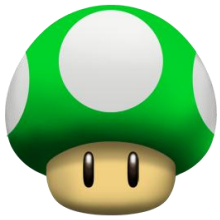


# INF 1771 – Inteligência Artificial

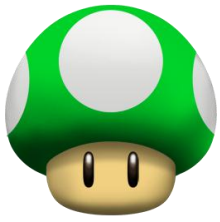
## Aula 14 – Aprendizado de Máquina

Edirlei Soares de Lima  
<elima@inf.puc-rio.br>



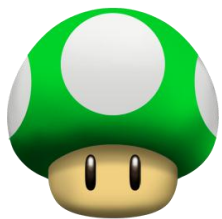
# Agentes Vistos Anteriormente

- ❏ **Agentes baseados em busca:**
  - ❏ Busca cega
  - ❏ Busca heurística
  - ❏ Busca local
  
- ❏ **Agentes baseados em lógica:**
  - ❏ Lógica proposicional
  - ❏ Lógica de primeira ordem
  
- ❏ **Agentes baseados em planejamento:**
  - ❏ Planejamento de ordem parcial
  - ❏ Planejamento em ambientes não-determinísticos
  
- ❏ **Agentes baseados em conhecimento estatístico:**
  - ❏ Redes Bayesianas
  - ❏ Lógica Fuzzy

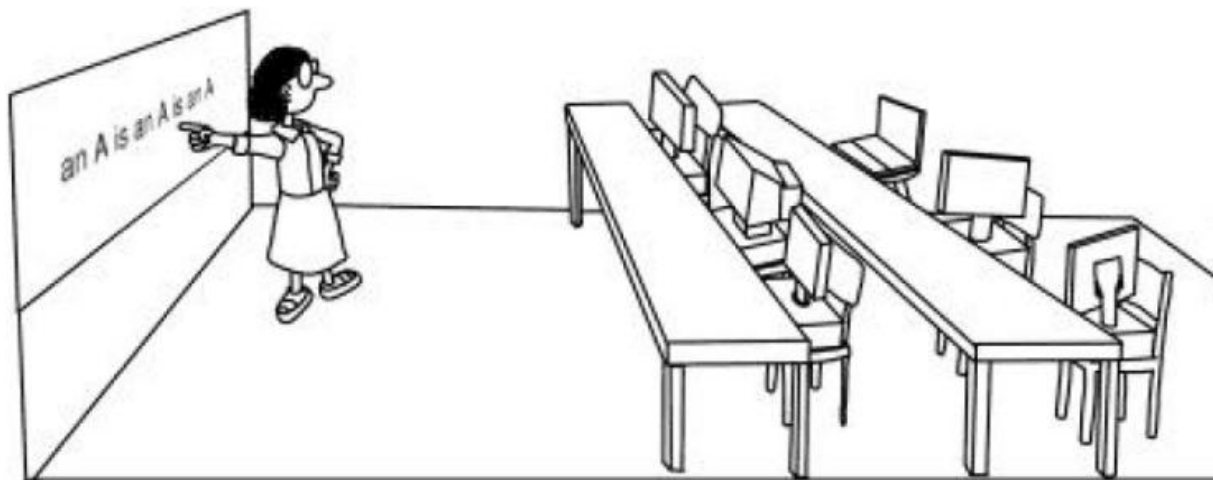


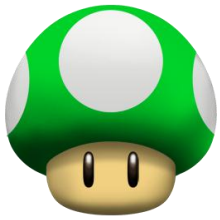
# Introdução

- ❗ **Computadores realmente são capazes de aprender?**
- ❗ Infelizmente ainda não sabemos exatamente como fazer computadores aprender de uma maneira similar a maneira como os **humanos aprendem.**
- ❗ Entretanto, existem **algoritmos** que são eficientes em certos tipos de tarefas de aprendizagem.



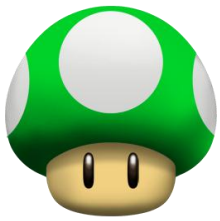
# O que é Aprendizagem de Máquina?





