

Plataformas Cognitivas e Ferramentas de Desenvolvimento de IA

Prof. Dr. Peterson Belan
belan@uni9.pro.br

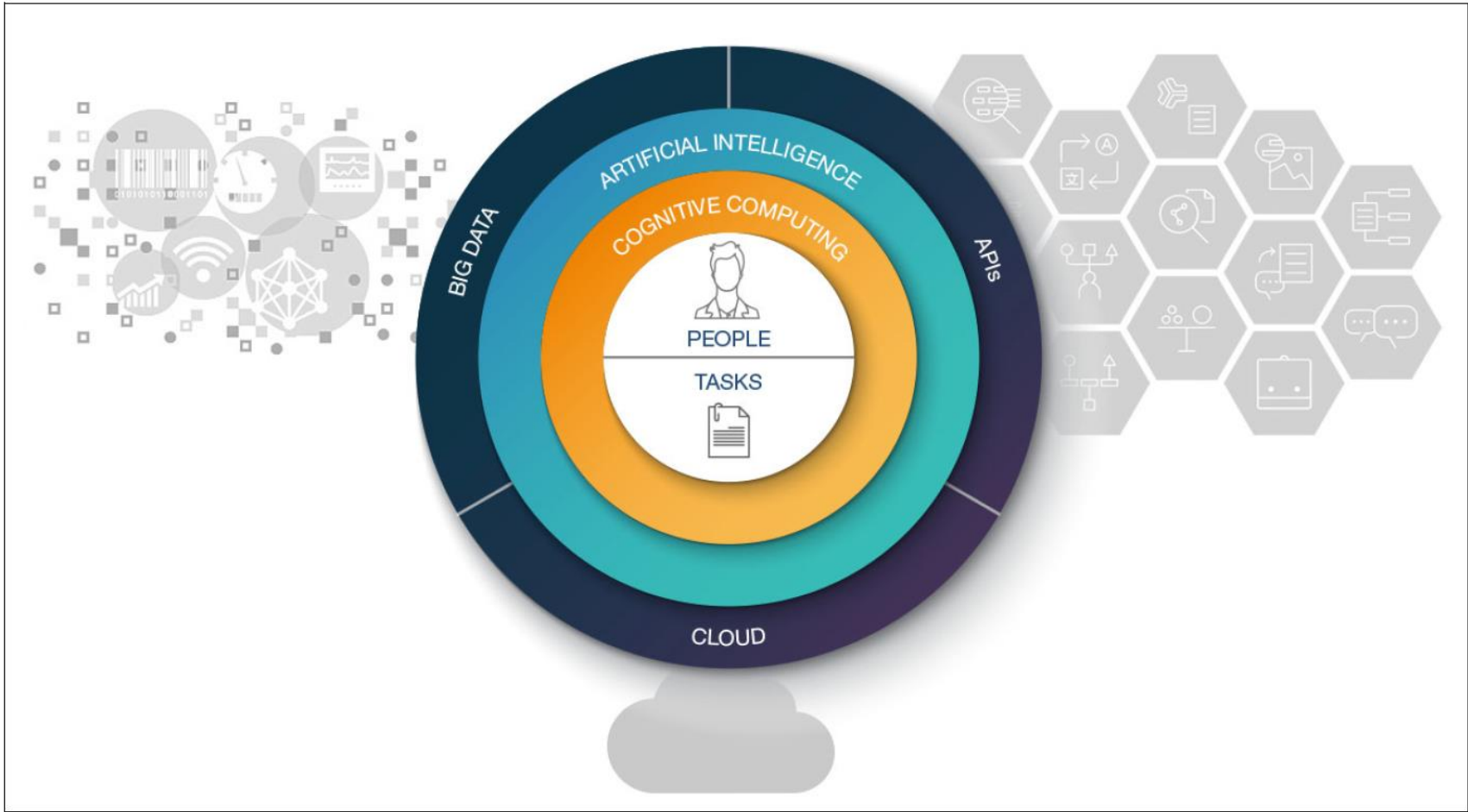
Plataformas Cognitivas

A cognição consiste no processo que a mente humana utiliza para adquirir conhecimento a partir de informações recebidas.

