição tem nome de *Normal Padronizada* (Papoulis and Pillai, 2002).

$$E[X] = \mu = \frac{\sum_{i=1}^n x_i}{n} \quad (2)$$

$$\sigma^2 = E[X^2] - E[X]^2 = \frac{\sum_{i=1}^n (x_i - \mu)^2}{n - 1} \quad (3)$$

No entanto, muitos sistemas físicos que são regidos por uma distribuição normal possuem diferentes médias e variâncias. Nesse caso é necessário deslocar essas características para que todas as variáveis aleatórias sejam regidas pela distribuição normal padronizada. Esse processo chama-se *padronização para escore* Z e pode ser entendido pela Eq. 4, sendo $X \sim N(\mu, \sigma^2)$ e $Z \sim N(\mu = 0, \sigma^2 = 1)$ (Papoulis and Pillai, 2002).

$$Z = X - \frac{\mu}{\sigma} \quad (4)$$

3 Trabalhos Relacionados

Propostas de hardware eletrônico também são encontradas na literatura. Em (Andria et al.,

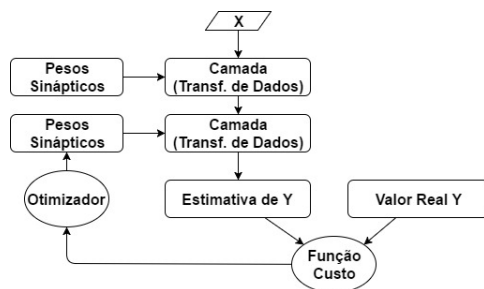


Figura 2: Processo de Atualização de Pesos Sinápticos (Chollet, 2018).

2016), propõe-se o uso de um microcomputador Raspberry pi conectado com um conversor OBD-II/Bluetooth para a análise dos dados de um carro comercial, através da rede CAN (*Controller Area Network*). É utilizado como extrator de características, funções como média e desvio padrão da aceleração longitudinal bem como a razão RPM/Velocidade, para identificar a mudança de marcha e assim o perfil do motorista. A saída do classificador é dada como: direção moderada ou direção agressiva.

Motivando-se pela identificação de pilotos a partir do comportamento de direção, (Quek and Ng, 2013) fez um comparativo entre Máquinas de Vetor de Suporte (SVM) e Regressão Logística Multinomial para a classificação de 6 pilotos de tal forma que os motoristas são identificados um por vez. Utiliza-se como vetor de características a velocidade do carro, o consumo de combustível, aceleração longitudinal e lateral e velocidade angular em curvas. Obteve-se 90% de sucesso com o uso da SVM e 84% para a regressão lógica multinomial.

Mais ainda, encontram-se também trabalhos com a utilização sensores embarcados em *smartphones*. Em (Van Ly et al., 2013), fez uso dos sensores inerciais de um smartphone, bem como dados reais através da rede CAN de um carro comercial, para a classificação de dois motoristas com a motivação de mapear o comportamento de direção dos usuários para fornecer *feedbacks* e evitar manobras perigosas. A classificação foi feita a partir da contagem de ocorrência de eventos como frenagem, aceleração e manobra em curvas que formam dois vetores de características, inserido em dois modelos: um com aprendizado supervisionado (Máquinas de Vetor de Suporte) e outro não-supervisionado (K-Means). A saída dos modelos é a classificação dos usuários como piloto 1 ou 2.

De maneira similar aos trabalhos anteriores, em (Meseguer et al., 2013) foi desenvolvida uma plataforma web através de um aplicativo para Android chamado DrivingStyles. O smartphone é conectado ao carro através de um scanner OBD-II/Bluetooth e envia os dados coletados da rede CAN para a nuvem. Variáveis médias como rpm, velocidade e aceleração são medidas em um período de amostragem de 10 segundos e então analisadas através de duas redes neurais, independentes, que fazem a identificação do tipo de terreno (rodovia de velocidade, estrada, urbano) bem como o grau de agressividade do modo de condução do motorista. A rede neural fornece uma taxa de sucesso de 98% para a classificação do tipo do terreno e 77% para o estilo de condução.

Quatro anos depois, em (Meseguer et al., 2017), utilizou-se da mesma plataforma web para coletar dados como fluxo de combustível, velocidade linear, entre outras variáveis, para medir o

consumo instantâneo de combustível e correlacionando com o estilo de dirigibilidade dos motoristas, mencionando o trabalho anterior.

