



# Inteligência Artificial Aplicada



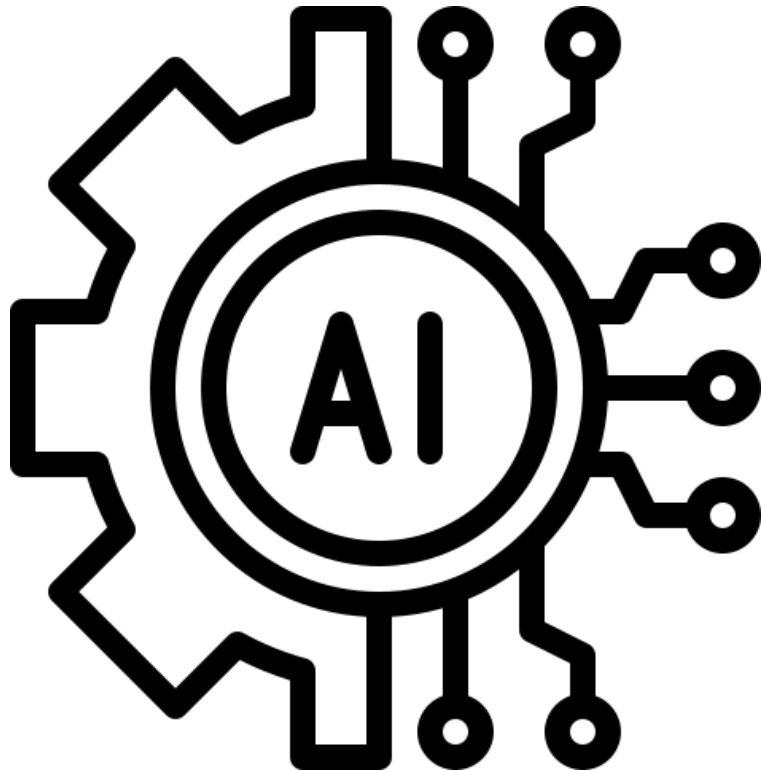


# AGENDA

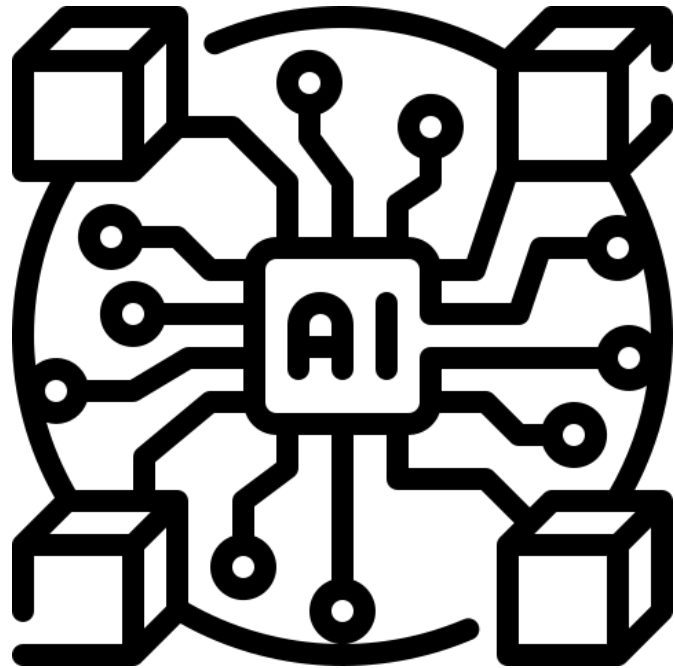
- Apresentação;
- Informações sobre a disciplina:
  - Plano de ensino;
  - Aspectos metodológicos (aulas, avaliações, ferramentas etc.);
- Introdução a conceitos de inteligência artificial.

# Plano de aula/ensino

# O que é Inteligência Artificial?



- ❖ Estudo de agentes que recebem percepções do ambiente e executam ações (RUSSELL; NORVIG, 2013).



- ❖ Processamento de linguagem natural;
- ❖ Representação de conhecimento;
- ❖ Raciocínio automatizado;
- ❖ Aprendizado de máquina;
- ❖ Visão computacional;
- ❖ Robótica.

(RUSSELL; NORVIG, 2013).



