

UNIVERSIDADE EDUARDO MONDLANE

Faculdade de Ciências

Departamento de Matemática e Informática

IT-7

Matemática e Informática

Aplicação da Inteligência Artificial  
na Gestão de Alterações  
de Apólices

BIBLIOTECA U. E. M.  
R. 8 9920  
DATA 14.9.2004  
AQUISIÇÃO  
COTA IT-7

Amina Bacar

IT-7

R.E. 9920

# Trabalho de Licenciatura

## *Aplicação da Inteligência Artificial na Gestão de Alterações de apólices*

**Autor:** Amina Carina Momade Bacar

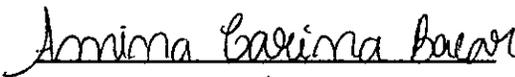
**Supervisor:** Dr. Enrique Rosa

Junho, 1997

## Declaração de Honra

“Declaro que o presente trabalho é o resultado das minhas próprias investigações e que o mesmo foi realizado apenas para ser submetido como trabalho de **Licenciatura em Informática** na **Universidade Eduardo Mondlane**”.

Maputo, aos 30 de Junho de 1997

  
(Amina Carina Momade Bacar)

## Agradecimentos

Em obediência aos impulsos causados pela minha humilde rede neural, sinto-me na obrigação de agradecer a todos que tornaram possível a realização deste trabalho.

Ao Dr. Enrique Rosa que aceitou a difícil tarefa de supervisionar este trabalho.

À Exi, em particular ao eng.º José Murta e ao eng.º Orlando Grosso, pelo fornecimento do material necessário à minha investigação.

À Emose, em particular ao eng.º Chembeze pelo fornecimento da informação necessária à investigação.

Ao dr.º Aurélio Macamo, à dr.ª Marisa Balas, à Dr.ª Magally M., ao Sr. Alfredo Norton, à Sr.ª Naima Valigy, à Sr.ª Chehnaz Calumia, ao Sr. Kalpendra Meggi e ao Sr. Nilesh Meggi, pela atenção dispensada durante a realização do trabalho.

Um agradecimento especial vai para os meus pais e irmãos pela ajuda e apoio dispensado durante o curso e durante a realização do trabalho.

E alguém muito especial, muito obrigado pela força.

## Tabela de Símbolos e Abreviaturas

<b>Símbolo / Abreviatura</b>	<b>Significado</b>
Emose	Empresa Moçambicana de Seguros, E.E
Exi	Engenharia e Comercialização de Sistemas Informáticos
IA	Inteligência Artificial

## Índice

<b>RESUMO</b> .....	<b>1</b>
<b>INTRODUÇÃO</b> .....	<b>2</b>
<b>1. DEFINIÇÃO DO PROBLEMA</b> .....	<b>4</b>
1.1. ALTERAÇÕES DE APÓLICES E SUAS DEFINIÇÕES.....	10
1.2. CASOS DE ESTUDO.....	13
<b>2. INTELIGÊNCIA ARTIFICIAL</b> .....	<b>18</b>
2.1. SISTEMAS BASEADOS NO CONHECIMENTO .....	20
2.1.1. <i>Representação do conhecimento</i> .....	23
2.2. REDES NEURAIS .....	31
2.2.1. <i>Conceitos gerais</i> .....	33
2.2.2. <i>Processo de desenvolvimento de uma rede neural</i> .....	37
2.3. COMPARAÇÃO ENTRE SISTEMAS BASEADOS NO CONHECIMENTO E REDES NEURAIS .....	43
<b>3. SOLUÇÃO DO PROBLEMA</b> .....	<b>46</b>
3.1. PAPEL DO SISTEMA BASEADO NO CONHECIMENTO NA ANÁLISE DAS ALTERAÇÕES.....	49
3.2. USO DE UMA REDE NEURAL NA CLASSIFICAÇÃO DAS ALTERAÇÕES.....	51
3.2.1. <i>Aprendizagem da rede</i> .....	53
<b>4. CONCLUSÕES E RECOMENDAÇÕES</b> .....	<b>58</b>
<b>5. REFERÊNCIAS BIBLIOGRÁFICAS</b> .....	<b>60</b>
<b>ANEXOS</b> .....	<b>62</b>
1. PROPOSTA DE CONTRATO .....	62
2. APÓLICE DE SEGURO .....	65
3. ACTA ADICIONAL.....	66
4. PSEUDOCÓDIGO PARA O CÁLCULO DO PRÉMIO SIMPLES DA APÓLICE .....	67
5. PSEUDOCÓDIGO PARA O CÁLCULO DO PRÉMIO TOTAL DA APÓLICE.....	68
6. PSEUDOCÓDIGO PARA O CÁLCULO DO VALOR DO RECIBO SUPLEMENTAR .....	69
7. PSEUDOCÓDIGO PARA O CÁLCULO DO VALOR DO RECIBO DE ESTORNO.....	70
8. LISTAGEM DE ALTERAÇÕES DE APÓLICES .....	71
9. BIBLIOGRAFIA NÃO REFERENCIADA .....	72

## **Resumo**

Embora o Ramo Automóvel da Emose E.E, Empresa Moçambicana de Seguros, esteja informatizado, algumas incongruências nas alterações dos dados da apólice de seguro, provocados em muitos casos por uma incorrecta ordem na sua execução, deixam a empresa insatisfeita.

A qualquer momento, um segurado pode solicitar um grupo de alterações aos dados da sua apólice de seguro. Ao se executar as alterações solicitadas numa determinada sequência, obtém-se um resultado que pode ser diferente do obtido usando uma outra sequência. Assim, é importante que se determine a sequência correcta de execução das alterações, com vista a obter um resultado correcto.

O presente trabalho propõe uma solução para a determinação dessa sequência, usando técnicas de Inteligência Artificial, combinando num único sistema os conceitos sobre sistemas baseados no conhecimento e redes neurais.

## **Introdução**

As empresas recorrem à informática com o objectivo de vencer a crescente concorrência no mercado, e obter altos níveis de eficiência na prestação dos seus serviços.

O crescente número de empresas a usarem computadores pessoais e sistemas digitais leva à expansão da informática, exigindo que os sistemas operem de uma forma simples e inteligente.

É nesta óptica que a Empresa Moçambicana de Seguros, Emose E. E., solicitou a Exi, Engenharia e Comercialização de Sistemas Informáticos, à informatização gradual da empresa.

Apesar da informatização do ramo automóvel, que funciona auxiliando-se no Sistema Integrado de Gestão de Seguros - SIGES, elaborado pela Exi, as alterações dos dados da apólice de seguro continuam incoerentes. O resultado da execução de um grupo de alterações numa apólice, pode ser diferente se a sequência em que elas são efectuadas não for a mesma.

O problema consiste em determinar a sequência adequada de execução das alterações, que conduza a um resultado correcto. Não existe definido um método que faça isso. Os técnicos de seguro obtêm essa sequência analisando a informação existente sobre cada alteração, a sua relação com as restantes e baseando-se na experiência de trabalho que possuem.

Uma forma de obter a sequência correcta é utilizando técnicas de Inteligência Artificial pois, ela lida com problemas onde o conhecimento sobre o domínio do problema e a experiência de um grupo de pessoas é relevante.

A solução apresentada usa um sistema híbrido, juntando um sistema baseado no conhecimento e uma rede neural. O sistema baseado no conhecimento analisa as alterações solicitadas e fornece a informação que a rede neural necessita para a classificação de cada alteração.

O objectivo geral deste trabalho é propor uma solução, que possibilite a determinação da ordem correcta de execução das alterações dos dados da apólice de seguro, usando técnicas de Inteligência Artificial.

Assim, os objectivos específicos deste trabalho são:

- ☛ Identificar técnicas de Inteligência Artificial para a resolução de problemas de natureza não algorítmica e que dependem da experiência de um grupo de pessoas.
- ☛ Propor uma solução ao problema da determinação da sequência correcta de execução das alterações, usando o resultado do objectivo anterior.

De forma a obter informação sobre o problema em questão, durante a realização do trabalho, foram efectuadas diversas entrevistas aos técnicos de seguro da Emose e aos analistas de sistemas do SIGES

A proposta de uma solução ao problema foi obtida através de consultas à Internet, à bibliografia disponível e a informação adquirida através das entrevistas.

Certas figuras deste relatório, foram elaboradas usando alguns diagramas do método orientado a objectos (Booch, 1994).

Com vista a alcançar os objectivos traçados, o relatório foi dividido em três capítulos. O primeiro apresenta uma descrição exhaustiva do problema, fornecendo os conceitos necessários para a sua compreensão. A importância do problema e a necessidade de solucioná-lo são ilustradas através de casos de estudo.

O capítulo dois fornece algumas técnicas de Inteligência Artificial, mais concretamente dos sistemas baseados no conhecimento e sistemas de redes neurais, indicando as suas aplicações.

Finalmente, o capítulo três, propõe uma solução do problema exposto no capítulo um, usando os conceitos fornecidos pelo capítulo dois. A solução apresentada junta, num único sistema, os conceitos de sistemas baseados no conhecimento e redes neurais.

## 1. Definição do problema

A Empresa Moçambicana de Seguros, Emose E. E., dedica-se a actividade de seguros de diversos ramos. Existem ramos a funcionar manualmente, como por exemplo o ramo de transportes e engenharia, ramos semi-informatizados, como é o caso do ramo de vida e acidentes de pessoais e ramos já informatizados como é o caso do ramo automóvel.

O Sistema Integrado de Gestão de Seguros – SIGES – é um sistema informático que permite o registo, consulta e actualização de informação necessária para a actividade ligada ao seguro.

O SIGES controla toda a documentação necessária para a actividade de seguro, tais como a emissão da apólice de seguros, avisos de cobrança, diversos recibos, actas adicionais, cartas dirigidas aos segurados, e outros, para além de manter, durante um certo período, vários arquivos com informação de interesse ligada aos documentos, objecto seguro e segurados. O SIGES permite a emissão de um conjunto de mapas associados aos sinistros e ao controlo das indemnizações e prémios.

O funcionamento da actividade de seguro do ramo automóvel, já informatizado, é feito sob o controle do SIGES; uma representação dessa actividade é dada pela figura 1.2.

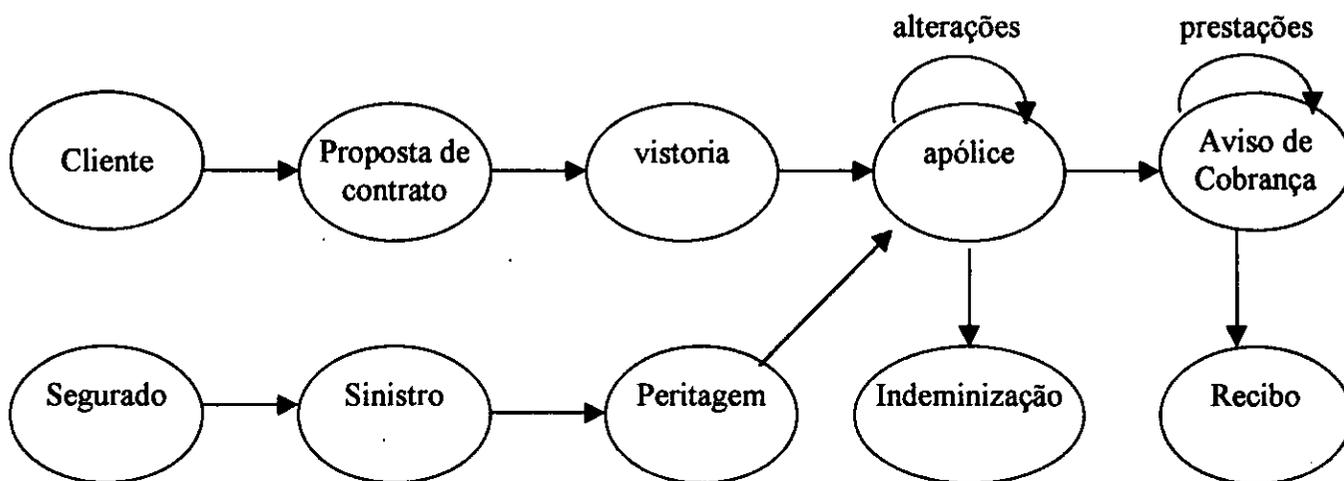


Figura 1.2: Representação da actividade do seguro automóvel (Macamo,1996)

Um seguro no ramo automóvel inicia quando um cliente chega à Emose a fim de segurar a sua viatura. Ele preenche um formulário designado de proposta de contrato, que tem como informação,

os dados do cliente, os valores dos capitais a segurar e os dados da viatura em causa, como mostra o anexo 1.

Depois da entrega da proposta do contrato, faz-se uma vistoria à viatura. Se o parecer da vistoria e dos responsáveis dos ramo forem favoráveis, a proposta de contrato é registada no SIGES e é emitida a apólice de seguro e o recibo correspondente a primeira prestação do prémio.

**Definição 1.1 (Apólice de Seguro):-** Uma Apólice de Seguro é um documento escrito descrevendo o contrato de seguro. O anexo 2 mostra um exemplo de uma apólice de seguro de automóveis.

**Definição 1.2 (Prémio da Apólice):-** O Prémio da Apólice é o valor pago pelo cliente ao abrigo de um contrato de seguros (Karim [b]). Ele é calculado aplicando uma taxa ao capital assegurado pelo cliente. A taxa aplicada depende do tipo de cobertura do seguro: seguro de responsabilidade civil ou seguro de danos próprios.

**Definição 1.3 (Seguro de Responsabilidade Civil):-** O Seguro de Responsabilidade Civil é o seguro que é feito contra terceiros; ele tem três coberturas distintas (Karim [a]):

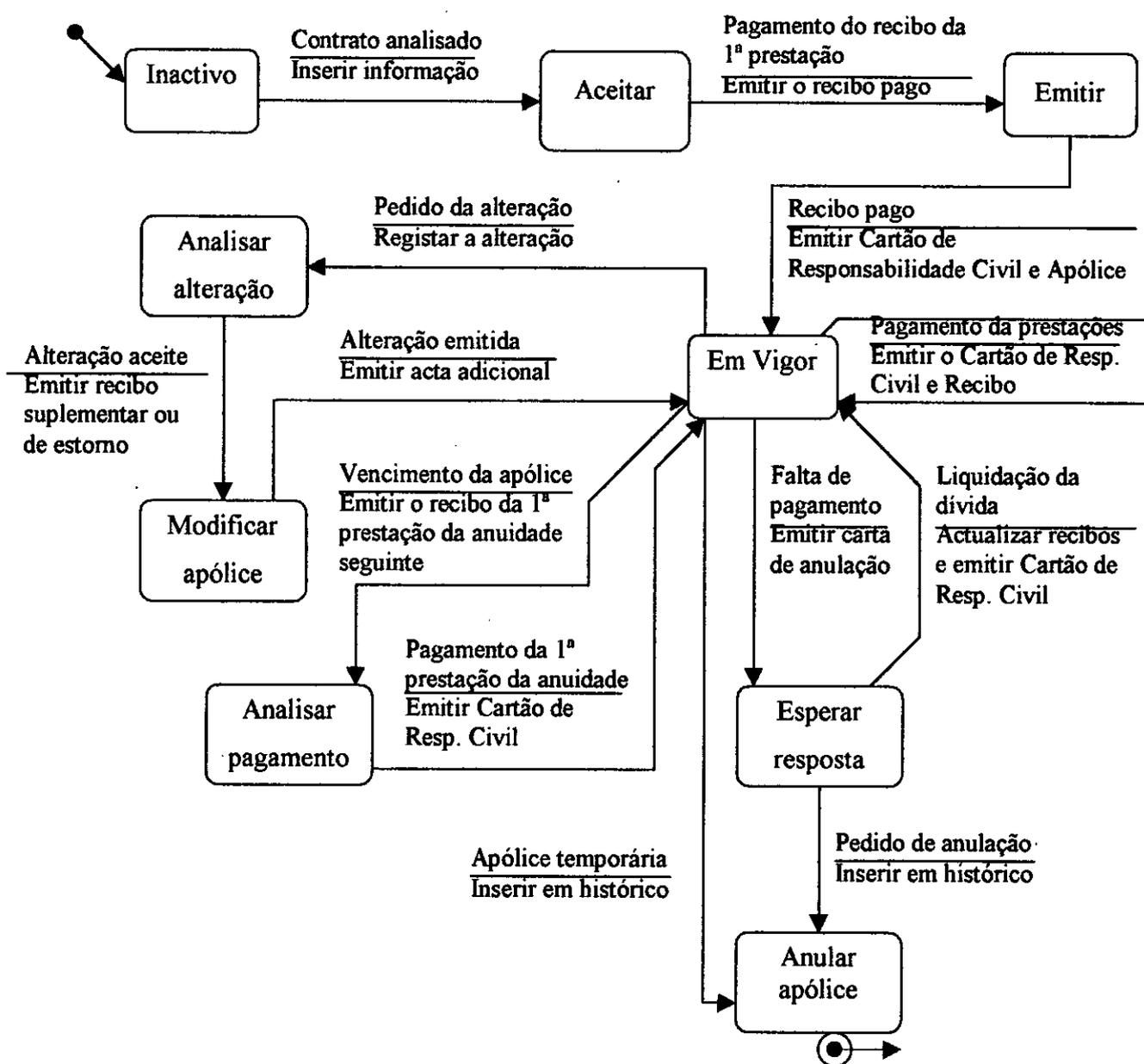
1. Responsabilidade civil por prejuízos ou danos causados (fisicamente ou no património) a terceiros não passageiros de transportes colectivos. Esta cobertura é obrigatória para todo o seguro automóvel.
2. Responsabilidade civil por prejuízo ou danos causados a passageiros de veículos usados em transportes colectivos.
3. Responsabilidade civil por acidentes corporais sofridos pelos passageiros transportados gratuitamente.

