

# Implementando Redes Neurais para Classificação de Tendências do Mercado de Troca de Divisas

Rodrigo Picinin Boratto, Luis Augusto Mattos Mendes

Departamento de Ciência da Computação – Universidade Presidente Antonio Carlos  
(UNIPAC)

Barbacena – MG – Brasil

rwrz@yodsoft.com, luisaugustomendes@yahoo.com.br

**Resumo.** *Este artigo consiste na criação de processos para classificação do mercado de troca de divisas utilizando inteligência artificial de forma a reconhecer padrões harmônicos já existentes.*

**Palavras-Chave:** *forex; inteligência artificial; rede neural; mercado internacional de divisas; padrão harmônico.*

## 1. Introdução

Um dos problemas da ciência é a previsão do futuro; “*Dado o passado, como poderemos prever o futuro?*” [1]. Uma tentativa clássica de previsão seria criar um modelo baseado no passado e esperar que ele se repita no futuro. O interesse pela previsão se torna mais interessante quando envolve dinheiro, tal como prever os números da sena, quando a ação vai subir, ou quando ela vai cair. Temos ainda neste campo de previsão, o mercado de troca de divisas, ou apenas mercado *Forex* (*Foreign Exchange*), que vamos abordar neste trabalho.

Este mercado *Forex*, de acordo com *Triennial Central Bank Survey of Foreign Exchange and Derivatives Market* de abril de 2007, movimentava 3.2 trilhões de dólares ao dia [2]. Este valor, comparado a outros mercados, representa uma grande liquidez, como pode ser visto no texto “Introdução ao Mercado de Foreign Exchange”, onde afirma que todos os mercados de ações dos Estados Unidos combinados não alcançam 3% do volume total negociado no mercado *Forex*. Ao contrário dos outros mercados, no *Forex* não é necessário lucrar apenas no crescimento do mesmo, pode-se lucrar tanto em operações de compra (*Long*) e venda (*Short*), que em comparação com os outros mercados, isso o torna mais volátil e arriscado mercado do mundo [3].

Para se diminuir os riscos do mercado *Forex* adotam-se várias técnicas: indicadores técnicos<sup>1</sup>, análises fundamentalistas<sup>2</sup>. A análise técnica<sup>3</sup> é o estudo do histórico de informações do mercado de preço e volume de forma que tentar prever o futuro do mesmo, já a análise fundamentalista se baseia nas notícias recentes do mercado e nas influências que as mesmas refletem neste [3]. Estes tipos de análises

---

<sup>1</sup> Analisa-se o passado com cálculos matemáticos, tais como a média dos últimos dias, volume de compra e venda, entre outros.

<sup>2</sup> Ao contrário da análise técnica (ou de indicadores), se baseia em notícias que beneficiam o mercado de alguma forma, tais como índice de desemprego de um país ou a taxa de juros do mesmo.

<sup>3</sup> Análise que utiliza-se dos indicadores técnicos.

ajudam a especular o mercado *Forex* de forma a se basear no histórico passado e em estatísticas, esperando que se repitam no futuro. Utiliza-se da união das mesmas para tentar obter lucros, porém sempre em algum momento “o mercado muda”<sup>4</sup> sua tendência e faz com que os especuladores errem e percam dinheiro.

Entretanto existem alguns autores que afirmam que existem certos padrões que se repetem com maior precisão no mercado *Forex*, estes são denominados padrões harmônicos [6]. O primeiro padrão harmônico encontrado [6] foi o denominado *Gartley Pattern*, que foi publicado em 1935 por H.M. Gartley em seu livro *Profits in the Stock Market* [12]. Segundo Derek, este padrão é muito eficiente e funciona desde 1935 com uma média de acertos de 70% (informação verbal)<sup>5</sup>. Além deste, temos os padrões conhecidos como *Bat Pattern*, *Crab Pattern* e *Butterfly Pattern* [8], e todos procuram especular para qual direção o mercado vai, se vai subir ou se vai cair.

Visto a existência destes padrões e de acordo com alguns autores os mesmos funcionam [6] [8] [12] e com uma margem de acertos maior que 70% [13], há maneiras de definir tendências neste mercado e encontrar padrões com o uso de cálculos matemáticos. Em consequência disso, seria possível também o uso de inteligência artificial<sup>6</sup> para identificação destes padrões, já que elas também se baseiam em cálculos matemáticos.

Muitos trabalhos buscam implementar várias técnicas conhecidas de inteligência artificial para prever o mercado *Forex*, porém aparentemente nenhum trabalho publicado obteve resultados que garantissem com precisão a previsão de uma tendência ou padrão de mercado [1] [4] [5]. Mas estes autores procuraram encontrar novos padrões, ao invés de tentar melhorar os resultados dos padrões já existentes.