Este presente trabalho irá abordar conceitos mencionados nesta seção, como uso de redes neurais artificiais em conjunto com conceitos estatísticos para a formulação do vetor de características. Aqui faz-se a separação entre outros trabalhos apresentados na literatura pois é feita a classificação quatro de pilotos de um veículo compacto *off-road* do tipo Baja SAE, ao invés veículos de terreno urbano.

4 Estudo de Caso - O Veículo Baja SAE

O Veículo utilizado como material de estudo neste trabalho foi o do tipo Baja SAE da equipe Car-Kará da UFRN. Ele é um veículo compacto *off-road* desenvolvido com o propósito exclusivamente acadêmico.

O carro é construído através de estrutura tubular em aço que oferece maior resistência a impactos e que reforça a segurança do condutor em caso de capotamento. Segundo regras oficiais da competição, a largura máxima não deve exceder 1.62m e não há restrições para o comprimento, porém, recomenda-se um comprimento máximo de 2.74m (SAE Brasil, 2018).

Em relação ao *powertrain*, o motor utilizado no veículo também é padronizado pela SAE. Trata-se um motor de combustão interna, à gasolina, monocilindro de 305 cilindradas produzido pela *Briggs & Stratton*. Já o sistema de transmissão de torque do motor para as rodas é de livre escolha para as equipes. No entanto, é bem comum encontrar transmissões do tipo CVT (*Continuous Variable Transmission*) nos carros tipo Baja SAE, podendo atingir velocidades máximas em torno de 60 km/h.

A eletrônica do veículo é composta por unidades de controle eletrônicas (ECU) dispostas ao longo do carro e são responsáveis pela leitura de diversas variáveis tais como RPM, velocidade linear, nível de combustível, entre outras. As centrais eletrônicas são interconectadas através do barramento de comunicação CAN e as informações são exibidas para o piloto através de um display TFT 2.8", como mostra a Figura 3.

No entanto, a versão atual da eletrônica presente no veículo ainda está em fases de aprimoramentos e não conta com saídas externas de informações para análise *off-line*. A fim de corrigir o problema de exportação da informação e análise dos dados, inseriu-se um data logger profissional para obtenção dos objetivos deste trabalho.

5 Metodologia

Nesta seção será descrito todo o procedimento para a obtenção do objetivo deste trabalho, desde



Figura 3: Exibição das informações ao piloto através de um Display TFT.

o a organização dos testes reais no veículo *off-road*, a captura dos dados através de um datalogger automotivo. Esses dados serão utilizados com o pós processamento para a análise de identificação *off-line* via Matlab.

5.1 Sistema de Aquisição de Dados

A captura dos dados embarcados no carro foi realizada através de um data logger profissional. O modelo é EVO 4 da fabricante italiana AIM (AiM Tech Srl, 2010), sendo comumente utilizado por equipes de Formula SAE, Karts e Bajas. Ele possui extrema robustez a intempéries do ambiente, podendo ser inserido em qualquer outro tipo de veículo. Além disso, seu *hardware* é extremamente poderoso pois conta com cinco canais de entrada para sensores analógicos (e.g. transdutores lineares para curso de suspensão) e digitais (sensor de velocidade linear) além de interfaces de hardware para conexões de redes automotivas tais como CAN e K-Line. Mais ainda, o aparelho ainda conta com um GPS interno de alta precisão e acelerômetros para os eixos x, y e z.

O sistema de aquisição de dados deve permanecer em um local imune a interferências eletromagnéticas (e.g. motor) e mecânicas (e.g. peças móveis) as quais podem distorcer a captura dos sinais analisados. Mais ainda, acelerômetros devem ficar tão próximos forem possíveis do centro de gravidade do veículo a fim de evitar valores com *offset*. Dessa forma, o data logger está fixado, através de parafusos em baixo do banco do piloto, ao assoalho do carro, como mostra a Figura 4. Conectado ao data logger, está um fio que vem do primário da bobina de ignição e vai até o canal de entrada "RPM/K-line" para a medição das revoluções do motor. Além desse, é necessário a conexão com os polos positivos e negativos da bateria (12V) para a alimentação do sistema. Os dados foram descarregados para o computador através de um cabo USB com o data logger.



Figura 4: Fixação do Data Logger no Veículo.