# O que é Aprendizagem de Máquina?

- ❏ Aprender significa “**mudar para fazer melhor**” (de acordo com um dado critério) quando uma situação similar acontecer.
- ❏ Aprendizagem, **não é memorizar**. Qualquer computador pode memorizar, a dificuldade está em **generalizar** um comportamento para uma nova situação.



# Importancia do Aprendizado

## ❏ Por que é importante para um agente aprender?

- ❏ Os programadores não podem antecipar todas as situações que o agente pode encontrar.

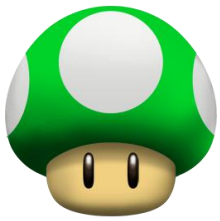
**Exemplo:** Um robô programado para andar em um único labirinto pode não saber andar em outros.

- ❏ Os programadores não podem antecipar todas as mudanças que podem acontecer com o passar do tempo.

**Exemplo:** Agente programado para prever as melhores opção de bolsa para investir precisa se adapta quando o ambiente muda.

- ❏ Os programadores nem sempre sabem encontrar a solução dos problemas diretamente.

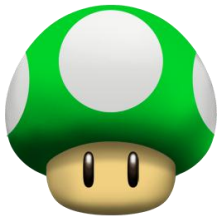
**Exemplo:** Programar um sistema para reconhecer faces não é algo trivial.



# Como Aprender Algo?

## 💡 Exemplos:

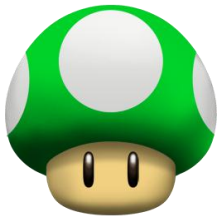
- 💡 Considerando um agente treinando para ser se tornar um motorista de táxi. Toda vez que o instrutor gritar "freio!" o agente pode aprender uma condição de quando ele deve frear.
- 💡 Ao ver várias imagens que contem ônibus, o agente pode aprender a reconhecê-los.
- 💡 Ao tentar ações e observar os resultados. Por exemplo, ao frear forte em uma estrada molhada pode aprender que isso não tem um efeito bom.



# Formas de Aprendizado

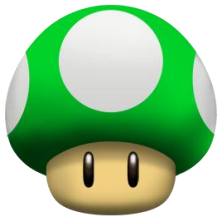
- 💡 Aprendizado Supervisionado
- 💡 Aprendizado Não Supervisionado
- 💡 Aprendizado Por Reforço





# Aprendizado Supervisionado

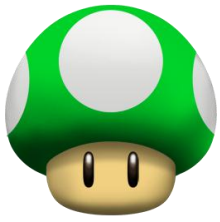
- ❏ Observa-se alguns pares de **exemplos de entrada e saída**, de forma a aprender uma **função que mapeia a entrada para a saída**.
- ❏ Damos ao sistema a “**resposta correta**” durante o processo de treinamento.
- ❏ É eficiente pois o sistema pode trabalhar diretamente com informações corretas.
- ❏ Útil para classificação, regressão, estimação de probabilidade condicional (qual é a probabilidade de um cliente com um determinado perfil comprar um determinado produto?)



# Aprendizado Supervisionado

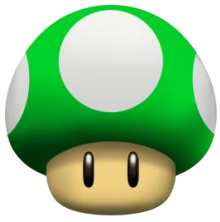
## 📌 Exemplo:

- 📌 Considerando um agente treinando para ser se tornar um motorista de táxi. Toda vez que o instrutor gritar "freio!" o agente pode aprender uma condição de quando ele deve frear.
- 📌 A entrada é formada pelos dados percebidos pelo agente através de sensores. A saída é dada pelo instrutor que diz quando se deve frear, virar a direita, virar a esquerda, etc.



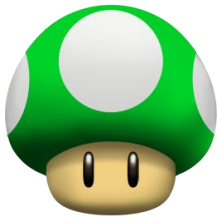
# Aprendizado Não Supervisionado

- ❏ O agente **reconhece padrões nos dados de entrada**, mesmo sem **nenhum feedback de saída**.
- ❏ Por exemplo, um agente aprendendo a dirigir pode gradualmente desenvolver um conceito de dias de bom tráfego e dias de tráfego congestionado mesmo sem nunca ter recebido exemplos rotulados por um professor.