Neste trabalho, diferentemente, tenta-se identificar padrões de mercado sem interesse na previsão de valores futuros. Tem-se como intenção encontrar os padrões harmônicos através da inteligência artificial, em especial o padrão de Gartley (por ser o primeiro padrão encontrado), no mercado *Forex*, pois de acordo com alguns estudos a inteligência artificial é uma boa alternativa para resolver problemas complexos e indefinidos. Ela também é capaz de aprender com exemplos, é tolerante a falhas, de forma que consegue lidar com ruídos, aceita dados incompletos, trabalha com problemas não lineares, e uma vez treinada, pode dar o resultado em alta velocidade [15] [16] (a baixo custo computacional). Sendo assim, a inteligência artificial torna-se ideal para encontrar padrões em um meio “caótico” como o mercado *forex* e melhorar os algoritmos lineares atualmente desenvolvidos para encontrar padrões harmônicos [10].

Este artigo foi dividido em seções para uma maior organização, a seção 2 tem como objetivo explicar melhor os padrões harmônicos, não só o Padrão de Gartley, mas também outros padrões para um melhor entendimento. A seção 3 tem como objetivo explicar sobre a inteligência artificial, métodos de aprendizado máquina, quais foram utilizados e por que. E por último, na seção 4, os resultados obtidos e a conclusão deste artigo.

---

<sup>4</sup> Expressão comumente utilizada entre especuladores, para expressar que a análise atual não serve para situação atual do mercado.

<sup>5</sup> Palestra apresentada por Derek Frey, em um seminário online sobre Padrões Harmônicos em 28 de Maio de 2009 [13].

<sup>6</sup> A Inteligência Artificial (IA) é uma área de pesquisa da ciência da computação dedicada a buscar métodos ou dispositivos computacionais que possuam ou simulem a capacidade humana de resolver problemas, pensar ou, de forma ampla, ser inteligente.

## 2. Revisão Bibliográfica

Esta seção busca revisar o conhecimento necessário para compreensão da parte prática e os resultados obtidos, ou seja, as seções 3 e 4.

### 2.1. Padrões Harmônicos

Os padrões harmônicos, segundo Carney [6], são alinhamentos específicos de preços que ocorrem em forma de ciclos, e que ocorreram em um histórico passado. A utilização destes padrões em mercado é denominado *Harmonic Trading*.

Também segundo Carney, *Harmonic Trading* é uma metodologia que utiliza o reconhecimento destes padrões de preços e na maior parte das vezes os alinham com as variações de *Fibonacci*<sup>7</sup> para determinar regiões muito prováveis de reversão no mercado financeiro, ou seja, onde o preço deve sofrer uma alta ou queda. Essa metodologia assume que padrões de operação ou ciclos se repetem como em vários padrões ou ciclos que repetem na vida. A chave é identificá-los para se poder entrar com uma operação no mercado de grande probabilidade, baseado nesse mesmo ciclo ou padrão que ocorreu em um histórico passado. Mesmo que esses padrões não sejam 100% certos, essas situações foram provadas historicamente. Se os padrões forem identificados corretamente é possível identificar oportunidades significativas com um risco bem limitado.

#### 2.1.1. Gartley Pattern

Como observado anteriormente, este padrão foi publicado em 1935 por H.M. Gartley em seu livro *Profits in the Stock Market* [12]. A análise técnica deste padrão visa identificar momentos de compra (*bullish*) ou venda (*bearish*) no mercado. A Figura 1 demonstra como o padrão funciona.

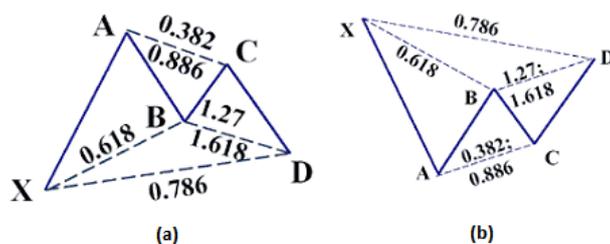


Figura 1. Gartley Pattern [6]

Na Figura 1, letra a, observa-se uma tendência de alta XA com uma reversão<sup>8</sup> em A. Utilizando as variações de *Fibonacci*, a descida AB deverá ter 61.8% da diferença de A menos X como demonstrado pela linha em XB. Em B, o preço sofre uma reversão novamente e de preferência BC deverá estar entre 61.8% e 78.6% da diferença de A menos B, como mostrado em AC. Em C, o preço sofre outra reversão com uma queda entre 127% e 161.8% da diferença de B menos C, como mostrado na linha BD. O valor em D é o ponto de **compra**, pois o preço deverá subir de acordo com o padrão [7].

<sup>7</sup> Referencia a série de *Fibonacci*.

<sup>8</sup> Quando o preço tende para um lado (cima/baixo) e logo em seguida ocorre um movimento contrário.

Na letra b desta mesma Figura 1, observam-se reversões contrárias, indicando em D o ponto de **venda**.