**Definição 1.4 (Seguro de Danos Próprios):-** O Seguro de Danos Próprios garante, a perda ou dano da viatura e/ou aos seus acessórios, quando nele se encontrarem, causado por um dos seguintes perigos: choque, capotamento, furto, incêndio, quebra de vidros (Karim [a]). Note-se que não pode existir um seguro de danos próprios sem que haja um seguro de responsabilidade civil.

Para o seguro de responsabilidade civil, existem capitais a segurar fixos. Dependendo do capital de seguro, o tipo e a cilindragem da viatura, é determinada uma taxa que pode ser agravada dependendo por exemplo da idade da viatura, a idade da carta de condução do condutor habitual da viatura (caso exista), e outros. Essa taxa é usada na obtenção do prémio da apólice.

Para o seguro de danos próprios, o capital a segurar é igual ao valor da viatura. A taxa a aplicar a esse valor, para calcular o prémio da apólice, é determinada usando a idade e tipo da viatura. A taxa pode sofrer agravamentos decididos pela chefia do ramo.

Existem dois tipos de prémio: prémio simples e prémio total. O **prémio puro** ou **simples** é o valor que apenas cobre as perdas. Mas como as operações de seguro envolvem custos e, é necessário ter uma margem para perdas pesadas e inesperadas para além de uma margem de lucro, ao prémio simples são acrescentadas percentagens para proporcionar as despesas, reservas e lucros. Este será o **prémio total** ou **comercial**. O anexo 4 mostra o algoritmo de cálculo do prémio simples e o anexo 5 o algoritmo de cálculo do prémio total.



*Figura 1.3: Diagrama de transição de estados da apólice (Macamo, 1996)*

O diagrama de transição de estados (Booch, 1994) da apólice de seguro, da figura 1.3, mostra o ciclo de vida de uma apólice desde a sua emissão até a sua anulação incluindo as alterações que ela possa sofrer.

Uma vez emitida, a apólice de seguro pode sofrer alterações nos seus dados. Como resultado das alterações é emitido um recibo suplementar ou de estorno e comunica-se ao segurado, através de uma acta adicional, as modificações do seu contrato de seguro.

**Definição 1.5 (Alteração):-** Uma Alteração é uma modificação aos dados da apólice de seguro. Ela desencadeia uma série de acções que, podem alterar o prémio da apólice, culminando com a emissão de um recibo suplementar ou de estorno.

Uma alteração pode ser solicitada pelo cliente ou exigida pelo ramo. As alterações solicitadas pelo cliente são aquelas em que o cliente pede alguma modificação nos dados da sua apólice. As alterações exigidas pelo ramo são aquelas que são impostas pela aplicação.

**Definição 1.6 (Recibo Suplementar):-** Um Recibo Suplementar é um documento contendo um valor adicional ao prémio da apólice, provocado por uma alteração, que deve ser pago pelo segurado.

Uma alteração de uma apólice de seguro pode aumentar o valor do prémio simples. A emissão de um recibo suplementar é feita tendo em conta o valor antigo e actual do prémio simples.

O recibo suplementar contém um valor comercial, que é calculado utilizando o prémio simples actual e o prémio simples antigo. A diferença dos valores anteriores é acrescida com valores comerciais (encargos, sobretaxa, selo, preço da acta adicional). O anexo 6 fornece o algoritmo de cálculo do valor do recibo suplementar.

**Definição 1.7 (Recibo de Estorno):-** Um Recibo de Estorno é um documento contendo um valor a ser pago ao cliente que é calculado quando existe uma alteração na apólice que reduz o valor do prémio simples.

Uma alteração de uma apólice de seguro pode diminuir o valor do prémio simples; a emissão de um recibo de estorno é feita tendo em conta o valor do prémio simples actual e o prémio simples antigo. O anexo 7 fornece o algoritmo de cálculo do valor do recibo de estorno.

**Definição 1.8 (Acta Adicional):-** Uma Acta Adicional é um documento que introduz alterações à apólice de seguro (Karim [b]); em certos casos, ela completa a apólice com elementos não indicados na proposta do contrato. Uma vez emitida, a acta adicional é integrada no contrato de seguro. Um exemplo de uma acta adicional é dado pelo anexo 3.

Uma apólice de seguro pode ter várias actas, cada uma das quais contém um conjunto de alterações. Por cada acta emitida pode existir, dependendo das alterações nela contidas, um recibo suplementar e/ou um recibo de estorno. Estas relações estão representadas na figura 1.4.

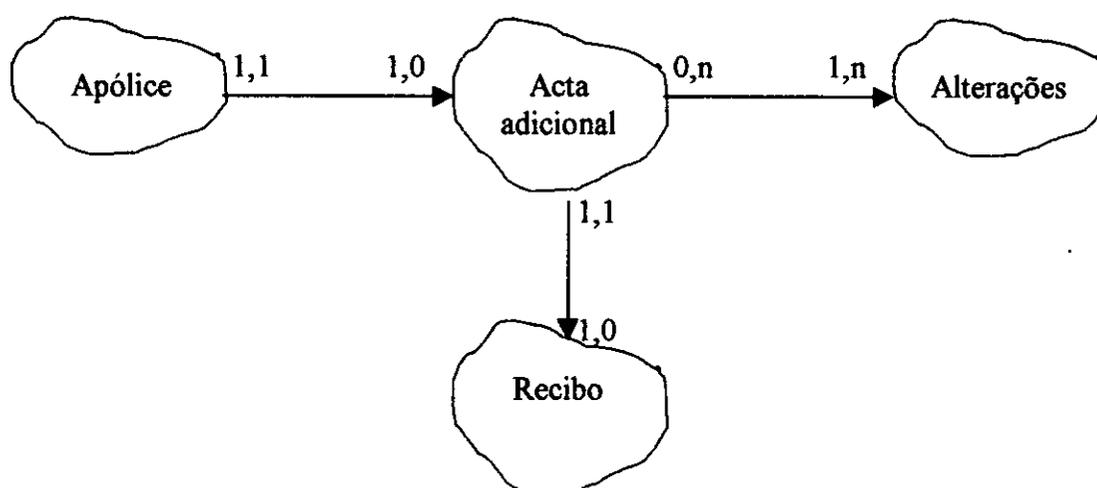


Figura 1.4: Relação entre as classes apólice, acta adicional, alterações e recibos

Cada alteração pode afectar o prémio simples da apólice resultante das acções nelas definidas. Quando existe um conjunto de alterações na mesma acta adicional, é importante observar a ordem de execução das alterações.

**Definição 1.9 (Ordem das Alterações):-** Ordem das Alterações é a sequência em que as alterações são executadas.

Para diferentes ordens de execução de alterações, pode-se obter valores diferentes do recibo emitido para cada acta adicional; portanto, o cálculo correcto do valor do recibo a emitir, depende de uma adequada ordem de execução das alterações.

No SIGES são registadas e executadas as alterações numa apólice de seguro mas, a ordem em que elas serão executadas depende da análise do técnico de seguro. O segurado pode solicitar um

conjunto de alterações na sua apólice de seguro, o técnico de seguros analisa as alterações pedidas e determina (manualmente) a ordem na qual elas devem ser introduzidas no SIGES. O sistema regista as alterações uma a uma e no fim, se o prémio da apólice foi modificado, pode emitir um recibo (de estorno e/ou suplementar), dependendo das alterações efectuadas.

Não existe definido, nem implementado, um método que possibilite a ordenação das alterações. Os técnicos de seguros determinam a ordem baseando-se no conhecimento que possuem sobre as alterações e nas suas experiências de trabalho.

Por definição, uma alteração tem acções a executar, essas acções podem estar relacionadas de alguma forma com as outras alterações; para o cálculo do prémio simples da apólice (veja anexo 4) pode-se estar a usar dados que tem haver com as outras alterações; por isso, se a ordem de execução das alterações, determinada pelo técnico de seguros não for a adequada, pode-se estar a executar uma alteração fazendo uso de dados que não estão correctos.

O problema consiste em encontrar a ordem adequada para a execução das alterações tal que, todas as alterações relacionadas com uma determinada alteração sejam previamente executadas. Note-se que não é o melhor prémio que se pretende mas sim o valor correcto do recibo a emitir.

A secção 1.1 deste capítulo contém definições de algumas alterações do ramo automóvel. Essas definições serão usadas para a análise de casos de estudo apresentados pela a secção 1.2. Esses casos exemplificam a importância da determinação da ordem de execução das alterações.

### **1.1. Alterações de apólices e suas definições**

No ramo automóvel, existe definida uma lista de alterações (veja o anexo 8) possíveis de serem executadas nas apólices de seguro.

De acordo com a definição 1.5, cada alteração tem a sua própria definição dada pelas acções que ela desencadeia e pelas relações que ela possui com outras alterações. A seguir são dadas as definições de algumas alterações do ramo automóvel.

A **alteração do capital de responsabilidade** significa o aumento ou diminuição do capital seguro de responsabilidade civil; esta alteração implica o recalculo do prémio simples da apólice e emissão de um recibo suplementar. Esta alteração não permite a emissão do recibo de estorno.

Ao fazer-se a **inclusão de danos próprios**, a apólice passa a cobrir os riscos de responsabilidade civil e danos próprios; ela implica a adição de cláusulas referentes a danos próprios e o recalculo do prémio simples da apólice. Esta alteração só permite que se emita um recibo suplementar.

Com a **exclusão de danos próprios**, a apólice passa a cobrir apenas os riscos de responsabilidade civil, deve-se retirar as cláusulas referentes aos danos próprios e recalculado do prémio simples da apólice. No final, pode-se emitir um recibo de estorno.

Na **alteração do capital de danos próprios**, aumenta-se ou reduz-se o capital a segurar para cobertura dos riscos de danos próprios. Depois de modificar o capital, deve-se recalculado o prémio simples da apólice. Esta alteração não prevê a emissão de um recibo de estorno isto é, só se emite um recibo suplementar.

A **alteração do titular da viatura** é a modificação do titular da apólice de seguro. Nesta alteração, para além de se alterar todos os dados referentes ao titular da apólice, altera-se os dados do condutor habitual da viatura, altera-se os descontos (existem descontos para instituições estatais, para segurados com seguros em outros ramos, e outros), altera-se os bónus da apólice e deve-se recalculado o prémio simples da apólice, emitindo apenas um recibo suplementar.

Na **alteração do condutor habitual da viatura** altera-se todos os dados referentes ao indivíduo que normalmente guia a viatura. Deve-se depois da alteração recalculado o prémio simples da apólice e

emitir um recibo. Esta alteração não permite que se emita um recibo de estorno, emitindo apenas um recibo suplementar.

Na **alteração da matrícula da viatura** apenas se modifica a matrícula da viatura assegurada, não sendo necessário o recálculo do prémio da apólice; assim sendo, não se emite nenhum tipo de recibo.

Ao se **substituir a viatura** assegurada, deve-se fazer uma vistoria a nova viatura, modificar-se todos os dados referentes às características da viatura segurada e recalcula-se o prémio simples da apólice. Esta alteração permite que se emita apenas um recibo suplementar.

A **alteração do uso da viatura** significa que uma viatura particular passa a ser usada para o serviço de aluguer ou vice-versa. Nesta alteração modifica-se a classificação e a utilidade da viatura; deve-se recalcular o prémio simples e emitir apenas um recibo suplementar.

Com a **alteração da data de vencimento** está-se a modificar o período de duração da apólice. Após alterar a data de término da apólice faz-se o recálculo das datas de pagamento das prestações, o recálculo do prémio total da apólice e emite-se um recibo suplementar ou de estorno.

Na **inclusão dos passageiros gratuitos** são incluídos no seguro, os passageiros transportados gratuitamente. Nesta alteração actualiza-se as cláusulas da apólice, recalcula-se o prémio simples e só se pode emitir um recibo suplementar.

Ao se **excluir da apólice os passageiros gratuitos**, deve-se retirar as cláusulas a eles referenciadas, recalcular o prémio simples da apólice. Esta alteração não prevê a emissão de um recibo de estorno.

A **alteração do número de prestações de pagamento da apólice** modifica as datas de pagamento das prestações, então deve-se recalcular as datas de pagamento das prestações para a anuidade seguinte. Esta alteração não emite nenhum tipo de recibo.

Faz-se a **reposição do capital** quando existe um sinistro cuja indemnização é igual ou superior ao capital seguro. Após a reposição do capital emite-se um recibo de reposição do capital.

A **anulação da apólice** é uma alteração que tanto pode ser solicitada pelo segurado ou imposta pelo ramo. Ela modifica o estado da apólice e não consta na acta adicional. Esta alteração prevê apenas a emissão de um recibo de estorno.

No levantamento da anulação da apólice faz-se a renovação da mesma; altera-se os seus bónus, recalcula-se o prémio simples e emite-se apenas um recibo suplementar. Esta alteração não tem acta adicional.

## **1.2. Casos de estudo**

Para melhor entendimento da importância do estabelecimento da ordem de execução das alterações, são aqui exemplificadas algumas situações onde a ordem das alterações é importante.

Os exemplos aqui apresentados são grupos de duas alterações para a mesma acta adicional. Isto não significa que só se pode solicitar duas alterações. Para a mesma acta, pode-se ter um número indeterminado de alterações.

O primeiro caso considera os problemas que podem existir ao solicitar-se a substituição da viatura segura; o segundo caso tem haver com o aumento do capital de responsabilidade civil da apólice e a redução da data de vencimento da apólice; o terceiro caso, engloba a alteração do capital de responsabilidade civil e a alteração do uso da viatura de aluguer para particular.

Os casos acima mencionados são considerados com base na apólice do anexo 2. A apólice refere-se a viatura que se destina ao serviço de aluguer, não cobrindo os riscos de danos próprios; ela possui uma única prestação, que inicia a 1 de Fevereiro de 1997 e termina a 1 de Fevereiro de 1998. Considera-se ainda que as alterações são efectuadas no dia 10 de Setembro de 1997.

Para o primeiro caso, considere-se que o segurado vendeu a sua viatura e adquiriu uma outra; ele solicita a substituição, na apólice, da antiga viatura pela nova.

Pode acontecer que a nova viatura já esteja segura na mesma seguradora; assim, o SIGES não permite a substituição da viatura.

Quando isto acontece, deve-se acrescentar, ao grupo das alterações solicitadas, a anulação da outra apólice. Sem essa anulação a substituição da viatura não é possível, uma vez que o SIGES não permite duas apólices para a mesma viatura.

Então, neste caso, primeiro é necessário anular a apólice da nova viatura e só depois executar a alteração solicitada.

Para o segundo caso, considere-se duas opções: A primeira opção, executa a alteração do capital de responsabilidade civil e depois a alteração da data de vencimento da apólice. A segunda opção, executa primeiro a alteração da data da apólice e depois a alteração do capital de responsabilidade civil.

Tomando em conta a primeira opção, faz-se primeiro o aumento do capital de responsabilidade civil e como resultado desta alteração, é emitido um recibo suplementar. Este recibo é emitido considerando a data de fim da apólice o dia 1 de Fevereiro de 1998 (Figura 1.5).

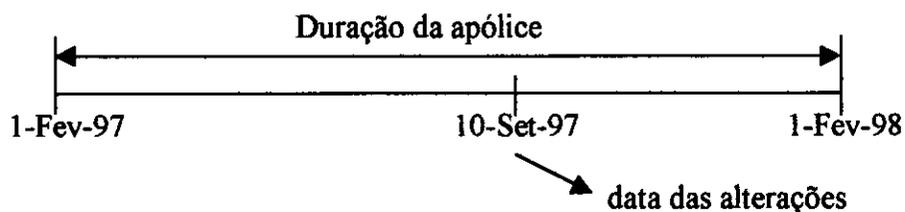


Figura 1.5: período de duração da apólice

Depois, reduz-se a data de vencimento da apólice, de 1 de Fevereiro de 1998 para 1 de outubro de 1997. Esta alteração, deveria emitir um recibo de estorno, mas, tal não acontece pois a data em que a alteração é executada ultrapassa o tempo regulamentado (6 meses) pela Emose para a atribuição de estorno (ver o anexo 7).

Assim, a 1 de Outubro de 1997, data de vencimento da apólice, a apólice deve ser renovada e um recibo para o efeito é emitido. Neste caso, existe um período, de 1 de Outubro de 1997 a 1 de Fevereiro de 1998, em que o segurado paga duas vezes pois, ele já tinha pago ao fazer-se a alteração do aumento do capital de responsabilidade civil (figura 1.6).

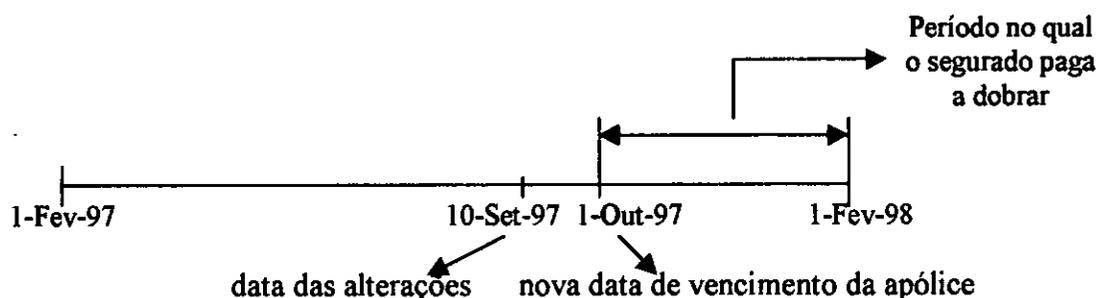


Figura 1.6: período em que o segurado paga a dobrar

Agora, observe-se a segunda opção que primeiro reduz a data de vencimento da apólice de 1 de Fevereiro de 1998 para 1 de Outubro de 1997; analogamente, nenhum recibo é emitido.