# Aprendizado Por Reforço

- ❏ O agente fica sabendo de uma série de reforços, **recompensas ou punições**.
- ❏ Por exemplo, a falta de uma gorjeta no final do percurso da ao agente de taxista uma indicação de que ele fez algo errado.
- ❏ Cabe ao agente reconhecer qual das ações antes do reforço foram as maiores responsáveis por isso.
- ❏ Não damos a “resposta correta” para o sistema. O sistema faz uma hipótese e determina se essa hipótese foi boa ou ruim.



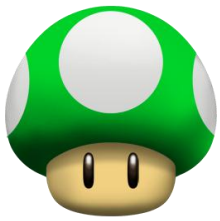
# Fases da Aprendizagem

## 💡 **Treinamento** (supervisionado)

- 💡 Apresenta-se exemplos ao sistema.
- 💡 O sistema “aprende” a partir dos **exemplos**.
- 💡 O sistema modifica gradualmente os seus parâmetros para que a saída se aproxime da saída desejada.

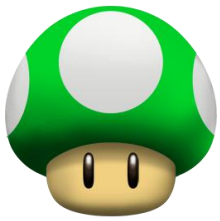
## 💡 **Utilização**

- 💡 Novos exemplos jamais visto são apresentados ao sistema.
- 💡 O sistema deve generalizar e reconhecê-los.



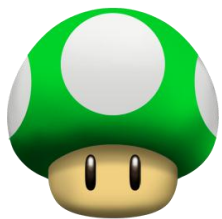
# Exemplos de Treinamento (Aprendizado Supervisionado)

	Atributos						
Exemplo	Atrib <sub>1</sub>	Atrib <sub>2</sub>	Atrib <sub>3</sub>	Atrib <sub>4</sub>	Atrib <sub>5</sub>	Atrib <sub>6</sub>	Classe
<b>X<sub>1</sub></b>	0.24829	0.49713	0.00692	-0.020360	0.429731	-0.2935	<b>1</b>
<b>X<sub>2</sub></b>	0.24816	0.49729	0.00672	0.0065762	0.431444	-0.29384	<b>1</b>
<b>X<sub>3</sub></b>	0.24884	0.49924	0.01047	-0.002901	0.423145	-0.28956	<b>3</b>
<b>X<sub>4</sub></b>	0.24802	0.50013	0.01172	0.001992	0.422416	-0.29092	<b>2</b>
<b>X<sub>5</sub></b>	0.24775	0.49343	0.01729	-0.014341	0.420937	-0.29244	<b>2</b>



# Classificação de Exemplos Desconhecidos

	Atributos						
Exemplo	Atrib <sub>1</sub>	Atrib <sub>2</sub>	Atrib <sub>3</sub>	Atrib <sub>4</sub>	Atrib <sub>5</sub>	Atrib <sub>6</sub>	Classe
<b>X<sub>1</sub></b>	0.22829	0.48713	0.00592	-0.010360	0.419731	-0.2845	?
<b>X<sub>2</sub></b>	0.21816	0.48729	0.00572	0.0045762	0.421444	-0.28484	?
<b>X<sub>3</sub></b>	0.23884	0.49824	0.01447	-0.003901	0.433145	-0.24956	?
<b>X<sub>4</sub></b>	0.23002	0.49013	0.02172	0.002992	0.412416	-0.28092	?
<b>X<sub>5</sub></b>	0.24575	0.49243	0.01029	-0.015341	0.430937	-0.28244	?



# Tipos de Problemas

## ❏ **Classificação:**

- ❏ Responde se uma determinada “entrada” pertence a uma certa classe.
- ❏ Dada a imagem de uma face: de quem é esta face (dentre um número finito).