### 2.1.2. Butterfly Pattern

Este padrão foi descoberto por Larry Pesavento e Bryce Gilmore em 1991 [6]. Ele segue as mesmas regras do Padrão de *Gartley*, o ponto de **compra/venda** continua sendo o ponto D, as mudanças estão apenas nas relações entre os pontos X, A, B, C e D, como pode ser observado na Figura 2.

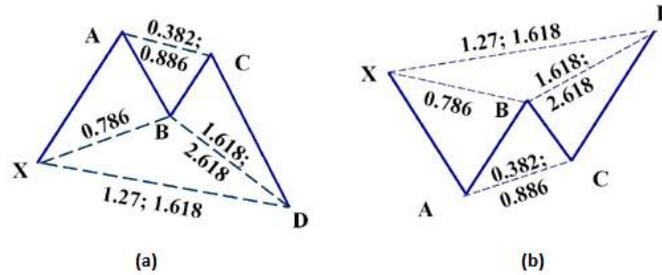


Figura 2. *Butterfly Pattern* [6]

### 2.1.3. Crab Pattern

Este padrão foi criado por Scott Carney em 2000. É um dos padrões de maior potencial segundo Scott Carney [6]. Seu ponto de **compra/venda** continua sendo o ponto D e as especificações funcionam da mesma forma que o padrão de Gartley, com mudanças ainda apenas nas relações entre os pontos, como pode ser observado na Figura 3.

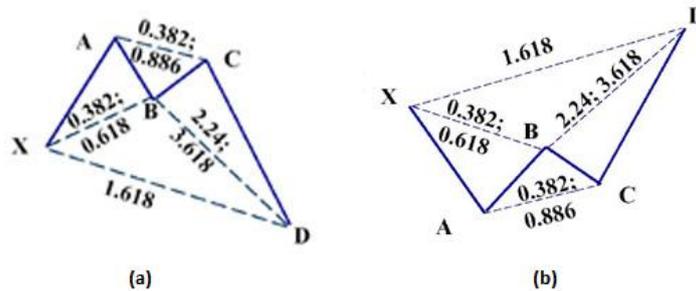


Figura 3. *Crab Pattern* [6]

### 2.1.4. Bat Pattern

Este padrão também foi criado por Scott Carney [6], porém em 2001. Tem um grande potencial de acerto segundo Scott Carney. O ponto de **compra/venda** também é o ponto D e as especificações também seguem o padrão do Gartley. Como pode ser observado na Figura 4.

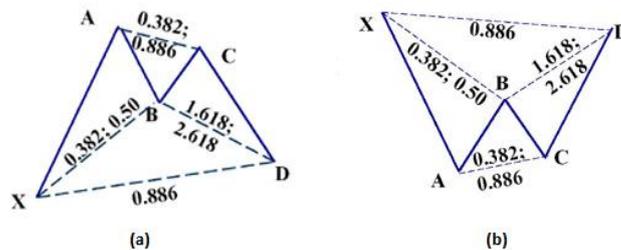


Figura 4. Bat Pattern

## 2.2. Inteligência Artificial e Forex

Como foi dito anterior, normalmente a Inteligência Artificial (IA) é utilizada para se criar algoritmos complexos, pouco definidos, que possam ter ruído nos dados ou até mesmo dados incompletos. Devido o *Forex* ser um mercado com muito ruído em seus dados, é que foi escolhida esta tecnologia para detecção dos padrões.

A técnica de IA utilizada neste artigo são as Redes Neurais, por serem comumente utilizadas para previsão de mercado *Forex*, e onde se encontra os melhores resultados [4] [5] [15] [16] [4].

A rede neural consiste em capturar os princípios básicos de manipulação de informação do cérebro humano e aplicar esse conhecimento na resolução de problemas que exigem aprendizado a partir da experiência. As redes neurais artificiais se fundamentam nos estudos sobre a estrutura do cérebro humano para tentar emular sua forma inteligente de processar informações. Apesar de se desconhecer a maneira pela qual o cérebro manipula informações complexas, sabe-se que a modelagem do conhecimento contido em um problema específico pode ser representada através de interconexões entre células nervosas. Estruturalmente a rede neural artificial se assemelha à rede neural biológica pela composição de seus neurônios e pela conexão entre eles [11].

Por meio destas conexões as redes neurais artificiais aprendem por meio adaptativo de tentativa e erro, ou seja, através do erro de seus resultados, ela procura aprender e gerar um novo resultado com um erro menor. Basicamente, existem dois mecanismos de aprendizado para redes neurais: aprendizado supervisionado e aprendizado não-supervisionado [17].

