



Instituto Politécnico de Castelo Branco
Escola Superior de Tecnologia

Escola Superior de Tecnologia
Curso de 1º Ciclo em Engenharia Informática
1º Semestre do 2º Ano
Unidade Curricular: Inteligência Artificial
Docente: Ana Paula Neves Ferreira da Silva
Ano Letivo 2022/2023
Trabalho Prático
Data de entrega: 4 de Janeiro

Aprendizagem utilizando uma rede neuronal

1. Objetivo

O trabalho prático de Inteligência Artificial tem como objetivo a aplicação dos conceitos apresentados nas aulas teórico-práticas sobre agentes aprendizes usando a linguagem de programação Python, introduzida nas aulas práticas. Para tal, os alunos devem utilizar uma **rede neuronal multicamada** de maneira que esta aprenda a **identificar o género de um animal cujas características são conhecidas**. É fornecido código que permite treinar uma rede para imitar funções binárias simples, o qual deverá ser adaptado a este novo problema.

2. O Problema

Neste trabalho pretendemos implementar um agente baseado em redes neurais **capaz de identificar o género (ave, peixe, etc...) de um animal cujas características (se tem penas, se produz leite, etc...) são conhecidas**. É fornecido código que define uma rede neuronal multicamada e implementa o mecanismo de treino por retro-propagação do erro. Também é fornecido **código exemplificativo de como utilizar uma rede neuronal para imitar funções booleanas simples como o AND, o OR e o XOR**.

Pretende-se adaptar este código para treinar uma **rede capaz de receber as características de um animal e corretamente identificar o seu género**. Para o treino é fornecido um **ficheiro adicional (zoo.txt)** que contém exemplos de **121 animais**, descritos por uma **lista de 18 atributos**. O primeiro atributo é o **nome do animal e não deverá ser utilizado no treino**. Os seguintes **16 atributos** são numéricos e deverão ser utilizados para criar as entradas para a rede. O **último atributo corresponde ao género do animal**. Os atributos são descritos num **segundo ficheiro fornecido (info.txt)**. A rede **deverá ter 7 saídas, uma para cada género possível (atributo 18)**. Por exemplo, quando a terceira saída da rede tiver o maior valor, isso querará dizer que o animal cujos dados foram fornecidos é um réptil (reptile).



Instituto Politécnico de Castelo Branco
Escola Superior de Tecnologia

Escola Superior de Tecnologia
Curso de 1º Ciclo em Engenharia Informática
1º Semestre do 2º Ano
Unidade Curricular: Inteligência Artificial
Docente: Ana Paula Neves Ferreira da Silva
Ano Letivo 2022/2023
Trabalho Prático
Data de entrega: 4 de Janeiro

3. Construção dos Conjuntos de Treino e Teste

Antes de iniciar esta primeira fase do trabalho, o aluno deverá analisar o código fornecido, o qual está comentado de forma a descrever o seu funcionamento. O aluno deverá ainda **testar o funcionamento das funções `train_and`, `train_or` e `train_xor`.**

Nesta primeira fase vamos implementar a função **`build_sets`** e uma função auxiliar chamada **`translate`** que irá ser chamada pela anterior. A função **`build_sets`** cria os conjuntos de treino e teste que irão ser utilizados no treino e avaliação da rede neuronal a partir dos dados armazenados no ficheiro **`zoo.txt`**. A função deve ler cada linha e transformá-la numa lista de valores, tendo em atenção o tipo dos valores lidos. A lista é posteriormente passada como argumento à função **`translate`**, a qual construirá um padrão de treino no formato adequado, discutido mais abaixo. A lista resultante será armazenada numa lista de padrões, cuja ordem deve ser randomizada (consultar método **`shuffle`** disponível no módulo **`random`**). Finalmente, a função deverá devolver duas listas, o conjunto de treino com os primeiros 67 padrões e o conjunto de teste com os restantes.

A função **`translate`** recebe cada lista de valores e transforma-a num padrão de treino. Cada padrão é uma lista com o seguinte formato **`[nome_do_animal, padrao_de_entrada, tipo_do_animal, padrao_de_saida]`**, em que os elementos da lista são obtidos da seguinte forma:

- **`nome_do_animal`** é uma string contida no primeiro valor da lista recebida como argumento;
- **`tipo_de_animal`** é uma string armazenada na última posição da mesma lista;
- **`padrao_de_entrada`** é uma lista de 0 e 1, contendo os valores dos atributos, sendo que o número de pernas deve também ser convertido numa lista de 0 e 1 (10 bits), concatenada com os restantes,

e.g. **`[0 0 0 0 1 0 0 0 0 0]`** -> 4 pernas;

- **`padrao_de_saida`** é uma lista de 0 e 1 que representa o tipo do animal, tendo 7 posições em que a única que estiver a 1 corresponde ao tipo do animal. E.g., **`[0 0 1 0 0 0 0]`** -> reptile, seguindo a ordem de valores do atributo apresentada no ficheiro **`info.txt`**.