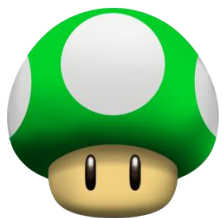
## ❏ **Regressão:**

- ❏ Faz uma predição a partir de exemplos.
- ❏ Predizer o valor da bolsa amanhã, dados os valores de dias e meses anteriores.

## ❏ **Estimação de Densidade:**

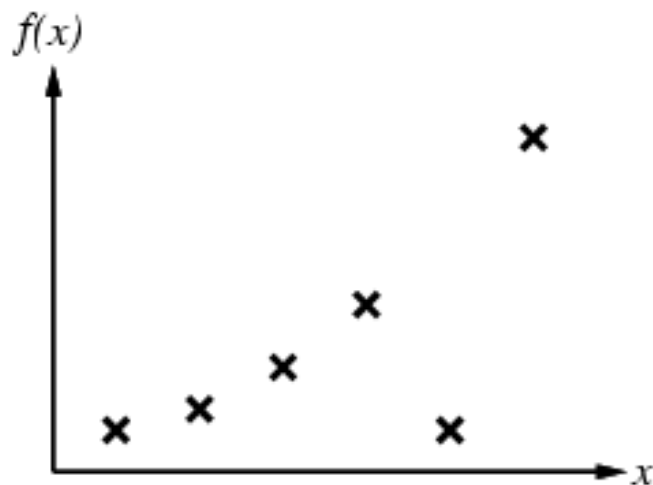
- ❏ Estima quais são as N categorias presente nos dados.





# Aprendizado Supervisionado

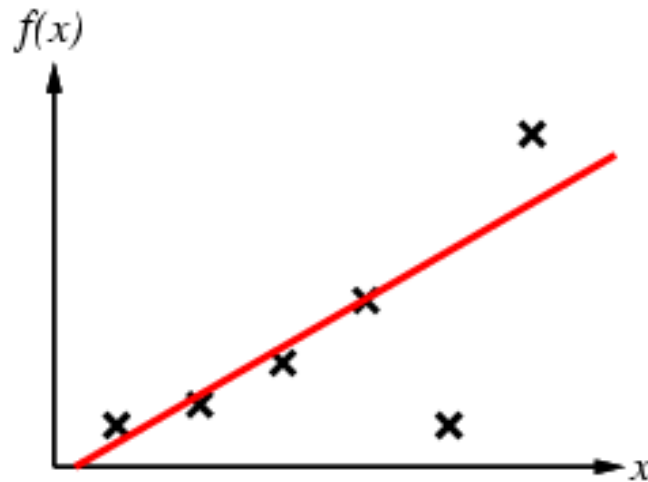
- ❏ Dado uma quantidade finita de dados para o treinamento, temos que derivar uma função  $h$  que se aproxime da verdadeira função  $f(\mathbf{x})$  (a qual gerou os dados e é desconhecida).
- ❏ Existe um número infinito de funções  $h$ .

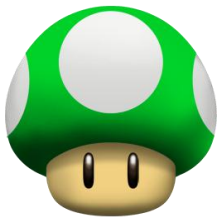




# Aprendizado Supervisionado

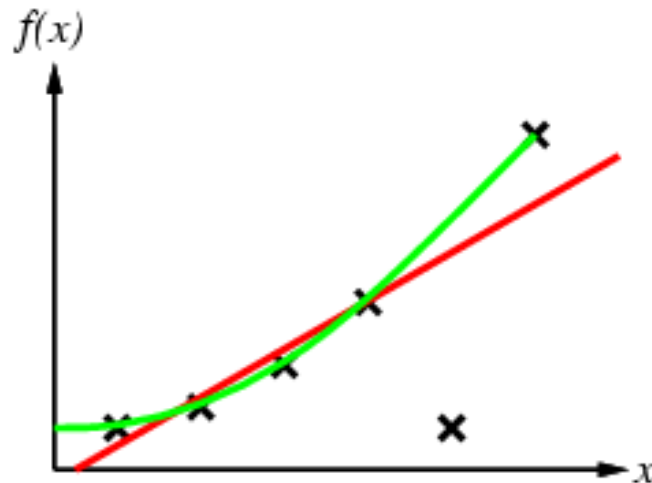
- ❏ Dado uma quantidade finita de dados para o treinamento, temos que derivar uma função  $h$  que se aproxime da verdadeira função  $f(x)$  (a qual gerou os dados e é desconhecida).
- ❏ Existe um número infinito de funções  $h$ .