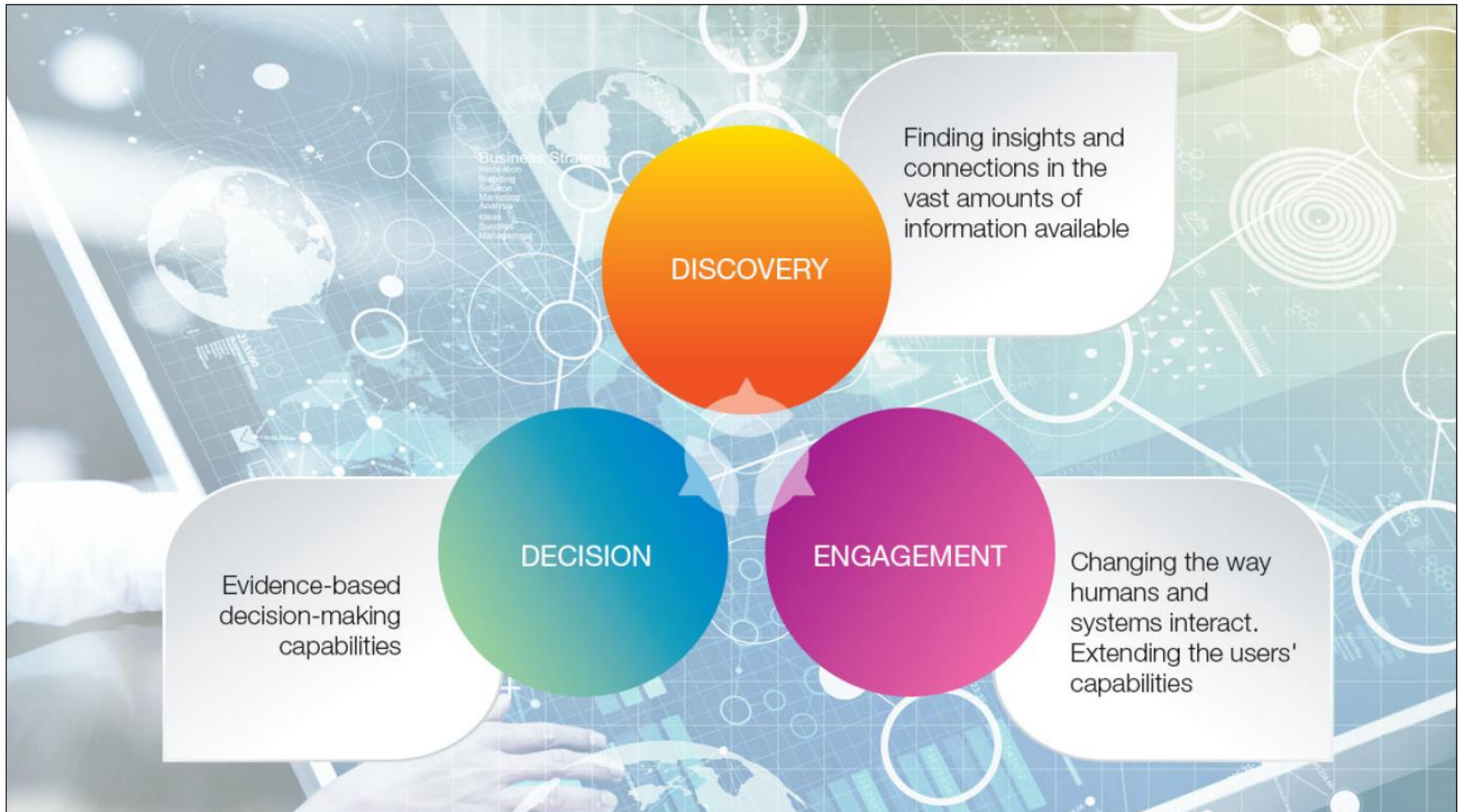
As Plataformas Cognitivas, entre as quais IBM Cloud, Microsoft Azure e Google Cloud, estão cada vez mais em evidência como ferramentas poderosas para prototipação e desenvolvimento de aplicações de Inteligência Artificial (IA).

Nessa disciplina, além de explorar algumas dessas plataformas, abordaremos também algumas ferramentas como o Weka (uso livre) e o Matlab (MathWorks) para a implementação e análise de algoritmos de IA e de Aprendizagem de Máquina.

Plataformas Cognitivas



Plataformas Cognitivas



IBM Watson

<https://www.ibm.com/watson/br-pt/>

IBM Watson



Conversation

Add a natural language interface to your application to automate interactions with your end users.

IBM



Language Translator

Translate text from one language to another for specific domains.

IBM



Personality Insights

The Watson Personality Insights derives insights from transactional and social media data to identify

IBM



Text to Speech

Synthesizes natural-sounding speech from text.