Depois, ao aumentar-se o capital de responsabilidade civil, o valor do recibo suplementar emitido é calculado apenas até a data de vencimento da apólice, como mostra a figura 1.7.

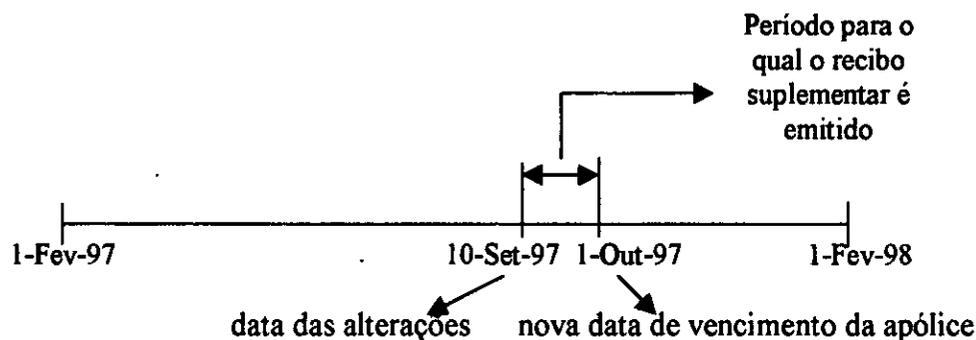


Figura 1.7: período considerado na emissão do recibo

Comparando as opções apresentadas para este caso, pode-se ver que na primeira opção o segurado faz o pagamento da apólice duas vezes para um determinado período; e isto já não acontece usando a ordem de execução das alterações dada pela segunda opção.

Para o terceiro caso, também se pode considerar duas opções: a primeira opção, executa primeiro a alteração do capital de responsabilidade civil e depois a alteração do uso da viatura.

Para uma melhor explicação do terceiro caso, serão utilizadas variáveis com a seguinte notação:

- Valor do capital de responsabilidade civil da apólice –  $CRC_i$
- Valor do prémio simples da apólice –  $PS_i$
- Valor do prémio total da apólice –  $PF_i$
- Valor do recibo suplementar emitido –  $RS_i$

onde

$i$  – representa o número de vezes em que a variável é calculada; ele pode tomar os valores 0,1,2, ..., sendo zero o valor que se encontra na apólice do anexo 2.

Assim, supondo que a unidade monetária seja o Metical, tem-se :

$$CRC_0 = 500.000.000,00$$

$$PS_0 = 420.000,00$$

$$PF_0 = 521.720,00$$

Analisando a primeira opção, faz-se o aumento do capital de responsabilidade civil, obtendo  $CRC_i$ ; como consequência desta alteração obtém-se um novo valor do prémio simples e um novo valor do prémio total; além disso, é emitido um recibo suplementar. Assim passa a ter:

$$CRC_1 = 1.000.000.000,00$$

$$PS_1 = 456.000,00$$

$$PF_1 = 565.568,00$$

$$RS_1 = 21.043,15$$

Depois desta alteração, executa-se a alteração do uso da viatura de aluguer para particular. De acordo com a definição desta alteração (dada na secção 1.1), o prémio simples da apólice é recalculado obtendo-se:

$$PS_2 = 420.000,00$$

$$PF_2 = 521.720,00$$

Observe-se que  $PS_2$  é menor do que  $PS_1$ , então, devia-se emitir um recibo de estorno mas, por definição, a alteração do uso da viatura não emite recibo de estorno. Terminando por aqui a execução das alterações solicitadas.

Analisando a segunda opção, faz-se primeiro a alteração do uso da viatura de aluguer para particular, assim, obtém-se um novo prémio simples e prémio total com os seguintes valores:

$$PS_3 = 336.000,00$$

$$PF_3 = 419.000,00$$

De notar que  $PS_3$  e  $PF_3$  são calculados considerando os valores da apólice do anexo 2

$$CRC_0 = 500.000.000,00$$

$$PS_0 = 420.000,00$$

$$PF_0 = 521.720,00$$

Nesta opção, o prémio simples também ficou reduzido, isto é,  $PS_3$  é menor do que  $PS_0$ , e analogamente não é emitido o recibo de estorno.

Agora, aumentando o capital de responsabilidade civil, tem-se um capital,  $CRC_2$ , e como consequência obtém-se um novo prémio simples  $PS_4$  e prémio total  $PF_4$ . Para esta alteração, é emitido um recibo suplementar com valor  $RS_2$ .

$$CRC_2 = 1.000.000.000,00$$

$$PS_4 = 420.000,00$$

$$PF_4 = 521.720,00$$

$$RS_2 = 49.073,84$$

Comparando o resultado final de cada sequência, pode-se observar que os prémios simples e finais são iguais, isto é

$$PS_2 = PS_4 \quad e \quad PF_2 = PF_4$$

mas, o valor do recibo suplementar, que o segurado deve pagar é diferente

$$PS_1 \neq PS_2$$

Neste caso, a ordem de execução correcta das alterações é a sequência dada pela segunda opção, uma vez que o valor que no final o cliente tem que pagar é calculado tendo em conta que a viatura se destina ao uso particular (que é o que se pretende); isto já não acontece na primeira opção onde o valor que o cliente tem que pagar é calculado tendo em conta que a viatura se destina ao serviço de aluguer.

Para a apólice aqui exemplificada, o valor correcto, que o cliente tem que pagar, é superior ao valor da ordem não correcta; para uma apólice com dados diferentes dos aqui exemplificados, a situação pode inverter ou seja, o valor correcto pode ser inferior ao valor resultante da ordem de execução não correcta.

Durante a análise dos casos, não se usou nenhum método para a determinação da ordem de execução correcta das alterações, isto porque não existe definido, nem implementado, um método para o efeito, para tal, usou-se o conhecimento existente sobre as alterações; conhecimento esse, relacionado com a definição de cada alteração e a relação existente entre a alteração e o resto dos dados da apólice.

Os casos aqui exemplificados, mostram como a ordem de execução das alterações é importante, uma vez que diferentes ordens das alterações levam a resultados diferentes.

Uma forma de solucionar o problema da determinação da ordem correcta das alterações é usando as técnicas de Inteligência Artificial. O capítulo que se segue irá debruçar-se sobre o assunto.

## **2. Inteligência Artificial**

Com o aparecimento dos computadores, os Homens viram algumas das suas actividades aliviadas; inicialmente os computadores executavam tarefas simples, mais tarde, o avanço da tecnologia e o aparecimento de linguagens de programação poderosas, permitiu a criação de programas para execução de actividades complexas. Esses programas executam uma determinada actividade seguindo passos bem definidos e estruturados ou seja, seguindo um algoritmo. Mas, mesmo assim, nem todas as necessidades do Homem estão satisfeitas; ele verificou que certas actividades não poderiam ser automatizadas usando algoritmos, por exemplo:

Um médico para fazer um diagnóstico baseia-se nos sintomas do doente; mas existem doenças com os mesmos sintomas ou com sintomas semelhantes então, ele chega a uma conclusão tendo em conta factos passados dessas doenças. A escolha de uma e não outra doença depende da experiência que o médico possui sobre esse tipo de doença. É exactamente esse conhecimento que leva o médico a escolher uma e não outra doença que, usando os programas estruturados não é possível automatizar.

De acordo com o problema exposto no capítulo anterior, podem existir dependências entre as alterações, e a ordem em que elas são feitas deve ser considerada pois, não se pode fazer primeiro uma determinada alteração A, dependente de uma B, sem que se faça a alteração B. Existem muitos dados que podem ser alterados e a solicitação de uma alteração ou de um conjunto de alterações é aleatório isto é, não se pode prever que um determinado cliente solicitará uma determinada alteração ou grupo de alterações. Os técnicos de seguro estabelecem a ordem das alterações baseando-se no conhecimento que possuem sobre elas e nas suas experiências de trabalho.

Para a solução deste tipo de problemas, o uso de conhecimento e das experiências dos peritos (médicos, técnicos de seguros, e outros) é indispensável.

A forma de representação do conhecimento e da experiência dos peritos em um determinado domínio, constitui um dos aspectos fundamentais para a resolução dos problemas. Outro aspecto é a definição de mecanismos capazes de manipular o conhecimento e a experiência representada com o objectivo de obter uma conclusão sobre a solução do problema.

Para responder a essas necessidades, surgiu uma nova área na ciência de computação, a Inteligência Artificial (Russel e Norving, 1995).

**Definição 2.1 (Inteligência Artificial):-** A Inteligência Artificial, IA, estuda as faculdades mentais através do uso de modelos computacionais (Charniak e McDermott, citado por Russel e Norving, 1995). A IA estuda a maneira como as pessoas resolvem um determinado problema não estruturado, analisando os passos que seguem, procurando compreender a lógica do seu pensamento.

Entende-se por lógica do pensamento humano o modo como os Homens raciocinam, ou seja, uma operação mental pela qual, de dois ou mais juízos (argumentos), tira-se outro. A IA para representar a lógica do pensamento faz uso da Lógica Proposicional (Johnsonbaugh, 1990) e do Cálculo de Predicados (Johnsonbaugh, 1990) ou da Lógica Fuzzy (Russel e Norving, 1995).

A IA é aplicada no processamento da linguagem natural, visão, sistemas periciais, redes neurais e outros.

O processamento da linguagem natural é a criação de programas computacionais capazes de compreender a linguagem natural. Com o processamento da linguagem natural pode-se por exemplo criar programas que questionam uma base de dados, programas capazes de extrair informação de textos, retirar documentos relevantes de uma colecção, traduzir de uma língua para outra. Exemplos de programas deste tipo são o SPANAM (Russel e Norving, 1995), que permite traduzir trechos do espanhol para inglês, o CHAT (Russel e Norving, 1995), que permite questionar uma base de dados geográfica.

A visão processada por computadores permite que o computador identifique e compreenda o que vê. O objectivo da visão neste caso é extrair informação necessária para execução de tarefas como a manipulação, navegação e reconhecimento de objectos.

Os sistemas baseados no conhecimento (sistemas periciais) são sistemas que permitem resolver um determinado problema. Eles fazem com que o computador actue como um especialista numa determinada área. Um exemplo deste tipo de sistemas é o INVEST (Trippi e Lee, 1996), que é um sistema que ajuda na tomada de decisões sobre investimentos bancários.

As redes neurais são sistemas que tomam uma decisão baseando em experiências acumuladas na resolução de problemas. Exemplo de redes neurais são a NETtalk (Russel e Norving, 1995), que aprende a pronúncia de um texto escrito em caracteres ASCII, o ALVINN (Russel e Norving, 1995), que aprende a conduzir um veículo observando a actuação de um condutor humano.

## **2.1. Sistemas baseados no conhecimento**

A experiência e o conhecimento de um perito sobre uma determinada área, são incorporados em um programa computacional capaz de actuar como um perito na resolução de problemas. A esse conjunto denomina-se sistema baseado no conhecimento.

**Definição 2.2 (Conhecimento):-** O Conhecimento é a compreensão ou noção que se tem sobre algo (Stefik, 1995). O conhecimento de um perito é a informação que ele possui sobre alguma coisa.

**Definição 2.3 (Experiência):-** A Experiência é o conhecimento ou a informação acumulada através de situações práticas da vida. Por vezes, ao se falar de conhecimento inclui-se também a experiência.

A obtenção do conhecimento é uma tarefa que envolve um grande esforço e custos elevados; isto porque pode-se levar muito tempo até conseguir extrair a informação do(s) perito(s) e, muitas vezes, os indivíduos que possuem o conhecimento não estão dispostos a revelá-lo ou quando o fazem, não o revelam completamente.

**Definição 2.4 (sistema baseado no conhecimento):-** Um sistema baseado no conhecimento, "*knowledge-based system*", é um programa computacional cuja execução é guiada por conhecimento pericial específico na resolução de problemas (Stefik, 1995).

Um sistema baseado no conhecimento pericial usa o conhecimento e procedimentos envolvendo inferência para resolver um determinado problema nas áreas de diagnóstico, prognóstico, desenho, tomada de decisões, e outras ; ele deve ser capaz de explicar as decisões e os raciocínios usados para chegar a uma determinada conclusão.

A arquitectura geral de um sistema baseado no conhecimento é formada por uma base de conhecimentos, um motor de inferência e interfaces como mostra a figura 2.1.

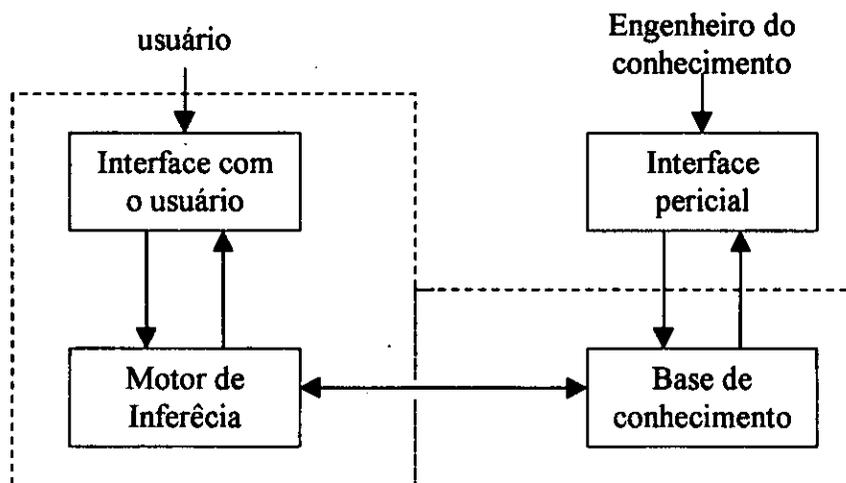


Figura 2.1: componentes de um sistema pericial

A **base de conhecimento** é o conjunto do conhecimento disponível para o sistema pericial isto é, o repositório do conhecimento usado pelo sistema. A base de conhecimentos é actualizada periodicamente para reflectir mudanças ou extensões do domínio do conhecimento.

O **interface com o usuário** é a parte do sistema que está ligada aos utilizadores. Este interface permite a comunicação entre o utilizador e o sistema.

O **interface pericial** é a parte do sistema ligada à introdução do conhecimento do perito. Este interface só está disponível ao engenheiro do conhecimento. Ele permite a actualização e teste da base de conhecimento; o interface fornece também ferramentas que permitem examinar o conteúdo da base de conhecimento.

O **motor de inferência** ou estrutura de controle permite a manipulação do conhecimento existente na base de conhecimento para satisfazer um determinado objectivo.

Geralmente, o motor de inferência e o interface com usuário são vistos com um único módulo chamado "*shell*". O "*shell*" é desenvolvido para ser independente da base de conhecimentos assim, é possível usar o mesmo "*shell*" com diferentes bases de conhecimentos. Isto permite que problemas com domínios diferentes usem o mesmo "*shell*" bastando que cada um tenha a sua base de conhecimentos. Actualmente existem inúmeros sistemas comercializáveis deste tipo como por exemplo o Babylon (Kantrowitz, 1997), que é um ambiente que permite o desenvolvimento de sistemas baseados no conhecimento para o diagnóstico; o G2 (Kantrowitz, 1997), que é um ambiente gráfico e orientado a objectos para a criação de aplicações inteligentes para a monitoração, diagnóstico e controle de eventos dinâmicos em ambientes reais ou simulados.

O conteúdo da base de conhecimentos, deve estar representado numa forma apropriada. Existem várias formas de o representar, por meio de regras, frames, redes semânticas, e outras (Rich e Knight, 1991).

Para a manipular o conhecimento representado numa determinada forma, é necessário uma estratégia ou método de pesquisa. Existem diversos métodos de pesquisa, como por exemplo, a pesquisa em profundidade, pesquisa por níveis, pesquisa pela melhor escolha, satisfação de restrições, análise de meios e fins, e outros (Rich e Knight, 1991).

Geralmente, um método de pesquisa procura, no espaço do problema, um caminho (possível solução) que vai de um estado inicial até um estado objectivo.

**Definição 2.5 (Estado):-** Um estado do problema é uma configuração dos objectos que fazem parte do problema. Um estado representa um ponto no espaço do problema.

**Definição 2.6 (Espaço do Problema):-** Espaço do Problema é o conjunto de todos os estados possíveis do problema, ou seja, o espaço do problema é o conjunto de todas as configurações possíveis dos objectos que fazem parte do problema.

Dentro do espaço do problema destacam-se os estados iniciais, que são as situações possíveis por onde o processo de resolução pode começar, e estados objectivos que são as situações aceites como solução do problema.

Na procura do caminho, que representa uma possível solução do problema, pode-se pensar para frente "*forward reasoning*" ou para trás "*backward reasoning*" (Rich e Knight, 1991).

Ao se fazer a pesquisa pensando para frente, parte-se de um estado inicial e procura-se um estado objectivo. Por exemplo, é solicitado um conjunto de alterações a uma apólice; a ordem em que elas são solicitadas representa o estado inicial; deve-se procurar uma sequência que forneça a ordem correcta das alterações, essa sequência representa o estado objectivo. Neste exemplo, inicio-se a pesquisa por um estado inicial e terminou-se quando se encontrou um estado objectivo.