# Disciplinas que contribuíram para a Inteligência Artificial



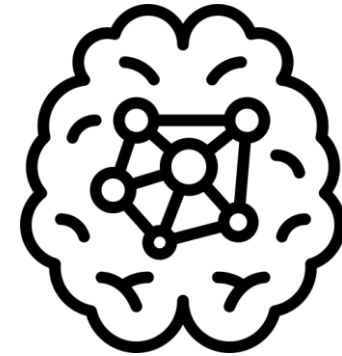
Filosofia



Matemática



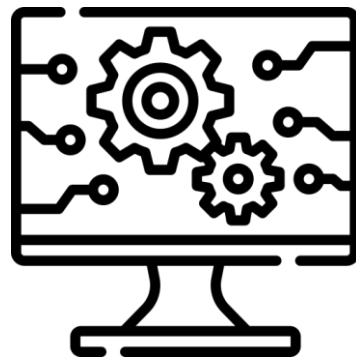
Economia



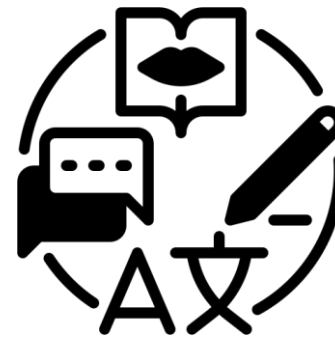
Neurociências



Psicologia






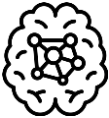
Ciência da Computação



Linguística

(RUSSELL; NORVIG, 2013).






Disciplina	Perguntas
	De onde vem o conhecimento? Como o conhecimento conduz à ação?
	O que pode ser computado? Como raciocinamos com informações incertas?
	Como devemos tomar decisões para maximizar a recompensa? Como devemos fazer isso quando a recompensa pode estar distante no futuro?
	Como o cérebro processa informações?

(RUSSELL; NORVIG, 2013).

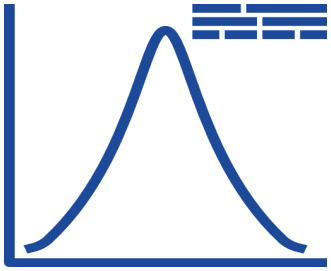




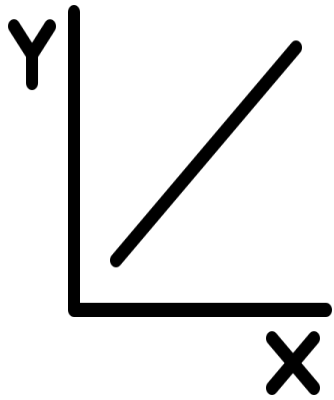
Disciplina	Perguntas
	Como os seres humanos e os animais pensam e agem?
	Como podemos construir um computador eficiente?
	Como a linguagem se relaciona com o pensamento?

(RUSSELL; NORVIG, 2013).

# Fundamentos necessários



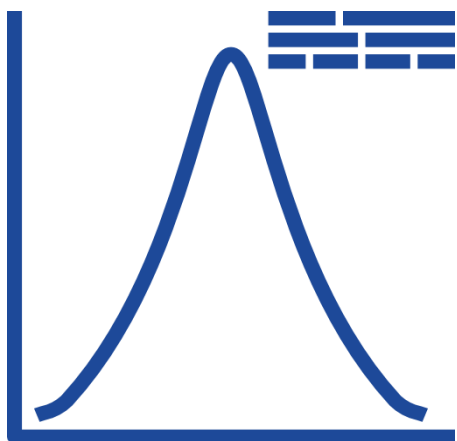
Estatística: **entender e modelar dados.**



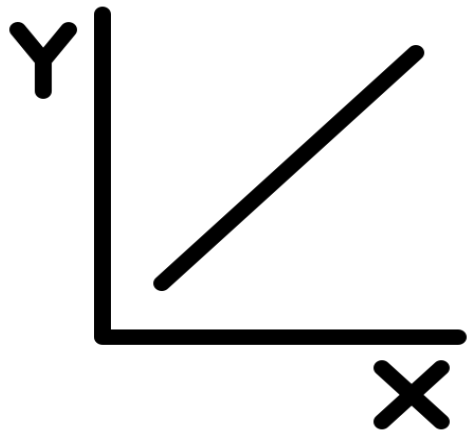
Matemática: **fundamentar e otimizar algoritmos.**



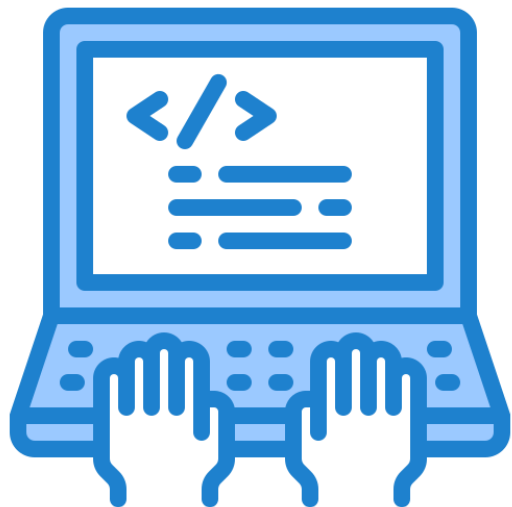
Programação: **implementar e escalar soluções.**



- Bases teóricas e métodos para análise e interpretação de dados;
- Entender incertezas e variabilidades nos dados;
- Contribui para inferências, testes de hipóteses e estimação de parâmetros.

