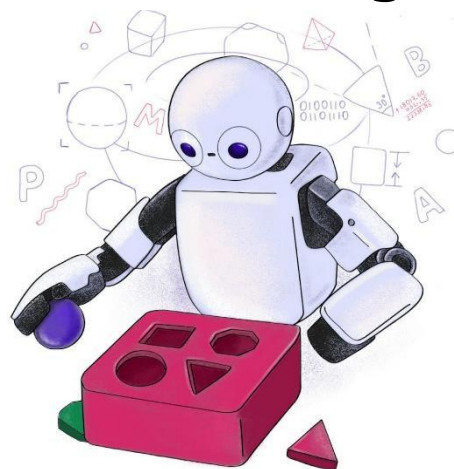


# TP555 - Inteligência Artificial e Machine Learning: *Introdução*



Esse material foi desenvolvido e gentilmente cedido pelo Prof. Dr. Felipe Augusto Pereira de Figueiredo, do Inatel. ([felipe.figueiredo@inatel.br](mailto:felipe.figueiredo@inatel.br))

***Inatel***

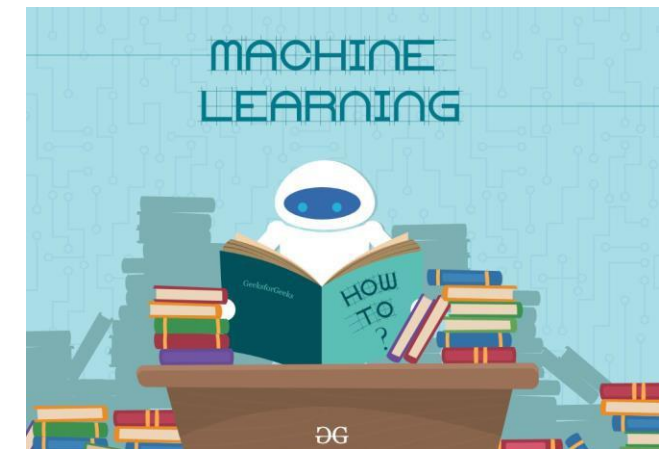
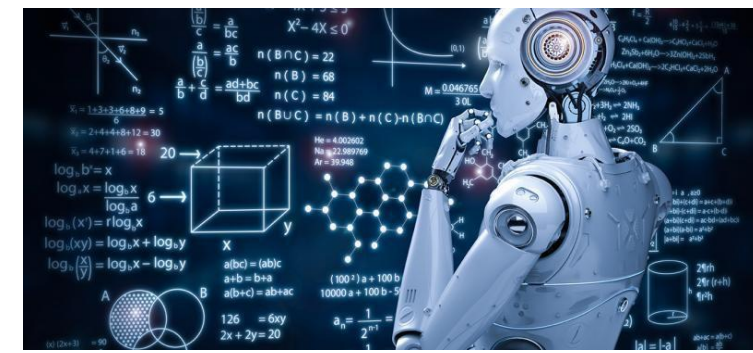
Prof. Dr. Luiz Augusto Melo Pereira  
[luiz.melo@inatel.br](mailto:luiz.melo@inatel.br)

# A disciplina

- Curso introdutório onde veremos os ***conceitos básicos*** de funcionamento de vários ***algoritmos de aprendizado de máquina*** ou do inglês, ***machine learning*** (ML).
- Não nos aprofundaremos nos conceitos matemáticos envolvidos, mas focaremos no uso e aplicação prática dos algoritmos.
- Porém, é necessário que vocês conheçam Python (básico/intermediário), conceitos de telecomunicações, álgebra linear, cálculo e estatística.
- Todo material do curso está disponibilizado no GitHub:
  - <https://github.com/luiz10ml/tp555-ml>

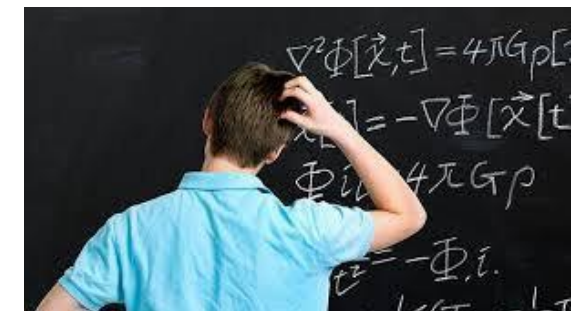
# Objetivo do curso

- O objetivo principal do curso é apresentar à vocês
  - os conceitos fundamentais da teoria por trás do ML.
  - um conjunto de ferramentas (i.e., os algoritmos) de ML.
- Ao final do curso vocês devem ser capazes de
  - Entender e discutir sobre os principais algoritmos de ML.
  - Compreender a terminologia utilizada na área.
  - Aplicar algoritmos de ML para a resolução de problemas.
  - Analisar e entender novos algoritmos de ML.
  - Criar seus próprios projetos.
  - Compreender artigos científicos que utilizam ML.
  - Realizar pesquisa que envolva ML e discutir os resultados obtidos.



# Avaliação do Curso

- 2 Provas
  - Cada uma valendo 35% da nota (datas ainda serão definidas).
  - Devem ser resolvidas individualmente durante a aula.
- 1 Projeto final
  - 20% da nota.
  - Pode ser feito em grupo de no máximo 2 alunos.
  - Tema deve ser escolhido por vocês.
  - Vejam descrição detalhada no Github.
  - Entrega da **descrição** até 06/04.
- 4 Estudos Dirigidos
  - 10% da nota.
  - Entrega pelo GitHub.
- Avaliação parcial do projeto final ocorrerá ao final do segundo mês de curso.
  - **Entregável:** relatório descrevendo a evolução do projeto e contendo resultados iniciais.