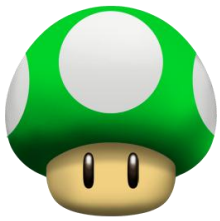




# Aprendizado Supervisionado

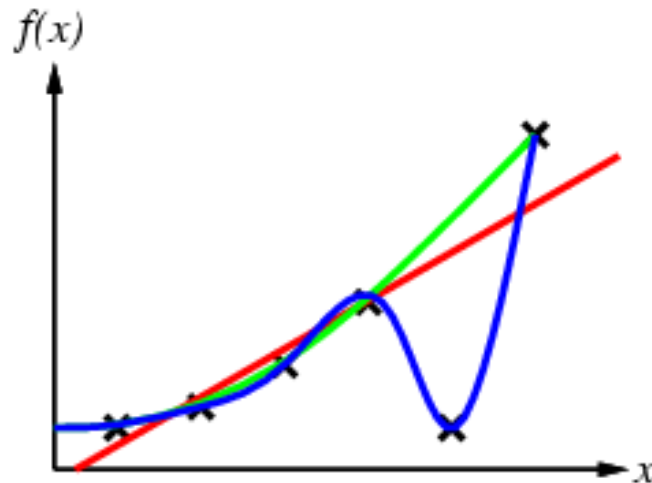
- ❏ Dado uma quantidade finita de dados para o treinamento, temos que derivar uma função  $h$  que se aproxime da verdadeira função  $f(x)$  (a qual gerou os dados e é desconhecida).
- ❏ Existe um número infinito de funções  $h$ .





# Aprendizado Supervisionado

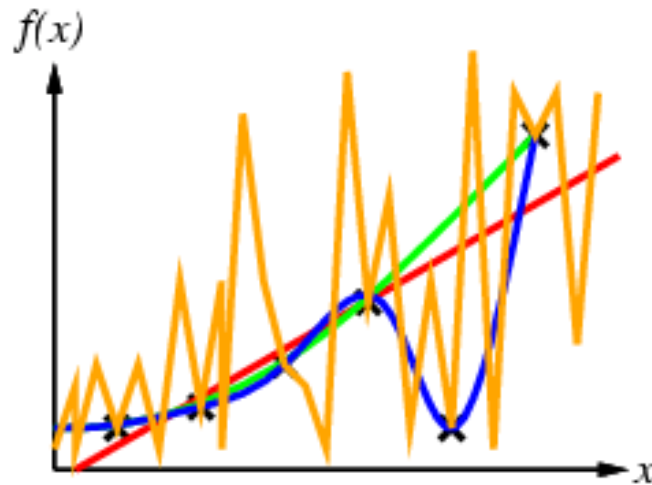
- ❏ Dado uma quantidade finita de dados para o treinamento, temos que derivar uma função  $h$  que se aproxime da verdadeira função  $f(x)$  (a qual gerou os dados e é desconhecida).
- ❏ Existe um número infinito de funções  $h$ .

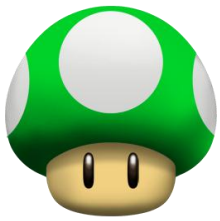




# Aprendizado Supervisionado

- ❏ Dado uma quantidade finita de dados para o treinamento, temos que derivar uma função  $h$  que se aproxime da verdadeira função  $f(x)$  (a qual gerou os dados e é desconhecida).
- ❏ Existe um número infinito de funções  $h$ .