As redes neurais, com suas conexões e mecanismos de aprendizado, necessitam de entradas, em ambos os mecanismos, e saídas, apenas no aprendizado supervisionado, para serem treinadas. As entradas são as informações que passamos às redes de forma que ela busca a correlação entre os valores, como no caso das redes não-supervisionadas, ou que ela altere o valor dos pesos de suas conexões até conseguir extrair as saídas previamente determinadas, como no caso das redes supervisionadas. Estas entradas e saídas, segundo Kondratenko e Kuperin [4], não devem depender de unidades de medidas e devem ser reduzidas a uma unidade de escala, ou seja, devem ser normalizadas.

### **2.2.1. Redes não-supervisionadas, aprendizado não-supervisionado**

Segundo Fonseca [17], estas redes têm a capacidade de determinar uma correlação entre os possíveis padrões de entrada e são particularmente úteis nos problemas em que as entradas variam com o tempo de forma conhecida. Pode-se considerar este mecanismo de aprendizado como sendo a capacidade que a rede possui de abstrair correlações entre os estímulos de modo a obter as respostas desejadas.

Este tipo de rede é capaz de identificar padrões apenas pelas correlações das entradas, sem qualquer intervenção no aprendizado. Por este motivo, é utilizado neste artigo como tentativa de reconhecimento dos padrões harmônicos.

### **2.2.2. Redes supervisionadas, aprendizado supervisionado**

Segundo Fonseca [17], maioria absoluta das aplicações existentes compõe-se de redes neurais com aprendizado supervisionado, que pode ser considerado como a capacidade que a rede possui de modificar o seu desempenho a partir da comparação entre a resposta obtida e a resposta desejada.

Desta forma, é possível passar entradas e as saídas desejadas, ou seja, respostas que se deseja obter com a rede neural e esta procura adaptar as entradas de forma a obter as respostas, saídas. Por este motivo, é utilizado neste artigo como tentativa de reconhecimento dos padrões harmônicos.

## **3. Metodologia**

Para se treinar uma rede neural existe alguns pontos que devem ser observados primeiramente: entradas, saídas, e formas de normalização. Esta seção será dividida em duas etapas, uma por mecanismo de aprendizado. No mecanismo de aprendizado não-supervisionado existem mais três divisões, a normalização das entradas e tipos de algoritmos utilizados e resultados obtidos, e no mecanismo de aprendizado supervisionado cinco divisões, normalização das entradas, tipo de algoritmo, entradas e saídas, explicação da ferramenta e resultados obtidos.

### **3.1. Aprendizado não-supervisionado**

A intenção de utilização deste tipo de aprendizado é que os padrões harmônicos existem justamente por possuir uma correlação especial entre seus pontos.

#### **3.1.1. Normalização das entradas**

Nas redes não-supervisionadas não é necessário haver uma normalização profunda das informações, ou seja, temos uma maior flexibilidade de possíveis entradas e podemos utilizar quaisquer valores, desde que sejam numéricos. Como dito anteriormente, normalizar é necessário, segundo Kondratenko e Kuperin [4], pois as entradas e saídas não devem depender de unidades de medidas e devem ser reduzidas a uma unidade de escala.

As entradas para as redes não-supervisionadas escolhidas foram:

1. Diferença do preço de abertura do *candle*<sup>9</sup> atual – preço anterior;
2. Seqüência de diferenças do preço de abertura do *candle* atual – preço anterior;

A escolha do item 1 foi feita como uma forma de normalização, visto que um valor de preço tem entre 3 e 5 casa decimais, e de acordo com Kondratenko e Kuperin [4], os valores para aprendizado devem oscilar entre 0 e 1. Nesta tentativa é obtido sempre um valor entre -1 e 1, próximo ao valor ideal proposto por Kondratenko.

A escolha do item 2 foi feita baseado no aprendizado no histórico passado, como em outros trabalhos [1] [4], desta forma tem-se todos os pontos dentro do período e espera-se que a rede seja capaz de encontrar suas correlações.

### 3.1.2. Tipos de algoritmos

O aprendizado não-supervisionado possui vários algoritmos de escolha destes numeros correlacionados. Neste artigo será utilizada a ferramenta Weka<sup>10</sup> [18], e ela oferece muitas escolhas, porém foram escolhidas duas:

1. *K-means*
2. *DBScan*

O primeiro algoritmo, segundo Pichiliani, tem como objetivo fornecer uma classificação de informações de acordo com os próprios dados. Esta classificação é baseada em análise e comparações entre os valores numéricos dos dados. Desta maneira, o algoritmo automaticamente vai fornecer uma classificação automática sem a necessidade de nenhuma supervisão humana, ou seja, sem nenhuma pré-classificação existente. Para gerar as classes<sup>11</sup>, o algoritmo faz uma comparação entre cada valor por meio da distância. Geralmente utiliza-se a distância euclidiana<sup>12</sup> para calcular o quão ‘longe’ uma ocorrência está da outra. Após o cálculo das distâncias o algoritmo calcula centróides para cada uma das classes. Conforme o algoritmo vai iterando, o valor de cada centróide é refinado pela média dos valores de cada atributo de cada ocorrência que pertence a este centróide. Com isso, o algoritmo gera k centróides e coloca as ocorrências de acordo com sua distância dos centróides. Assim cada classe é definida pela sua proximidade.