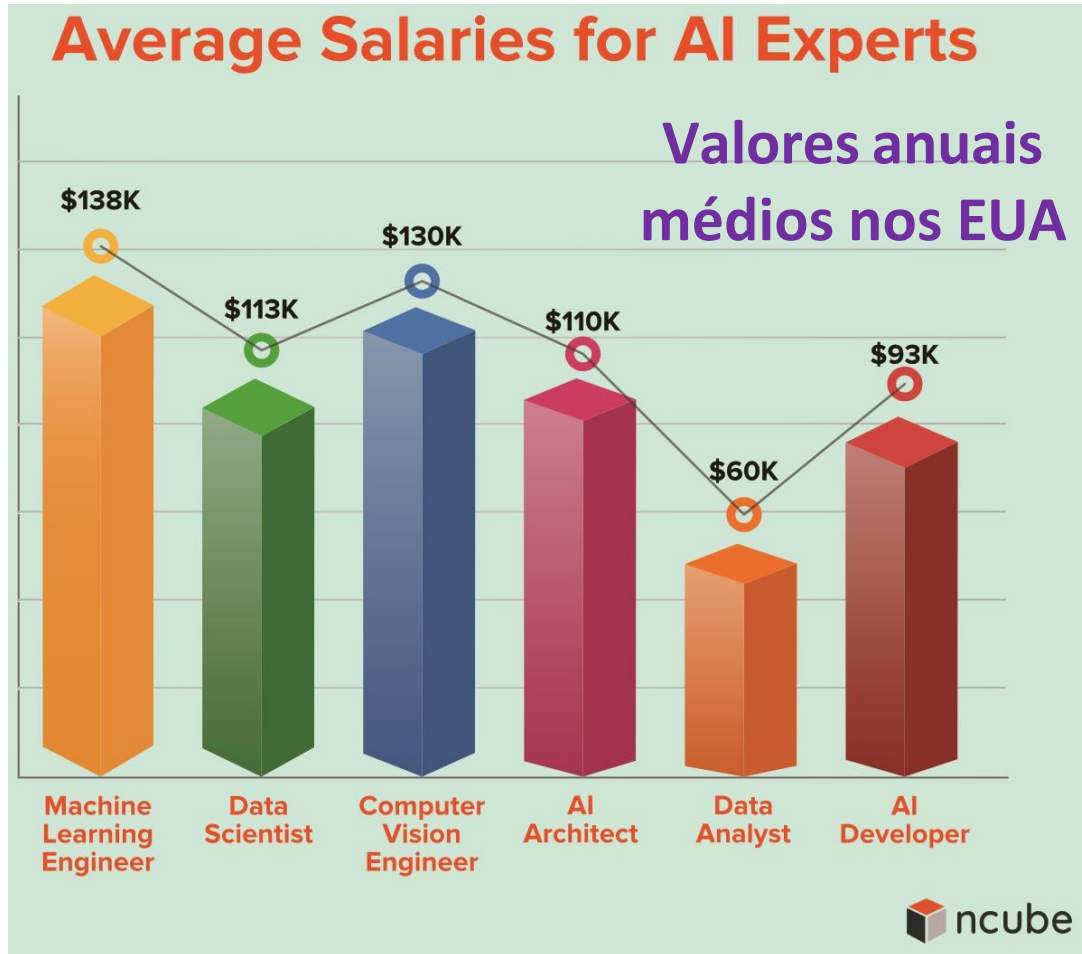


# Motivação



- **Emprego:** grandes empresas têm usado IA para
  - Automatizar tarefas repetitivas e demoradas.
  - Processar grandes volumes de dados para extração de *insights*.
  - Criar experiências personalizadas para seus clientes.
  - etc.
- Tudo isso para aumentar a eficiência dos processos, reduzir custos e, consequentemente, aumentar os lucros.
- Além das empresas de **tecnologia**, IA é empregada em empresas de **telecom**, **saúde**, **finanças**, **automação** e muitas outras.

# Motivação



- **Salário:** como as empresas têm se apoiado mais e mais em IA para aumentar seus lucros, elas têm tido uma demanda crescente por profissionais com esse conhecimento.
- Os *salários são muito bons*, pois a *demanda é alta, mas não existem tantos profissionais* com essa experiência no mercado.



# Motivação



NOKIA

Qualcomm

ZTE



- **Pesquisa:** a IA já tem sido empregada no 5G e terá um papel fundamental no desenvolvimento da próxima geração de redes móveis e sem fio, o 6G.
- Os algoritmos de IA serão usados nas *mais diversas camadas* da rede para auxiliar em tarefas que vão desde a *detecção de símbolos de uma modulação até segurança cibernética*.

# O que é inteligência artificial?



- Campo da computação que almeja *criar máquinas que imitem as capacidades humanas*.
- Que capacidades são essas?
  - Aprendizado e adaptação;
  - Comunicação;
  - Raciocínio e tomada de decisão;
  - Resolução de problemas;
  - Criatividade;
  - Movimento;
  - etc.





# O que é inteligência artificial?



- **Definição formal:** Capacidade de uma máquina de *receber estímulos* vindos do ambiente, *interpretá-los*, *aprender* com eles e usar o *conhecimento adquirido* para *tomar decisões e resolver problemas*, *interagindo* com o ambiente.



# O problema da inteligência artificial

*IA é uma área muito ampla que engloba várias aplicações (subáreas) diferentes.*



- Criar máquinas que **emulem** a inteligência humana é um **problema muito complexo e difícil de ser resolvido de uma só vez**.
- Sendo assim, **divide-se o problema em problemas menores**, chamadas de **subáreas da IA**.

# Subáreas da inteligência artificial

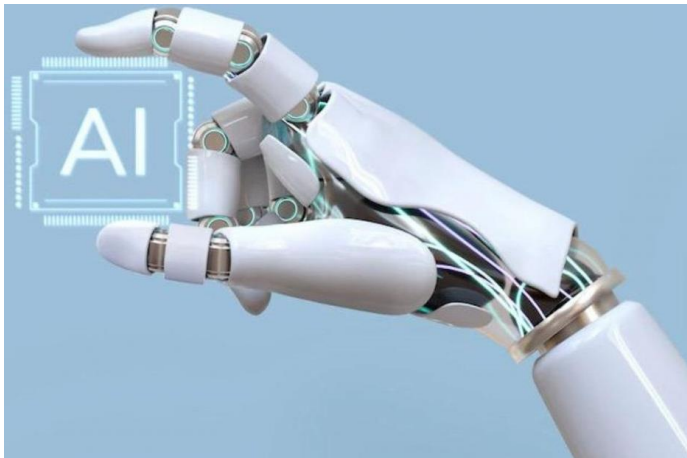


- Processamento de linguagem natural.
  - Compreensão e interpretação de linguagens humanas.
- Representação do conhecimento.
  - Extração e armazenamento eficiente de conhecimento do mundo real.
- Raciocínio automatizado.
  - Resolução de problemas complexos a partir de conhecimento prévio.
- Planejamento.
  - Criação de planos que permitam que uma máquina execute uma tarefa.

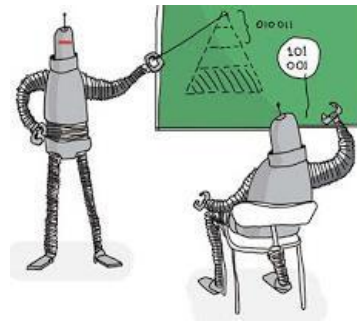
# Subáreas da inteligência artificial



- Visão computacional.
  - Compreensão e interpretação de imagens e vídeos.
- Robótica.
  - Criação de robôs capazes de realizar tarefas físicas e interagir com o ambiente.
- Aprendizado de máquina.
  - Criação de máquinas que aprendem através de experiências prévias.
- Inteligência artificial geral.
  - Criação de máquinas que se comportem como seres humanos. É a meta final da IA.



# Foco do curso



- Como vimos, IA é um área muito ampla e não teríamos tempo para ver todas as subáreas.
- Assim, focaremos no estudo dos fundamentos do **aprendizado de máquina**.
- Por quê?
  - **Caixa de ferramentas:** ML oferece **ferramentas para a análise e solução de vários problemas** em várias áreas, incluindo telecomunicações, de forma eficiente.
  - **Redução de complexidade e custo:** ML pode **reduzir o custo e a complexidade computacional** de procedimentos e processos que muitas vezes têm desempenho ótimo na teoria, mas que não são usados na prática devido a sua alta complexidade e/ou custo proibitivo.
  - **Oportunidades:** existem muitos **empregos** na área de análise, ciência e engenharia de dados, além de **pesquisas inovadoras**, que usam ML para a solução de problemas em diversas áreas.