Em geral, a pesquisa para frente é feita quando se tem as causas de um algo e pretende-se determinar a consequência ou o resultado dessas causas.

Pensando para trás, a pesquisa é feita partindo de um estado objectivo e procura-se um estado inicial, ou seja, tem-se um resultado e pretende-se saber quais as causas que o originaram.

### 2.1.1. Representação do conhecimento

O conhecimento e a experiência extraídos de um perito têm que estar de uma forma tal que seja compreendido e que seja capaz de ser incorporado em um programa computacional.

A forma como o conhecimento pode ser representado leva a distinção de quatro tipos de conhecimento: conhecimento relacional, herdável, inferencial e procedimental (Rich e Knight, 1991).

O conhecimento relacional corresponde a um conjunto de atributos e valores associados que juntos descrevem os objectos da base de conhecimento. Este é o modo como a informação se encontra num sistema de base de dados. Esta é a maneira mais simples de representar factos declarativos, ela não possui grande capacidade de inferência mas, pode servir de entrada de dados a motores de inferência poderosos.

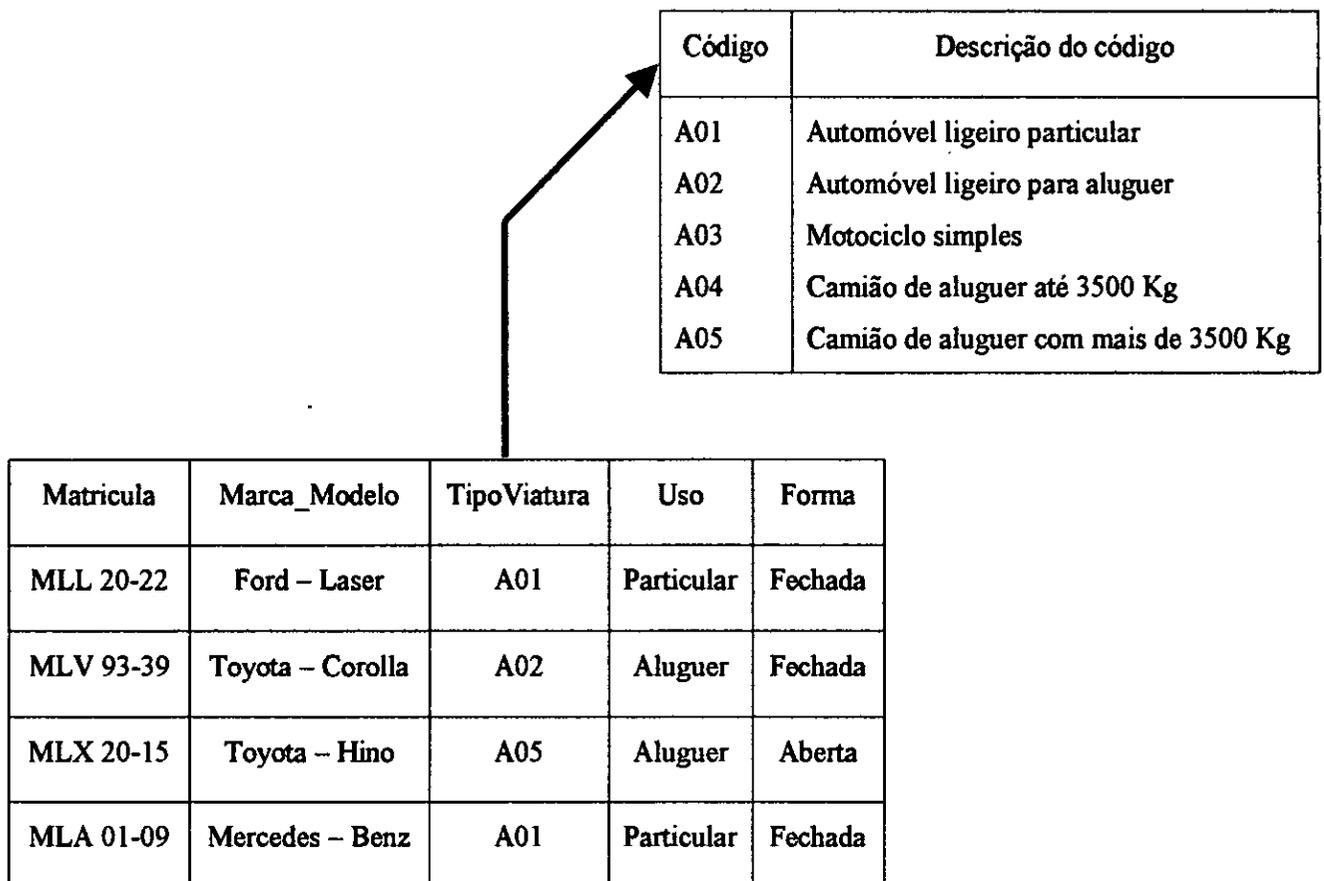


Figura 2.2: exemplo de conhecimento relacional

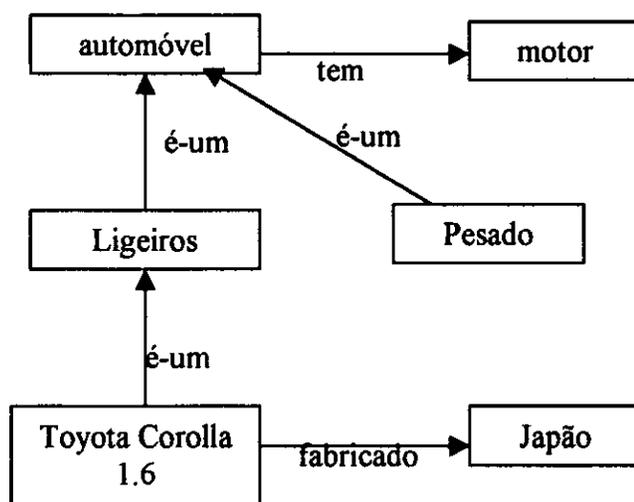
Um exemplo de conhecimento relacional é dado pela figura 2.2; ela fornece a informação sobre viaturas. No conhecimento relacional pode-se agrupar a informação em tabelas e relacioná-las através de um código.

O conhecimento relacional, pode ser acrescido com mecanismos de inferência que operam nas estruturas da representação, para tal, a estrutura deve ser desenhada para corresponder aos mecanismos de inferência desejados.

Uma das formas de inferência é a herança de atributos, onde elementos de classes específicas herdam atributos e valores de classes mais gerais onde elas estão incluídas. Para tal, os objectos devem ser organizados em classes e estas, por sua vez, devem estar organizadas numa hierarquia de generalização. O conhecimento com estas características é chamado de conhecimento herdável. O conhecimento herdável pode ser encontrado nas Redes Semânticas ou nos Frames (Rich e Knight, 1991).

Uma **rede semântica** descreve as propriedades dos objectos e as relações entre objectos, eventos, conceitos, situações ou acções.

Em uma rede semântica a informação é representada por um conjunto de nós interligados por arcos etiquetados. Um arco representa a relação entre os nós que ele conecta.



*Figura 2.3: exemplo de uma rede semântica*

Suponha-se que se pretende representar a informação sobre um determinado automóvel de marca Toyota. A figura 2.3 exemplifica uma forma de representar essa informação usando uma rede semântica. Como um ligeiro é um automóvel e o automóvel tem um motor, então o ligeiro também

possui um motor; isto acontece porque o ligeiro herda todos os atributos do automóvel. Como o Toyota Corolla 1.6 é um ligeiro, ele herda todos os atributos de ligeiro e também do automóvel.

Os frames também possuem a propriedade de herança, desde que eles representem uma classe ou uma instância de uma classe, criando uma hierarquia de classes. Se um frame representa uma subclasse de um outro frame, então ele pode herdar valores desse frame.

Um frame é uma colecção de atributos, chamados de *slots*, e valores associados que descrevem uma entidade em um domínio específico.

Usando o exemplo sobre automóveis dado nas redes semânticas, pode-se definir os seguintes frames:

**Automóvel**

*tem:* motor

*cilindragem:* > 50 cm<sup>3</sup>

**Pesado**

*é-um:* automóvel

*lugares:* > 9

*peso:* > 3500

**Ligeiros**

*é-um:* automóvel

*max-lugares:* 9

*peso-max:* 3500

**Toyota Corolla 1.6**

*instância:* ligeiro

*fabricado:* Japão

*instância:* Toyota

*combustível:* gasolina

*num\_lugares:* 5

*cor:* azul

*caixa:* fechada

Nos frames, a herança de atributos procede-se do mesmo modo que nas redes semânticas. Usando o exemplo acima, um automóvel da marca Toyota Corolla 1.6, possui uma cilindragem superior a 50 cm<sup>3</sup>, isto porque ele é um ligeiro que, por sua vez é um automóvel e os automóveis têm cilindragem superior a 50 cm<sup>3</sup>.

A propriedade de herança não é a única forma de se obter inferência, pode-se usar a lógica para descrever inferências, desde que exista um procedimento de inferência capaz de explorar o conhecimento existente. Isto é o que se encontra no conhecimento inferencial.

O conhecimento inferencial pode ser encontrado nos contextos. Um contexto permite que se exclua da análise certas informações. Por exemplo, considerando a classificação das viaturas dada pela a figura 2.2, pode-se criar o seguinte contexto:

Se o automóvel é ligeiro, com certeza exclui-se os tipos A03, A04 e A05

No conhecimento procedimental existe um procedimento que especifica o que fazer e quando se deve fazer. Um exemplo comum deste tipo de conhecimento é a representação usando regras de produção (Rich e Knight, 1991).

Uma regra é uma frase condicional que permite definir relações lógicas entre os objectos do domínio do problema. Por exemplo:

Se é alterado o capital de responsabilidade civil então  
deve-se actualizar o prémio simples da apólice

Se é alterado o capital de danos próprios então  
deve-se actualizar o prémio simples da apólice

Se é aumentado o capital de responsabilidade civil e é aumentado o capital de danos próprios então deve-se emitir recibo suplementar

Se o automóvel é ligeiro e é usado para aluguer então o tipo de viatura é A02

Uma regra contém procedimentos que indicam quando e como ela deve ser aplicada. As regras podem ter várias interpretações, por exemplo:

Se condição P então conclusão C

Se situação S então acção A

Se condição  $C_1$  e  $C_2$  ocorrem então C não ocorre

Em certos casos, como em diagnósticos médicos, é necessário ter um conhecimento subjectivo. Neste caso, as regras são modificadas pela adição de um grau de certeza, por exemplo:

Se condição A então conclusão B segue com certeza F

Para que se tenha uma boa representação do conhecimento, não basta a identificação do tipo de conhecimento a usar, é também necessário observar as seguintes propriedades (Rich e Knight, 1991): adequação representacional, adequação inferencial, eficácia inferencial e eficácia aquisicional.

A adequação representacional é a habilidade de representar todo o tipo de conhecimento necessário ao domínio do problema, isto significa que qualquer informação relevante sobre o domínio do problema deve ser possível de ser representada.

A adequação inferencial é a habilidade de manipular as estruturas representacionais de modo a derivar novas estruturas, correspondentes ao novo conhecimento inferido através do antigo ou seja, o conhecimento deve estar representado de tal modo que seja possível obter novas informações a partir da informação existente.

Eficácia inferencial é a habilidade de incorporar, nas estruturas do conhecimento, informações adicionais que possam ser usadas para focar a atenção do mecanismo de inferência na direcção mais promissora.

Eficácia aquisicional é a habilidade de adquirir novas informações facilmente. Uma pessoa pode inserir directamente na base de conhecimentos uma nova informação; mas é desejável que o programa seja capaz de controlar a aquisição do novo conhecimento sem grande interferência humana.

### 2.1.1.1. O uso da lógica na solução de problemas

A lógica foi evoluindo a medida que a necessidade de representar o conhecimento tornou-se maior. Inicialmente usou-se a Lógica de Proposições (Johnsonbaugh, 1990) para representar afirmações verdadeiras ou falsas. Como a lógica de proposições não considera a dependência dos dados, passou-se a usar o Cálculo de Predicados (Johnsonbaugh, 1990), mas com a necessidade de representar factos qualitativos, surge a Lógica Fuzzy (Russel e Norving, 1995).

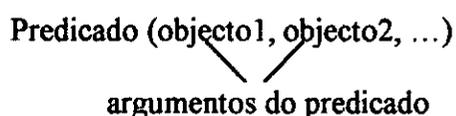
Uma proposição é uma afirmação sobre objectos, do domínio do problema, que pode ser verdadeira ou falsa. Por exemplo

“A viatura possui um seguro”.

“A alteração do titular da viatura afecta o valor do prémio da apólice”.

A lógica de proposições lida apenas com a veracidade ou falsidade da afirmação e não considera a dependência dos dados. A lógica de predicados remedeia esta situação permitindo a manipulação de afirmações sobre objectos e permitindo o uso de variáveis e funções de variáveis.

O predicado e os seus argumentos formam uma proposição (figura 2.4). Por isso, todas as operações da lógica de proposições podem ser aplicadas aos predicados.



*Figura 2.4: o predicado e os seus argumentos*

Considerando a proposição

“A viatura possui um seguro”

pode-se formular o seguinte predicado:

possui (viatura, seguro)

Considere-se agora a seguinte afirmação

“A viatura é velha”

Não está claro se a afirmação é ou não verdadeira. Para se dizer algo acerca da sua veracidade é necessário saber quantos anos a viatura tem e comparar com os padrões normais de velhice das viaturas.

Se a veracidade da afirmação não está clara, não se pode representar a afirmação usando a lógica de proposições, pois está indicada claramente a veracidade da afirmação. Como um predicado forma uma proposição, também não se pode representar usando o cálculo de predicados.

O mesmo acontece com as afirmações

“o João é alto”

“hoje está muito quente”

“o Pedro é velho”.

Pode-se representar a altura de uma pessoa ou a sua idade mas como se representa o facto que alguém é alto ou velho? O mesmo acontece com a temperatura, pode-se registar a temperatura do dia mas a mesma questão é colocada: como representar o facto de que “hoje está muito quente”?

A **lógica fuzzy** aparece para responder a essas necessidades, permitindo que valores intermediários sejam definidos entre os convencionais Sim/Não ou Verdadeiro/Falso (Bauer, Nouak e Winkler, 1996). Isto é, valores entre o “completamente verdadeiro” e o “completamente falso” (Kantrowitz, Horstkotte e Josly, 1993). Ela foi iniciada nos anos 60 por Lotfi A. Zadeh, professor de ciência de computação na Universidade de Califórnia em Berkeley.

**Definição 2.5 (subconjunto fuzzy):**- Dado um conjunto universo  $S$ , pode-se definir um subconjunto fuzzy  $F$  de  $S$  como um conjunto de pares ordenados, onde o primeiro elemento do par é um elemento de  $S$  e o segundo é um valor no intervalo  $[0,1]$ , com apenas um par ordenado para cada elemento de  $S$  (Horstkotte, 1996). Isto é,

$$F = \{ (x,y) : x \in S \wedge y \in [0,1] \subset \mathfrak{R} \wedge (\forall z \in [0,1], (x,y) \in F \Rightarrow (x,z) \notin F) \}$$

O grau de veracidade da frase “ $x$  está em  $F$ ” é determinado pela procura do par ordenado cujo primeiro elemento seja  $x$ . O grau de veracidade da frase é dado pelo segundo elemento do par; o valor um representa o “completamente verdadeiro” e o valor zero o “completamente falso”.

Considere o seguinte exemplo: seja  $S$  o conjunto universo de viaturas. Pretende-se saber o grau de velhice de uma determinada viatura. Pode-se definir o subconjunto fuzzy ‘velho’ da seguinte maneira:

$$\text{Velho}(x) = \begin{cases} 0, & \text{se idade}(x) < 10 \text{ anos} \\ (\text{idade}(x) - 10 \text{ anos})/5 \text{ anos}, & \text{se } 10 \leq \text{idade}(x) \leq 15 \text{ anos} \\ 1, & \text{se idade}(x) > 15 \text{ anos} \end{cases}$$

Por exemplo, o grau de velhice de uma viatura com 12 anos é 0.4.

Por vezes confunde-se os valores fuzzy com os valores probabilísticos, isto não é correcto pois os valores fuzzy não possuem a principal propriedade dos valores probabilísticos, a aditividade, que diz que a soma dos valores tem que ser um. Por outro lado, todas as distribuições probabilísticas são subconjuntos fuzzy. Isto significa que a probabilidade é um caso particular dos conjuntos fuzzy.

## 2.2. Redes neurais

Segundo a definição 2.1, a Inteligência Artificial baseia-se no funcionamento do cérebro humano. A maioria das técnicas de IA, como os sistemas baseados no conhecimento, restringem-se ao processamento sequencial e apenas permitem algumas representações do conhecimento e da lógica. Essas técnicas usam princípios psicológicos do sistema nervoso.

O cérebro é composto por células, denominadas neurónios, que processam a informação. Os neurónios encontram-se agrupados em redes, cada uma das quais contendo milhares de neurónios conectados entre si, formando uma arquitectura paralela e distribuída.

A arquitectura paralela permite que o processamento de informação seja rápido, enquanto que a representação distribuída permite generalizações, que é a habilidade de lidar com situações similares, mas não idênticas, a experiências passadas.

Como forma de imitar as características do processamento do cérebro, surgiram as redes neurais que baseiam-se em alguns princípios fisiológicos do sistema nervoso.

**Definição 2.6 (Redes Neurais):**- Uma Rede Neural Artificial (*"Artificial Neural Network"*), é uma tecnologia de processamento de informação inspirada nos estudos do cérebro e do sistema nervoso humano (Trippi e Turban, 1996).