IBM



Discovery

Add a cognitive search and content analytics engine to applications.

IBM



Natural Language Classifier

Natural Language Classifier performs natural language classification on question texts. A user

IBM



Retrieve and Rank

Add machine learning enhanced search capabilities to your application

IBM



Tone Analyzer

Tone Analyzer uses linguistic analysis to detect three types of tones from communications: emotion, attitude

IBM



Document Conversion

Converts a HTML, PDF, or Microsoft Word™ document into a normalized HTML, plain text, or

IBM



Natural Language Understanding

Analyze text to extract meta-data from content such as concepts, entities, emotion, relations, sentiment

IBM



Speech to Text

Low-latency, streaming transcription

IBM



Visual Recognition

Find meaning in visual content! Analyze images for scenes, objects, faces, and other content. Choose

IBM

IBM Watson

Watson é a plataforma de serviços cognitivos da IBM para negócios. Algumas das APIs disponíveis atualmente:

Idioma:

- Conversação
- Conversão de documento
- Tradutor de idiomas
- Compreensão da linguagem natural
- Percepções de personalidade

Discurso:

- Fala para texto
- Texto para fala

Visão:

- Reconhecimento Visual

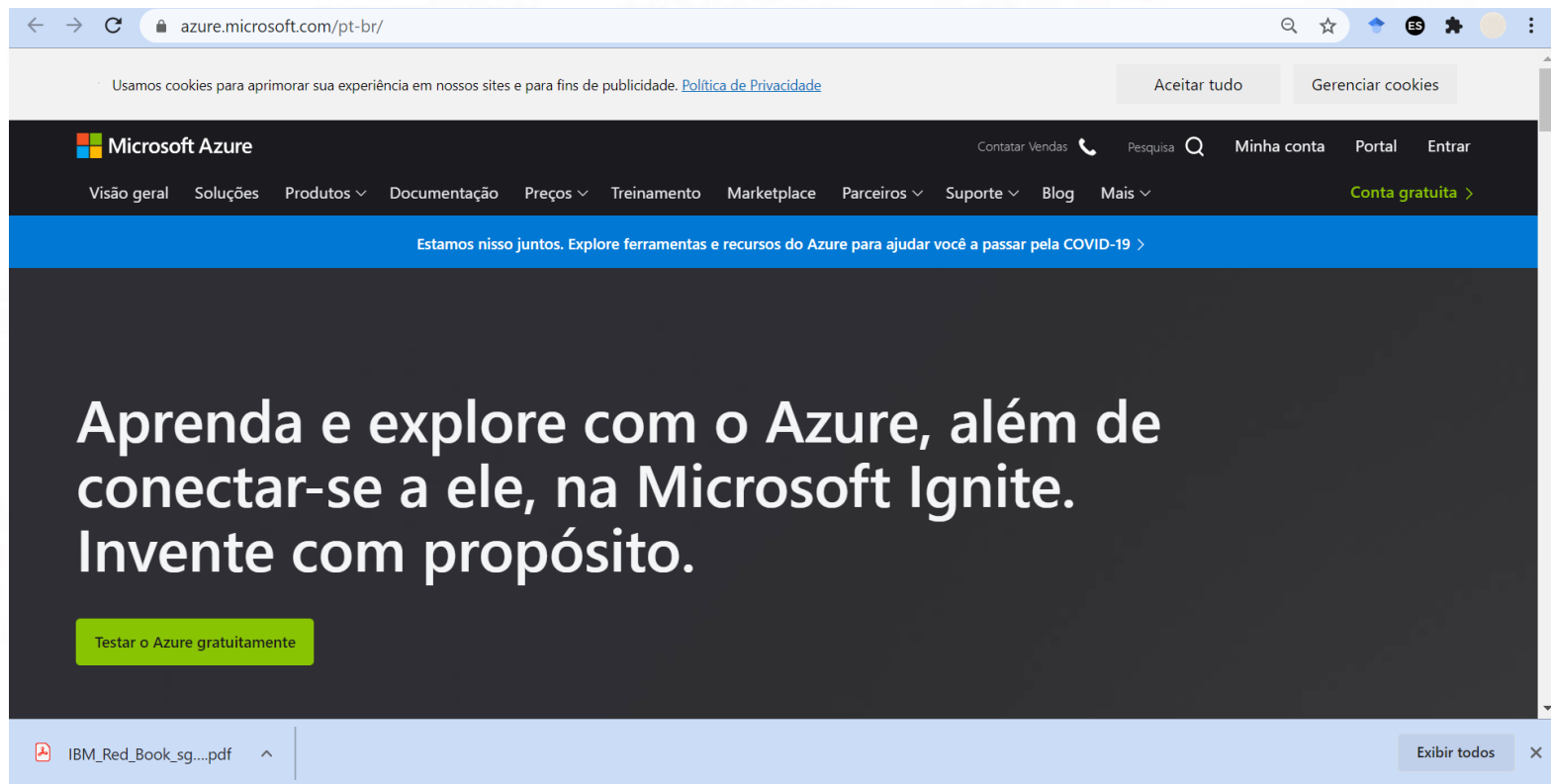
Insights de dados:

- Descoberta
- Discovery News

<https://www.ibm.com/watson/br-pt/>

Microsoft Azure

O Microsoft Azure é uma plataforma destinada à execução de aplicativos e serviços, baseada nos conceitos da computação em nuvem.



Weka¹

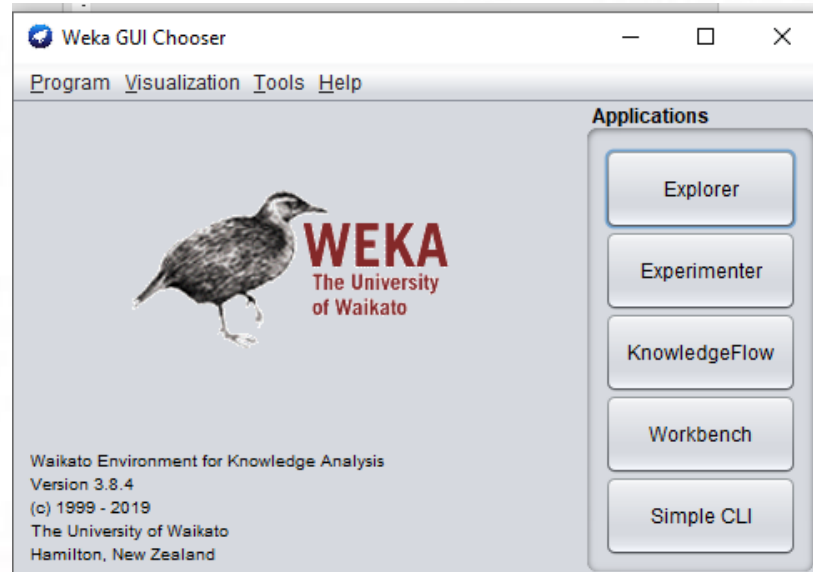
O Weka (*Waikato Environment for Knowledge Analysis*) é uma ferramenta que agrega algoritmos de inteligência artificial e de aprendizagem de máquina.

Trata-se de uma das mais populares ferramentas de mineração de dados em ambiente acadêmico.

O Weka teve seu desenvolvimento iniciado em 1993, usando Java, na Universidade de Waikato, Nova Zelândia sendo adquirido posteriormente por uma empresa no final de 2006. O Weka encontra-se licenciado ao abrigo da General Public License sendo portanto possível estudar e alterar o respectivo código fonte.