O terceiro algoritmo possui um foco diferente, visto que o K-means analisa apenas 2 planos e traça a distancia baseado neste plano bi-dimensional, o DBScan faz uma busca espacial no plano tri-dimensional. Consegue encontrar padrões criando e inserindo as informações em um plano com profundidade [19]. Por existir essa possibilidade tri-dimensional, foi escolhido.

---

<sup>9</sup> Forma gráfica para representar um agrupamento de preços do mercado em certo período, um candle é composto por 4 itens: preço de abertura, maior preço, menor preço e preço de fechamento.

<sup>10</sup> Software comumente utilizado para se criar redes neurais supervisionadas e não supervisionadas.

<sup>11</sup> Classe é o resultado obtido pelo aprendizado não-supervisionado, também chamado de *cluster*. [1] [17]

<sup>12</sup> Em matemática, distância euclidiana (ou distância métrica) é a distância entre dois pontos, que pode ser provada pela aplicação repetida do teorema de Pitágoras. Aplicando essa fórmula como distância, o espaço euclidiano torna-se um espaço métrico. [22]

### 3.1.3. Resultados obtidos

Primeiramente, deve-se definir qual o resultado esperado. Neste tipo de aprendizado, busca-se que o algoritmo seja capaz de definir quais são os pontos do padrão de *Gartley* e os colocar em uma classe separada do resto.

O primeiro teste foi utilizando K-means e a diferença de preços como entrada. Pode-se observar o resultado na Figura 5.



**Figura 5 – Resultado obtido com k-means e diferença de preços.**

Como se pode observar na Figura 5, foi esperado 2 classes: padrão harmônico e outros. Porém o algoritmo trouxe uma separação dos valores de forma a dividi-los em 2 grupos distantes, porém não é o resultado esperado para este trabalho, pois não demonstra nenhum padrão harmônico.

O segundo teste foi utilizando K-means e os dados seqüenciais de preço como entrada. Pode-se observar o resultado na Figura 6.



**Figura 6 – Resultado obtido com k-means e preços seqüenciais.**

Como se pode observar na Figura 6, os resultados, mesmo aumentando o número de classes, continuam a dividi-las pela proximidade dos preços e não pela sua “correlação harmônica”. Apareceram vários grupos, todos separados pela proximidade de preços.

O terceiro e último teste foi utilizando o DBScan e dados sequenciais como entradas. Pode-se observar o resultado na Figura 7.



**Figura 7 – Resultado obtido com DBScan e preços sequenciais.**

Neste último teste, pode-se observar que os valores estão mais alternados e a relação entre eles ainda continua pela distancia. Então seu resultado também não é satisfatório para este artigo.

Como se pode observar, nenhum dos algoritmos foi capaz de encontrar algum tipo de padrão harmônico conhecido. Os algoritmos não-supervisionados classificaram o mercado pela distância dos dados e não conseguiram identificar padrões propostos.

### **3.2. Aprendizado supervisionado**

Comumente o aprendizado supervisionado é utilizado para previsão em mercado *Forex*, como já foi citado anteriormente neste artigo. Devido a isso, foi escolhido este tipo de aprendizado, e devido a capacidade de aprender problemas complexos. Visto também que o aprendizado não-supervisionado não obteve resultados satisfatórios.

#### **3.2.1. Normalização das entradas**

Os algoritmos supervisionados, segundo Kondratenko e Kuperin [4], precisam de entradas normalizadas compreendidas entre 0 e 1. Devido a isso, todas as entradas descritas nesse trabalho para aprendizado supervisionado foram divididas por 10. Assim, um preço que possua valor “1,5555”, passará a ter o valor “0,15555”. Esta forma de normalização resolve a necessidade de se ter os números entre 0 e 1.

#### **3.2.2. Tipo de algoritmo**

Analisados em outros trabalhos [4] [5] [11] [15] [17], o algoritmo mais utilizado é o *Backpropagation* e este é o que se obtém os melhores resultados.

Segundo Beale citado por Alessandro Mueller [11], o *Backpropagation* funciona da seguinte forma: ao se apresentar um determinado padrão de entrada a uma rede neural não treinada e o respectivo padrão de saída, uma saída aleatória é produzida. A partir da saída produzida pela rede é calculado um erro, representando a diferença entre o valor obtido e o desejado. O objetivo consiste, então, em reduzir continuamente o erro até um determinado valor aceitável. Isto é alcançado pelo ajuste dos pesos entre as conexões dos neurônios pela aplicação da regra Delta Generalizada<sup>13</sup>, que calcula o erro

<sup>13</sup> A regra delta implementa um gradiente descendente no quadrado da soma do erro para funções de ativação lineares [23].

para alguma unidade particular e propaga esse erro para a camada anterior. Cada unidade tem seus pesos ajustados de modo a minimizar o erro da rede.