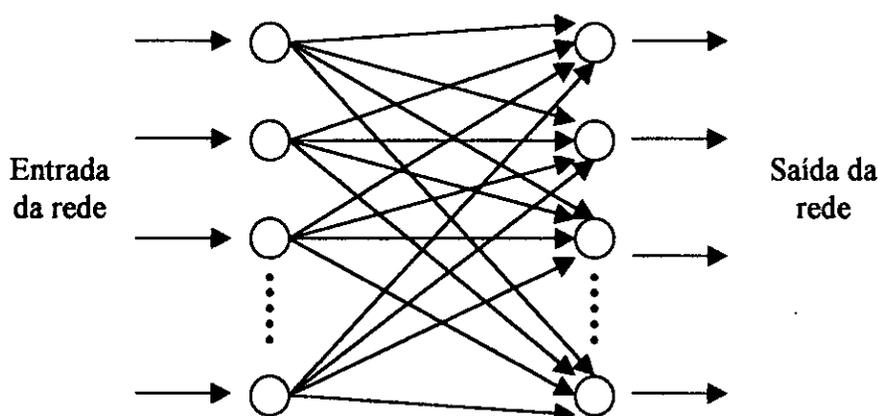


Figura 2.5: Exemplo de uma rede neural

O primeiro princípio usado é que o cérebro possui uma arquitectura paralela e distribuída feita de neurónios interconectados, cada um dos quais possuindo uma capacidade computacional limitada.

O segundo princípio utilizado é que a aprendizagem do neurónio ocorre alterando a relação que existe entre os neurónios. Essa relação pode ser expressada por um valor numérico atribuído à ligação entre dois neurónios.

Portanto, as redes neurais artificiais, são compostas por neurónios ligados entre si formando uma arquitectura paralela e distribuída, como mostra a figura 2.5. Uma rede neural aceita várias entradas, executa sobre elas diversas operações e produz uma ou mais saídas.

A secção seguinte fornece alguns conceitos básicos sobre as redes neurais. Esses conceitos são usados na explicação dos passos a seguir no desenvolvimento de um sistema de redes neurais.

### 2.2.1. Conceitos gerais

Uma rede neural é composta por neurónios ligados entre si; cada ligação tem um peso ou valor numérico a ela associada.

**Definição 2.7 (Neurónio):-** Um Neurónio é um elemento de processamento que compõe a rede. Cada neurónio recebe uma ou mais entradas, processa-as e devolve uma única saída, que pode ser uma entrada para outro neurónio como mostra a figura 2.6.

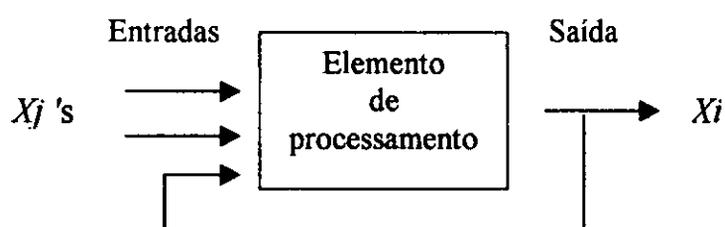


Figura 2.6: Componentes de um neurónio.

**Definição 2.8 (Entrada):-** A Entrada é um valor numérico inicial ao neurónio; cada entrada corresponde a um único atributo a ter em conta durante o processamento da informação.

**Definição 2.9 (Saída):-** A Saída é um valor que representa o resultado do processamento da informação.

**Definição 2.10 (Peso):-** O Peso é um valor matemático que expressa a relação entre dois neurónios ligados entre si.

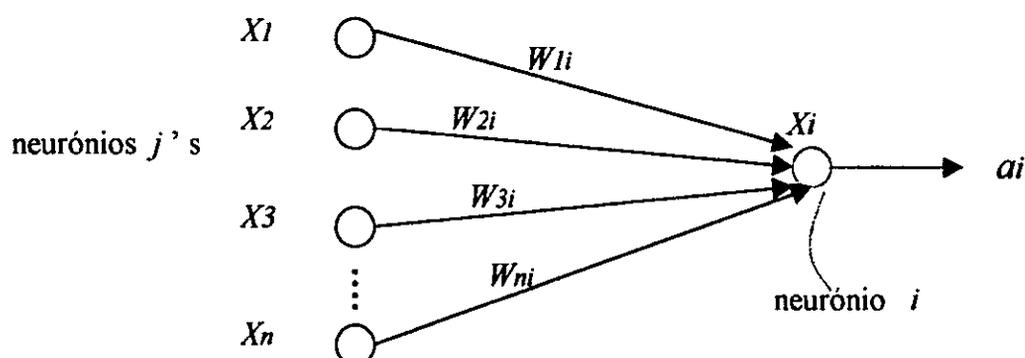


Figura 2.7: Ligações entre os neurónios

Considere-se a rede dada pela figura 2.7; o estado do neurónio  $i$  no momento  $t$ , designa-se estado de activação de  $i$  e denota-se por  $X_i(t)$ . A saída do neurónio  $i$  no momento  $t$  é denotada por  $a_i(t)$ . Ela é determinada aplicando uma função  $f_i$  ao estado de activação. Essa função é denominada função de activação, as funções mais usadas são “step” e “sigmoid”.

$$a_i(t) = f_i[ X_i(t) ]$$

Na função “step”, a saída do neurónio é igual a um se o estado de activação excede o limiar da função  $h_i$  e zero caso contrário.

$$f_i(x) = \theta(x - h_i) = \begin{cases} 1, & \text{se } x \geq h_i \\ 0, & \text{caso contrario} \end{cases}$$

Para a função “sigmoid” tem-se que

$$f_i(x) = \frac{1}{1 + e^{-[x - h_i]}}$$

O procedimento geral da rede é determinado pela estrutura e pesos das ligações entre os neurónios. Os pesos das ligações são representados através da matriz  $W$ , onde o seu  $ij$ -ésimo componente  $W_{ij}$  representa o peso da ligação entre o neurónio  $j$  e o neurónio  $i$ . Se  $W_{ij}$  é igual a zero, então não existe uma ligação directa entre os neurónios  $j$  e  $i$ .

O sinal de entrada entre  $j$  e  $i$  é dado por  $W_{ij}a_j(t)$ . Este sinal e o estado de activação corrente do neurónio, determinam um novo estado de acordo com uma regra de activação. A regra mais usada é aquela em que o estado de activação do neurónio no tempo  $t+1$  é dado pela soma linear de todas as entradas no tempo  $t$ .

$$X_i(t+1) = \sum_{j=1}^N W_{ij}a_j(t) + I_i$$

onde  $I_i$  representa uma entrada externa, que pode ser dada por um dispositivo analógico ou por outras partes do sistema de processamento de informação no qual a rede neural está embebida.

Em redes neurais onde os neurónios competem para se tornarem activos, tem-se uma forma iterativa de se obter o estado de activação:

$$X_i(t+1) = \sum_{j=1}^N W_{ij}f_j[ X_j(t) ] + I_i$$

Assim, dados os valores iniciais de todas as activações, as futuras activações podem ser calculadas, numa rede neural em que os pesos e entradas são fixas. Neste caso, a matriz de pesos  $W$  constitui aquilo que a rede 'conhece' e com ela pode-se determinar a saída de uma entrada arbitrária.

Geralmente, não é possível especificar a "priori" os pesos das ligações, uma vez que o conhecimento é distribuído por toda a rede, em vez de ser representado localmente. Por isso, é necessário um procedimento de aprendizagem para modificar, através de experiências, a relação entre os neurónios, de forma a obter as activações.

Outro aspecto que caracteriza a rede neural é a forma como os neurónios estão organizados, ou seja, a arquitectura ou estrutura da rede. Existem duas arquitecturas padrão: redes "feed-forward" de múltiplas camadas e redes recorrentes (figura 2.8).

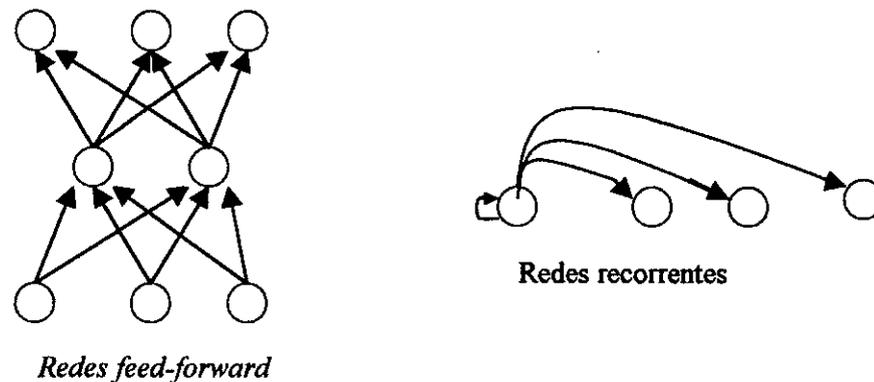


Figura 2.8: Arquitecturas de uma rede neural

As redes "feed-forward" de múltiplas camadas são constituídas por uma camada de entrada, uma de saída e um número, maior ou igual a zero, de camadas intermédias ou ocultas. Nas redes "feed-forward" os neurónios estão ligados unidireccionalmente formando um grafo direccionado acíclico (Johnsonbaugh, 1990); nestas redes, cada neurónio é ligado apenas aos neurónios da camada seguinte, não existem ligações entre neurónios da mesma camada ou de camadas anteriores.

Nas redes recorrentes a saída de um neurónio serve de entrada aos neurónios que lhe alimentaram ou seja, que lhe serviram de entrada, isto significa que a rede é alimentada tanto para frente como para trás. Esta é a maneira como o cérebro funciona.

A arquitectura da rede e o tipo de dados usados pela rede determinam o algoritmo de aprendizagem a usar durante o processo de treino da rede.

**Definição 2.11 (Algoritmo de Aprendizagem):-** Um Algoritmo de Aprendizagem, "*Learning Algorithms*", fornece os passos e parâmetros a seguir durante o processo de aprendizagem da rede neural. Os algoritmos de aprendizagem podem ser divididos em aprendizagem supervisionada e aprendizagem não supervisionada (Sarle, 1997).

A aprendizagem supervisionada actua como se existisse um 'professor' a orientar a aprendizagem. Na aprendizagem não supervisionada a rede é autónoma, ela aprende as propriedades do conjunto de dados sem nenhuma orientação e reflecte-as na sua saída.

**Definição 2.12 (Processo de Aprendizagem):-** O Processo de Aprendizagem ou Treino da rede neural é o processo no qual a matriz dos pesos da rede  $W$ , é ajustada por forma a produzir uma saída desejável (Trippi e Turban, 1996).

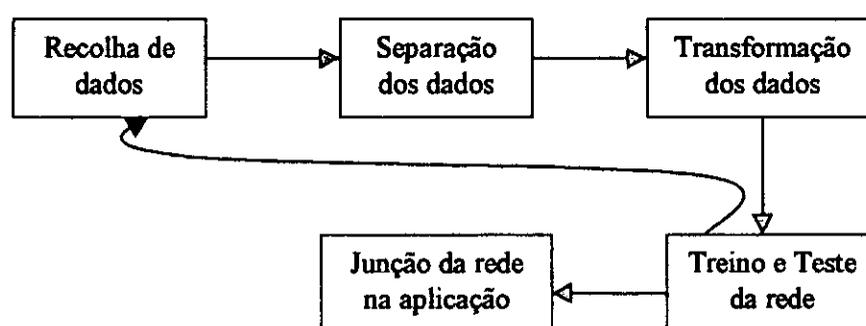
A matriz dos pesos  $W$  é ajustada com base nos exemplos fornecidos à rede, tendo em conta uma taxa de aprendizagem, "*learning rate*". Este parâmetro determina o tempo em que o processo de aprendizagem é feito. Durante o processo de treino da rede, a taxa de aprendizagem pode-se manter fixa ou pode variar. Quando ela varia ao longo do treino, é actualizada usando a diferença entre a saída actual da rede e a saída desejável.

Tendo os conceitos básicos sobre as redes neurais, a secção seguinte apresenta um processo para o desenvolvimento de uma rede neural.

## 2.2.2. Processo de desenvolvimento de uma rede neural

O processo de desenvolvimento de uma rede neural envolve os seguintes passos (Trippi e Turban, 1996): recolha dos dados, separação dos dados em conjunto de dados para o treino e conjunto de dados para o teste, transformação dos dados, treino e teste da rede e a junção da rede desenvolvida na aplicação.

Antes de juntar a rede desenvolvida na aplicação, os passos anteriores podem ser repetidos, as vezes que forem necessárias, conforme a figura 2.9.



*Figura 2.9: Passos usados no desenvolvimento de uma rede neural*

Na recolha dos dados, procura-se todos dados relevantes ao sistema. Os dados recolhidos devem cobrir as operações de rotina e também as condições e excepções na fronteira do domínio do problema.

Os dados recolhidos são separados aleatoriamente em um conjunto de treino e um conjunto de teste. O conjunto de dados para o treino é usado para ajustar a matriz de pesos  $W$ , de modo a se efectuar o processo de aprendizagem da rede; os dados para o teste são usados para validar a rede, permitindo a verificar a sua actuação com dados não conhecidos pela rede.

Durante a recolha dos dados deve-se decidir a quantidade de exemplos necessários para o conjunto de treino e para o conjunto de teste. Se os exemplos disponíveis não forem suficientes para a aprendizagem da rede, pode-se obter grandes desvios durante o teste da rede.

Uma outra decisão a tomar é a escolha da função de activação a ser aplicada. A selecção da função de activação é determinada pela natureza dos dados e por aquilo que a rede deve aprender.

Por exemplo, se a rede deve dar o resultado de uma operação booleana OR, a melhor função a aplicar é a “*step*” que dá valores zero ou um; mas se o problema envolve a aprendizagem de ‘médias’ a melhor função a aplicar é a “*sigmoid*”.

A rede neural trabalha apenas com valores numéricos, cujo domínio depende do domínio de aplicação da função de activação escolhida, por isso, antes de usar os dados recolhidos, deve-se transformá-los.

Os dados não numéricos devem ser convertidos para valores numéricos, por exemplo, um atributo cujo valor seja Sim ou Não, pode ser convertido para 0 (correspondente a Não) e 1 (correspondente a Sim).

Uma outra forma de converter dados não numéricos é a seguinte: suponha-se que se pretende converter o atributo estado civil; o estado civil admite quatro valores possíveis (solteiro, casado, viuvo e divorciado), então este atributo pode ser convertido para quatro entradas na rede, cada uma das quais correspondente a um valor possível. Se por exemplo o estado civil é ‘solteiro’, a entrada da rede correspondente a ‘solteiro’ receberá o valor 1 e as restantes entradas terão valor 0.

Os dados numéricos cujos valores possíveis pertencem a um intervalo diferente ao do domínio da função de activação, tem que ser convertidos.

Após a transformação do dados, deve-se escolher arquitectura da rede, escolhendo o número de nós que cada camada deve ter e o número de camadas ocultas (se necessário).

Ao se escolher a arquitectura da rede, deve-se escolher a taxa de aprendizagem. Estas são duas tarefas bastante difíceis porque geralmente não se sabe quantos neurónios são necessários, se é necessário ou não ter camadas ocultas, se o parâmetro escolhido é o mais adequado, e outros. Estes dados, podem depois serem optimizados se o teste da rede apresentar grandes desvios.

Tendo a arquitectura da rede, deve-se escolher o algoritmo de aprendizagem a usar durante o processo de treino da rede. Existem diversos algoritmos tanto para a aprendizagem supervisionada como para a aprendizagem não supervisionada.

O algoritmo mais conhecido é o “*back-propagation*” (Russel e Norving, 1995). O “*back-propagation*” é um método de aprendizagem supervisionada que permite o treino de redes “*feed-forward*” com múltiplas camadas. Este algoritmo é uma generalização do “*perceptron*” (Russel e

Norving, 1995), que é um algoritmo de aprendizagem para redes “*feed-forward*” com duas camadas (entrada e saída).

No processo de treino da rede usando o algoritmo “*back-propagation*”, são executados os seguintes passos: cálculo das saídas, comparação das saídas com as respostas desejáveis e o ajuste dos pesos para que o processo se repita.

Inicialmente, os pesos são atribuídos aleatoriamente; gera-se a saída da rede e compara-se com a saída desejável; se as saídas forem diferentes, ajusta-se os pesos da rede e volta-se a gerar uma outra saída. O objectivo é minimizar (de preferência que seja zero), a diferença entre a saída gerada pela rede e a saída desejável; isto é feito alterando incrementalmente os pesos.

Depois da execução do treino faz-se o teste da rede para examinar a sua performance; isto é feito usando os pesos derivados, durante o treino, para medir a habilidade da rede na classificação dos dados de teste. Se o teste da rede revelar grandes desvios, o conjunto de treino deve ser reexaminado e o processo de treino reactivado e, em certos casos, deve-se analisar a arquitectura da rede e a taxa de aprendizagem escolhida.

Uma das partes mais importantes no desenvolvimento de uma rede neural é o processo de treino da rede. Para melhor compreensão deste processo, considere-se uma rede neural que resolve uma operação booleana OR.

Caso	Entradas		Z – Saídas desejáveis
	X <sub>1</sub>	X <sub>2</sub>	
1	0	0	0
2	0	1	1
3	1	0	1
4	1	0	1

Tabela 2.1: Definição da operação booleana OR

O conjunto de dados possíveis para a operação OR e as respectivas saídas desejáveis, são dados pela tabela 2.1. Como os valores que a operação booleana admite são 0 ou 1 então, a melhor função de activação a aplicar é a função “*step*”.