# O que é o aprendizado de máquina?



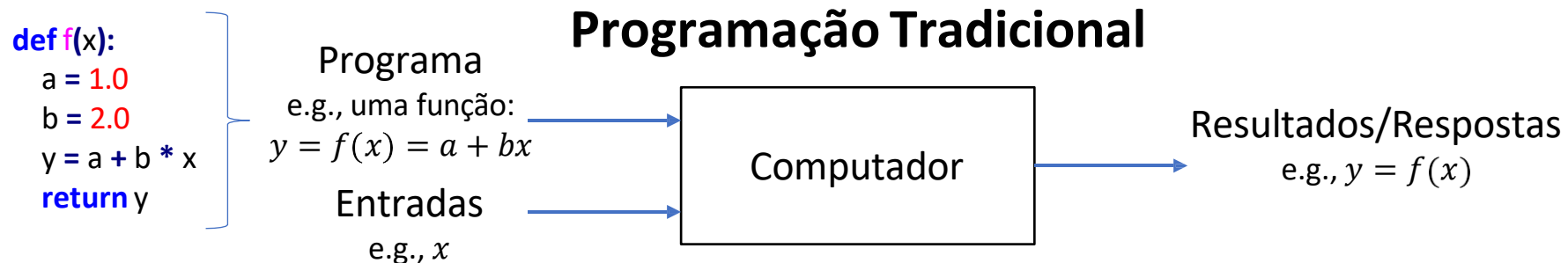
- É uma das subáreas da inteligência artificial.
- O termo foi cunhado em 1959, pelo cientista da computação Arthur Samuel, que o definiu como o  
*“Campo de estudo que dá aos computadores a habilidade de **aprender sem serem explicitamente programados**.”*
- Mas como eles aprendem?
  - Através de **experiências prévias, induz-se** conhecimento nas máquinas.
- Algoritmos de ML são **orientados a dados** pois eles **dependem fortemente de um conjunto de dados** para **aprender** uma **solução geral**.





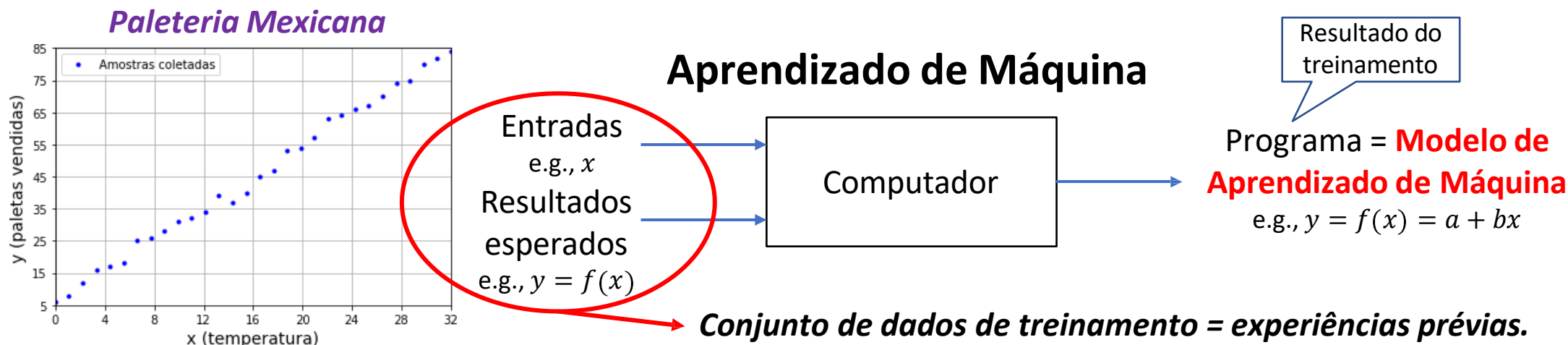
# Paradigma da programação tradicional

- Para entendermos o que é o aprendizado de máquina, vamos fazer um ***paralelo com a programação tradicional.***
- Na programação tradicional, o **programador cria as regras** (i.e., programa) **que mapeiam as entradas,  $x$ , nas saídas,  $y = f(x)$ .**
- Imaginem um problema onde a **solução geral,  $f(x)$** , é a equação de uma reta.



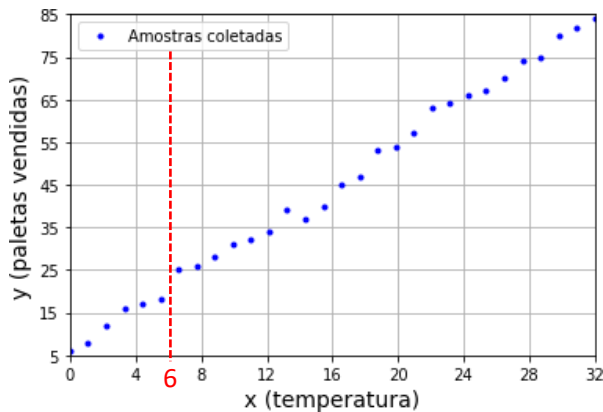
# O paradigma do aprendizado de máquina

- “... *aprender sem serem explicitamente programados*.”
- Esse trecho pode ser entendido se reorganizarmos a figura anterior.
- No ML, nós fornecemos as *entradas* e as *respostas esperadas* ao computador e deixamos que ele *aprenda*, através de *treinamento*, um *modelo* (i.e., as regras) que *mapeie* as entradas nas respostas esperadas.

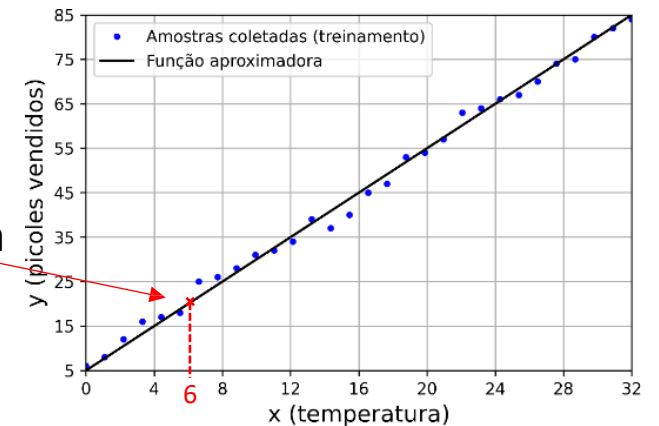


# Generalização

- Porém, não basta que o algoritmo de ML aprenda um modelo que faça um bom mapeamento ***apenas para os dados do conjunto de treinamento***.
- O **algoritmo** de ML deve **treinar** um **modelo** que **aprenda** uma **solução geral**, ou seja, que **generalize para entradas não vistas durante o treinamento**.



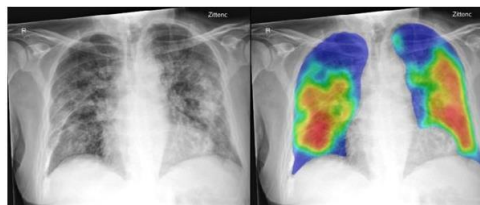
Qual é a estimativa de paletas vendidas quando a temperatura é de  $\approx 6$  graus (valor não visto durante o treinamento)?



A partir do mapeamento aprendido (i.e., **reta**), o modelo gera como saída o valor 23, que é **coerente com o restante dos dados**.