Instituto Politécnico de Castelo Branco
Escola Superior de Tecnologia

Escola Superior de Tecnologia
Curso de 1º Ciclo em Engenharia Informática
1º Semestre do 2º Ano
Unidade Curricular: Inteligência Artificial
Docente: Ana Paula Neves Ferreira da Silva
Ano Letivo 2022/2023
Trabalho Prático
Data de entrega: 4 de Janeiro

Em resumo, e para exemplificar, a primeira linha do ficheiro zoo.txt deve resultar no seguinte padrão:

hair eggs airbone pulcator backbone venomous domestic
" [aardvark, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 4, 0, 0, 1, mammal] "

feather milk aquatic tooth breathe fin tail cube size
deve ser transformado em:

['aardvark', [1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1], 'mammal', [1, 0, 0, 0, 0, 0, 0, 0]]

nº de patas
tipo do animal

4. Treino da Rede Neural

Nesta segunda fase vamos implementar três funções. A primeira função chama-se **train_zoo**, recebe o conjunto de treino e deve criar a rede neuronal e chamar a função **iterate** para a treinar durante **n** iterações. Esta função funciona de forma semelhante às funções de treino fornecidas no código, devendo, em cada iteração de treino, percorrer todos os padrões armazenados no conjunto de treino, chamando a função **iterate**, a qual recebe um padrão de entrada (lista dos atributos do animal) e de saída (lista com a codificação binária do género do animal).

A segunda função, **test_zoo**, tem como objetivo testar o desempenho da rede treinada. Para tal deve percorrer o conjunto de teste, e, para cada padrão, chamar a função **forward** passando como argumentos a rede treinada e a lista de atributos do animal. De seguida, a função deve analisar a lista de saídas da rede (`net['y']`), chamando a terceira função a implementar (**retranslate**). Esta função deve devolver o género do animal correspondente à saída da rede com maior valor. A função **test_zoo** deve comparar o valor devolvido pela função auxiliar com o verdadeiro género do animal, armazenado no padrão de treino, calculando a percentagem de respostas corretas. Para cada padrão deve ainda imprimir uma linha semelhante à seguinte:

The network thinks mongoose is a mammal, it should be a mammal

Após a impressão das linhas com o resultado referente a cada padrão de treino, deve ainda imprimir o valor do desempenho final da rede (percentagem de respostas corretas), por exemplo:

Success rate: 94.12



Instituto Politécnico de Castelo Branco
Escola Superior de Tecnologia

Escola Superior de Tecnologia
Curso de 1º Ciclo em Engenharia Informática
1º Semestre do 2º Ano
Unidade Curricular: Inteligência Artificial
Docente: Ana Paula Neves Ferreira da Silva
Ano Letivo 2022/2023
Trabalho Prático
Data de entrega: 4 de Janeiro

Por último, a função `run`, que será a função principal do nosso programa, deverá reunir as funções anteriormente implementadas: desde a construção dos conjuntos de treino e teste, até ao treino e teste da rede.

5. O que deve ser entregue

Os alunos devem submeter no moodle o ficheiro `.py` com o código desenvolvido (os nome dos alunos deve ser colocado em comentário no início do ficheiro), assim como um ficheiro PDF com um pequeno relatório, com no máximo 3 páginas, onde apresentam os resultados obtidos no trabalho experimental realizado. No trabalho experimental devem efetuar experiências para 3 redes diferentes e para cada uma devem apresentar o desempenho da rede ao fim de 100, 200 e 300 iterações, para três valores diferentes da constante η que define o ritmo de aprendizagem. De notar que o relatório deve descrever a arquitetura das 3 redes treinadas. Os resultados do trabalho experimental devem ser acompanhados de uma pequena reflexão sobre os mesmos.

O relatório deve também ser entregue em papel.



Instituto Politécnico de Castelo Branco
Escola Superior de Tecnologia

Escola Superior de Tecnologia
Curso de 1º Ciclo em Engenharia Informática
1º Semestre do 2º Ano
Unidade Curricular: Inteligência Artificial
Docente: Ana Paula Neves Ferreira da Silva
Ano Letivo 2022/2023
Trabalho Prático
Data de entrega: 4 de Janeiro

```
; Number of Instances: 101
; Number of Attributes: 18 (animal name 15 Boolean attributes 2 numerics)
; Attribute Information: (name of attribute and type of value domain)

; Attribute animal {aardvark antelope bass bear boar buffalo calf carp catfish cavy cheetah chicken chub clam crab crayfish crow deer dogfish dolphin dove duck
elephant flamingo flea frog fruitbat giraffe girl gnat goat gorilla gull haddock hamster hare hawk herring honeybee housefly kiwi ladybird lark leopard lion lobster lynx
mink mole mongoose moth newt octopus opossum oryx ostrich parakeet penguin pheasant pike piranha pitviper platypus polecat pony porpoise puma pussycat
raccoon reindeer rhea scorpion seahorse seal sealion seasnake seawasp skimmer skua slowworm slug sole sparrow squirrel starfish stingray swan termite toad
tortoise tuatara tuna vampire vole vulture wallaby wasp wolf worm wren}

; Attribute hair { 0 1}
; Attribute feathers { 0 1}
; Attribute eggs { 0 1}
; Attribute milk { 0 1}
; Attribute airborne { 0 1}
; Attribute aquatic { 0 1}
; Attribute predator { 0 1}
; Attribute toothed { 0 1}
; Attribute backbone { 0 1}
; Attribute breathes { 0 1}
; Attribute venomous { 0 1}
; Attribute fins { 0 1}
; Attribute legs {0 1 2 3 4 5 6 7 8 9}
; Attribute tail { 0 1}
; Attribute domestic { 0 1}
; Attribute catsize { 0 1}
; Attribute type { mammal bird reptile fish amphibian insect invertebrate } (

; Instances (101):
```

17 ostrinta totais