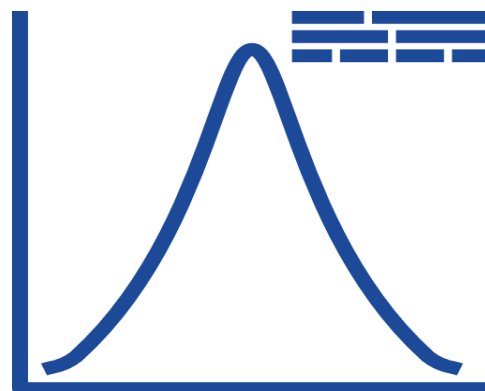


- Álgebra linear e cálculo são fundamentais para redes neurais e otimização.
- Teoria dos grafos e otimização permitem resolver problemas complexos.
- Garante a eficiência e precisão dos modelos

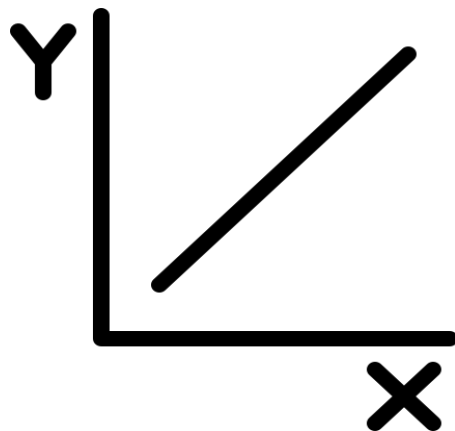


- Linguagens como Python e tornam a IA acessível.
- Estruturas de dados e algoritmos permitem manipular e processar dados em larga escala.
- Facilita a criação, treinamento e implantação de modelos.

Tipo de dados	Análise de dados	Outros
<ul style="list-style-type: none"> <li>• Numéricos</li> <li>• Categóricos</li> </ul>	<ul style="list-style-type: none"> <li>• Média,</li> <li>• Mediana,</li> <li>• Moda</li> <li>• Desvio padrão</li> <li>• Amplitude</li> </ul>	<ul style="list-style-type: none"> <li>• Probabilidade</li> <li>• Correlação</li> <li>• Testes de hipóteses</li> </ul>



Álgebra Linear	Otimização	Teoria dos grafos
<ul style="list-style-type: none"> <li>• Vetores,</li> <li>• Matrizes,</li> <li>• Autovalores.</li> </ul>	<ul style="list-style-type: none"> <li>• Funções convexas,</li> <li>• Métodos de gradiente.</li> </ul>	<ul style="list-style-type: none"> <li>• Grafos,</li> <li>• Caminhos,</li> <li>• Conectividade</li> </ul>





Conceitos básicos	Estrutura de dados	Algoritmos
<ul style="list-style-type: none"> <li>• Variáveis,</li> <li>• Condicionais,</li> <li>• Loops,</li> <li>• Funções</li> </ul>	<ul style="list-style-type: none"> <li>• Listas,</li> <li>• Dicionários,</li> <li>• Pilhas,</li> <li>• Filas,</li> <li>• <i>DataFrames</i></li> </ul>	<ul style="list-style-type: none"> <li>• Busca,</li> <li>• Ordenação,</li> <li>• Recursão.</li> </ul>