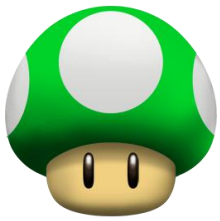
# Generalizar é Difícil

## ❗ Não queremos aprender por memorização

- ❗ Boa resposta sobre os exemplos de treinamento somente.
- ❗ Fácil para um computador.
- ❗ Difícil para os humanos.

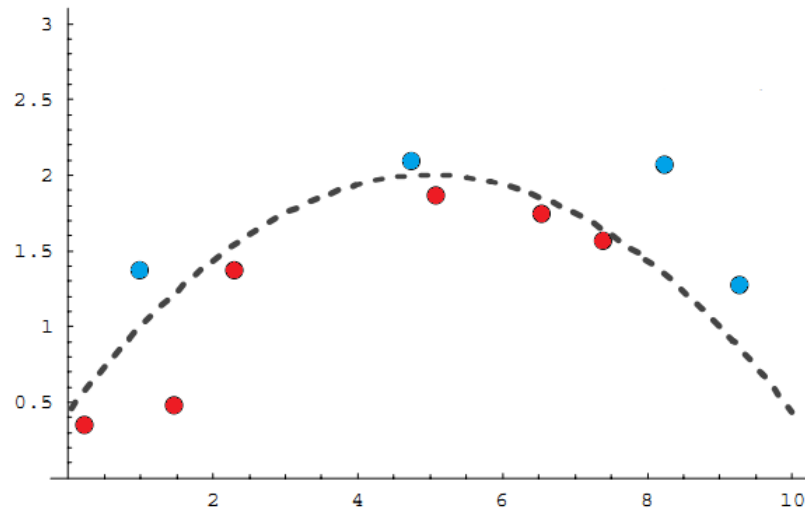
## ❗ Aprender visando generalizar

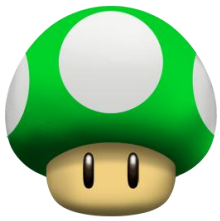
- ❗ Mais interessante.
- ❗ Fundamentalmente mais difícil: diversas maneiras de generalizar.
- ❗ Devemos extrair a essência, a estrutura dos dados e não somente aprender a boa resposta para alguns casos.



# Exemplo

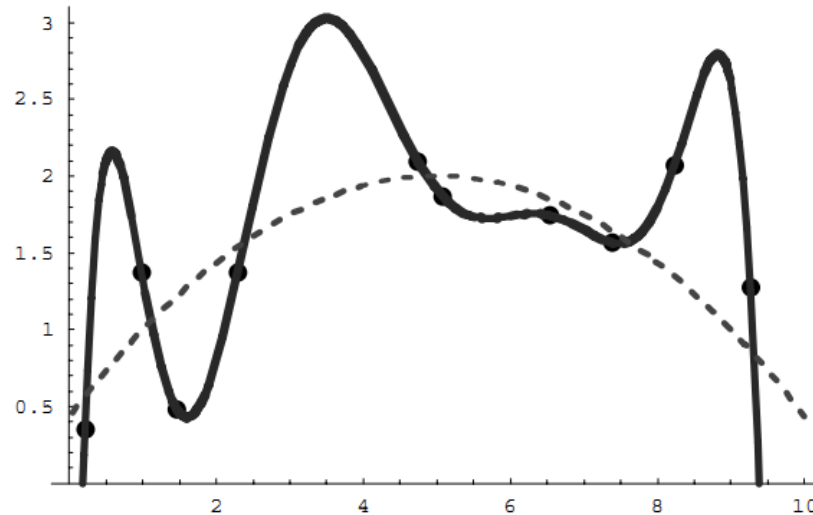
- ❏ Função-alvo  $f$  (melhor resposta possível).



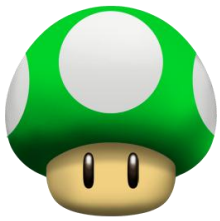


# Exemplo – Overfitting

- ❏ Erro baixo sobre os exemplos de aprendizagem. Mais elevado para os de teste.