# Exemplos de aplicações de ML em várias áreas

- **Transporte:** veículos autônomos (e.g., Tesla, Waymo).
- **Negócios:** recomendação de produtos e conteúdos (e.g., Amazon e Netflix).
- **Educação:** pontuação automatizada de fala em testes de Inglês.
- **Saúde:** detecção e diagnóstico de doenças (câncer, Alzheimer, pneumonia, COVID-19, etc.).
- **Finanças:** detecção de fraudes com cartão de crédito.
- **Tecnologia:** assistentes pessoais (e.g., *Siri, Alexa, Cortana*, etc.).



# Principais motivos da difusão do ML

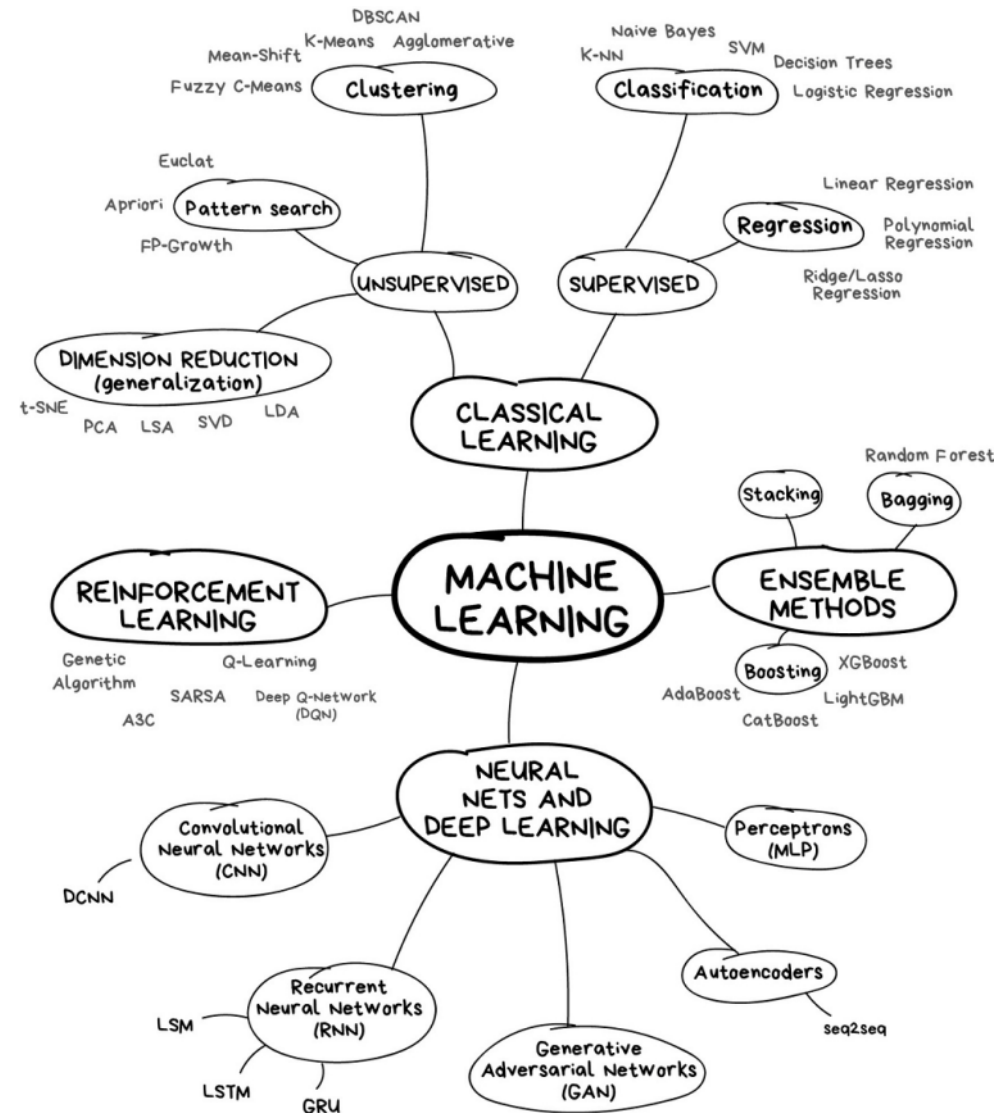
- A possibilidade de **analisar e extrair informações úteis de enormes volumes de dados** (centenas de terabytes por dia) disponíveis atualmente, o que seria impossível para nós, **vale ouro**, pois tem grande potencial para **aumentar o lucro das empresas**.
- O surgimento de **recursos computacionais poderosos** tais como GPUs, FPGAs e CPUs com múltiplos cores.
- Surgimento de **novas estratégias de aprendizagem**, e.g., *deep-learning, deep reinforcement-learning, generative adversarial learning, transformers* etc.
- Disponibilidade de **bibliotecas que facilitam o desenvolvimento** de soluções com ML.



# Paradigmas de aprendizado de máquina

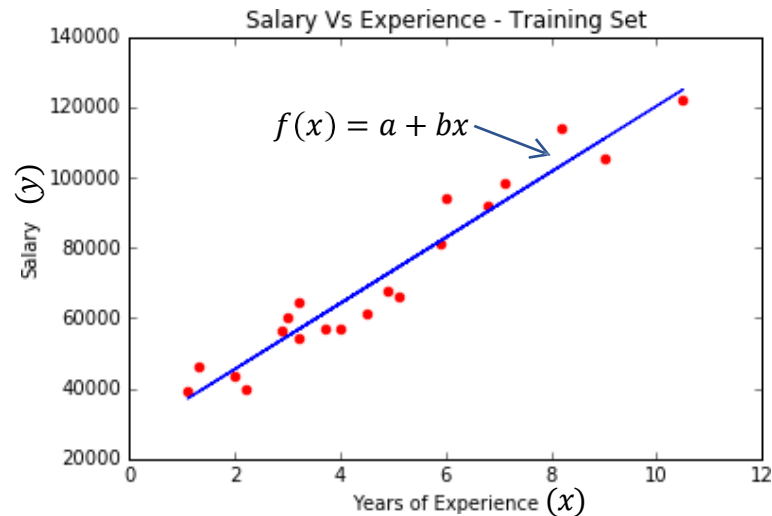
Os algoritmos de aprendizado de máquina podem ser agrupados de acordo com o **tipo de aprendizado que realizam**:

- Supervisionado
- Não-Supervisionado
- Semi-Supervisionado
- Por Reforço
- Metaheurístico