# Nivelamento



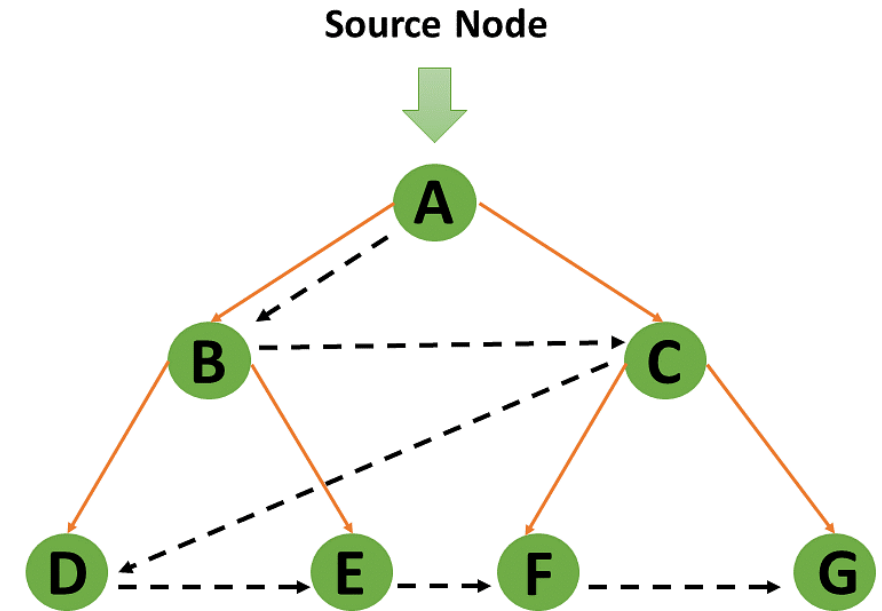


# Noções básicas dos algoritmos



Categoria	Descrição	Algoritmos/técnicas	Exemplos de aplicação
Métodos de busca	Encontram soluções em um espaço de estados.	BFS, A*	GPS traçando a melhor rota.
Raciocínio temporal	Modelam eventos ao longo do tempo.	Cadeias de Markov, Redes Bayesianas	Previsão do tempo, reconhecimento de fala.
Lógica fuzzy	Lida com incertezas e valores intermediários.	Conjuntos fuzzy, Inferência fuzzy	Controle de temperatura em ar-condicionado.
Representação do conhecimento	Estrutura e organiza informações para tomada de decisão.	C4.5, Árvores de decisão	Diagnóstico médico, sistemas especialistas.
Modelos de aprendizado	Ajustam pesos para identificar padrões em dados.	Redes Neurais (Perceptron, MLP)	Reconhecimento facial, chatbots.

## Busca em largura (BFS – *Breadth-First Search*)



- **Ideia principal:** explora todos os nós em um nível antes de avançar para o próximo.
- **Aplicações:** navegação em mapas, resolução de quebra-cabeças.

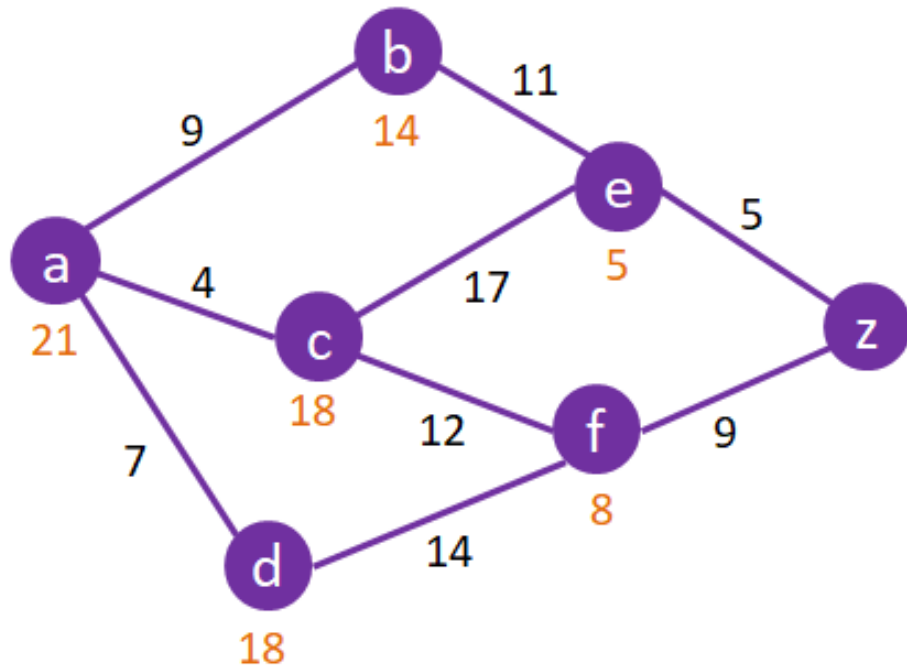
## **BFS – encontrar um amigo**

**Situação:** você está em uma festa e quer encontrar seu amigo. Você decide perguntar primeiro para todas as pessoas mais próximas antes de avançar para outras mais distantes.