5.2 Preparo do experimento

A organização da proposta deste trabalho inicia-se a partir da discussão do melhor traçado de pista e da seleção dos pilotos de Baja. A seleção do perfil de pilotos dentre a equipe Car-Kará UFRN seguiu o padrão proposto em (Katzourakis et al., 2012), onde sugere-se a escolha de pilotos experientes e inexperientes para a formação do conjunto de dados. Ao todo, seis diferentes pilotos participaram do experimento de pilotagem de um mini carro *off-road* de tal forma que alguns estiveram em mais um teste, enquanto que outros estavam presentes em apenas dois deles. A fim de termos o maior número de amostras dos pilotos, foram selecionados apenas quatro deles, sendo dois pilotos experientes (P1 e P3), um com nível intermediário (P2) e um novato (P4), como retratado pela Tabela 1.

Tabela 1: Distribuição dos pilotos nas pistas.

	P1	P2	P3	P4
Pista 1	X	X	X	
Pista 2	X	X	X	
Pista 3	X	X	X	
Pista 4	X	X	X	
Pista 5	X	X		X
Pista 6	X	X	X	X
Pista 7	X	X	X	X

Além disso, para a formação da pista de teste, foram selecionados alguns espaços gramados nas dependências do campus universitário da UFRN. Neles foram feitas sete traçados, em dias diferentes, que serão chamados de pista 1, pista 2, ..., pista 7. Todas elas têm diferentes tipos de obstáculos como areia fofa, troncos e rampas. Cada um dos pilotos teve a possibilidade de rodar algumas voltas em cada pista e, após o término delas, foi feita a aquisição dos dados. Com a aquisição de dados foi feita a classificação dos pilotos *off-line*.

5.3 Classificação dos pilotos

O procedimento de classificação dos pilotos pode ser dividido em dois estágios: organização e montagem do conjunto de dados; e otimização dos parâmetros de criação da Rede Neural Artificial.

5.3.1 Organização e Montagem do Conjunto de Dados

No primeiro estágio, foi utilizado o software Race Studio 2, da própria AIM Sports, para a exportação dos dados das voltas dos pilotos, em formato de espaço de trabalho para Matlab. Esse processo é repetido continuamente para todas as pistas de cada piloto. Dentre as variáveis possíveis de serem exportadas, foram selecionadas cinco. Elas estão descritas na Tabela 2

Tabela 2: Variáveis Utilizadas

Variável	Descrição
Engine [RPM]	Rev. por minuto do motor
GPS_LatAcc.[g]	Aceleração lateral de acordo com o GPS
GPS_LonAcc.[g]	Aceleração longitudinal de acordo com o GPS
GPS_Speed [km/h]	Velocidade linear
GPS_Gyro [deg/s]	Velocidade angular

Em seguida os dados são importados para tratamento no Matlab. Já nesse *software*, os dados referentes as diferentes pistas de um mesmo piloto são concatenado, transformando em uma matriz de células, na qual cada coluna representaria uma pista e cada célula dessa coluna seriam as variáveis descritas anteriormente. Os dados também são normalizados a fim de se ter média zero e variância unitária, através da Eq. 4.

Uma vez com a matriz de células, é feita a divisão dos vetores colunas em 3 ou 5 sessões, e em cada parte é computada algumas estatísticas, listadas abaixo, de todas as variáveis presentes na Tabela 2, sendo “i” variando de 1 até 3 ou 5 divisões.

- Média (setor i);
- Variância (setor i);
- Valor máximo (setor i);
- Valor mínimo (setor i);

Esse processo é similar ao encontrado em (Van Ly et al., 2013). Em seguida, a concatenação dos vetores das estatísticas de cada setor, gera o vetor características de cada piloto que será, então, entrada do modelo neural.

A Figura 5 refere-se a sequência lógica para a criação do conjunto de dados de treinamento e teste utilizados na rede neural.

5.3.2 Otimização dos Parâmetros da Rede Neural

Após o conjunto de dados ser salvo, entra-se no segundo estágio. Aqui, faz-se a importação dos

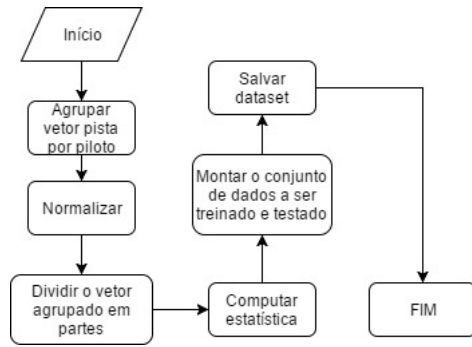


Figura 5: Fluxograma para a montagem do conjunto de dados.

dados e fazemos a divisão entre dados para treinamento da rede neural (75%) e dados para teste (25% de amostras não vistas anteriormente).