Como são necessários apenas dois valores para a execução da função, a rede deve ter duas entradas  $X_1$  e  $X_2$ , e uma saída  $Y$ , como mostra a figura 2.10.

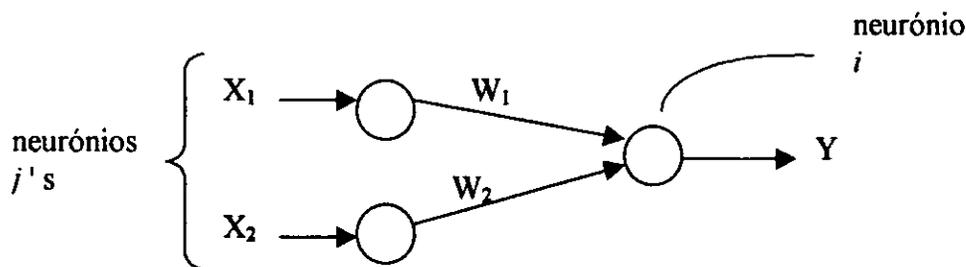


Figura 2.10: Rede neural para a operação OR

A matriz de pesos da rede  $W$  é composta por

$$W = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

que inicialmente atribuem-se valores aleatórios. Por exemplo,

$$W = \begin{pmatrix} 0.1 \\ 0.3 \end{pmatrix}$$

Para o treino da rede, vai-se usar o algoritmo de aprendizagem "perceptron". Nele, os pesos são inicialmente atribuídos valores aleatórios. Calcula-se a saída da rede e compara-se com a saída desejável. Depois os pesos são actualizados tendo em conta o erro obtido.

Para o treino da rede, considera-se  $\alpha$ , a taxa de aprendizagem, igual a 0.2 e a função de activação

$$F(k) = \begin{cases} 1, & \text{se } k \geq 0.5 \\ 0, & \text{se } k < 0.5 \end{cases}$$

onde  $K = \sum W_j X_j$ .

No momento  $t$ , a saída da rede será dada por

$$Y = a_i(t) = F(\sum W_j X_j)$$

onde  $j = 1, 2$ .

Supondo que no primeiro momento,

$$\begin{cases} X_1 = 0 \\ X_2 = 0 \\ t = 1 \end{cases}$$

então,

$$Y = a_i(1) = F(0.1 \cdot 0 + 0.3 \cdot 0) = F(0) = 0$$

A diferença entre a saída da rede e a saída desejável é dada por

$$\Delta = Z - Y \quad \text{onde } Z \text{ é a saída desejável}$$

Para

$$t = 1,$$

$$\Delta = 0 - 0 = 0$$

Usando essa diferença, actualiza-se os pesos, usando para tal a seguinte fórmula:

$$W_j (\text{final}) = W_j (\text{inicial}) + \alpha \Delta X_j$$

Então

$$W_1 (\text{final}) = 0.1 + 0.2 * 0 * 0 = 0.1 \quad \text{e}$$

$$W_2 (\text{final}) = 0.3 + 0.2 * 0 * 0 = 0.3$$

No momento seguinte, se

$$\begin{cases} X_1 = 0 \\ X_2 = 1 \\ t = 2 \end{cases}$$

então

$$Y = ai(2) = F(0.1 * 0 + 0.3 * 1) = F(0.3) = 0$$

$$\Delta = 1 - 0 = 1$$

logo

$$W_1 (\text{final}) = 0.1 + 0.2 * 1 * 0 = 0.1 \quad \text{e}$$

$$W_2 (\text{final}) = 0.3 + 0.2 * 1 * 1 = 0.5$$

Os dados obtidos podem ser resumidos na tabela 2.2, que é parte da primeira iteração do processo de treino da rede.

X <sub>1</sub>	X <sub>2</sub>	Z	Pesos iniciais		Y	Δ	Pesos finais	
			W <sub>1</sub>	W <sub>2</sub>			W <sub>1</sub>	W <sub>2</sub>
0	0	0	0.1	0.3	0	0.0	0.1	0.3
0	1	1	0.1	0.3	0	1.0	0.1	0.5

*Tabela 2.2: Parte da primeira iteração do processo de treino*

Os valores obtidos no fim de quatro iterações são dados pela tabela 2.3. Observe-se que na quarta iteração, a diferença entre a saída da rede e a saída desejável é nula, para todas as entradas possíveis. Assim, usando os pesos derivados por essa iteração, a rede neural pode fornecer, para qualquer entrada, o valor correcto da operação booleana OR.

Iteração	X <sub>1</sub>	X <sub>2</sub>	Z	Pesos iniciais		Y	Δ	Pesos finais	
				W <sub>1</sub>	W <sub>2</sub>			W <sub>1</sub>	W <sub>2</sub>
1	0	0	0	0.1	0.3	0	0.0	0.1	0.3
	0	1	1	0.1	0.3	0	1.0	0.1	0.5
	1	0	1	0.1	0.5	0	1.0	0.3	0.5
	1	1	1	0.3	0.5	1	0.0	0.3	0.5
2	0	0	0	0.3	0.5	0	0.0	0.3	0.5
	0	1	1	0.3	0.5	0	1.0	0.3	0.7
	1	0	1	0.3	0.7	0	1.0	0.5	0.7
	1	1	1	0.5	0.7	1	0.0	0.5	0.7
3	0	0	0	0.5	0.7	0	0.0	0.5	0.7
	0	1	1	0.5	0.7	1	0.0	0.5	0.7
	1	0	1	0.5	0.7	0	1.0	0.7	0.7
	1	1	1	0.7	0.7	1	0.0	0.7	0.7
4	0	0	0	0.7	0.7	0	0.0	0.7	0.7
	0	1	1	0.7	0.7	1	0.0	0.7	0.7
	1	0	1	0.7	0.7	1	0.0	0.7	0.7
	1	1	1	0.7	0.7	1	0.0	0.7	0.7

**Tabela 2.3: resultado do treino da rede neural**

### **2.3. Comparação entre sistemas baseados no conhecimento e redes neurais**

Existe uma diferença sensível entre uma rede neural e um sistema baseado no conhecimento; A forma como o conhecimento é representado pode ser a mesma mas, o modo que aquisição é diferente.

Em um sistema baseado no conhecimento, o engenheiro do conhecimento compõe as regras com base no conhecimento adquirido através da observação de peritos da área. Este processo pode ser moroso e caro, uma vez que se leva semanas, meses e as vezes anos até obter essa informação. O processo de alteração da base de conhecimentos também pode ser um processo complexo, as vezes sendo necessário voltar a programar.

Os sistemas de redes neurais não possuem uma base de conhecimento predefinida; o computador 'aprende' com base em exemplos de um domínio de aplicação e depois gera regras para a base de conhecimento. A adição de novas informações é feita voltando a treinar a rede, incorporando novos exemplos ou adicionando nova informação aos exemplos já existentes.

A maioria dos sistemas baseados no conhecimento usam regras para representar o conhecimento abstracto por isso, eles não conseguem realmente lidar com conhecimento inexacto, inconsistente ou incompleto; ao contrário das redes neurais que, durante a fase de treino da rede podem obter uma situação completa partindo de dados incompletos.

As redes neurais também têm as suas desvantagens, a grande dificuldade que se enfrenta é a incapacidade do sistema de indicar os passos usados para chegar a solução do problema Esses passos existem na forma de uma matriz de pesos (das ligações entre os neurónios); esta matriz, de momento, não pode ser traduzida num algoritmo compreensível ou útil fora da rede.

Um outro problema encontrado em uma rede neural é que, ao contrário de um sistema baseado no conhecimento ou qualquer programa computacional convencional, ela não permite que se chegue a cálculos intermédios ou que se faça o "debug" do processamento. Para se testar a consistência e fidelidade do sistema deve-se monitorar a saída.

A escolha de uma ou de outra abordagem depende da proporção do conhecimento existente para a solução do problema.

Se o conhecimento está pouco formalizado e existem poucos peritos na área, então é conveniente o uso de redes neurais. Mas se existe grande conhecimento sobre o domínio de aplicação, então o melhor é usar um sistema baseado no conhecimento.

Sem exemplos uma rede neural não funciona mas, um sistema baseado no conhecimento pode ser construído. Sem o conhecimento de um domínio de aplicação, um sistema baseado no conhecimento não pode ser construído mas, uma rede neural pode ser aplicada se existirem exemplos sobre o domínio.

Combinando as características de cada abordagem, Trippi e Turban (1996), apresentam um sistema híbrido para a análise da situação financeira de uma empresa. O sistema incorpora o processamento simbólico, processamento paralelo distribuído e uma base de dados (figura 2.11).

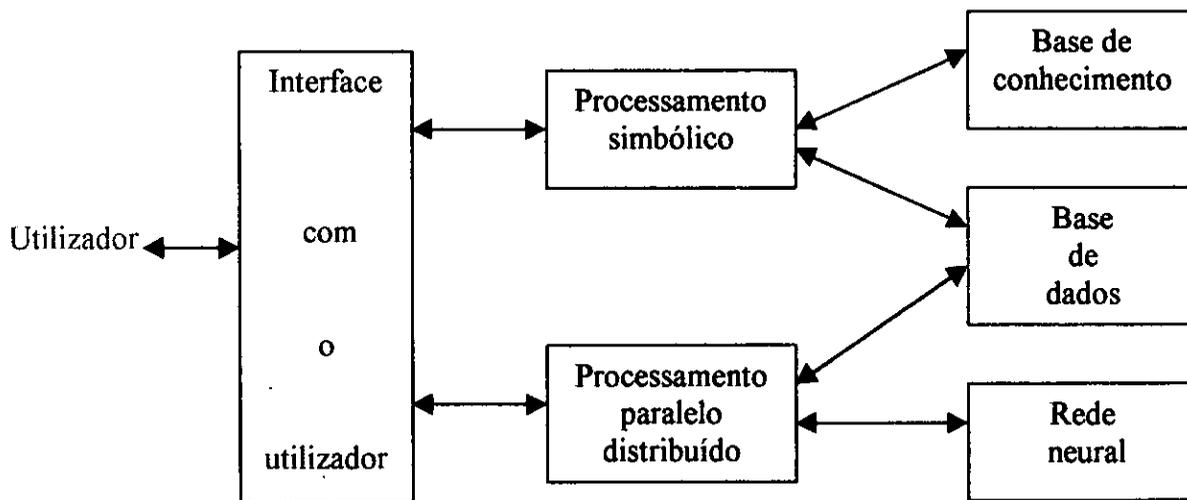


Figura 2.11: Diagrama de um sistema híbrido

Para a construção do sistema híbrido usou-se um "shell" para as redes neurais, o NeuroShell da Ward Systems, um "shell" para o sistema baseado no conhecimento, o KnowledgePro da Knowledge Garden, e a linguagem de programação Dbase III Plus.

O NeuroShell permite o armazenamento de dados em ficheiros ASCII e a sua transferência para outros programas; ele possui uma versão "run-time" que pode ser chamada noutra programa e correr em "background".

O KnowledgePro possui uma linguagem de comandos para a manipulação de símbolos. Esses comandos, permitem que dados importados de um ficheiro possam ser organizados para o

processamento simbólico. Este “*shell*” permite também a formatação de dados em diferentes estruturas de ficheiros, de modo a exportá-los para outros programas.

A comunicação entre os dois “*shells*” é feita através da linguagem de programação que coordena o funcionamento do sistema híbrido.

Os parâmetros usados para a análise financeira de uma empresa são armazenados em uma base de dados comum (figura 2.11). A rede neural treina essa informação e classifica-a. A interpretação dessa informação é feita através do processamento simbólico, que para tal, faz uso do conhecimento existente na base de conhecimento.

### **3. Solução do problema**

No capítulo 1 deste trabalho foi apresentado um problema existente no sistema informático, SIGES, a funcionar na Emose. Neste capítulo apresentar-se-á uma forma de solucionar esse problema.

O problema consiste em determinar a ordem correcta de execução das alterações, uma vez que, para um conjunto de alterações solicitadas na mesma acta adicional, diferentes ordens de execução de alterações podem ter valores diferentes do recibo emitido.

Cada alteração possui uma definição, que descreve as acções a serem executadas quando a alteração é efectuada e o resultado dessas acções.

Em geral, a execução de uma alteração resulta na modificação de um dos seguintes dados: Valor do capital seguro, valor da taxa a aplicar ao capital seguro, período da duração da apólice ou valor do prémio total.

Essa informação é dada, implicitamente, pela definição da alteração. Por exemplo: a alteração do uso da viatura, modifica a taxa a aplicar ao capital seguro, uma vez que a taxa pode sofrer agravamentos dependendo do uso da viatura. Outro exemplo é: a alteração do titular da viatura para além de modificar os dados do titular da apólice e do condutor habitual da viatura, modifica os descontos e os bónus da apólice, estes dados modificam o prémio total da apólice.

Segundo a análise dos casos apresentados pela secção 1.2, a primeira alteração a ser executada é aquela que modifica o período da apólice. Isto porque o cálculo do valor do recibo é feito com base nesse período e, o objectivo é encontrar a ordem de execução das alterações que forneça o valor correcto do recibo a emitir.

Depois de se efectuar as alterações que modificam o período da apólice, deve-se efectuar as alterações que modificam a taxa a aplicar ao capital seguro. Depois desta, efectuam-se as alterações que modificam o prémio total.

Para se determinar a ordem correcta das alterações, cada alteração deve ser classificada de forma a obter a posição correcta em que ela deve ser executada. Isto é feito usando uma rede neural.

Para que a rede neural forneça um resultado correcto, ela precisa de conhecimento sobre a alteração que está a analisar e também conhecimento sobre as restantes alterações solicitadas. Essa informação é fornecida por um sistema baseado no conhecimento.

Assim, obtém-se um sistema híbrido, que é uma forma de obter a ordem correcta de execução das alterações. O sistema híbrido possui um sistema baseado no conhecimento, um sistema de redes neurais e um interface de ligação, como mostra a figura 3.1.

O interface de ligação, permite a ligação entre o sistema híbrido e o utilizador ou o SIGES, isto porque o sistema híbrido pode estar embebido no SIGES ou pode funcionar independentemente.

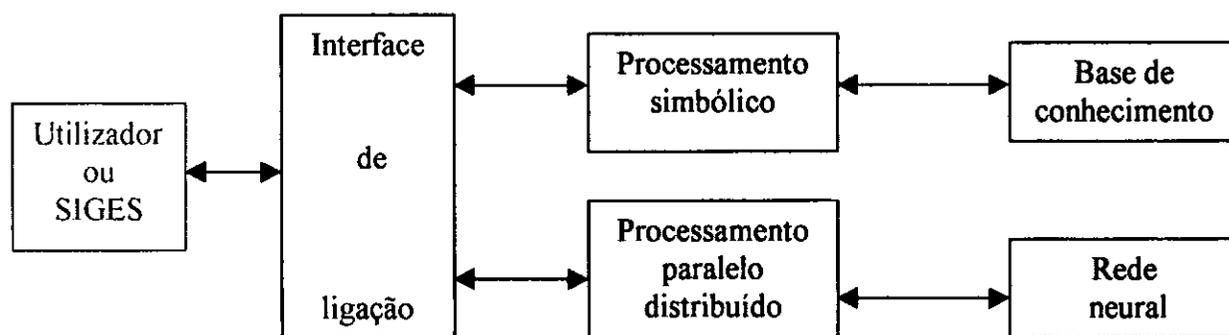


Figura 3.1: Arquitectura do sistema híbrido

O sistema baseado no conhecimento faz o processamento simbólico da informação. Ele recebe as alterações que devem ser ordenadas, analisa-as e fornece toda a informação que a rede neural necessita para a classificação da alteração. Ele faz isso usando uma base de conhecimento que contém toda a informação relacionada com as alterações.

O sistema de redes neurais faz o processamento paralelo e distribuído da informação. A rede neural determina se a alteração pode ou não ser efectuada no momento, com base na informação de entrada que recebe do sistema baseado no conhecimento.

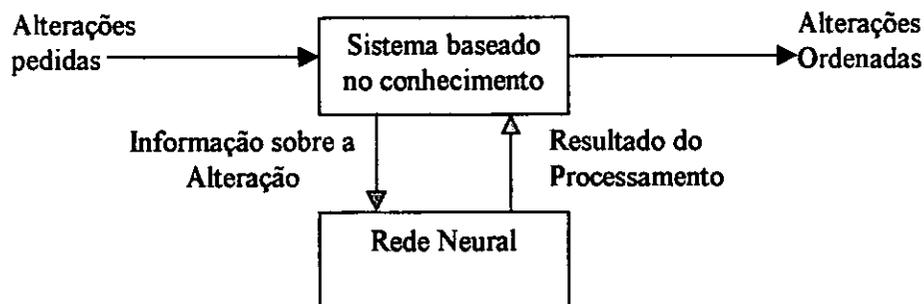


Figura 3.2: Diagrama do funcionamento do sistema híbrido

Se a alteração não pode ser efectuada no momento, então ela é colocada no fim do grupo das alterações pedidas e uma outra alteração é fornecida à rede. Caso contrário ela é removida do grupo

das alterações pedidas, é colocada no fim do grupo das alterações ordenadas e outra alteração é fornecida à rede.

Assim, quando o grupo de alterações pedidas não tiver nenhuma alteração, o grupo das alterações ordenadas terá a ordem correcta de execução das alterações. Esta ordem é enviada para fora do sistema híbrido, através do interface com o utilizador, como mostra a figura 3.2.

As secções seguintes deste capítulo apresentam o funcionamento de cada parte do sistema híbrido.