### **Passos:**

1. Pergunte para todas as pessoas ao seu redor.
2. Se nenhuma delas for seu amigo, pergunte para os amigos delas.
3. Continue até encontrar seu amigo.

## Busca A\* (A – estrela)



- **Ideia principal:** Algoritmo heurístico que prioriza caminhos mais promissores.
- **Fórmula:**  $f(n) = g(n) + h(n)$

# A\*– caminho no Google Maps

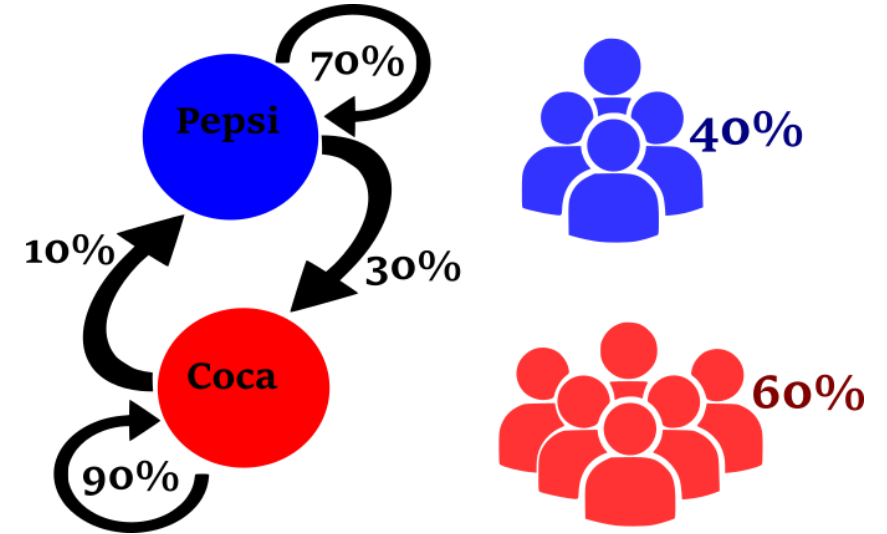
**Situação:** você está indo de casa para a universidade e quer o caminho mais rápido.

## Passos:

1. Você pode pegar ruas menores (menos trânsito) ou avenidas (mais rápidas, mas podem estar congestionadas).
2. O Google Maps calcula custo real (distância já percorrida) e custo estimado (quanto falta).
3. O melhor caminho é aquele com menor custo total.



# Cadeias de Markov



- **Ideia principal:** Processo estocástico onde o próximo estado depende apenas do estado atual.
- **Propriedade de Markov:**  $P(X_{t+1} | X_t, X_{t-1}, \dots, X_0) = P(X_{t+1} | X_t)$

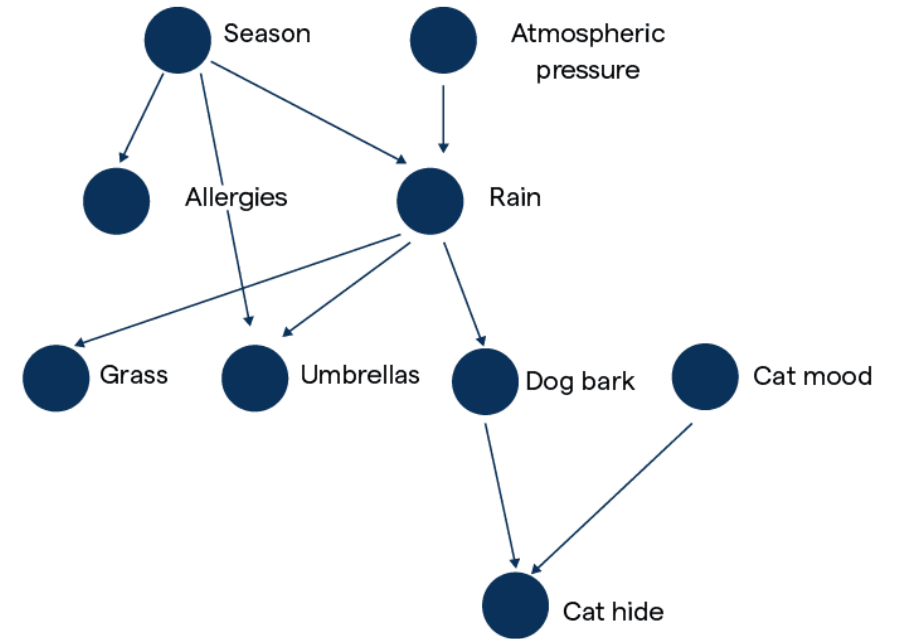
# cadeias de Markov – previsão do tempo

**Situação:** se hoje está chovendo, qual a chance de amanhã também chover?

## **Passos:**

1. Se os últimos dias foram chuvosos, há alta chance de continuar assim.
2. Se fez sol nos últimos dias, a chance de sol amanhã aumenta.
3. O futuro depende apenas do estado atual e não do histórico completo.