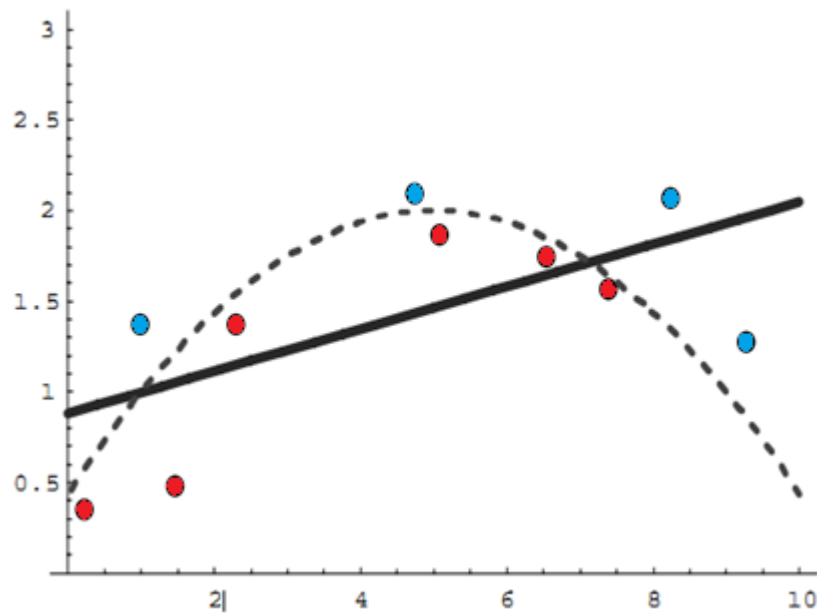






# Exemplo – Underfitting

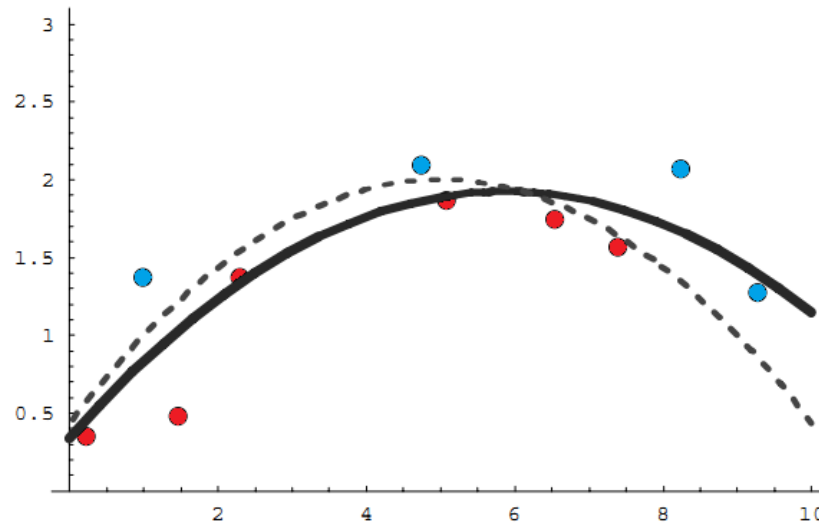
- Escolhemos um modelo muito simples (linear): erro elevado na aprendizagem.

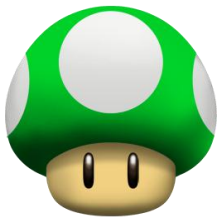




# Exemplo – Um Bom Modelo

- ❏ O modelo é suficientemente flexível para capturar a forma curva da função  $f$  mais não é suficiente para ser exatamente igual a função  $f$ .





# Teoria de Aprendizado Computacional

- ❓ Como sabemos se a hipótese  $h$  está próxima da função-alvo  $f$ , se não conhecemos o que é  $f$ ?
- ❓ Este é um aspecto de uma questão mais abrangente: como saber se um algoritmo de aprendizado produziu uma teoria que preverá corretamente o futuro?
- ❓ Qualquer hipótese que é consistente com um conjunto suficientemente grande de exemplos é pouco provável de estar seriamente errada.