Se a alteração pode ser efectuada no momento, ela é retirada do grupo das alterações solicitadas e posta no fim do grupo das alterações ordenadas e uma nova alteração é fornecida à rede. Isto é feito até que o grupo de alterações solicitadas esteja vazio.

### 3.2. Uso de uma rede neural na classificação das alterações

A rede neural usa a informação de uma alteração, proveniente do sistema baseado no conhecimento para determinar se a alteração pode ou não ser efectuada de momento.

Para este sistema, a rede neural deve possuir três camadas: entrada, intermédia e saída, como mostra a figura 3.3.

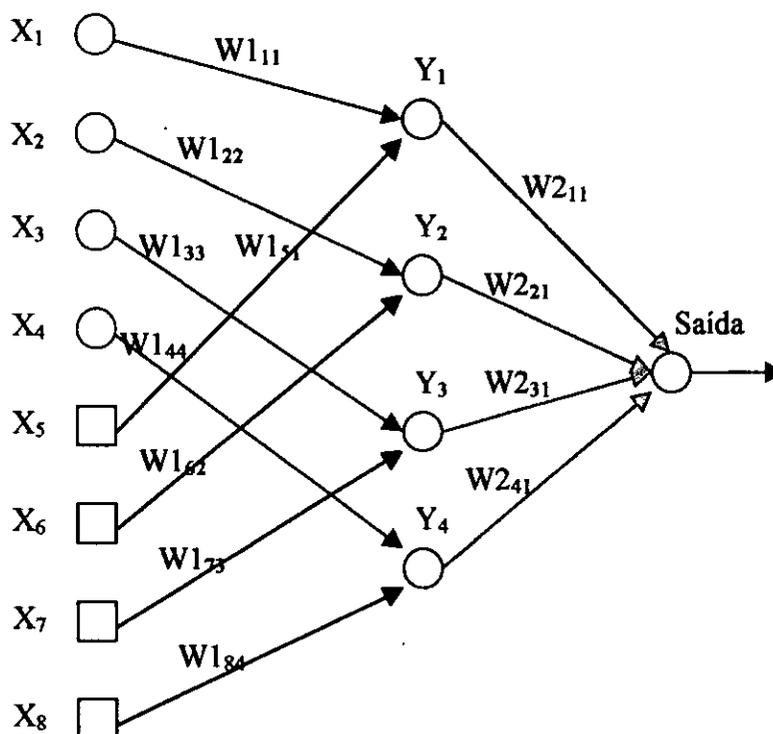


Figura 3.3: Estrutura da rede neural

As entradas  $X_1$ ,  $X_2$ ,  $X_3$  e  $X_4$  contêm informação sobre a alteração que a rede está a analisar. E as entradas  $X_5$ ,  $X_6$ ,  $X_7$  e  $X_8$  contêm informação sobre as restantes alterações solicitadas. Estas entradas são adicionadas à rede por ser necessário relacionar cada alteração com todo o grupo das alterações solicitadas.

A entrada  $X_1$  da rede, indica se a alteração modifica o prémio total da apólice. Ele toma valor um se a alteração modifica esse dado ou valor zero caso contrário.

Análogamente, a entrada  $X_2$  indica se a alteração modifica ou não o valor do capital seguro.  $X_3$  indica se a alteração modifica ou não o valor da taxa a aplicar ao capital seguro e  $X_4$  indica se a alteração modifica ou não o período de duração da apólice.

A entrada  $X_5$  da rede, indica se existe alguma alteração (nas restantes), que modifica o valor do prémio total da apólice.  $X_6$  indica se existe alguma alteração que modifica o valor do capital seguro,  $X_7$  o valor da taxa a aplicar ao capital seguro e  $X_8$  o período de duração da apólice.

Cada uma destas entradas ( $X_5$ ,  $X_6$ ,  $X_7$  e  $X_8$ ) é combinada com a entrada correspondente da alteração para formar os neurónios da camada oculta. Por exemplo, a entrada  $X_5$ , que indica se existe uma alteração que modifique o valor do prémio total, é combinada com a entrada  $X_1$ , que indica se a alteração modifica o valor do prémio total da apólice.

O neurónio resultante dessa combinação fornece a contribuição do prémio total sobre fazer-se ou não a alteração no momento, ou seja, se a alteração que se está a analisar modifica o prémio da apólice e existe, nas restantes alterações, uma que também modifica, então, pelo prémio total da apólice, a alteração pode ser efectuada no momento.

Se a alteração não modifica o prémio da apólice e no resto das alterações existe uma que o modifique, então pelo prémio da apólice, a alteração não pode ser efectuada no momento. A Tabela 3.1, fornece os casos possíveis, onde o resultado da combinação é um se a alteração pode ser efectuada no momento e zero caso contrário.

$X_1$	$X_5$	$Y_1$
1	1	1
0	1	0
1	0	1
0	0	1

Tabela 2.1: Definição da operação booleana OR

Assim obtém-se os restantes neurónios da camada oculta. Os neurónios desta camada são, por sua vez, combinados para fornecer o resultado final da rede.

Os pesos das ligações entre os neurónios da camada de entrada e a camada oculta são denotados por  $w1_{ij}$ , onde  $i$  indica os neurónios da camada de entrada e  $j$  os da camada oculta. Os pesos das ligações entre a camada oculta e a camada de saída são denotados por  $w2_{jk}$ , onde  $j$  indica os neurónios da camada oculta e  $k$  os da camada de saída.

A secção seguinte mostra passo a passo como o processo de aprendizagem é feito, usando para tal o algoritmo "backpropagation".

### 3.2.1. Aprendizagem da rede

Para que a rede classifique correctamente é necessário que ela aprenda. Para isso deve-se treinar a rede de modo a obter os pesos correctos. Neste caso, vai-se usar para o efeito o algoritmo de aprendizagem "backpropagation". Este algoritmo possui os seguintes passos (Russel e Norving, 1995):

➤ Para cada exemplo fornecido faça

- Calcule  $O$ , a saída para esse exemplo;
- Calcule  $Err$ , o erro obtido para cada neurónio da camada de saída

$$Err \leftarrow T - O \quad \text{onde } T \text{ é a saída desejável da rede}$$

- Actualize os pesos das ligações dos neurónios da camada de saída

$$W_{ij} \leftarrow W_{ij} + \alpha * a_i * Err_i * f'(k)$$

$$\{ \text{onde } f' \text{ é a primeira derivada da função de activação e } k = \sum (W_{ij} * a_i) \}$$

- Para cada camada subsequente da rede faça

- Calcule  $\Delta$ , o erro para cada neurónio

$$\Delta_i \leftarrow f'(k) \sum_j W_{ij} \Delta_j$$

- Actualize os pesos das ligações da camada

$$W_{zi} \leftarrow W_{zi} + \alpha * \alpha_k * \Delta_i$$

Para o caso que pretende-se solucionar, vai-se considerar a taxa de aprendizagem,  $\alpha$  igual a 0.2 e para a função de activação vai-se considerar a função "step", com o limiar 0.5, isto é,

$$F(k) = \begin{cases} 1, & \text{se } k \geq 0.5 \\ 0, & \text{se } k < 0.5 \end{cases}$$

Considere-se que são solicitadas as seguintes alterações: Inclusão dos passageiros gratuitos, alteração do capital de responsabilidade civil, alteração do titular da viatura, substituição da viatura e alteração da data de vencimento da apólice. Este constitui o grupo de alterações solicitadas.

A inclusão dos passageiros gratuitos e a alteração do titular da viatura, modificam o valor do prémio da apólice; a alteração do capital de responsabilidade civil modifica o valor do capital seguro; a substituição da viatura modifica o valor da taxa a aplicar ao capital seguro e a alteração da data de vencimento da apólice modifica o período de duração da apólice.

Inicialmente os pesos das matrizes são atribuídos valores aleatórios, assim, seja:

$$W1 = \begin{pmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.3 & 0 \\ 0 & 0 & 0 & 0.4 \\ 0.5 & 0 & 0 & 0 \\ 0 & 0.2 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.8 \end{pmatrix} \quad W2 = \begin{pmatrix} 0.6 \\ 0.1 \\ 0.5 \\ 0.4 \end{pmatrix}$$

A primeira alteração a ser avaliada é a inclusão dos passageiros gratuitos, que modifica o prémio total da apólice. As restantes alterações modificam todos os dados, assim, a rede neural recebe de entrada:

$$\begin{cases} X_1 = 1 \\ X_2 = 0 \\ X_3 = 0 \\ X_4 = 0 \\ X_5 = 1 \\ X_6 = 1 \\ X_7 = 1 \\ X_8 = 1 \end{cases}$$

Seguindo o algoritmo tem-se

$$Y_1 = f(X_1 * W1_{11} + X_5 * W1_{51}) = f(1 * 0.1 + 1 * 0.5) = f(0.6) = 1$$

$$Y_2 = f(X_2 * W1_{22} + X_6 * W1_{62}) = f(0 * 0.5 + 1 * 0.2) = f(0.2) = 0$$

$$Y_3 = f(X_3 * W1_{33} + X_7 * W1_{73}) = f(0 * 0.3 + 1 * 0.5) = f(0.5) = 1$$

$$Y_4 = f(X_4 * W1_{44} + X_8 * W1_{84}) = f(0 * 0.4 + 1 * 0.8) = f(0.8) = 1$$

As saídas desejáveis para estes neurónios são

$$\begin{cases} Z_{Y1} = 1 \\ Z_{Y2} = 0 \\ Z_{Y3} = 0 \\ Z_{Y3} = 0 \end{cases}$$

A saída da rede é dada por

$$\begin{aligned} \text{Saida} &= f(Y_1 * W2_{11} + Y_2 * W2_{21} + Y_3 * W2_{31} + Y_4 * W2_{41}) = f(1 * 0.6 + 0 * 0.1 + 1 * 0.5 + 1 * 0.4) \\ &= f(1.5) = 1 \end{aligned}$$

A saída desejável T, da rede é zero, agora, calcula-se o erro obtido e actualiza-se os pesos das ligações. Como a função "step" não possui primeira derivada, vai usar a seguinte fórmula para o cálculo do erro

$$\text{Err} = T - \text{saida}$$

assim tem-se que:

$$\text{Err} = 0 - 1 = -1$$

Actualizando os pesos das ligações da camada de saída, obtém-se

$$W_{211} = W_{211} + \alpha * Y_1 * \text{Err} = 0.6 + 0.2 * 1 * (-1) = 0.6 - 0.2 = 0.4$$

$$W_{221} = W_{221} + \alpha * Y_2 * \text{Err} = 0.1 + 0.2 * 0 * (-1) = 0.1$$

$$W_{231} = W_{231} + \alpha * Y_3 * \text{Err} = 0.5 + 0.2 * 1 * (-1) = 0.5 - 0.2 = 0.3$$

$$W_{241} = W_{241} + \alpha * Y_4 * \text{Err} = 0.4 + 0.2 * 1 * (-1) = 0.4 - 0.2 = 0.2$$

O erro da camada seguinte é dado por

$$\Delta_1 = Y_1 - Z_{Y1} = 1 - 1 = 0$$

$$\Delta_2 = Y_2 - Z_{Y2} = 0 - 0 = 0$$

$$\Delta_3 = Y_3 - Z_{Y3} = 1 - 0 = 1$$

$$\Delta_4 = Y_4 - Z_{Y4} = 1 - 0 = 1$$

assim,

$$W_{111} = W_{111} + \alpha * X_1 * \Delta_1 = 0.1 + 0.2 * 1 * 0 = 0.1$$

$$W_{122} = W_{122} + \alpha * X_2 * \Delta_2 = 0.5 + 0.2 * 0 * 0 = 0.5$$

$$W_{133} = W_{133} + \alpha * X_3 * \Delta_3 = 0.3 + 0.2 * 0 * 1 = 0.3$$

$$W_{144} = W_{144} + \alpha * X_4 * \Delta_4 = 0.4 + 0.2 * 0 * 1 = 0.4$$

$$W_{151} = W_{151} + \alpha * X_5 * \Delta_1 = 0.5 + 0.2 * 1 * 0 = 0.7$$

$$W_{162} = W_{162} + \alpha * X_6 * \Delta_2 = 0.2 + 0.2 * 1 * 0 = 0.2$$

$$W_{173} = W_{173} + \alpha * X_7 * \Delta_3 = 0.5 + 0.2 * 1 * 1 = 0.7$$

$$W_{184} = W_{184} + \alpha * X_8 * \Delta_4 = 0.8 + 0.2 * 1 * 1 = 1.0$$

Assim as matrizes passam a ter os seguintes valores

$$W1 = \begin{pmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.3 & 0 \\ 0 & 0 & 0 & 0.4 \\ 0.7 & 0 & 0 & 0 \\ 0 & 0.2 & 0 & 0 \\ 0 & 0 & 0.7 & 0 \\ 0 & 0 & 0 & 1.0 \end{pmatrix} \quad W2 = \begin{pmatrix} 0.4 \\ 0.1 \\ 0.3 \\ 0.2 \end{pmatrix}$$

Os valores obtidos na análise desta alteração, podem ser esquematizados na tabela 3.2.

X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	X <sub>4</sub>	X <sub>5</sub>	X <sub>6</sub>	X <sub>7</sub>	X <sub>8</sub>	Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>	Y <sub>4</sub>	Saída desejável de Y				Erro obtido em Y				Saída	T	Err	
												Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>	Y <sub>4</sub>	Δ <sub>1</sub>	Δ <sub>2</sub>	Δ <sub>3</sub>	Δ <sub>4</sub>				
1	0	0	0	1	1	1	1	1	0	1	1	1	0	0	0	0	0	0	1	1	1	0	-1

Tabela 3.2: Dados do treino da primeira alteração

O valor obtido na saída da rede (zero), significa que a inclusão dos passageiros gratuitos não pode ser efectuada neste momento. Continuando com o treino, é fornecida à rede uma outra alteração, a alteração do capital de responsabilidade civil. Procedendo do mesmo modo, obtém-se os valores apresentados pela tabela 3.3, com as seguintes matrizes de pesos:

$$W1 = \begin{pmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.3 & 0 \\ 0 & 0 & 0 & 0.4 \\ 0.9 & 0 & 0 & 0 \\ 0 & 0.2 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.8 \end{pmatrix} \quad W2 = \begin{pmatrix} 0.2 \\ -0.1 \\ 0.1 \\ 0 \end{pmatrix}$$

X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	X <sub>4</sub>	X <sub>5</sub>	X <sub>6</sub>	X <sub>7</sub>	X <sub>8</sub>	Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>	Y <sub>4</sub>	Saída desejável de Y				Erro obtido em Y				Saída	T	Err
												Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>	Y <sub>4</sub>	Δ <sub>1</sub>	Δ <sub>2</sub>	Δ <sub>3</sub>	Δ <sub>4</sub>			
0	1	0	0	1	0	1	1	1	1	1	1	0	1	0	0	1	0	-1	-1	1	0	-1

***Tabela 3.3: Dados resultantes da alteração do capital de responsabilidade civil***

O processo de treino prossegue, fornecendo mais alterações, até que a rede possua pesos, que para uma entrada arbitrária, o erro produzido seja mínimo.

Tendo os pesos correctos, o sistema híbrido é capaz de determinar a ordem correcta de execução das alterações, para qualquer grupo de alterações solicitadas na apólice de seguro.

#### **4. Conclusões e recomendações**

Ao final deste trabalho pode-se tirar as seguintes conclusões:

1. A determinação da ordem correcta de execução das alterações aos dados da apólice de seguro, depende do grau do conhecimento e da experiência do técnico de seguros.
2. Os casos de estudo apresentados no trabalho, demonstram a importância da determinação da ordem correcta de execução das alterações de apólices na gestão de seguros.
3. As técnicas de Inteligência Artificial permitem solucionar problemas não estruturados, onde o conhecimento sobre o domínio do problema e a experiência de um grupo de pessoas são indispensáveis para a sua solução.
4. Uma das técnicas de Inteligência Artificial muito usada é o sistema baseado no conhecimento. Este permite solucionar problemas relacionados com o diagnóstico, tomada de decisões, planeamento, desenho, e outros.
5. A aquisição e representação do conhecimento são os processos fundamentais na construção de um sistema baseado no conhecimento.
6. Um outro método de Inteligência Artificial é a rede neural que, para a solução de um problema não necessita de uma base de conhecimento predefinida.  
Para fornecer uma solução, a rede 'aprende' as características do problema através de exemplos.
7. Uma proposta de solução para o problema da ordem das alterações é a criação de um sistema híbrido, composto por um sistema baseado no conhecimento e uma rede neural.
8. O sistema baseado no conhecimento, do sistema híbrido, verifica para cada alteração solicitada, se a alteração modifica ou não o valor do prémio total da apólice, o valor do capital seguro, a taxa a aplicar ao capital seguro e o período de duração da apólice.  
Para além da informação sobre os dados que a alteração modifica, o sistema baseado no conhecimento fornece à rede neural, informação sobre os dados que as restantes alterações solicitadas modificam.

9. A rede neural do sistema híbrido, recebe a informação a partir do sistema baseado no conhecimento, e determina se a alteração pode ou não ser efectuada no momento.  
Se a alteração pode ser efectuada no momento, é retirada do grupo das alterações solicitadas e colocada no grupo das alterações ordenadas. Caso contrário, se ela não pode ser efectuada no momento, é colocada no fim do grupo das alterações solicitadas.  
Para os dois casos, uma nova alteração é fornecida à rede neural.
10. A rede neural apresentada no sistema híbrido, possui oito entradas e uma saída. Ela é uma rede "*feed-forward*" com três camadas, entrada, intermédia e saída.
11. Usando exemplos reais, a aprendizagem da rede é feita usando o algoritmo de aprendizagem "*backpropagation*", modificando assim a matriz de pesos da rede.