# Redes Bayesianas



- **Ideia principal:** Grafos direcionados acíclicos onde os nós representam variáveis aleatórias e as arestas indicam dependência probabilística.
- **Teorema de Bayes:** 
$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

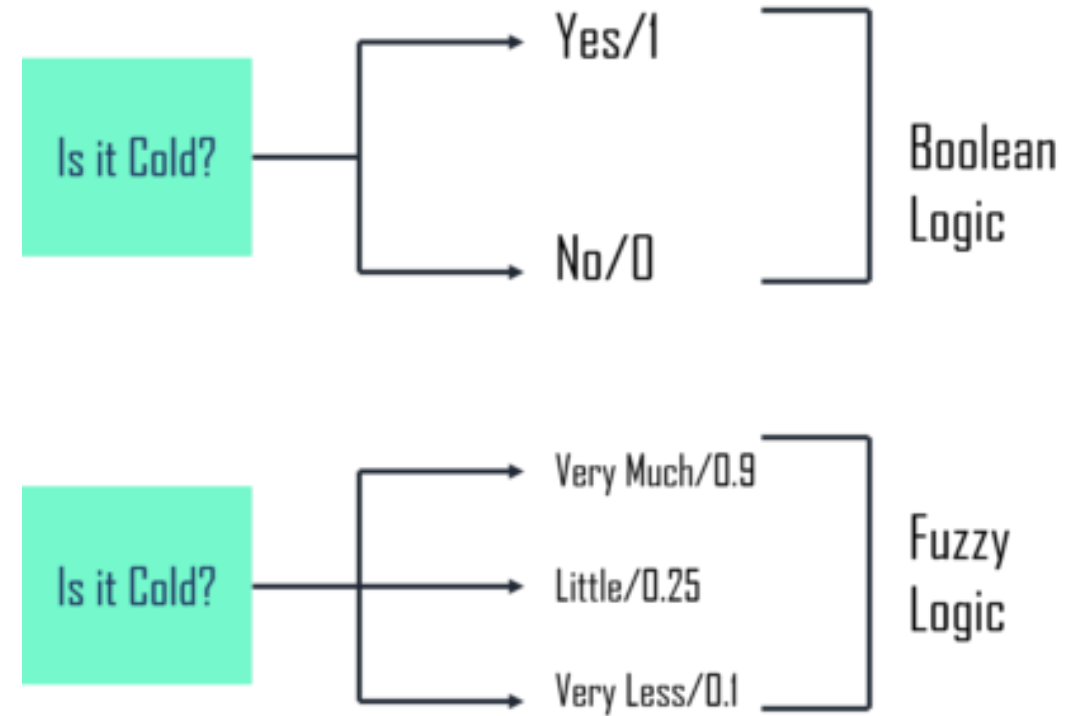
# rede Bayesiana – diagnóstico médico

**Situação:** um médico avalia sintomas para determinar a probabilidade de uma gripe.

## **Passos:**

1. Se um paciente tem febre e dor de cabeça, ele **pode** estar gripado.
2. Se ele também tem dor no corpo, a probabilidade aumenta.
3. Se ele tomou medicamento recente, a chance diminui.

# Lógica fuzzy



- **Ideia principal:** Lógica que permite valores intermediários entre verdadeiro (1) e falso (0).
- **Funcionamento:** trabalha com regras e graus de pertinência.