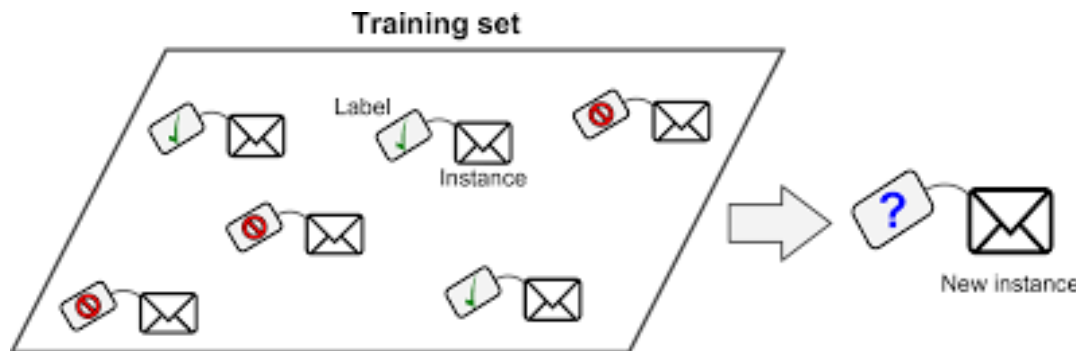
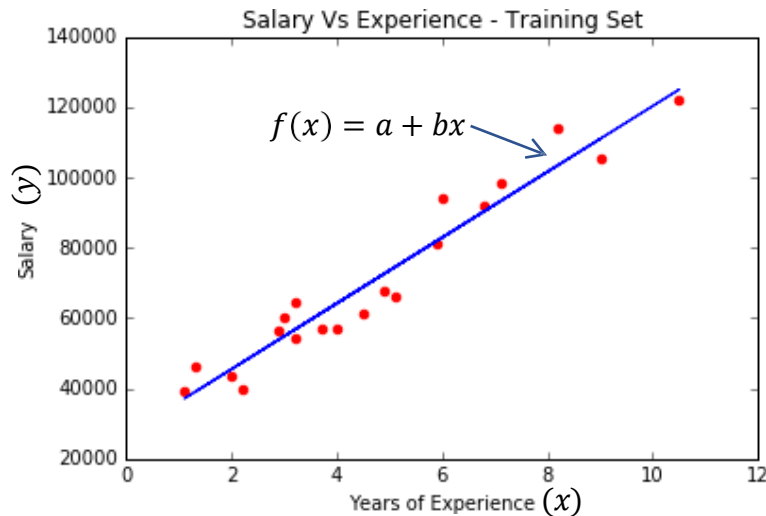
# Aprendizado supervisionado



$x = [x_1, x_2, x_3, \dots]$ : data, remetente, assunto, etc.  
 $y$ : spam ou ham

- No aprendizado supervisionado, o **algoritmo de ML tem acesso às saídas esperadas,  $y$ , chamadas de rótulos (ou labels, em inglês), para o conjunto de valores de entrada, chamados de atributos,  $x$ .**
- Em outras palavras, cada **exemplo de treinamento** é composto pelos valores de entrada,  $x$  (**atributos**), e sua saída correspondente,  $y$  (**rótulo**).
  - OBS.: Em alguns casos, têm-se mais de uma saída esperada,  $y$ .

# Aprendizado supervisionado

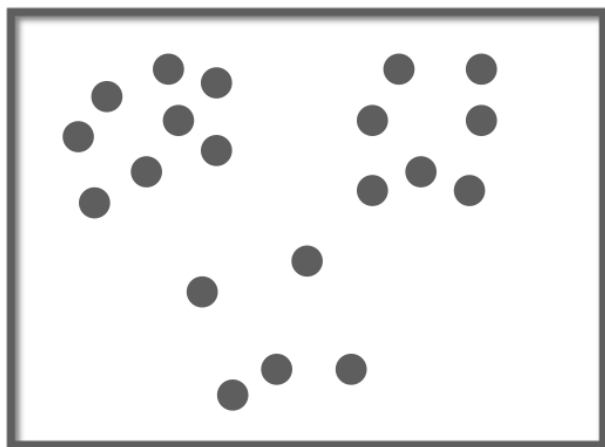


$x = [x_1, x_2, x_3, \dots]$ : data, remetente, assunto, etc.  
 $y$ : spam ou ham

- **Objetivo:** os algoritmos *supervisionados* devem *aprender* uma *função* (i.e., o modelo) que *mapeie* as entradas  $x$  nas saídas esperadas,  $y$ , ou seja,  $y = f(x)$ .
- Esse tipo de aprendizado é dividido em *problemas* de *regressão* e *classificação*.
  - **Regressão:** o rótulo,  $y$ , pertence a um *conjunto infinito* de valores, i.e., números reais.
    - **Exemplo:** anos de experiência versus salário.
  - **Classificação:** o rótulo,  $y$ , pertence a um *conjunto finito e discreto* de valores, i.e., número de possíveis classes.
    - **Exemplo:** filtro de spam.

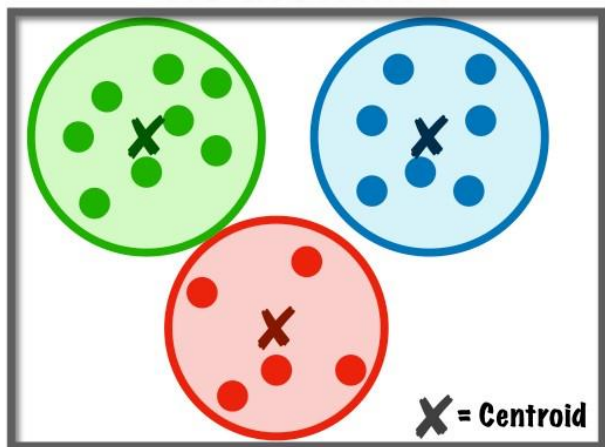
# Aprendizado não-supervisionado

Dados não rotulados



Algoritmo de aprendizado não-supervisionado

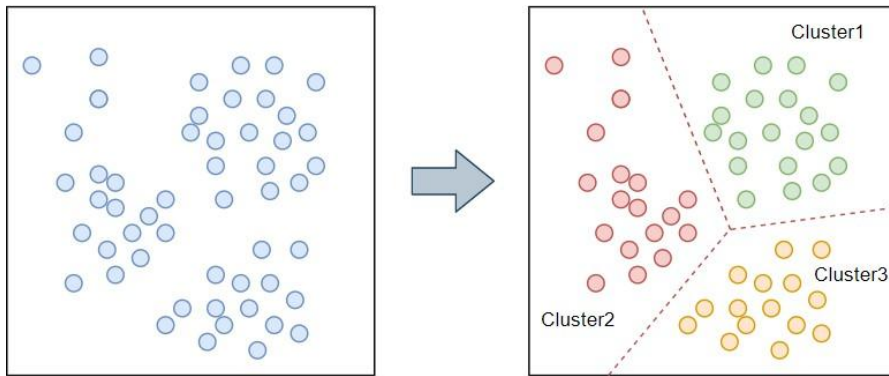
Dados agrupados por proximidade



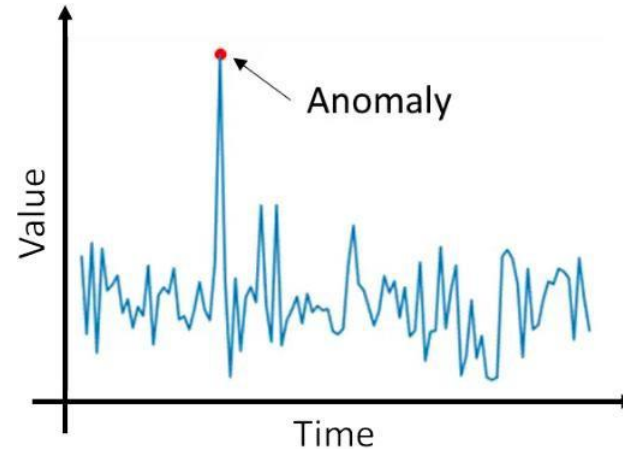
- Neste tipo de aprendizado, os algoritmos de ML **não têm acesso as saídas esperadas,  $y$ .**
- Os algoritmos só têm acesso aos **atributos,  $x$ .**
- **Objetivo:** os algoritmos devem **aprender/descobrir** padrões, muitas vezes ocultos, presentes nos dados se baseando apenas, por exemplo, na **similaridade** entre os **atributos,  $x$**  ou seja, **sem a presença de rótulos.**