A fim de se encontrar os melhores parâmetros da rede neural (número de neurônios nas camadas ocultas), foram criados dois *scripts*: um que incrementava apenas o número de neurônios da primeira camada oculta, chamando de $H1$. Essa variável foi iniciada com valor de 10 e seu valor final foi setado como 80, o passo incremental é de 2 neurônios. Já o outro *script* otimizava tanto a primeira camada oculta $H1$, quanto a segunda camada oculta ($H2$). Determinou-se o valor inicial de $H1_{inicial} = 10$, o valor final $H1_{final} = 80$ e passo incremental de 4. Da mesma maneira, determinou-se o valor inicial de $H2$ como 2 e valor final como 50, com passo incremental também de 4 unidades.

A rede então é treinada através do algoritmo *Gradient Descent Backpropagation*, com os números de neurônios ocultos, na interação atual. As funções de ativação utilizadas nas camadas ocultas são do tipo sigmóide, enquanto que a camada de saída utiliza-se da *softmax*, por ser um problema de classificação (Bishop, 1995). A função custo utilizada é o *erro médio quadrático*. Além disso, é utilizada a parada antecipada, de atualização dos pesos sinápticos, por erro de validação, a fim de aprimorar a taxa de generalização do classificador.

Em seguida, a rede é testada com amostras não vistas anteriormente e assim o erro de teste é verificado e caso o valor seja o mínimo até então, a rede é salva e o número $H1$ é incrementado. O fluxograma que representa o processo treinamento e teste da rede neural que otimiza apenas a primeira camada oculta está retratado na Figura 6.

Do mesmo modo ocorre quando executado o *script* que otimiza a segunda camada oculta neurônios. A rede é testada com amostras não vistas anteriormente e se o erro de teste for menor que o mínimo, a rede é salva e o valor de $H2$ é incrementado para a próxima interação. Quando $H2$ atinge o valor máximo o valor de $H1$ passa a ser incrementado também. O fluxograma que representa o processo treinamento e teste da rede

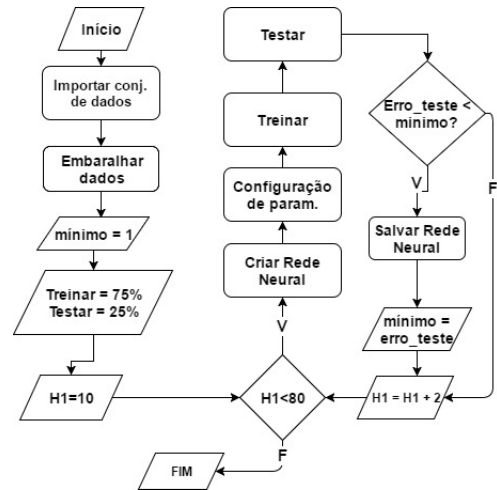


Figura 6: Fluxograma do processo de treinamento da primeira camada oculta da rede neural.

neural que otimiza apenas as camadas ocultas 1 e 2 está retratado na Figura 7.

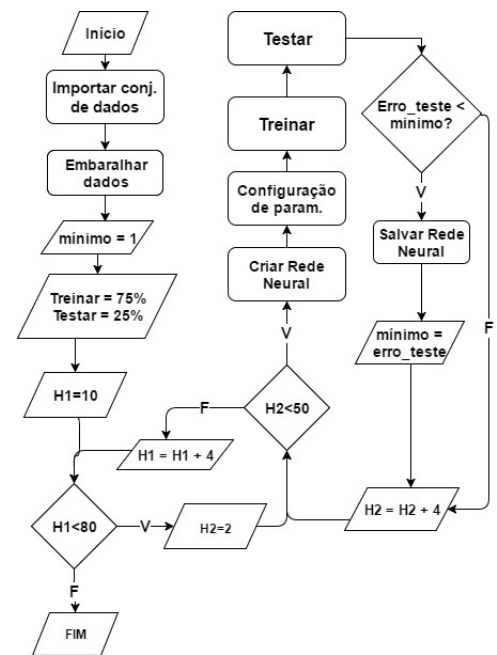


Figura 7: Fluxograma do processo de treinamento das camadas ocultas 1 e 2 da rede neural.

Uma vez executado o *script* de otimização, é chegado o momento de verificar a consistência dos parâmetros. Dessa forma, roda-se um outro programa que importa a rede neural salva e retesta-a com o conjunto de dados embaralhados.