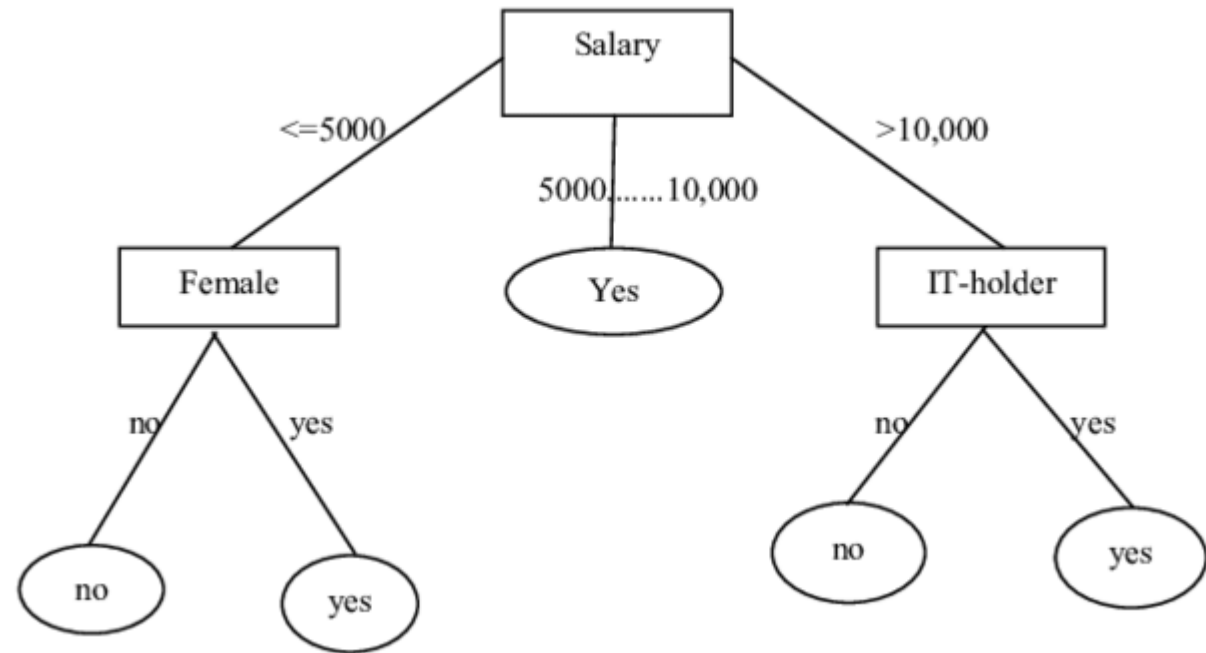
# Lógica fuzzy – controle de um ventilador

**Situação:** Você quer ajustar a velocidade do ventilador, mas não quer apenas "ligado" ou "desligado".

## Regras:

1. Se está muito quente, o ventilador gira rápido.
2. Se está morno, o ventilador gira médio.
3. Se está frio, o ventilador gira devagar.

## C4.5



- **Ideia principal:** algoritmo que cria árvores de decisão para classificação.
- **Critério:** entropia e ganho de informação para dividir os nós.

## C4.5 – detectando spam

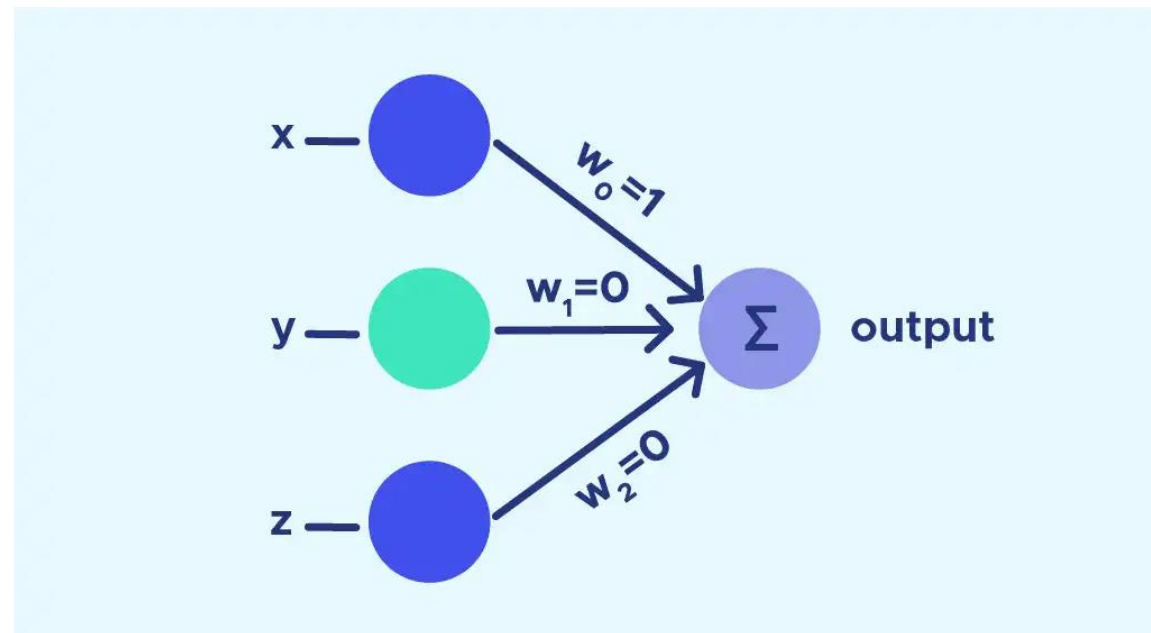
**Situação:** seu e-mail precisa decidir se uma mensagem é **spam** ou **não spam**.