# Aprendizado não-supervisionado

Clusterização



Detecção de Anomalias

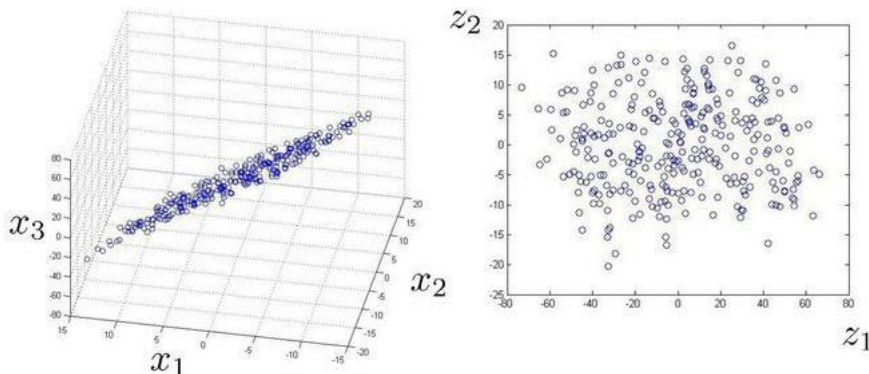


- Esses algoritmos de ML tratam problemas de

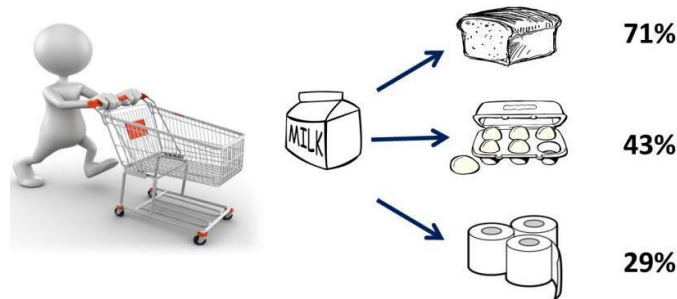
- **clusterização,**
- **detecção de anomalias (outliers),**
- **redução de dimensionalidade, e**
- **aprendizado de regras de associação.**

Redução de dimensionalidade

Reduce data from 3D to 2D



Regras de associação



Of transactions that included milk:

- 71% included bread
- 43% included eggs
- 29% included toilet paper

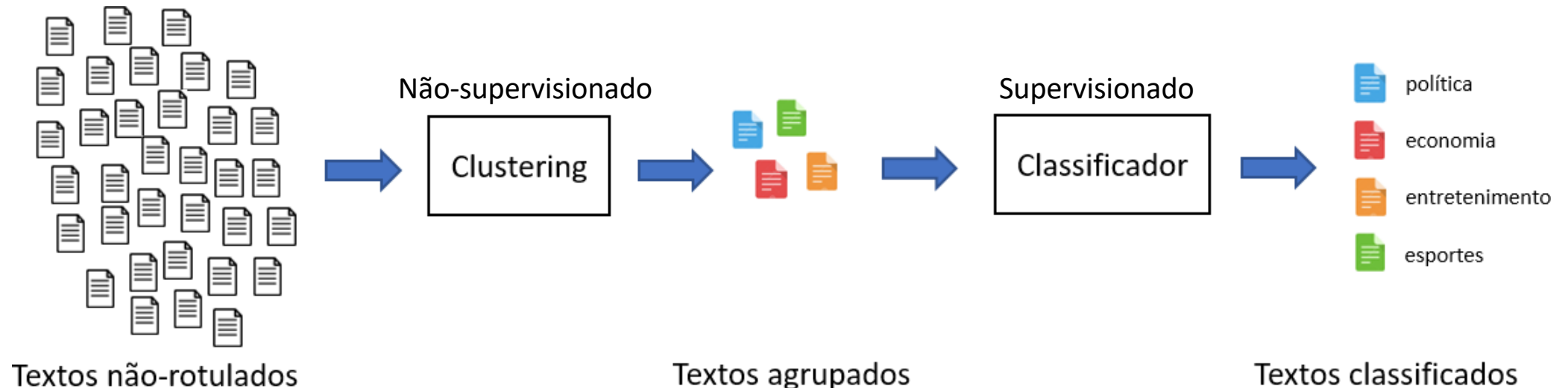
# Aprendizado semi-supervisionado



- Neste tipo de aprendizado, os algoritmos de ML têm acesso a ***exemplos de treinamento com e sem rótulos***.
- Geralmente envolve uma ***pequena quantidade de dados rotulados*** e uma ***grande quantidade de dados não-rotulados***.
- É de grande ajuda em casos onde se ter uma grande quantidade de ***dados rotulados é muito demorado, caro ou complexo***.

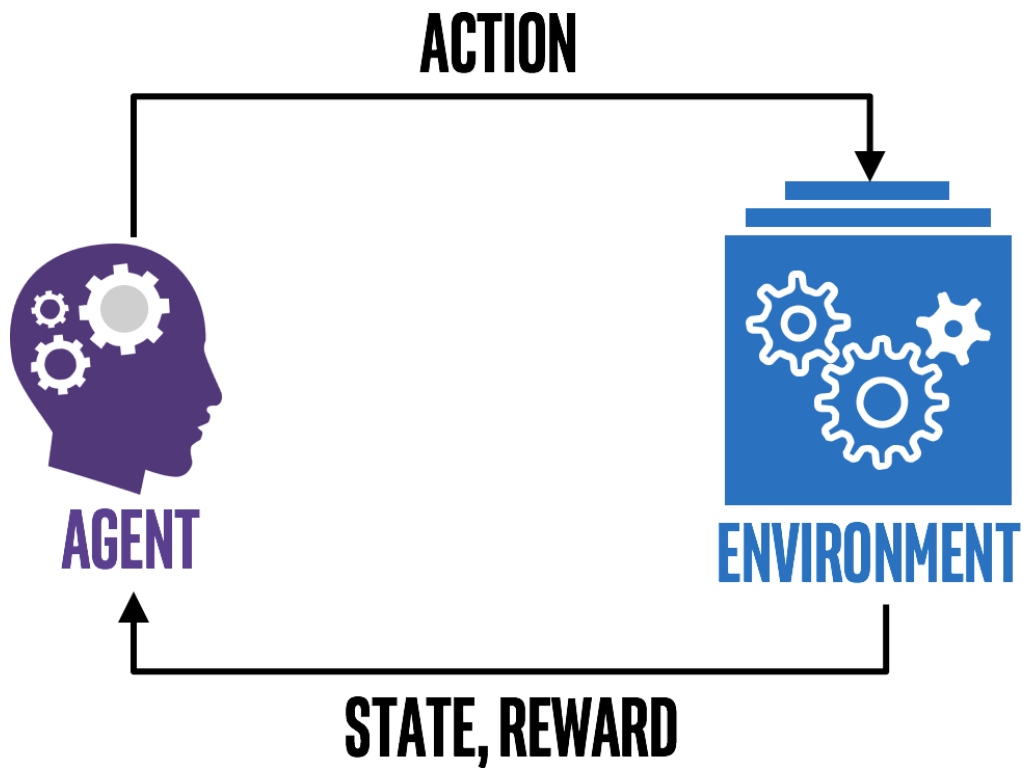
# Aprendizado semi-supervisionado

- Algoritmos de aprendizagem semi-supervisionada são o resultado da **combinação de algoritmos não-supervisionados e supervisionados**.
- Uma maneira de realizar aprendizado semi-supervisionado é combinar algoritmos de **clusterização** e **classificação**.
  - Por exemplo, como **classificaríamos** uma quantidade massiva de textos **não-rotulados** da internet nas **quatro** classes abaixo ?