As recomendações a serem feitas ao trabalho aqui apresentado são:

- Testar a solução apresentada. Optimizar a estrutura da rede neural e a escolha do algoritmo de aprendizagem com base no resultado do teste.
- Deve-se aprofundar os meios que conduziram a escolha da taxa de aprendizagem utilizada no processo de aprendizagem da rede neural do sistema híbrido.
- Implementar o sistema híbrido, escolhendo um "*shell*" apropriado para o sistema baseado no conhecimento e um para a rede neural.

## 5. Referências bibliográficas

- [Bauer, Nouak e Winkler, 1996] Bauer, P., S. Nouak e R. Winkler (1996). A Brief Course in Fuzzy Logic and Fuzzy Control, versão 1.2. <http://www.fll.uni-linz.ac.at/fuzzy/fuzzy.html>.
- [Booch, 1994] Booch, G. (1994). Object-Oriented Analysis and Design with Applications, 2ª Edição, 589 pp. EUA, The Benjamin/Cumming Publishing company, Inc.
- [Horstkotte, 1996] Horstkotte, E. (1996). Frequently Asked Questions About Fuzzy Logic. <http://www.antinlinks.com/fuzzy/faq.html>.
- [Johnsonbaugh, 1990] Johnsonbaugh, R. (1990). Discrete Mathematics, 2ª Edição, 705 pp. EUA, Macmillan Publishing Company.
- [Kantrowitz, 1997] Kantrowitz, M. (1997). Frequently Asked Questions About Expert Systems Shells. <ftp://ftp.cs.cmu./user/ai/pubs/faqs/ai/expert/faq.html>.
- [Kantrowitz, Horstkotte e Joslyn, 1993] Kantrowitz, M., E. Horstkotte, e C. Joslyn, (1993). Frequently Asked Questions About Fuzzy Logic & Fuzzy Expert Systems. <ftp://ftp.cs.cmu./user/ai/pubs/faqs/ai/expert/faq.html>
- [Karim (a)] Karim, O. (Não publicado). Introdução Geral aos Seguros, Volume 1.
- [Karim (b)] Karim, O. (Não publicado). Introdução Geral aos Seguros, Volume 2.
- [Macamo, 1996] Macamo, A. F. (1996). Pseudo- Metodologia de Análise Orientada- a- Objectos e sua Aplicação na Gestão de Seguros. Tese de Licenciatura. 77 pp. Maputo, Universidade Eduardo Mondlane.

- [Rich e Knight, 1991] Rich, E. e K. Knight (1991). Artificial Intelligence, 2ª Edição, 691 pp. Índia, Tata Mcgraw hill Edition.
- [Russell e Norving, 1995] Russell, S. e P. Norving (1995). Artificial Intelligence, A Modern Approach, 932 pp. EUA, Prentice Hall International Edition.
- [Sarle, 1997] Sarle, W. S. (1997). Frequently Asked Questions About Neural Networks.  
<ftp://ftp.sas.com/pub/neral/Faq.html>.
- [Stefik, 1995] Stefik, M. (1995). Introduction to Knowledge Systems, 870 pp. EUA, Morgan Kaufmann publishers, Inc.
- [Trippi e Turban, 1996] Trippi, R. R. e E. Turban (1996). Neural Networks in Finance and Investing, edição revisada, 821 pp. EUA, IRWIN.
- [Trippi e Lee, 1996] Trippi, R. R. e J. K. Lee (1996). Artificial Intelligence in Finance & Investing - state of the Art technologies for Securities Selection and Portfolio Management, 246 pp. EUA, Irwin.



**RAMO AUTOMÓVEIS** Número da proposta: 1234567890

Características do veículo a segurar: Número do assegurado: 1234567890

Matrícula: M 2 K 1 5 1 3 4 Tipo veículo: (Prescritivo pelos serviços)

Marca: 1 5 0 2 0 7 3 6 2 Modelo:

N.º motor: W E N D 4 4 5 5 Ano constr.: 1 9 9 6

N.º chassis: Forma: Aberta  Fechada

Força: 1 0 0 0 N.º de lugares: 4 N.º cilindros: 4 Carga Util: 1 2 0 0 Kg Uso: ALUGUER

Categoria: Pesado  Leveiro

**Preencher caso o veículo seja pesado**

Peso bruto: Capacidade de carga: (tons) Reboca atrelado?  Sim  Não Matrícula: 1 2 3 4 5 6 7 8 9 0

Cor do veículo: Valor: (Moeda)

O rádio está incluído  Sim  Não

**Preencher caso o veículo tenha rádio incluído**

Marca do rádio: Valor:

Incluída a quebra de vidros  Sim  Não

Incluído o roubo de acessórios  Sim  Não

Transporta passageiros gratuitos  Sim  Não

Crédor hipotecário

**Condutor Habitual do veículo**

Tipo e número do documento de identificação do condutor

Passaporte  D.I.R.E.  Bilhete de identificação N.º de documento: 1 2 3 4 5 6 7 8 9 0

Apelido: Telex:

Nome: Data de nascimento:

Morada: Data de emissão da carta:

Profissão: Local de emissão da carta:

N.º da carta de condução: Data de emissão da carta:

Tipo de carta:  A - Motos com ou sem carro Local de emissão da carta:

B - Leves

C - Pesados e Tractores

D - Pesados de passageiros

**Riscos e capitais que se pretendem segurar**

Código	Riscos	Capitais	Moeda
I	Responsabilidade Civil (Assinale com X escolhendo apenas um capital)	<input type="checkbox"/> 10 000 000	MT
		<input type="checkbox"/> 50 000 000	MT
		<input type="checkbox"/> 100 000 000	MT
		<input type="checkbox"/> 200 000 000	MT
		<input type="checkbox"/> 500 000 000	MT
<input type="checkbox"/> 1 000 000 000	MT		
Assinale com X os riscos a serem cobertos			
II	Risco II (Transportes Semi-Colectivos de Passageiros)		
	Dados Próprios	Valor dos Riscos	Franquia Facultativa
III	Choque, Colisão ou capotamento	III	
IV	Furto ou Roubo	IV	
V	Incêndio, Raio ou Explosão	V	
Q.I.V.	Quebra isolada de vidros		
R.L.A.	Roubo isolado de acessórios		
E.N.	Forças da Natureza		



2. Apólice de seguro

<b>APÓLICE N.º</b> 90055		<b>DATA DA EMISSÃO</b> 23/05/1997		<b>PRÊMIO SIMPLES</b> 520.000,00		
<b>NOME DO EX.º SEGURADO - MORADA</b> UNIVERSIDADE EDUARDO MORAES CAMPUS UNIVERSITÁRIO - CIDADE DE MAPUTO				<b>INÍCIO ÀS 12 HORAS DE</b> 01/02/1997 <b>DURAÇÃO</b> 08 ano e seguintes <b>TERMO ÀS 12 HORAS</b> 01/02/08 cada ano		
<b>VEÍCULO SEGURO:</b>		<b>PAGAMENTO E DATA DE VENCIMENTO DAS PRESTAÇÕES</b> UNICA 01/02/08 cada ano				
<b>MATRÍCULA</b> MLK-45-34		<b>RESPONSABILIDADES CIVIL</b> Garantia Máxima 500.000.000,00		<b>RISCOS E CAPITAIS SEGUROS</b> <b>DANOS PRÓPRIOS:</b> Choque, Colisão ou Capotamento Furto ou Roubo Incêndio, Rato ou Explosão		
<b>APARELHO RÁDIO-RECEPTOR FM E SEUS ACESSÓRIOS:</b> Marca - Valor		<b>FRANQUIA A CARGO DO SEGURADO NOS RISCOS DE DANOS PRÓPRIOS COM EXCLUSÃO DA QUEBRA ISOLADA DE VIDROS E ROUBO DE ACESSÓRIOS - ver cláusulas especiais n.º 3 e 4</b>				
<b>Passageiros</b>	<b>Quatro isolada de vidros</b>	<b>Roubo de acessórios</b>	<b>Motivo da alteração</b>			
S/N	Não	Não				
<b>CARACTERÍSTICAS DO VEÍCULO SEGURO</b>						
<b>N.º do motor</b>	<b>Ano de const.</b>	<b>N.º cil.</b>	<b>Força</b>	<b>Lug./Carga</b>	<b>Forma</b>	<b>Uso</b>
BEH1455	1994	4	1400	4/1200,00	Perchada	UTILIDADE
<b>CREDORES HIPOTECÁRIOS COM INTERESSE NESTE SEGURO</b>						
<b>CLAUSULAS ESPECIAIS APLICÁVEIS - ver texto no verso</b> N.º 001						
<b>OUTRAS DECLARAÇÕES</b> DANOS DECLARADOS PELA EMOSE: 1. NENHUM						

3. Acta Adicional

APÓLICE N.º		ACTA ADICIONAL		DATA DA EMISSÃO		PREMIO SIMPLES	
90056				23/06/1997		426.000,00	
NOME DO EX.º SEGURADO - MORADA						INICIO ÀS 12 HORAS DE 02/10/1997	
UNIVERSIDADE EDUARDO MONDLANE CAMPUS UNIVERSITARIO - CIDADE DE ALFUTO						DURAÇÃO um ano e seguintes TERMO ÀS 12 HORAS 01/02/de cada ano	
VEÍCULO SEGURO:		PAGAMENTO E DATA DE VENCIMENTO DAS PRESTAÇÕES					
MATRÍCULA		UNICA 01/02/97 cada ano					
N.º - 45-34		RISCOS E CAPITALIS SEGUROS					
MARCA		RESPONSABILIDADES CIVIL			DANOS PRÓPRIOS		
ISUZU/401		Gazetas Máxima			Choque, Colisão ou Capotamento Furto ou Roubo Incêndio, Razo ou Explosão		
APARELHO RÁDIO-RECEPTOR FIXO E SEUS ACESSÓRIOS		FRANQUIA A CARGO DO SEGURADO NOS RISCOS DE DANOS PRÓPRIOS COM EXCLUSÃO DA QUEBRA ISOLADA DE VIDROS E ROUBO DE ACESSÓRIOS - ver cláusulas especiais n.º 3 e 4					
Marca		Quebra isolada de vidros		Roubo de acessórios		Motivo de alteração	
SUA		NÃO		NÃO		ALTERAÇÃO DO USO-ALUGUER PARA PARTICULAR AUMENTO DE CAPITAL DE R. CIVIL	
CARACTERÍSTICAS DO VEÍCULO SEGURO							
N.º do motor		Ano de const.		N.º cil.		Força	
1004153		1994		4		1400	
						Lug./Carga	
						4/1200,00	
						Forma	
						Fechado	
						Uso	
						PARTICULAR	
CREDOR HIPOTECÁRIO COM INTERESSE NESTE SEGURO							
CLÁUSULAS ESPECIAIS APLICÁVEIS - ver texto no verso							
N.º 001							
OUTRAS DECLARAÇÕES							
DANOS DECLARADOS PELA ENOSE: NENHUM							
ESTA ACTA ADICIONAL FAZ PARTE INTEGRANTE DO CONTRATO DE SEGURO A QUE DIL RESPEITO							

#### **4. Pseudocódigo para o cálculo do prémio simples da apólice**

```
{ Ler Descontos }  
IF Tem_Descontos THEN  
    Desconto ←  $\Sigma$  Descontos  
END IF  
  
{ Ler Agravamentos }  
IF Tem_Agravamentos THEN  
    Agravamento ←  $\Sigma$  Agravamentos  
END IF  
  
IF Tem_Danos_Próprios THEN  
    TaxaDP ←  $\Sigma$  (Taxa de cada risco* Valor da Viatura)  
    PrémioSimplesDP ← TaxaDP + Agravamento - Desconto  
END IF  
  
PrémioSimplesRC ← TaxaRC + Agravamento - Desconto  
PrémioSimples ← PrémioSimplesDP + PrémioSimplesRC
```

### **5. Pseudocódigo para o cálculo do prémio total da apólice**

```
IF nº_prestações > 1 THEN
    PrémioPresta ← PrémioSimples/ nº_prestações
END IF
IF tem_bónus THEN
    IF anos_bónus > 4 THEN
        anos_bónus ← 4
    END IF
END IF
IF nº_prestações >1 THEN
    PrémioBónus ← anos_bónus *PrémioPresta * PercentagemBónus
    PrémioAgravado ← PrémioPresta + PrémioPresta * (2*nº_prestações/100)
ELSE
    PrémioAgravado ← PrémioSimples
    PrémioBónus ← anos_bónus *PrémioSimples * PercentagemBónus
END IF
PrémioTotal ← PrémioAgravado - PrémioBónus
PrémioTotal ← PrémioTotal + PrémioTotal * encargos/100
IF primeiro_recibo_contrato OR alteração THEN
    PrémioTotal ← PrémioTotal + PreçoApóliceActa
END IF
PrémioTotal ← PrémioTotal + PrémioTotal * Sobretaxa /100
PrémioTotal ← PrémioTotal + PrémioTotal * Selo /100
IF primeiro_recibo_contrato OR alteração THEN
    PrémioTotal ← PrémioTotal + PreçoCartãoRespCivil
END IF
```

## **6. Pseudocódigo para o cálculo do valor do recibo suplementar**

Período ← numero de dias a partir da data da alteração até a data da última prestação paga

ValorRecibo ← (PrémioSimplesActual – PrémioSimplesAnterior) \* Período

IF n°\_prestações > 1 THEN

    PrémioPresta ← ValorRecibo/ n°\_prestações

END IF

IF tem\_bónus THEN

    IF anos\_bónus > 4 THEN

        anos\_bónus ← 4

    END IF

END IF

IF n°\_prestações > 1 THEN

    PrémioBónus ← anos\_bónus \* PrémioPresta \* PercentagemBónus

    PrémioAgravado ← PrémioPresta + PrémioPresta \* (2\*n°\_prestações/100)

ELSE

    PrémioAgravado ← PrémioSimples

    PrémioBónus ← anos\_bónus \* PrémioSimples \* PercentagemBónus

END IF

ValorRecibo ← PrémioAgravado - PrémioBónus

ValorRecibo ← ValorRecibo + ValorRecibo\* Encargos/100

ValorRecibo ← ValorRecibo + PreçoApóliceActa

ValorRecibo ← ValorRecibo + ValorRecibo + Sobretaxa/100

ValorRecibo ← ValorRecibo + ValorRecibo\* Selo/100

ValorRecibo ← ValorRecibo + PreçoCartãoRespCivil

### **7. Pseudocódigo para o cálculo do valor do recibo de estorno**

Período ← numero de dias a partir da data da alteração até a data da última prestação paga

ValorRecibo ← (PrémioSimplesAnterior \* período /2) \* 365

IF apólice\_teve\_bónus THEN

    IF (anos\_bónus >4) THEN

        anos\_bónus ← 4

    END IF

    ValorRecibo ← ValorRecibo - (PrémioSimplesAnterior\* anos\_bónus\* PercentagemBónos)

END IF

IF Primeira\_Anuidade THEN

    IF (DataAlteração - DataVencimentoApólice) ≤ 3 meses THEN

        ValorRecibo ← ValorRecibo/2

    ELSE

        IF (3 meses < (DataAlteração - DataVencimentoApólice) ≤ 6 meses) THEN

            ValorRecibo ← ValorRecibo/4

        ELSE

            { Não Tem Estorno }

            Goto 10

        END IF

    END IF

ELSE

    ValorRecibo ← ValorRecibo/2

END IF

ValorRecibo ← ValorRecibo + ValorRecibo\* Encargos/100

ValorRecibo ← ValorRecibo + ValorRecibo + Sobretaxa/100

ValorRecibo ← ValorRecibo + ValorRecibo\* Selo/100

10 : { fim do algoritmo }

### **8. Listagem de alterações de apólices**

A Emose possui diversos dados que podem ser modificados dependendo do ramo considerado; as alterações mais solicitadas no ramo automóvel são:

- ☞ Alteração (aumento ou redução) do capital de responsabilidade civil;
- ☞ Inclusão ou exclusão de danos próprios;
- ☞ Alteração (aumento ou redução) do capital de danos próprios;
- ☞ Alteração do titular da viatura;
- ☞ Alteração do condutor habitual da viatura;
- ☞ Alteração da matrícula da viatura;
- ☞ Substituição da viatura;
- ☞ Alteração do uso da viatura: de particular para aluguer ou vice-versa;
- ☞ Alteração da data de vencimento da apólice;
- ☞ Inclusão ou exclusão na apólice os passageiros gratuitos;
- ☞ Alteração do número de prestações de pagamento da apólice;
- ☞ Alteração do local de cobrança;
- ☞ Alteração do nome da empresa;
- ☞ Alteração da morada do cliente;
- ☞ Alteração do vencimento fixo;
- ☞ Alteração do tipo de instituição: de entidade oficial para normal ou vice-versa, isto quando o cliente é uma instituição;
- ☞ Reposição do capital gasto;
- ☞ Anulação da apólice;
- ☞ Levantamento da anulação da apólice;

**9. Bibliografia não referenciada**

Lucena, C. (1987). Inteligência Artificial e Engenharia de Software, 305 pp. Brasil, Jorge Zahar Editor.

Keller, R. (1987). Expert System Technology, Development and application, 246 pp. EUA, Prentice Hall, Inc.

Weiss, S. M. e C. A. Kulikowski (1991). Computer Systems that Learn, Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning and Expert Systems, 223 pp. EUA, Morgan Kaufmann publishers, Inc.