1. <https://www.cs.waikato.ac.nz/ml/weka/>

Weka



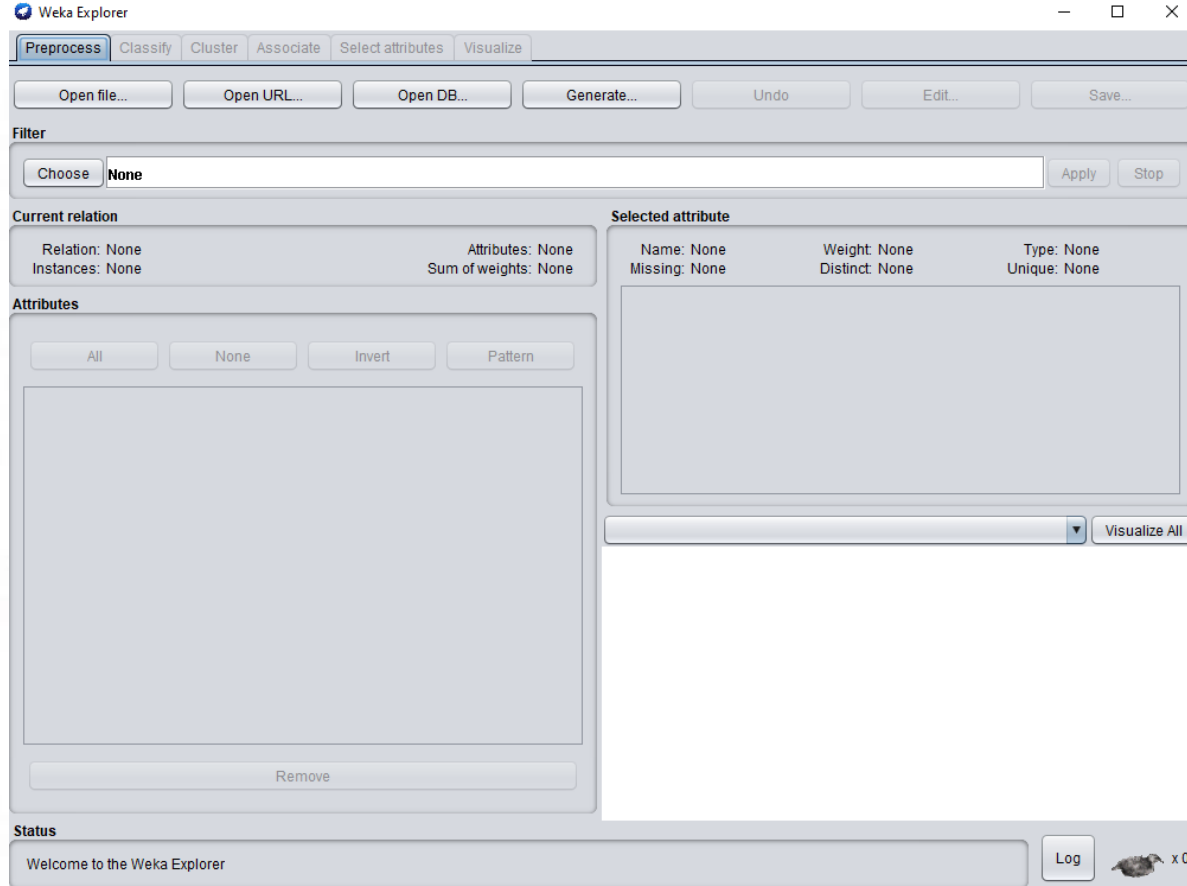
Explorer: classificadores/filtros

Experimenter: comparação de performances de algoritmos (o processamento pode levar horas, dias, semanas ou meses...)

KnowledgeFlow: Interface gráfica para o “desenho” dos métodos

Simple CLI: Interface em linha de comando

Weka



Em “Explorer” estão os algoritmos de pré-processamento, classificação, clusterização, associação, seleção de atributos e visualização de dados

Formato de arquivo no Weka (.arff)

@relation jogar_tenis

@attribute aspecto {chuva, nuvens, sol}

@attribute temperatura {ameno, fresco, quente}

@attribute humidade {elevada, normal}

@attribute vento {forte, fraco}

@attribute jogar {nao, sim}

@data

sol	,	quente	,	elevada	,	fraco	,	nao
sol	,	quente	,	elevada	,	forte	,	nao
nuvens	,	quente	,	elevada	,	fraco	,	sim
chuva	,	ameno	,	elevada	,	fraco	,	sim
chuva	,	fresco	,	normal	,	fraco	,	sim
chuva	,	fresco	,	normal	,	forte	,	nao
nuvens	,	fresco	,	normal	,	fraco	,	sim
sol	,	ameno	,	elevada	,	fraco	,	nao



MATLAB¹

O MATLAB (MATrix LABoratory) é um software interativo de alta performance voltado para o cálculo numérico. Ele integra análise numérica, cálculo com matrizes, processamento de sinais e construção de gráficos em ambiente fácil utilização, ao contrário da programação tradicional.

MATLAB pode ser usado como um shell interativo, ou por meio de scripts na linguagem MATLAB (M-código).

O software dispõe de diversas extensões (toolboxes), muitas delas voltadas para IA, aprendizagem de máquina, processamento de sinais, além de muitas outras. Há ainda o Simulink, um ambiente de simulação baseado em diagrama de blocos e plataforma para Model-Based Design.