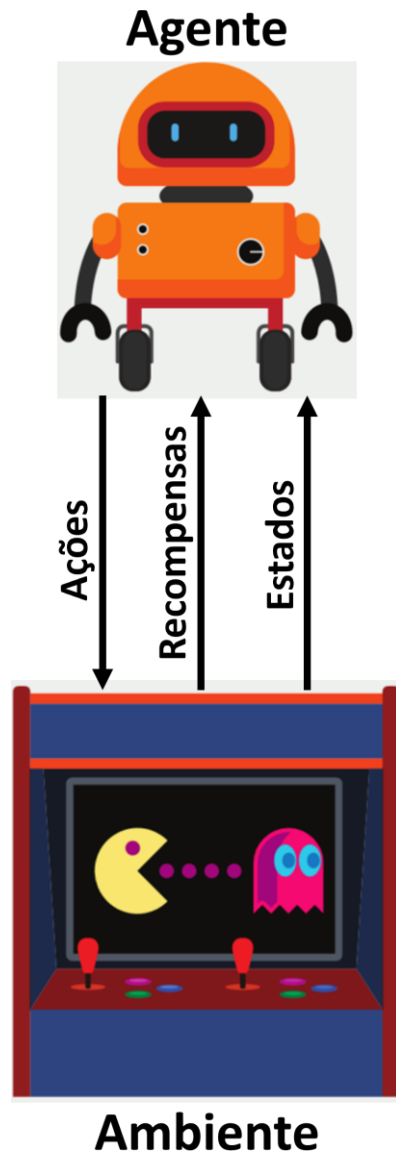


# Aprendizado por reforço



- Abordagem de aprendizado totalmente diferente das anteriores, pois *não temos exemplos de treinamento*, sejam eles rotulados ou não.
- O algoritmo de aprendizado por reforço, chamado de *agente* nesse contexto, aprende como se comportar em um *ambiente* através de interações do tipo *tentativa e erro*.
- O *agente* observa o *estado* do *ambiente*, seleciona e executa uma *ação* e recebe um *reforço + ou - (recompensa)* em consequência da *ação* tomada.

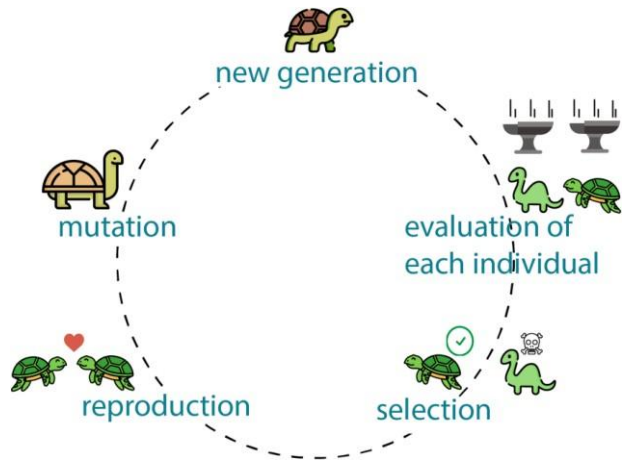
# Aprendizado por reforço



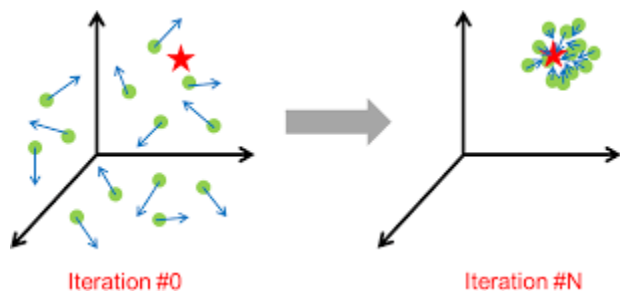
- Repetindo estes passos inúmeras vezes, o agente aprende por si só qual a melhor *estratégia*, chamada de *política*, para obter o *maior acúmulo possível de recompensas positivas ao longo do tempo*.
- Uma *política* define qual *ação* o *agente* deve escolher quando o *ambiente* estiver em um determinado *estado*.
- Portanto, a *política* é uma *função que mapeia os estados do ambiente em ações* que o *agente* deve tomar para *maximizar as recompensas*.

# Aprendizado metaheurístico

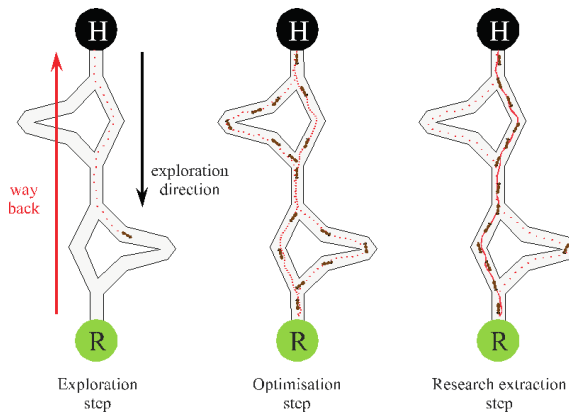
## Algoritmo genético



## Otimização por exame de partículas



## Otimização por colônia de formigas

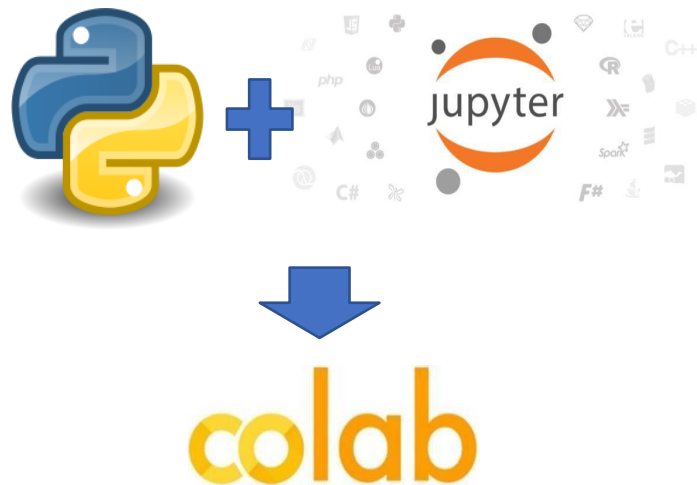


- **Metaheurísticas** são algoritmos usados para encontrar soluções de **forma rápida e genérica**, mas muitas vezes **sub-ótimas**, para **problemas complexos de otimização**.
- São aplicadas a problemas onde **métodos convencionais** são impraticáveis, devido à **alta complexidade**, ou o **espaço de busca é extremamente grande**.