Por considerar o *Backpropagation* capaz de aprender padrões de mercado através destes ajustes de pesos entende-se que o mesmo será suficiente para o escopo deste trabalho.

### 3.2.3. Entradas e saídas

A escolha das entradas e das suas respectivas saídas é primordial para o aprendizado das redes supervisionadas. São utilizados neste trabalho três tipos combinações de entradas e saídas para o treinamento das mesmas, como podem ser vistas na Tabela 1.

#	Entradas	Saídas
1	Valor de fechamento do <i>candle</i> nos pontos X, A, B e C	Ponto D.
2	Seqüência de 50 maiores e menores preços de cada <i>candle</i>	Os cinco pontos do padrão harmônico, X, A, B, C e D
3	Valor(es) do(s) ponto(s) anteriores e seqüência de 46 a 50 maiores ou menores preços de cada <i>candle</i> .	Apenas um dos pontos, B, C ou D.

**Tabela 1 – Entradas e saídas das Redes Neurais**

No primeiro item, são colocados os valores dos pontos X, A, B e C e busca-se o ponto D. Desta forma, seria necessário apenas passar informação de quatro preços para se obter o quinto e principal ponto.

No segundo item, acredita-se que através de um conjunto de informações do passado poderá se encontrar correlações entre certos itens do conjunto de forma a encontrar o padrão harmônico. Então, se passa informação de 50 *candles* e espera-se que a rede aprenda quais desses *candles* representa um padrão harmônico.

No terceiro item, cria-se três redes neurais, uma para encontrar cada ponto. Na primeira rede, são informados 48 *candles* mais os pontos X e A como entrada e espera-se o valor de B. Na segunda rede, são informados 47 *candles* mais os pontos X, A e B, e espera-se o valor de C. Na terceira rede, são informados 46 *candles* mais os pontos X, A, B e C, e espera-se o valor de D. Desta forma, é possível popular uma rede com a saída da outra e no final obter todos os pontos do padrão harmônico.

As informações de preços utilizadas para as entradas foram “capturadas” utilizando uma plataforma de mercado financeiro, chamada *Metatrader4*<sup>14</sup> [20]. Esta plataforma possibilita a criação de algoritmos de análise financeira em tempo real e a mesma possui um algoritmo de licença livre que identifica os padrões harmônicos [10]. Para este trabalho foi utilizado este algoritmo, chamado ZUP<sup>15</sup> [10], e modificado<sup>16</sup> [9]

<sup>14</sup> Software para operação em mercado financeiro, utilizada também para desenvolver aplicações para o mercado [20].

<sup>15</sup> Algoritmo feito para plataforma *Metatrader4* para reconhecimento de padrões harmônicos.

<sup>16</sup> Inseriu-se o código necessário para extrair as informações do indicador para arquivo, visto que é necessário para se treinar as redes. Nenhuma modificação na essência do mesmo foi feita. [9]

de forma que sempre que o padrão de *Gartley* ocorra no mercado, as entradas descritas na Tabela 1 são “capturadas” e armazenadas em arquivo.

Com estas informações foram criadas duas bases para utilização nas redes neurais: para o treinamento das redes, período de 01/07/2009 até 31/07/2009 e para efetuar os testes, período de 01/08/2009 até 31/08/2009. Para cada base foram armazenadas suas respectivas entradas e saídas, já normalizadas, em arquivo e depois salvas em formato correto para utilização no *software* “FANN” [21]. Mais informações sobre a ferramenta utilizada podem ser encontradas na próxima sessão.

### 3.2.4. Ferramenta utilizada

Os treinamentos e testes foram feitos utilizando a API de Redes Neurais FANN e sua interface gráfica “FannExplorer”. A FANN é uma biblioteca gratuita de redes neurais que implementa redes neurais artificiais de várias camadas em C com suporte para redes conectadas ou não. Ela é multi-plataforma e aceita valores fixos ou de ponto flutuante. Ela inclui um *framework* de fácil manuseio para se treinar grandes grupos de dados. Também é fácil, versátil, bem documentada e principalmente rápida [21].

Através da sua interface, “FannExplorer”, é possível criar redes neurais sem ter que escrever nenhuma linha de código, apenas é necessário alterar as configurações do *software*. As configurações utilizadas para todos os testes foram de acordo com a Tabela 2.

Propriedade	Valor
Training algorithm	Resilent
Error function	Tanh
Cascade training	Marcado
Max number of neurons	200
Stop meaning square error falls below	$1 * 10^{-7}$

**Tabela 2 – Configurações das Redes Neurais**

A primeira propriedade refere-se ao algoritmo que será utilizado para o treinamento das redes. Esta opção, *Resilent*, significa que utilizaremos o *Backpropagation* de forma *Resilent*, que significa que o erro obtido em uma iteração será repassado entre as camadas anteriores como se fosse uma nova entrada. A função de erro *Tahn*, refere-se ao calculo de erro utilizado.