6 Resultados

Utilizando-se do procedimento descrito anteriormente, quatro experimentos foram testados: dois com a divisão das células do vetor de características para 3 e 5 partes através da rede neural treinada apenas com uma camada oculta; outros dois

testes foram feitos com a rede neural com duas camadas ocultas.

Para o primeiro tipo de experimento, não se verificou uma taxa de classificação satisfatória. Encontrou-se que 87,5% das amostras foram classificadas corretamente para a divisão do vetor de características em 3 partes. Mais ainda, apenas 81% de amostras foram classificadas com êxito quando testado o conjunto de dados a partir da divisão em 5 partes do vetor de características.

No entanto, para o segundo tipo de experimento, executou-se o *script* de busca dos melhores parâmetros e foi encontrado que para valores $H1=34$ e $H2=26$ tem-se uma taxa de sucesso em torno de 97%, para a divisão do vetor de características em 3 partes. Enquanto que quando executado o teste em que as variáveis do vetor de características são divididas em 5 partes, encontrou-se um novo conjunto ótimo de parâmetros, sendo $H1=64$ e $H2=22$. Com esse experimento, a taxa de sucesso para o conjunto de teste ficou em torno de 93%.

É válido ressaltar que a melhor configuração para o teste em uma única camada ocorreu quando o vetor de características foi dividido em 5 partes. No entanto, quando executado o teste com a interação para duas camadas ocultas, a melhor configuração ocorreu com o vetor de características dividido em 3 partes.

Mais ainda, a Figura 8 retrata as Densidades de Distribuição de Probabilidade dos erros de teste para os experimentos em 3 e 5 divisões. É possível perceber que para o teste com 3 divisões, a probabilidade de ocorrência dos erros entre 0 e 20% é maior do que quando comparado com o teste em 5 divisões.

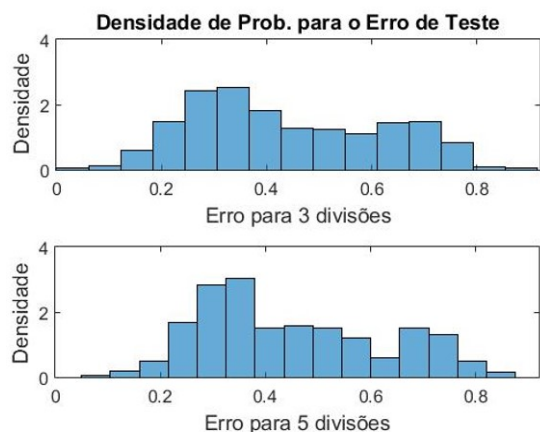


Figura 8: Densidades de probabilidade dos erros de teste

Apesar de um reduzido tamanho de amostras, esse resultado demonstra que a metodologia utilizada mostrou ser efetiva quando comparada com os resultados encontrados por (Van Ly et al., 2013), que utilizando o método k-means ob-

teve uma taxa de sucesso em torno de 80% e 60% para o método de Máquinas de Vetor de Suporte (SVM).

Por fim, ainda foram executados testes de classificação com combinações dos seis pilotos tomadas duas a duas. Esse experimento é semelhante ao feito em (Quek and Ng, 2013). A taxa de acerto nas combinações demonstraram bastante satisfatórias tanto para a divisão em 3 quanto em 5 sessões, de tal forma que obteve-se 100% de classificação em ambos os casos, para a maioria das combinações entre pilotos. Apenas para o teste entre os pilotos 2 e 3 que a taxa de acerto ficou em torno de 94% para a divisão de 3 setores e em torno de 97% para a divisão de 5 setores. A Tabela 3 sumariza os resultados obtidos neste trabalho.

Tabela 3: Taxa de acerto da Rede Neural para o experimento dos quatro pilotos.

Nº de camadas Ocultas	Divisões	Sucesso
Uma camada	5 divisões	87.50%
	3 divisões	81.30%
Duas camadas	5 divisões	93.80%
	3 divisões	96.90%

7 Conclusões

Em resumo, com o uso da metodologia proposta neste trabalho, uma alta taxa de sucesso na classificação, tanto nas combinações de pilotos tomados dois a dois quanto na classificação geral utilizando o conjunto de dados dos quatro indivíduos, foi alcançada.

É válido ressaltar que o teste da rede neural com apenas uma única camada oculta demonstrou performance média inferior quando comparado ao teste executado em uma rede neural com duas camadas ocultas, o que faz sentido por causa da melhor extração de características do modelo.