# Aprendizado metaheurístico

- Características das metaheurísticas:
  - não ***garantem que uma solução ótima seja encontrada***, mas podem encontrar uma ***solução suficientemente boa*** (i.e., sub-ótima).
  - são estratégias ***que orientam o processo de busca através do espaço de soluções***.
  - não são específicas de um problema, ou seja, ***são genéricas***.
  - funcionam bem mesmo em dispositivos com ***capacidade computacional limitada*** (e.g., dispositivos IoT).
- São, em sua grande maioria, algoritmos inspirados pelo ***processo de seleção natural*** (e.g., algoritmo genético) ou no ***comportamento de grupos de animais*** (e.g., algoritmos de otimização por exame de partículas e por colônia de formigas).

# Executando códigos



- Usaremos ***Python*** como linguagem de programação.
  - Fácil de aprender, possui várias bibliotecas, é a linguagem mais utilizada em ML e é *open-source* e gratuita.
- Usaremos ***documentos virtuais*** chamados de ***notebooks Jupyter*** para desenvolvimento e documentação de código.
  - Podemos adicionar equações, gráficos e texto, além de código.
- Para executá-los, usaremos o ***Google Colaboratory***, que é um ambiente computacional gratuito hospedado na nuvem.
- Portanto, ***vocês não precisam instalar nada***, apenas terem um navegador web e conexão com a internet.

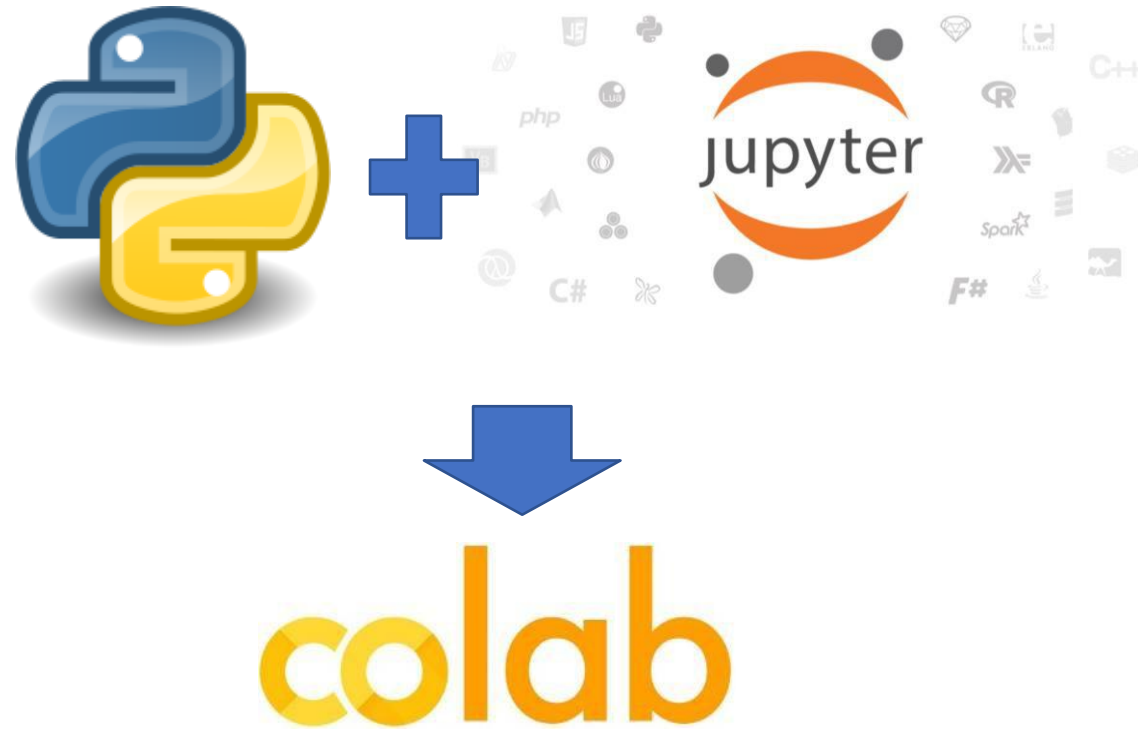
# Google Colaboratory (Colab)



- **Colab**: aplicação web gratuita que permite a criação, edição e execução de *notebooks Jupyter* em navegadores web.
- É um produto da Google.
- Vantagens:
  - Grande número de servidores.
  - Rápida inicialização e processamento do código.
  - Fornece acesso a GPUs e TPUs gratuitamente.
  - Notebooks podem ser salvos no seu Google Drive, evitando que você perca seu código.
- Desvantagem
  - Por hora, suporta apenas a execução de códigos escritos em Python.
- URL: <https://colab.research.google.com/>



# Exemplo de uso dos notebooks Jupyter



[Figura\\_2D.ipynb](#)

# Referências

- [1] Stuart Russell e Peter Norvig, “*Artificial Intelligence: A Modern Approach*,” Prentice Hall Series in Artificial Intelligence, 3rd ed., 2015.
- [2] Aurélien Géron, “*Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*”, 1st ed., O'Reilly Media, 2017.
- [3] Levy Boccato, “Notas de aula do curso Tópicos em Sistemas Inteligentes II - Aprendizado de Máquina” (IA006), disponíveis em [https://www.dca.fee.unicamp.br/~lboccato/ia006\\_2s2019.html](https://www.dca.fee.unicamp.br/~lboccato/ia006_2s2019.html) (2019).
- [4] Joseph Misiti, “*Awesome Machine-Learning*,” on-line data base with several free and/or open-source books (<https://github.com/josephmisiti/awesome-machine-learning>).
- [5] C. M. Bishop, “*Pattern Recognition and Machine Learning*,” Springer, 1st ed., 2006.
- [6] Coleção de livros, <https://tinyurl.com/mp64ksye>

# Avisos

- Vocês já podem começar a trabalhar nas listas #0 e #1.
  - <https://github.com/luiz10ml/tp555-ml/tree/main/listas>
  - A descrição do projeto final está disponível no [Github](#).
  - Se atentem ao prazo de entrega da **proposta** do projeto.
- Horário de Atendimento
  - Todas as quinta-feiras das 15:30 às 17:30.
  - Presencialmente (prédio 3, segundo andar).

Perguntas?

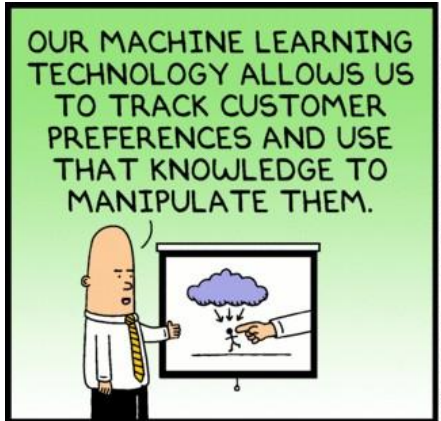
Obrigado!



Dilbert.com DilbertCartoonist@gmail.com



2-2-13 ©2013 Scott Adams, Inc./Dist. by Universal Uclick



Dilbert.com DilbertCartoonist@gmail.com



1-31-13 ©2013 Scott Adams, Inc./Dist. by Universal Uclick



www.dilbert.com scottadams@aol.com



©2000 United Feature Syndicate, Inc.