A terceira propriedade, *Cascade Training*, é o treinamento em cascata. Com esta opção marcada não é necessário definir o número de camadas, nem o número de neurônios, a única coisa que é necessária é a quarta propriedade, *Max number of neurons*, e a quinta, *Stop meaning square error falls below*, que define até quantos neurônios a rede poderá ter antes de exibir os resultados ou a margem de erro cair abaixo da quinta propriedade. Essa propriedade consiste em um algoritmo que, por tentativa e erro, procura as melhores funções de ativação e topologia das redes, com um único foco: encontrar um bom resultado com pequena margem de erro e o mínimo de neurônios possíveis. Este algoritmo faz alterações na rede até esbarrar em um dos valores, ou na propriedade quatro ou na propriedade cinco, ou seja, ou se tem até 200 neurônios na rede ou se encontra um erro abaixo de  $1 * 10^{-7}$ .

### 3.2.5. Resultados obtidos

Após selecionadas as entradas, e configurado a ferramenta da forma correta, inicia-se os treinamentos e testes. Após algumas horas de treinamentos, através do algoritmo de “*Cascade training*”, obtiveram-se topologias diferenciadas para cada rede, conforme a Tabela 3.

#	No. Input Neurons	No. Output Neurons	No. Layers			No. Total Neurons			No. Total of Connections		
1	4	1	23			27			341		
2	100	5	38			142			4951		
3	50	1	B	C	D	B	C	D	B	C	D
			8	3	8	58	53	58	378	103	378

**Tabela 3 – Topologia das Redes Neurais**

Pode-se observar na Tabela 3 a topologia das redes neurais: no item #1, tem-se 4 neurônios de entrada no total, 1 neurônio de saída, 23 camadas, sendo 1 de entrada, 1 de saída e 21 invisíveis, um total de 27 neurônios e um total de 341 conexões; no item #2 tem-se a mesma disposição mas com valores diferentes; já no item #3, tem-se em número de camadas, total de neurônios e total de conexões tabelas com as colunas B, C, D, isto acontece pois no item #3 tem-se na verdade 3 redes neurais, onde B é a rede que procura o resultado do PONTO B, e assim sucessivamente. Então, para a rede do item #3, ponto B, temos 8 camadas, 58 neurônios totais e um total de 378 conexões; para os outros pontos, tem-se a mesma disposição mas com valores diferentes.

Finalmente, os resultados obtidos após inúmeras iterações podem ser observados na Tabela 4.

#	Square Error		
1	2.7157411 * 10 <sup>-8</sup>		
2	5.6676181 * 10 <sup>-6</sup>		
3	B	C	D
	1.6241628 * 10 <sup>-7</sup>	8.0490338 * 10 <sup>-8</sup>	1.6187609 * 10 <sup>-7</sup>

**Tabela 4 – Resultados Obtidos**

De acordo com o mercado *Forex*, um valor de preço possui até 4 casas decimais (10<sup>-4</sup>) e este valor normalizado para o aprendizado da rede foi dividido por 10, então considera-se que a margem de erro ideal seria abaixo da quinta casa, ou seja, menor que 10<sup>-5</sup>. De acordo com a Tabela 4, todas as redes obtiveram resultados satisfatórios e com uma margem de erro abaixo da esperada, como por exemplo, a rede item #1, obteve 2.7157411 \* 10<sup>-8</sup>, o que demonstra um erro de até 0,000000027157411 para cima ou para baixo do valor procurado.

A rede neural de número 1 obteve menor erro, onde a mesma procura encontrar o valor do ponto D. Desta forma, a rede consegue prever que os dados quatro pontos do padrão de *Gartley*, onde estará o quinto elemento. Sendo que ao unir todos os cinco

elementos teremos o padrão de Gartley. As redes neurais de numero dois e três também obtiveram ótimos resultados.

#### **4. Considerações Finais**

Este trabalho prova que as redes neurais são capazes de identificar padrões já conhecidos em mercado *Forex* e confirma a existência dos mesmos. Obteve sucesso no treinamento das redes e resultados com erros abaixo do esperado, como pode ser observado na Tabela 4.

Obtiveram-se resultados não satisfatórios com as redes não supervisionadas, sendo que as mesmas não foram capazes de encontrar quaisquer padrões harmônicos aqui descritos. Devido a isso, continuou-se o trabalho utilizando redes supervisionadas, das quais se obteve sucesso e bons resultados.