### **Passos:**

1. Se o e-mail contém a palavra “grátis”, pode ser spam.
2. Se também tem “promoção”, a chance de ser spam aumenta.
3. Se o remetente for confiável, pode não ser spam.



# Perceptron



- **Ideia principal:** modelo matemático inspirado em neurônios biológicos.
- **Fórmula:**  $y = f(\sum_{i=1}^n w_i * x_i + b)$

# Perceptron – classificação de figuras

**Situação:** treinar uma IA para identificar gatos e cachorros.

## **Passos:**

- 1.o algoritmo Perceptron recebe características como "tem bigode?" e "tem orelhas pontudas?".
- 2.Ele decide se a imagem é de um gato ou cachorro.
- 3.Se errar, ajusta os pesos e tenta de novo.



# Ferramentas





 pandas



TensorFlow

kaggle



seaborn



UC Irvine  
Machine Learning  
Repository

 plotly



statsmodels



## Referências Slides:

GÉRON, A. **Mãos à obra**: aprendizado de máquina com Scikit-Learn & TensorFlow. Rio de Janeiro: Alta Books, 2019.

RUSSELL, S.; NORVIG, P. **Inteligência artificial**. Rio de Janeiro: Elsevier Editora Ltda, 2013.

## Recomendações:

FIELD, A.; MILES, J.; FIELD, Z. **Discovering statistics using R**. Londres: SAGE, 2012.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep learning**. Cambridge: MIT Press, 2016.

KNAFLIC, C. N. **Storytelling com dados**: um guia sobre visualização. Rio de Janeiro: Alta Books, 2019.



# “Pensando” como uma IA



**Objetivo:** classificar os dados com base em variáveis como **experiência** e *commits* por mês.

ID	Experiência (anos)	Commits (mês)	Desempenho
1	2	10	Ruim
2	5	30	Bom
3	3	15	Bom
4	1	5	Ruim
5	4	20	Bom
6	2	8	Ruim

ID	Experiência (anos)	Commits (mês)	Desempenho
7	3	12	
8	1	7	
9	4	18	
10	2	6	



ID	Experiência (anos)	Commits (mês)	Desempenho
7	3	12	Bom
8	1	7	Ruim
9	4	18	Bom
10	2	6	Ruim

**Objetivo:** aplicar lógica fuzzy para classificar a prioridade de tarefas no desenvolvimento de software com base em características como complexidade, urgência e importância.

As variáveis são:

1. **Complexidade:** complexidade da tarefa, que pode ser *Baixa*, *Média* ou *Alta*.
2. **Urgência:** urgência da tarefa, que pode ser *Baixa*, *Média* ou *Alta*.
3. **Importância:** importância da tarefa, que também pode ser *Baixa*, *Média* ou *Alta*.

Com base nas variáveis, a tarefa será atribuída uma prioridade: **Baixa**, **Média** ou **Alta**.

ID	Complexidade	Urgência	Importância	Prioridade
1	Baixa	Baixa	Baixa	<b>Baixa</b>
2	Média	Alta	Alta	<b>Alta</b>
3	Alta	Média	Baixa	<b>Média</b>
4	Média	Alta	Média	<b>Alta</b>
5	Alta	Alta	Alta	<b>Alta</b>
6	Baixa	Baixa	Alta	<b>Média</b>

## Regras prioridade

1. Se **complexidade** é *Baixa* e **urgência** é *Baixa* e **importância** é *Baixa*, a **prioridade** é *Baixa*.
2. Se **complexidade** é *Alta* e **urgência** é *Alta* e **importância** é *Alta*, a **prioridade** é *Alta*.
3. Se **complexidade** é *Média* e **urgência** é *Alta* e **importância** é *Média*, a **prioridade** é *Alta*.
4. Se **complexidade** é *Alta* e **urgência** é *Média* e **importância** é *Baixa*, a **prioridade** é *Média*.

ID	Complexidade	Urgência	Importância	Prioridade
7	Alta	Baixa	Média	
8	Média	Alta	Alta	
9	Baixa	Média	Alta	
10	Alta	Alta	Baixa	

ID	Complexidade	Urgência	Importância	Prioridade
7	Alta	Baixa	Média	<b>Média</b>
8	Média	Alta	Alta	<b>Alta</b>
9	Baixa	Média	Alta	<b>Média</b>
10	Alta	Alta	Baixa	<b>Alta</b>