1. <https://www.cs.waikato.ac.nz/ml/weka/>

MATLAB

MATLAB R2016b

HOME PLOTS APPS

Search Documentation

FILE VARIABLE CODE SIMULINK ENVIRONMENT RESOURCES

C: > Program Files > MATLAB > R2016b > bin >

Current Folder

- Name ^
- m3iregistry
- registry
- util
- win64
- deploytool.bat
- lcdata.xml
- lcdata.xsd
- lcdata_utf8.xml
- matlab.exe
- mbuild.bat
- mcc.bat
- mex.bat
- mex.pl
- mexext.bat
- mexsetup.pm

Details

Workspace

Name ^	Value
--------	-------

Command Window

```
fx >> |
```

MATLAB – toolbox Otimização (AG)

The screenshot displays the MATLAB R2016b environment with the Optimization Tool window open. The tool is configured for the Genetic Algorithm Solver. The interface is divided into three main sections: Problem Setup and Results, Options, and Quick Reference.

Problem Setup and Results:

- Solver: **ga - Genetic Algorithm**
- Problem: Fitness function and Number of variables fields are empty.
- Constraints: Linear inequalities (A, b), Linear equalities (Aeq, beq), Bounds (Lower, Upper), Nonlinear constraint function, and Integer variable indices fields are empty.
- Run solver and view results: Use random states from previous run. Buttons: Start, Pause, Stop.
- Current iteration: [] Clear Results

Options:

- Population: Population type: Double vector; Population size: Use default: 50 for five or fewer variables, otherwise 200; Specify: []; Creation function: Constraint dependent.
- Initial population: Use default: []; Specify: []
- Initial scores: Use default: []; Specify: []
- Initial range: Use default: [-10;10]; Specify: []
- Fitness scaling: Scaling function: Rank
- Selection: Selection function: Stochastic uniform

Quick Reference:

- Genetic Algorithm Solver**
- This tool corresponds to the ga function.
- Click to expand the section below corresponding to your task.
- Problem Setup and Results**
- ▶ Problem
- ▶ Constraints
- ▶ Run solver and view results
- Options**
- Specify options for the Genetic Algorithm solver.
- ▶ Population
- ▶ Fitness scaling
- ▶ Selection
- ▶ Reproduction
- ▶ Mutation
- ▶ Crossover

MATLAB – toolbox Fuzzy

MATLAB R2016b

HOME PLOTS APPS

New Script New Open Compare Import Data Save Workspace New Variable Open Variable Clear Workspace Analyze Code Run and Time Clear Commands Simulink Layout Preferences Set Path Parallel Add-Ons Help Community Request Support Learn MATLAB

FILE VARIABLE CODE SIMULINK ENVIRONMENT RESOURCES

C:\Program Files\MATLAB\R2016b\bin

Current Folder

- Name ^
- m3registry
- registry
- util
- win64
- deploytool.bat
- lcdata.xml
- lcdata.xsd
- lcdata_utf8.xml
- matlab.exe
- mbuild.bat
- mcc.bat
- mex.bat
- mex.pl
- mexext.bat
- mexsetup.dm

Details

Workspace

Name ^	Value
--------	-------

Command Window

```
>> fuzzy  
fx >>
```

Fuzzy Logic Designer: Untitled

File Edit View

input1 output1

FIS Name: Untitled FIS Type: mamdani

And method	min	Current Variable	
Or method	max	Name	input1
Implication	min	Type	input
Aggregation	max	Range	[0 1]
Defuzzification	centroid		

Help Close

System "Untitled": 1 input, 1 output, and 0 rules

MATLAB – Árvore de Decisão

MATLAB R2016b

The image displays the MATLAB R2016b environment with a decision tree viewer window open. The viewer shows a tree structure for classifying Iris species based on features x_3 and x_4 . The tree starts with a root node $x_3 < 2.45$. If true, it leads to the 'setosa' class. If false, it leads to a node $x_4 < 1.75$. If true, it leads to the 'virginica' class. If false, it leads to a node $x_3 < 4.95$. If true, it leads to a node $x_4 < 1.65$, which then branches into 'versicolor' (if true) and 'virginica' (if false). If false, it leads to the 'virginica' class.

```
>> load fisheriris % load data
ctree = fitctree(meas, species) % fit decision tree
view(ctree) % text description of tree

Decision tree for class: species
1 if  $x_3 < 2.45$  then node 1
2 class = setosa
3 if  $x_4 < 1.75$  then node 3
4 if  $x_3 < 4.95$  then node 4
5 class = virginica
6 if  $x_4 < 1.65$  then node 6
7 class = virginica
8 class = versicolor
9 class = virginica

>> view(ctree, 'mode', 'text')
fx >>
```

Name	Value
ctree	1x1 ClassificationTree
meas	150x4 double
species	150x1 cell