Porém mesmo com os resultados satisfatórios não significa que podem ser utilizados no mercado em tempo real. As redes são capazes de identificar que os padrões existem dadas as entradas, mas não são capazes de identificar exatamente onde estão. Os resultados aqui não foram testados e comprovados no mercado e como trabalho futuro propõe-se a utilização das redes feitas neste artigo utilizando a mesma plataforma, *Metatrader4* [20], por um período de tempo de dois meses juntamente com o ZUP [10], comparando os resultados de ambos os algoritmos para se comprovar se os resultados são realmente sólidos.

#### **Referências Bibliográficas**

- [1] Pavlidis, N.G., Plagianakos, V.P., Tasoulis, D.K. and Vrahatis, M.N. *Financial Forecasting through Unsupervised*, Greece, 2006.
- [2] *Triennial Central Bank Survey of Foreign Exchange and Derivatives Market Activity in April 2007 – Preliminary global results*. Disponível em: <<http://www.bis.org/publ/rpfx07.htm>>. Acesso em 5 mar. 2009.
- [3] *Introdução ao Mercado de Foreign Exchange*. Disponível em: <<http://www.avafx.com/pt/resources/educ/forex-intro/>>. Acesso em 17 jun. 2009.
- [4] Kondratenkol, V.V. and Kuperin, Yu.A. *Using Recurrent Neural Networks To Forecasting of Forex*, USA.
- [5] Diaconescu, E. *The use of NARX Neural Networks to predict Chaotic Time Series*, ROMANIA.

- [6] Carney, M. Scott. *Harmonic Trader*. Disponível em: <<http://www.harmonictrader.com/>>. Acesso em 17 jun. 2009.
- [7] Investopedia – Forbes digital company. *Gartley Pattern*. Disponível em: <<http://www.investopedia.com/terms/g/gartley.asp>>. Acesso em 17 jun. 2009.
- [8] La Canada Capital Management, LLC. *The Butterfly Pattern*. Disponível em: <<http://harmonicedge.com/butterfly.html>>. Acessado em 17 jun. 2009.
- [9] BORATTO, P. Rodrigo. *Algoritmo para identificação de padrões harmônicos*. Disponível em: <[http://www.yodsoft.com/tcc/anexo\\_1.html](http://www.yodsoft.com/tcc/anexo_1.html)>. Acessado em 23 nov. 2009.
- [10] Chrisch. *ZUP Indicator*. Disponível em: <<http://zupindicator.com/>>. Acessado em 2 nov. 2009.
- [11] MUELLER, Alessandro. *Uma aplicação de redes neurais artificiais na previsão do mercado acionário*. Disponível em: <<http://www.eps.ufsc.br/disserta96/mueller/index/index.htm>>. Acessado em 2 nov. 2009.
- [12] GARTLEY, Harold M. *Profits in the Stock Market.*, 1935.
- [13] FREY, Derek. *Webminar about Harmonic Patterns*. Apresentado em 28 de maio de 2009.
- [14] CROSS, Sam Y. *The Foreign Exchange Market in the United States*, 1998. Disponível em: <<http://www.newyorkfed.org/education/addpub/usfxm/>>. Acessado em 7 de dezembro de 2009.
- [15] KALOGIROU, Soteris. *Artificial intelligence systems in energy and renewable energy applications*, 2008. Disponível em: <[http://www.scitopics.com/Artificial\\_intelligence\\_systems\\_in\\_energy\\_and\\_renewable\\_energy\\_applications.html](http://www.scitopics.com/Artificial_intelligence_systems_in_energy_and_renewable_energy_applications.html)>. Acessado em 7 de dezembro de 2009.
- [16] ZETTLEMOYER, Luke S., PASULA, Hanna M., KAELBLING, Leslie P. *Learning Planning Rules in Noisy Stochastic Worlds*.
- [17] FONSECA, Ijar M.. *Inteligência Artificial - Redes Neurais*.
- [18] University of Waikato. *Weka*. Disponível em: <<http://www.cs.waikato.ac.nz/ml/weka/>>. Acessado em 7 de dezembro de 2009.
- [19] ESTER, Martin, KRIEGEL, Hans-Peter, SANDER, Jiirg, XU, Xiaowei. *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*.
- [20] Metaquotes. *Metatrader 4*. Disponível em: <<http://www.metaquotes.net/>>. Acessado em 7 de dezembro de 2009.
- [21] *FANN*. Disponível em: <<http://leenissen.dk>>. Acessado em 7 de dezembro de 2009.
- [22] Distância Euclidiana. Disponível em: <[http://pt.wikipedia.org/wiki/Dist%C3%A2ncia\\_euclidiana](http://pt.wikipedia.org/wiki/Dist%C3%A2ncia_euclidiana)>. Acessado em 7 de dezembro de 2009.

[23] MENDES FILHO, Elson F.. *Uma Introdução à Redes Neurais Artificiais*.  
Disponível em: <[http://elson\\_mendes.sites.uol.com.br/rn/rn\\_backp.html](http://elson_mendes.sites.uol.com.br/rn/rn_backp.html)>. Acessado  
em 7 de dezembro de 2009.