Os resultados alcançados a partir de duas camadas ocultas de neurônios e a divisão do vetor de características em 3 partes demonstraram um aumento em 20% de classificações quando comparado com outras propostas semelhantes a esta, mesmo utilizando-se de um cenário *off-road*. Dessa forma, confirmam-se que os testes foram satisfatórios e atingiu-se o objetivo proposto.

Alguns trabalhos futuros, em relação ao *feedback* de comportamento de direção aos pilotos envolvidos, são propostos. Eles podem ser entendidos como diminuição/aumento do ritmo de corrida, previsão de tempo de parada nos boxes, entre outros. Propõe-se também o aprimoramento do sistema de aquisição de dados do veículo da Equipe Car-Kará, a fim de confrontar as informações adquiridas pelo data logger da AIM Sports com a arquitetura de *hardware* desenvolvida até então. Caso as informações sejam consistentes,

seria interessante rodar a rede neural a partir dos dados da eletrônica Car-Kará e comparar os resultados com este trabalho.

Além disso, é pertinente o teste com outros algoritmos de *Machine Learning*, encontrados na literatura, tais como: SVM, Regressão Logística, Camadas Ocultas de Markov, entre outros.

Agradecimentos

Os autores deste trabalho gostariam de agradecer a equipe Car-Kará de Baja SAE, pela disponibilização do veículo de competição e dos pilotos para a execução dos testes; a UFRN, pela infraestrutura e apoio acadêmico; e ao CNPq, pelo apoio financeiro.

Referências

- AiM Tech Srl (2010). Evo 4 datalogger, <http://www.aim-sportline.com>. Acessado em 21/02/2018.
- Andria, G., Attivissimo, F., Di Nisio, A., Lanzolla, A. M. and Pellegrino, A. (2016). Development of an automotive data acquisition platform for analysis of driving behavior, *Measurement: Journal of the International Measurement Confederation* **93**: 278–287.
- Bishop, C. (1995). *Neural networks for pattern recognition*, Vol. 103.
- Chollet, F. (2018). *Deep learning with Python*, Manning Publications, New York, NY.
- Fleming, W. J. (2001). Overview of automotive sensors, *IEEE Sensors Journal* **1**(4): 296–308.
- Fleming, W. J. (2008). New automotive sensors - A review, *IEEE Sensors Journal* **8**(11): 1900–1921.
- Goldwin, B. (2017). Formula 1: Mercedes team turns to big data in quest for victory in grand prix, <http://www.computerweekly.com/feature/Formula-1-Mercedes-team-turns-to-big-data-in-quest-for-victory-in-Grand-Prix>. Acessado em 07/02/2018.
- Haykin, S. (2001). *Redes Neurais: Princípios e práticas*, 2 edn, Bookman.
- Katzourakis, D. I., Velenis, E., Abbink, D., Happee, R. and Holweg, E. (2012). Race-car instrumentation for driving behavior studies, *IEEE Transactions on Instrumentation and Measurement* **61**(2): 462–474.
- Meseguer, J. E., Calafate, C. T., Cano, J. C. and Manzoni, P. (2013). DrivingStyles: A smartphone application to assess driver behavior, *Proceedings - International Symposium on Computers and Communications* pp. 535–540.
- Meseguer, J. E., Toh, C. K., Calafate, C. T., Cano, J. C. and Manzoni, P. (2017). Drivingstyles: A mobile platform for driving styles and fuel consumption characterization, *Journal of Communications and Networks* **19**(2): 162–168.
- Papoulis, A. and Pillai, S. (2002). *Probability, random variables and stochastic processes*, McGraw-Hill, McGraw-Hill.
- Pishro-Nik, H. (2014). *Introduction to Probability, Statistics, and Random Processes*, Kappa Research, LLC.
- Quek, Z. F. and Ng, E. (2013). Driver Identification by Driving Style, pp. 1–4.
- SAE Brasil (2018). Competição Baja SAE Brasil - Regras e Relatórios, <http://portal.saebrasil.org.br/programas-estudantis/baja-sae-brasil>. Acessado em 06/03/2018.
- Spencer, G. (2017). Artificial intelligence and formula one: Bots on pole position in the race for technology, <https://news.microsoft.com/en-my/2017/09/19/artificial-intelligence-formula-one-bots-pole-position-race-technology>. Acessado em 07/02/2018.
- Van Ly, M., Martin, S. and Trivedi, M. M. (2013). Driver classification and driving style recognition using inertial sensors, *IEEE Intelligent Vehicles Symposium, Proceedings (Iv)*: 1040–1045.